

Die Hardware des DOS-Juniors

Um den Junior-Computer in einen DOS-Computer umzuwandeln, sind einige Hardware Modifikationen nötig. Aber keine Angst! Es müssen keine Leiterbahnen durchgeschnitten oder andere mechanische Eingriffe gemacht werden. Nur ein IC ist auf der Interface-Platine des Junior-Computers Huckepack auf ein anderes zu löten. Für den Anschluß eines EPSON-Matrix-Druckers ist ein Interface für die BUSY-Leitung erforderlich. Da dieses Interface aus nur drei Widerständen, einem Transistor und einer Diode besteht, läßt es sich leicht im "fliegenden Aufbau" neben dem V 24/RS-232-Konnektor befestigen.

Floppy-Disk-Interface für Junior

... und andere 6502-Computer II

Im zweiten und letzten Teil dieser Artikelserie beschreiben wir, was an der Hardware des Junior-Computers geändert werden muß, um die Software von Ohio Scientific auf dem Junior- oder jedem anderen 6502-Computer laufen zu lassen. Ein neues EPROM ist erforderlich, um bei der Initialisierung (Reset) des Computers Software von der Diskette zu laden. Das Source-Listing für das Monitorprogramm im EPROM ist als "Paperware" erhältlich.

Sehen wir uns nochmals das Schaltbild der Junior-Interface-Platine an (Bild 1). Dort wird das NOR-Gatter N33 durch ein NAND-Gatter ersetzt. Die Leitung 8K0 oder EX ist jetzt nicht mehr im Adreßraum \$0000...1FF aktiv. Der neue Adreßraum mit dem NAND-Gatter ist E000...FFFF. Diese Adressierung hat für die Speicherbausteine des Junior-Computers folgende Konsequenzen (siehe Bild 2):

> Im Adreßraum \$0000...BFFF liegen 48 kByte Dynamic-RAM. Dynamic-RAM hat den Vorteil, daß es billig ist und wenig Strom braucht. Drei Dynamic-RAM-Karten (siehe Elektor, April 1982) genügen, um diesen 48-kByte-Adreßraum mit Schreib/Lesespeicher zu belegen.

> Der Adreßdekoder auf der Standard-Junior-Computer-Platine (IC6) dekodiert den Adreßraum \$E000...FFFF. Die Speicherbausteine auf der Standard-Junior-Computer-Platine bekommen deshalb folgende Adressen zugeordnet:

EPROM IC2, Typ 2708: \$FC00...FFFF

PIA, RAM, TIMER, Typ 6532: \$FA00...FBFF (Paperware!)

RAM IC4 und IC5, Typ 2114: \$E000...E3FF

Die Speicherbausteine auf der Junior-Interface-Platine haben die Adressen:

VIA IC1, Typ 6522: \$F800...F9FF (Paperware!)

RAM IC2 und IC3, Typ 2114: \$E400...E7FF

EPROM IC4 und IC5, Typ 2716: \$E800...F7FF

Die zweite Hardware-Modifikation betrifft das Interface für die BUSY-Leitung des EPSON-Druckers. Bild 3 zeigt, wie dieses Interface an den Junior angeschlossen wird. Das Relais Re1 kann entfallen. Die LED D4 ist in dieser Anordnung zum BUSY-Indikator umfunktioniert und läuft parallel zur BUSY-Lampe im EPSON.

Die Zeichnungen von Bild 4 und Bild 5 verdeutlichen, wie die Hardware-Modifikationen durchgeführt werden. Nach der Durchführung dieser Modifikationen muß nur noch das EPROM vom Typ 2708 (ESS 515) in den Sockel auf der Standard-Junior-Computer-Platine gesteckt werden. Die beiden EPROMs IC4 und IC5 auf der Junior-Interface-Platine (PM und TM) sind nicht mehr erforderlich, da die Input/Output-Programme für die Druckersteuerung im 2708-EPROM untergebracht sind. Der Adreßraum, der IC4 und IC5 zugeordnet ist, ist jetzt frei für User-Programme. Und worauf Sie noch unbedingt achten müssen: Zwischen den Lötunkten "R" und "S" auf der Junior-Interface-Platine muß eine Drahtbrücke angebracht sein (WITH)!!!

Der Junior-Computer ist jetzt in einen DOS-Computer verwandelt. Stecken Sie jetzt mindestens zwei oder besser drei Dynamic-RAM-Karten auf den Busprint des Junior-Computers. Die Leitungen für die Adreßdekodierung auf den RAM-Karten sind wie folgt zu legen:

RAM-Karte 1: U – 0
V – 1

X – 2
Y – 3

RAM-Karte 2: U – 4
V – 5
X – 6
Y – 7

RAM-Karte 3: U – 8
V – 9
X – A
Y – B

Stecken Sie das Floppy-Disk-Interface noch nicht auf die Busplatine des Juniors. Schließen Sie wie üblich die Spannung am Computer an, und drücken Sie auf die <RST>-Taste des Hex-Keyboards. Das Display des Juniors muß jetzt aufleuchten. Die Kommandotasten <AD>, <DA>, <+> und <GO> haben dieselbe Bedeutung wie früher. Nur die <PC>-Taste wurde umfunktioniert. Ist das Disk-Operating-System einmal in den Computer geladen, dann

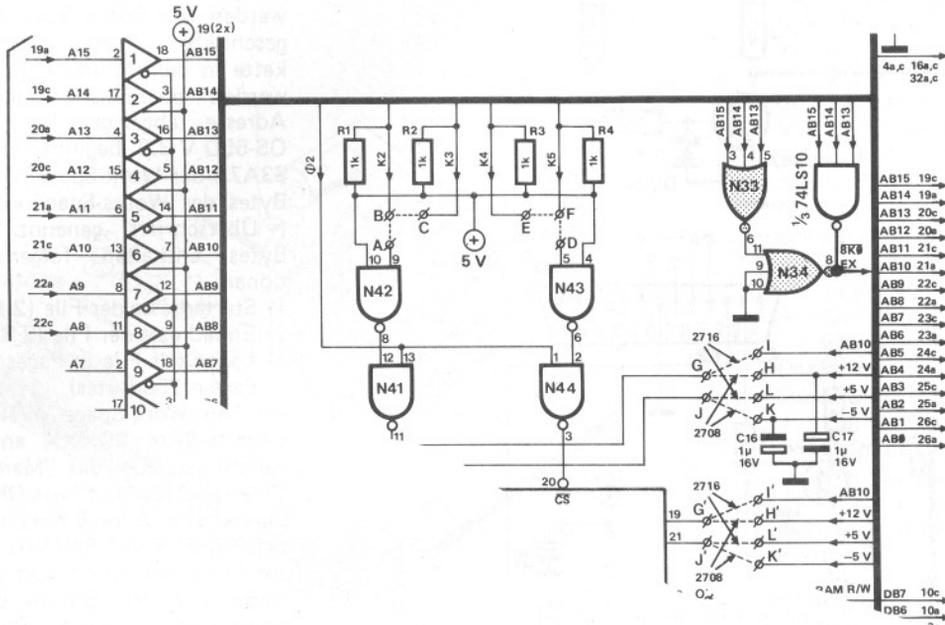


Bild 1. Das Schaltbild der Interface-Karte des Junior-Computers. Die Gatter N33 und N34 werden durch ein NAND-Gatter 74LS10 ersetzt. Damit ergibt sich für die Speicher auf dem Standard-Junior-Computer und auf der Interface-Karte ein neuer Speicherbereich.

kann mit der <PC>-Taste der DOS-Command-Interpreter aufgerufen werden. Doch später mehr darüber.

Bild 2 zeigt die Speicheraufteilung des DOS-Junior-Computers. Diese Speicheraufteilung gilt auch für alle anderen 6502-Computer, die mit dem Floppy-Interface verbunden werden. Dabei können die Adressen \$C000...FBFF von Computer zu Computer verschieden sein. Wichtig ist nur, daß der Computer im unteren Adreßbereich mindestens 32 kByte zusammenhängendes RAM adressieren kann. Sollte der Adreßbereich von FC00...FFFF bei einem Nicht-Junior-Computer bereits belegt sein, dann muß die Bootstrap-Software in einen anderen Speicherbereich gelegt werden. Das dürfte aber mit der "Paperware" nicht schwierig sein.

Die Software des DOS-Juniors

Die Software des DOS-Junior-Computers orientiert sich an den modernsten Maßstäben für Computer-Design. Das heißt, der Computer wird mit einem Minimum an ROM-Intelligenz ausgerüstet und erhält soviel wie möglich RAM. Die Vorteile dieses Computer-Systems liegen auf der Hand:

Da die Systemsoftware, sei es BASIC, FORTH, Assembler oder ein Wordprocessor von der Diskette in kurzer Zeit in den Computer geladen werden können, wird für den ROM-Adreßraum nur wenig Platz verschwendet. Im ROM ist nur soviel Intelligenz vorhanden, daß der Computer das Hexdisplay und das Keyboard bedienen sowie vom/zum Elektterminal empfangen/senden kann. Ein weiterer Teil der ROM-Intelligenz des Juniors besteht darin, Track 0 von der Diskette in den Junior-Computer zu laden. BASIC, Assembler etc. sind also nicht

2

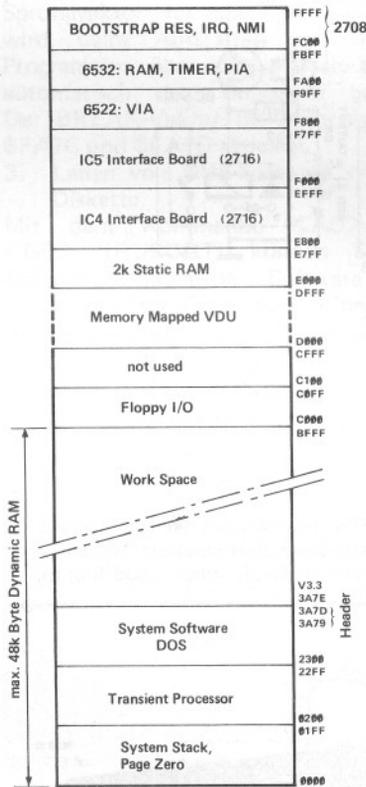


Bild 2. Die neue Speicher-Aufteilung für den DOS-Junior-Computer. Aufgrund dieser Erweiterungen ist der Work-Space bei OS-65D V 3.3 um 4 kByte kürzer. Der Transient-Processor überschreibt den Adreßraum \$2200...22FF der DOS-Software.

mehr im ROM abgelegt, sondern werden von der Floppy in den Computer geladen. Man spricht dann in diesem Fall von "portabler Software". Portable Software hat den entscheidenden Vorteil, daß diese leicht modifizierbar ist. Mußte früher ein Satz neue EPROMs in den Computer gesteckt werden, wenn Softwarefehler entdeckt wurden oder das System den neuesten Gegebenheiten angepaßt werden sollte, so muß bei portabler Software nur eine neue Diskette ins Drive gesteckt werden. Software-Fehler lassen sich somit leicht eliminieren, und das System kann dem modernsten Stand der Technik leicht angepaßt werden. Diese Vorteile nutzen wir beim DOS-Junior voll aus.

Sehen wir uns nochmals Bild 2 an. Page 0 und der System-Stack liegen im Adreßbereich \$0000...01FF. Der Adreßraum \$0200...22FF ist für den "Transient Processor" vorgesehen. Was ist ein Transient Processor? Darunter ist die Software zu verstehen, die das Computer-System arbeitsfähig macht. Arbeiten Sie beispielsweise in BASIC, dann ist der Transient-Processor der BASIC-Interpreter. Oder arbeiten Sie beispielsweise in Maschinensprache, dann ist der Assembler oder der Extended-Monitor von Ohio Scientific der Transient-Processor.

Die System-Software zur Steuerung der Floppy-Drives und zur Steuerung des Druckers sowie eines "Memory Mapped Video Display Unit" nehmen ca. 4 kByte RAM-Speicher ein. Bei der Version OS-65D V 3.1 nimmt die DOS-Software den Speicherraum \$2300...3278 ein. Bei der Adresse \$327E...beginnt der Work-Space oder der Bereich des Speichers, in dem das Programm des Programmierers abgelegt wird. Wenn Sie ein BASIC-Programm in den Computer

3

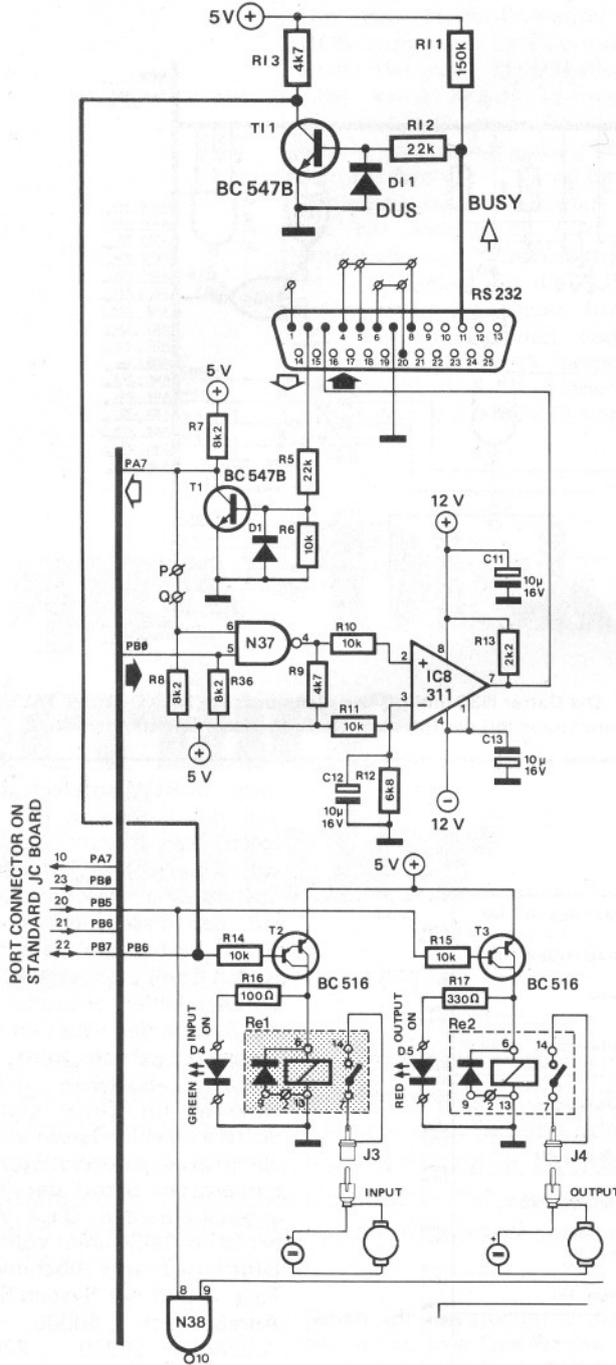


Bild 3. So wird das Interface für den EPSON-Drucker an PB6 angeschlossen. Ist PB6 logisch 1, dann wird der Sende-Ausgang (RS-232-Konnektor Pin 3) des Computers blockiert. Ist kein Drucker angeschlossen, dann sorgt Widerstand R11 dafür, daß der Sende-Ausgang geöffnet ist.

4

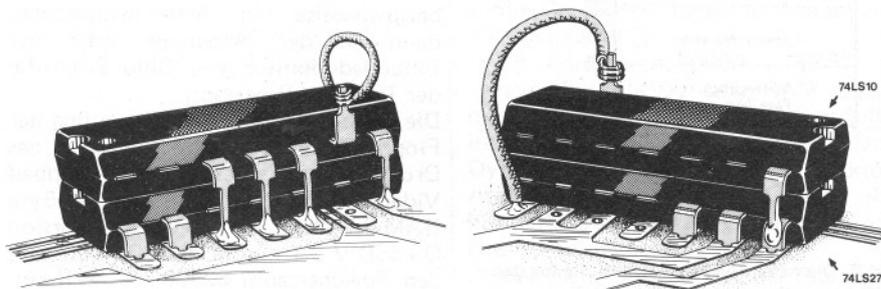


Bild 4. Durch diese Hardware-Modifikation erhält der Junior-Computer eine neue Speicher-Aufteilung. Aus der Zeichnung ist die "Huckepack-Montage" der ICs ersichtlich.

tippen, dann wird dieses ab der Adresse \$327E... abgelegt. Ab dieser Adresse werden die Daten zum Floppy Drive geschrieben. Daten, die von der Diskette in den Computer gelesen werden, werden ebenfalls wieder ab dieser Adresse abgelegt. Bei der Version OS-65D V 3.3 beginnt ab der Adresse \$3A7E der Work-Space. Die ersten fünf Bytes des Works-Space werden Header (= Überschrift) genannt. Diese fünf Bytes enthalten folgende Informationen:

- 1) Startadresse der File (2 Bytes)
- 2) Endadresse der File (2 Bytes)
- 3) Länge der File in Pages (1 Byte, eine Page = 256 Bytes)

An den Work-Space schließt sich die 4-kByte-Seite \$DXXX an. In diesem Adreßraum liegt das "Memory Mapped Video Display Unit" von Ohio Scientific. Eine solche Video-Einheit unterscheidet sich vom ELEKTERMINAL dadurch, daß der Computer direkt über den Datenbus Daten auf den Schirm des Monitors transportieren kann. Der Datenaustausch zwischen Schirm und Computer ist beim "Memory Mapped Video Display Unit" (oder kurz: VDU) viel schneller, als bei einem Video-Interface mit V 24/RS 232-Schnittstelle.

Im Adreßraum \$E000... E7FF liegen 2 kByte statisches RAM. Früher lag dieses RAM im Bereich \$0000... 07FF. Diesen RAM-Bereich verwenden wir in Zukunft für den Object-Code, der vom Assembler erzeugt wird. Sie können jetzt eine Source File mit dem Assembler assemblieren, auf Wunsch ein Listing ausdrucken und den vom Assembler erzeugten Maschinen-Code direkt vom Computer ausführen lassen. Weitere Einzelheiten können Sie dem Assembler-Manual von Ohio Scientific entnehmen.

Im Adreßraum \$E800... FFFF liegen IC4 und IC5 auf der Interface-Karte des Juniors sowie die beiden Interface-Bausteine 6522/6532 und das Bootstrap-EPROM 2708. Die beiden EPROMs IC4 und IC5 auf der Interface-Platine des Juniors stehen jetzt dem Programmierer frei zur Verfügung. Hausgemachte Software oder Programme, die man nicht von der Floppy in den Computer laden möchte, kann auf Dauer in zwei EPROMs (2716) geladen werden.

Das Bootstrap-EPROM (ESS515)

Das Bootstrap-EPROM ist im Adreßbereich \$FC00... FFFF adressiert. Es enthält nur 1 kByte an Software. Die Software im Bootstrap-EPROM läßt sich in acht Untergruppen aufteilen:

1. Hex-Display-Monitor. Dieses Programm hat große Ähnlichkeit mit dem originalen Monitor des Juniors. Die Kommandos <AD>, <DA>, <GO>, <+> haben ihre Bedeutung beibehalten. Nur die <PC>-Taste hat eine andere Funktion. Mit der <PC>-Taste läßt sich der DOS-Command-Interpreter direkt vom Hex-Key-Keyboard aus aufrufen. Der Hex-Display-Monitor wird in erster

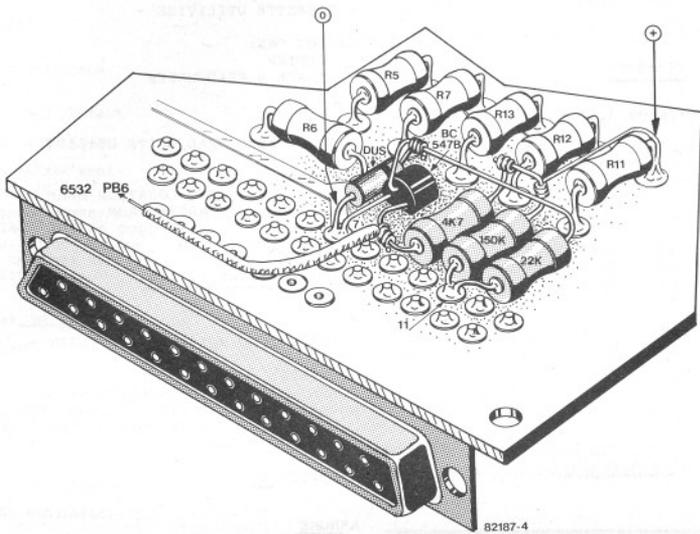


Bild 4. Auf diese Weise wird das EPSON-Interface auf die Interface-Karte des Junior-Computers gesetzt.

Linie dazu verwendet, um die Software auf den Disketten von Ohio Scientific modifizieren zu können. Ohio-Software läßt sich somit in Junior-Software umwandeln.

Mit dem Hex-Display-Monitor lassen sich wichtige Startadressen aufrufen und mit der <GO>-Taste starten.

- RESBAS* \$FF17
- RESDOS* \$FF34
- VONE* \$FFE2
- VTHREE* \$FFE8

2. Laden des BASIC-Interpreters von der Diskette.

Mit dem Kommando <AD> FF17 <GO> (RUBOUT) können Sie BASIC von der Diskette in den Junior-Computer laden. Um Kommandotasten auf dem Hex-Keyboard von den Tasten des ELEKTERMINALS zu unterscheiden, werden wir in Zukunft die Kommandotasten des Hex-Keyboards zwischen <>-Klammern und Kommandotasten auf dem ELEKTERMINAL zwischen ()-Klammern schreiben.

Nachdem BASIC geladen ist und sich der Computer auf dem Terminal "meldet", ist der Transient-Processor auf Programmunterbrechungen vorbereitet. Während des Ausdrucks können Sie mit der (BREAK)-Taste auf dem Terminal ein BASIC-Programm unterbrechen. Wurde die (BREAK)-Taste während eines LIST-Kommandos betätigt, dann drückt der BASIC-Interpreter "Break" aus. Haben Sie einmal mit den RUN-Kommando ein BASIC-Programm gestartet, und wollen Sie den Ausdruck des Programmes stoppen, dann drücken Sie einfach auf die (BREAK)-Taste. Der BASIC-Interpreter antwortet dann "BREAK IN LINE X". Alle Programmvariablen und Pointer werden jetzt auf den Stack gerettet.

Das Programm können Sie nach der Unterbrechung mit dem CONT-Statement wieder starten. Der indirekte Sprungvektor für die (BREAK)-Taste wird beim Aufstarten des BASIC-Programmes über die Adresse \$FF17 automatisch vom Computer gesetzt. Der BREAK-Vektor ist in der Adresse \$FA7C und \$FA7D abgelegt.

3. Laden von DOS-Software von der Diskette.

Mit dem Kommando <AD> FF34 <GO> (RUBOUT) können Sie in Zukunft hauseigene Software von ELEKTOR in ihren Junior-Computer

laden. Die Adresse DOS ist also für zukünftige Erweiterungen und "Nicht-Ohio-Software" gedacht.

4. Anpassung einer Ohio-Diskette OS-65D V 3.1.

Eine Ohio-Scientific-Diskette OS-65D V 3.1 können Sie mit dem Kommando <AD> FFE2 <GO> (RUBOUT) ihrem Junior Computer anpassen. Wenn der Monitor bei der Adresse VONE* \$FFE2 gestartet wird, ereignet sich folgendes:

- Der Computer plaziert den Schreib/ Lesekopf des Drives auf Track 0.
- Der Computer liest die Daten auf Track 0 und legt die gelesenen Daten an der Adresse \$2200... im Speicher ab.
- Der Computer plaziert den Schreib/ Lesekopf des Drives auf Track 1.
- Der Computer liest die Daten auf Track 1 und legt die gelesenen Daten an der Adresse \$2A00... im Speicher ab. Das gesamte Disk-Operating-System ist jetzt in den Computer geladen und kann vom Programmierer mit dem Hex-Keyboard modifiziert werden.

Nachdem Track 0 und Track 1 in den Computer geladen sind, erfolgt ein Sprung zum Hex-Display-Monitor, und der Computer meldet sich mit *Track 0 & 1* auf dem Drucker zurück.

5. Anpassung einer Ohio-Diskette OS-65D V 3.3-Tutorial-Disk 5.

Eine Ohio-Scientific-Diskette OS-65D V 3.3-Tutorial-Disk 5 können Sie mit dem Kommando <AD> FFE8 <GO> (RUBOUT) ihrem Junior Computer anpassen. Für Track 0 und Track 1 gilt dasselbe wie bei 4. Nachdem aber Track 0 und Track 1 in den Computer geladen sind, ereignet sich folgendes:

- Der Computer bewegt den Schreib/ Lesekopf des Drives auf Track 6. Die Daten auf Track 6, Sektor 2 werden

6

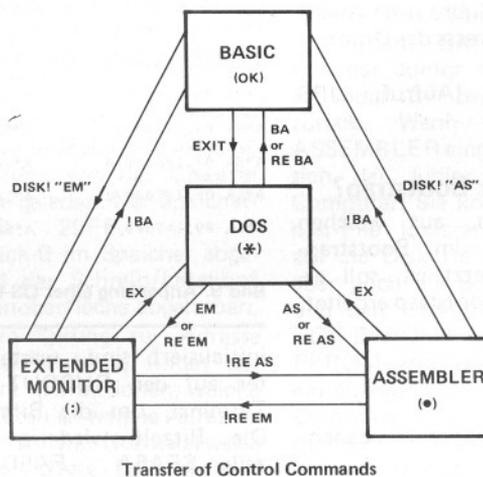


Bild 6. Der Kommando-Transfer zu den unterschiedlichen Transient-Processoren: von BASIC nach ASSEMBLER, vom ASSEMBLER nach EXTENDED MONITOR usw. und umgekehrt. Die "Schaltzentrale" ist das DOS. Mit ihm lassen sich die Transient-Processoren BASIC, ASSEMBLER und EXTENDED MONITOR von der Diskette in den Computer laden.

10

```

RUN"BXEXEC*"

BASIC EXECUTIVE FOR
OS-65D V3.1

JUNE 25, 1980 RELEASE

FUNCTIONS AVAILABLE:

CHANGE- ALTER WORK-
SPACE LIMITS

DIR- PRINT DIRECTORY

UNLOCK- UNLOCK SYSTEM
FOR END USER MODI-
FICATIONS
    
```

FUNCTION? UNLOCK

SYSTEM OPEN

OK

RUN

```

BASIC EXECUTIVE FOR
OS-65D V3.1

JUNE 25, 1980 RELEASE

FUNCTIONS AVAILABLE:

CHANGE- ALTER WORK-
SPACE LIMITS

DIR- PRINT DIRECTORY

UNLOCK- UNLOCK SYSTEM
FOR END USER MODI-
FICATIONS
    
```

FUNCTION? DIR

LIST ON LINEPRINTER INSTEAD OF DEVICE # 1 ? NO

OS-65D VERSION 3.0
-- DIRECTORY --

FILE NAME	TRACK	RANGE
OS65D3	0	- 12
BXEXEC*	14	- 14
CHANGE	15	- 16
CREATE	17	- 19
DELETE	20	- 20
DIR	21	- 21
DIRSRT	22	- 22
RANLST	23	- 24
RENAME	25	- 25
SECDIR	26	- 26
SQLST	27	- 28
TRACE	29	- 29
ZERO	30	- 31
ASAMPL	32	- 32

50 ENTRIES FREE OUT OF 64

OK

11

OK
LIST

```

10 REM DIRECTORY UTILITY FOR OS-65D VERSION 3.0
20 REM
30 NF=0
40 PN=11897
50 DEF FNA(X)=10*INT(X/16)+X-16*INT(X/16)
BREAK
OK
    
```

NEW

OK

RUN"DIR"

LIST ON LINEPRINTER INSTEAD OF DEVICE # 1 ? NO

OS-65D VERSION 3.0
-- DIRECTORY --

FILE NAME	TRACK	RANGE
OS65D3	0	- 12
BXEXEC*	14	- 14

BREAK IN 11110

OK

CONT

CHANGE	15	- 16
CREATE	17	- 19
DELETE	20	- 20
DIR	21	- 21

BREAK IN 11100

OK

Bild 11. Überprüfung der (BREAK)-Taste beim LIST-Kommando und beim RUN-Statement.

12

A*CA 0200=13,1

A*GO 0200

- DISKETTE UTILITIES -

SELECT ONE:

- 1) COPIER
- 2) TRACK 0 READ/WRITE
- ? 1

- DISKETTE COPIER -

FROM DRIVE (A/B/C/D)? A

TO DRIVE (A/B/C/D)? B

STARTING TRACK? 2

ENDING TRACK (INCLUSIVE)? 32

READY (Y/N)? Y

ANOTHER (Y/N)? N

A*

Bild 12. Stehen zwei Floppy-Disk-Laufwerke zur Verfügung (Drive A und Drive B), dann läßt sich eine OS-65D-V3.1-Diskette sehr einfach anpassen. Die Kopierarbeit erledigt der Computer automatisch.

dann wird der BASIC-Interpreter von der Diskette in den Speicher geladen. Der BASIC-Interpreter ist bei OS-65D V 3.1 auf den Tracks 2... 4 untergebracht. Bei OS-65D V 3.3 sind der BASIC-Interpreter und diverse Erweiterungen des Editors auf den Tracks 2... 4, Track 6 und Track 13 untergebracht. Nachdem der Interpreter geladen ist, erfolgt ein Sprung zum Cold-Start-Entry des BASIC-Interpreters (\$20E4). Das System meldet sich jetzt mit dem Prompt "OK" zurück.

— Das System ist jetzt noch nicht offen für die Statements LIST, CONT usw. Auch eine BASIC-File kann noch nicht angefertigt werden. Das einzige BASIC-Statement, das der Computer kennt, ist RUN. Wenn Sie eine BASIC-File erzeugen möchten, dann laden Sie das Programm BXEXEC* in den Computer mit dem Kommando: RUN "BXEXEC*"

Der Computer stellt sich dann mit einem Menü von Optionen vor. Wählen Sie bei Version OS-65D V 3.1 die Option "UNLOCK" und bei OS-65D V 3.3 die Option "9". Der Computer meldet sich mit dem Prompt "SYSTEM OPEN". Geben Sie jetzt ein "NEW"-Statement, und der Work-Space des Juniors ist formatiert für Ihre BASIC-File.

Das DOS und seine Möglichkeiten

Mit dem Disk-Operating-System von Ohio, das auf Track 0 und Track 1 der Diskette untergebracht ist, läßt sich der Junior-Computer leicht in diverse Transient-Processors umwandeln. Bild 6 zeigt, wie die einzelnen Verbindungen vom DOS zu den einzelnen Processoren hergestellt und wieder aufgehoben werden können. Wenn Sie beispielsweise das DOS an der Adresse \$FF17 starten, verwandelt sich der Junior in einen BASIC-Computer. Wollen Sie den Junior in einen Assembler-Computer verwandeln, dann müssen Sie EXIT eingeben, um den BASIC-Interpreter zu verlassen. Nach dem EXIT-Kommando meldet sich der Junior mit dem DOS Prompt A* oder B* usw. auf dem Drucker zurück. Wenn Sie jetzt As oder ASSEMBLER eingeben, dann verwandelt sich der Junior in einen Assembler-Computer. Sie können jetzt eine Assembler-File kreieren und mit dem DOS auf die Diskette schreiben. Sie können aber auch die Source-File assemblieren und den Object-Kode mit dem ELEKTOR-EPROM-Programmer direkt ins EPROM programmieren, ohne daß Sie ein einziges Byte eintippen müssen. Der Computer erledigt diese Aufgabe automatisch! Nähere Angaben über das Arbeiten mit dem Assembler entnehmen Sie dem Assembler-Manual von Ohio Scientific. Achtung! Der Assembler und der erweiterte Monitor sind bei der Version OS-65D V 3.1 nicht auf der Diskette. Nur die Version OS-65D V 3.3 hat standardmäßig den Assembler und den erweiterten Monitor (EM) auf der Diskette.

— Ist der Index-Impuls abgeklungen, dann wird das Control-Register im ACIA (6850) gesetzt. Das Übertragungsformat des ACIAs ist: ein Startbit, acht Datenbits, ein Even-Parity-Bit und ein Stoppbit.

— Der Computer liest das erste Byte von der Diskette. Dieses Byte ist die hochwertige Startadresse des Speicherbereichs, in dem Track 0 abgelegt wird (= \$22). Das zweite Byte von der Diskette ist das niederwertige Byte der Startadresse (= \$00). Beide Bytes werden in den Bump-Pointer (= Ladezeiger) geladen. Der Bump-Pointer zeigt somit auf die Adresse \$2200. Nun liest der Computer das dritte Byte von der Diskette. Dieses Byte gibt darüber Auskunft, wieviele Pages Track 0 lang ist (= \$08).

— Die folgenden Bytes sind zwei

kByte Software in Maschinensprache. Diese Bytes werden von der Diskette in den Computer geladen. Der Speicherbereich ist \$2200... 29FF.

— Nachdem Track 0 im Speicher abgelegt ist, wird der Schreib/Lesekopf von der Diskettenoberfläche abgehoben, und es folgt ein Sprung zur Adresse \$2200. Ab dieser Adresse findet der Computer weitere Instruktionen, welche Tracks und Sektoren an welche Adressen geladen werden sollen. Normalerweise werden die zwei kByte Software auf Track 1 in den Adreßbereich \$2A00... 31FF geladen. Track 0 und Track 1 enthalten zusammen 4 kByte Software in Maschinensprache, die das gesamte Disk-Operating-System arbeitsfähig macht.

— Ist das Disk-Operating-System (= DOS) in den Computer geladen,

13

Bild 13. Anpassung einer OS-65D-V3.1-Diskette. Über einen Sprung in den DOS-Command-Interpreter (GO 2276) wird BASIC geladen und mit dem Utility-Programm BEXEC* kopiert. Schließlich führt man einen "Probelauf" durch, um sicher zu sein, daß die Diskette richtig angepaßt wurde.

<RST>
<AD> FFEB
<GO>
(RUBOUT)
TRACK 0&1
<PC>
A*GO 2276
OS-65D V3.0
OSI 9 DIGIT BASIC
COPYRIGHT 1977 BY MICROSOFT
33921 BYTES FREE
Ok
RUN"BEXEC*"
OS-65D Tutorial disk five - Sept. 16, 1981
1 > Directory
2 > Create a new file
3 > Change a file name
4 > Delete file from diskette
5 > Create blank data diskette
6 > Create data diskette with files
7 > Create buffer space for data files
8 > Single or dual disk drive copier
9 > Enter OS-65D system
Type the number of your selection and depress RETURN ? 8
- Diskette copier -
Copy from which drive (A/B/C/D) ? A
Copy to which drive (A/B/C/D) ? A
What is the last track to be copied (Inclusive) <0-39> ? 39
Are you ready to start copying (Y/N) ? Y
Insert master diskette -- press <RETURN> ?
Reading --
Insert blank diskette -- press <RETURN> ?
Initializing --
Track 01 - 01/08
Track 02 - 01/08
Track 03 - 01/08
Track 04 - 01/08
Track 05 - 01/08
Track 06 - 01/01 - 02/01 - 03/01 - 04/02
Track 07 - 01/08
Track 08 - 01/08
Track 09 - 01/08
Track 10 - 01/08
Track 11 - 01/01 - 02/01 - 03/01 - 04/01 - 05/01 - 06/01 - 07/01
Track 12 - 01/01 - 02/01 - 03/01 - 04/01
Track 13 - 01/08
Track 14 - 01/08
Insert master diskette -- press <RETURN> ?
Reading --
Insert blank diskette -- press <RETURN> ?
Track 15 - 01/08
Track 16 - 01/08
Track 17 - 01/08
Track 18 - 01/08
Track 19 - 01/08
Track 20 - 01/08
Track 21 - 01/08
Track 22 - 01/08
Track 23 - 01/08
Track 24 - 01/08
Track 25 - 01/08
Track 26 - 01/08
Track 27 - 01/08
Track 28 - 01/08
Insert master diskette -- press <RETURN> ?
Reading --
Insert blank diskette -- press <RETURN> ?
Track 29 - 01/08
Track 30 - 01/08
Track 31 - 01/08
Track 32 - 01/08
Track 33 - 01/08
Track 34 - 01/08
Track 35 - 01/08
Track 36 - 01/08
Track 37 - 01/08
Track 38 - 01/08
Track 39 - 01/05 - 02/02
Insert master diskette -- press <RETURN> ?
Please, put the tutorial disk in drive A and depress <RETURN>.

TRACK 0&1
<AD> 2217
<DA> 4C
<+> 40
<+> 22
<AD> 2245
<DA> 4C
<+> 76
<+> 22
<AD> 2285
<DA> 8E
<+> C6
<+> 2A
<+> 4C
<+> B3
<+> 22
<AD> 2EB4
<DA> 4C
<+> B0
<+> 2E
<PC>
A*CA 0200=06,4
A*GO 0200
- TRACK ZERO READ/WRITE UTILITY -
COMMANDS:
Rnnnn - READ INTO LOCATION nnnn.
Wnnnn/gggg,p - WRITE FROM nnnn FOR p PAGES
WITH gggg AS THE LOAD VECTOR
E - EXIT TO OS-65D
COMMAND? W2200/2200,8
- TRACK ZERO READ/WRITE UTILITY -
COMMANDS:
Rnnnn - READ INTO LOCATION nnnn.
Wnnnn/gggg,p - WRITE FROM nnnn FOR p PAGES
WITH gggg AS THE LOAD VECTOR
E - EXIT TO OS-65D
COMMAND? E
A*SA 01,1=2A00/8
A*
<RST>
<AD> FF17
<GO>
(RUBOUT)
OS-65D V3.0
OSI 9 DIGIT BASIC
COPYRIGHT 1977 BY MICROSOFT
34177 BYTES FREE
Ok
RUN"BEXEC*"
OS-65D Tutorial disk five - Sept. 16, 1981
1 > Directory
2 > Create a new file
3 > Change a file name
4 > Delete file from diskette
5 > Create blank data diskette
6 > Create data diskette with files
7 > Create buffer space for data files
8 > Single or dual disk drive copier
9 > Enter OS-65D system
Type the number of your selection and depress RETURN ? 1
Directory utility
Directory of which drive ?
Type A,B,C or D and depress RETURN <A> ?
Do you want to list the directory to the printer (Yes or No) <No> ?
-- Directory --
File name Track range

OS65D3 0 - 13
BEXEC* 14 - 16
COPIER 17 - 18
CHANGE 19 - 20
CREATE 21 - 22
DELETE 23 - 23
DIR 24 - 24
RANLST 25 - 26
RENAME 27 - 27
SECDIR 28 - 28
SEQLST 29 - 30
TRACE 31 - 31
ZERO 32 - 33
ASAMPL 34 - 34
ATNENB 35 - 35
COLORS 36 - 36
MODEM 37 - 38
COMPAR 39 - 39
46 Entries free out of 64
Depress RETURN to continue ?

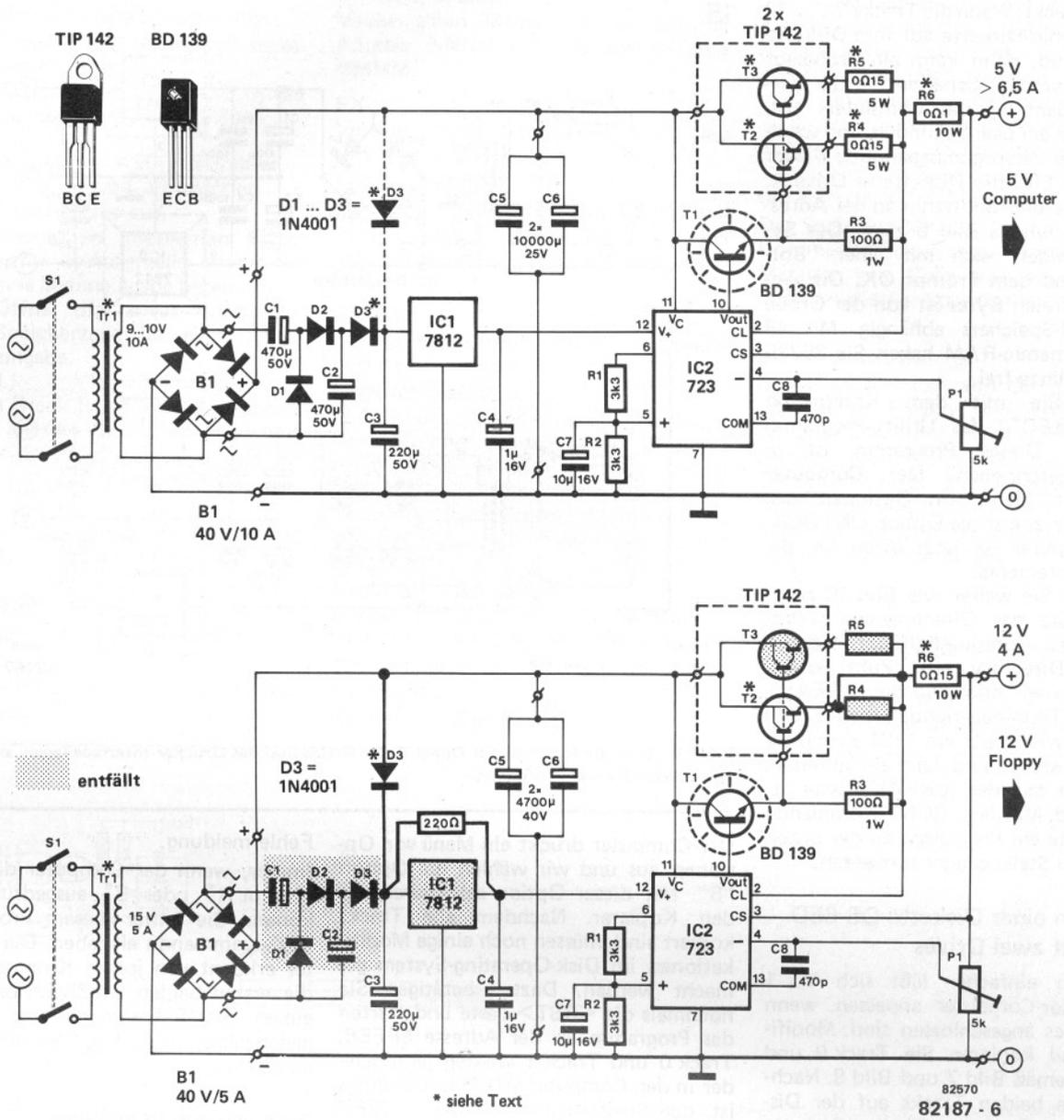


Bild 14. Diese beiden "Kraftpakete" versorgen den DOS-Junior-Computer zuverlässig mit genügend Strom. Man kann beide auf Elektor-Platinen aufbauen. Sie "reichen" für einen vollständigen Junior-Computer inklusive vier Laufwerke.

Anpassen einer Diskette OS-65D V 3.1 mit einem Drive

Wenn Sie nur ein Drive haben, läßt sich V 3.1 nur mit großer und umständlicher Tipparbeit dem Junior-Computer anpassen. Die Bilder 7, 8, 9 zeigen, wie's gemacht wird. Zunächst starten wir mit dem Hex-Keyboard das System. An der Adresse \$FFE2 beginnt ein Programm, mit dem sich eine V 3.1-Diskette "modifizierfreundlich" in den Junior laden läßt. Nach dem Drücken der (RUBOUT)-Taste lädt der Computer Track 0 und Track 1 in den Speicher und meldet sich mit *Track 0 & 1*. Das Display leuchtet auf, die Vektoren für die Input- und Output-Routinen wurden automatisch geladen, und der Programmierer kann jetzt an den Adressen \$2217, \$2245 und \$2283 einige Bytes ändern. Nachdem die Bytes gemäß Bild 7 geändert sind, erfolgt ein Sprung zum DOS-Command-Interpreter

mit der <PC>-Taste. Das DOS meldet sich mit dem Prompt A*. Das Kommando CA 0200 = 13,1 lädt von der Diskette Track 13, Sektor 1. Die Daten werden ab der Adresse 0200 im Computer abgelegt. Das Programm, das wir soeben in den Junior geladen haben, ist ein Disketten-Kopierer und das Track-0-Read/Write-Utility. Mit dem Track-0-Read/Write-Utility-Programm können Sie das modifizierte DOS auf Track 0 Ihrer Diskette schreiben. Doch zuvor müssen Sie noch Ihre Diskette initialisieren. Nehmen Sie deshalb die Ohio-Diskette aus dem Drive und stecken Sie Ihre Diskette hinein. Schließen Sie die Türe des Drives. Bild 8 zeigt, wie's weiter geht: Tippen Sie IN. Dieses Kommando initialisiert die Diskette. Das System fragt: "ARE YOU SURE?" Ihre Antwort ist Y(ES). Das laute Klickgeräusch des Drives verstummt erst wieder, nachdem auf alle 39 Tracks der Diskette die Formatierungsinforma-

tion geschrieben ist. Nach der Initialisierung ist die Diskette für Read/Write-Operationen vorbereitet. Verfahren Sie jetzt wie in Bild 8 gezeigt ist. Zunächst schreiben Sie mit dem Kommando W 2200/2200,8 acht Pages Software auf Track 0. Die Startadresse ist \$2200, und der Ladevektor für den Bootstrap ist ebenfalls \$2200. Die erste Hälfte des DOS ist somit auf die Diskette geschrieben. Das Kommando SA 01,1 = 2A00/8 rettet auf Track 1, Sektor 1 einen Datenblock mit einer Länge von acht Pages, beginnend bei der Startadresse \$2A00. Mit diesem Kommando retten sie den zweiten Teil des DOS auf die Diskette. Da mit dem Kopierer von V3.1 nicht von Drive A zu Drive A kopiert werden kann, muß Track für Track von Hand kopiert werden. Bild 9 zeigt, wie's gemacht wird. Dabei ist zu beachten, daß bei den "CA"-Kommandos die Ohio-Diskette im Drive A ist, während bei

den "SA"-Kommandos Ihre Diskette im Drive A steckt. Wenn die Tracks 2...32 von der Ohio-Diskette auf Ihre Diskette kopiert sind, dann kann ein Probelauf gestartet werden. Schalten Sie die Versorgungsspannung des Computers aus. Warten Sie ein paar Sekunden, und schalten Sie die Versorgungsspannung wieder an. Legen Sie Ihre Diskette in Drive A. Starten Sie den Bootstrap an der Adresse \$FF17, wie es Bild 9 zeigt. Das System "meldet" sich mit einer "Botschaft" und dem Prompt OK. Die Anzahl der freien Bytes ist von der Größe des RAM-Speichers abhängig. Mit 48 kByte Dynamic-RAM haben Sie 36225 Speicherplätze frei.

Starten Sie mit dem Kommando RUN"BEXEC*" das Utility-Programm BEXEC*. Dieses Programm ist in BASIC geschrieben. Der Computer druckt ein Menü von Optionen aus. Wählen Sie zuerst die Option UNLOCK. Der Computer ist jetzt offen für alle BASIC-Statements.

Verfahren Sie weiter wie Bild 10 zeigt. Starten Sie das Directory-Utility-Programm. Der Computer druckt jetzt das gesamte Directory aus. Zum Schluß überprüfen wir noch, ob die (BREAK)-Taste des Terminals richtig arbeitet. Bild 11 zeigt, wie zuerst ein LIST-Kommando unterbrochen und dann ein laufendes Programm mit der (BREAK)-Taste beendet wird. Mit dem CONT-Kommando können Sie ein Programm an der unterbrochenen Stelle wieder fortsetzen.

Anpassen einer Diskette OS-65D V3.1 mit zwei Drives

Wesentlich einfacher läßt sich V3.1 dem Junior-Computer anpassen, wenn zwei Drives angeschlossen sind. Modifizieren und kopieren Sie Track 0 und Track 1 gemäß Bild 7 und Bild 8. Nachdem diese beiden Tracks auf der Diskette sind, entnehmen Sie Ihre Diskette dem Drive A und stecken Sie diese in Drive B. Stecken Sie jetzt die Ohio-Diskette in Drive A. Verfahren Sie gemäß Bild 12. Der Computer kopiert jetzt automatisch alle Tracks, beginnend bei Track 2 und endend bei Track 32 von der Ohio-Diskette in Drive A auf Ihrer Diskette in Drive B. Nach dieser Prozedur ist Ihre Diskette mit der Ohio-Software dem Junior-Computer angepaßt.

Anpassen einer Diskette OS-65D V3.3

Die Diskette V3.3 läßt sich wesentlich einfacher anpassen als die Diskette V3.1. Es wird nur ein Drive benötigt. Bild 13 zeigt, wie dabei vorzugehen ist. Zuerst laden wir wieder Track 0 und Track 1 in den Computer. Da das DOS im Speicher ist, laden wir mit dem Kommando GO 2276 den BASIC-Interpreter. Der Prompt OK zeigt, daß der Interpreter geladen ist. Jetzt laden wir das Utility-Programm BEXEC* mit dem Kommando: RUN"BEXEC*"

15

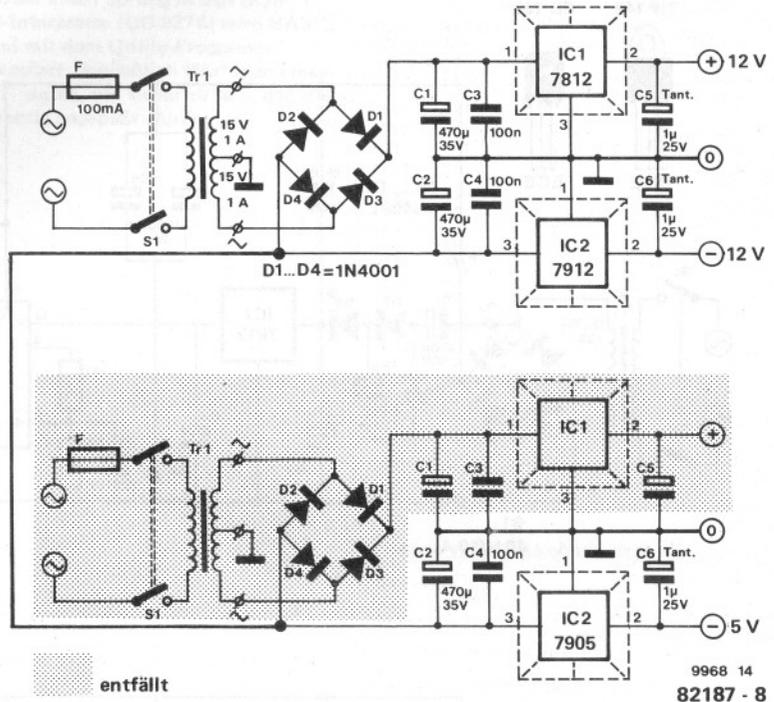


Bild 15. Auch die Netzteile für Dynamische-RAMs und das Drucker-Interface lassen sich leicht mit Elektor-Platinen aufbauen.

Der Computer druckt ein Menü von Optionen aus und wir wählen die Option "8". Mit dieser Option selektieren wir den Kopierer. Nachdem alle Tracks kopiert sind, müssen noch einige Modifikationen im Disk-Operating-System gemacht werden. Dazu betätigen Sie nochmals die <RST>-Taste und starten das Programm an der Adresse \$FFE8. Track 0 und Track 1 werden jetzt wieder in den Computer geladen. Allerdings ist der Speicherraum \$2200...22FF nicht mehr vom BASIC-Interpreter überschrieben. An den Adressen \$2217, \$2245, \$2285 und \$2E84 nehmen Sie gemäß Bild 13 die Modifikationen vor. Nachdem Sie diese Modifikationen durchgeführt haben, laden Sie von Track 6, Sektor 4 das Track-0-Read/Write-Programm. Mit diesem Programm schreiben Sie wieder das modifizierte DOS auf Track 0 der Diskette. Das Kommando SA 01, 1 = 2A00/8 rettet acht Pages DOS-Software auf Track 1. Sie haben jetzt eine V3.3-Diskette, die dem Junior Computer angepaßt ist. An der Adresse \$FF17 können Sie BASIC starten. Das DOS und der BASIC-Interpreter werden automatisch nach dem Betätigen der (RUBOUT)-Taste in den Junior-Computer geladen (siehe Bild 13).

Der DOS-Command-Interpreter

Wie bereits erwähnt, hat das Disk-Operating-System einen eigenen Command-Interpreter. Die wichtigsten Kommandos wollen wir an dieser Stelle besprechen. Wird ein Kommando nicht richtig eingegeben, dann erfolgt eine

Fehlermeldung.

Immer, wenn der Computer den DOS-Prompt A* oder B* ausgedruckt hat, können Sie hinter diesem Prompt ein DOS-Kommando eingeben. Der Computer erkennt von jedem Kommando nur die ersten beiden Großbuchstaben. Bei einem SAVE-Kommando ist es also gleichgültig, ob Sie SA, SAV oder SAVE eingeben.

Das Kommando AS oder ASM

Der Computer lädt von der momentan selektierten Floppy den Assembler und den Extended-Monitor. Nachdem dieses Programm in Maschinensprache geladen ist, erfolgt ein Sprung zum Cold-Start-Entry des Assemblers. Der Assembler ist mit einem zeilenorientierten Editor verbunden.

Das Kommando EM

Der Computer lädt von der momentan selektierten Floppy den Assembler und den Extended-Monitor. Nachdem dieses Programm in Maschinensprache geladen ist, erfolgt ein Sprung zum Extended-Monitor. Der Extended-Monitor ist ein Programm, mit dessen Hilfe sich ein Programm in Maschinensprache leicht überprüfen läßt. Der Extended-Monitor hat seinen eigenen Command-Interpreter. Die wichtigsten Kommandos sind:

- !STRING
- Sende "STRING" zum DOS-Command-Interpreter als ein Kommando.
- aNNNN
- Öffne die Speicherzelle mit der Adresse NNNN für die Ausführung der folgenden

Unterkommandos:

- (LF) Öffne nächste Speicherzelle.
- (CR) Schließe die momentan adressierte Speicherzelle.
- (D)(D) Schreibe die Daten DD in die momentan adressierte Speicherzelle.
- (") Drucke den ASCII-Charakter der momentan adressierten Speicherzelle aus.
- (/) Mache die momentan adressierte Speicherzelle erneut für eine Dateneingabe bereit.
- (*) Öffne die zuvor adressierte Speicherzelle für eine Dateneingabe.

BN, LLLL

Setze den Break Point mit der Nummer N an die Adresse LLLL. Der Zahlenbereich der Break-Point-Nummer N ist 1...8.

EN
Eliminiere den Break-Point mit der Nummer N.

A
Drucke den Accuinhalt aus, der sich beim letzten Break-Point ergeben hat.

C
Starte das Programm nach dem letzten Break-Point.

DNNNN, MMMM
Mache einen Memory-Dump von der Adresse NNNN bis exklusive Adresse MMMM.

EX
Verlasse den Extended-Monitor und springe zum DOS.

FNNNN, MMMM = DD
Fülle den Speicherbereich, beginnend bei der Adresse MMMM und endend bei der Adresse MMMM-1, mit den Daten DD.

GNNNN
Springe zur Adresse NNNN und arbeite dort das Programm ab.

HNNNN, MMMM(OP)
Aufruf des hexadezimalen Rechners. Der Rechner druckt das Resultat NNNN (OP) MMMM aus, wobei (OP) = +, -, *, / sein kann. Es lassen sich Addition, Subtraktion, Multiplikation und Division von hexadezimalen Zahlen durchführen.

MNNNN = MMMM, LLLL
Bewege den Speicherbereich MMMM... NNNN-1 in den Speicherbereich, der bei der Adresse NNNN beginnt (Move Kommando).

RMMM = NNNN, LLLL

Der Extended-Monitor hat einen sogenannten Relokator. Maschinenprogramme lassen sich somit in einen anderen Speicherbereich bewegen. Der Computer übernimmt die Korrektur aller absoluten Adressen: "Relokiere" den Speicherbereich zwischen NNNN... LLLL-1 in den Speicherbereich, der bei der Adresse MMMM beginnt.

Der Extended-Monitor hat noch weitere Kommandos, doch diese können Sie den Manuals von Ohio entnehmen (leider nur in englischer Sprache).

Das Kommando BA

Der Computer lädt von der momentan selektierten Floppy den BASIC-Interpreter. Nachdem der BASIC-Interpreter geladen ist, erfolgt ein Sprung zum Cold-Start-Entry. Der Interpreter gibt darüber Auskunft, wie viele Speicherplätze im System frei sind und meldet sich mit dem Prompt OK.

Das Kommando CA NNNN = TT,S oder CALL NNN = TT,S

Lade die Daten auf Track TT, Sektor S von der Diskette in den Computer. Lege

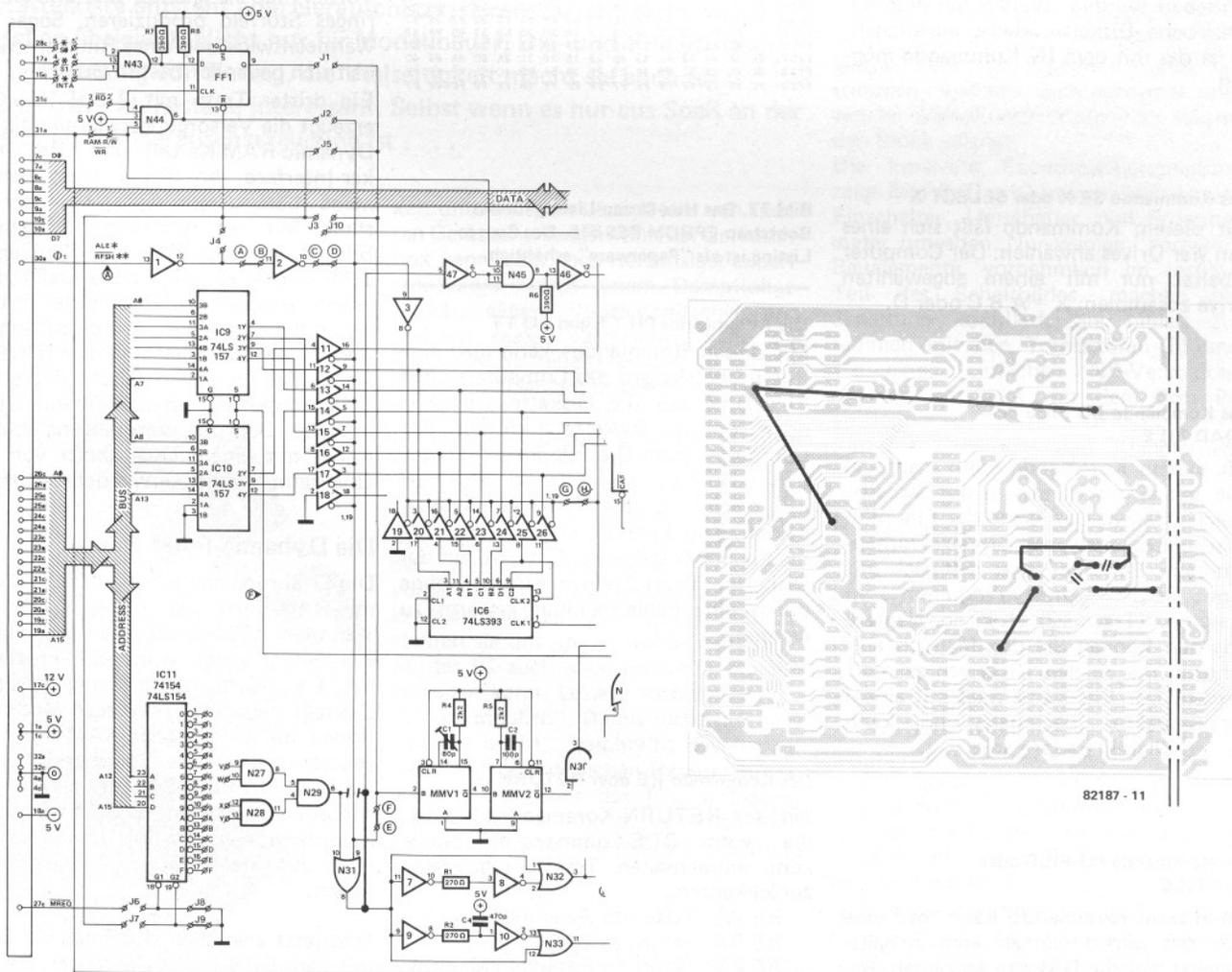


Bild 16. Die Modifikationen tragen entscheidend zur richtigen Funktion der Dynamic-RAM-Karte bei.

die Daten im Speicherbereich NNNN... ab. Der Zahlenbereich für TT ist 1...39 und der Zahlenbereich für S ist 1...8.

Das Kommando SA TT,S = NNNN/8PP oder SAVE TT,S = NNNN/P

Rette die Daten im Speicherbereich NNNN... für P Pages auf die Diskette. Die Track-Nummer ist TT, die Sektor-Nummer ist S und die Anzahl der Pages, die auf die Diskette geschrieben werden ist P. Der Zahlenbereich für TT ist 01...39, der von S ist 1...8 und der von P ist ebenfalls 1...8.

Das Kommando DI TT oder DIR TT

Mit diesem Kommando läßt sich überprüfen, wieviele Sektoren auf Track TT sind. Der Zahlenbereich von TT ist 01...39.

Das Kommando IN oder INIT

Das Kommando IN initialisiert eine neue Diskette, auf die noch keine Daten geschrieben wurden. Wollen Sie eine beschriebene Diskette wieder auslöschen, so ist das mit dem IN Kommando möglich.

Das Kommando SE X oder SELECT X

Mit diesem Kommando läßt sich eines von vier Drives anwählen. Der Computer arbeitet nur mit einem angewählten Drive zusammen. X = A,B,C oder D.

Das Kommando LO FILE oder LOAD FILE

Mit diesem Kommando kann man eine File mit einem Namen in den Arbeitsspeicher laden. Der Name der File muß aber im Directory der Diskette vorhanden sein. Ein File-Name läßt sich mit dem CREATE-Utility-Programm im Directory (Track 12) erzeugen. Nähere Angaben können Sie den Manuals entnehmen, die zu den Disketten geliefert werden. Der File-Name muß mit einem Alfa-Charakter beginnen und kann 1...6 Charaktere lang sein.

Das Kommando PU FILE oder PUT FILE

Mit diesem Kommando kann man eine File mit einem Namen vom Arbeitsspeicher auf die Diskette schreiben. Bevor die File auf die Diskette geschrieben werden kann, muß der File-Name im Directory vorhanden sein.

17

Hex-Dump-Listing for Bootstrap-EPROM ESS515. Header: 0 1 2 3 4 5 6 7 8 9 A B C D E F. Content: FC00: A9 1E BD 83 FA A9 01 85 FD A2 FF 9A 78 D8 20 7C FC10: FC D8 2B 7C FC F8 28 7C FC F8 28 7C FC F8 28 7C FC20: C9 13 D8 07 A2 FF 9A D8 6C FA 00 C9 10 D0 06 A9 FC30: 01 85 FD D0 14 C9 11 D0 06 A9 00 85 FD F0 0A C9 FC40: 12 D0 09 FE E6 DA D0 02 E6 FB 4C 0D FC C9 14 D0 03 FC50: 4C D0 FE C9 15 10 F2 85 FF A4 FD D0 0D B1 FA 0A FC60: 0A 0A 0A 05 FF 91 FA 4C 49 FC A2 04 06 FA 26 FB FC70: CA D8 F9 A5 FA 05 FF 85 FA 4C 49 FC A8 00 B1 FA FC80: 85 F9 A9 7F 8D 01 FA A2 08 A5 FB 2B 08 FC A5 FA FC90: 2D 08 FC A5 F9 2D 08 FC A9 00 8D B1 FA A0 03 A2 FCA0: 00 A9 FF 8E 82 FA E8 E8 2D 80 FA 88 D8 F5 A9 06 FCB0: 8C 82 FA 09 80 49 FF 60 48 84 FC 4A 4A 4A 4A 20 FCC0: CB FC 68 29 0F 20 CB FC A4 FC 68 A8 B9 08 FD 8D FCD0: 08 FA 82 7A A0 FF 88 D8 D8 06 8C 82 FA 28 E8 FCE0: 68 A2 21 A0 81 20 A1 FC D8 07 0E 27 D8 F5 A9 15 FCF0: 68 A0 FF 0A 80 03 C8 10 FA 8A 29 0F 4A AA 98 10 FDD0: 03 18 69 07 CA D0 FA 60 48 79 24 30 19 12 02 78 FDE0: 00 18 08 03 46 21 06 0E 28 1E FD 6C FD 08 28 28 FDF0: FD 20 CF FD 28 2F FD 68 A0 08 8C 01 C9 A9 5D FDE0: 18 C8 24 C3 FD 85 FE A4 28 C5 FD 85 FD 28 C5 FD FDA0: C8 8C 03 C0 88 02 C8 0E 00 8C 0C 8C 02 C8 68 A9 FDB0: FD 08 89 A9 82 C0 08 FC 1C A9 FF 8D 02 C8 28 FDC0: CE FD 29 F7 8D 02 C8 2C 0E FD 09 08 8D 02 C8 A2 FDD0: 18 20 BA FD F0 DD A2 7F 8E 02 C8 20 D7 FC AD 00 FDE0: C8 38 FD AD 08 C8 18 FB A9 03 8D 18 C8 A9 58 8D FDF0: 18 C8 24 C3 FD 85 FE A4 28 C5 FD 85 FD 28 C5 FD FDA0: 85 FF A8 00 28 C5 FD 91 FD C8 D8 F8 E6 FE C6 FF FDB0: D8 F2 86 FE A9 FF 8D 02 C8 68 A8 F8 88 D8 FD 55 FDC0: FF CA D8 F6 68 AD 10 C8 4A 90 FA AD 11 C8 68 D8 FDD0: 78 A9 67 8D 82 FA A9 08 8D 88 FA A2 FC 8E 5A FA FDE0: A2 FF 8E 5B FA EA A9 7F 8D 81 FA 4A 8D 83 FA A2 FDF0: 03 8E 59 FA 2C 08 FA 38 FB 28 4F FE 4E 5F FA 6E FE00: 5E FA AD 5E FA 8D 5C FA AD 5F FA 8D 5D FA A2 08 FE10: 20 72 FE 20 2B FE C9 7F D8 B5 60 2C 08 FA 38 FB FE20: 8E 61 FA A2 08 20 72 FE 20 81 FE 2C 08 FA 10 09 FE30: 38 6E 62 FA CA D8 F1 F0 07 18 6E 62 FA CA D8 E8 FE40: 28 81 FE AD 62 FA 29 7F 8D 63 23 AE 61 FA 68 18 FE50: 5E FA AD 5E FA 8D 5C FA AD 5F FA 8D 5D FA A2 08 FE60: 2C 80 FA 10 EA AD 5A FA 8D 5E FA AD 5B FA 8D 5F FE70: FA 68 AD 5C FA 8D 5E FA AD 5D FA 8D 5F FA 38 AD 5E FE80: 8C AD 5A FA 8D 5E FA AD 5B FA 8D 5F FA 38 AD 5E FE90: FA 89 01 8D 5E FA AD 5F FA E9 00 8D 5F FA EA EA FEAB: 8B 8B 8E FEAC: AD 82 FA 29 FE 8D 82 FA 28 81 FE 87 4E 62 FA FEBC: 98 30 AD 82 FA 09 81 8D 82 FA 28 81 FE CA D8 ED FED0: AE 59 FA AD 82 FA 09 01 8D 82 FA 28 81 FE CA D8 FE00: F2 2C 08 FA 10 04 AE 68 FA 60 2C 08 FA 10 FB 6C FEF0: 7C FA AD 82 FA 29 FE 8D 82 FA 18 98 CD 20 03 FF FF00: 4C 51 2A A9 27 8D 82 FA A9 08 8D 88 FA A9 7F 8D FFB0: 01 FA 48 8D 83 FA 68 A9 28 8D 7C FA A9 FF 8D 7D FFC0: FA A9 08 8D 7A FA A5 FC 8D 7B FA 4C 18 FD A9 03 FFD0: 8D 25 23 68 A9 51 8D 7C FA A9 2A 8D 7D FA A9 00 FFE0: 8D 7A FA A9 FC 8D 7B FA 4C 18 FD A9 03 FFF0: FA 28 1E FD A9 28 8D A3 26 A9 01 8D 5E 26 20 BC FF00: 26 A9 2A 85 FF 28 54 27 86 FE 28 67 29 A9 01 8D FF10: 21 23 8D 22 23 8D C5 2A 28 C6 29 A9 1A 8D 01 23 FF20: 8D 03 23 A9 A2 8D 11 23 8D 13 23 A9 FE 8D 02 23 FF30: 8D 04 23 8D 12 23 8D 14 23 8D 16 23 8D 18 23 FF40: BC 26 20 67 29 EE 5E 26 A9 00 85 FE 8E 5F 28 67 FF50: 29 A9 81 8D 5E 26 A9 13 28 BC 26 A9 32 85 FF A9 FF60: 74 85 FE 28 54 27 28 67 29 28 61 27 28 73 2D 8D FF70: 8A 2A 54 52 41 43 4B 28 3B 26 31 2A 8D 8A 8B 4C FF80: 08 FC 20 51 FF 4C C9 FF 20 51 FF 4C 9A FF FF FF FFF0: FF FF

Bild 17. Das Hex-Dump-Listing für das Bootstrap-EPROM ESS515. Das Source-Listing ist als "Paperware" erhältlich.

Die Kommandos PU TT und LO TT

Mit diesen Kommandos kann man eine File im Workspace des Computers ohne File-Name auf die Diskette schreiben oder von der Diskette zurück in den Computer lesen. Die File im Workspace darf allerdings nicht länger als zwei kByte sein. Von der Verwendung dieser beiden Kommandos ist abzuraten, da man leicht bestehende Software auf der Diskette überschreiben kann, ohne durch eine Fehlermeldung gewarnt zu werden.

Das Kommando RE oder RETURN

Mit den RETURN-Kommandos können Sie vom DOS-Command-Interpreter zum momentanen Transient-Prozessor zurückkehren:

- RE AS return to Assembler
RE BA return to BASIC
RE EM return to Extended Monitor
Weitere Kommandos können Sie den Manuals von Ohio entnehmen. Wir wollen hiermit die Software-Anpassungen

und die Besprechung des Disk-Operating-Systems abschließen und noch auf ein paar Hardwarebesonderheiten des DOS-Junior-Computers eingehen.

Hardware-Besonderheiten eines 6502-DOS-Computers

Computer, auf denen ein Disk-Operating-System installiert ist, benötigen ein gutes Netzteil. Für den DOS-Junior sollte ein Netzteil vorgesehen sein, das genügend Reserven hat und impulsfest ist. Wenn Sie einen erweiterten Junior bereits haben, dann können Sie das vorhandene Netzteil weiterhin verwenden. Nur für die Drives ist ein neues 12 V/4 A-Netzteil zu bauen. Sollten Sie aber noch kein Netzteil haben, dann können Sie die Schaltungen von Bild 14 und Bild 15 verwenden. Diese Netzteile versorgen den gesamten DOS-Computer und sind einfach und betriebssicher nachzubauen. Die gesamte Schaltung läßt sich auf nur vier ELEKTOR-Platinen unterbringen: 2 mal ESS 82570 und 2 mal ESS 9968-5a. Es werden zwei Transformatoren benötigt, die folgende Eigenschaften haben sollten: 9...10 V/10 A und 15 V/4,4...5 A. Beide Trafos sollten einen Ringkern haben.

Ringkerntrafos sind etwas teurer als geschichtete Trafos, haben jedoch den Vorteil, daß sie leichter sind und ein geringes Störfeld produzieren. Sogar die Wärmeentwicklung eines Ringkerntrafos hält sich bei voller Last in Grenzen.

Ein dritter Trafo mit 2 mal 15 V/1 A erzeugt die Versorgungsspannung für die Dynamic-RAM-Karten und das Drucker-Interface. An diesen Trafo werden keine besonderen Anforderungen gestellt. Mit den Netzteilschaltungen in Bild 14 und Bild 15 werden Sie niemals Schwierigkeiten beim DOS-Junior oder jedem anderen 6502-Computer haben, da diese reichlich überdimensioniert und kurzschlußsicher sind. Die Netzteile lassen sich in einem Flachgehäuse von 300 mm mal 70 mm mal 200 mm unterbringen. Bei der Verdrahtung sollten Kabel mit einem Querschnitt von 1,5 Quadratmillimeter verwendet werden.

Die Dynamic-RAM-Karte.

Die Erfahrung hat gezeigt, daß die Dynamic-RAM-Karte sich "weigert" mit bestimmten 6502-Prozessoren zusammenzuarbeiten. Auch vom 6809-Prozessor hat diese Karte bisher wenig gehalten. Deshalb haben wir ein paar Modifikationen auf der Dynamic-RAM-Karte angebracht, die diese Speicherkarte auf jedem 6502-6809-System enorm betriebssicher machen. Bild 16 zeigt alle Modifikationen, die wir im Schaltbild und auf der Platine vorgenommen haben.

Und jetzt wünschen wir Ihnen viel Spaß mit dem DOS-Junior-Computer, der zur Zeit der preiswerteste und beliebteste Personal-Selbstbau-Computer mit Disk-Operating-System ist!