MCS

MICRONIC COMPUTER SYSTEME GMBH

ALPHA 1

BAND 1

MICROCOMPUTER

ALPHA 1

Handbuch I

Erste Auflage Oktober 1977

Copyright by MCS

MCS verfügt über alle Rechte der deutschen und aller fremdsprachigen Ausgaben.
Nachdrucke und Vervielfältigungen, auch auszugsweise, sind nur mit ausdrücklicher Genehmigung von MCS gestattet.

MCS
Micronic Computer Systeme GmbH
Tennstedter Str. 18 D-1000 Berlin 46
Tel.: (030) 711 20 55 Telex 04 0184187

Vorwort

- Per aspera ad astra -

Mit der stürmischen Entwicklung, die in den letzten Jahren auf dem Gebiet der "Microprocessor"-Technik eingesetzt hat, werden interessierte Laien wie Fachleute der Elektronik gleichermaßen vor das Problem gestellt:

Wie kann ich mich in dieses Gebiet einarbeiten?

Auch "wir",die Verfasser dieses Buches, die gleichzeitig die Entwickler des ALPHA 1 und BETA 8 sind, standen einmal vor dieser Frage. Mit diesem Buch und ALPHA 1 wollen wir Ihnen die Erfahrungen und Kenntnisse, die das Verständnis eines Microcomputers ausmachen, in einer Form anbieten, die Ihnen das Lernen erleichtert und Sie gründlich mit dem Gerät vertraut macht. Auch Sie werden Zeit brauchen, um sich einzuarbeiten, doch mit ALPHA 1 haben Sie die Möglichkeit, viele Fehler und Hindernisse, die wir "noch" zu bezwingen hatten, zu vermeiden.

Unser Dank gilt allen Mitarbeitern, die einen grossen Teil ihrer Freizeit zur Verfügung gestellt haben, um das Projekt ALPHA 1 fertigzustellen. Besonderer Dank gilt auch den Familienmitgliedern, Verwandten und Freunden, die hierfür Verständnis hatten.

Berlin im Oktober 1977

Inhaltsverzeichnis

Kapitel	1	Der Microcomputer ALPHA 1	1- 1
Kapitel	2	Inbetriebnahme	2- 1
Kapitel	3	Das ALPHA 1-System	3- 1
	3.1	Aufbau	3- 1
	3.1.1	CPU-Karte	3- 1
	3.1.2	Anzeige und Bedienungselemente	3- 1
	3.2	Applikationsanschluß	3- 2
	3.3	Kommunikationsanschluß	3- 4
	3.4	BUS-Anschluß	3- 6
	3.5	ALPHA 1 Speicherzuordnung	3- 8
Kapitel	4	Bedienung des ALPHA 1	4- 1
	4.1	Tastatur und Anzeige	4- 1
	4.2	Datenterminal	4- 1
	4.3	Magnetbandrecorder	4- 4
Kapitel	5	Erweiterung des Systems	5- 1
Kapitel	6	Die Programmierung des ALPHA 1	6- 1
	6.1	Einführung	6- 1
	6.2	Einige Tastenfunktionen	6- 2
	6.3	Hexadezimalzahlen	6- 3
	6.4	ROM-Speicher	6- 4
	6.5	Miniprogramm	6- 4
	6.6	Programmtest mit SINGLE oder SLOWSTEP	6- 6
	6.7	Weitere Tastenfunktionen	6- 6
Kapitel	7	Benutzung eines Magnetband-Speichers	7- 1
	7.1	Abgleich	7- 3
Kapitel	8	Benutzung eines Datenterminals	8- 1
	8.1	Bedienung (Zuordnung der Tasten)	8- 1
Kapitel	9	Struktur des Betriebssystems MONA	9- 1
	9.1	Datenformat der Magnetbandroutine	9- 5
	9.2	Datenformat der Lochstreifenroutine	9- 7
Kapitel	10	Spiel-und Testprogramme	10- 1
	10.1	MONA's Uhr	10- 1
	10.2	Weitere Programme (mitgelieferte Cassette)	10- 2
Kapitel	11	Technischer Anhang	11- 1
	11.1	Listing des Betriebssystems MONA	11- 2
	11.2	Kurzanleitung Betriebssystem MONA	11-26
	11.3	Anschlußzuordnung 6502, 6532, 2708, 2716, (32 x 8 PROM)	11-31
	11.4	MCS-BUS	11-36
	11.5	Zuordnung 7-Segment Alphabet	11-37
	11.6	Hinweise auf Band 2	11-38

Kapitel 1

Der Microcomputer ALPHA 1

Die Firma MCS beglückwünscht Sie zu Ihrer Entscheidung in der Wahl des Microcomputers ALPHA 1. Mit diesem Gerät besitzen Sie einen nach dem letzten Stand der Technik aufgebauten Microcomputer, der sich sowohl als Lehr- wie auch als Entwicklungshilfsgerät einsetzen läßt. Das nach kommerziellen Maßstäben entwickelte Gerät wird Ihnen in Verbindung mit den technischen Unterlagen die Möglichkeit bieten, sich in die modernste Technik der Elektronik einzuarbeiten.

Der ALPHA 1 baut sich auf die Struktur des industriellen ebenfalls von MCS hergestellten BETA 8-Systems auf. Er ist zu diesem aufwärts kompatibel. Entsprechend dieser Konzeption befindet sich die CPU-Platine steckbar als Europakarte auf Ihrem ALPHA 1-Microcomputer. Diese Tatsache gestattet es Ihnen, später, wenn notwendig, zu dem BETA 8-System überzugehen. Das Herzstück dieser CPU-Karte ist der Microprozessor vom Typ 6502.

Außer der CPU-Karte benötigt der Microcomputer noch weitere umfangreiche Hardware, diese befindet sich in dem pultförmigen Gehäuse.

Erst in Verbindung mit dem Vorhandensein von Software entsteht der Begriff Computer. Die residente Software, von MCS entwickelt, befindet sich in einem ROM-Speicherbaustein, der das Betriebssystem MONA enthält und sich ebenfalls in dem Pultgehäuse befindet. Erst dieses Betriebssystem gestattet es Ihnen, die verschiedensten Betriebsarten des Microcomputers ALPHA 1 in Betrieb zu nehmen.

Es ist selbstverständlich, daß Ihr ALPHA 1 über einen frei programmierbaren Speicher für Daten und eigene Programme verfügt. Dieser Typ von Speicher wird RAM genannt. Darüber hinaus verfügt Ihr System noch über einen "Timer" und über 16 Eingangs- bzw. Ausgangsanschlüsse. Der "Timer" wie auch die Eingangs- bzw. Ausgangsanschlüsse sind frei programmierbar. Mit den letztgenannten können

Sie externe digitale Steuerungsprobleme lösen.

Da Ihr ALPHA 1 aufwärts kompatibel zu dem BETA 8-System ist, ergeben sich bei einer späteren Erweiterung keinerlei Interfaceprobleme. Dies wird erreicht durch eine konsequente Bus-Struktur und durch die Verwendung genormter Europakarten. (Es wird bei diesem System und dessen Erweiterung kein Lötkolben für irgendwelche Hardwarearbeiten benötigt!!!) Die CPU-Karte beinhaltet bereits u.a. jetzt schon sämtliche Daten und Adresstreiber, die es gestatten, das System praktisch beliebig zu erweitern.

Zur Inbetriebnahme Ihres ALPHA 1 gibt es zwei Möglichkeiten: Je nach entsprechenden Vorkenntnissen ist es ratsam, die Kapitel 1 bis 5 zu lesen oder aber bei Kapitel 6 zu beginnen.

Dieses Handbuch kann Ihnen nicht alles über die Hardware und über die Programmierung des Microprocessors 6502 sagen. Ja nach persönlicher Aufgabenstellung wird es deshalb notwendig sein, sich später intensiver mit der Hardware bzw. Software zu beschäftigen. Für diese Zwecke fügen wir zwei Originalbücher des Herstellers des Microprocessors 6502 bei. Für die Richtigkeit der Angaben in diesen Büchern kann MCS keine Gewähr übernehmen. (Der Ersthersteller des Microprocessors 6502 war die Firma MOS-Technology. Weitere Hersteller des Processors sind die Firmen Rockwell und die Firma Synertek).

Kapitel 2

Inbetriebnahme

Die Lieferung unseres ALPHA 1 Microcomputers setzt sich zusammen aus folgenden Elementen:

1 Stück ALPHA 1
1 " CPU-Karte M 622
1 " Netzkabel
1 " Daten-Cassette
1 " DIN-A5-Ordner mit folgendem Inhalt:
Handbuch Band I, Hardware Manual, Programming Manual, Instruktionskarte, Datenblätter 6502, 6532.

Damit Ihr Minicomputer funktionstüchtig wird, ist es notwendig, daß Sie die CPU-Karte auf das Pultgehäuse stecken.

!Ein wichtiger Hinweis! Der Microprocessor 6502 ist ein MOS-Schaltkreis, obwohl alle Ein- und Ausgänge dieses Schaltkreises gegen statische Aufladung geschützt sind, sollten hohe Spannungen ferngehalten werden!

Hohe statische Aufladungen Ihres Körpers können in trockenen Räumen entstehen, insbesondere dann, wenn diese mit Teppichböden ausgelegt sind. Um Schäden an Ihrem Processor zu vermeiden, sollten Sie kurz vor dem Anfassen der CPU-Karte ein Erdpotential (Wasserleitung, Heizung) berühren.

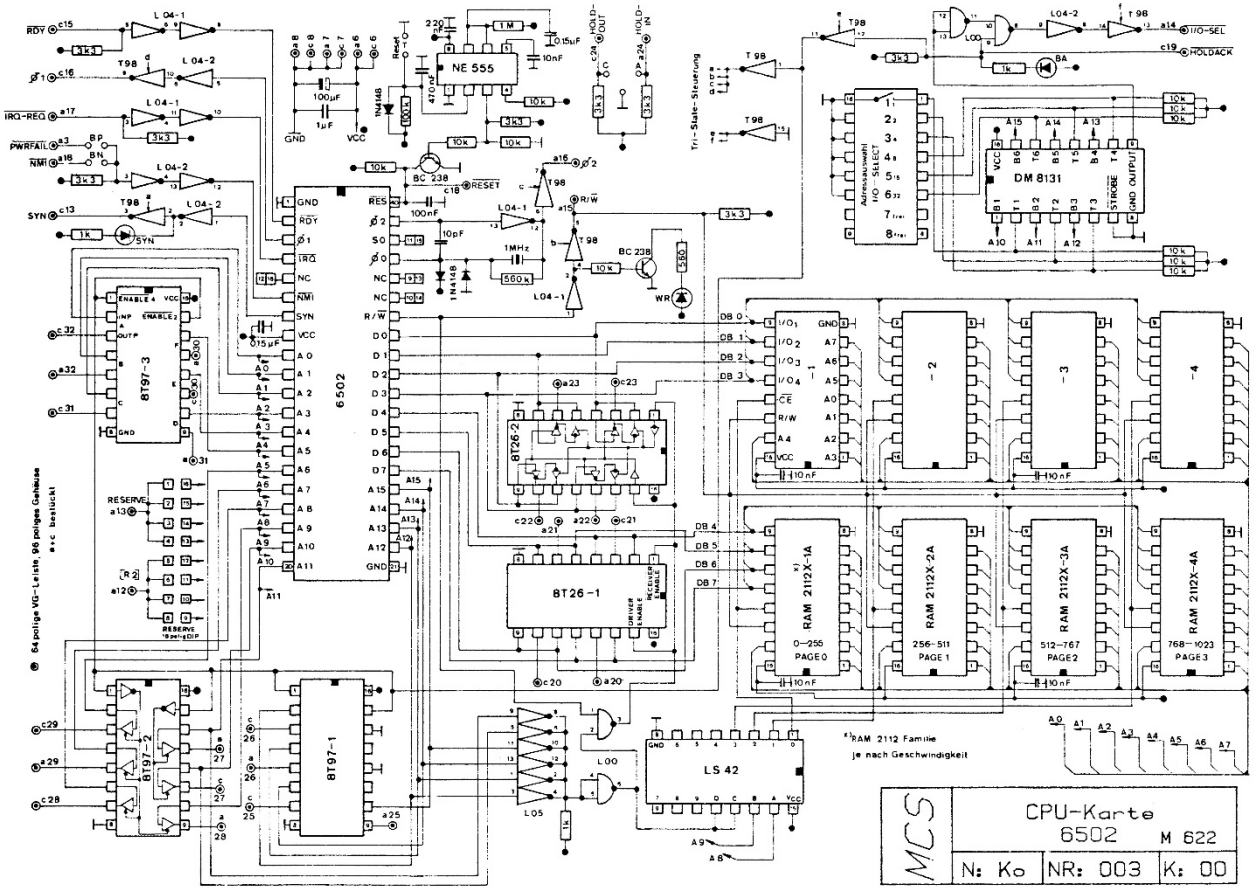
Nachdem Sie dies getan haben, setzen Sie bitte die CPU-Karte auf. Die Bauelemente dieser Karte müssen Ihnen zugewandt sein. Achten Sie bitte darauf, daß die Karte einwandfrei auf dem dafür vorgesehenen Stecker sitzt.

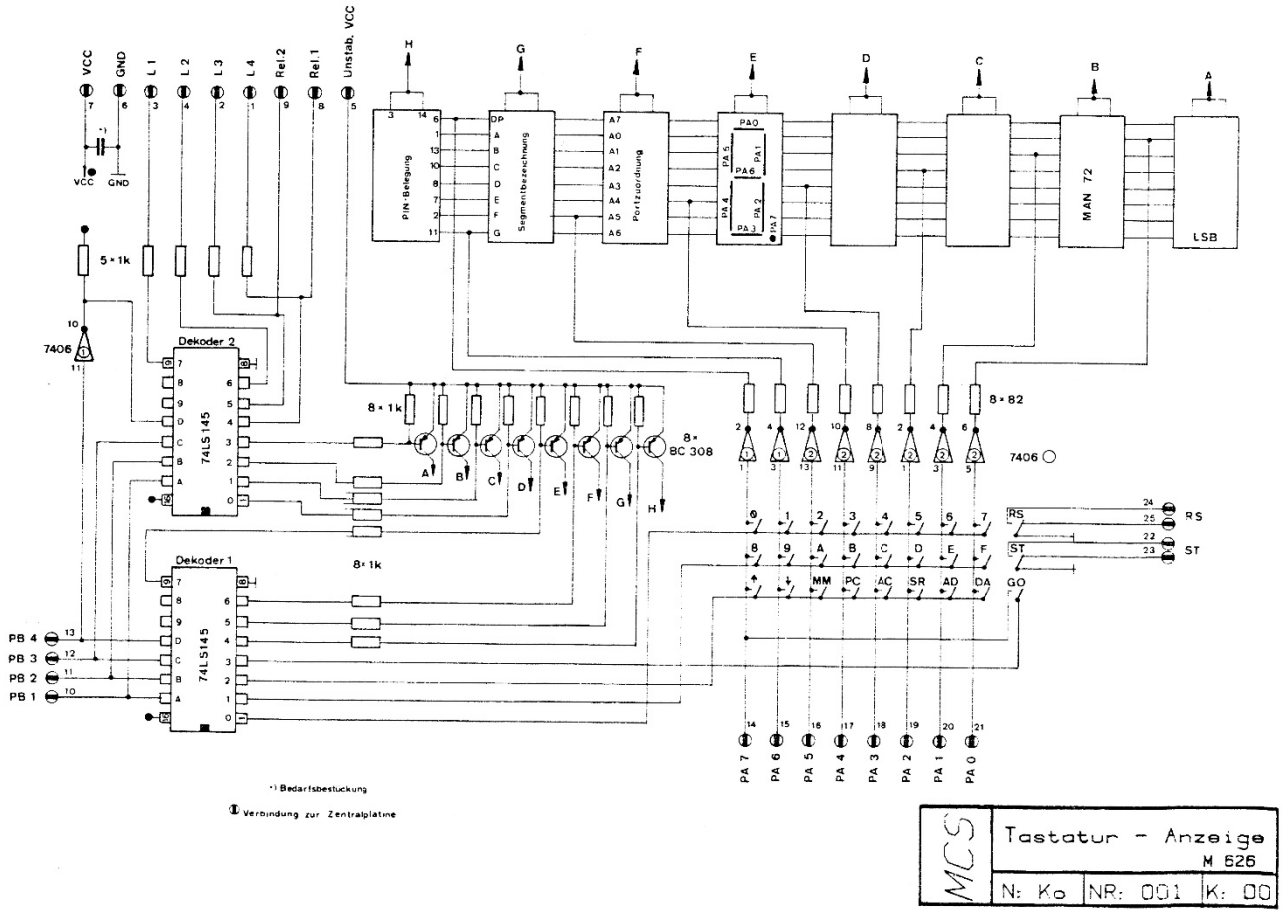
Nachdem dies geschehen ist, können Sie nun Ihren ALPHA 1 an die Netzspannung (220 Volt 50 Hz) anschließen. An der Rückseite des Gerätes befindet sich der Netzschalter Ihres Gerätes. Das Gerät ist eingeschaltet, wenn sich der Schalter "oben" befindet. Für

einen ersten Funktionstest müssen sich die beiden Schalter auf der Frontplatte in den Stellungen "N" (normal) und Key (Keybord= Tastatur) befinden.

Nach dem Einschalten des Gerätes erscheint in der Anzeige eine sechsstellige Information. Damit ist Ihr Gerät betriebsbereit. Die funktionierende Anzeige gewährleistet, daß alle wesentlichen Teile einwandfrei arbeiten.

Die Rückwand Ihres ALPHA 1 ist als Kühlfläche ausgebildet. Bitte beachten Sie, daß die Temperatur dort nach längerer Betriebsdauer etwa 50° C erreichen darf.

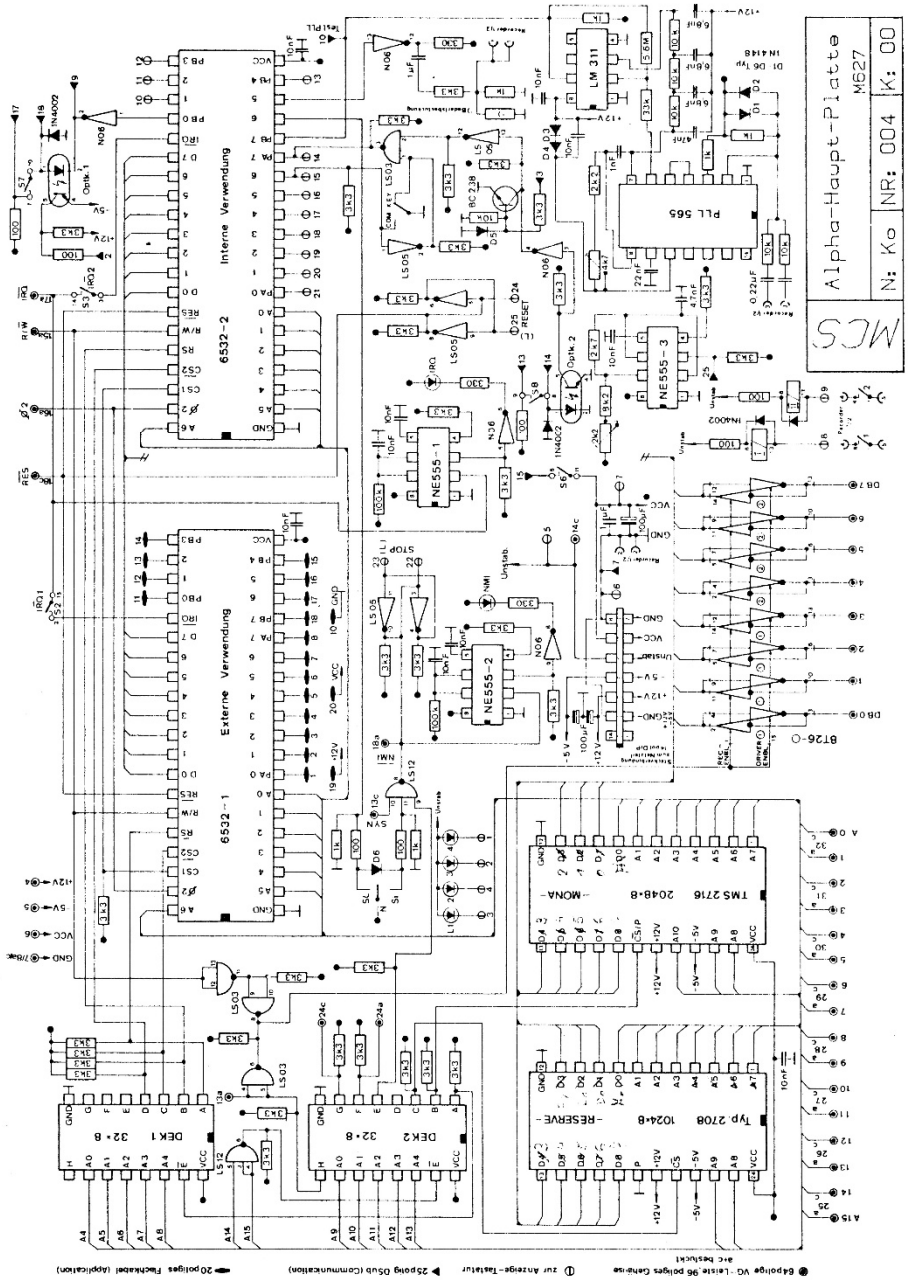

6502
NE 555
DM 8131
LS 42
8T97-3
8T97-2
8T97-1
8T26-2
8T26-1
RAM 2112X-1A
RAM 2112X-2A
RAM 2112X-3A
RAM 2112X-4A
0–255 PAGE0
256–511 PAGE1
512–767 PAGE2
768–1023 PAGE3
*) RAM 2112 Familie je nach Geschwindigkeit
MCS
CPU-Karte 6502 M 622
N: Ko
NR: 003
K: 00

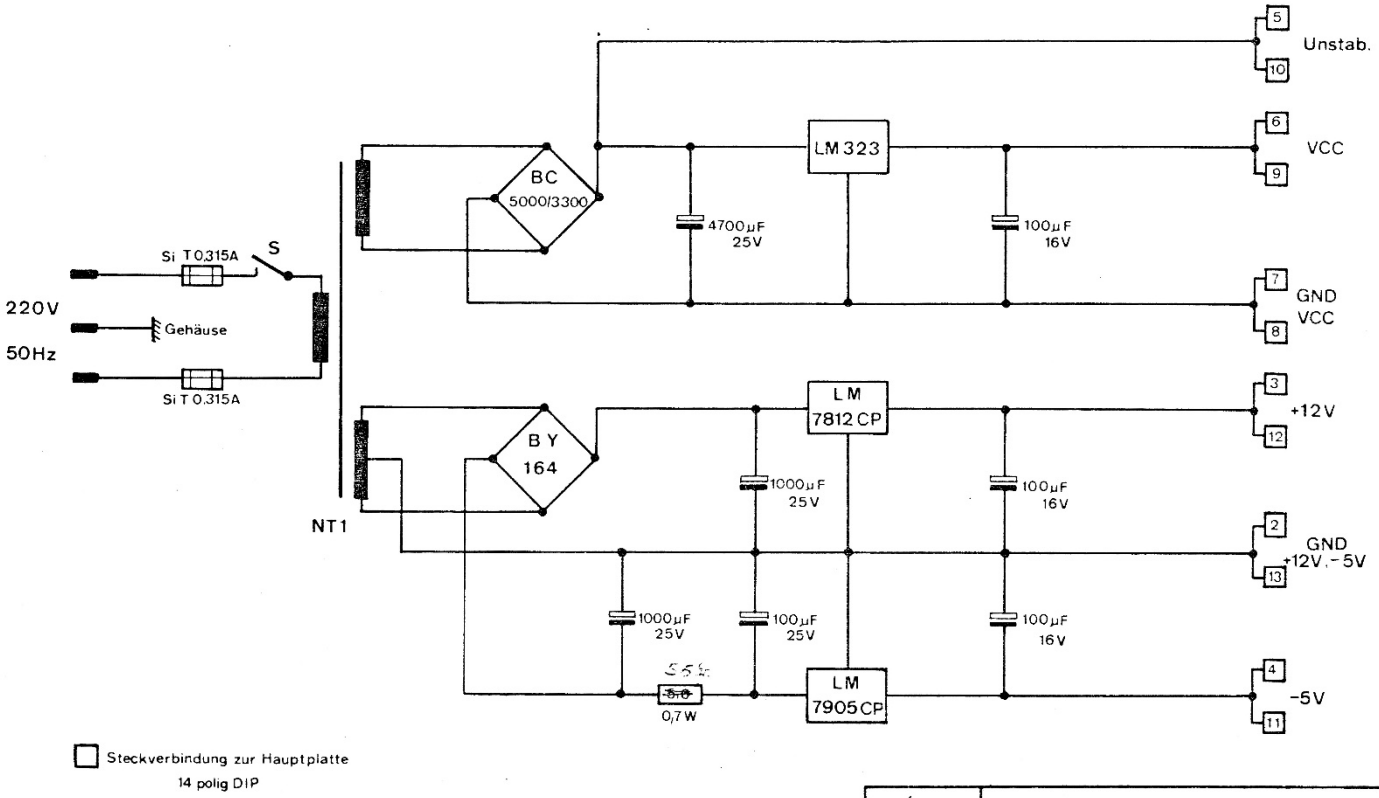


*) Bedarfsbestückung

Verbindung zur Zentralplatine

MCS	Tastatur - Anzeige M 626		
	N: Ko	NR: 001	K: 00


Alpha-Haupt-Platte
M627
N: Ko
NR: 004
K: 00
SJW
6532-2
Interne Verwendung
6532-1
Externe Verwendung
NE555-1
NE555-2
NE555-3
LM 311
PLL 565
TMS 2716 2048*8 -MONA-
Typ 2708 1024*8 -RESERVE-
DEK1 32*8
DEK2 32*8
64polige VG-Leiste 96 poliges Centronic
zur Anzeige-Tastatur
25polig DSub (Communication)
20polliges Flachkabel (Application)



MCS	Netzteil Alpha 1 M625		
	N: Ko	NR: 006	K: 00

Kapitel 3

Das ALPHA 1-System

Nachdem Sie sich mit der ersten Inbetriebnahme von der grundsätzlichen Funktionstüchtigkeit überzeugen konnten, erfolgt nun die Hardware-Beschreibung des Systems, unterstützt durch Schaltbilder und Diagramme.

3.1 Aufbau

Entsprechend dem modularen Aufbau des Systems gliedert sich die Beschreibung in die Funktion der CPU-Karte und in die Funktion des Anzeige- und Bedienungspultes. Bild 1 gibt das Schaltbild der CPU-Platine wieder. Die Abbildungen 2 und 3 geben das Schaltbild der Anzeige und Bedienungseinheit wieder. Das Schaltbild 4 stellt das Netzteil dar.

3.1.1 CPU-Karte

Auf der CPU-Karte befindet sich als zentrales Bauelement der Microprocessor 6502. Dieser 8-Bit Processor ist mit den übrigen Systemkomponenten durch drei voneinander unabhängige Busleitungen verbunden. Der 16-Bit Adress-Bus gestattet den Zugriff auf einen Speicherbereich von 64 K-Bytes. Über einen bidirektionalen 8-Bit Daten-Bus werden die Daten und Befehle von dem Processor zu jeder Speicherzelle und von dort zurück übertragen. Der dritte Bus dient zur Steuerung des Systems.

Außer dem zentralen Steuerelement befinden sich noch auf der Karte die sogenannten Daten- und Adresstreiber, der 1K-Byte große Ramspeicher und entsprechende Logik für weitere Steuerungs-Funktionen des Systems. Als frequenzbestimmendes Element für den Takt des Microprocessors ist ein 1MHz-Quarz eingesetzt.

3.1.2 Anzeige und Bedienungselemente

Die Anzeige und Bedienungselemente sind auf zwei getrennten Platinen untergebracht. Die Anzeige und Tastatur befindet

sich auf der Platine-Nr. M626 siehe Schaltbild 2. Die restlichen Bedienungselemente sind auf der Zentralplatine (Nr. M627) s. Schaltbild 3. Auf dieser letztgenannten Platine sind die zentralen Bauelemente die Peripheriebausteine 6532 (2 Stück) und der Festwertspeicher TMS 2716. Jeder dieser Peripheriebausteine verfügt über 16 freiprogrammierbare Eingabe/Ausgabe-Leitungen, einem Timer und einem 64-Byte RAM. In Verbindung mit dem in dem Speicherbaustein TMS 2716 gespeicherten Betriebssystem - MONA - wird ein Peripheriebaustein 6532 für die interne Steuerung Ihres ALPHA 1 benötigt.

Es folgt eine Erklärung der Funktion der verschiedenen Kontroll- und Bedienungselemente. (Siehe auch hierzu Abb. 5). Das Gerät hat zwei Betriebsartenschalter. Der eine Schalter wählt aus, ob mit der eigenen Tastatur und Anzeige gearbeitet wird (KEY) oder ob mit einem externen Dateneingabegerät gearbeitet werden soll (Stellung COM). Der andere Betriebsartenschalter hat 3 Stellungen. Die mittlere Stellung des Schalters (N) ist die Normalbetriebsart. In dieser Stellung werden Programme eingegeben und gestartet. Die beiden anderen Betriebsarten des Schalters bezeichnet mit(SI) für Single Step und (SL) für Slow Step dienen zum Testen von Programmen. In beiden Betriebsarten läuft ein Programm nicht mehr automatisch ab, sondern es wird schrittweise abgearbeitet. Weitere Erklärungen hierzu im Kapitel 6.6.

3.2 Applikationsanschluß

An diesen 20poligen Flachkabelstecker sind die 16 frei adressierbaren Ein- bzw. Ausgangsleitungen des Peripheriebausteins 6532-1 angeschlossen. Außer diesen Anschlüssen liegen an dieser Steckverbindung eine Betriebsspannung von +12V, eine Betriebsspannung von 5V und die gemeinsame Masse des Systems. Dadurch wird der Anwender in die Lage versetzt, sich kleine Interface-Hilfsschaltungen aufzubauen, die er noch direkt aus diesem System betreiben kann.

APPLIKATIONSANSCHLUSS

PIN	BELEGUNG	BEM.
1	PA0	
2	PA1	
3	PA2	
4	PA3	
5	PA4	
6	PA5	
7	PA6	
8	PA7	
9		
10	GND	
11	PB0	
12	PB1	
13	PB2	
14	PB3	
15	PB4	
16	PB5	
17	PB6	
18	PB7	
19	+12V	
20	Vcc (+5V)	

▼ zeigt auf P1 - die rote Leitung führt an P1

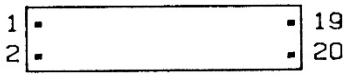


Bild 6

Die Strombelastbarkeit der 12V-Spannung beträgt maximal 25 mA, die der 5V-Spannung maximal 150 mA. Die Kontaktzuordnung dieser Steckverbindung nebst weiteren technischen Daten sind in Abbildung 6 enthalten.

3.3 Kommunikationsanschluß

Der Kommunikationsanschluß bietet sehr viele Möglichkeiten, um in Verbindung mit externen Geräten das ALPHA 1-System zu steuern. Um die verschiedenen Betriebsarten dieses Anschlusses benutzen zu können, muß in jedem Fall der Betriebsartenschalter in Stellung COM (Communication) gebracht werden.

Folgende Geräte können an den Kommunikationsstecker angeschlossen werden:

a) Datenterminals. Hierunter werden Geräte verstanden, die mit einer ASCII-Tastatur ausgerüstet sind und meist einen Bildsichtschirm haben. Diese Geräte haben überwiegend eine V24-Schnittstelle, über welche die empfangenen und gesendeten Daten übertragen werden. Die Anschlußzuordnung in dieser Betriebsart ist ebenfalls genormt. Um diese Art von Geräte an das ALPHA 1-System anschließen zu können, ist das ALPHA 1-System mit einer V24-Schnittstelle ausgerüstet.

b) Betrieb mit TTY. Die Teletype (TTY) ist ein auf dem Weltmarkt sehr verbreitetes Gerät bestehend aus einer Tastatur (ASCII), einem mechanischen Drucker und evtl. einem Lochstreifen-Stanz- und Lesegerät. Die Schnittstelle dieses Gerätes ist eine Stromschnittstelle. Um diesen Gerätetyp zu betreiben, ist das ALPHA 1-System mit einer aktiven Stromschnittstelle (20 mA bzw. 60 mA) ausgerüstet.

c) Um Lochstreifen einzulesen, kann Ihr ALPHA 1 mit einem kleinen Lochstreifenlesegerät ausgerüstet werden, welches eine direkte "Handeingabe" des Lochstreifens ge-

KOMMUNIKATIONSANSCHLUSS

PIN	BELEGUNG	BEM.
1		
2	V24 Ausgang	
3	V24 und TTL Eingang	
4		
5		
6		
7	GND	
8		
9	Ausgang TTL	
10	Test PLL	
11		
12		
13	Eingang Stromschnittstelle	aktiv (+)
14	Eingang Stromschnittstelle	aktiv (-)
15	Vcc (+5V) (geschaltet ü. S6)	
16		
17	Ausgang Stromschnittstelle	aktiv (+)
18	Ausgang Stromschnittstelle	aktiv (-)
19		
20		
21		
22		
23		
24		
25	Takt Ausgang	19,2KHz

Stecker - Sicht auf die Steckseite

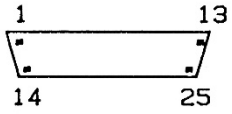


Bild 7

stattet. Die elektrischen Voraussetzungen zum Betrieb eines solchen Gerätes sind im ALPHA 1 vorhanden. Geräte dieser Art sind auf dem Markt verfügbar.

d) Bei vielen Versuchsaufbauten wird ein TTL-Pegel benötigt. Auch diese Schnittstelle befindet sich am Kommunikationsstecker.

Für Testzwecke befindet sich noch an dieser Steckverbindung die Betriebsspannung (V_{CC}) und der Ausgang des PLL-Kreises, der zum Einlesen von Tonbanddaten dient.

Bestimmte Betriebsarten schließen sich untereinander aus. Um mit möglichst wenigen Bauelementen auszukommen, ist auf der Hauptplatine des ALPHA 1 durch DIL-Schalter bzw. durch Kontaktbrücken die Möglichkeit geschaffen worden, diese verschiedenartigen Betriebsarten einzustellen. Bei Anschluß eines externen Gerätes informieren Sie sich vorher gründlich über dessen Ein- bzw. Ausgangsschnittstelle. Insbesondere ist bei Benutzung der Stromschnittstelle auf die entsprechende Polarität zu achten bzw. darauf, ob der Empfänger bzw. Sender aktiv oder passiv ist. Die Kontaktzuordnung des Kommunikationssteckers finden Sie auf der Abb. 7. Die Zuordnung der Auswahlschalter für die verschiedenen Möglichkeiten der Kommunikation ist in Abb. 8 erklärt.

3.4 Bus-Anschluß

Der Bus-Anschluß ist die interne Schnittstelle zwischen der CPU-Karte und den im Pult befindlichen Bauelementen. Wie schon bereits erwähnt, basiert der ALPHA 1 auf dem Bus-orientierten Microprocessorsystem BETA 8. Bis auf wenige Ausnahmen sind sowohl der CPU- wie auch der Pultstecker hinsichtlich der Definition in den Kontaktzuordnungen identisch.

An den Kontakten des Steckverbinders, der sich im Pult befindet, liegen an den PINS 24A und 24C zwei voneinander unabhängige Reserveadressen. An PIN 13A erscheint ein Low-Signal,

wenn der Adressbereich der inneren ALPHA-Steuerung angesprochen wird. PIN 14C ist mit der unstabilisierten V_{CC}-Spannung belegt. Alle anderen Anschlüsse sind, soweit sie angeschlossen sind, mit dem MCS-Bus identisch.

3.5 ALPHA 1 Speicherzuordnung

Der im ALPHA 1 eingesetzte Microprocessor 6502 gestattet die Adressierung von max. 64K-Byte. Das ALPHA 1-System nutzt hiervon nur einen geringen Teil aus. Jeder Adresse entspricht ein 8Bit-Daten- bzw. Befehlswort.

In ALPHA 1-System werden vier verschiedene Baugruppen von den Adressen erreicht:

1.) Die ROM-Einheit. In dieser ist das Betriebssystem MONA gespeichert. Es besteht weiterhin die Möglichkeit, vom Anwender ein Programm in der Größe von max. 1K-Byte dort zu speichern.

2.) Die RAM-Einheit. Diese dient zum Zwischenspeichern von Daten bzw. der Entwicklung von Anwenderprogrammen.

3.) Die I/O PORT-Einheit. Diese enthält verschiedene Register, die es gestatten, externe Peripherie-Einheiten sowohl als Eingang wie auch als Ausgang, an den ALPHA 1-Microcomputer anzuschließen.

4.) Die Timer-Einheit. Diese gestattet es auf einfache Art Zeitverzögerungen zu erzeugen.

I/O Ports und Timer werden über den Datenbus gesteuert.

In Bild 9 ist die Speicherbelegung des ALPHA 1 graphisch dargestellt. Für ein besseres Verständnis unterteilt man den gesamten Speicherbereich von 64K in Blöcke zu je 1K (1024 Speicherzellen). Jeder dieser Blöcke wird dann nochmal in vier gleiche "Seiten" zu je 256 Speicherzellen aufgeteilt.

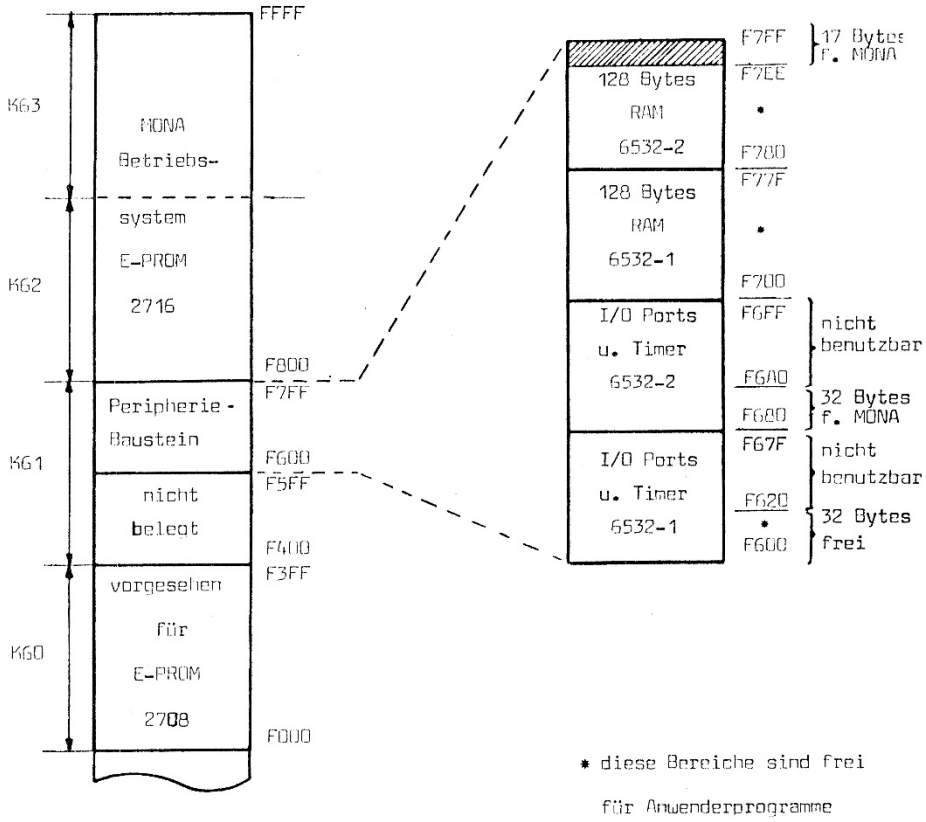
Das Betriebssystem MONA befindet sich in den Speicherblöcken K63 und K62. Die obere Hälfte des Speicherblocks K61 (512 Byte) wird benutzt zur Adressierung der beidenPeripherie-Bausteine 6532, in denen sich auch die Timer befinden. Die genaue Adreßzuordnung innerhalb dieses Bereiches ist vergrößert dargestellt. Der Decoder 1 führt auf der Hauptplatine eine Decodierung in Stufen zu 128 Byte durch. Dadurch ergeben sich innerhalb des Adreßbereiches für die I/O PORT's und Timer jeweils Adreßlücken,

die nicht benutzt werden. Die untere Hälfte des Bereiches K61 ist nicht belegt.

Der Bereich K6Ø ist decodiert für ein vorgesehenes EPROM. An dieser Stelle kann der Benutzer eigene Programme resident ablegen. Zusammenfassend sieht man, daß die oberen 4K des möglichen Adreßbereiches des Microprozessors durch das ALPHA 1-System belegt sind.

Für die Funktion des Processors ist es notwendig, daß im unteren KØ-Bereich ein RAM-Speicher zur Verfügung steht. Auf der CPU-Karte befindet sich hierfür 1K RAM. Innerhalb dieses Bereiches sind die Seite "O" und Seite "1" von besonderer Bedeutung, dies ist vergrößert dargestellt. Ein wichtiger Hinweis: es dürfen bei eigenen Programmen nicht diejenigen Speicherzellen benutzt werden, die für das Betriebssystem MONA reserviert sind. Ein besonderer Hinweis gilt der Seite "1". Diese Seite ist für den Stack reserviert. D.h. der Processor legt auf dieser RAM-Seite unter bestimmten Bedingungen Informationen ab. Die Organisation hierbei ist so, daß der Processor automatisch diesen RAM-Bereich vom Ende der Seite "1" nach unten belegt. Es ist deshalb nicht erlaubt, Anwenderprogramme in diesen Bereich des RAM s abzuspeichern, sofern man sich nicht davon überzeugt hat, daß hierbei der obere Bereich des Stacks überschrieben werden kann. Das eigene Betriebssystem benutzt nur wenige Stack-Adressen.

Eine Zusammenstellung wichtiger Adressen finden Sie unter Kapitel 9.


FFFF
K63
MONA
Betriebs-
system
E-PROM
2716
K62
F800
F7FF
Peripherie-
Baustein
K61
F600
F5FF
nicht
beleat
F400
F3FF
vorgesehen
für
E-PROM
2708
K60
F000
128 Bytes
RAM
6532-2
128 Bytes
RAM
6532-1
I/O Ports
u. Timer
6532-2
I/O Ports
u. Timer
6532-1
F7FF
F7EE
F780
F77F
F700
F6FF
F6A0
F680
F67F
F620
F600
17 Bytes
f. MONA
nicht
benutzbar
32 Bytes
f. MONA
nicht
benutzbar
32 Bytes
frei
* diese Bereiche sind frei
für Anwenderprogramme

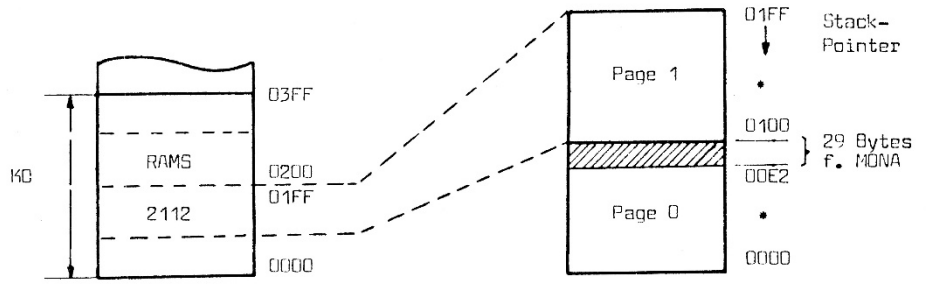

03FF
K0
RAMS
2112
0200
01FF
0000
Page 1
Page 0
01FF
Stack-
Pointer
0100
00E2
29 Bytes
f. MONA
0000

Bild 9

Kapitel 4

Bedienung des ALPHA 1

Nach den grundsätzlichen Erklärungen zum Hardware-Aufbau sollen nun die Bedienungselemente und der Betrieb mit externen Geräten erklärt werden. Als Unterstützung hierfür dient Bild 5 Sie können dort alle wichtigen Elemente sehen.

4.1 Tastatur und Anzeige

Die Tastatur des ALPHA wird vom Betriebssystem MONA kontrolliert. Die Tasten Ø - F gestatten die Eingabe von Hexadezimalzahlen. Die übrigen Tasten dienen der Steuerung des Computers über das Betriebssystem.

Die hexadezimale Anzeige dient als Ausgabeeinheit des Computers. Die Anzeige wird dazu so angesteuert, daß alle 7 (8) Segmente einer Ziffer frei kombinierbar sind. Daraus ergibt sich eine große Zahl von zusätzlichen Symbolen (s. Tabelle11.5).

Die Helligkeit der Anzeige kann in Abhängigkeit von den dargestellten Zeichen schwanken.

4.2 Datenterminal

An die Kommunikation-Steckverbindung (Bild 7) lassen sich verschiedene Daten Ein- und Ausgabegeräte anschließen. Eine individuelle Anpassung kann durch Umstecken von Brücken auf der Hauptplatine durchgeführt werden. Die Anschlußanordnung des Steckers finden Sie in Bild 8.

Für den Betrieb eines Datenterminals müssen die Brücken S7 und S8 geschlossen werden. Damit ist die V24- bzw. TTL-Schnittstelle in Betrieb. Sind die beiden Brücken offen, so ist die Stromschnittstelle in Betrieb und es kann z.B. eine Teletype angeschlossen werden. Das Gerät ist ab Werk auf V24 Betrieb eingestellt.

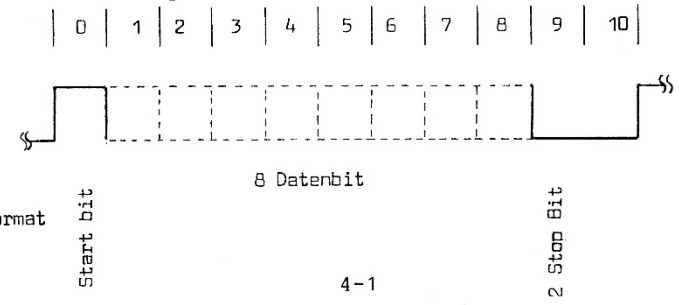


Bild 12

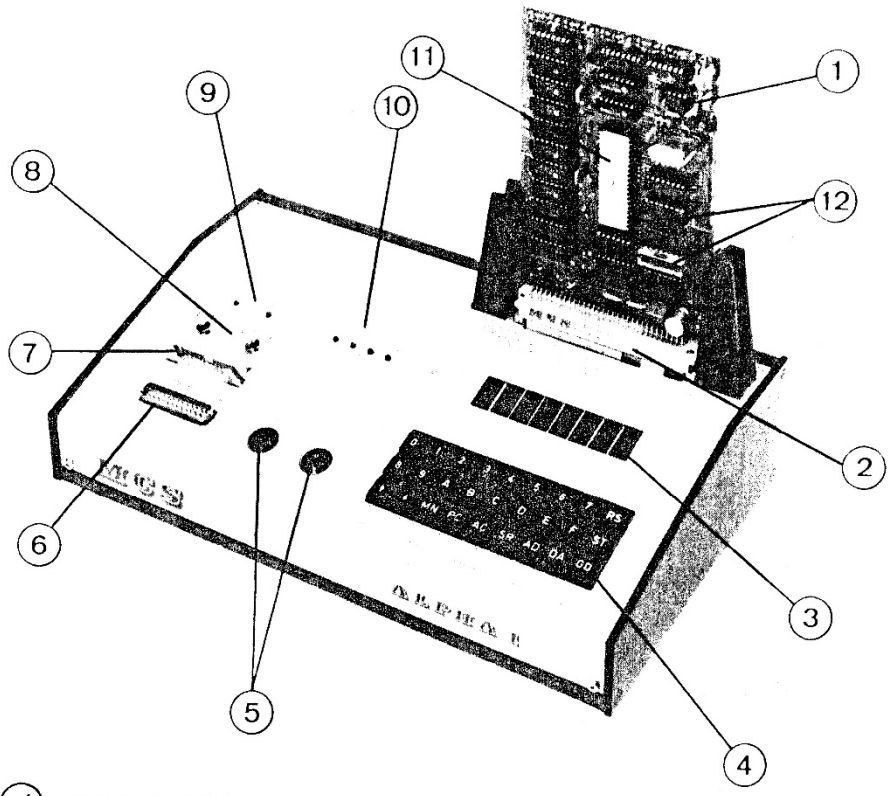


Bild 5

(1) CPU-Karte M622

(2) BUS-Steckverbindung

(3) Display

(4) Tastatur

(5) 1 Anschluß zum Schreiben auf Magnetband
2 Anschluß zum Lesen vom Magnetband

(6) Anschluß externer Peripheriegeräte

(7) Ein- Ausgabe- Anschluß für eigene Anwendungen

(8) Betriebsartenschalter

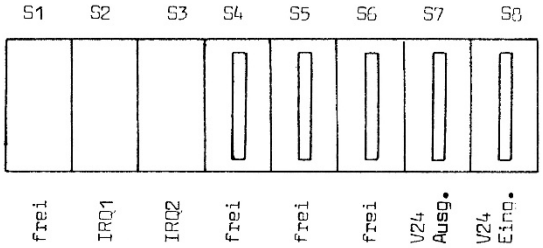
(9) Interrupt - Anzeige

(10) Hilfskontrolleuchten

(11) Mikroprocessor 6502

(12) Nur bestückt in Verbindung mit BETA 8 System

Zuordnung Kommunikationsrecker
(Grundbelegung)



S4, S5 } Brücken als Reserve für IRQ

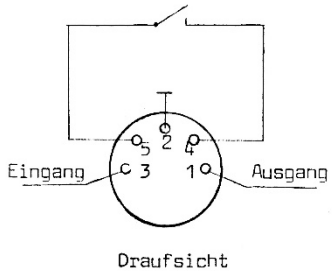
S6 Brücke nur notwendig wenn Vcc an Kommunikationstecker benötigt

S7, S8 } Brücken notwendig bei V24 und TTL Betrieb

S7, S8 } entfernen für Stromschnittstelle

Bild 8

Recorderanschluß 1 und 2



Ausgangspegel		1V
Quellewiderstand		= 1K
Eingangspegel	Min.	100mV
	Max.	5V
Eingangswiderstand		10K

Bild 10

Nach dem Restart setzt MONA die Baudrate auf 110 BAUD.

Datenformat: Ein Start-Bit, 8 ADCII-Bits, 2 Stop-Bit, s. auch Bild 12.

Mit der Brücke S6 wird die Spannung V_{CC} an die Steckverbindung gelegt, z.B. für Lochstreifengeräte. Die Brücken S2 und S3 werden dann eingesetzt, wenn eine Interruptverarbeitung von den Bausteinen 6532-1 oder 6532-2 gewünscht wird.

In Kapitel 8 wird gezeigt, wie man auch höhere Baudraten einstellen kann. Dort werden auch Sondertasten erklärt.

4.3 Magnetband

ALPHA 1 bietet die Anschlußmöglichkeit für zwei Magnetbandgeräte. Sie können damit besonders leicht Magnetbänder kopieren. Das Abspielgerät wird dazu an Buchse 2 angeschlossen, das Aufnahmegerät an Buchse 1. Wenn die Geräte einen externen Start/Stop-Anschluß besitzen, können diese über die Recorderbuchsen auch ferngesteuert werden. Dazu schließen Relais-Kontakte die Anschlüsse 4 und 5 der fünfpoligen Recorderbuchsen kurz.

Die verwandten Signalpegel sind an beiden Bandbuchsen identisch. Anschluß 1 ist der Ausgang, Anschluß 3 ist der Eingang und Anschluß 2 ist der Ground (GND) (s. auch Bild 10).

Die Aussteuerung bei der Aufnahme sollte etwas unterhalb (-6db) gegenüber Vollaussteuerung liegen.

Kapitel 5

Systemerweiterung

Da der Microcomputer ALPHA 1 als BUS-orientiertes System konzipiert wurde, besteht somit die Kompatibilität zu dem von MCS entwickelten Microprozessorsystem BETA 8. Dies bedeutet für den Besitzer von ALPHA 1 den Einsatz von allen Karten des BETA 8-Systems. Obwohl alle Komponenten einzeln erhältlich sind, ist es jedoch ratsam, den Grundaufbau von BETA 8, bestehend aus Gehäuse, Mutterplatine für 16 Eurokarten und Netzteil als komplette Einheit einzusetzen. In dieses System wird dann die CPU-Karte von ALPHA 1 als zentrale Steuereinheit eingesetzt und das Pult über eine Busverlängerungskarte angeschlossen. Je nach Problemstellung können dann die RAM, ROM, I/O u.a. Karten des BETA 8-Systems als Erweiterung eingesetzt werden.

Kapitel 6

Die Programmierung des ALPHA 1

6.1 Einführung

Wenn Sie Ihren ALPHA 1 - Computer in Betrieb genommen haben, können Sie sofort mit dem Programmieren beginnen.

Der Betrieb mit den von uns gelieferten Programmen demonstriert Ihnen schon die vielfältigen Möglichkeiten, die ALPHA 1 bietet (testen Sie doch einmal die Digitaluhr aus Kapitel 10). Ein echtes Erfolgserlebnis wird sich bei Ihnen aber sicher erst einstellen, wenn Ihr erstes eigenes Programm läuft.

Es wird eine gewisse Zeit dauern, bis Sie den richtigen Dreh bei der Programmierung heraushaben. Nach Überwindung dieser ersten Hürde werden Sie aber in der Lage sein, eigene Steuerprogramme ebenso zu schreiben, wie Sie vielleicht die Möglichkeit entdecken, ein eigenes Spiel zu programmieren.

In diesem Kapitel werden Sie in die Programmiertechniken des ALPHA 1 eingeführt, so daß Sie nach und nach Ihren eigenen Weg bei der Handhabung des Microcomputers finden. Fassen Sie möglichst oft den Mut zu eigenen Versuchen. Gerade das Experimentieren an kleinen Testprogrammen und deren Veränderung erleichtert das Verständnis für die Arbeitsweise des Microcomputers und bringt Sie in Ihrem Wissensstand weiter. Wir haben Ihren ALPHA 1 mit dem Betriebssystem MONA (= Monitor ALPHA) besonders für diesen Zweck vorbereitet. Sie können sehr leicht kleine (und wenn Sie wollen natürlich auch große!) Programme direkt über die Tastatur schreiben, korrigieren und über das eingebaute Display kontrollieren. Über besondere Betriebsarten (Single-Step = Einzelschritt, Slow-Step = langsam) können Sie Ihre Programme testen und Ihre Arbeitsweise beobachten.

Nutzen Sie diese Möglichkeit z. B. durch Abwandlungen der später beschriebenen Programme oder verwirklichen Sie Ihre eigenen Ideen.

6.2 Einige Tastenfunktionen

[RS] Nach dem Einschalten der Versorgungsspannung führt Ihr ALPHA 1 einen automatischen RESTART durch.

Das bedeutet, daß Ihr Microcomputer in einen definierten Grundzustand gebracht wird. Wenn Sie glauben, einen Fehler gemacht zu haben, können Sie den Grundzustand auch durch Betätigen der Taste -RS- wieder herstellen.

Folgende Parameter werden durch -RS- betroffen:
(Wenn Sie diese Begriffe noch nicht kennen, lesen Sie einfach bei der nächsten Taste weiter. Sie können später hier nachschlagen).

1.) der Stackpointer wird auf 01FF gesetzt
2.) der Status wird auf 00 gesetzt
3.) die Aufzeichnungsraten für den Magnetbandspeicher werden auf "normal" gesetzt
4.) die Baudrate für das Terminal wird auf 110 Baud gesetzt
5.) die Interruptvektoren werden auf das Betriebssystem MONA eingestellt
6.) der Tastenvektor wird auf MONA gestellt
7.) die Anzeigeart wird auf hexadezimal geschaltet und der Adressmodus gesetzt.

[0] bis [F] Zum Verständnis dieser Tasten benutzen Sie sie bitte gleich:
Drücken Sie -RS- 0000
Die von Ihnen eingegebene Zahl erscheint links im Display mit erleuchteten Punkten. Es ist die Adresse des Speicherplatzes 0000. Sein Inhalt wird rechts angezeigt und ist zunächst undefiniert und zufällig beim Einschalten des Gerätes entstanden. Speicher dieses Typs werden RAM = Random Access Memory genannt und können von Ihnen nach Belieben belegt werden.

Sie haben über 1000 derartige Speicherplätze im ALPHA 1 für Ihre Programme zur Verfügung - das reicht schon für erstaunlich viele Anwendungen.

DA Drücken Sie nun die Taste -DA- (Daten). Die leuchtenden Punkte unter dem Inhalt des Speichers 0000 zeigen Ihnen, daß Sie nun Zugang zu diesem Speicherplatz haben.

Durch Betätigung der Tasten -0- bis -F- können Sie den Inhalt be-
↑ liebig verändern. Geben Sie einmal Ihre Glückszahl ein und drücken Sie -↑-.
Es erscheint die nächste Speicherzelle 0001. Die Taste -↓- bringt
↓ sie wieder zurück: es erscheint wieder Ihre Glückszahl als Inhalt der Speicherzelle 0000 - der Speicher hält also die von Ihnen eingegebenen Zahlen fest!

AD Sie können die Dateneingabe wieder verlassen, indem Sie -AD- (Adresse) drücken. Nun leuchten wieder die Punkte unter der Adresseneingabe und die Betätigung der Tasten -0- bis -F- bringt einen anderen Speicherplatz in die Anzeige.

6.3 Hexadezimal Zahlen

Wie Sie vielleicht schon bemerkt haben, werden hier auch die Buchstaben A bis F als Zahlen verwandt. Um dies zu verstehen, möchten wir Ihnen das hexadezimale Zahlensystem erklären:
Drücken Sie mehrmals die Taste -↑-. Sie sehen die Speicherplätze in aufsteigender Reihenfolge mit Zahlen, die Sie gewohnt sind. Bei 0009 halten Sie bitte an. Was wird die nächste Zahl sein? Man würde meinen 0010, und das wäre auch richtig, wenn ALPHA 1 das dezimale Zahlensystem benutzen würde. ALPHA 1 hat aber 6 weitere Zahlen, die über 9 hinausgehen, weil er ein Computer-Zahlensystem benutzt, das "hexadezimal" genannt wird. Drücken Sie jetzt wieder die Taste -↑-, es erscheint die Adresse 000A.

Erst nach der Adresse 000F erscheint 0010 und nun fängt alles wieder von vorne an.

Merken Sie sich also, daß auch die Buchstaben A - F "nur" Zahlen sind und lassen Sie sich nicht durch sie verwirren. Wenn Sie einmal etwas Zeit haben, können Sie z.B. im 6502 - Programmierhandbuch mehr über dieses Zahlensystem lesen.

6.4 ROM = Read Only Memory

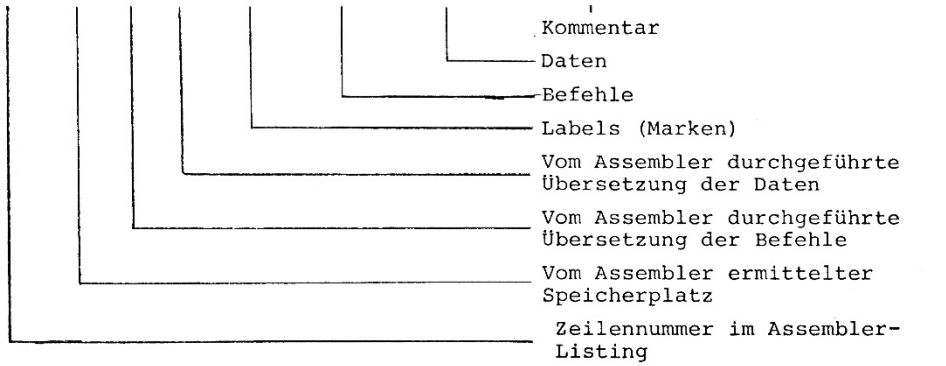
Bitte wählen Sie nun die Adresse F82A an. Ihr Inhalt wird A 9 sein. Bitte überprüfen Sie selbst, daß dieser Inhalt nicht veränderbar ist. Es ist eine Speicherzelle, deren Inhalt fest vorgegeben ist und nur gelesen werden kann. Dieser sogenannte ROM-Speicher enthält das Programm MONA (MONITOR ALPHA), das u.a. für Sie die Anzeige leuchten läßt, die Tasten abfragt und auch Lochstreifen stanzen kann.

6.5 Mini-Programm A:

Verändern der Speicherplätze 0201 bis 0210.
Dies ist ein sehr einfaches Programm und es tut nichts Besonderes, aber es gibt Ihnen etwas Sicherheit im Umgang mit den Tasten und Sie lernen einiges über die Arbeitsweise des Microcomputers. Bitte laden Sie das Programm zunächst, später erklären wir, wie es funktioniert.

Im folgenden sehen Sie die Programmliste, nach der Sie das Programm laden können.

```
 SEITE 1      MINIPROG. A    auf MICRONIC-650X-ASSEMBLER
Lauf 1 beendet                      0  Fehler
 0010            *
 0020            *
 0030 0000                    ORG    $0000      ERSTE PROG.-ADRESSE
 0040 0000 A0 10    GO        LDY    ##10       ES SOLLEN 16 SPEICHERZELLEN
 0050 0002 A9 FF              LDA    ##$FF      AUF "FF" GESETZT WERDEN,
 0060 0004 99 0002  WEITER    STA    $0200,Y    VON 0200+Y AN
 0070 0007 88                 DEY               ABWAERTS,
 0080 0008 D0 FA              BNE    WEITER     BIS Y=0,
 0090 000A 4C 52F8  ENDE      JMP    $F852      DANN SPRUNG INS BETRIEBSSYSTEM
 0100 000D                    END               ENDE DES ASSEMBLER-LISTINGS
```



Sie laden das Programm, indem Sie die Adresse 0000 einstellen. Drücken Sie die Taste -DA- und dann die erste der eingerahmten Zahlen, also A 0. Dies ist nun der neue Inhalt des Speicherplatzes 0000. Nach Drücken der -↑- Taste können Sie 10 in 0001 schreiben usw. Wenn Sie F8 eingegeben haben, drücken Sie AD 0000 und überprüfen Sie den richtigen Inhalt noch einmal mit -↑-. Vor dem Programmstart vergewissern Sie sich, daß der Betriebsartenschalter auf N = NORMAL steht! Nun gehen Sie wieder nach 0000 und starten Sie das Programm mit -GO-. Wenn in der Anzeige eine neue Adresse erscheint, haben Sie Ihr erstes eigenes Programm laufen lassen - wir gratulieren!

Das Programm hat den Inhalt der Speicherplätze 0201 bis 0210 auf "FF" gesetzt. Das ging in weniger als 1/1000 Sekunden - zu schnell um die Anzeige auch nur flackern zu sehen. Wählen Sie einmal 0201 an und betrachten Sie den Inhalt - hat es geklappt?

Nun zur Funktion des Programms:
Bitte schauen Sie auf die Spalte Mnemonic des Listings:
LDY$10 bedeutet lade das Indexregister Y im Microprozessor mit der Hexa-Zahl 10 ($ steht für Hexa). Die Abkürzung LDY muß für den Microcomputer in einen für ihn verständlichen Hexacode übersetzt werden. Für LDY lautet die Übersetzung A0. Diese finden Sie in der Spalte Inhalt unter der Adresse 0000. Die nächste Adresse 0001 hat die von Ihnen eingegebene Zahl 10 zum Inhalt.

In der nächsten Zeile wird mit LDA (= Lade den Accumulator) eine Zahl, in diesem Fall FF in den Accumulator geschrieben. Anschließend wird dieser Accumulatorinhalt unter 0200 + Y = 0210 abgespeichert, Y war ja 10. Nun wird Y um 1 decrementiert (vermindert) und überprüft, ob es bereits 0 ist. Der Befehl BNE bedeutet branch if not equal (= verzweige wenn nicht Null). Im ersten Durchlauf ist Y tatsächlich nicht Null, sondern 0F und so wird nach "WEITER" verzweigt. Der Accumulatorinhalt wird im nächsten Schritt wieder nach 0200 + Y geschrieben. Y war nun 0F, damit war die Summe aus 0200 und Y = 020F. Nun wird Y wieder decrementiert usw. Wenn Y = 0 ist, wird nicht wieder verzweigt, sondern ein JMP (SPRUNG) Befehl nach F852 durchgeführt. Dies ist der Anfangspunkt von MONA.

6.6 Das Testen von Programmen mit Slow- und Single Step

Sie können sich das Programm auch einmal im Slow-Step zeigen lassen. Gehen Sie dazu wieder nach OOOO und starten Sie das Programm, nachdem Sie den Schalter auf Slow-Step geschaltet haben. Sie sehen nun jeden Befehl angezeigt, bevor er ausgeführt wird. Schauen Sie, wie die Schleife "Weiter" durchlaufen wird. Um im Slow-Step anzuhalten, stellen Sie den Schalter wieder auf Normal.

In der Stellung Signle-Step bleibt das Programm am nächsten Befehl stehen. Sie können mit -GO- fortschreiten usw.

6.7 Weitere Tastenfunktionen

Wenn Sie zwischen zwei Schritten einmal den Inhalt des Accumu-
AC lators sehen wollen, so halten Sie an und drücken Sie -AC-. Sie sehen nun die Adresse eines Speicherplatzes, der eine Kopie des
SR Accumulators enthält. Unter -SR- finden Sie ebenso eine Kopie des Statusregisters. Wenn Sie -SR- -↑- drücken, sehen Sie eine Kopie des X-Registers und nach nochmaligem -↑- eine des Y-Registers. Wenn Sie zurück zum nächsten auszuführenden Programm-
PC schritt wollen, drücken Sie -PC-,-der Microcomputer hat sich gemerkt, welches der nächste Schritt im Programm war.

Übrigens werden die Register-Kopien nach jedem durchgeführten Single- oder Slow-Step auf den neuesten Stand gebracht, wie Sie z. B. durch Anschauen des X-Registers -SR- - - nach mehreren "WEITER"-Schleifen sehen können. Die Kopien sind notwendig, da die "echten" Arbeitsregister des Microprocessors nicht direkt zugänglich sind.

Eine besonders angenehme Eigenschaft von MONA ist die Übersetzung des Hexacodes in Mnemonic.

Gehen Sie bitte nochmals nach OOOO,schalten Sie auf Slow und drücken Sie -MN-. Sehen Sie die Wirkung? Der Hexacode A2 ist in LDY übersetzt worden. Nun drücken Sie -GO- und verfolgen Sie, wie das Programm Schritt für Schritt arbeitet.

Auch beim Erstellen von Programmen ist die Mnemonic-Darstellung nützlich. Sie können sogar wieder in den Adreß-Modus gehen und die Mnemonic-Darstellung beibehalten. Erst ein Druck auf die -DA- Taste schaltet wieder auf Hexa-Darstellung um. Beachten Sie dabei, daß MONA sich bemüht, jeden Hexacode - also auch echte Daten - zu übersetzen. Nur wenn dies nicht möglich ist, wird - - - angezeigt, d.h. es gibt keine Übersetzung.

[ST] Die Stop-Taste benötigen Sie, wenn Sie ein eigenes Programm stoppen wollen. Sie löschen damit einen IRQ (Interrupt Request = Unterbrechungs-Anforderung) aus.

Dies bedeutet, daß der Processor als nächstes einen bestimmten Programmschritt (hier den Anfang von MONA) ausführt, unabhängig davon, welcher Befehl sonst im normalen Ablauf an der Reihe gewesen wäre.

Damit ist Ihr Programm gestoppt und MONA läuft wieder.

Kapitel 7

Benutzung eines Magnetband-Speichers

Als Magnetband-Speicher läßt sich z.B. jeder beliebige Cassettenrecorder der Unterhaltungselektronik benutzen.

Nach Anschluß eines solchen Gerätes versuchen Sie einmal, zur Übung Ihr Miniprogramm auf Band zu übertragen:

Schreiben Sie: AD F7FO

DA	OO	Startadresse
↑	OO	
↑	OD	Endadresse + 1
↑	OO	
↑	O1	Identifikationsnummer
AD	FE1B	Start des Aufzeichnungsprogramms

Sie haben damit die Startadresse und die Endadresse sowie eine Identifikationsnummer vorgegeben. Starten Sie das Band in Stellung Aufnahme und drücken Sie-GO:

Wenn die Anzeige wieder erleuchtet wird, ist die Aufzeichnung beendet.

Nun können Sie das Band zurückspulen, das Programm ist bereits auf ihm gespeichert. Glauben Sie es nicht? Dann schalten Sie ALHPH 1 einfach aus. - Sie wissen ja: das Programm im RAM ist nun verloren. Nach dem Wiedereinschalten wählen Sie bitte die Adresse F7F4 an und setzen Sie den Inhalt dieses Speicherplatzes auf O1. Dies ist die Identifikationsnummer,unter der MONA Ihre Bandaufzeichnung wiedererkennt. Nun gehen Sie zum Anfang des Bandleseprogramms (Adresse FECB) und drücken Sie GO. Starten Sie das Band auf Wiedergabe in mittlerer Lautstärke und warten Sie, bis die ALPHA 1-Anzeige wieder leuchtet. Wenn Sie die erste Adresse Ihres Programms OOOO sehen, ist alles glatt gegangen und Ihr Programm steht wieder an der alten Stelle. Falls FFFF erscheint oder die Anzeige gar nicht wieder erleuchtet wird, ist beim Lesen des Magnetbandes ein Fehler aufgetreten. - Vielleicht haben Sie das Band nicht richtig ausgesteuert - versuchen Sie es noch einmal.

Zusätzlich zu der digitalen Information liegt an den Recorderbuchsen je ein Relaiskontakt an. Sie können damit die Start-und Stop-Funktionen Ihres Recorders von MONA steuern lassen.

Wenn Sie z.B. die GO-Taste zur Aufnahme eines Bandes drücken, so wird sofort die Lampe L4 erleuchtet und gleichzeitig werden die Anschlüsse Pin 4 + 5 an Bu 1 über ein Relais verbunden. Dies kann Ihren Recorder starten. Die digitalen Daten werden erst nach einer Verzögerung von ca. 4 Sec. gesandt, so daß Ihr Magnetband zunächst eine stabile Geschwindigkeit erreichen kann. Das Relais fällt wieder ab, wenn die Aufzeichnung beendet ist.

Über BU2 erhalten Sie ebenfalls zusätzlich zur digitalen Information eine Verbindung der Pins 4 + 5, wenn MONA's Band-Leseprogramm läuft. Wenn Sie an der Adresse (Start Band-lesen) die GO-Taste drücken, wird sofort die Lampe L3 erleuchtet und ein Relais zieht parallel dazu an. Auch dies kann wieder Ihren Recorder starten.

Wenn Sie einmal ein besonders schönes Programm geschrieben haben und z.B. die Kopie des Datenbandes einem Freund geben wollen, so taucht das Problem auf: Wie kopiere ich ein Band?

Die einfachste Lösung, nämlich das Kopieren von Band zu Band, können wir Ihnen leider nicht empfehlen - es birgt zu viele Fehlerquellen.

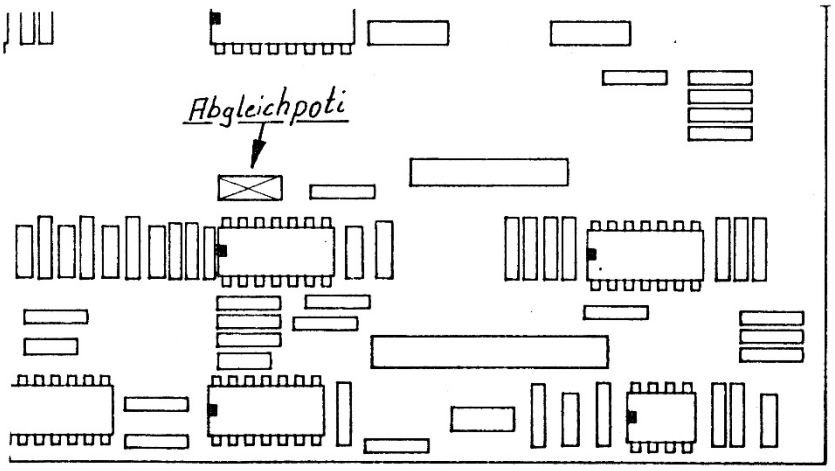
Vielmehr sollten die Daten beim Lesen überprüft werden und dann "frisch" aufgezeichnet werden. Dies ist aber durch die 2. Bandbuchse besonders einfach, wenn Sie 2 Recorder besitzen. Verbinden Sie das wiedergebende Gerät mit der Buchse 2 und das aufnehmende Gerät mit der Buchse 1. Wenn beide Geräte entsprechend angeschlossen sind, können Sie nun ohne Umstecken, einfach durch Starten des Bandleseprogramms ein Programm einlesen und nach Setzen der Parameter für ID, Anfangs-und Endadresse nach Band 1 überspielen, indem Sie das Schreibprogramm starten. Die Start-Stop-Steuerung erfolgt automatisch durch MONA.

7.1 Abgleich

Der Abgleich des Tonbandinterfaces ist von uns vorgenommen worden. Ein neuerlicher Abgleich ist nur notwendig, wenn ein Bauelement in diesem Interface ausgewechselt wurde.

Abgleichen heißt, den VCO (spannungsgesteuerter Oszillator), der im PLL (Phase Locked Loop)-Baustein integriert ist, auf die Mitte der beiden Eckfrequenzen 3,7 kHz und 2,4 kHz einzustellen. Diese Mittenfrequenz von 3,05 kHz erzeugt MONA, wenn Sie die Adresse FF93 einstellen und -GO- drücken. Verbinden Sie nun den Recorderausgang mit dem Recordereingang an einer der Recorderbuchsen (Pin 1 mit Pin 3). Schließen Sie an Pin 10 des Kommunikationssteckers ein Voltmeter oder Oszillosscope an. Verstellen Sie nun das 2K2 Trimmpotentiometer, bis die Spannung zwischen OV und 5V bzw. 5V und OV umschaltet.

Damit ist der Abgleich beendet.



Ausschnitt Hauptplatte

Kapitel 8

Benutzung eines Datenterminals

Nach dem Anschluß entsprechend Kap. 4.2 haben Sie mit einem Datenterminal folgende Bedienungsmöglichkeiten:

Nachdem der Betriebsartenschalter in Stellung -COM- gebracht wurde, meldet sich das Betriebssystem mit:

MONA XXXX YY ZZZ

Hierbei bedeutet -X- eine Adresse, -Y- ist die zu dieser Adresse gehörende Date in hexadezimal und, sofern möglich, zeigt -Z- den mnemonischen Code der Date.

8.1 Bedienung

Die Betätigung der Spacetaste (Leertaste) -SP- beendet die Adresseingabe, d.h. die zuletzt gewählten Zeichen (O....F) werden als Adresse interpretiert. Da keine führenden Nullen eingegeben werden brauchen, bedeutet die Betätigung nur dieser Taste, daß die Adresse Null angewählt wurde. Die Darstellung ist

OOOO YY ZZZ

Die Betätigung der Punkttaste -.- beendet die Dateneingabe, d.h. die zwei zuletzt gewählten Zeichen (O....F) werden als Date interpretiert. Gleichzeitig wird die Adresse um "eins" erhöht (inkrementiert). Da auch bei der Dateneingabe keine führenden Nullen eingegeben werden müssen, bedeutet die Betätigung nur dieser Taste, daß in die gerade angewählte Adresse der Inhalt Null geschrieben wird. Die Darstellung ist:

OOOO Y_1Y_1 $Z_1Z_1Z_1$ (vor Betätigung von -.-)

OOOl Y_2Y_2 $Z_2Z_2Z_2$ (nach Betätigung von -.-)

In Adresse OOOO steht jedoch jetzt die Date OO (Betätigen Sie die SP-Taste für diese Kontrolle).

Bei Betätigung der Carriagereturn-Taste (Wagenrücklauf) -CR- wird die Adresse um eins erhöht, ohne Beeinflussung der Daten.

Manche Datenterminals haben den Kombinationsbefehl -CRLF- (Carriagereturn-Linefeed = Wagenrücklauf und Zeilenwechsel), dieser kann ebenfalls hierfür benutzt werden.

Bei Betätigung der Linefeedtaste (Zeilenwechsel) wird die angezeigte Adresse um eins erniedrigt (dekrementiert).

Die Taste -G- startet ein Programm mit der Adresse, die vorher angewählt wurde.

Mit der Taste -Q- wird die Ausgabe eines Lochstreifens eingeleitet. Zuvor muß die Endadresse der Daten unter F7F2 = Adressbyte "Low" und unter F7F3 = Adressbyte "High" abgelegt werden.
Wählen Sie nun die Anfangsadresse der zu stanzenden Daten und drücken Sie -Q-.

Die Taste -L- startet die Leseroutine für Lochstreifen. Die Startadresse, unter der die ersten Daten abgelegt werden sollen, wird vom Lochstreifen übernommen.

Die Taste -S- gestattet die Umschaltung der Baudrate. Die folgenden 4 Zeichen geben die Geschwindigkeit an, mit der ab diesen Zeitpunkt übertragen werden soll. Die Rückmeldung von MONA wird Ihr Terminal nicht verstehen, bevor Sie es nicht auf die neue Baudrate umstellen.

Baudrate:

75	=	606E
110	=	4B8C
150	=	5366
300	=	3D32
600	=	3B18
1200	=	040C
2400	=	6004
4800	=	4C01

DELETE oder RUBOUT veranlaßt MONA, die Systemmeldung zu senden. Vorher geschriebene Daten werden ignoriert.

Kapitel 9

Struktur des Betriebssystems MONA

9.0 Befehlsablauf

Hier können Sie den schematischen Programmablauf verfolgen.
Für genauere Einzelheiten lesen Sie bitte das Programm-Listing.

Mit dem Einschalten der Netzversorgung führt der Microprocessor 65OX einen automatischen Restart durch. MONA startet so definiert am "RESTART"-Eingang (Adresse Γ822). Es werden dort einige Register und insbesondere der Stack initialisiert und dann zum Punkt STP gesprungen. Dort endet auch die STOP-Routine die über den IRQ (BRK) oder NMI-Vektor erreicht werden kann und deren Aufgabe es ist, die CPU Register in das RAM zu kopieren.

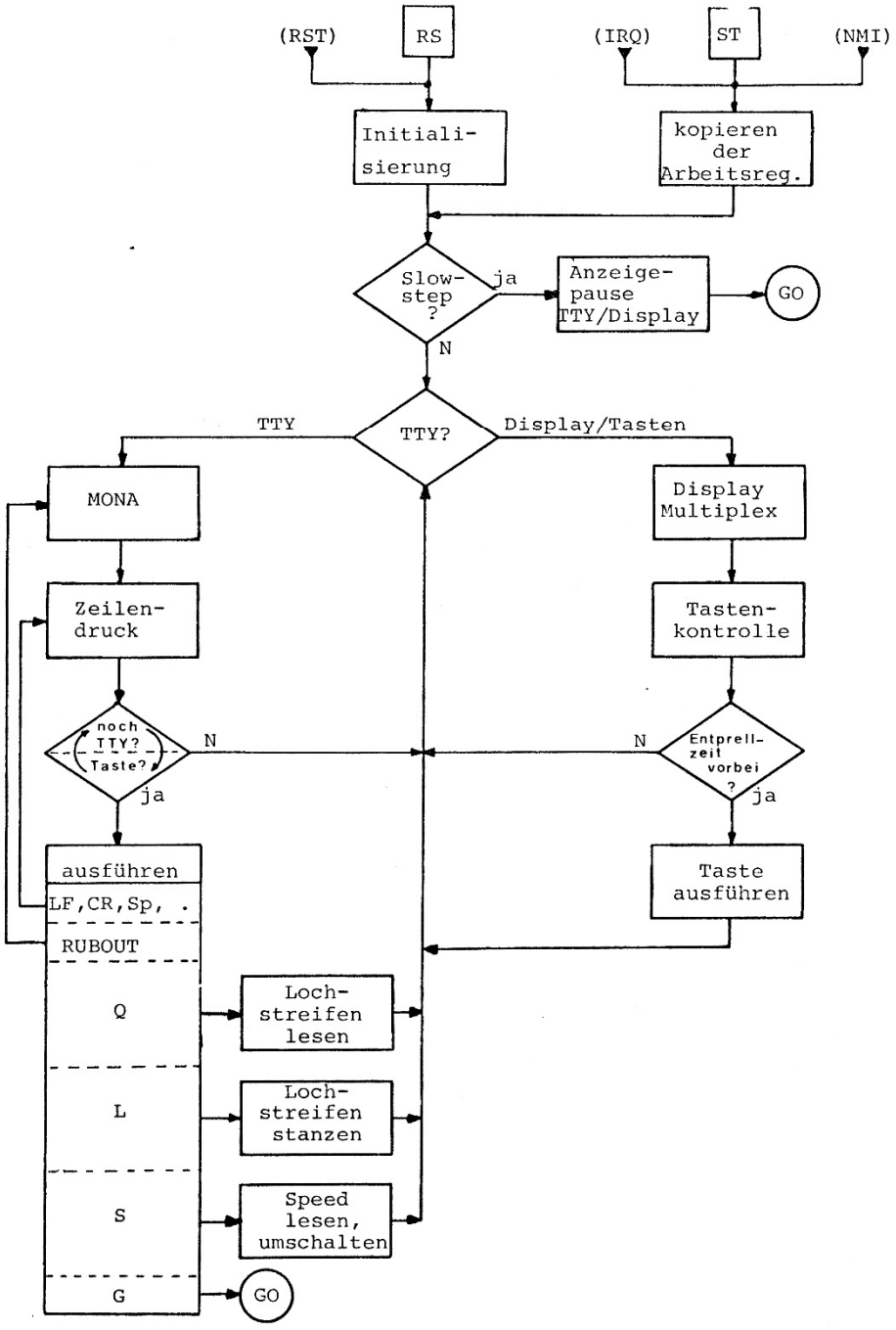
Vom Punkt STP an arbeiten die beiden Eingangsroutinen gemeinsam weiter. Es wird zunächst der SL-Schalter überprüft und entsprechend evtl. sofort ein neuer GO-Befehl ausgeführt, sonst wird der TTY/KEY Schalter überprüft und in die TTY- oder Display-Tasten Routine gesprungen.

1. Display/Tasten Zyklus
   In einem fortlaufenden Zyclus wird für einige Millisekunden das Display eingeschaltet, dann werden die Tasten nach einem geschlossenen Kontakt abgefragt das Display wieder eingeschaltet usw.

   Sofern in mehreren aufeinanderfolgenden Zyclen eine Taste als gedrückt registriert wurde, wird diese Taste einmal ausgeführt.

2. TTY
   Es wird nach der Systemmeldung MONA der gegenwärtige Stand von ADRH und ADRL mit dem Inhalt dieser Speicherzelle ausgedruckt. Danach wird die Eingabe von ASCII-Zeichen erwartet. Nach Ausführung der eingegebenen Zeichen wird wieder eine Zeile gedruckt usw.

Flußdiagramm MONA



MONA-Organisation

Adresse	Funktion	
FFFE	höheres	IRQ-BYTE
FFFE	niederes	
FFFD	höheres	RST-BYTE
FFFC	niederes	
FFFB	niederes	NMI-BYTE
FFFA	höheres	
FFB2	Anfangsadresse Uhr	
FF93	Anfangsadresse PLL Test	
FECB	Datenband Lesen	
FE1B	Datenband Schreiben	
FD92	Disassembler Programmteil	
FC6A	Disassembler Tabellenteil	
FC48	Anfangsadresse Hexazeichen lesen	
FBF9	Anfangsadresse Lochstreifen lesen	
FB7D	Anfangsadresse Lochstreifen schreiben	
FB4C	Anfangsadresse Tastenkontrolle	
FAAF	Anfangsadresse Hexazeichen schreiben	
FA86	Anfangsadresse ASCII-Zeichen schreiben	
FA4F	Anfangsadresse Carriage Return-Schreiben	
F9F1	Anfangsadresse ASCII-Zeichen Lesen	
F852	Stop Eingang, BRK Eingang	
F822	RST Eingang	

Besondere Speicherzellen des Betriebssystems

Adresse	Bedeutung	Abkürzung
F7FF	höheres Byte indirekter	IRQH
F7FE	niederes IRQ Vektor	IRQL
F7FD	höheres Byte indirekter	NMIH
F7FA	niederes NMI Vektor	NMIL
F7F9	höheres Byte indirekter	TAHIGH
F7F8	niedere Tastenvektor	TALOW
F7F6	höheres Byte Baudrate	BAUDH
F7F5	niederes	BAUDL
F7F4	Identifikationsnr. Datenband	ID
F7F3	höheres Byte Endadresse	ENDEH
F7F2	niederes Magnetband/Lochstr.	ENDEL
F7F1	höheres Byte Startadresse	STRTH
F7FØ	niederes Magnetband	STRTL
F7EF	Systemmeldung ein/aus	ECHO
F684	Intervalltimer MONA	
F683	Richtungsregister für Port B	PBDD
F682	Port B	PB
F681	Richtungsregister für Port A	PADD
F68Ø	Port A	PA
ØØEB	Y-Register Kopie	YSAVE
ØØEA	X-Register Kopie	XSAVE
ØØE9	Status Kopie	STATUS
ØØE8	Stakpointer Kopie	STPOIN
ØØE7	Akkumulator	AKKU
ØØFØ	höheres Adressbyte	ADH
ØØEF	niederes Adressbyte	ADL

9.1 Datenformat der Magnetbandroutine

Zur Aufzeichnung von Daten auf ein Magnetband wird ein besonderes Datenformat benötigt um eine hohe Datensicherung zu erreichen.

Jede aus dem Speicher gelesene 8 Bit breite Date (1 Byte) wird in zwei 4 Bit-Hexadezimale Zahlen zerlegt (0.....F) und als 7-Bit ASCII-Zeichen interpretiert.

Vor jede Aufzeichnung, die beliebig lang sein kann, werden 100 SYN-Zeichen (ASCII16) und ein "*" (ASCII2A) gesetzt. Es folgt die zweistellige Identifikationsnummer (ID). Die zur Kennzeichnung unterschiedlicher Aufzeichnungen dient. Die nächsten 4 Zeichen (Adresse "Low", Adresse "High") bestimmen die Startadresse von der an die Daten gelesen bzw. wieder abgelegt werden. Die nun folgenden Daten werden durch ein "/" (ASCII2F) und die 4stellige Prüfsumme (hexadezimale Addition) abgeschlossen.

Die Aufzeichnung wird durch 2 EOT-Zeichen (ASCII04) beendet.

Zur Aufzeichnung der einzelnen Bit's werden zwei Frequenzen gebildet und zwar 2,4 kHz und 3,6 kHz. Die Entscheidung ob ein Bit "high" oder "low" ist, liegt ausschließlich in dem Verhältnis beider Frequenzen innerhalb eines bestimmten Zeitraums. Die Übertragung eines Bits wird mit 3,6 kHz begonnen und mit 2,4 kHz beendet. Das Bit ist "low" wenn 2/3 3,6 kHz und 1/3 2,4 kHz Anteile vorliegen. Bei 1/3 3,6 kHz und 2/3 2,4 kHz ist das Bit "high". Ein Bit ist also immer gleich lang.

Die Aufzeichnungsgeschwindigkeit ist von der Taktanzahl der beiden Frequenzen für 1 Bit abhängig. Diese kann verändert werden um höhere Aufzeichnungsgeschwindigkeiten zu erreichen.

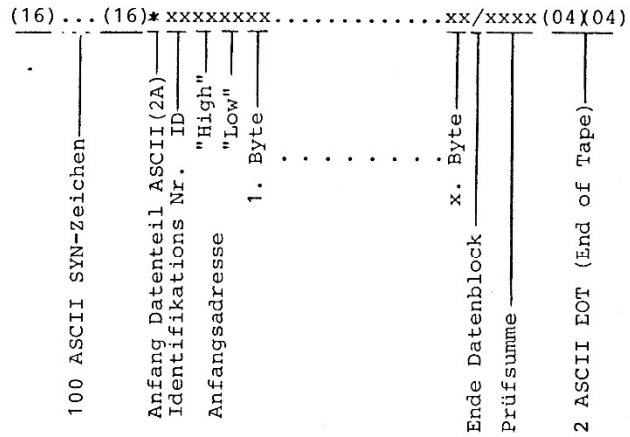
Rate	Freq. 1 (00FA)	Freq. 2 (00FC)
Normal	6	9
Schnell	4	6
Hyper	2	3

Beim Restart wird die Geschwindigkeit "normal" vorgegeben.

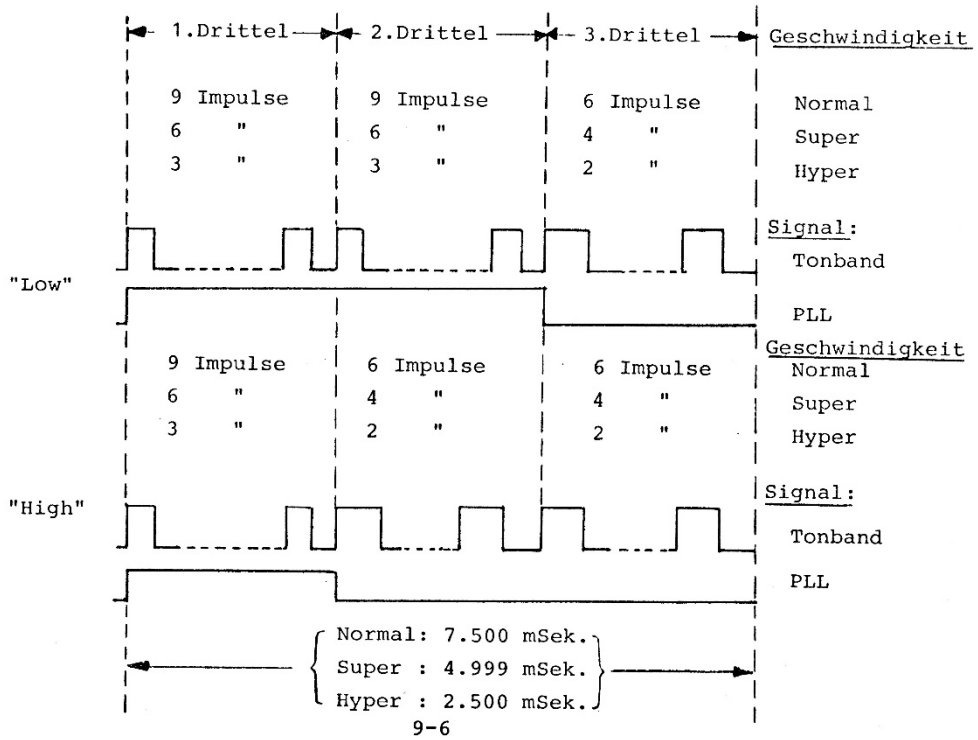
Beim Lesen vom Band wird die digitale Information von einem als FSK (Frequenzy Shifted Key)-Demodulator geschalteten PLL (Phase Locked Loop) gewonnen. Dieser erkennt die Frequenz

3,6 kHz als "high" und 2,4 kHz als "low" und führt die Information dem Processor zu.

Datenblock:



1 Bit:



9.2 Datenformat der Lochstreifenroutine

Das Stanzen und Lesen von Lochstreifen wird über die serielle Schnittstelle durchgeführt. Die Daten werden im folgenden Format übertragen:

Jede 8 Bit (1Byte) breite Date wird in zwei 4-Bit Daten zerlegt. Jede dieser zwei hexadezimalen Zahlen wird dann als entsprechendes ASCII-Zeichen (0....F) interpretiert und ausgestanzt. Beim Lesen ist es umgekehrt.

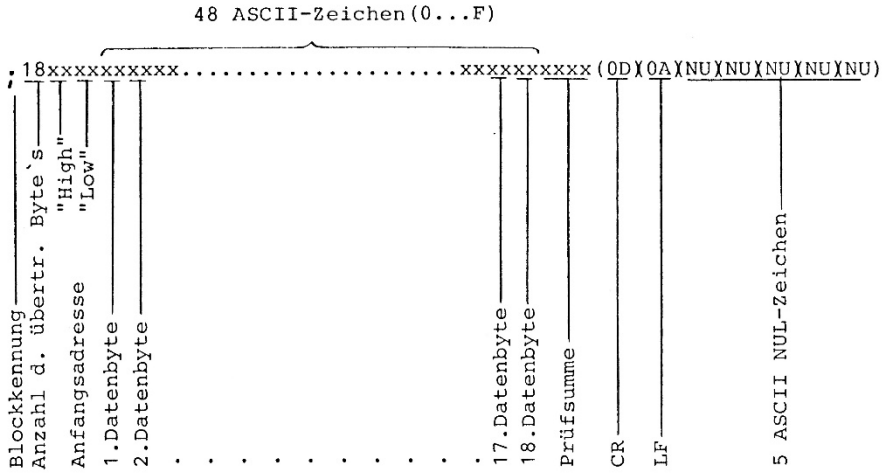
Die Daten können nur als Blöcke zu 48 ASCII-Zeichen, das entspricht 24 Byte, gestanzt bzw. gelesen werden. Jeder dieser Blöcke beginnt mit einem ";". Das Leseprogramm ignoriert alle ankommenden Zeichen bis es zum ersten Mal ein ";" erkennt.

Nun folgt die Anzahl der zu übertragenden Byte's in Hexadezimal (18). Die nächsten 4 Zeichen (Adressbyte "high", Adressbyte "low") bezeichnen die Adresse des ersten zu übertragenden Byte's, unter der es steht bzw. abgelegt werden soll.

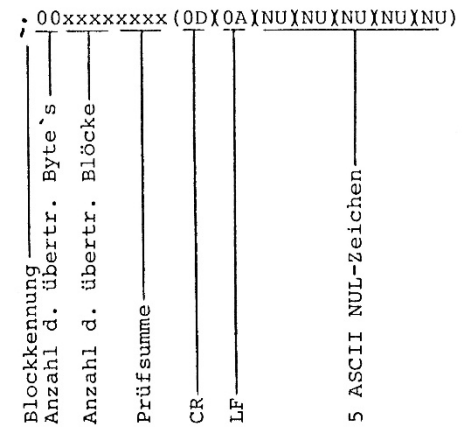
Nach den 48 Zeichen folgt eine vierstellige Prüfsumme (2 Byte) die sich aus der hexadezimalen Addition aller Zeichen außer dem ";" ergibt. Diese Prüfsumme wird beim Lesen ausgewertet. Den Abschluß eines Block's bilden ein Wagenrücklauf (CR = Carriage Return), ein Zeilenvorschub (LF = Line Feed) und 5 ASCII-Nullzeichen. Den Abschluß der Übertragung von einem oder mehreren Blöcken wird durch einen Block gekennzeichnet, der keine Daten enthält. Nach dem ";" steht also OO. Die folgenden 4 Zeichen (2 Byte's) geben die Anzahl der übertragenen Blöcke an. Danach folgt wieder die Prüfsumme, ein CR und ein LF.

Lochstreifenformat

Block:



Übertragung Ende:



Kapitel 10

Spiel und Testprogramme

10.1 MONA's Uhr

Um Ihnen die Möglichkeit zu geben, ein nützliches Programm ohne großen Aufwand laufen zu lassen, haben wir im ROM ein Digital-Uhr-Programm gespeichert.

Stellen Sie zunächst die momentane Zeit ein, indem Sie unter der Adresse

OODA die Stunden eingeben

und unter OOEO die Minuten (z.B. die nächste volle Minute).

Setzen Sie

OOE6 und OOE4 auf OO und gehen Sie zur

Adresse FFB3.

Warten Sie die volle Minute ab (Telephonzeitzeichen oder Fernseh/Radio-Zeitzeichen) und drücken Sie GO. Die Digital-Uhr läuft mit der Quarzgenauigkeit.

10.2 Weitere Programme (mitgelieferte Cassette)

Wie Sie sicher schon gelesen haben, sind die Tastatur sowie die LED-Anzeige von ALPHA I frei programmierbar! Außerdem können Sie die freien I/O-Leitungen voll in Ihren Programmen verwenden. Die auf der mitgelieferten Magnetbandcassette aufgezeichneten Programme sollen dies veranschaulichen.

Probieren Sie die Spiele einmal durch. Schon das "Ablaufen lassen" der Programme macht Spaß. Noch interessanter dürfte es werden, wenn Sie nach kurzer Zeit den logischen Ablauf der Programme verstanden haben - und sie ggfs. individuell modifizieren.

Nun aber zu konkreten Beispielen!

a. Einarmiger Bandit ID = ∅ 01

Laden Sie das Programm mit der NONA Magnetbandleseroutine (siehe Kapitel 7). Der Identifier (hexadezimale Kennzahl) ist ∅ 01 und starten Sie es mit der Adresse ∅ 0200. Die Spielbank gibt Ihnen ein Anfangskapital von 25,-- DM. Drücken Sie jetzt wieder die GO-Taste oder jede beliebige andere außer "RS" und "ST" - und die drei Räder beginnen sich zu drehen. Jedes Spiel, welches Sie durch Tastendruck starten, kostet 1,-- DM. Stellt sich eine Gewinnkombination ein, wird der entsprechende Betrag zum momentanen Kapital hinzuaddiert.

Der Maximalgewinn beträgt 15,-- DM (Anzeige von 3 Balken). Mit den anderen Glückskombinationen werden Sie sich bestimmt schnell vertraut machen. Meistens werden Sie jedoch verlieren. In Las Vegas ergeht es Ihnen bestimmt auch nicht anders..... Das Spiel ist zu Ende, wenn Ihnen das Kleingeld ausgeht - bekanntlich erhalten nur 80jährige in Begleitung ihrer Eltern Kredit .... es sei denn, Ihre Bank macht Ausnahme, versuchen Sie, das Programm neu zu starten und Sie werden schon sehen... Also dann viel Spaß!

```
 SEITE 1      Einarmiger Bandit    aus DIGITRONIC-6502-ASSEMBLER
Lauf 1 beendet                    0  Fehler
0010       0000     FENST   EQU    $00        (FENSTER)
0020       0007     BTRAG   EQU    $07
0030       0008     PFEIL   EQU    $08
0040       0009     GEWIN   EQU    $09
0050       000A     WAIT    EQU    $0A
0060       000B     TUMBL   EQU    $0B
0070              *
0080       FB4C     TASTE?  EQU    $FB4C
0090       F812     TAB2    EQU    $F812      (HEX:7SEG)
0100       F681     PADD    EQU    $F681
0110       F680     PA      EQU    $F680
0120       F682     PB      EQU    $F682
0130              *
0140 0200                     ORG    $200
0150              *
0160 0200 A9 25    GO         LDA    #$25       (ANFANGSKAPITAL)
0170 0202 85 07               STA    BTRAG
0180 0204 20 C202             JSR    KONV
0190 0207 A9 00               LDA    #$00
0200 0209 85 08               STA    PFEIL
0210 020B 85 02               STA    FENST+2
0220              *
0230 020D 20 8F02 LPA         JSR    DISP       (ANZEIGE BIS GO-TASTE
0240 0210 D0 FB               BNE    LPA        LOSGELASSEN IST)
0250 0212 E6 0B   ROLL        INC    TUMBL
0260 0214 20 8F02             JSR    DISP       (ANZEIGE BIS TASTE
0270 0217 F0 F9               BEQ    ROLL       GEDRUECKT)
0280 0219 A9 03               LDA    #3
0290 021B 85 08               STA    PFEIL
0300 021D F8                  SED
0310 021E 38                  SEC
0320 021F A5 07               LDA    BTRAG
0330 0221 E9 01               SBC    #$01
0340 0223 85 07               STA    BTRAG
0350 0225 20 C202             JSR    KONV
0360 0228 26 0B               ROL    TUMBL
0370              *
0380 022A 20 8F02 LPB         JSR    DISP
0390 022D C6 0A               DEC    WAIT
0400 022F D0 F9               BNE    LPB
0410 0231 A6 08               LDX    PFEIL
0420 0233 A5 0B               LDA    TUMBL      (ERZEUGE EINE
0430 0235 29 06               AND    #$06       KOMBINATION)
0440 0237 09 40               ORA    #$40
0450 0239 95 02               STA    FENST+2,X
0460 023B 46 0B               LSR    TUMBL
0470 023D 46 0B               LSR    TUMBL
0480 023F C6 08               DEC    PFEIL
0490 0241 D0 E7               BNE    LPB
0500              *BERECHNE GEWINN
0510 0243 A5 05               LDA    FENST+5    (PRUEFE AUF
0520 0245 C5 04               CMP    FENST+4    GLEICHHEIT)
0530 0247 D0 37               BNE    NOMA!
0540 0249 C5 03               CMP    FENST+3
0550 024B D0 33               BNE    NOMAT
0560 024D A2 10               LDX    #$10       (DM15,- FUER 3 BALKEN)
0570 024F C9 40               CMP    #$40
0580 0251 F0 0D               BEQ    PAY
```

```
0590 0253 A2 08          LDX    #$8      (DM10,-)
0600 0255 C9 42          CMP    #$42
0610 0257 F0 07          BEQ    PAY
0620 0259 A2 06          LDX    #$6      (DM5,-)
0630 025B C9 44          CMP    #$44
0640 025D F0 01          BEQ    PAY
0650 025F CA             DEX
0660           *GEWONNEN!!! GEWINN IN X-REGISTER
0670 0260 86 09   PAY    STX    GEWIN
0680 0262 A9 80   PAX    LDA    #$80
0690 0264 85 0A          STA    WAIT
0700 0266 20 8F02 LPC    JSR    DISP
0710 0269 C6 0A          DEC    WAIT
0720 026B D0 F9          BNE    LPC
0730 026D C6 09          DEC    GEWIN
0740 026F F0 9C          BEQ    LPA
0750 0271 18             CLC              (ADDIERE DEN GEWINN
0760 0272 F8             SED
0770 0273 A5 07          LDA    BTRAG     ZUM BETRAG)
0780 0275 69 01          ADC    #$01
0790 0277 B0 94          BCS    LPA
0800 0279 85 07          STA    BTRAG
0810 027B 20 C202        JSR    KONV
0820 027E D0 E2          BNE    PAX
0830 0280 A2 03   NOMAT  LDX    #3
0840 0282 C9 46          CMP    #$46      (KLEINER GEWINN?)
0850 0284 F0 DA          BEQ    PAY
0860 0286 20 8F02 LOK    JSR    DISP
0870 0289 A5 07          LDA    BTRAG
0880 028B D0 80          BNE    LPA
0890 028D F0 F7          BEQ    LOK
0900           * ANZEIGEROUTINE*
0910 028F'A6 08'  DISP   LDX    PFEIL
0920 0291 10 02          BPL    INDIS     (TROMMEL DREHEN)
0930 0293 F6 03   OVER   INC    FENST+3,X
0940 0295 CA      INDIS  DEX
0950 0296 10 FB          BPL    OVER
0960 0298 A9 FF          LDA    #$FF
0970 029A 8D 81F6        STA    PADD
0980 029D A0 29          LDY    #$29
0990 029F A2 05          LDX    #$05
1000 02A1 B5 00   LICHT  LDA    FENST,X
1010 02A3 8C 82F6        STY    PB        (FENSTER ANZEIGEN)
1020 02A6 8D 80F6        STA    PA
1030 02A9 D8             CLD
1040 02AA A9 FF          LDA    #$FF
1050 02AC E9 01   ZIP    SBC    #$01
1060 02AE D0 FC          BNE    ZIP
1070 02B0 8D 82F6        STA    PB
1080 02B3 C8             INY
1090 02B4 C8             INY
1100 02B5 CA             DEX
1110 02B6 10 E9          BPL    LICHT
1120 02B8 E8             INX
1130 02B9 8E 81F6        STX    PADD
1140 02BC 20 4CFB        JSR    TASTE?
1150 02BF C9 19          CMP    #$19
1160 02C1 60             RTS
1170           * BETRAG KONVERTIEREN*
1180 02C2 A5 07   KONV   LDA    BTRAG
```

```
SEITE 3         Einarmiger Bandit

1190 02C4 29 0F          AND    #$0F
1200 02C6 AA             TAX
1210 02C7 BD 12F8        LDA    TAB2,X     (UEBERSETZE BETRAG
1220 02CA 85 00          STA    FENST      IN LED-KODE)
1230 02CC A5 07          LDA    BTRAG
1240 02CE 4A             LSR    AC
1250 02CF 4A             LSR    AC
1260 02D0 4A             LSR    AC
1270 02D1 4A             LSR    AC
1280 02D2 AA             TAX
1290 02D3 BD 12F8        LDA    TAB2,X
1300 02D6 85 01          STA    FENST+1
1310 02D8 60             RTS
1320 02D9                END
```

```
SEITE 4         Einarmiger Bandit

FENST   0000     BTRAG   0007     PFEIL   0008     GEWIN   00C
WAIT    000A     TUMBL   000B     TASTE?  FB4C     TAB2    F81
PADD    F681     PA      F680     PB      F682     GO      02C
LPA     020D     ROLL    0212     LPB     022A     PAY     02E
PAX     0262     LPC     0266     NOMAT   0280     LOK     02E
DISP    028F     OVER    0293     INDIS   0295     LICHT   02F
ZIP     02AC     KONV    02C2
Lauf  2 beendet                0  Fehler
```

b. Bauer Brösel ID = ∅ 02

Sie sind Bauer Brösel und Ihre Getreideernte ist in Gefahr. Ameisen (A), Buntspechte (B), Chinchilas (C) - das sind südamerikanische Beutelratten - wollen sich daran guttun. Außerdem strolchen ein Fuchs (F), der Dackel (D) vom Nachbarn sowie der kürzlich aus dem naheliegenden Zirkus ausgebrochene Elefant (E) herum. Sie können die Störenfriede verscheuchen, indem Sie sie bei ihren Namen rufen, d.h. die entsprechenden Tasten A bis F drücken. Die Tiere reagieren nicht auf die falsche Taste, schließlich können Sie ja einen Elefanten nicht mit einer "Vogelscheuche" verjagen!

Wenn Sie sich mit den einzelnen Tieren vertraut gemacht haben, können Sie das Spiel leicht ändern und somit interessanter gestalten:

1. Die Geschwindigkeit, mit der die Tiere über die Anzeige huschen, wird durch den Inhalt der Speicherzelle ∅ 02A0 kontrolliert. Ein größerer bzw. kleinerer Wert beschleunigt oder verlangsamt das Spiel.

2. Die Tiere können nur verscheucht werden, wenn sie vollständig im Getreidefeld (dem Display) sind oder dieses noch nicht vollständig verlassen haben. Sicherlich ist es interessanter und schwieriger, wenn Sie die Tiere nur dann verjagen können, wenn sie sich vollständig im Display befinden. Auf Adresse ∅ 005D (Fangt-1) können Sie prüfen, ob das Tier das Feld verläßt oder auch nicht.

3. Jedes Tier belegt 8 RAM-Speicherplätze. Erfinden Sie Ihre eigenen exotischen Tiere, indem Sie die Daten der entsprechenden Speicheradressen (∅ 0200 bis ∅ 0235) verändern. Die Bits 0 bis 6 einer RAM-Zelle entsprechen fortlaufend den Segmenten A bis G der zugehörigen 7-Segmentanzeige.

   Starten Sie das Spiel mit der Adresse ∅ 0236 und nun viel Spaß.

```
 SEITE 1     Bauer Broesel   auf MICRONIC-650X-ASSEMBLER
Lauf 1 beendet                    0  Fehler
 0010       006F     KORN    EQU     $6F
 0020       0072     DELAY   EQU     $72
 0030       006E     GOT     EQU     $6E
 0040       007D     FLAG    EQU     $7D
 0050       0073     KEY     EQU     $73         (TASTE)
 0060       005E     FENST   EQU     $5E         (FENSTER)
 0070       0066     WINGS   EQU     $66
 0080       0070     POINL   EQU     $70         (POINTER)
 0090       0071     POINH   EQU     $71
 0100       F852     DONE    EQU     $F852
 0110       F686     ZUFAL   EQU     $F686
 0120       FB4C     TASTE?  EQU     $FB4C
 0130       F681     PADD    EQU     $F681
 0140       F680     PA      EQU     $F680
 0150       F682     PB      EQU     $F682
 0160       000B     ZEIT    EQU     $0B
 0170       000A     U1      EQU     $0A
 0180       00FF     PUNKT   EQU     $FF
 0190       FB27     MLICHT  EQU     $FB27
 0200 0200           *       ORG     $200
 0210                *
 0220 0200 06        INDEX   FCB     $06,$0E,$16,$1E,$26,$2E
      0201 0E
      0202 16
      0203 1E
      0204 26
      0205 2E
 0230 0206 08                FCB     $08,$00,$00,$00,$00,$00,$00,$00
      0207 00
      0208 00
      0209 00
      020A 00
      020B 00
      020C 00
      020D 00
 0240 020E 01                FCB     $01,$39,$39,$08,$00,$00,$00,$00
      020F 39
      0210 39
      0211 08
      0212 00
      0213 00
      0214 00
      0215 00
 0250 0216 58                FCB     $58,$68,$4C,$40,$00,$00,$00,$00
      0217 68
      0218 4C
      0219 40
      021A 00
      021B 00
      021C 00
      021D 00
 0260 021E 63                FCB     $63,$58,$4F,$00,$00,$00,$00,$00
      021F 58
      0220 4F
      0221 00
      0222 00
      0223 00
      0224 00
```

```
     0225 00
0270 0226 71              FCB    $71,$1D,$41,$1F,$01,$00,$00,$00
     0227 1D
     0228 41
     0229 1F
     022A 01
     022B 00
     022C 00
     022D 00
0280 022E 63              FCB    $63,$38,$4C,$40,$00,$00,$00,$00
     022F 38
     0230 4C
     0231 40
     0232 00
     0233 00
     0234 00
     0235 00
0290               *
0300               *
0310 0236 A2 18   START   LDX    ##18       (GETREIDE)
0320 0238 86 6F           STX    KORN
0330 023A A9 00           LDA    #0
0340 023C 95 56   SLOOP   STA    FENST-8,X (FENSTER LOESCHEN)
0350 023E CA              DEX
0360 023F 10 FB           BPL    SLOOP
0370 0241 A2 0E   TEST    LDX    ##0E
0380 0243 B5 5E   TLOOP   LDA    FENST,X    (FENSTER LEER?)
0390 0245 D0 3B           BNE    WEITER
0400 0247 CA              DEX
0410 0248 10 F9           BPL    TLOOP
0420 024A E6 6E           INC    GOT        (JA:ERZEUGE NEUES TIER)
0430 024C A5 7D           LDA    FLAG
0440 024E F0 09           BEQ    MEHR
0450 0250 C6 6E           DEC    GOT
0460 0252 C6 6F           DEC    KORN
0470 0254 D0 03           BNE    MEHR
0480 0256 4C 52F8         JMP    DONE
0490               *
0500 0259 AD 86F6 MEHR    LDA    ZUFAL      (ZUFALLSZAHL)
0510 025C 4A              LSR    AC
0520 025D 4A              LSR    AC
0530 025E 4A              LSR    AC
0540 025F 4A              LSR    AC
0550 0260 4A              LSR    AC
0560 0261 C9 06           CMP    ##06       (6 TIERE)
0570 0263 90 02           BCC    ERZ
0580 0265 29 03           AND    ##03
0590 0267 18        ERZ   CLC
0600 0268 AA              TAX                (TIERART NACH X)
0610 0269 69 0A           ADC    ##0A
0620 026B 85 73           STA    KEY        (TASTE DEKODIEREN)
0630               *
0640 026D BC 0002         LDY    INDEX,X
0650 0270 84 70           STY    POINL
0660 0272 A9 02           LDA    #2
0670 0274 85 71           STA    POINH      ("TIERADRESSE" NACH POINTER)
0680 0276 A0 07           LDY    #7
0690 0278 B1 70   ALOOP   LDA    (POINL),Y
0700 027A 99 6600         STA    WINGS,Y
0710 027D 88              DEY
```

```
0720 027E 10 F8          BPL    ALOOP
0730 0280 84 7D          STY    FLAG       (FLAG=FF:TIER KOMMT)
0740              *
0750 0282 A2 07   WEITER LDX    #7
0760 0284 B5 66   CLOOP  LDA    WINGS,X    (TIER IM DISPLAY?)
0770 0286 D0 21          BNE    NOKEY      (NEIN:IGNORIERE TASTE)
0780 0288 CA             DEX
0790 0289 10 F9          BPL    CLOOP
0800 028B A9 00          LDA    #0
0810 028D 8D 81F6        STA    PADD
0820 0290 A9 10          LDA    #$10
0830 0292 85 0B          STA    ZEIT
0840 0294 20 4CFB TZYK   JSR    TASTE?
0850 0297 C9 19          CMP    #$19
0860 0299 F0 0E          BEQ    NOKEY
0870 029B C6 0B   NOCH   DEC    ZEIT
0880 029D D0 F5          BNE    TZYK
0900              *  TASTE ENTPRELLT!
0910 029F C5 73          CMP    KEY        (RICHTIGES TIER GENANNT?)
0920 02A1 D0 06          BNE    NOKEY
0930 02A3 A5 7D          LDA    FLAG
0940 02A5 10 02          BPL    NOKEY      (GEHT TIER ZURUECK?)
0950 02A7 E6 7D          INC    FLAG       (SCHEUCHE TIER ZURUECK)
0960 02A9 C6 72   NOKEY  DEC    DELAY
0970 02AB D0 20          BNE    NOMOVE     (WARTESCHLEIFE)
0980 02AD A9 20          LDA    #$20
0990 02AF 85 72          STA    DELAY
1000 02B1 A5 7D          LDA    FLAG       (BEWEGE TIER
1010 02B3 30 0D          BMI    KOMM       NACH LINKS
1020 02B5 A2 0E          LDX    #$0E       NACH RECHTS)
1030 02B7 B5 56   RLOOP  LDA    FENST-8,X
1040 02B9 95 57          STA    FENST-7,X
1050 02BB CA             DEX
1060 02BC D0 F9          BNE    RLOOP
1070 02BE 86 56          STX    FENST-8
1080 02C0 F0 0B          BEQ    NOMOVE     UNBED. SPRUNG
1090              *
1100 02C2 A2 EC   KOMM   LDX    #$EC
1110 02C4 B5 6E   CMLP   LDA    FENST+16,X
1120 02C6 95 6D          STA    FENST+15,X
1130 02C8 E8             INX
1140 02C9 30 F9          BMI    CMLP
1150              *
1160 02CB A2 00          LDX    #0
1170 02CD 86 FF   NOMOVE STX    PUNKT
1180 02CF A2 37          LDX    #$37
1190 02D1 A0 07          LDY    #7
1200 02D3 B9 5E00 LICHT  LDA    FENST,Y
1210 02D6 84 0A          STY    U1         (8 ANZEIGEZIFFERN)
1220 02D8 20 27FB        JSR    MLICHT
1230 02DB CA             DEX
1240 02DC CA             DEX
1250 02DD CA             DEX
1260 02DE CA             DEX
1270 02DF A4 0A          LDY    U1
1280 02E1 88             DEY
1290 02E2 10 EF          BPL    LICHT
1300 02E4 4C 4102        JMP    TEST
1310 02E7                END
```

KORN	006F	DELAY	0072	GOT	006E	FLAG	007D
KEY	0073	FENST	005E	WINGS	0066	POINL	0070
POINH	0071	DONE	F852	ZUFAL	F686	TASTE?	F84C
PADD	F681	PA	F680	PB	F682	ZEIT	000B
U1	000A	PUNKT	00FF	MLICHT	FB27	INDEX	0200
START	0236	SLOOP	023C	TEST	0241	TLOOP	0243
MEHR	0259	ERZ	0267	ALOOP	0278	WEITER	0282
CLOOP	0284	TZYK	0294	NOCH	029B	NOKEY	02A9
RLOOP	02B7	KOMM	02C2	CMLP	02C4	NOMOVE	02CD
LICHT	02D3						

Lauf 2 beendet 0 Fehler

c. Verschieben von Daten im RAM ID = $ 03

Dieses Programm soll Ihnen beim Schreiben Ihrer eigenen Programme ein echtes Hilfsmittel sein. Sie werden sicher einmal in die Verlegenheit kommen, ein altes Programm überschreiben zu müssen, wollen es aber vorher retten. Sie können dazu einmal die Tonbandroutinen verwenden oder es in einen anderen RAM-Speicher umladen. Dazu speichern Sie die Anfangs- und Endadressen Ihres Programmes wie folgt ab:

niederwertige Anfangsadresse nach $ 00D0
höherwertige Anfangsadresse nach $ 00D1
niederwertige Endadresse nach $ 00D2
höherwertige Endadresse nach $ 00D3

niederwertige Anfangsadresse des neuen Speicherbereiches nach $ 00D4
höherwertige Anfangsadresse des neuen Speicherbereiches nach $ 00D5

Starten Sie das Verschiebeprogramm mit der Adresse $ F780.

Da erfahrungsgemäß Anwenderprogramme im unteren 1K RAM-Speicherbereich abgelegt werden, wurde dieses in einen Bereich gelegt, der seltener benutzt wird. Es wurde jedoch so geschrieben, daß es sich ohne weiteres selbst in einen anderen Speicherbereich umladen kann. Die vorgenannten Parameteradressen auf der Seite 0 bleiben jedoch erhalten.

Es sei noch angemerkt, daß durch das Umladen das alte Programm nicht zerstört wird, es wird nur dupliziert.

```
 SEITE 1      TRANS   auf MICROMIC-65XX-ASSEMBLER
Lauf 1 beendet                    0  Fehler
 0010 F780                  ORG    $F700
 0020       00D0    AANFL   EQU    $D0
 0030       00D1    AANFH   EQU    $D1
 0040       00D2    AENDEL  EQU    $D2
 0050       00D3    AENDEH  EQU    $D3
 0060       00D4    NANFL   EQU    $D4
 0070       00D5    NANFH   EQU    $D5
 0080 F780 A2 00            LDX    #$00
 0090 F782 D8               CLD
 0100 F783 A1 D0    LOOP    LDA    (AANFL,X) DATA DER LAUFENDEN ADRESSE
 0110 F785 81 D4            STA    (NANFL,X) UMGESPEICHERT
 0120 F787 18               CLC
 0130 F788 A5 D0            LDA    AANFL
 0140 F78A 69 01            ADC    #$01
 0150 F78C 85 D0            STA    AANFL
 0160 F78E A5 D1            LDA    AANFH
 0170 F790 69 00            ADC    #$00
 0180 F792 85 D1            STA    AANFH      ALTE ANF.ADR.ERHOEHT
 0190 F794 A5 D4            LDA    NANFL
 0200 F796 18               CLC
 0210 F797 69 01            ADC    #$01
 0220 F799 85 D4            STA    NANFL
 0230 F79B A5 D5            LDA    NANFH
 0240 F79D 69 00            ADC    #$00
 0250 F79F 85 D5            STA    NANFH      NEUE ANF.ADR.ERHOEHT
 0260 F7A1 38               SEC
 0270 F7A2 A5 D2            LDA    AENDEL
 0280 F7A4 E5 D0            SBC    AANFL
 0290 F7A6 A5 D3            LDA    AENDEH
 0300 F7A8 E5 D1            SBC    AANFH
 0310 F7AA 10 D7            BPL    LOOP       ALTE END.-ANF.ADRESSE
 0320 F7AC 4C 52F8          JMP    $F852      IST POSITIV
 0330 F7AF                  END


 SEITE 2        TRANS

 AANFL   00D0        AANFH   00D1        AENDEL  00D2        AENDEH  00D3
 NANFL   00D4        NANFH   00D5        LOOP    F783
 Lauf 2 beendet                0  Fehler
```

Kapitel 11

Technischer Anhang

11.1 Listing des Betriebssystems MONA

```
 SEITE 1     MONA A 1.10.77   auf MICRONIC-650X-ASSEMBLER
Lauf 1 beendet                  0  Fehler
 0010      FD92    MNEMON EQU    $FD92
 0020      FE96    TAPOUT EQU    $FE96
 0030      F680    PA     EQU    $F680
 0040      F681    PADD   EQU    $F681
 0050      F682    PB     EQU    $F682
 0060      F683    PBDD   EQU    $F683
 0070      F694    T:1    EQU    $F694
 0080      F696    T:64   EQU    $F696
 0090      F697    T:1024 EQU    $F697
 0100      F687    VORBEI EQU    $F687
 0110      F686    TREAD  EQU    $F686
 0120      F7FF    IRQH   EQU    $F7FF      BREAKVEKTOR
 0130      F7FE    IRQL   EQU    $F7FE
 0140      F7FB    NMIH   EQU    $F7FB      STOPVEKTOR
 0150      F7FA    NMIL   EQU    $F7FA
 0160      F7F9    TAHIGH EQU    $F7F9
 0170      F7F8    TALOW  EQU    $F7F8
 0180      F7F7    TASTE  EQU    $F7F7
 0190      F7F6    BAUDH  EQU    $F7F6
 0200      F7F5    BAUDL  EQU    $F7F5
 0210      F7F4    ID     EQU    $F7F4
 0220      F7F3    ENDEH  EQU    $F7F3
 0230      F7F2    ENDEL  EQU    $F7F2
 0240      F7F1    STRTH  EQU    $F7F1
 0250      F7F0    STRTL  EQU    $F7F0
 0260      F7EF    ECHO   EQU    $F7EF
 0280           *ZEROPAGE ADRESSEN DES BETRIEBSSYSTEMS
 0290      00FF    PUNKT  EQU    $FF
 0300      00FE    COPO   EQU    $FE
 0310      00FD    MNEM3  EQU    $FD
 0320      00FC    FREQ2  EQU    $FC
 0330      00FB    MNEM2  EQU    $FB
 0340      00FA    FREQ1  EQU    $FA
 0350      00F9    MNEM1  EQU    $F9
 0360      00F8    ADRH   EQU    $F8
 0370      00F7    ADRL   EQU    $F7
 0380      00F6    CHECKH EQU    $F6
 0390      00F5    CHECKL EQU    $F5
 0400      00F4    OBUF   EQU    $F4
 0410      00F3    IBUF   EQU    $F3
 0420      00F2    INH    EQU    $F2
 0430      00F1    INL    EQU    $F1
 0440      00F0    ADH    EQU    $F0
 0450      00EF    ADL    EQU    $EF
 0460      00EE    DSPART EQU    $EE
 0470      00ED    Z2     EQU    $ED
 0480      00EC    Z1     EQU    $EC
 0490      00EB    YSAVE  EQU    $EB
 0500      00EA    XSAVE  EQU    $EA
 0510      00E9    STATUS EQU    $E9
 0520      00E8    STPOIN EQU    $E8
 0530      00E7    AKKU   EQU    $E7
 0540      00E6    SEC    EQU    $E6
 0550      00E5    ZEIT   EQU    $E5
 0560      00E4    SEC:4  EQU    $E4
 0565      00E3    TPFLG  EQU    $E3
 0570      00E0    MIN    EQU    $E0
 0580      00DA    STD    EQU    $DA
```

```
0590            *
0600 F800               ORG     $F800
0610 F800 6C FAF7       JMP     (NMIL)
0620 F803 6C FEF7       JMP     (IRQL)
0630 F806 20       TAB0 FCC     " ANON"
     F807 41
     F808 4E
     F809 4F
     F80A 4D
0640 F80B 00       TAB1 FCB     $00,$00,$00,$00,$00,$0A,$0D
     F80C 00
     F80D 00
     F80E 00
     F80F 00
     F810 0A
     F811 0D
0650 F812 3F       TAB2 FCB     $3F,$06,$5B,$4F,$66,$6D,$7D,$07
     F813 06
     F814 5B
     F815 4F
     F816 66
     F817 6D
     F818 7D
     F819 07
0660 F81A 7F            FCB     $7F,$6F,$77,$7C,$39,$5E,$79,$71
     F81B 6F
     F81C 77
     F81D 7C
     F81E 39
     F81F 5E
     F820 79
     F821 71
0670 F822 A2 FF   RESTRT LDX    #$FF
0680 F824 9A             TXS
0690 F825 86 E8          STX    STPOIN
0700 F827 E8             INX
0710 F828 86 E9          STX    STATUS
0720 F82A A9 06          LDA    #06
0730 F82C 85 FA          STA    FREQ1
0740 F82E A9 09          LDA    #$09
0750 F830 85 FC          STA    FREQ2
0760 F832 A9 4B          LDA    #$4B
0770 F834 8D F5F7        STA    BAUDL     SETZEN VON 110 BAUD
0780 F837 A9 8C          LDA    #$8C
0790 F839 8D F6F7        STA    BAUDH
0800 F83C A2 F9          LDX    #$F9
0810 F83E 8E F9F7        STX    TAHIGH    TASTENVEKTOR AUF MONA
0820 F841 CA             DEX
0830 F842 8E FBF7        STX    NMIH
0840 F845 8E FFF7        STX    IRQH
0850 F848 A9 52          LDA    #$52
0860 F84A 8D FAF7        STA    NMIL      VEKTOREN ZEIGEN AUF STOP
0870 F84D 8D FEF7        STA    IRQL
0880 F850 D0 16          BNE    STP
0890 F852 85 E7   STOP   STA    AKKU      STOP EINGANG -ACCU RETTEN
0900 F854 68             PLA
0910 F855 85 E9          STA    STATUS
0920 F857 68             PLA
0930 F858 85 EF          STA    ADL       RETTEN DER BENUTZER-ADRESSE
0940 F85A 85 F7          STA    ADRL
```

```
0950 F85C 68              PLA
0960 F85D 85 F0           STA    ADH
0970 F85F 85 F8           STA    ADRH
0980 F861 86 EA           STX    XSAVE
0990 F863 84 EB           STY    YSAVE
1000 F865 BA              TSX
1010 F866 86 E8           STX    STPOIN
1020 F868 A9 00   STP     LDA    #00
1030 F86A 8D EFF7         STA    ECHO       SYSTEHMELDUNG EIN
1040 F86D 8D 81F6         STA    PADD       PORT INITIALISIEREN
1050 F870 48              PHA               STATUS INIT.
1060 F871 28              PLP
1070 F872 A9 3F           LDA    #$3F
1080 F874 8D 82F6         STA    PB
1090 F877 8D 83F6         STA    PBDD
1100 F87A 2C 82F6         BIT    PB
1110 F87D 50 1A           BVC    DISP
1120 F87F A9 FF   SLOWST  LDA    #$FF
1130 F881 85 E5           STA    ZEIT
1140 F883 2C 80F6         BIT    PA         TTY?
1150 F886 70 03           BVS    SLOW1
1160 F888 20 5BFA         JSR    ZEILE      EINE ZEILE AUSIRUCKEN
1170 F88B 2C 80F6 SLOW1   BIT    PA
1180 F88E 50 03           BVC    SLOW2
1190 F890 20 D6FA         JSR    DISPLY     MULTIPLEX
1200 F893 C6 E5   SLOW2   DEC    ZEIT
1210 F895 D0 F4           BNE    SLOW1
1220 F897 F0 5B           BEQ    GO
1230 F899 A9 C0   DISP    LDA    #%11000000 Z=1:HEXA;V=1:PDR
1240 F89B 85 EE           STA    DSPART
1245 F89D 85 E5           STA    ZEIT
1260 F89F E6 E5   TZ      INC    ZEIT
1262 F8A1 2C 80F6 DTCYC   BIT    PA         DISPLAY/TASTE-ZYCLUS
1270 F8A4 50 6F           BVC    TTY
1280 F8A6 20 D6FA         JSR    DISPLY
1290 F8A9 20 4CFB         JSR    TASTE?
1300 F8AC C9 19           CMP    #$19       =KEINE TASTE
1310 F8AE D0 04           BNE    NOCH
1320 F8B0 A9 0A           LDA    #10        ENTPRELLZEIT
1330 F8B2 85 E5           STA    ZEIT
1340 F8B4 C6 E5   NOCH    DEC    ZEIT       TASTE NOCH GEDRUECKT
1350 F8B6 30 E7           BMI    TZ
1360 F8B8 D0 E7           BNE    DTCYC
1370 F8BA C9 10           CMP    #16
1380 F8BC 10 25           BPL    FUNK       FUNKTIONSTASTE
1390 F8BE 0A              ASL    AC
1400 F8BF 0A              ASL    AC
1410 F8C0 0A              ASL    AC
1420 F8C1 0A              ASL    AC
1430 F8C2 8D F7F7         STA    TASTE
1440 F8C5 A2 04           LDX    #04
1450 F8C7 A0 00           LDY    #00
1460 F8C9 24 EE   ART?    BIT    DSPART
1470 F8CB 70 0E           BVS    ADRES
1480 F8CD B1 F7   DATA    LDA    (ADRL),Y
1490 F8CF 0E F7F7         ASL    TASTE
1500 F8D2 2A              ROL    AC
1510 F8D3 91 F7           STA    (ADRL),Y
1520 F8D5 CA      DATA1   DEX
1530 F8D6 D0 F1           BNE    ART?
```

```
1540 F8D3 4C A1F8 CYCEND JMP    DTCYC
1550 F8DB 0A     ADRES  ASL    AC
1560 F8DC 26 F7         ROL    ADRL
1570 F8DE 26 F8         ROL    ADRH
1580 F8E0 4C D5F8       JMP    DATA1
1590 F8E3 C9 18  FUNK   CMP    #$18      GO
1600 F8E5 F0 0D         BEQ    GO
1610 F8E7 0A            ASL    AC        *2
1620 F8E8 29 0F         AND    #$0F      MASKIERUNG HIGH
1630 F8EA 18            CLC
1640 F8EB 69 94         ADC    #TSATZ-$F900 PRESET MONA-TASTEN
1650 F8ED 8D F8F7       STA    TALOW
1660 F8F0 38            SEC
1670 F8F1 6C F8F7       JMP    (TALOW)
1680            *
1690 F8F4 A6 E8  GO     LDX    STPOIN    UEBERTRAGEN DER BENUTZER-
1700 F8F6 9A            TXS               REGISTER
1710 F8F7 A5 F8         LDA    ADRH
1720 F8F9 48            PHA
1730 F8FA A5 F7         LDA    ADRL
1740 F8FC 48            PHA
1750 F8FD A5 E9         LDA    STATUS
1760 F8FF 48            PHA
1770 F900 A6 EA         LDX    XSAVE
1780 F902 A4 EB         LDY    YSAVE
1790 F904 A5 E7         LDA    AKKU
1800 F906 40            RTI               SPRUNG ZUM BENUTZER-PROG.
1810            *
1820 F907 20 48FC SPEED JSR    HEXIN     1.WERT FUER SPEED LESEN
1830 F90A 48            PHA
1840 F90B 20 48FC       JSR    HEXIN
1850 F90E 8D F6F7       STA    BAUDH
1860 F911 68            PLA
1870 F912 8D F5F7       STA    BAUDL
1880 F915 A9 00  TTY    LDA    #00
1890 F917 85 C3         STA    TPFLG
1900 F919 2C EFF7       BIT    ECHO
1910 F91C 30 0B         BMI    TTY0
1930 F91E A2 0B  MONA   LDX    #11
1940 F920 BD 86F8 MONA1 LDA    TAB0,X
1950 F923 20 86FA       JSR    TTYOUT
1960 F926 CA            DEX
1970 F927 10 F7         BPL    MONA1
1975 F929 30 0D         BMI    MONA2
1980 F92B 2C 80F6 TTY0  BIT    PA        WENN PA6=1 (=V)
1990 F92E 70 A8         BVS    CYCEND    NACH DISPL/TASTE
2000 F930 2C EFF7       BIT    ECHO
2010 F933 30 06         BMI    TTY1
2020 F935 20 4FFA       JSR    CARRET
2025 F938 20 5EFA MONA2 JSR    ZEILE0
2030 F93B A9 00  TTY1   LDA    #00
2040 F93D 85 F1         STA    INL       LOESCHEN INPUT
2050 F93F 85 F2         STA    INH
2060 F941 20 F1F9 TTY2  JSR    ASCIIN    CHARACTER ERWARTEN
2070 F944 2C 80F6       BIT    PA
2080 F947 70 8F         BVS    CYCEND    UMGESCHALTET?
2090 F949 C9 2E         CMP    #'.       ZELLE AENDERN
2100 F94B F0 37         BEQ    NEUDAT
2110 F94D C9 0D         CMP    #$0D      RETURN=NEUE ADRESSE
2120 F94F F0 39         BEQ    TDOWN
```

```
2130 F951 C9 0A          CMP    #$0A       LINEF.=VORIGE ADRESSE
2140 F953 F0 3A          BEQ    TUP
2150 F955 C9 7F          CMP    #$7F       RUBOUT=SYSTEMMELDUNG
2160 F957 F0 C5          BEQ    MONA
2170 F959 C9 20          CMP    #$20       LEERTASTE = NEUE ADRESSE
2180 F95B F0 1C          BEQ    ADR
2190 F95D C9 53          CMP    #'S        SPEED AENDERN
2200 F95F F0 A6          BEQ    SPEED
2210 F961 C9 47          CMP    #'G        GO
2220 F963 F0 8F          BEQ    GO
2230 F965 C9 4C          CMP    #'L        LADEN LOCHSTREIFEN
2240 F967 F0 0A          BEQ    GLADEN
2250 F969 C9 51          CMP    #'Q        STANZEN LOCHSTREIFEN
2260 F96B F0 09          BEQ    GDUMP
2270 F96D 20 2DFA        JSR    HEXA
2280 F970 4C 41F9        JMP    TTY2
2290 F973 4C F9FB GLADEN JMP    LADEN
2300 F976 4C 7DFB GDUMP  JMP    DUMP
2310               *
2320 F979 A5 F1   ADR    LDA    INL
2330 F97B 85 F7          STA    ADRL
2340 F97D A5 F2          LDA    INH
2350 F97F 85 F8          STA    ADRH
2360 F981 4C 2BF9        JMP    TTY0
2370               *
2380 F984 A0 00   NEUDAT LDY    #00
2390 F986 A5 F1          LDA    INL
2400 F988 91 F7          STA    (ADRL),Y
2410 F98A 20 55FC TDOWN  JSR    INCADR
2420 F98D B0 9C          BCS    TTY0
2430 F98F 20 5DFC TUP    JSR    DECADR
2440 F992 B0 97   R3     BCS    TTY0
2450               *
2460 F994 B0 F4   TSATZ  BCS    TDOWN
2470 F996 B0 F7          BCS    TUP
2480 F998 B0 0A          BCS    TMNEMO
2490 F99A B0 0E          BCS    TPROGC
2500 F99C B0 1A          BCS    TAKKU
2510 F99E B0 14          BCS    TSTATU
2520 F9A0 B0 26          BCS    TADRES
2530 F9A2 B0 1E          BCS    TDATA
2540               *
2550 F9A4 A9 00   TMNEMO LDA    #%00000000 DATA/MNEMONIC
2560 F9A6 85 EE          STA    DSPART
2570 F9A8 B0 E8          BCS    R3
2580 F9AA A5 EF   TPROGC LDA    ADL
2590 F9AC 85 F7          STA    ADRL
2600 F9AE A5 F0          LDA    ADH
2610 F9B0 85 F8          STA    ADRH
2620 F9B2 B0 DE          BCS    R3
2630 F9B4 A9 E9   TSTATU LDA    #STATUS
2640 F9B6 B0 02          BCS    STORE
2650 F9B8 A9 E7   TAKKU  LDA    #AKKU
2660 F9BA 85 F7   STORE  STA    ADRL
2670 F9BC A9 00          LDA    #00
2680 F9BE 85 F8          STA    ADRH
2690 F9C0 B0 D0          BCS    R3
2700 F9C2 A9 80   TDATA  LDA    #%10000000 DATA/HEXA
2710 F9C4 85 EE          STA    DSPART
2720 F9C6 B0 CA          BCS    R3
```

```
2730 F9C8 A9 40   TADRES LDA   #%01000000 ADRESSE/XX
2740 F9CA 05 EE          ORA   DSPART
2750 F9CC 85 EE          STA   DSPART
2760 F9CE B0 C2          BCS   R3
2770              *
2780 F9D0 48      HPAUSE PHA
2790 F9D1 AD F6F7        LDA   BAUDH
2800 F9D4 4A             LSR   AC        /2
2810 F9D5 4C DCF9        JMP   PAUSE0
2820 F9D8 48      PAUSE  PHA
2830 F9D9 AD F6F7        LDA   BAUDH
2840 F9DC 8D 96F6 PAUSE0 STA   T:64
2850 F9DF 2C 87F6 PAUSE1 BIT   VORBEI
2860 F9E2 10 FB          BPL   PAUSE1
2870 F9E4 AD F5F7        LDA   BAUDL
2880 F9E7 8D 94F6        STA   T:1
2890 F9EA 2C 87F6 PAUSE2 BIT   VORBEI
2900 F9ED 10 FB          BPL   PAUSE2
2910 F9EF 68             PLA
2920 F9F0 60             RTS
2930              *
2940 F9F1 2C 80F6 ASCIIN BIT   PA
2950 F9F4 70 2C          BVS   R1        UMGESCHALTET
2960 F9F6 30 F9          BMI   ASCIIN    NOCH KEIN STARTBIT
2970 F9F8 A9 01          LDA   #01       PORT B ROLLBEREIT
2980 F9FA 8D 83FG        STA   PBDD
2990 F9FD 20 D0F9        JSR   HPAUSE    0.5 BIT WARTEZEIT
3000 FA00 0E 82F6        ASL   PB        STARTBIT ECHO
3010 FA03 20 D8F9        JSR   PAUSE     1 BIT WARTEZEIT
3020 FA06 A0 07          LDY   #07       FUER 7 ASCII ZEICHEN
3030 FA08 2E 80F6 ASC1   ROL   PA
3040 FA0B 66 F3          ROR   IBUF
3050 FA0D 2E 80F6        ROL   PA
3060 FA10 2E 82F6        ROL   PB
3070 FA13 20 D8F9        JSR   PAUSE
3080 FA16 88             DEY
3090 FA17 D0 EF          BNE   ASC1
3100 FA19 46 F3          LSR   IBUF      FUER PARITAET
3110 FA1B 38             SEC
3120 FA1C 2E 82F6        ROL   PB
3130 FA1F 20 D8F9        JSR   PAUSE     0.5 BIT GESCHENKT
3140 FA22 A9 3F   R1     LDA   #$3F
3150 FA24 8D 83F6        STA   PBDD
3160 FA27 8D 82F6        STA   PB
3170 FA2A A5 F3          LDA   IBUF
3180 FA2C 60             RTS
3190              *
3200 FA2D C9 30   HEXA   CMP   #'0       KLEINER ALS ASCII "0" ?
3210 FA2F 30 1D          BMI   R2
3220 FA31 C9 47          CMP   #'G       GROESSER ALS ASCII "F" ?
3230 FA33 10 19          BPL   R2
3240 FA35 C9 41          CMP   #'A       ALPHA-ZEICHEN ?
3250 FA37 30 03          BMI   ZAHL
3260 FA39 18             CLC
3270 FA3A 69 09          ADC   #09       HEXA KORREKTUR
3280 FA3C 0A      ZAHL   ASL   AC
3290 FA3D 0A             ASL   AC
3300 FA3E 0A             ASL   AC
3310 FA3F 0A             ASL   AC
3320 FA40 A0 04          LDY   #04
```

```
3330 FA42 2A       TRANS  ROL    AC
3340 FA43 26 F1           ROL    INL
3350 FA45 26 F2           ROL    INH
3360 FA47 88              DEY
3370 FA48 D0 F8           BNE    TRANS
3380 FA4A A5 F1           LDA    INL
3390 FA4C A0 00           LDY    #00
3400 FA4E 60       R2     RTS
3410          *
3420 FA4F A2 06   CARRET LDX    #06
3430 FA51 BD 0BF8 CARET1 LDA    TAB1,X
3440 FA54 20 86FA         JSR    TTYOUT
3450 FA57 CA              DEX
3460 FA58 D0 F7           BNE    CARET1
3470 FA5A 60              RTS
3480          *
3490 FA5B 20 4FFA ZEILE  JSR    CARRET
3500 FA5E A2 03   ZEILE0 LDX    #03
3510 FA60 20 D2FB         JSR    PRTSQ2
3520 FA63 20 84FA         JSR    SPACE
3530 FA66 A0 00           LDY    #00
3540 FA68 B1 F7           LDA    (ADRL),Y
3550 FA6A 20 B2FA         JSR    OUTBT
3560 FA6D 20 84FA         JSR    SPACE
3570 FA70 20 92FD         JSR    MNEMON
3580 FA73 A2 00           LDX    #00
3590 FA75 B5 F9   ZEILE1 LDA    MNEM1,X
3600 FA77 20 86FA         JSR    TTYOUT
3610 FA7A E8              INX
3620 FA7B E8              INX
3630 FA7C E0 06           CPX    #$06
3640 FA7E D0 F5           BNE    ZEILE1
3650 FA80 20 84FA         JSR    SPACE
3660 FA83 60              RTS
3670          *
3680 FA84 A9 20   SPACE  LDA    #$20
3690 FA86 A0 01   TTYOUT LDY    #$01
3700 FA88 8C 83F6         STY    PBDD
3710 FA8B 8C 82F6         STY    PB
3720 FA8E A0 08           LDY    #$08
3730 FA90 20 D8F9         JSR    PAUSE     EVTL 11.BIT
3740 FA93 CE 82F6         DEC    PB
3750 FA96 20 D8F9 ASC01  JSR    PAUSE
3760 FA99 4A              LSR    AC
3770 FA9A 2E 82F6         ROL    PB
3780 FA9D 88              DEY
3790 FA9E D0 F6           BNE    ASC01
3800 FAA0 A9 3F           LDA    #$3F
3810 FAA2 8D 82F6         STA    PB
3820 FAA5 8D 83F6         STA    PBDD
3830 FAA8 20 D8F9         JSR    PAUSE     PARITAETSBIT
3840 FAAB 20 D8F9         JSR    PAUSE
3850 FAAE 60              RTS
3860          *
3870 FAAF 20 EBFB OUTBYT JSR    CHECK
3880 FAB2 48       OUTBT  PHA
3890 FAB3 4A              LSR    AC
3900 FAB4 4A              LSR    AC
3910 FAB5 4A              LSR    AC
3920 FAB6 4A              LSR    AC
```

```
3930 FAB7 20 BFFA        JSR    HEXOUT
3940 FABA 68             PLA
3950 FABB 20 BFFA        JSR    HEXOUT
3960 FABE 60             RTS
3970              *
3980 FABF 29 0F   HEXOUT AND    #$0F      NUR LOWER BYTE
3990 FAC1 C9 0A          CMP    #$0A      ALPHA-ZEICHEN?
4000 FAC3 18             CLC
4010 FAC4 30 02          BMI    NUMN
4020 FAC6 69 07          ADC    #07
4030 FAC8 69 30   NUMN   ADC    #$30
4040 FACA 24 E3          BIT    TPFLG     TAPEAUSGANG=FF,TTYAUSGANG=00
4050 FACC 30 04          BMI    TAPE
4060 FACE 20 06FA        JSR    TTYOUT
4070 FAD1 60             RTS
4080 FAD2 20 96FE TAPE   JSR    TAPOUT
4090 FAD5 60             RTS
4100              *
4110 FAD6 A2 00   DISPLY LDX    #$00
4120 FAD8 86 FF          STX    PUNKT     PUNKT LOESCHEN
4130 FADA A2 29          LDX    #$29      FUER LSD
4140 FADC 24 EE          BIT    DSPART
4150 FADE 50 02          BVC    DSP1
4160 FAE0 C6 FF          DEC    PUNKT     AUF FF I.E.MINUS
4170 FAE2 A5 F8   DSP1   LDA    ADRH
4180 FAE4 20 1BFB        JSR    LLICHT
4190 FAE7 20 21FB        JSR    RLICHT
4200 FAEA A5 F7          LDA    ADRL
4210 FAEC 20 1BFB        JSR    LLICHT
4220 FAEF 20 21FB        JSR    RLICHT
4230 FAF2 E8             INX
4240 FAF3 E8             INX
4250 FAF4 A0 00          LDY    #00
4260 FAF6 84 FF          STY    PUNKT     PUNKT LOESCHEN
4270 FAF8 B1 F7          LDA    (ADRL),Y
4280 FAFA 24 EE          BIT    DSPART
4290 FAFC 70 02          BVS    DSP2
4300 FAFE C6 FF          DEC    PUNKT     PUNKT SETZEN
4310 FB00 24 EE   DSP2   BIT    DSPART
4320 FB02 10 0A          BPL    MNDSP
4330 FB04 20 1BFB        JSR    LLICHT
4340 FB07 20 21FB        JSR    RLICHT
4350 FB0A 24 EE          BIT    DSPART
4360 FB0C 30 0C          BMI    R5
4370 FB0E 20 92FD MNDSP  JSR    MNEMON
4380 FB11 B5 C6   DSP3   LDA    $C6,X
4390 FB13 20 27FB        JSR    MLICHT
4400 FB16 E0 39          CPX    #$39
4410 FB18 D0 F7          BNE    DSP3
4420 FB1A 60        R5   RTS
4430              *
4440 FB1B 85 F4   LLICHT STA    OBUF      LINKES HALBBYTE DES ACCU NACH
4450 FB1D 4A             LSR    AC        DISLPAY LAUT X
4460 FB1E 4A             LSR    AC
4470 FB1F 4A             LSR    AC
4480 FB20 4A             LSR    AC
4490 FB21 29 0F   RLICHT AND    #$0F
4500 FB23 A8             TAY               SIEBENSEGMENT-TABELLE
4510 FB24 B9 12F8        LDA    TAB2,Y    HOLEN
4520 FB27 24 FF   MLICHT BIT    PUNKT
```

```
4530 FB29 10 02          BPL    LICHT1
4540 FB2B 09 80          ORA    #%10000000
4550 FB2D 8D 80F6 LICHT1 STA    PA
4560 FB30 A0 FF          LDY    ##FF
4570 FB32 8C 81F6        STY    PADD
4580 FB35 8E 82F6        STX    PB          LICHT AN
4590 FB38 E8             INX
4600 FB39 E8             INX
4610 FB3A A0 96          LDY    #150
4620 FB3C 88       WARTEN DEY               WARTEZEIT FUER LICHT
4630 FB3D D0 FD          BNE    WARTEN
4640 FB3F A9 3F          LDA    ##3F
4650 FB41 8D 82F6        STA    PB          LICHT AUS
4660 FB44 A9 00          LDA    #00
4670 FB46 8D 81F6        STA    PADD        PORT A WIEDER EINGANG
4680 FB49 A5 F4          LDA    OBUF
4690 FB4B 60             RTS
4700           *
4710 FB4C A2 27   TASTE? LDX    ##27
4720 FB4E 8E 82F6 REIHE  STX    PB
4730 FB51 AD 80F6        LDA    PA          SPALTE ?
4740 FB54 49 FF          EOR    ##FF        INVERTIEREN
4750 FB56 D0 09          BNE    TAST0
4760 FB58 CA             DEX
4770 FB59 CA             DEX
4780 FB5A E0 1F          CPX    ##1F
4790 FB5C D0 F0          BNE    REIHE
4800 FB5E A9 19          LDA    ##19        KEINE TASTE
4810 FB60 60             RTS
4820 FB61 A0 00   TAST0  LDY    #00
4830 FB63 0A      TAST1  ASL    AC          Y WIRD SPALTEWERT
4840 FB64 B0 03          BCS    WERT
4850 FB66 C8             INY
4860 FB67 10 FA          BPL    TAST1       (IMMER)
4870 FB69 8A      WERT   TXA
4880 FB6A 29 0F          AND    ##0F        MASKIERUNG DES TAPE-BITS
4890 FB6C 4A             LSR    AC          ZEILENWERT/2
4900 FB6D AA             TAX
4910 FB6E 98             TYA
4920 FB6F CA      AD8    DEX
4930 FB70 30 05          BMI    R6
4940 FB72 18             CLC
4950 FB73 69 08          ADC    ##08        PLUS 8 FUER DIE NAECHSTE REI
4960 FB75 D0 F8          BNE    AD8
4970 FB77 A2 3F   R6     LDX    ##3F
4980 FB79 8E 82F6        STX    PB
4990 FB7C 60             RTS
5000           *
5010 FB7D A9 00   DUMP   LDA    #00
5020 FB7F 85 F1          STA    INL
5030 FB81 85 F2          STA    INH
5040 FB83 20 E3FB DUMP0  JSR    CHKCLR
5050 FB86 20 4FFA DUMP1  JSR    CARRET
5060 FB89 A9 3B          LDA    #';
5070 FB8B 20 86FA        JSR    TTYOUT
5080 FB8E A5 F7          LDA    ADRL
5090 FB90 CD F2F7        CMP    ENDEL
5100 FB93 A5 F8          LDA    ADRH
5110 FB95 ED F3F7        SBC    ENDEH
5120 FB98 90 13          BCC    DUMP3
```

```
5130 FB9A A5 F1          LDA   INL       ZAHL DER BLOECKE
5140 FB9C 85 F7          STA   ADRL
5150 FB9E A5 F2          LDA   INH
5160 FBA0 85 F8          STA   ADRH
5170 FBA2 A9 00          LDA   #00       LETZTES RECORD
5180 FBA4 20 CDFB        JSR   PRTSEQ    MIT 00 IN STARTSEQUENZ GEHEN
5190 FBA7 20 D8FB        JSR   CHKPRT
5200 FBAA 4C A1F8        JMP   DTCYC     ZURUECK ZUM HAUPTPROG.
5210 FBAD A9 18   DUMP3  LDA   #$18      FUER 24 BYTES
5220 FBAF 20 CDFB        JSR   PRTSEQ
5230 FBB2 A2 18          LDX   #$18
5240 FBB4 A0 00   DUMP4  LDY   #00
5250 FBB6 B1 F7          LDA   (ADRL),Y
5260 FBB8 20 AFFA        JSR   OUTBYT
5270 FBBB 20 35FC        JSR   INCADR
5280 FBBE CA             DEX
5290 FBBF D0 F3          BNE   DUMP4     NOCH NICHT 18 BYTES
5300 FBC1 20 D8FB        JSR   CHKPRT
5310 FBC4 E6 F1          INC   INL
5320 FBC6 D0 02          BNE   DUMP6
5330 FBC8 E6 F2          INC   INH
5340 FBCA 4C 83FB DUMP6  JMP   DUMP0
5350           *
5360 FBCD A2 03   PRTSEQ LDX   #03       ACCU STANZEN (=18 DEK =00)
5370 FBCF 20 AFFA PRTSQ1 JSR   OUTBYT
5380 FBD2 B5 F5   PRTSQ2 LDA   ADRH-3,X
5390 FBD4 CA             DEX
5400 FBD5 D0 F8          BNE   PRTSQ1
5410 FBD7 60             RTS
5420             *
5430 FBD8 A5 F6   CHKPRT LDA   CHECKH
5440 FBDA 20 B2FA        JSR   OUTBT
5450 FBDD A5 F5          LDA   CHECKL
5460 FBDF 20 B2FA        JSR   OUTBT
5470 FBE2 60             RTS
5480             *
5490 FBE3 D8      CHKCLR CLD
5500 FBE4 A9 00          LDA   #00
5510 FBE6 85 F5          STA   CHECKL
5520 FBE8 85 F6          STA   CHECKH
5530 FBEA 60             RTS
5540             *
5550 FBEB 48      CHECK  PHA
5560 FBEC 18             CLC
5570 FBED 65 F5          ADC   CHECKL
5580 FBEF 85 F5          STA   CHECKL
5590 FBF1 A5 F6          LDA   CHECKH
5600 FBF3 69 00          ADC   #00       CARRY ?
5610 FBF5 85 F6          STA   CHECKH
5620 FBF7 68             PLA
5630 FBF8 60             RTS
5640             *
5650 FBF9 20 E3FB LADEN  JSR   CHKCLR
5660 FBFC 20 F1F9 LADEN1 JSR   ASCIIN
5670 FBFF C9 3B          CMP   #';       WARTEN AUF SEMIKOLON
5680 FC01 D0 F9          BNE   LADEN1
5690 FC03 20 48FC        JSR   HEXIN
5700 FC06 20 EBFB        JSR   CHECK
5710 FC09 AA             TAX             ZAHL DER BYTES
5720 FC0A 20 48FC        JSR   HEXIN
```

```
5730 FC0D 85 F8          STA    ADRH
5740 FC0F 20 EBFB        JSR    CHECK
5750 FC12 20 48FC        JSR    HEXIN
5760 FC15 85 F7          STA    ADRL
5770 FC17 20 EBFB        JSR    CHECK
5780 FC1A 8A             TXA
5790 FC1B F0 11          BEQ    LADEN3
5800 FC1D 20 48FC LADEN2 JSR    HEXIN
5810 FC20 D0 22          BNE    ERROR
5820 FC22 91 F7          STA    (ADRL),Y
5830 FC24 20 EBFB        JSR    CHECK
5840 FC27 20 55FC        JSR    INCADR
5850 FC2A CA             DEX
5860 FC2B D0 F0          BNE    LADEN2
5870 FC2D E8             INX
5880 FC2E 20 48FC LADEN3 JSR    HEXIN
5890 FC31 C5 F6          CMP    CHECKH
5900 FC33 D0 0F          BNE    ERROR
5910 FC35 20 48FC        JSR    HEXIN
5920 FC38 C5 F5          CMP    CHECKL
5930 FC3A D0 08          BNE    ERROR
5940 FC3C 8A             TXA
5950 FC3D D0 BA          BNE    LADEN
5955 FC3F A2 0B          LDX    #11
5956 FC41 4C 20F9 LADEN4 JMP    MONA1
5970 FC44 A2 0C   ERROR  LDX    #12
5990 FC46 D0 F9          BNE    LADEN4
6000          *
6010 FC48 20 F1F9 HEXIN  JSR    ASCIIN
6020 FC4B 20 2DFA        JSR    HEXA
6030 FC4E 20 F1F9        JSR    ASCIIN
6040 FC51 20 2DFA        JSR    HEXA
6050 FC54 60             RTS
6060          *
6070 FC55 E6 F7   INCADR INC    ADRL
6080 FC57 D0 02          BNE    R8
6090 FC59 E6 F8          INC    ADRH
6100 FC5B 38      R8     SEC
6110 FC5C 60             RTS
6120 FC5D 38      DECADR SEC
6130 FC5E A5 F7          LDA    ADRL
6140 FC60 E9 01          SBC    #01
6150 FC62 85 F7          STA    ADRL
6160 FC64 B0 02          BCS    R9
6170 FC66 C6 F8          DEC    ADRH
6180 FC68 38      R9     SEC
6190 FC69 60             RTS
9999 FC6A                END
```

```
 SEITE 1     MONA B 1.10.77   auf MICRONIC-650X-ASSEMBLER
Lauf 1 beendet                    0  Fehler
 0010            * TEIL 2 :  MNEMONIC + DATENBAND
 0020            *
 0030 FC6A               ORG    $FC6A
 0040 FC6A 69      CODE  FCB    $69,$6D,$65,$61,$71,$75,$7D,$79
      FC6B 6D
      FC6C 65
      FC6D 61
      FC6E 71
      FC6F 75
      FC70 7D
      FC71 79
 0050 FC72 1F4D    ADC   FDB    $1F4D
 0060 FC74 29              FCB    $29,$2D,$25,$21,$31,$35,$3D,$39
      FC75 2D
      FC76 25
      FC77 21
      FC78 31
      FC79 35
      FC7A 3D
      FC7B 39
 0070 FC7C 1F8F    AND   FDB    $1F8F
 0080 FC7E 0E              FCB    $0E,$06,$0A,$16,$1E
      FC7F 06
      FC80 0A
      FC81 16
      FC82 1E
 0090 FC83 1F09    ASL   FDB    $1F09
 0100 FC85 90              FCB    $90
 0110 FC86 FF4B    BCL   FDB    $FF4B
 0120 FC88 B0              FCB    $B0
 0130 FC89 FF41    BCS   FDB    $FF41
 0140 FC8B F0              FCB    $F0
 0150 FC8C FFF7    BEQ   FDB    $FFF7
 0160 FC8E 2C              FCB    $2C,$24
      FC8F 24
 0170 FC90 FF88    BIT   FDB    $FF88
 0180 FC92 30              FCB    $30
 0190 FC93 FFDE    BMI   FDB    $FFDE
 0200 FC95 D0              FCB    $D0
 0210 FC96 FFE0    BNE   FDB    $FFE0
 0220 FC98 10              FCB    $10
 0230 FC99 FF96    BPL   FDB    $FF96
 0240 FC9B 50              FCB    $50
 0250 FC9C FF21    BVC   FDB    $FF21
 0260 FC9E 70              FCB    $70
 0270 FC9F FF23    BVS   FDB    $FF23
 0280 FCA1 18              FCB    $18
 0290 FCA2 0FB7    CLC   FDB    $0FB7
 0300 FCA4 D8              FCB    $D8
 0310 FCA5 0FBD    CLD   FDB    $0FBD
 0320 FCA7 58              FCB    $58
 0330 FCA8 0FBB    CLI   FDB    $0FBB
 0340 FCAA B8              FCB    $B8
 0350 FCAB 0FB5    CLV   FDB    $0FB5
 0360 FCAD C9              FCB    $C9,$CD,$C5,$C1,$D1,$D5,$DD,$D9
      FCAE CD
      FCAF C5
      FCB0 C1
```

```
     FCB1 D1
     FCB2 D5
     FCB3 DD
     FCB4 D9
0370 FCB5 0F8F    CMP    FDB    $0F8F
0380 FCB7 E0             FCB    $E0,$EC,$E4
     FCB8 EC
     FCB9 E4
0390 FCBA 0F4B    CPX    FDB    $0F4B
0400 FCBC C0             FCB    $C0,$CC,$C4
     FCBD CC
     FCBE C4
0410 FCBF 0F48    CPY    FDB    $0F40
0420 FCC1 CE             FCB    $CE,$C6,$D6,$DE
     FCC2 C6
     FCC3 D6
     FCC4 DE
0430 FCC5 6F48    DEC    FDB    $6F48
0440 FCC7 CA             FCB    $CA
0450 FCC8 6F45    DEX    FDB    $6F45
0460 FCCA 88             FCB    $88
0470 FCCB 6F42    DEY    FDB    $6F42
0480 FCCD 49             FCB    $49,$4D,$45,$41,$51,$55,$5D,$59
     FCCE 4D
     FCCF 45
     FCD0 41
     FCD1 51
     FCD2 55
     FCD3 5D
     FCD4 59
0490 FCD5 BFB4    EOR    FDB    $BFB4
0500 FCD7 EE             FCB    $EE,$E6,$F6,$FE
     FCD8 E6
     FCD9 F6
     FCDA FE
0510 FCDB 4F59    INC    FDB    $4F59
0520 FCDD E8             FCB    $E8
0530 FCDE 4F56    INX    FDB    $4F56
0540 FCE0 C8             FCB    $C8
0550 FCE1 4F53    INY    FDB    $4F53
0560 FCE3 4C             FCB    $4C,$6C
     FCE4 6C
0570 FCE5 7F1F    JMP    FDB    $7F1F
0580 FCE7 20             FCB    $20
0590 FCE8 7FE5    JSR    FDB    $7FE5
0600 FCEA A9             FCB    $A9,$AD,$A5,$A1,$B1,$B5,$BD,$B9
     FCEB AD
     FCEC A5
     FCED A1
     FCEE B1
     FCEF B5
     FCF0 BD
     FCF1 B9
0610 FCF2 CFDE    LDA    FDB    $CFDE
0620 FCF4 A2             FCB    $A2,$AE,$A6,$BE
     FCF5 AE
     FCF6 A6
     FCF7 BE
0630 FCF8 CFDA    LDX    FDB    $CFDA
0640 FCFA A0             FCB    $A0,$AC,$A4,$B4,$BC
```

```
     FCFB AC
     FCFC A4
     FCFD B4
     FCFE BC
0650 FCFF CFD7    LDY    FDB    $CFD7
0660 FD01 4E             FCB    $4E,$46,$4A,$56,$5E
     FD02 46
     FD03 4A
     FD04 56
     FD05 5E
0670 FD06 CF95    LSR    FDB    $CF95
0680 FD08 09             FCB    $09,$0D,$05,$01,$11,$15,$1D,$19
     FD09 0D
     FD0A 05
     FD0B 01
     FD0C 11
     FD0D 15
     FD0E 1D
     FD0F 19
0690 FD10 3F4C    ORA    FDB    $3F4C
0700 FD12 48             FCB    $48
0710 FD13 5FA4    PHA    FDB    $5FA4
0720 FD15 08             FCB    $08
0730 FD16 5FA8    PHP    FDB    $5FA8
0740 FD18 68             FCB    $68
0750 FD19 5F68    PLA    FDB    $5F68
0760 FD1B 28             FCB    $28
0770 FD1C 5F6C    PLP    FDB    $5F6C
0780 FD1E 2E             FCB    $2E,$26,$2A,$36,$3E
     FD1F 26
     FD20 2A
     FD21 36
     FD22 3E
0790 FD23 8FE8    ROL    FDB    $8FE8
0800 FD25 6A             FCB    $6A,$66,$76,$6E,$7E
     FD26 66
     FD27 76
     FD28 6E
     FD29 7E
0810 FD2A 8FE4    ROR    FDB    $8FE4
0820 FD2C 40             FCB    $40
0830 FD2D 8F4A    RTI    FDB    $8F4A
0840 FD2F 60             FCB    $60
0850 FD30 8F48    RTS    FDB    $8F48
0860 FD32 E9             FCB    $E9,$ED,$E5,$E1,$F1,$F5,$FD,$F9
     FD33 ED
     FD34 E5
     FD35 E1
     FD36 F1
     FD37 F5
     FD38 FD
     FD39 F9
0870 FD3A 2FC4    SBC    FDB    $2FC4
0880 FD3C 38             FCB    $38
0890 FD3D 2F88    SETC   FDB    $2F88
0900 FD3F F8             FCB    $F8
0910 FD40 2F8E    SED    FDB    $2F8E
0920 FD42 78             FCB    $78
0930 FD43 2F8C    SEI    FDB    $2F8C
0940 FD45 8D             FCB    $8D,$85,$81,$91,$95,$9D,$99
```

```
      FD46 C5
      FD47 81
      FD48 91
      FD49 95
      FD4A 9D
      FD4B 93
0950  FD4C 2FA7    STA    FDB    $2FA7
0960  FD4E 8E              FCB    $8E,$86,$96
      FD4F 86
      FD50 96
0970  FD51 2FA3    STX    FDB    $2FA3
0980  FD53 8C              FCB    $8C,$84,$94
      FD54 84
      FD55 94
0990  FD56 2FA0    STY    FDB    $2FA0
1000  FD58 AA              FCB    $AA
1010  FD59 DF7F    TAX    FDB    $DF7F
1020  FD5B A8              FCB    $A8
1030  FD5C DF7C    TAY    FDB    $DF7C
1040  FD5E BA              FCB    $BA
1050  FD5F DF8E    TSX    FDB    $DF8E
1060  FD61 8A              FCB    $8A
1070  FD62 DF33    TXA    FDB    $DF33
1080  FD64 9A              FCB    $9A
1090  FD65 DF34    TXS    FDB    $DF34
1100  FD67 98              FCB    $98
1110  FD68 DF06    TYA    FDB    $DF06
1120  FD6A 43      ASCII  FCC    /CASO1FDJRMHELTYBHXVO/
      FD6B 41
      FD6C 53
      FD6D 4F
      FD6E 49
      FD6F 50
      FD70 44
      FD71 4A
      FD72 52
      FD73 4D
      FD74 4E
      FD75 45
      FD76 4C
      FD77 54
      FD78 59
      FD79 42
      FD7A 48
      FD7B 58
      FD7C 56
      FD7D 51
 1130 FD7E 39      SIEBSG FCB    $39,$77,$6D,$3F,$06,$73,$5E,$1E,$50,$37
      FD7F 77
      FD80 6D
      FD81 3F
      FD82 06
      FD83 73
      FD84 5E
      FD85 1E
      FD86 50
      FD87 37
 1140 FD88 54              FCB    $54,$79,$38,$78,$6E,$7C,$76,$14,$6A,$67
      FD89 79
      FD8A 38
```

```
     FD8B 78
     FD8C 6E
     FD8D 7C
     FD8E 76
     FD8F 14
     FD90 6A
     FD91 67
1150            *
1160 FD92 8A       NEMON  TXA
1170 FD93 48              PHA
1180 FD94 98              TYA
1190 FD95 48              PHA
1200 FD96 A2 00           LDX    #00
1210 FD98 BD 6AFC MNEMO1  LDA    CODE,X
1220 FD9B 09 F0           ORA    #$F0
1230 FD9D C9 FF           CMP    #$FF
1240 FD9F D0 04           BNE    NOMN
1250 FDA1 E8              INX
1260 FDA2 E8              INX
1270 FDA3 F0 6C           BEQ    DATEN
1280 FDA5 BD 6AFC NOMN    LDA    CODE,X
1290 FDA8 E8              INX
1300 FDA9 A0 00           LDY    #00
1310 FDAB D1 F7           CMP    (ADRL),Y
1320 FDAD D0 E9           BNE    MNEMO1
1330 FDAF BD 6AFC SMN     LDA    CODE,X
1340 FDB2 09 F0           ORA    #$F0
1350 FDB4 C9 FF           CMP    #$FF
1360 FDB6 F0 03           BEQ    MNF
1370 FDB8 E8              INX
1380 FDB9 D0 F4           BNE    SMN
1390 FDBB A9 14    MNF    LDA    #20
1400 FDBD 85 EC           STA    Z1
1410 FDBF BD 6AFC         LDA    CODE,X
1420 FDC2 18              CLC
1430 FDC3 4A              LSR    AC
1440 FDC4 4A              LSR    AC
1450 FDC5 4A              LSR    AC
1460 FDC6 4A              LSR    AC
1470 FDC7 2C 80F6         BIT    PA
1480 FDCA 50 05           BVC    SSEG
1490 FDCC 06 EC           ASL    Z1          *2
1500 FDCE 18              CLC
1510 FDCF 69 14           ADC    #20
1520 FDD1 A8       SSEG   TAY
1530 FDD2 85 FE           STA    COPO
1540 FDD4 B9 6AFD         LDA    ASCII,Y
1550 FDD7 85 F9           STA    MNEM1
1560 FDD9 E8              INX
1570 FDDA BD 6AFC         LDA    CODE,X
1580 FDDD 4A              LSR    AC
1590 FDDE 4A              LSR    AC
1600 FDDF 4A              LSR    AC
1610 FDE0 4A              LSR    AC
1620 FDE1 18              CLC
1630 FDE2 65 FE           ADC    COPO
1640 FDE4 69 01           ADC    #01
1650 FDE6 C5 EC           CMP    Z1
1660 FDE8 30 03           BMI    CO1
1670 FDEA 38              SEC
```

```
1680 FDEB E9 14          SBC   #20
1690 FDED A8      CO1    TAY
1700 FDEE 85 FE          STA   COPO
1710 FDF0 B9 6AFD        LDA   ASCII,Y
1720 FDF3 85 FB          STA   MNEM2
1730 FDF5 BD 6AFC        LDA   CODE,X
1740 FDF8 29 0F          AND   #$0F
1750 FDFA 18             CLC
1760 FDFB 65 FE          ADC   COPO
1770 FDFD 69 01          ADC   #01
1780 FDFF C5 EC          CMP   Z1
1790 FE01 30 03          BMI   CO2
1800 FE03 38             SEC
1810 FE04 E9 14          SBC   #20
1820 FE06 A8      CO2    TAY
1830 FE07 B9 6AFD        LDA   ASCII,Y
1840 FE0A 85 FD          STA   MNEM3
1850 FE0C 68      CO3    PLA
1860 FE0D A8             TAY
1870 FE0E 68             PLA
1880 FE0F AA             TAX
1890 FE10 60             RTS
1900 FE11 A9 40   DATEN   LDA   #$40
1910 FE13 85 F9          STA   MNEM1
1920 FE15 85 FB          STA   MNEM2
1930 FE17 85 FD          STA   MNEM3
1940 FE19 D0 F1          BNE   CO3
1950                     OPT   LIST
1960           * DATENBAND AUFZEICHNUNG
1970           * STARTADRESSE NACH STRTH,STRTL
1980           * STOPADRESSE NACH ENDEH,ENDEL
1990           * IDENTIFIKATIONS-NR. NACH ID
2000 FE1B 20 E3FB TAUFZ  JSR   CHKCLR
2020 FE1E A9 39          LDA   #$39      EINSCHALTEN DES BANDMOTORS 1
2030 FE20 8D 82F6        STA   PB
2040 FE23 85 F4          STA   OBUF
2050 FE25 A2 0F          LDX   #$0F      BANDVORLAUF
2060 FE27 A9 FF   VORLF  LDA   #$FF      16 MAL
2065 FE29 85 E3          STA   TPFLG
2070 FE2B 8D 97F6        STA   T:1024    1024*256 ZYCLEN
2080 FE2E 2C 87F6 VORLF1 BIT   VORBEI    =CA. 4 SEC
2090 FE31 10 FB          BPL   VORLF1
2100 FE33 CA             DEX
2110 FE34 D0 F1          BNE   VORLF
2120 FE36 A2 64          LDX   #100      100 SYNCH.-ZEICHEN
2130 FE38 A9 16          LDA   #$16
2140 FE3A 20 87FE        JSR   XOUT
2150 FE3D A9 2A          LDA   #'*
2160 FE3F 20 96FE        JSR   TPOUT
2170 FE42 AD F4F7        LDA   ID
2180 FE45 20 B2FA        JSR   OUTBT
2190 FE48 AD F0F7        LDA   STRTL
2200 FE4B 85 F7          STA   ADRL
2210 FE4D 20 AFFA        JSR   OUTBYT
2220 FE50 AD F1F7        LDA   STRTH
2230 FE53 85 F8          STA   ADRH
2240 FE55 20 AFFA        JSR   OUTBYT
2250 FE58 A0 00   INFO   LDY   #00
2260 FE5A B1 F7          LDA   (ADRL),Y
2270 FE5C 20 AFFA        JSR   OUTBYT
```

```
2280 FE5F 20 55FC        JSR   INCADR
2290 FE62 A5 F7          LDA   ADRL
2300 FE64 CD F2F7        CMP   ENDEL
2310 FE67 A5 F8          LDA   ADRH
2320 FE69 ED F3F7        SBC   ENDEH
2330 FE6C 90 EA          BCC   INFO
2340 FE6E A9 2F          LDA   #'/
2350 FE70 20 96FE        JSR   TPOUT
2360 FE73 A5 F5          LDA   CHECKL
2370 FE75 20 B2FA        JSR   OUTBT
2380 FE78 A5 F6          LDA   CHECKH
2390 FE7A 20 B2FA        JSR   OUTBT
2400 FE7D A9 04          LDA   #04        END-OF-TAPE
2410 FE7F A2 02          LDX   #02
2420 FE81 20 87FE        JSR   XOUT
2430 FE84 4C 38FF        JMP   R10
2440            *
2450 FE87 86 EC   XOUT   STX   Z1         HILFSREGISTER Z
2460 FE89 48      XOUT1  PHA
2470 FE8A 20 96FE        JSR   TPOUT
2480 FE8D 68             PLA
2490 FE8E C6 EC          DEC   Z1
2500 FE90 D0 F7          BNE   XOUT1
2510 FE92 60             RTS
2520            *
2530 FE93 C4       TAB4  FCB   $C4,$00,$7F 3600HZ/2400HZ
     FE94 00
     FE95 7F
2540            *
2550 FE96 A0 07   TPOUT  LDY   #07
2560 FE98 84 E5          STY   ZEIT
2570 FE9A A0 02   DREI   LDY   #02
2580 FE9C 84 ED          STY   Z2
2590 FE9E B6 FA   TON    LDX   FREQ1,Y
2600 FEA0 48             PHA
2610 FEA1 2C 87F6 TON1   BIT   VORBEI
2620 FEA4 10 FB          BPL   TON1
2630 FEA6 B9 93FE        LDA   TAB4,Y
2640 FEA9 8D 94F6        STA   T:1
2650 FEAC A5 F4          LDA   OBUF
2660 FEAE 49 20          EOR   #$20
2670 FEB0 8D 82F6        STA   PB
2680 FEB3 85 F4          STA   OBUF
2690 FEB5 CA             DEX
2700 FEB6 D0 E9          BNE   TON1
2710 FEB8 68             PLA
2720 FEB9 C6 ED          DEC   Z2
2730 FEBB F0 05          BEQ   TON2
2740 FEBD 30 07          BMI   TON3
2750 FEBF 4A             LSR   AC
2760 FEC0 90 DC          BCC   TON
2770 FEC2 A0 00   TON2   LDY   #00
2780 FEC4 F0 D8          BEQ   TON
2790 FEC6 C6 E5   TON3   DEC   ZEIT
2800 FEC8 10 D0          BPL   DREI
2810 FECA 60             RTS
2820            *
2830            * DATENBAND LESEN
2840 FECB A9 3B          LDA   #$3B
2850 FECD 8D 82F6        STA   PB
```

```
2860 FED0 20 E3FB        JSR   CHKCLR
2870 FED3 20 69FF SYN    JSR   RDBIT     SYNCHRONISIEREN
2880 FED6 46 F2          LSR   INH
2890 FED8 05 F2          ORA   INH
2900 FEDA 85 F2          STA   INH
2910 FEDC C9 16   TEST   CMP   #$16      SYNCH-ZEICHEN ?
2920 FEDE D0 F3          BNE   SYN
2930 FEE0 20 55FF        JSR   RDCHT
2940 FEE3 C6 F2          DEC   INH
2950 FEE5 10 F5          BPL   TEST
2960 FEE7 C9 2A          CMP   #'*
2970 FEE9 D0 F1          BNE   TEST
2980 FEEB 20 48FF        JSR   RDBYT     AUF GEHT'S
2990 FEEE CD F4F7        CMP   ID        DAS MUSS ID SEIN
2991 FEF1 F0 05          BEQ   IDOK
2992 FEF3 AD F4F7        LDA   ID
2993 FEF6 D0 DB          BNE   SYN
3000 FEF8 A2 FE   IDOK   LDX   #$FE      -2
3010 FEFA 20 48FF ADDR   JSR   RDBYT
3020 FEFD 95 F9          STA   ADRH+1,X
3030 FEFF 95 F1          STA   ADH+1,X
3040 FF01 20 EBFB        JSR   CHECK
3050 FF04 E8             INX
3060 FF05 30 F3          BMI   ADDR
3070 FF07 A2 02   BYTE   LDX   #02
3080 FF09 20 55FF DUBL   JSR   RDCHT
3090 FF0C C9 2F          CMP   #$2F      END ?
3100 FF0E F0 12          BEQ   WIND
3110 FF10 20 2DFA        JSR   HEXA
3120 FF13 D0 2B          BNE   ER
3130 FF15 CA             DEX
3140 FF16 D0 F1          BNE   DUBL
3150 FF18 91 F7          STA   (ADRL),Y
3160 FF1A 20 EBFB        JSR   CHECK
3170 FF1D 20 55FC        JSR   INCADR
3180 FF20 B0 E5          BCS   BYTE
3190 FF22 20 48FF WIND   JSR   RDBYT
3200 FF25 C5 F5          CMP   CHECKL
3210 FF27 D0 17          BNE   ER
3220 FF29 20 48FF        JSR   RDBYT
3230 FF2C C5 F6          CMP   CHECKH
3240 FF2E D0 10          BNE   ER
3250 FF30 A5 EF          LDA   ADL
3260 FF32 85 F7          STA   ADRL
3270 FF34 A5 F0          LDA   ADH
3280 FF36 85 F8          STA   ADRH
3290 FF38 A9 3F   R10    LDA   #$3F
3300 FF3A 8D 82F6        STA   PB
3310 FF3D 4C A1F8        JMP   DTCYC
3320 FF40 A9 FF   ER     LDA   #$FF
3330 FF42 85 F8          STA   ADRH
3340 FF44 85 F7          STA   ADRL
3350 FF46 D0 F0          BNE   R10
3360            *
3370 FF48 20 55FF RDBYT  JSR   RDCHT
3380 FF4B 20 2DFA        JSR   HEXA
3390 FF4E 20 55FF        JSR   RDCHT
3400 FF51 20 2DFA        JSR   HEXA
3410 FF54 60             RTS
3420            *
```

```
3430 FF55 86 EC    RDCHT  STX    Z1
3440 FF57 A2 08           LDX    ##08
3450 FF59 20 69FF RDCHT1 JSR    RDBIT
3460 FF5C 2A              ROL    AC
3470 FF5D 66 F3           ROR    IBUF
3480 FF5F CA              DEX
3490 FF60 D0 F7           BNE    RDCHT1
3500 FF62 A5 F3           LDA    IBUF
3510 FF64 2A              ROL    AC
3520 FF65 4A              LSR    AC          PARITAET GELOESCHT
3530 FF66 A6 EC           LDX    Z1
3540 FF68 60              RTS
3550            *
3560 FF69 2C 82F6 RDBIT  BIT    PB
3570 FF6C 30 FB           BMI    RDBIT
3580 FF6E AD 86F6         LDA    TREAD
3590 FF71 A0 FF           LDY    ##FF
3600 FF73 8C 96F6         STY    T:64
3610 FF76 A0 14           LDY    ##14
3620 FF78 88       P1     DEY
3630 FF79 D0 FD           BNE    P1
3640 FF7B 2C 82F6 P2     BIT    PB
3650 FF7E 10 FB           BPL    P2
3660 FF80 38              SEC
3670 FF81 ED 86F6         SBC    TREAD
3680 FF84 A0 FF           LDY    ##FF
3690 FF86 8C 96F6         STY    T:64
3700 FF89 A0 07           LDY    ##07
3710 FF8B 88       P4     DEY
3720 FF8C D0 FD           BNE    P4
3730 FF8E 49 FF           EOR    ##FF
3740 FF90 29 80           AND    ##80
3750 FF92 60              RTS
3760            *
3770 FF93 A9 3F   PLLTST LDA    ##3F
3780 FF95 8D 82F6 PLL1   STA    PB
3790 FF98 85 F4           STA    OBUF
3800 FF9A A2 8D           LDX    #141
3810 FF9C 8E 94F6         STX    T:1
3820 FF9F 2C 87F6 PLL2   BIT    VORBEI
3830 FFA2 10 FB           BPL    PLL2
3840 FFA4 A5 F4           LDA    OBUF
3850 FFA6 49 20           EOR    ##20
3860 FFA8 D0 EB           BNE    PLL1
3861            *
3865 FFAA 00       TAB5   FCB    $00,$06,$0C,$0A
     FFAB 06
     FFAC 0C
     FFAD 0A
3866 FFAE 24       TAB6   FCB    $24,$60,$60,$04
     FFAF 60
     FFB0 60
     FFB1 04
3870            *
3875 FFB2 D8       UHR    CLD
3880 FFB3 2C 87F6 UHR0   BIT    VORBEI
3890 FFB6 10 FB           BPL    UHR0
3900 FFB8 A9 78           LDA    #120
3910 FFBA 8D 94F6         STA    T:1
3920 FFBD 2C 87F6 UHR1   BIT    VORBEI
```

```
 3930 FFC0 10 FB          BPL    UHR1
 3940 FFC2 A9 F4          LDA    #$44
 3950 FFC4 8D 47F6        STA    T+1024
 3960 FFC7 A0 32          LDY    #$32
 3970 FFC9 84 E5          STY    ZEIT
 3980 FFCB A2 29    MUX   LDX    #$29
 3990 FFCD B5 B1    ANZEIG LDA   $B1,X
 4000 FFCF 20 1BFB        JSR    LICHT
 4010 FFD2 20 21FB        JSR    KLICHT
 4020 FFD5 E8             INX
 4030 FFD6 E8             INX
 4040 FFD7 E0 3B          CPX    #$3B
 4050 FFD9 D0 F2          BNE    ANZEIG
 4060 FFDB C6 E5          DEC    ZEIT
 4070 FFDD D0 EC          BNE    MUX
 4080 FFDF A0 03          LDY    #03
 4090 FFE1 BE AAFF SET    LDX    TABS,Y
 4100 FFE4 B5 DA          LDA    STD,X
 4110 FFE6 18             CLC
 4120 FFE7 F8             SED
 4130 FFE8 69 01          ADC    #01
 4135 FFEA 95 DA          STA    STD,X
 4140 FFEC D9 AEFF        CMP    TAB6,Y
 4150 FFEF D0 C1          BNE    UHR
 4160 FFF1 A9 00          LDA    #00
 4170 FFF3 95 DA          STA    STD,X
 4180 FFF5 88             DEY
 4190 FFF6 10 E9          BPL    SET
 4200 FFF8 30 B8          BMI    UHR
 4310 FFFA 00F8           FDB    $00F8
 4320 FFFC 22F8           FDB    $22F8
 4330 FFFE 03F8           FDB    $03F8
FEHLER15 Zeile 9999
 9999 0000                END
```

MNEMON	FD92	TAPOUT	FE96	PA	F680	PADD	F681
PB	F682	PBDD	F683	T:1	F694	T:64	F696
T:1024	F697	VORBEI	F687	TREAD	F686	IRQH	F7FF
IRQL	F7FE	NMIH	F7FB	NMIL	F7FA	TAHIGH	F7F9
TALOW	F7F8	TASTE	F7F7	BAUDH	F7F6	BAUDL	F7F5
ID	F7F4	ENDEH	F7F3	ENDEL	F7F2	STRTH	F7F1
STRTL	F7F0	ECHO	F7EF	PUNKT	00FF	COPO	00FE
MNEM3	00FD	FREQ2	00FC	MNEM2	00FB	FREQ1	00FA
MNEM1	00F9	ADRH	00F8	ADRL	00F7	CHECKH	00F6
CHECKL	00F5	OBUF	00F4	IBUF	00F3	INH	00F2
INL	00F1	ADH	00F0	ADL	00EF	DSPART	00EE
Z2	00ED	Z1	00EC	YSAVE	00EB	XSAVE	00EA
STATUS	00E9	STPO1N	00E8	AKKU	00E7	SEC	00E6
ZEIT	00E5	SEC:4	00E4	TFFLG	00E3	MIN	00E0
STD	00DA	TAB0	F806	TAB1	F80B	TAB2	F817
RESTRT	F822	STOP	F852	STP	F868	SLOWST	F87F
SLOW1	F88B	SLOW2	F893	DISP	F899	TZ	F89F
DTCYC	F8A1	NOCH	F8B4	ART?	F8C9	DATA	F8CD
DATA1	F8D5	CYCEND	F8D8	ADRES	F8DB	FUNK	F8E3
GO	F8F4	SPEED	F907	TTY	F915	MONA	F91E
MONA1	F920	TTY0	F92B	MONA2	F938	TTY1	F93B
TTY2	F941	GLADEN	F973	GDUMP	F976	ADR	F979
NEUDAT	F984	TDOWN	F98A	TUP	F98F	R3	F992
TSATZ	F994	TMNEMO	F9A4	TPROGC	F9AA	TSTATU	F9B4
TAKKU	F9B3	STORE	F9DA	TDATA	F9C2	TADRES	F9C8
HPAUSE	F9D0	PAUSE	F9D8	PAUSE0	F9DC	PAUSE1	F9DF
PAUSE2	F9EA	ASCIIN	F9F1	ASC1	FA08	R1	FA22
HEXA	FA2D	ZAHL	FA3C	TRANS	FA42	R2	FA4E
CARRET	FA4F	CARET1	FA51	ZEILE	FA5B	ZEILE0	FA5E
ZEILE1	FA75	SPACE	FA84	TTYOUT	FA86	ASC01	FA96
OUTBYT	FAAF	OUTBT	FAB2	HEXOUT	FABF	NUMM	FAC8
TAPE	FAD2	DISPLY	FAD6	DSP1	FAE2	DSP2	FB00
MNDSP	FB0E	DSP3	FB11	R5	FB1A	LLICHT	FB18
RLICHT	FB21	MLICHT	FB27	LICHT1	FB2D	WARTEN	FB3C
TASTE?	FB4C	REIHE	FB4E	TAST0	FB61	TAST1	FB63
WERT	FB69	AD8	FB6F	R6	FB77	DUMP	FB7D
DUMP0	FB83	DUMP1	FB86	DUMP3	FBAD	DUMP4	FBB4
DUMP6	FBCA	PRTSE0	FBCD	PRTS01	FBCF	PRTS02	FBD2
CHKPRT	FBD8	CHKCLR	FBE3	CHECK	FBEB	LADEN	FBF9
LADEN1	FBFC	LADEN2	FC1D	LADEN3	FC2E	LADEN4	FC41
ERROR	FC44	HEXIN	FC48	INCADR	FC55	R8	FC5B
DECADR	FC5D	R9	FC68	CODE	FC6A	ADC	FC72
AND	FC7C	ASL	FC83	BCL	FC86	BCS	FC8
BEQ	FC8C	BIT	FC90	BMI	FC93	BNE	FC9e
BPL	FC99	BVC	FC9C	BVS	FC9F	CLC	FCA2
CLD	FCA5	CLI	FCA8	CLV	FCAB	CMP	FCB5
CPX	FCBA	CPY	FCBF	DEC	FCC5	DEX	FCC8
DEY	FCCB	EOR	FCD5	INC	FCDB	INX	FCDE
INY	FCE1	JMP	FCE5	JSR	FCE8	LDA	FCF2
LDX	FCF8	LDY	FCFF	LSR	FD06	ORA	FD10
PHA	FD13	PHP	FD16	PLA	FD19	PLP	FD1C
ROL	FD23	ROR	FD2A	RTI	FD2D	RTS	FD30
SBC	FD3A	SETC	FD3D	SED	FD40	SEI	FD43
STA	FD4C	STX	FD51	STY	FD56	TAX	FD59
TAY	FD5C	TSX	FD5F	TXA	FD62	TXS	FD65
TYA	FD68	ASCII	FD6A	SIEBSG	FD7E	NEMON	FD92
MNEMO1	FD98	NOMN	FDA5	SMN	FDAF	MNF	FDBB
SSEG	FDD1	C01	FDED	C02	FE06	C03	FE0C
DATEN	FE11	TAUFZ	FE1B	VORLF	FE27	VORLF1	FE2E
INFO	FE58	XOUT	FE87	XOUT1	FE89	TAB4	FE93

TPOUT	FE96	DREI	FE9A	TON	FE9E	TON1	[illegible]
TON2	FEC2	TON3	FEC6	SYN	FED3	TES1	[illegible]
IOOK	FEF8	ADDR	FEFA	BYTE	FF07	DUBI	FF09
WIND	FF22	R10	FF38	ER	FF40	RDBIT	FF48
RDCNT	FF55	RDCNT1	FF59	RDBIT	FF69	PJ	FF78
P2	FF7B	P4	FF8B	PLLTST	FF93	PLL1	FF95
PLL2	FF9F	TAB5	FFAA	TAB6	FFAE	UHR	FFC2
UHR0	FFB3	UHR1	FFBD	MUX	FFCB	ANZEIG	FFCD
SET	FFE1						

Lauf 2 beendet

11.2 Kurzanleitung "MONA"-Betriebssystem auf ALPHA 1

[RS] Nach Netzeinschalten oder Drücken der RESTART-Taste wird das System neu initialisiert, d. h.

Kommunikations-Baudrate	= 110 Baud
Systemmeldung	= ein (Display leuchtet bzw. Terminal schreibt MONA)
NMI	= Stopvektor zeigt auf $ F852
IRQ	= Breakvektor zeigt auf $ F852
Status	= 00
Bandaufzeichnungsrate	= normal
Tastenvektor zeigt auf	"MONA"-Tastenprogramm
PORTS	= "MONA"-Bedingungen

der RESTART-Vektor selbst zeigt direkt auf $ F822

[ST] Die STOP-Taste läßt einen indirekten Sprung nach $ F852 erfolgen:

Accumulator Status Stackpointer laufende Adresse X- u. Y-Indexregister	werden entsprechend ihrem Wert im Moment des STOPs in die entsprechend benannten RAM-Adressen kopiert.

Dann wird der echte Status auf 00 gesetzt, die Systemmeldung wird eingeschaltet, und die Ports werden entsprechend den "MONA"-Bedingungen initialisiert.
Danach wird im Betriebssystem weitergearbeitet.

[GO] Die GO-Taste
Im "Normal"-Betrieb startet ein Programm an der in diesem Moment eingestellten Adresse, es werden die beim letzten STOP kopierten (bzw. durch RESTART gesetzten) Werte für: Accu, X- u. Y-Indexregister, Statusregister und Stackpointer aus den Kopien im RAM zurück in die echten Register kopiert, das Betriebssystem wird verlassen und das Benutzerprogramm läuft selbständig.

[↑] [↓] Funktionstasten: -↑- Die Adresse wird um eins erhöht.
-↓- Die Adresse wird um eins erniedrigt.

[AC] Stellt die Adresse der Accu-Kopie ein.

[SR] Stellt die Adresse der Status-Kopie ein.

[PC] Stellt den Inhalt von ADL, ADH ein.
Das ist die beim letzten Stop kopierte Adresse.

[DA] Wählt den Datenmodus an (Daten können verändert werden). Die Anzeige erfolgt hexadezimal.

[MN] Wählt den Datenmodus an (Daten können verändert werden). Die Anzeige erfolgt in Mnemonics.

[AD] Wählt den Adreßmodus an (Die Adresse kann geändert werden). Die Wahl des Adreßmodus beeinflußt die Darstellungsart der Daten nicht.

Single-Step — Im Single-Step-Betrieb arbeitet die GO-Taste zunächst genau wie im Normal-Betrieb, nach Bearbeitung eines Befehls wird aber automatisch ein "STOP" durchgeführt, d. h., es laufen alle unter STOP genannten Operationen ab. Es wird danach wieder im Betriebssystem fortgefahren. Nach nochmaligem Drücken der GO-Taste wird jeweils der nächste Schritt im Programm abgearbeitet.

Slow-Step — Im Slow-Step geschieht alles wie unter Single-Step, es wird aber nach der Beendigung eines GO-STOP-Zyklus (1 Befehlszyklus) nur für eine kurze Wartezeit im Betriebssystem verweilt (ca. 1 Sek.) und dann wieder ein automatischer GO-Befehl gegeben usw. ...

Dienstprogramm

a. Magnetband

Aufzeichnung

Um z. B. ein Programm mit der ersten Adresse 0210 und der letzten Adresse 0275 mit der I. D. 04 zu speichern, setzen Sie

F7F0	auf	10	Startadresse
F7F1	"	02	
F7F2	"	76	Endadresse + 1
F7F3	"	02	
F7F4	"	04	I. D.

Stellen Sie die Adresse FE1B ein und starten Sie das Band in Stellung Aufnahme. Drücken Sie die GO-Taste. Bei automatischer Start-Stop-Steuerung über das Motor-Kontroll-Relais von BU 1 Pin 4-5 beträgt die Bandvorlaufzeit ca. 4 Sek., bevor die Datenübertragung beginnt. Diese Pause wird in jedem Falle nach dem Drücken der GO-Taste eingehalten.

Sie können mit einem guten Aufzeichnungsgerät auch eine höhere Aufzeichnungsrate benutzen, als "MONA" beim RESTART vorgibt. Dazu werden vor der Aufzeichnung die Register FREQ1 und FREQ2 verändert:

RATE	FREQ1 (00FA)	FREQ2 (00FC)
Normal	6	9
Schnell	4	6
Hyper	2	3

Wiedergabe

Geben Sie die I. D. des zu ladenden Programms nach F7F4.
Starten Sie das Leseprogramm bei FECB.
Starten Sie das Magnetband.

Programme, die eine andere I. D. haben, werden ignoriert. Wenn Sie aber I. D. OO wählen, wird das nächste vollständige Programm gelesen, unabhängig von seiner I. D. Beim Erleuchten der Anzeige ist das Lesen beendet. Es erscheint die 1. Adresse des gelesenen Programms. FFFF bedeutet: Fehler beim Lesen. Das Leseprogramm kann alle Aufzeichnungen, d. h. "Normal", "Schnell" und "Hyper" ohne Änderung lesen. An Buchse 2 wird während des Lesens ein Kontakt über Pin 4-5 geschlossen.

b. Lochstreifen

Stanzen

Im Betrieb mit einem Terminal können Sie auch Lochstreifen stanzen. Geben Sie die letzte zu stanzende Adresse nach F7F2/F7F3, wählen Sie die erste zu stanzende Adresse an und drücken Sie Q auf der Tastatur.

Laden

Drücken Sie "L" auf der Tastatur und starten Sie den Lochstreifen. Wenn ein Fehler auftritt, meldet sich "MONA" mit einem Fragezeichen.

c. Uhr

Setzen der Register:

OODA = Stunden
OOEO = Minuten
OOE6 = Sekunden
OOE4 = OO

Programmstart bei FFB2.

d. Datenterminal

Schalter auf COM: es kommt sofort die Systemmeldung mit Baudrate 110.

SP = Abschluß der Adreßeingabe. Die letzten vier Zeichen werden gelesen. Führende Nullen können weggelassen werden.

. = Abschluß der Dateneingabe. Die Daten unter der gerade angezeigten Adresse können geändert werden. Führende Nullen können weggelassen werden.

CR = Adresse wird um eins erhöht.

Lf = Adresse wird um eins erniedrigt.

G = GO.

Q = DUMP Lochstr. MOS-Code.

L = Lesen Lochstr. MOS-Code.

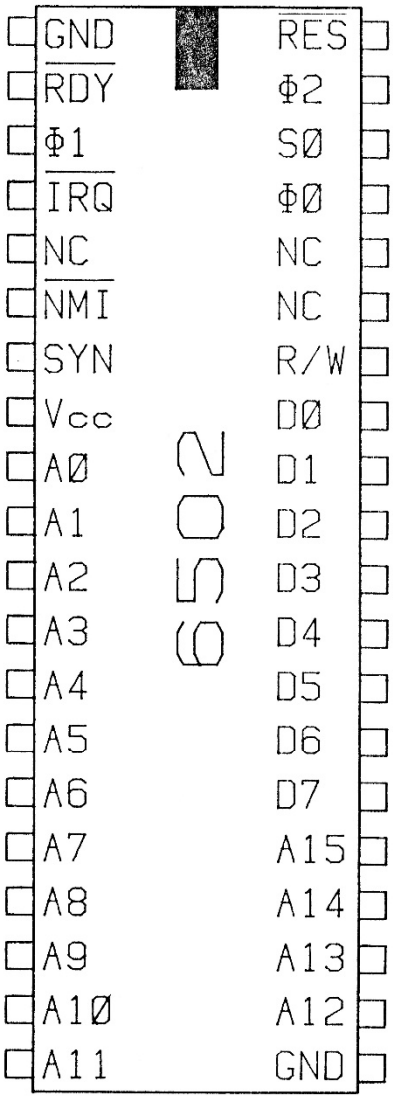
S = Anfang der Geschwindigkeitseingabe nach Baudratentabelle (z. B. S040C = 1200 Baud).

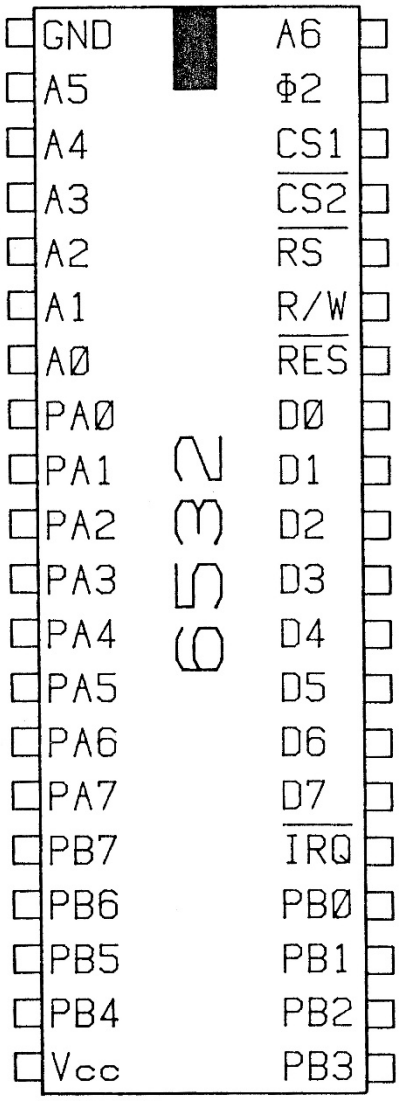
RUBOUT oder DELETE = Systemmeldung, Daten ignorieren.

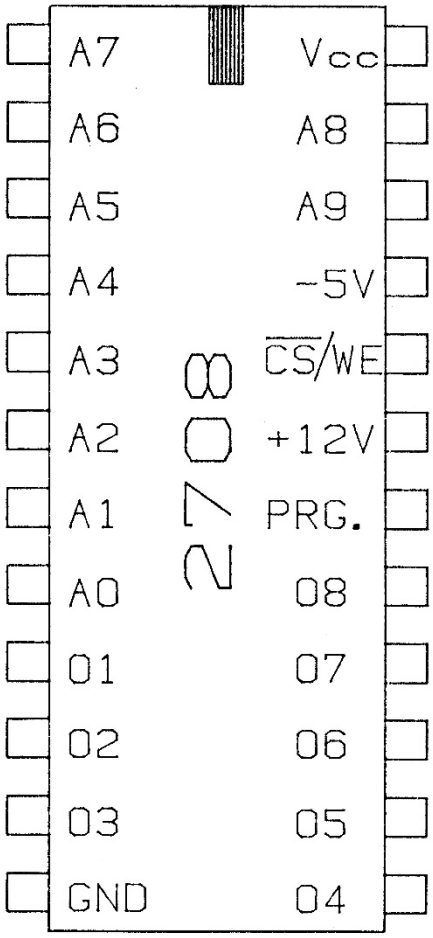
Baudraten-Tabelle	BAUDL (F7F5)	BAUDH (F7F6)
75	60	6E
110	4B	8C
150	53	66
300	3D	32
600	3B	18
1.200	04	0C
2.400	60	04
4.800	4C	01

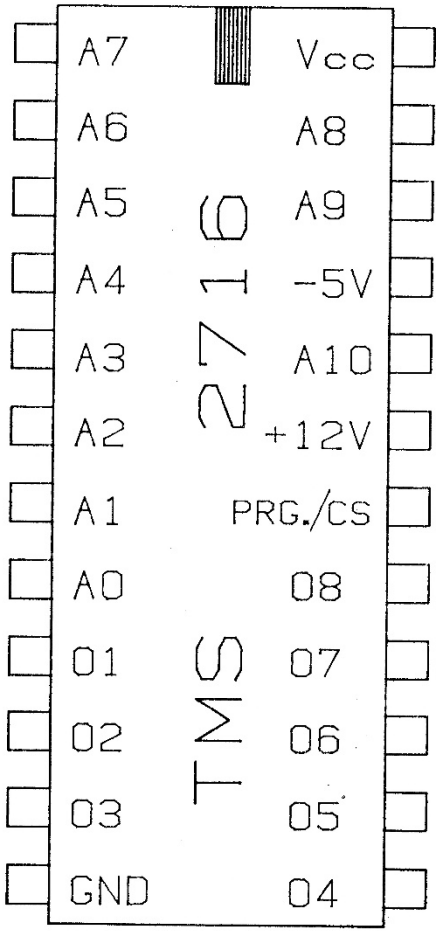
Für "Schnellesen" von Lochstreifen oder vom Terminal kann die Systemantwort abgeschaltet werden durch Setzen des "Echo"-Registers (ØF7EF) auf Ø10.
Durch Setzen des "Echo"-Registers auf Ø00 oder durch einen Stop wird die Systemantwort wieder eingeschaltet.

11.3 Anschlußzuordnung

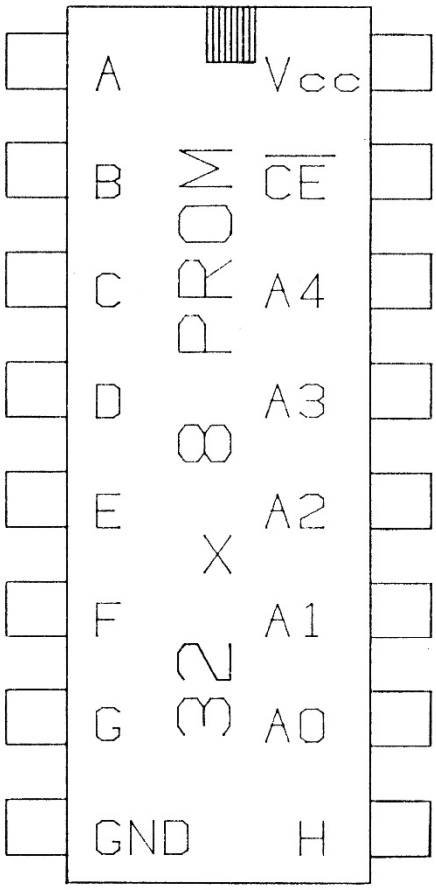



GND
A5
A4
A3
A2
A1
A0
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB7
PB6
PB5
PB4
Vcc
A6
Φ2
CS1
CS2
RS
R/W
RES
D0
D1
D2
D3
D4
D5
D6
D7
IRQ
PB0
PB1
PB2
PB3


A7
A6
A5
A4
A3
A2
A1
A0
O1
O2
O3
GND
Vcc
A8
A9
-5V
CS/WE
+12V
PRG.
O8
O7
O6
O5
O4
2716


A7
A6
A5
A4
A3
A2
A1
A0
O1
O2
O3
GND
Vcc
A8
A9
-5V
A10
+12V
PRG./CS
O8
O7
O6
O5
O4
TMS 2716

E-PROM



PROM

MCS-BUS

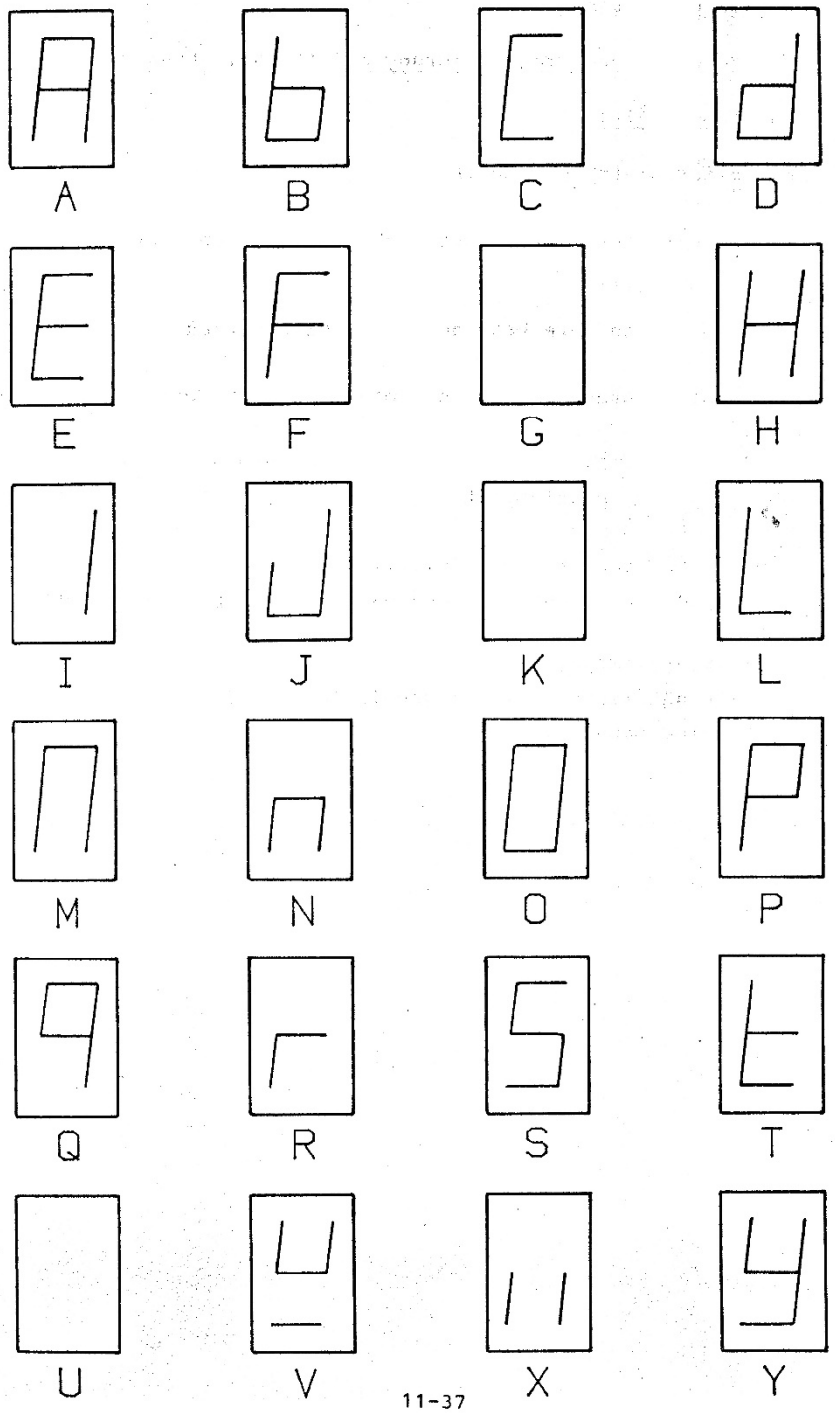
PIN	a	c	
1	~ Phase A ca. 20V	~ Phase A	PWR
2	~ Phase B ca. 20V	~ Phase B	
3	$\overline{\text{PWRFAIL}}$	+5V BAT	
4	+12V	+12V	
5	-12V	-12V	
6	+5V [Vcc]	+5V [Vcc]	
7	GND	GND	
8	GND	GND	
9	PRØ	PR1	INTERR.
10	PR2	PR3	
11	PR4 (RØ)	PR5 (R1)	
12	PR6 (R2)	PR7 (R3)	
13	RESERVE	SYN	CONTROL
14	$\overline{\text{I/O-SEL}}$	$\overline{\text{NOTREADY}}$	
15	R/$\overline{\text{W}}$	$\overline{\text{RDY}}$	
16	Φ2	Φ1	
17	$\overline{\text{IRQ-REQ}}$	$\overline{\text{IRQ-ACK}}$	
18	$\overline{\text{NMI}}$	$\overline{\text{RESET}}$	
19	$\overline{\text{HOLDREQ}}$	$\overline{\text{HOLDACK}}$	
20	$\overline{\text{DB7}}$	$\overline{\text{DB6}}$	DATA
21	$\overline{\text{DB5}}$	$\overline{\text{DB4}}$	
22	$\overline{\text{DB3}}$	$\overline{\text{DB2}}$	
23	$\overline{\text{DB1}}$	$\overline{\text{DBØ}}$	
24	HOLD IN [PRIOR]	HOLD OUT [PRIOR]	CT
25	AB15	AB14	ADDRESS
26	AB13	AB12	
27	AB11	AB1Ø	
28	AB 9	AB 8	
29	AB 7	AB 6	
30	AB 5	AB 4	
31	AB 3	AB 2	
32	AB 1	AB Ø	

INTERR. [RESERVE]

VG 64pol [96pol. Geh.]
VG 95324
DIN 41612

MCS-FB 019

11.5 Zuordnung 7-Segment-Alphabet



11.6 Hinweise auf Band 2

Fortgeschrittene Programmierung auf ALPHA 1/ALPHA 2

Aus dem Inhalt:

Ein Blick hinter die Kulissen:

- Detaillierte Erklärung wichtiger MONA-Unterprogramme

Do it yourself:

So können Ihre Programme MONA mitbenutzen

Hey-Music: Lernen Sie gutes Timing und Alpha wird zur Musik-Box.

Mond-Landung: Ein komplettes Spielprogramm - das viel Treibstoff spart.

Ein kleiner Schritt für Sie.....:
Alphas Verbindung zur Außenwelt - der Application-Anschluß

Super-Alarm-Anlage:
Ein nützliches Anwendungsbeispiel das den Application-Anschluß benutzt.

DIE UNTERPROGRAMME DES BETRIEBSSYSTEMS MONA

Aufbau

Das Betriebssystem MONA ist so aufgebaut, daß Programmteile, die auch für Anwenderprogramme von Interesse sind, leicht zugänglich sind und eine Rückkehr in das Anwenderprogramm gestatten.
Im Folgenden werden diese Programmteile beschrieben und Beispiele für häufig wiederkehrende Anwendungen gegeben.

Tastenfunktionen

Ein häufig auftretendes Problem besteht darin, in ein laufendes Programm durch Tastendruck Informationen einzugeben. Hierfür bietet MONA das Programm "TASTE". Dieses Programm beginnt auf Adresse FB4C.
Wenn ein Benutzerprogramm mit JSR Taste (2Ø 4CFB) in dieses Programm springt, so steht nach dem Rücksprung aus dem Unterprogramm (RTS auf Adresse FB7C bzw. FB6Ø) die Information über die gedrückte Taste im Akkumulator.
Wenn keine Taste gedrückt war, so ist der Inhalt des Akkumulators gleich $ 19 (siehe Zeile 48ØØ des Assembler-Listings).
Für die Tasten gelten folgende Akkumulatorwerte:

Taste	Im Dezimalbetrieb	Im Hexa-Betrieb
Ø-9	ØØ-Ø9	ØØ-Ø9
A-F	1Ø-15	ØA-ØF
↑	16	10
↓	17	11
MN	18	12
PC	19	13
AC	20	14
SR	21	15
AD	22	16
DA	23	17
GO	24	18
keine Taste gedrückt	25	19

Die Tasten Reset (RS) und Stop (ST) können nicht benutzt werden.

Wenn ein Programm eine Eingabe erwartet, so kann z. B. periodisch "Taste?" benutzt werden und durch Vergleich des Akkumulatorinhalts mit festen Werten jeder Taste eine Funktion zugeordnet werden. Zu beachten ist, daß für die Entprellung der Tasten gesorgt werden muß, d. h. es ist sicherzustellen, daß eine erkannte Taste z. B. in mehreren aufeinanderfolgenden Zyklen gedrückt ist oder, daß der Wert $19 nach dem Loslassen für einige Zyklen stabil bleibt, um Doppeleingaben zu vermeiden bzw. vom Prellen zu unterscheiden. Beispiele für solche Entprellprogramme zeigt MONA von Zeile 1290 bis Zeile 1360, sowie in einer vereinfachten Form "Bauer Brösel" (Zeile 820 bis 890), es sind auch andere Möglichkeiten denkbar.

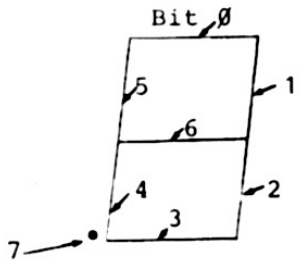
Displayfunktionen

Für die Anzeige von Daten im Display sind die Programme

LLICHT	SB1B
RLICHT	FB21
MLICHT	FB27

wesentlich.
Um zum Beispiel die beiden linken Stellen des Displays mit den Zeichen A und 7 leuchten zu lassen, wird $A7 in den Akkumulator und $29 in das X-Register geladen. $29 in X ist gleichbedeutend mit der am weitesten links stehenden Anzeige. Die nächste Stelle hat (in X) den Wert $2B, dann 2D, 2F, 31 usw. jeweils um 2 incrementiert.

Wird mit $A7 im Akkumulator und $29 in X mit JSR LLICHT gearbeitet, so leuchtet für einige Millisekunden das A im Display. Nach der Rückkehr aus dem Unterprogramm ist X bereits für die nächste Stelle incrementiert und JSR RLICHT läßt die 7 leuchten.
LLICHT läßt also das linke (bzw. höherwertige) Halb-Byte in der durch X vorgegebenen Stelle erscheinen, RLICHT das rechte (oder niederwertige) Halb-Byte. Das Unterprogramm MLICHT gestattet die Darstellung beliebiger Segmentkombinationen zur Erzeugung von Sonderzeichen, dabei wird der gesamte Akkumulatorinhalt (d. h. beide Halb-Bytes) zur Segmentbezeichnung nach folgendem Schema herangezogen:



Beispiel: $94 = % 1001 0100 = 

Wird kein Bit gesetzt (Akkumulator gleich $00), so bleibt die Anzeige dunkel.

MONA benutzt z. B. im Display/Tastenzyklus abwechselnd die Programme TASTE? und Anzeigeprogramme. Daraus wird ersichtlich, daß die Anzeigeprogramme nicht "dauernd" benutzt werden müssen, sondern nur "gelegentlich" angesprungen werden müssen. Entscheidend für die Häufigkeit ist nur ein ausreichend ruhig stehendes Gesamtbild. Steht beim Aufruf eines "LICHT"-Unterprogrammes im Register PUNKT ein negativer Wert (d. h. ist Bit 7 gleich 1), so wird in der Anzeige der PUNKT ergänzt (siehe Zeile 4520 und folgend).

Zeichen-Eingabe

Allgemeines

Die Einspeisung von ASCII-Zeichen kann mit Geschwindigkeiten zwischen 110 und 4800 Baud über den Kommunikationsstecker erfolgen. Es wird intern PA7 des Bausteins 6532/2 benutzt. Die Kontrolle über diese Eingabe hat das Programm ASCIIN (Adresse $F9F1). Es wartet auf ein Startbit (d. h. Null an PA7) oder Umschaltung auf Keyboard-Betrieb. Nach Erkennen eines Startbits wird mit der vorgegebenen Baudrate gelesen und die einlaufende ASCII-Information im Akkumulator festgehalten. Mit einer zeitlichen Versetzung von 0,5 Bit wird ein Echo an PORT BØ erzeugt, d. h. das Echo ist identisch mit der vom Processor erkannten Bitfolge. Um die Baudrate zu erhalten, wird zur Bestimmung der Bitlänge das Unterprogramm PAUSE benutzt.

Texteingabe, ASCII

Beispiel: Es sollen in einen Pufferbereich auf Page Ø zwischen ØØØØ und ØØ1Ø ASCII Zeichen eingegeben werden:

```
        LDX     #00
LOOP    JSR     ASCIIN
        STA     ØØØØ, X
        INX
        CPX     #1Ø
        BNE     LOOP
        BRK
```

Hexazeicheneingabe, ASCII

MONA kann mit dem Unterprogramm "Hexa" (FA2D) ASCII-Zeichen in Hexa-Zeichen umwandeln.
Die 4 zuletzt von Hexa verarbeiteten Zeichen sind unter INH (ØØF2) und INL (ØØF1) gespeichert.

Beispiel: Es soll vom Terminal aus dem IRQ-Vektor geändert werden:

```
START   JSR     ASCIIN      Hole ein Zeichen
        CMP     #$ØD        Ist es Return?
        BEQ     END
        JSR     HEXA        Umwandlung in Hexa
        JMP     START       Hole nächstes Zeichen bei
                            START
END     LDA     INH
        STA     IRQH        Übertragung der letzten
        LDA     INL         gültigen Zeichen
        STA     IRQL

        JMP     $FF         Sprung ins Betriebssystem
```

Sie können nun Zeichen eingeben, solange Sie wollen, die letzten von "Return" (=$ØD) geschriebenen gültigen Hexazeichen werden nach IRQL und IRQH übertragen.
Gültige Hexazeichen sind solche zwischen Ø - 9 und A F.

Wird die Hexaroutine mit einem Wert im Akkumulator angesprungen, der kein gültiges Hexazeichen darstellt, so ist nach der Rückkehr die Zero-flag = Ø, war das Zeichen gültig, so ist die Zero-flag = 1.

Zeichenausgabe

Allgemeines

Die Zeichenausgabe erfolgt unter den gleichen Bedingungen die unter 4.1. angegeben sind.

Textausgabe ASCII

Zur ASCII-Zeichenausgabe kann das Unterprogramm TTY OUT benutzt werden. Es überträgt das im Akkumulator stehende Zeichen zum Terminal.

Beispiel: Der unter 4.2. im Pufferbereich abgelegte Text soll wieder ausgegeben werden:

```
        LDX     #ØØ
LOOP2   LDA     ØØØØ,X
        JSR     TTYOUT
        INX
        CPX      1O
        BNE      LOOP2
        BRK
```

HEXIN (ØFC48) faßt beides zusammen

Hexazeichenausgabe, ASCII

Ein im Akkumulator stehendes Byte kann als zwei Halbbytes in Hexadarstellung zerlegt werden.
Diese Hexadarstellung kann durch OUTBYT (FAAF) als zwei ASCII-Zeichen ausgegeben werden.

Steht z. B. im Akkumulator die Bitfolge "0011 0001", so erscheint beim Benutzen der OUTBYT-Routine "31" auf dem Terminal.

Space, Return

Da die Leertaste sehr häufig benutzt wird, gibt es die Möglichkeit, besonders einfach das "Leer"-Zeichen zu erzeugen:

> Das Benutzen der Routine SPACE (FA84) erzeugt ein Leerfeld.

Ebenso wird der Wagenrücklauf (Carriage Return) und die Zeilenfortschaltung oft benötigt, dafür kann die Routine CARRET (FA4F) benutzt werden. Sie erzeugt ferner 5 ASCII "0"-Charakter, um einen einwandfreien Wagenrücklauf auch bei langsamen Geräten zu ermöglichen.

WEITERE UNTERPROGRAMME
======================

NAME	ADRESSE	AUFGABE
HPAUSE	F9D0	HALBE BITBREITE WARTEN
DISPLY	FAD6	INHALT UND ADRESSE IN ADRL/H IN DISPLAY SCHREIBEN
CHKCLR	FBE3	CHECKSUMME AUF NULL SETZEN
CHECK	FBE8	INHALT DES AKKUMULATORS ZUR CHECKSUMME ADDIEREN
INCADR	FC55	ADRESSPOINTER (ADRL/H) INCREMENTIEREN
DECADR	FC5D	ADRESSPOINTER DECRMENTIEREN
MEMON	FD92	INHALT DES AKKUMULATORS INTERPRETIEREN UND DREI OP-CODE-BUCHSTABEN IN F9/FB/FD GEBEN
XOUT	FE87	X-MAL AKKUMULATORINHALT AUF BAND SCHREIBEN
TPOUT	FE96	AKKUMULATORINHALT AUF BAND SCHREIBEN
RDBYT	FF48	BYTE VOM MAGNETBAND LESEN
RDCHT	FF55	ASCII-ZEICHEN VOM BAND LESEN
RDBIT	FF69	BIT VOM BAND LESEN
PLLTST	FF93	DAUERTON AN MAGNETBANDAUSGANG ERZEUGEN

TEIL 2

DIE PERIPHERIEBAUSTEINE

DIE TIMER

UEBUNG : TONGENERATOR

Der Peripheriebaustein 6532

1. Der Timer

Ihr ALPHA besitzt zwei spezielle Timer, die von Programmen angesprochen werden können. Ein Timer wird vom Betriebssystem MONA, z. B. bei der Ein- und Ausgabe von ASCII-Buchstaben oder bei den Magnetbandroutinen benutzt, steht also den Anwendungsprogrammen bedingt zur Verfügung, der zweite Timer ist völlig frei für Anwenderprogramme.

1.1. Allgemeine Eigenschaften der Timer

Ein Timer kann per Programm mit einer Zahl X zwischen 0 und 255 geladen werden. Von dieser Zahl aus zählt der Timer bei wahlweise jedem 1. 8. 64. oder 1024. Systemtakt um 1 abwärts. Der gegenwärtige Zählstand kann jeweils abgelesen werden. Wenn der Timer bis auf Null heruntergezählt ist, wird die Zählrate automatisch auf 1 gesetzt und so weiter abwärts gezählt (also ab FF). Beim Nulldurchgang wird ein Flag gesetzt und wahlweise ein IRQ (interrupt request) ausgelöst.

1.2. Teilerrate

Ob beim 1. 8. 64. oder 1024. Systemtakt um 1 abwärts gezählt wird, hängt davon ab, unter welcher Adresse die Zahl X abgelegt wurde.
Gleichzeitig bestimmt die Adresse, ob beim Nulldurchgang ein IRQ ausgelöst wird.

Für MONA´s Timer gilt:

Adresse	Teilerrate	Interrupt bei Nulldurchgang
F694	1	nein
F695	8	nein
F696	64	nein
F697	1024	nein
F69C	1	ja
F69D	8	ja
F69E	64	ja
F69F	1024	ja

Für den freien Timer gilt:

Adresse	Teilerrate	I R Q
F614	1	nein
F615	8	nein
F616	64	nein
F617	1024	nein
F61C	1	ja
F61D	8	ja
F61E	64	ja
F61F	1024	ja

Um im ALPHA einen Interrupt (IRQ) auszulösen, muß für den MONA-Timer die Brücke S1 und für den freien Timer die Brücke S2 gesteckt sein (siehe ALPHA Handbuch S. 4-3).

Jede Lese- und Schreiboperation zum Timer setzt die IRQ-Anforderung des Timers zurück.

1.3. Testen auf Nulldurchgang

Durch Lesen der Adresse F687 (F607 beim freien Timer) kann festgestellt werden, ob der Timer bereits bis auf Null gezählt hat. Solange dies noch nicht der Fall ist, ist Bit 7 der von F687 gelesenen Zahl = Ø. Nach dem Erreichen der Null ist dieses Bit = 1.

Durch die Befehlsfolge

```
BIT F687        (oder F6Ø7)
BPL FB
```

wird z. B. eine Schleife solange durchlaufen bis Bit 7 gesetzt wird, also der Timer "fertig" ist.

1.4. Lesen des Zählerstandes

Zu jeder beliebigen Zeit kann der Stand des Timers von der Adresse F686 (F6Ø6 beim freien Timer) gelesen werden, dabei wird die IRQ-Bedingung von Punkt 1.2. zurückgesetzt. Das Lesen der Zeit ist ebenso von F68E (F6ØE) möglich, dabei wird die IRQ-Bedingung für den Nulldurchgang eingeschaltet.

Das Lesen des Timers nach einem Nulldurchgang ermöglicht es festzustellen, wieviel Zeit seit dem Nulldurchgang vergangen ist.

1.5. Beispiele

1.5.1. ohne I R Q

Aufgabe: Ein Strich soll über das Display des ALPHA wandern. Der Sprung zur nächsten Stelle soll nach der Wartezeit erfolgen, die das einmalige Laden des Timers maximal erlaubt. Es ist der MONA-Timer zu benutzen.

Lösung:

```
0200 A2 29    LDX   #$29   ; linkes Digit
0202 A9 FF    LDA   #$FF   ; für Maximalzeit
0204 8D 97F6  STA   $F697  ; T: 1024, kein IRQ
```

```
0207 A9 FF    LDA #FF    alle Segmente an
                         (anderer Wert=anderes Bild)
0209 EO 39    CPX #39    rechtes Digit erreicht?
020B DO 02    BNE Ø2     nein
020D A2 29    LDX #29    ja, zurücksetzen
020F 20 27FB  JSR MLICHT
0212 2C 87F6  BIT VORBEI Bit 7?
0215 10 FB    BPL FB
0217 4C 0202  JMP 0202
```

Wenn Sie das Programm bei 0200 starten, springen zunächst "Achten" über das Display. Durch Änderung des Wertes bei 0208 können Sie alle anderen Zeichen erzeugen, also auch einen Strich- versuchen Sie es.

Übung: Verändern Sie die geladene Zeit zu anderen Werten. Schreiben Sie das Programm auf den freien Timer um.

1.5.2. mit IRQ

Aufgabe: Es sollen am Tonband (bzw. Verstärker-) Ausgang Töne im Hörbereich erzeugt werden, deren Frequenz durch den freien Timer über IRQ verändert werden soll, so daß eine aufsteigende Tonfolge entsteht.

Lösungsweg:

Der menschliche Hörbereich liegt etwa zwischen 20 Hz und 18 kHz, altersabhängig wird dieser geringer. Wir beschränken uns hier auf den Frequenzbereich zwischen 50 Hz und 12 kHz, da dieser Bereich von fast allen Menschen wahrgenommen werden kann. 50 Hz entsprechen einem Zeitintervall von 20 ms, 12 kHz entsprechen ~83 µs. Es muß zunächst eine Zeitschleife erstellt werden, die innerhalb dieser Zeitgrenzen die Spannung an Port B5 (Tonbandausgang am MONA Baustein 6532-2) umschaltet.
Bitte lesen Sie dazu das im Listing "Nfsweep" stehende Programmstück TONSCHLEIFE.
Hier wird in Zeile 240 der Input/Output Baustein des Betriebssystems mit einem Bitmuster versorgt, das am Anschluß PB5 (Tonbandausgang) ein "High" anlegt. Später, in Zeile 350 wird durch die EOR Verknüpfung mit $2Ø ein "Low" angelegt usw..
Die Dauer bis zur Umschaltung wird durch die da-

zwischenliegende Zeitschleife bestimmt. Die Grundschleife von "WEITER" bis Zeile 310 dauert ~83 µs. Diese Schleife wird so oft durchlaufen, wie durch X vorgegeben wird.
X wird in Zeile 270 vom Speicher "FREQ" übernommen. Bitte geben Sie zunächst dieses Programmstück von Ø2ØF bis Ø22A ein und schließen Sie einen Verstärker oder ein anderes Wiedergabegerät an einen der Tonbandausgänge an.

Achtung! Bitte stellen Sie die Lautstärke auf "leise"!
Nun starten Sie das Programm bei Ø2ØF und erhöhen die Lautstärke auf einen angenehmen Wert. Überzeugen Sie sich davon, daß eine Veränderung der Speicheradresse ØØFA (=FREQ) zu verschiedenen Tonhöhen führt.
Die Tonhöhe soll nun automatisch verändert werden:
Die IRQ Service-Routine ab Zeile 380 soll bei jedem IRQ angesprungen werden. Sie löscht zunächst die IRQ-Anforderung, incrementiert dann das FREQ-Register und startet den Timer neu. Danach wird an der Stelle weiter gearbeitet an der die Arbeit bei der IRQ-Anforderung unterbrochen wurde.
Es bleibt nun nur noch die Aufgabe, den IRQ-Vektor auf die Service-Routine einzustellen und den IRQ-Timer zu laden.
Dies geschieht mit der vorgesetzten INIT-Routine.

Anregungen zu Übungen:

1. Verändern Sie die Sweepgeschwindigkeit, indem Sie die Zeitspanne zwischen den Interrupt-Anforderungen verstellen.
2. Erzeugen Sie eine fallende Tonfolge.
3. Versetzen Sie den Frequenzbereich zu höheren und tieferen Werten.

2. Die Input/Output-Anschlüsse

Pro Peripheriebaustein 6532 stehen 16 frei programmierbare Anschlüsse zur Verfügung. Ein Baustein wird für die Tastatur, das Display, Schalter etc. des ALPHA verwandt, die Anschlüsse des zweiten Bausteins sind auf den Stecker mit der Bezeichnung Application geführt (Anschlußbelegung siehe Handbuch 1 S. 3-3) und stehen dem Anwender zur Verfügung.

2.1. Die Daten- Richtungsregister

Unter der Adresse F6Ø1 erreichen Sie das Datenrichtungsregister der -8-"A"-Ports, unter der Adresse F6Ø3 das der -8-"B"-Ports. Dabei entspricht ein dorthin geschriebenes Byte mit seinen 8 Bits direkt den 8 Anschlüssen. Ist ein Bit im Richtungsregister auf "1" gesetzt, so ist der entsprechende Anschluß ein Ausgang, ist das Bit="0", so ist der Anschluß

```
 SEITE 1      Ne sweep mit Timer 1+2    auf MICRONIC-650X-ASSEMBLER
Lauf 1 beendet                    0  Fehler
 0020            ********************* M C S ************************
 0030            *
 0040            *   MONA SYMBOLE:
 0050      00F4    OBUF    EQU     $F4
 0060      F694    T01M00  EQU     $F694       MONA TIMER OHNE IRQ
 0070      F687    VORBEI  EQU     $F687       NULLDURCHGANGSFLAG
 0080      00FA    FREQ    EQU     $FA         ZAHL DER SCHLEIFEN BIS UMSCHALT
 0090      F7FE    IRQL    EQU     $F7FE       INTERRUPT-VEKTOR
 0100      F7FF    IRQH    EQU     $F7FF
 0110      F61F    T12FRI  EQU     $F61F       FREIER TIMER 1024 MIT IRQ
 0120      F605    TIRQRS  EQU     $F605       TIMER 2 IRQ RESET
 0130      F682    PB      EQU     $F682
 0140            *
 0150 0200                ORG     $0200
 0160            *****INITIALISIERUNG DES IRQ-VEKTORS UND DES TIMERS***
 0170 0200 A9 2B  INIT    LDA     #$2B
 0180 0202 8D FEF7        STA     IRQL
 0190 0205 A9 02          LDA     #$02
 0200 0207 8D FFF7        STA     IRQH
 0210 020A A9 80          LDA     #$80
 0220 020C 8D 1FF6        STA     T12FRI
 0230            ********************TONSCHLEIFE************************
 0240 020F A9 3F  TON83T  LDA     #$3F        TONBANDAUSGANG INITIALISIEREN
 0250 0211 8D 82F6 TON83  STA     PB
 0260 0214 85 F4          STA     OBUF        UND STAND MERKEN
 0270 0216 A6 FA          LDX     FREQ        AKTUELLE ZAHL DER DURCHLAEUFE
 0280 0218 A9 53  WEITER  LDA     #83
 0290 021A 8D 94F6        STA     T01M00
 0300 021D 2C 87F6 WARTE  BIT     VORBEI
 0310 0220 10 FB          BPL     WARTE
 0320 0222 E8             INX
 0330 0223 D0 F3          BNE     WEITER
 0340 0225 A5 F4          LDA     OBUF
 0350 0227 49 20          EOR     #$20        UMSCHALTEN.
 0360 0229 D0 E6          BNE     TON83
 0370            *********IRQ SERVICE ROUTINE*****************************
 0380 022B AC 06F5 SERVI  LDY     TIRQRS      IRQ LOESCHEN
 0390 022E E6 FA          INC     FREQ        FREQUENZ ERHOEHEN
 0400 0230 A0 80          LDY     #$80        TIMER NEU LADEN
 0410 0232 8C 1FF6        STY     T12FRI
 0420 0235 40             RTI
 0430            ************************************************
 9999 0236                END
```

ein Eingang. Beim Einschalten der Betriebsspannung sind alle Anschlüsse automatisch als Eingänge geschaltet und bleiben so, bis sie vom Programm umgeschaltet werden.

Beispiel: Port A1, A3, A7 sollen als Ausgang geschaltet werden, die übrigen Anschlüsse A0, A2, A4, A5, A6 sollen Eingang bleiben.

Lösung: Es wird 86 (Hex) in das Register F6Ø1 geschrieben.

Die Datenregister

Unter der Adresse F6ØØ kann Port A gelesen und beschrieben werden, unter F6Ø2 Port B.

Beispiel: (mit I/O Tester M655 oder M647)

Vom Kontrollpult aus soll in das laufende Betriebssystem MONA eine binäre Zahl eingegeben werden und im Display des ALPHA hexadezimal angezeigt werden.
Schließen Sie dazu den Testadapter über das beigefügte Kabel an den ALPHA-Application-Anschluß an und bringen Sie alle Schalter in die Grundstellung nach oben.
Wählen Sie auf der ALPHA-Tastatur die Adresse F6ØØ an.
Sie werden dort den Datenwert "FF" finden. Schalten Sie nun am Testadapter einige Schalter des Ports A um, die Anzeige im ALPHA wird sich entsprechend ändern.

PA7 als Interrupteingang und/oder flankensensitiver Eingang:

Neben der normalen Funktion als E/A Port kann das Port PA7 auch als flankensensitiver Eingang und/oder Interrupteingang geschaltet werden. Die Kontrolle ob Interrupt enable, disable, pos. Flanke oder neg. Flanke geschieht durch einen einen Schreibbefehl in vier verschiedenen Adressen. Die Daten auf dem Datenbus zu dieser Zeit werden ignoriert.
Die Adressen der Applikations P/A sind:

F604	IRQ gesperrt
F605	IRQ durchgeschaltet
F606	positiv Flankengetriggert
F607	negativ Flankengetriggert

Wird die gewählte Flanke erkannt, so wird das Datenbit DB6 im Flagregister F607 H.
Das Lesen dieses Registers setzt das Bit zurück.

Ist der Interrupt enabled, so wird ein IRQ erzeugt und das Flag DB6 gesetzt.

Der Reset disabled den Interrupt und setzt die negative Flankensensivität.

Hinweis: Das Flag DB6 wird auch gesetzt, wenn PA7 als normaler Port arbeitet und auch wenn der Interrupt disabled ist. Weiterhin kann er gesetzt werden durch die Systeminitialisierung, z. B. ändern der Flankensensivität usw.
Es ist daher anzuraten, das Flag immer zu löschen, bevor der Interrupt freigegeben wird.

ALPHA - ALARM

EIN INTERESSANTES BEISPIEL FUER DIE ANWENDUNG
EINES MIKROKOMPUTERS IM HAUSHALTS-BEREICH IST
DIE REALISIERUNG EINER ALARMANLAGE.

DIESER ARTIKEL ZEIGT SCHALTUNGS - UND PRØGRAMMIER-
VORSCHLAEGE FUER DEN MIKROCOMPUTER ALPHA, DIE
DURCH DEN ANWENDER NACH SEINEN EIGENEN BEDUERF-
NISSEN ZUSAMMENGESTELLT WERDEN KOENNEN.

A UEBERWACHUNGSTEIL

DIE ZU UEBERWACHENDEN GEGENSTAENDE WIE Z.B. FENSTER
TUEREN, GEMAELDE ODER ANDERE GEGENSTAENDE MUESSEN
AN DAS KONTROLLGERAET ALPHA ANGESCHLOSSEN WERDEN.
DAZU SIND ZWEI VERSCHIEDENE WEGE MOEGLICH :
1. SERIENSCHALTUNG
DIESE KOMMT IN BETRACHT, WENN EINE GRUPPE VON OBJEKTEN
INSGESAMT GESICHERT WERDEN SOLL, OHNE DASS ALPHA DIE
EINZELOBJEKT UNTERSCHEIDEN MUSS. BEISPIELE DAFUER
SIND ALLE ZUGAENGE ZU EINEM RAUM, ALLE GEMAELDE
ODER ALLE FENSTER EINES HAUSES.
ES WIRD DAZU EINE RINGLEITUNG VERLEGT, DIE ALLE
OBJEKTE EINER GRUPPE VERBINDET UND ANFANGS - UND END-
PUNKT BEI ALPHA HAT. AN JEDEM OBJEKT BEFINDET SICH
EIN SENSOR DER IM RUHEZUSTAND DAFUER SORGT,
DAS DIE RINGLEITUNG GESCHLOSSEN IST. DAMIT SIND
ANFANGS- UND ENDPUNKT AN ALPHA LEITEND MITEINANDER
VERBUNDEN.
2. EINZELSCHALTUNG
SOLLEN EINZELNE GEGENSTAENDE GESONDERT UEBERWACHT WERDEN,
SO BENOETIGEN SIE EINE EIGENE ALARMLEITUNG. SIE KOENNEN
DAMIT VON ALPHA IDENTIFIZIERT WERDEN UND AUCH DURCH
BESONDERE PROGRAMMTEILE GESONDERT BEHANDELT WERDEN.

ALPHA KANN IN DER GRUNDAUSFUEHRUNG UEBER DEN APPLICATION-
ANSCHLUSS BIS ZU 16 SOLCHER RING- ODER EINZELANSCHLUESSE
(AUCH GEMISCHT) AUFNEHMEN.

SOFTWARE
ZUNAECHST SOLL ALPHA UEBERPRUEFEN, OB SICH ALLE SCHALTER
ORDNUNGSGEMAESS IM GESCHLOSSENEN ZUSTAND BEFINDEN.
EIN GEOEFFNETER KONTAKT SOLL ERKANNT WERDEN.
NEBENBEDINGUNGEN (ZEIT, WEITERE OFFENE KONTAKTE ETC.)
WERDEN ZUNAECHST NICHT BEACHTET.
DAS FOLGENDE PROGRAMMSTUECK IST SO AUFGEBAUT, DASS ES ALS
UNTERPROGRAMM BENUTZT WERDEN KANN. ES ORIENTIERT SICH
AN DER HARDWARE-KONFIGURATION IN BILD 2.
ES WIRD VORAUSGESETZT, DASS DAS RICHTUNGSREGISTER VON
PORT A AUF $X0 GESETZT WURDE (X STEHT FUER BELIEBIG,
0 FUER DIE BITFOLGE 0000), D.H. MINDESTENS DIE 4
NIEDERWERTIGEN BITS MUESSEN AUF "0" = EINGANG STEHEN,
DER REST WIRD IGNORIERT.
SOOFT DAS UNTERPROGRAMM DURCHLAUFEN WIRD, WERDEN ALLE
KONTAKTE UEBEPRUEFT. SIND ALLE GESCHLOSSEN, SO HOERT
DAS PROGRAMM OHNE UMWEGE BEI "TSTEND" AUF. WIRD EIN KONTAKT
GEOEFFNET, SO WIRD EIN WEITERES UNTERPROGRAMM "ALARM"
DURCHLAUFEN. IM X-REGISTER STEHT DABEI DIE OBJEKT-
(ODER SCHLEIFEN-) NUMMER, AN DER DER GEOEFFNETE KONTAKT
ERKANNT WURDE.

```
TESTE   LDA PORTA       ;BITMUSTER HOLEN
        ORA #$F0        ;REST-BITS MASKIEREN
        EOR #$FF        ;UND ALLES INVERTIEREN
        CLC             ;ALS FLAG FUER "ALPHA"
        BEQ TSTEND      ;WENN ALLES "0" IST KEIN ALARM
        LDX #1          ;EIN KONTAKT IST OFFEN
TST1    LSR A           ;WELCHER ?
        BCC TST2        ;DIESER NICHT
        JSR ALARM       ;ABER DIESER, X ZEIGT STELLE
TST2    INX
        CPX #8          ;SCHON ALLE GEPRUEFT ?
        BNE TST1
        SEC             ;ALS FLAG
TSTEND  LDX #0          ;ALLES IN ORDNUNG
TSTEN   RTS
```

B ALARMKONDITIONEN

EIN SCHWERWIEGENDER MANGEL KONVENTIONELLER ALARMANLAGEN LIEGT IN DEN VERSCHIEDENTLICH VORKOMMENDEN FEHLALARMEN. DURCH ERSCHUETTERUNGEN ODER LUFTZUG WERDEN ALARMSENSOREN KURZZEITIG AUSGELOEST UND FUEHEREN SOFORT ZUM ALARM, OHNE DASS EIN "ERNSTFALL" VORLIEGT.
AN DIESER STELLE KANN EINE "INTELLIGENTE" ANLAGENSTEUERUNG EINEN INTERESSANTEN VORTEIL BIETEN : SIE KANN DURCH UEBERPRUEFEN VON RANDBEDINGUNGEN FESTSTELLEN, OB EIN ALARM WIRKLICH AUSGELOEST WERDEN SOLL, ODER OB ES SICH WAHRSCHEINLICH UM EINE STOERUNG HANDELT.
FUER SOLCHE PLAUSIBILITAETS-PRUEFUNGEN KOENNEN WIR NATUERLICH KEIN PATENT-REZEPT GEBEN, DENN DIE BEDINGUNGEN SEHEN IN JEDEM ANWENDUNGSFALL ANDERS AUS, IM FOLGENDEN GEBEN WIR DAHER NUR EINIGE TIPS FUER MOEGLICHE KONTROLLEN:

A WENN BEI VOLLER HAUSBELEUCHTUNG, LAUFENDEM FERNSEH-GERAET UND OFFENEN ZIMMERTUEREN JEMAND EIN GEMAELDE BERUEHRT, SO WIRD EIN ALARM WOHL NICHT ANGEBRACHT SEIN. GESCHIEHT DIES JEDOCH IM DUNKELN, ODER BEI GESCHLOSSENEN ZIMMERTUEREN UND OFFENER VERANDATUER .....

B FENSTER FUEHREN OFT ZU RAEUMEN, IN DENEN SELBST KEINE "STEHLENSWERTEN" GEGENSTAENDE SIND. DIE ERSCHUETTERUNG EINES FENSTERKONTAKTES MUSS ALSO NICHT GLEICH ALARM AUSLOESEN, DENN AUCH DER WIND KANN EINMAL AN EINEM FENSTER RUETTELN. WENN JEDOCH AUCH DIE ZUGEHOERIGE ZIMMERTUER GEOEFFNET WIRD, DURCH DIE EIN EINBRECHER IN WICHTIGE RAEUME GELANGEN KOENNTE, SO WIRD EIN ALARM SINNVOLL.

C ALLE ALARMFUNKTIONEN KOENNEN AUCH MIT WARTEZEITEN VERKNUEPFT WERDEN. SO KANN DIE ANZEIGE IM DISPLAY SOFORT ERFOLGEN (STUFE 1), DAS EINSCHALTEN DES WARN-TONES ABER ERST BEI EINER LAENGER ALS Z.B. 10 SEK. ANHALTENDEN AUSLOESUNG EINSETZEN, EINE WARNLAMPE (Z.B. RUNDUMLEUCHTE VOR DEM HAUS) KANN WIEDERUM VERZOEGERT WERDEN. DENKBAR IST AUCH, DAS DIE WARTE-ZEITEN SICH IN ABHAENGIGKEIT VON DER TAGESZEIT AENDERN.

ZUR UEBERPRUEFUNG DERARTIGER NEBENBEDINGUNGEN WURDE DAS UNTERPROGRAMM "CONDIT" EINGEFUEHRT. ES UEBERPRUEFT DIE AN PORT B6 UND PORT B7 ANLIEGENDEN BEDINGUNGEN FUER DIE ENTSCHEIDUNG, OB ZUSAETZLICH ZUR ALARMSTUFE 1 AUCH DIE ALARMSTUFEN 2 UND DREI EINGESCHALTET WERDEN SOLLEN.
DENKBAR WAEHRE HIER EINE ANZAHL VON WEITEREN UNTERPRO-GRAMMEN, DIE JE NACH ALARMQUELLE (X-REGISTER) BESONDERE NEBENBEDINGUNGEN UEBERPRUEFEN. ES KOENNTE HIER AUCH EIN ZAEHLER TESTEN, OB DER ENTSPRECHENDE KONTAKT LANGE ODER NUR KURZZEITIG GEOEFFNET WURDE.

C AUSGABE

ALPHA SOLL DAUERND DEN ZUSTAND DER GESAMTANLAGE ANZEIGEN. DAZU SOLL ZUNAECHST DAS DISPLAY VON ALPHA BENUTZT WERDEN :
IM RUHEZUSTAND SOLL ALS CODEWORT "ALPHA" IM DISPLAY STEHEN, IM ALARMFALL SOLL EINE BEZEICHNUNG DER QUELLE ERSCHEINEN.
DIE SOFTWARE GREIFT DAZU AUF EINE TABELLE VON ANZEIGE-WORTEN ZURUECK, DIE FUER DAS BEISPIEL IN BILD 2 SO AUSSEHEN KANN :

```
TABEL .BYTE $40,$77,$38,$73,$76,$77,$40,$40 ;ALPHA
.BYTE $7D,$50,$3E,$73,$73,$79,0,6 ;GRUPPE 1
.BYTE $40,$7C,6,$38,$5E,$79,$50,$40 ;-BILDER-
.BYTE $40,$78,$50,$79,$6D,$3F,$50,$40;-TRESOR-
.BYTE $78,$79,$50,$50,$77,$6D,$6D,$79;TERRASSE
```

ALS ZWEITE AUSGABE-MOEGLICHKEIT STELLEN WIR HIER EINE WEITERE VARIANTE VON TONPROGRAMMEN VOR. DER ANWENDER MAG DIE ENTSPRECHENDE UNTERROUTINE (ALRM2) DURCH ABWANDLUNG AUCH ZU ANDEREN KLAENGEN BRINGEN.
IM RUHEZUSTAND WIRD NUR DIE DISPLAY-ROUTINE DURCHLAUFEN, IM ALARMFALL WIRD ZWISCHEN OPTISCHEM UND AKUSTISCHEM ALARM HIN- UND HERGESCHALTET.
DAS FOLGENDE LISTING ZEIGT EINE ZUSAMMENFASSUNG DER BEIDEN MOEGLICHKEITEN, SIE WERDEN ALS ALARMSTUFE 1 UND ALARM-STUFE 2 BEHANDELT. DER AKUSTISCHE ALARM IST DURCH ANLEGEN EINER LOGISCHEN "NULL" AN PORT B 7 ABZUSCHALTEN.
ALS DRITTE ALARMSTUFE, DIE UEBER PORT B 6 GESPERRT WERDEN KANN, WIRD AN PORT B 0 EINE LOGISCHE "EINS" ANGELEGT. DAMIT KOENNTE Z.B. EINE RUNDUM-WARNLEUCHTE EINGESCHALTET WERDEN.

D BEISPIEL FUER EIN VOLLSTAENDIGES PROGRAMM

DIE OBEN BESCHRIEBENEN BEISPIELE SIND IM FOLGENDEN ASSEMBLER-LISTING ZU EINEM VOLLSTAENDIGEN PROGRAMM ZUSAMMENGEFASST. ZUM TESTEN WURDE DER I/O- TESTER M647 BENUTZT. EBENSO EIGNET SICH DER I/O TESTER M655.

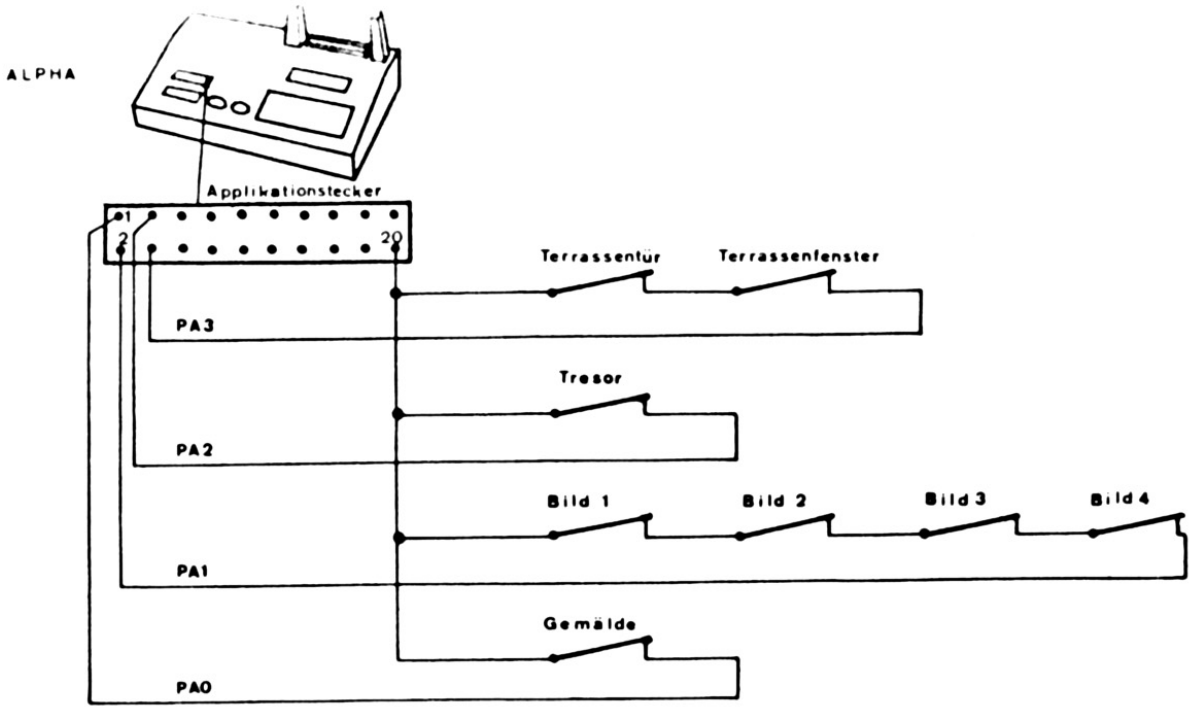


Bild1 : Beispiel für den Anschluss von Objekten in Serie und einzeln.

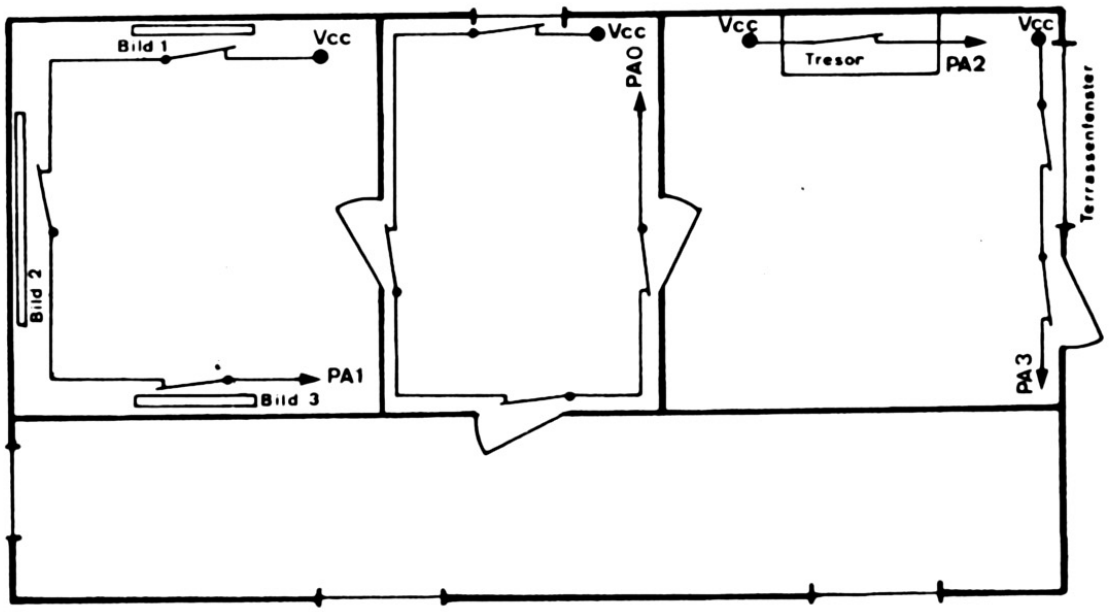


Bild2 : Im Programm benutztes Beispiel für die Verteilung der Sensoren.

```
LINE # LOC      CODE       LINE

 0010  0200              ;*******************************************************
 0020  0200              ;  A L A R M A N L A G E
 0030  0200              ;  EIN 6502 - PROGRAMM
 0040  0200              ;  FUER ALPHA/BETA
 0050  0200              ;  MCS BERLIN
 0060  0200              ;*******************************************************
 0070  0200              ;
 0080  0200              ;
 0090  0200                     *=$F600
 0100  F600              PORTA  *=*+1
 0110  F601              PORTAR *=*+1            ;RICHTUNGSREGISTER
 0120  F602              PORTB  =*+1
 0130  F603              PORTBR *=*+1            ;RICHTUNGSREGISTER
 0140  F604              ;  MONA PORTS FUER DISPLAY, TASTEN ETC.
 0150  F604                     *=$F680
 0160  F680              MPORTA *=*+1
 0170  F681              MPRTAR *=*+1
 0180  F682              MPORTB *=*+1
 0190  F683              MPRTBR *=*+1
 0200  F684              TIMER1 =$F695
 0210  F684              VORBEI =$F687
 0220  F684              ;
 0230  F684              LICHT  =$FB2D           ;MONA-UNTERPROGRAMM
 0240  F684              ;
 0250  F684              OBUF   =$F4
 0260  F684              ;
 0270  F684              ; EINIGE ZEROPAGE-ADRESSEN DIE FUER DAS PROGRAMM
 0280  F684              ; BENOETIGT WERDEN :
 0290  F684              YSAVE  =$D0
 0300  F684              WORT   =$D1
 0310  F684              ZEIT   =$D2
 0320  F684              FREQ   =$D3
 0330  F684              ;
 0340  F684                     *=$200
 0350  0200  40          TABEL  .BYTE $40,$77,$38,$73,$76,$77,$40,$40  ;ALPHA
 0350  0201  77
 0350  0202  38
 0350  0203  73
 0350  0204  76
 0350  0205  77
 0350  0206  40
 0350  0207  40
 0360  0208  7D                 .BYTE $7D,$50,$3E,$73,$73,$79,0,6  ;GRUPPE 1
 0360  0209  50
 0360  020A  3E
 0360  020B  73
 0360  020C  73
 0360  020D  79
 0360  020E  00
 0360  020F  06
 0370  0210  40                 .BYTE $40,$7C,6,$38,$5E,$79,$50,$40  ;-BILDER-
 0370  0211  7C
 0370  0212  06
 0370  0213  38
 0370  0214  5E
 0370  0215  79
 0370  0216  50
 0370  0217  40
```

```
0360  0210  40                .BYTE $40,$78,$50,$79,$6D,$3F,$50,$40 ;IRL'SUR
0360  0211  78
0360  0212  50
0360  0213  79
0360  0214  6D
0360  0215  3F
0360  0216  50
0360  0217  40
0390  0220  78                .BYTE $78,$79,$50,$50,$77,$6D,$6D,$79 ;TERRASSE
0390  0221  79
0390  0222  50
0390  0223  50
0390  0224  77
0390  0225  6D
0390  0226  6D
0390  0227  79
0400  0228                    ;
0410  0228  A9 00             INITIA LDA #0             ;ALLES EINGANG
0420  022A  8D 01 F6                 STA PORTAR
0430  022D  A9 01                    LDA #%00000001
0440  022F  8D 02 F6                 STA PORTB
0450  0232  8D 03 F6                 STA PORTBR        ;PORTB1 = AUSGANG
0460  0235                    ;
0470  0235  20 62 02          RUHE   JSR TESTE         ;UEBERPRUEFE ALLE SENSOREN
0480  0238  B0 FB                    BCS RUHE          ;KEINE "ALPHA" ANZEIGE
0490  023A  20 3F 02                 JSR ANZEIG
0500  023D  F0 F6                    BEQ RUHE
0510  023F                    ;
0520  023F  8A                ANZEIG TXA
0530  0240  A2 64                    LDX #100
0540  0242  86 D2                    STX ZEIT
0550  0244  0A                       ASL A             ; MIT 8 MULTIPLIZIEREN
0560  0245  0A                       ASL A
0570  0246  0A                       ASL A
0580  0247  85 D1                    STA WORT
0590  0249  A5 D1             ANZG0  LDA WORT
0600  024B  85 D0                    STA YSAVE
0610  024D  A2 29                    LDX #$29          ;FUER DIE LINKE ANZEIGENSTELLE
0620  024F  A4 D0             ANZG1  LDY YSAVE
0630  0251  B9 00 02                 LDA TABEL,Y       ;ZEICHEN AUS TABELLE NEHMEN
0640  0254  E6 D0                    INC YSAVE         ;FUER NAECHSTES ZEICHEN
0650  0256  20 2D FB                 JSR LICHT         ;UND IN ANZEIGE LT. X BRINGEN
0660  0259  E0 39                    CPX #$39          ;RECHTS ANGEKOMMEN'
0670  025B  D0 F2                    BNE ANZG1         ;NEIN MEHR ZEIGEN
0680  025D  C6 D2                    DEC ZEIT
0690  025F  D0 E8                    BNE ANZG0
0700  0261  60                       RTS
0710  0262                    ;
0720  0262                    ;
0730  0262  AD 00 F6          TESTE  LDA PORTA         ;BITMUSTER HOLEN
0740  0265  09 F0                    ORA #$F0          ;REST-BITS MASKIEREN
0750  0267  49 FF                    EOR #$FF          ;UND ALLES INVERTIEREN
0760  0269  18                       CLC               ;ALS FLAG FUER "ALPHA"
0770  026A  F0 0E                    BEQ TSTEND        ;WENN ALLES "0" IST KEIN ALARM
0780  026C  A2 01                    LDX #1            ;EIN KONTAKT IST OFFEN
0790  026E  4A                TST1   LSR A             ; WELCHER ?
0800  026F  90 03                    BCC TST2          ;DIESER NICHT
0810  0271  20 7D 02                 JSR ALARM         ;ABER DIESER, X ZEIGT STELLE
0820  0274  E8                TST2   INX
0830  0275  E0 08                    CPX #8            ;SCHON ALLE GEPRUEFT ?
0840  0277  D0 F5                    BNE TST1
0850  0279  38                       SEC               ;ALS FLAG
0860  027A  A2 00             TSTEND LDX #0            ;ALLES IN ORDNUNG
0870  027C  60                TSTEN  RTS
0880  027D                    ;
```

```
0890  027D  48          ALARM  PHA               ;AKKU
0900  027E  8A                 TXA               ;UND X
0910  027F  48                 PHA               ;RETTEN
0920  0280  20 3F 02           JSR ANZEIG        ;ALARMSTUFE 1
0930  0283  20 9F 02           JSR CONDIT        ;ALARM-KONDITIONEN ?
0940  0286  D0 03              BNE ALM1          ;BRANCH WENN NICHT STUFE 2
0950  0288  20 AE 02           JSR ALRM2
0960  028B  AD 02 F6     ALM1   LDA PORTB
0970  028E  29 FE              AND #%11111110    ;BIT 0 ABSCHALTEN
0980  0290  8D 02 F6           STA PORTB
0990  0293  20 9F 02           JSR CONDIT
1000  0296  B0 03              BCS RESTOR
1010  0298  20 CF 02           JSR ALRM3
1020  029B  68           RESTOR PLA               ;WERTE ZURUECK
1030  029C  AA                 TAX
1040  029D  68                 PLA
1050  029E  60                 RTS
1060  029F               ;
1070  029F  A2 FF        CONDIT LDX #$FF
1080  02A1  38                 SEC
1090  02A2  2C 02 F6           BIT PORTB         ;BEDINGUNGEN PRUEFEN
1100  02A5  10 02              BPL COND1         ;BRANCH WENN ALARMSTUFE 2 AUS
1110  02A7  A2 00              LDX #0
1120  02A9  50 01        COND1 BVC COND2
1130  02AB  18                 CLC
1140  02AC  8A            COND2 TXA
1150  02AD  60                 RTS
1160  02AE               ;
1170  02AE  A9 FF        ALRM2  LDA #$FF
1180  02B0  85 D3              STA FREQ          ;FUER TIEFEN ANFANGSTON
1190  02B2  AD 02 F6           LDA PORTB
1200  02B5  85 F4        ALRM21 STA OBUF
1210  02B7  8D 82 F6           STA MPORTB
1220  02BA  A5 D3              LDA FREQ
1230  02BC  C6 D3              DEC FREQ
1240  02BE  F0 0E              BEQ ALRM9         ;ENDE TON
1250  02C0  8D 95 F6           STA TIMER1
1260  02C3  2C 87 F6     ALRM22 BIT VORBEI
1270  02C6  10 FB              BPL ALRM22        ;INTERVALL LAEUFT NOCH
1280  02C8  A5 F4              LDA OBUF
1290  02CA  49 20              EOR #$20          ;BIT UMSCHALTEN
1300  02CC  D0 E7              BNE ALRM21
1310  02CE  60           ALRM9  RTS ENDE DES TONS
1320  02CF               ;
1330  02CF  AD 02 F6     ALRM3  LDA PORTB
1340  02D2  09 01              ORA #%00000001    ;BIT 0 EINSCHALTEN
1350  02D4  8D 02 F6           STA PORTB
1360  02D7  60                 RTS
1370  02D8               ;
1380  02D8                     .END


ERRORS = 0000
```

SYMBOL TABLE

PORTA	F600	PORTAR	F601	PORTB	F602	PORTBR	F603
MPORTA	F680	MPRTAR	F681	MPORTB	F682	MPRTBR	F683
TIMLR1	F695	VORBE1	F687	LICHT	FB2D	OBUF	00F4
YSAVE	00D0	WORT	00D1	ZEIT	00D2	FREQ	00D3
TABEL	0200	INITIA	0228	RUHE	0235	ANZEIG	023F
ANZG0	0249	ANZG1	024F	TESTE	0262	TST1	026E
TST2	0274	TSTEND	027A	TSTEN	027C	ALARM	027D
ALM1	028B	RESTOR	029B	CONDIT	029F	COND1	02A9
COND2	02AC	ALRM2	02AE	ALRM21	02B5	ALRM22	02C3
ALRM9	02CE	ALRM3	02CF				

END OF ASSEMBLY