

Der Monitor 11/87 belegt 4 kB Speicher im Adressbereich 0000 - 0FFF H.

Inbetriebnahme LLC2: Von Kassette laden :
CLOAD "MONITOR" und starten mit DEBUG.

AC 1 : MONITOR auf 4 kB Eprom programmieren (4* U 555, 2* 2716). Einsatz der 4 Eprom anstelle 2 K - Monitor V.3.1 und 2 K - Minibasic (Datei SCCH-Monitor in AC1-Kodierung am Anfang der Kassette).

Weitere Monitor-Versionen : Eprom-Monitor LLC2 und Version für K 7659 -Tastatur siehe gesonderte Anleitungen.

Funktionen Monitor 11/87:

Der SCCH-Monitor umfaßt das Betriebssystem und einen eingebauten Maschinensprache-Monitor (Debugger). Das Betriebssystem (Software) dient der Verbindung der Computerhardware mit den Nutzerprogrammen (Anwendersoftware). Es realisiert die Ein/Ausgabe (Tastatur, Bildschirm, Kassetteninterface, V 24-Schnittstelle, die Verwaltung des Speichers und ist erweiterungsfähig. Die nutzbaren Betriebssystemsroutinen und die Belegung des Arbeitsspeichers sind den Tabellen im Anhang zu entnehmen.

Einschalten: Nach dem Einschalten des Computers übernimmt SCCH-Monitor die Initialisierung des Systems und meldet sich mit der Überschriftzeile " MONITOR SCCH..." und 2 Zeilen weiter mit dem Promtsymbol (Doppelkreuz), dahinter blinkt der Cursor. Das Betriebssystem befindet sich jetzt im Maschinensprache-Monitor, es lassen sich die Befehle des Debuggers aufrufen, Anwenderprogramme laden und starten oder in eine andere Betriebssystemebene springen (Basicinterpreter, Programmpaket X).

Tastatur: Neben der Eingabe von Zeichen und Zahlen erfolgt in Verbindung der Kontrolltaste die Eingabe von Steuerfunktionen. Diese sind notwendig, um z.B. fehlerhafte Eingaben zu korrigieren (Zeichen überschreiben, einfügen, löschen usw.).

Steuer-/Editier-/Kursortasten: (Taste Control und Taste Buchstabe gleichzeitig drücken)

Code	Taste	Funktion
01 H	A	Home, Cursor oben links
02 H	B	Bildschirm ab Cursorposition löschen
03 H	C	Zeile ab Cursorposition löschen
04 H	D	Delete; Zeichen löschen, Zeile rückt nach links
05 H	E	Insert; Space einfügen, Zeile rückt nach rechts
06 H	F	Cursor an den Anfang der Zeile
07 H	G	BEL, akustisches Signal
08 H	H	Cursor nach links
09 H	I	Cursor nach rechts
0A H	J	Cursor nach unten
0B H	K	Cursor nach oben
0C H	L	Bildschirm löschen, Cursor Home
0D H	M	CR, Cursor an Anfang nächster Zeile; Enter
0E H	N	Cursor direkt positionieren (siehe Seite 2)
0F H	O	Tabulator 3 Spalten
10 H	P	Folgende Zeilen normal darstellen
11 H	Q	Folgende Zeilen invers darstellen (bis Zeilenende bzw. Ctl. + P)
18 H	X	Drucker ein, V 24 - Schnittstelle
19H	Y	Ein-/Ausgabe normal, Drucker aus
5F H, 7F H		Cursor nach links und Zeichen löschen (Del)

Ctrl. + N: Direkte Kursorpositionierung

Nach Ctrl. + N steht der Kursor Home, Mit dem Zifferntasten dezimale

Bildschirmposition angeben:

1. Reihenposition (00 bis 31)

2. Zeilenposition (00 bis 63)

Der Kursor kann an jede beliebige Bildschirmposition gebracht werden. Dargestellte Zeichen werden dabei nicht verändert (Der Kursor blinkt abwechselnd zum Zeichen, Fehlerhafte Zeichen können nur durch Überschreiben mit anderen Zeichen, mit Delete oder mit DEL/Rubout korrigiert werden.

Um den Kursor um mehrere Bildschirmpositionen zu bewegen oder Zeichen mehrfach zu schreiben, braucht die entsprechende Taste nur gedrückt zu bleiben. Das Zeichen wird mit erhöhtem Tempo ständig wiederholt (Repetierfunktion). Wird der Bildbereich nach oben verlassen, erscheint der Kursor an gleicher Stelle am unteren Bildrand.

BREAK-Taste/NMI: Die entprellte MNI-Taste ist mit dem NMI-Eingang der CPU zu verbinden. Im Monitor dient diese Taste zum Programmabbruch mit Rücksprung in die Maschinensprache-Monitor-Ebene, Registeranzeige mit #R (CR), Programmfortsetzung mit # G (CR) möglich.

Grafiktaste: Diese Taste wird am PIO (AC1 - D17). LLC2 - D44) B2 angeschlossen (siehe Anhang). Achtung: 1 Byte im Monitorprogramm ändern! Mit gedrückter Grafiktaste schreibt die Tastatur die Grafikzeichen ab Code 128.

Hinweis: im Maschinensprache-Monitor werden Zeichen ab Code 128 nicht dargestellt. Diese Taste wird daher erst in Verbindung mit GRAFIK-BASIC-Interpreter V.3.1 sinnvoll. Auch wird ein Zeichengenerator-Eprom 2K erforderlich. Die Grafikzeichen entsprechen dem Z 1013 (auch KC 85/1, KC87).

Befehlsroutinen des Monitors:

Befehle können eingegeben werden, sobald sich der Monitor mit seinem Promtsymbol (Doppelkreuz) meldet. Dabei hat die Befehlszeile folgende Form:

Xaaaa bbbb cccc

X - ist das entsprechende Zeichen, mit dem das Kommando abgekürzt wird. Hinter dem Kommandozeichen können bis zu 3 max. 4-stellige Argumente folgen (hexadezimal). Hinter jedem Argument muss mindestens 1 Leerzeichen stehen, führende Nullen können entfallen. Werden weniger Argumente angegeben, als zu dem Kommando erforderlich, wird für die Fehlenden der Wert 0 gebildet.

X: die Angabe des Doppelpunktes hinter dem Kommandozeichen bewirkt, dass die Argumente des vorhergehenden Kommandos verwendet werden. die Befehlszeile wird mit der Taste "Enter" (entspricht Taste "CR") vom Bildschirm übernommen, analysiert und ausgeführt. Wird ein Befehl nicht verstanden, erfolgt Ausschrift des Help-Menüs (siehe Debuggerfunktionen "?").

Die Übernahme der kompletten Befehlszeile vom Bildschirm in den Computer hat den Vorteil, dass Korrekturen jederzeit noch möglich sind oder ein Befehl mehrfach ausgeführt wird, indem der Kursor erneut in die Befehlszeile positioniert wird (beliebige Position hinter Doppelkreuz).

Debugger:

Der Debugger dient insbesondere der Testung, Fehlersuche, Entwicklung von Maschinenprogrammen. Weiterhin sind Kommandos zur Ein/Ausgabesteuerung (Kassette, Drucker) vorhanden.

Die Kommandozeichen sind mit den geforderten Argumenten zu übergeben. In Klammern stehende Argumente sind nicht unbedingt erforderlich.

Tabelle 1 Maschinensprache - Monitor / Debugger

Befehlssatz:

A	Arithmetik	Addition, Subtraktion, Disp, Dezimal
b	Basic	Start des Basic-Interpreters V. 3.2
B	Breakpoint	Unterbrechung setzen, nur im RAM
C	Compare	Vergleich von zwei Speicherbereichen
D	Dump	Hexdump und ASCII-Darstellung, sowie Modifikation
e	Requelle	Start des Reassemblers (von Kassette laden)
E	Einzelschritt	Befehl abarbeiten und Register anzeigen
F	Find	Suchen nach Bytes und Zeichenketten
G	Go	Sprung in Programm ab Adresse PC
H	I/O	Zuordnung der Ein- und Ausgabekanäle
I	Init	RSA löschen, SP initialisieren
J	Jump	Sprung in das Programm ab Adresse
K	Kassette save	Speichert Programm od. Datei auf Kassette, Turbo
L	Load	Läd ein Programm od. Datei von Kassette, Turbo
M	Memory	Hexdump und Zeilenprüfbit, Modifikation
N	CRC	Checksumme berechnen
O	Test	Testet RAM ab Adresse
P	Pattern	Füllt Speicherbereich mit Datenbyte
Q	Load AC1	Laden von Kassette, AC1-Format
R	Register	Ausgaben der RSA-Register und Modifikation
S	Save AC1	Speichert Programm im AC1-Format auf Kassette
T	Tranfer	Kopiert Speicherbereich in anderen Speicherbereich
U	U 555	Programmieren U 555 / 2708
V	Verify AC1	Vergleicht Speicherinhalt und Kassette im AC1-Format
W	Lesen Eprom	Auslesen von U 555/2708 in den Speicher
X	Exit	Sprung in das Betriebssystem Programmpaket X
Z	Verify	Vergleicht Speicher, Kassette in Turbo-Tape
>	größer als	Ändert beliebige Anzahl von Byte ab Adresse
?	Help	Liste der verfügbaren Kommandos

§ aaaa bbbb cccc

Compiler

Ermöglicht das Verschieben von Maschinenprogrammen im Speicher. Das Programm ist in den neuen Arbeitsbereich zu laden (Transfer über T oder versehen mit Q+, Q- laden).

Befehl § ausführen.

aaaa= alte Anfangs-, bbbb= altes Ende-, cccc= neue Anfangsadresse

Der Compiler rechnet jetzt folgende Befehle um: Unterprogrammaufrufe, Sprungbefehle, direkte und indirekte Adressierung über Doppelregister, soweit diese im Adreßbereich liegen. Z.B. C3, CD, CC, C4, 21, 22, 3A, DD2A, FD21, ED48, ED53. Um Fehler festzustellen, werden die geänderten Befehle mit nachfolgenden Bytes ausgeschrieben (Anfang der Zeile Adressen).

Compiler bei Bedarf von Kassette nachladen

A aaaa bbbb

Arithmetik

Es werden berechnet: Sprungabstand (für relative Sprungbefehle), Summe, Differenz (aaaa zu bbbb), dezimaler Wert aaaa.

b

Basic

Kaltstart des Basicinterpreters V. 3. 2 (Modul 1)

B aaaa

Breakpoint

Setzen eines Softwarehaltepunktes auf die Adresse aaaa. Der Breakpoint wird bei J-Befehl geladen (RST 38 H, 1 Byte-Befehl). Mit Erreichen des Breakpoint eines Programmablaufes wird das Originalbyte wiederhergestellt und der Monitor springt in den Einzelschrittmodus (siehe E). Jeder Breakpoint wird nur einmal aktiviert, erforderlichenfalls ist der gleiche Breakpoint neu einzugeben.

C aaaa bbbb cccc

Compare

Vergleicht die Speicherinhalte ab aaaa und bbbb mit der Länge cccc. Ungleichheiten werden mit Adress- und Datenbyte durch Taste "CR" fortlaufend angezeigt. Jede andere Taste bewirkt vorzeitigen Abbruch.

D aaaa (bbbb)

Dump

Der Speicherbereich von Adresse aaaa bis bbbb wird angezeigt.

Angezeigt wird: Adresse, 8 Datenbytes, ASCII-Interpretation. Das Bildrollen stoppt nach 19 Zeilen. Taste "CR" läßt das Bild 19 Zeilen, Taste "Space" 1 Zeile weiterrollen. Datenbytes können überschrieben werden, dazu wird der Cursor in die entsprechende Zeile positioniert und die erforderlichen Datenbytes überschrieben. Die geänderten Bytes der Zeile werden mit Taste "CR" übernommen. Wurde versucht, Eprom-Datenbytes zu überschreiben, erfolgt eine Fehlermeldung (Error) und der richtige Inhalt wird wieder angezeigt.

e

Requelle

e aaaa bbbb cccc

Reassembliert Programm ab Adresse aaaa bis bbbb (cccc= Anfang Programm, somit verschobenes Laden möglich und legt ab \$ 6000 die Quelle ab. Weitere Bearbeitung mit EDAS*4 möglich 2k oder Save mit K.: 1. Lauf-Marken ablegen ab \$ 3000. 2. Lauf: Taste A bis Ende, Taste D Einzelschritt DEFB, Taste CR Einzelschritt Befehl.

1) Bei Bedarf von Kassette nachladen. (c) J. Beisler

2) EDAS*4 siehe FA 1/87, Starten: J4000 6000 und aktivieren mit 1.

E aaaa ; E

Einzelschritt

Befehlsabarbeitung eines Anwenderprogramms ab Adresse aaaa oder ab PC (E:) im Schrittmodus.

Die Prozessor-Register werden automatisch mit den Werten des RSA geladen, welche bei Bedarf vorher mit Befehl R gesetzt und angezeigt werden können. Der Stackpointer ist vorher unbedingt mit R oder I auf definierten Wert zu bringen!

Angezeigt werden bei ausführlichen Befehl: Adresse, Befehl, 4 Hauptregister, Flag. In der nächsten Bildzeile wird die Adresse und Bytes des nachfolgenden Befehls bereits angezeigt.

Befehlsausführung jeweils mit der Taste "CR".

Gesamt werden 32 Befehlszeilen gleichzeitig auf dem Bildschirm dargestellt, womit sich Programmfehler gut zurückverfolgen lassen.

Weitere Funktionen:

Taste "R" Sprung in Befehl R. Es kann der gesamte Registersatz dargestellt und neu gesetzt werden (siehe R).

Taste "B" Anzeige des alten Breakpoint, ein neuer Breakpoint kann eingegeben werden. Somit können z.B. UP-Aufrufe übersprungen werden.

Taste "G" Fortsetzung des Programmablaufs in Echtzeit bis zum Erreichen eines neuen Breakpoint.

Taste "Q" Einzelschrittmodus beenden, Rückkehr in Monitor. Die CPU-Register werden in der RSA zwischengespeichert. Der Einzelschritt kann jederzeit mit E: an gleicher Stelle fortgesetzt werden.

Folgende Befehle dürfen nicht im zu testenden Programm stehen:

IMO; IM1; DI; EI; LD I,A

Breakpoint kann nur im RAM gesetzt werden, Schrittmodus ist im ROM/RAM-Bereich möglich. Der Kanal 0 der CTC darf ebenfalls im Programm nicht verändert werden.

F aaaa bbbb cc dd ...

Finding String Hex

F aaaa bbbb 'ABC....'

Finding String ASCII

Der Speicher wird von der Adresse aaaa bis bbbb nach der Bytefolge cc dd ... (Länge beliebig) durchsucht. Suche nach ASCII-Zeichen ist ebenfalls möglich, diese ist hinter dem Hochkomma anzugeben. Sämtliche im Speicher gefundene Zeichenketten werden aufgeschrieben (Anfangsadresse). Ist die Kette sooft vorhanden, dass nicht alle Adressen auf den Schirm passen, wird die Suche abgebrochen; günstig ist vorher den Bildschirm zu löschen. bbbb max. FFFE.

G **Go on**

Start eines Programms ab Adresse PC, siehe Befehl J.

H **I/O - Byte**

Angezeigt wird das Ein/Ausgabe-Byte, welches die Zuordnung der Ein-Ausgabeknäle darstellt. Normal (nach Einschalten, auch Taste BREAK/NMI) ist der Wert 11, somit Eingabe Tastatur und Ausgabe Bildschirm. Bit-Zuordnung:

bit 0	Tastatur	bit 4	Bildschirm
bit 1	V 24 (Rs 232c)	bit 5	V 24 (RS 232 c)
bit 2	Reserve	bit 6	Reserve
bit 3	User	bit 7	User

Mit 0-3 sind für Eingabe, bit 4-7 für Ausgabe zuständig.

Zur Eingabe ist max. 1 Kanal zugelassen, Ausgabe beliebige Anzahl.

Nach dem angezeigten kann ein neuer Wert eingegeben werden, Übernahme mit Taste "CR". Nur Taste "CR" behält den alten Wert.

I **Initialize**

Löschen des RSA (RSA= Zwischenspeicher der CPU-Register für Anwenderprogramme. Die RSA wird genutzt bei den Befehlen: B; E; G; J; R). Der Anwenderstack wird initialisiert, Adresse wird angezeigt (SP xxxx).

J aaaa **Jump**

Sprung in Anwenderprogramm ab Adresse aaaa. Die CPU-Register werden mit den Werten der RSA geladen. Ein zuvor eingegebener Breakpoint wird aktiviert. Wird im Programmablauf ein Breakpoint erreicht, erfolgt Registeranzeige und der weitere Ablauf im Einzelschritt (siehe E).

K aaaa bbbb ("name") **Kassette bespielen Programm Turbo-Tape**
K aaaa bbbb +("name") **Kassette bespielen Datei Turbo-Tape**

Abspeichern eines Files von Adresse aaaa bis bbbb auf Kassette. Ein Name kann bis zu 16 Zeichen lang sein. Das Bandgerät ist mit dem akustischen Signal zu starten, bei den weiteren ab ak. Signal wieder abzuschalten (Stellung Aufnahme). Bei Einsatz der elektronischen Start/Stop-Schaltung erfolgt der Start automatisch (siehe Schaltbild).

Am Ende wird die CRC-Prüfsumme berechnet und ausgeschrieben.

Turbo-Tape lädt 1 kByte in ca. 3 sec, bbbb bis FFCF zulässig.

Autostart und Interrupt siehe L.

L ("name") **Laden Programm Turbo-Tape**
L #("name") **Laden Datei Turbo-Tape**

Der Computer lädt ein Programm oder Datei in den Speicher, ein zuverlässiges Suchsystem ist vorhanden:

Ohne Namensangabe wird das nächste Programm geladen (# L (CR)).

Mit Namensangabe wird das Programm gleichen Namens geladen, andere Programme werden aufgelistet (Found name). Der Name kann beliebig abgekürzt werden, wenn Verwechslungen ausgeschlossen sind (z. B. L "BA (CR) für Laden des BASIC-Interpreter).

Von jedem ladenden bzw. gefundenen Programm werden angezeigt:

Name, Anfangsadresse, Endadresse, Filetype (siehe Anhang).

Nach dem Laden werden zusätzlich Lesefehler angezeigt (ERROR Blockendadressen) und Laden nach defekten RAM-Speicher (ERROR RAM Blockadressen). Werden mehr als 9 Blöcke fehlerhaft gelesen, erfolgt vorzeitiger Abbruch. Laden kann mit der Taste BREAK/NMI abgebrochen werden. Die Adressen werden nach arg. 1+2 übergeben, mit N; kann die CRC Prüfsumme angezeigt werden.

Autostart: Nach dem Laden prüft der Computer das Programm auf vorhandene Autostart-Sequenz. Ist diese vorhanden, erfolgt der Autostart auf entsprechender Adresse. Die Autostartsequenz kann vor jedem Programm gesetzt werden (5 Byte): **53 43 48 xx**; xx = Startadresse.

Interrupt: Der Interrupt wird während Save (K) und Load(L) gesperrt. Somit brauchen Interrupt-Programme (z. B. Uhr) nicht abgeschaltet werden.

Akustische Signale und Start/Stop siehe K.

M aaaa (bbbb)

Memory

Der Speicherbereich von Adresse aaaa bis bbbb wird angezeigt (Hexdump).

Angezeigt werden: Adresse, 16 Datenbytes, Zeilenprüfbyte. Das Bildrollen stoppt nach 16 Zeilen. Taste "CR" lädt das Bild 16 Zeilen, Taste "Space" 1 Zeile weiterrollen.

Datenbytes können beliebig überschrieben werden, siehe Befehl: D.

Nach Abbruch kann mit # K : (CR) der folgende Bereich angezeigt werden.

Das Zeilenprüfbyte wird entsprechend FA 11/86 berechnet und ermöglicht ein fehlerfreies Eintippen von Programm listings.

N aaaa bbbb

CRC - Prüfsumme

Prüfsummenberechnung über Speicher von Adresse aaaa bis bbbb entsprechend des CRC- (SDLC)-Standard.

O aaaa

RAM - Test

Ermitteln der höchsten verfügbaren RAM-Speicher-Adresse, aaaa entspricht dem RAM-Anfang (mindestens 1000). Der Test verändert den RAM-Inhalt nicht.

P aaaa bbbb cc

Pattern

Speicherbereich von aaaa bis bbbb mit Datenbyte cc füllen.

Q (aaaaa) +; -; = (name)

Laden Programm AC1-Kode

Der Computer lädt ein File von Kassette in den Speicher. Kodierung entspricht dem AC1-Standard (s. FA 10/84). Das Programm kann um aaaa verschoben (+ oder -) oder auf gleiche Adresse geladen werden (# Q=).

Ein Suchsystem ist vorhanden:

Ohne Angabe des Namens erfolgt Laden des 1. gefundenen Programms (# Q= (CR)).

Mit Angabe des Namens (z. B. # Q= BASIC (CR)) wird jedes gefundene Programm angezeigt. Bei Erreichen des Programms gleichen Namens erfolgt Laden. Der Name kann entsprechend Befehl L abgekürzt werden. Anführungszeichen sind nicht zu verwenden, dafür muss zwischen Befehl und Name genau 1 Leerzeichen stehen. Das Suchsystem vergleicht die ersten 6 Zeichen des Namens, weitere Zeichen sind ohne Bedeutung.

Während des Ladens blinkt zur Kontrolle ein Stern, nach dem Laden wird die Startadresse ausgeschrieben oder ein Autostart durchgeführt (siehe Befehl S) Start mit # J: (CR) möglich.

Akustisches Signal und Start/Stop siehe K. Interrupt siehe L.

Lesefehler: Das Laden wird abgebrochen, angezeigt werden Ursache (ERROR Checksum oder ERROR RAM) und Fehleradresse.

R (aa)

Register

Anzeige und Ändern der CPU-Anwender-Register (RSA). # R (CR) erfolgt Anzeige des gesamten Registersatzes, des Breakpoint, Breakpointsequenz sowie gesetzte Flags anhand ihres Symbols.

Zum Ändern einzelner Register ist für aa der Name des jeweiligen Registerpaares anzugeben: HL, DE, BC, AF, HL', DE', BC', AF', BP, IX, IY, PC. Der alte Bereich wird angezeigt, gefolgt von #. Ein neuer Wert ist anzugeben, Übernahme mit (CR).

S aaaa bbbb cccc name

Save AC1-Kodierung

Ein Programm von Adresse aaaa bis bbbb wird auf Kassette gespeichert. Kodierung entspricht dem AC1-Standard. Der Name kann bis zu 40 Zeichen lang sein (wird das Programm mit AC1-2k-Monitor geladen, erfolgt Anzeige der ersten 6 Zeichen). cccc ist die Startadresse des Programms. Wird Auto-Start vorgesehen, ist CCCC die Adresse der Autostartsequenz (siehe L). Somit kann die Autostartsequenz beliebig im Programm liegen: empfohlen wird vor Programmanfang. Akustisches Signal, Start/Stop, Interrupt siehe K und L.

T aaaa bbbb cccc

Transfer

Speicherbereich verschieben von Adresse aaaa nach bbbb. cccc ist die Länge. Überschneiden der Bereiche ist zulässig.

U aaaa

U 555 programmieren

(Befehl nicht in Monitorversion K 7659-Tastatur, wird von Kassette geladen!) aaaa ist die Anfangsadresse des zu programmierenden Speicherbereichs, Länge 1 kByte. Nach Start testet das Programm den Eprom, ob programmieren möglich ist (Inhalt einer Zeile von H auf L setzen). Wenn nicht, erfolgt Fehlermeldung "CLS EPROM", der Eprom ist jetzt zu löschen. Das Programmieren erfolgt nach Vorschrift U 555 (2708) und dauert ca. 65 sec. Das Ende wird akustisch signalisiert und die Eprom-Prüfsumme (CRC) ausgeschrieben. Bei fehlerhafter Programmierung werden Adresse, Fehlerbyte, Datenbyte fortlaufend mit Taste "CR" angezeigt.

V= (name)

Verify AC1-Kodierung

Vergleich einer Bandaufzeichnung mit Speicheroriginal, um Datenfehler aufgrund mangelhaften Bandmaterials festzustellen.

W aaaa

U 555 auslesen

(Befehl nicht in Monitorversion K 7659-Tastatur, von Kassette laden!) 1 kB-Eprom auslesen in den RAM-Speicher ab Adresse aaaa und berechnen der Prüfsumme (CRC).

X

Exit

Sprung in weiteres Betriebssystem. Vorgesehen ist Programmpaket X ab \$ E000. Näheres siehe Bauanleitung Modul 1.

Z ("name")

Verify Turbo-Tape

Vergleich einer Bandaufzeichnung mit Speicherbereich, um Bandfehler festzustellen. Bei Verify Datei zusätzlich Stern (*) vor Name

> aaaa bb cc ...

Beschreiben Speicherbereich

Hinter der Anfangsadresse aaaa können beliebig lange Bytefolgen eingegeben werden, welche durch mindestens 1 Leerzeichen voneinander getrennt sind. Die Übernahme erfolgt mit Taste "CR", gleichzeitig wird der beschriebene Bereich geordnet mit 8 Datenbytes, ASCII-Interpretation angezeigt. Schreibfehler werden ebenfalls angezeigt (def. RAM, ROM- Bereich).

? Help

Help

Der Speicherbereich wird nach Startkommandos durchsucht und diese aufgelistet. Neben des Debuggers werden Startbefehle geladener Anwenderprogramme angezeigt.

Startroutinen des Monitor:

Diese werden anhand eines Kodemuster im Adressbereich \$ 400 bis \$ C000 gesucht. Jede Routine hat die Form:

00 09 xx 0D ... Programm ... C9

xx ist der ASCII-Code des Start-Kennbuchstaben, Eigene Programme brauchen nur in diesen Rahmen gepackt zu werden und können über Kennbuchstaben (Zeichen/Ziffer) gestartet werden.

Tubo-Tape:

Es werden 4 File-Typen unterschieden:

P	lauffähiges Maschinenprogramm	B	Basicprogramm
D	Maschinendatei	F	Datenfeld (Basic)

Die Filetypen B und F können nur mit Basicinterpreter V.3.1 0 oder 4 geladen werden. Im Monitor erfolgt lediglich Anzeige des Namens, Filetype, Anfangs- und Endadresse.

Ein-/Ausgabe

Die Verbindung des Computers mit der Umwelt erfolgt über die Ein/Ausgabe. Es stehen jeweils 4 Möglichkeiten zur Verfügung:

Eingabe : Tastatur, V24 (Rs 232c), Reserve, User

Ausgabe: Bildschirm, V 24 (RS 232c), Reserve, User

Die Ein/Ausgabe-Kanäle werden über das I/O-Byte adressiert (siehe Befehl H). Nach einschalten des Computers wird Ein/ausgabebyte (Adresse \$ 1824) auf \$ 11 gesetzt, somit Tastatureingabe und Bildschirmausgabe.

Setzen der Ein/Ausgabe:

- Debuggerbefehl H für beliebige Zuordnungen
- Taste Control + Taste X schaltet zusätzlich Ausgabe über V 24 ein.
- Taste Control + Taste Y schaltet Ein/Ausgabe auf Standard (\$ 11).
- Taste BREAK/NMI setzt Ein/Ausgabe auf Standard (\$ 11).
- RESET-Taste verändert Ein/Ausgabe nicht!

Die Ausgabe verarbeitet Datenbytes 00-7F H, womit der ASCII-Code (0-7F) verarbeitet wird, bit 7 wird dabei auf 0 gesetzt. Die Ausgabe läßt sich auf 00-FF H umstellen, indem bit 7 des V24-Kontrollregister gesetzt wird.

Adressen:	Eingabekanal	\$ 08	(dez. 8)	
	Ausgabekanal	\$ 10	(dez. 16)	
	I/O-Byte	\$ 1821		
	V24-Kontrollregister	\$ 1820		
	User-Eingabe	\$ 18F0	USER-Ausgabe	\$ 18F3
	V24-Eingabe UP	\$ 01D6	V24-Ausgabe UP	\$ 0D F9

V24-Schnittstelle (RS 232c)

Die standardisierte Schnittstelle ermöglicht den Anschluss von Drucker, Modem u. a. Peripheriegeräten und die Kopplung mit weiteren Computern. Die Schnittstelle arbeitet mit TTL-Pegel, zum Anschluss von Peripherie wird in der Regel eine Pegelanpassung (-12V, +12V) benötigt, siehe Schaltbild.

V24 ist eine serielle Schnittstelle. Zur Kopplung können wahlweise 3 Leitungen (3 Draht Handshake) oder 5 Leitungen (X Draht-Handshake) eingesetzt werden. Letzteres ist zur Kopplung mit langsamer Peripherie (Drucker) notwendig. Zur Kopplung von 2 Computern genügt der schnelle 3-Draht-Handshake. Über das Kontrollregister werden die Standardwerte der Schnittstelle eingestellt, siehe Tabelle. Nach Einschalten des Computers sind folgende Werte voreingestellt: \$ 02

9600 Baud
 1 Stop-Bit
 ohne Parität
 X-Draht-Handshake
 8 Datenbit (Ausgabe bit 7 auf 0 gesetzt)

Somit kann übliche Peripherie (Drucker) unmittelbar betrieben werden, z. B. Schreibmaschine S 6006.

Eventuell niedrigere Baud-Rate wählen, günstig 300 bis 2400 Baud!

V24 - Kontrollregister (\$ 1820):

Baudrate	bit 2	bit 1	bit 0	dez.
300	1	1	1	7
600	1	1	0	6
1200	1	0	1	5
2400	1	0	0	4
4800	0	1	1	3
9600	0	1	0	2
19200	0	0	1	1

Stop-Bits (Ausgabe) :	bit 3	dez.
1	0	0
2	1	8

Parität (Ausgabe):	bit 5	bit 4	dez.
8 Datenbit, ohne P.	0	0	0
7 Datenbit+P. ungerade	0	1	16
7 Datenbit+P. gerade	1	0	32

Handshake (Ausgabe)	bit 6	dez.
3 - Draht	1	64
X - Draht	0	0
Ausgabe:	bit 7	dez
8 Datenbit	1	128
7 Datenbit 0-6 + bit 7- 0	0	0

Der Eingabekanal arbeitet ohne Paritätskennung, eine Einstellung der Stop-Bits und Handshake ist nicht notwendig.

Hardware: Die serielle Schnittstelle verwendet 4 Kanäle (a0-A3) der PIO 2 (LLC2 : D 47). Bei AC1 ist PIO 2 zu erweitern, Adressierung über IOSEL 2.

Die PIO hat gegenüber einer SIO den Vorteil des geringeren Aufwandes. Die Baudraten werden softwaremäßig getaktet und sehr genau eingehalten (+-0%).

PIO 2	A0	Eingang Daten	RxD	Buchse:	1
	A1	Ausgang Daten	TxD	entspr. KC85	3
	A2	Eingang empfangsbereit	CTS		4
	A3	Ausgang sendebereit	DTR		5

Die Leitungen werden auf eine 5-polige Diodenbuchse gelegt. 2 Computer (AC1/LLC2) können über ein Überspielkabel gekoppelt werden. Die Kopplung mit weiteren Computern (z. B. Commodore 64, +4, KC 85) erfolgt entsprechend der Angaben in den Handbüchern. Es können problemlos Daten ausgetauscht werden (z.B. Basicprogramme).

Serielle IEC - Schnittstelle:

Es handelt sich nicht um die standardisierte Commodore-Schnittstelle.

Ermöglicht wird somit die Verwendung der Commodore-Peripherie, insbesondere Diskettenlaufwerk Typ 1541, 1570, 1571 und Commodore-Drucker. Im Monitor sind keine Routinen zur Unterstützung dieser Schnittstelle enthalten. Diese Routinen werden im CP/M-kompatiblen System und in Basic V.4 implementiert.

Hardware.

			Buchse 6 PIN
PIO 2	A4	SRQ	1
	A5	ATN	3
	A6	CLK	4
	A7	DATA	5
	RESET	RESET	6

EPROM-Programmer 2708 - 27512:

Der Eprommer arbeitet mit PIO 2 (LLC2 D47) Port A + B.

Zu beachten ist, dass PIO 2 mehrfach genutzt wird (V24, IEC). Vor Anschluss des Programmers V.1.1/3 ist die Peripherie zu trennen. Port A der PIO 2 ist auf Modus 2 zu initialisieren, Eprommer V.1.1/2 entsprechend erweitern (Eprommer V 1.3 ist bereits erweitert!):

LD A,BF H	AC1: 3EBF	LLC2: 3EBF
OUT PIO 2/A	D30A	D3E6

Als Trick kann auch Befehl # U (CR) gestartet werden, anschließend Eprommer V.1.1 # * (CR).

Joystick

Es können 1 oder 2 Stück angeschlossen werden, siehe Schaltbild. Die Abfrage des Joysticks 1 kann in Anwenderprogrammen über UP "Joy" erfolgen, siehe UP-Verzeichnis.

Weitere Betriebssysteme:

Die Anwendung weiterer Betriebssysteme ist im Monitor vorbereitet. Vorgesehen sind Basic-Interpreter V.3.2/4 und Programmpaket X (CP/M).

Die weiteren Betriebssysteme werden fest auf Eprom installiert; entsprechend Bauanleitung Modul 1. (Bei Bedarf Info 9/87 anfordern)

Welchem Betriebssystem die höhere Priorität zusteht, kann der Anwender frei festlegen. Um in das jeweilige Betriebssystem zu gelangen, gibt es verschiedene Möglichkeiten:

Starten des Basic-Interpreters V.3.3: # b (CR)

(Rückkehr in den Monitor mit BYE).

** S C C H M o n i t o r * Version Nov. 1987 * Seite 10 **

Starten des Programmpaket X:

1. # X (CR)
2. Taste X drücken und RESET betätigen
3. Taste X während des Einschaltens des Computers drücken

Mit Einsatz des Basicinterpreters V.4 ist vorgesehen, beim Einschalten des Computers direkt in Basic zu springen. Basic V.4 wird vom Betriebssystem automatisch erkannt, indem beim Einschalten des Computers (bzw. RESET) für Modul 1 die Speicherkonfiguration Basic eingestellt wird und anschließend die ersten 3 Speicherzellen (\$ 2000 - \$ 2002) abgefragt werden. Befindet sich dort die Bitfolge 53 43 48, erkennt der Computer ein Autostart-Modul und springt an die Adresse, die in \$2003/\$2004 steht. Der Autostart kann verhindert werden, indem beim Einschalten bzw. RESET die Taste "CR" gedrückt wird: Sprung in den Monitor, oder Taste "X" gedrückt: Sprung in Programmpaket X.

Betriebssystem Monitor SCCH:

Ablauf nach Einschalten des Computers:

- Interrupt wird gesperrt, - SP initialisiert, - Initialisierung der Peripherie (PIO, CTC), - Warmstartvektor abfragen, - Sprungtabelle für RST, NMI in den RAM übertragen, - Warmstartvektor, - Ein/Ausgabe-Byte, - V24-Kontrollbyte in den RAM übertragen, - Anwender-RAM löschen, Interruptregister wird gesetzt, - Abfrage Tastatur auf gedrückte Taste X oder CR: Taste X gedrückt: Konfiguration für Modul 1 einstellen, Sprung nach \$E000.

Taste CR gedrückt:

Die Routine zur Erkennung Autostart wird übersprungen.

- Modul 1 prüfen auf Auto-Start-Routine, wenn ja, Sprung zu Modul1
- Bildschirm löschen und Überschrift, - Tastaturschleife/Warten auf Eingabe Degugger/Monitor

Betriebssystem, Routinen, Unterprogramme:

Adresse	Funktion	(verwendete Register)
\$ 0008	Eingabekanal, normal Tastatur	(A)
\$ 0010	Ausgabekanal, normal Bildschirm	(A)
\$ 0018	Zeichenkettenausgabe bis einschließlich das Byte bit 7 gesetzt ist	
\$ 0020	Warmstart Betriebssystem	
\$ 0028	1 byte auf Kassette save, AC1-Format	
\$ 0030	1 byte von Kassette laden, AC1-Format	
\$ 0033	Ausgabe über Bildschirm	(A)
\$ 0038	Breakpoint	
\$ 0066	MNI-Befehl, Programmunterbrechung	
\$ 00A5	UP "Eingabekanal"	(A)
\$ 011D	UP "NS 30" Zeitschleife 30 ms	
\$ 0128	UP "Inline", liest Zeile von Bildschirm ein (Adresse in \$ 181A)	
\$ 0146	UP - Routine zu InHex	
\$ 0172	UP "InHex", wandelt eine 4-stellige Zahl im ASCII-Code in Hexzahl um. DE = Zeiger auf Zahl (Bildschirm), Ergebnis in HL (DE, HL)	
\$ 0183	UP "OutHex", gibt A als 2-stellige Hexzahl auf Schirm aus	(A)
\$ 019A	UP "Out HL", gibt HL-Register als 4-stellige Hexzahl auf Schirm aus	(HL)
\$ 01A5	UP, CPU-Register ausräumen nach RSA	
\$ 01BC	UP, CPU-Register einräumen von RSA	
\$ 01D6	V24 - Eingabe	(A)
\$ 01DE	Warteschleife Koomandozeicheneingabe	
\$ 0220	Tabelle RST, NMI, Hilfsregister, V24, I/O (Voreinstellung)	
\$ 0240	Einsprung bei Programmfehler, Register werden gerettet in die RSA	
\$ 0243	Einsprung NMI, Register werden in die RSA gerettet, I/O und PIO werden auf Standard gesetzt	
\$ 0268	UR "ERROR", schreibt Zeichenkette ERROR	

\$ 0272 UP "akustisches Signal"
 \$ 0287 UP "Ton", B= Tonlänge, C= Tonhöhe (BC)
 \$ 0297 UP "Taste", testet Tastaturtastatur, kehrt bei gedrückter Taste
 nach 18 ms mit dem Code zum Akku zurück, keine Taste-Rückkehr mit

** S C C H Monitor * Version Nov. 1987 * Seite 11 **

gesetzten Z-Flag und A=0 Code steht in \$ 1822 (A)
 \$ 02F6 Interrupt Tabelle PIO/CTC
 \$ 04CA Register mit Argumenten laden, aaaa=HL, bbbb=DE, cccc=BC
 Sprungverteiler:
 \$ 07EB Reserve
 \$ 07EB MS 30 (\$ 0110)
 \$ 07EE OUTHEX (\$ 0183)
 \$ 07F1 OUT HL (\$ 019A)
 \$ 07F4 INLINE (\$ 0128)
 \$ 07F7 INHEX (\$ 0172)
 \$ 07FA TASTE (\$ 0297)
 \$ 07FD Betco (\$ 01DE)
 \$ 0AD8 Tabelle Peripherie
 \$ 0AD8 UP "Zeitschleife 18 ms" zur Tastaturentprellung
 \$ 0EB4 UP "Joy", Abfrage Joystick 1, keine Taste I - Flag gesetzt und A=0,
 oben bit 0, unten bit 1, links bit 2, rechts bit 3, Feuerknopf, bit 4 von Akku gesetzt (A)
 \$ 0F22 Tabelle Tastenkode (nur Monitorversion 8*8 und 8*12-Tastatur
 \$ 0DF9 V24- Ausgabe: Vor Abruf HL mit 1821 H laden! (A)

Speicherbelegung (Memory Map):

Adresse	AC1	AC1/LLC2	LLC2
\$ 0000		Betriebssystem	
\$ 1000	Bildschirmspeicher		2 k freier RAM
\$ 1800		Zwischenspeicher Cursor	
\$ 1802		Sprungtabelle RST 8	
\$ 1805		" RST 10 H	
\$ 1808		" RST 18 H	
\$ 180B		" RST 20 H	
\$ 180E		" RST 28 H	
\$ 1811		" RST 30 H	
\$ 1814		" RST 38 H	
\$ 1817		" NMI	
\$ 181A		Zwischenspeicher "Inline"	
\$ 181C		Warmstartcode	
\$ 181E		Hilfsregister für 8x12-Tastatur (KC7639)	
\$ 1820		Kommandobyte V24	
\$ 1821		Ein/Ausgabebyte	
\$ 1822		Tastencode der zuletzt gedrückten Taste	
\$ 1823		Hilfsregister Routine	
\$ 1824		Hilfsregister Repetierfunktion Tastatur	
\$ 1825		Hilfsregister Cursorpositionierung (Ctl +N)	
\$ 1856		Stack	
\$ 18 57		Hilfsregister UP "Inline"	
\$ 185B		Zwischenspeicher Argument 1	
\$ 185D		Zwischenspeicher Argument 2	
\$ 185F		Zwischenspeicher Argument 3	
\$ 1861		RSA: AF', BC', DE', HL'	
\$ 1869		RSA: AF, BC, DE, HL	
\$ 1871		RSA: IX, IY, PC, SP	
\$ 1879		RSA: BP	
\$ 187B		RSA: Breakpointsequenz	
\$ 187E		Speicher Save/Load Turbo-Tape	
\$ 18D6		freier RAM	
\$ 18F0		User Ausgaberroutine	

\$ 18F3	User Ausgaberroutine
\$ 1900	Anwender-Speicher bis \$ FFFF
\$ 2000	Modul 1: Basicinterpreter V.4
\$ 6000	Arbeitsspeicher, Basicinterpreter
\$ 60F7	Anfang Basic-RAM

** S C C H M o n i t o r * V e r s i o n N o v . 1 9 8 7 * S e i t e 1 2 **

\$ C000	Bildspeicher 2k (Text) 16 k (Pixelgrafik)
\$ CB00	Modul 2: 2k Farb-RAM
\$ C000	Modul 1: ROM-Bank 1 und 2
\$ E000	Betriebssystem Programmpaket X
\$ FFFF	RAM-Ende

Peripherie:

AC1	LLC2	
\$ 00	\$ F8	CTC-Kanal 0-3
\$ 04	\$ EB	PIO 1
\$ 0B	\$ E4	PIO 2
\$ 14	\$ EC	Modul 1: Konfigurationsregister
\$ 15	\$ ED	Modul 3: RAM-Disk/Bank
\$ 16	\$ EE	Modul 2: Mehrfarbenregister
--	\$ E0	ROM abschalten

Belegung PIO 1/B (AC1 D 17, LLC2 D44) bei Originaltastatur

Kanal B	AC1	LLC2
B0	Ton-Ausgang	TB-Interface-Ausgang
B1	Joystick ges. A	TB-Interface-Eingang
B2	Grafiktaste	Grafiktaste
B3	Bildschirm-Mode	Bildschirm-Mode
B4	frei	Joystick ges. A
B5	Start/Stop-Schaltung	frei
B6	TB-Interface-Ausgang	Ton-Ausgang
B7	TB-Interface-Eingang	Start/Stop-Schaltung

Hinweis: Einige PIO-Kanäle werden vom Eprommer V.1.1;2 doppelt mitgenutzt.
Eprommer V.1.1;2 sollte daher steckbar sein und nur bei bedarf angeschlossen werden.

Abweichungen der Belegung bei 8*8 und 8*12 (K 7659-Tastatur):

Kanal B	8*8 (nur AC1)	8*12 (AC1, LLC2)
B1	Taste Shift	AC1; frei
B2	Taste Control	frei
B4	Joystick 1	frei

Einbau der Grafiktaste (nur AC1/LLC2 mit Originaltastatur):

Der einpolige rastende Schalter kommt an PIO 1/B2 und Masse. Von PIO 1/B2 min R=330Ohm und LED VQA 13 nach 5P anschließen. Die LED leuchtet bei eingeschalteter Grafiktastatur.

Änderung im Monitor notwendig:

AC1 : Adresse \$ 02A9 Datenbyte \$ FF in \$ 21 ändern

LLC2: Adresse \$ 02C7 Datenbyte \$ FF in \$ 21 ändern (ROM-Adresse!)

Der Eprom muss nicht gelöscht werden; Datenbyte überprogrammieren

SUPER - TAPE:

Aufgrund der veränderten Memory-Map sind alte Supertape-Programme nicht mehr ladbar! Dies betrifft bis jetzt nur Chessmaster V.2. Die neue Supertapeversion V.2 ist ab sofort zu verwenden. Diese Version wurde erheblich verbessert und belegt jetzt 2 kB (\$ E000-\$E7FF).

Kurzanleitung Supertape V.2:

Supertape ermöglicht, Programme im Supertape-Format zu save. Geladen werden Supertape-Aufzeichnungen normal mit der Funktion "L" des Monitor bzw. "CLOAD" des Basicinterpreters. Zum Laden wird Supertape somit nicht mehr benötigt!

Das Supertape-Format verfügt über folgende Eigenschaften:

Titelbild: Kurz nach Anzeige des Programmnamens erscheint ein Titelbild, indem sich z. B. Programm und Autor vorstellen können. Nach dem Titelbild wird das weitere Programm geladen.

Streifen: In den unteren 4 Bildzeilen ist ein bewegtes Muster, welches eine optische Kontrolle des Ladevorganges ermöglicht.

Autostart: Nach dem Laden erfolgt Autostart des Programms.

** S C C H M o n i t o r * V e r s i o n N o v . 1 9 8 7 * S e i t e 1 3 **

Kopierschutz: Supertape-Programme sind hochwertig kopiergeschützt, da z. B. die RESET- und BREAK-Funktion außer Funktion gebracht werden können.

Start von Supertape: Laden sie das Programm, welches sie im Supertape-Format save wollen. Falls sie mit Titelbild arbeiten, ist dieses zu gestalten. Möglich sind: von Hand, über ein kurzes Basicprogramm oder über grafisches Mal-/Zeichenprogramm.

Das Titelbild ist vom Bildschirm in den Bildspeicher von Supertape zu transferieren, z. B. AC1: # T 1000 EB00 (CR): ILLC2: T C000 EB00 800 (CR)

Starten sie Supertape (# X (CR)). Jetzt werden folgende Eingaben abgefragt:

Programmname (bis 16 Zeichen möglich)

Anfangsadr.: Anfangsadresse des Programms

Endadr. : Endadresse des Programms

Adr. Hilfslager: Adresse für den Hilfslager, möglich \$ 200 vor Anfangsadresse. Wenn dies nicht geht (ROM-Bereich/belegter RAM) in beliebigen freien Bereich von 2,2 kByte.

Startadresse : Startadresse des Programms für den Autostart.

Adr. BREAK/NMI : Damit wird die BREAK-Taste automatisch neu belegt, z. B. Startadresse des Programms. (normal \$ 2243)

Adr. RESET : Neubelegung der RESET-Taste. Damit kann auch die Taste außer Funktion gesetzt werden, auch hier die Startadresse des Programms möglich (no. 5A)

AC1/LLC2 : Es lassen sich Supertape-Aufnahmen für AC1 und LLC2 herstellen, unabhängig von ihrem Computertyp (die Programme müssen natürlich auf dem jeweiligen Computer laufen)

Titelbild (J/N) : Supertape geht auch ohne Titelbild

Start Save : Kassette starten (Aufn) und CR drücken.

Wiederholung J/N : Mit Wiederholung können unzählige Kassetten bespielt werden.

Hinweis : Verwenden sie Supertape nicht bei Weitergabe der kostenlosen SCCH-Programme, eine weitere Weitergabe wäre dann unmöglich.

Copyright: SCCH-Monitor 11/87 und Supertape sind (c) 1987 by E. Ludwig.

Alle Rechte vorbehalten. Programme und Anleitungen können zur persönlichen Verwendung kostenlos weitergegeben und vervielfältigt werden *unter Urheberangaben*.

Monitorversionen 11/87:

AC1:

V.8.0 für Originaltastatur und erweiterte Originaltastatur ersetzt alten Monitor Monitor V.3.1, V.6.0, V.7.0/7.3 ohne Hardwareänderungen.

V.8.1 für Softwaretastatur 8*8 Matrix, ersetzt Monitor V.6.1/7.1. Folgende Verbesserungen wurden vorgenommen:

Taste Shift und Taste Control als zusätzliche Tasten.

Taste *Shift* an PIO 1(B1, Taste *Control* an PIO 1/B2 und Masse.

Die LED Shift und LAE Control bleiben unverändert.

Für Shift empfiehlt sich eine 2. Taste Shift-Lock, diese wurde rastend ausgeführt.

Grafiktaste: Die Umschaltung der 8*8 Tastatur auf Grafikeingabe erfolgt durch gleichzeitiges Betätigen der Shift- und Controltaste.

V.8.2 für Softwaretastatur 8*12 Typ K 7659 besondere Anleitung anfordern.

LLC2:

V.9.1 für Originaltastatur, ersetzt alten Monitor V.7/8.1.

V.9.2 für K7659-Tastatur (8*12 Matrix). Einige Tastaturfunktionen wurden geändert: Taste "Tab"=Tabulator 8 Spalten, Taste "DF" neu Taste Shift-Lock, Taste "F13" neu Taste "DK", Taste "F12" neu Taste Grafik

Taste "DR" neu Taste "F12". Taste Shift und Control müssen zusammen mit jeweiliger Taste betätigt werden. Abschaltung von "Grafik" und "Shift Lock" über Taste "Shift".

V.9.3 Monitorversion für Urlader-Bytes, wird von Kassette geladen.

Die Monitorversionen werden als Datei mit auf ihrer Kassette aufgezeichnet. Ab 1/88 bitte Musterkassette bestellen!