

**Das andere  
Betriebssystem für  
den JU+TE Computer:**

**ES2.3**

**(Edit System)**

1989

## Wesentliche Merkmale

- komfortabler Bildschirmeditor (Full Screen Editor: FSE)
  - Bild mit 128\*128Punkten bzw. 16\*16Zeichen
  - schnelles Kassetteninterface mit Fehlerkorrekturmöglichkeit
  - Editor für Basic-Programme (EDI)
  - Maschinenmonitor (MON)
  - Tastatur mit echter SHIFT-Taste und Autorepeat
  - programmiert auf EPROM U2732 oder größer
  - Tastatur mit 62Tasten (für weniger Tasten siehe hinten unter Tastatur)
- 

## Notwendige Hardwareumbauten

### Grundplatine

- Am DL030 die Leiterzüge zu Pin 5 und 6 auftrennen und die Pins an +5V legen
- Den Leiterzug von Pin9 DL074 zu Pin8/9 der beiden D195 am DL074 auftrennen
- Schaltung nach Bild 1 einbauen, diese teilt 8MHz durch 2.5 und liefert den Bildpunkttakt von 3.2MHz
- Der DL074 ist nicht mehr notwendig, stört aber auch nicht

### Tastatur

- In die sonst ungenutzte Tastaturspalte 0 eine SHIFT-Taste, diese verbindet mit S0, und eine CONTROL-Taste, diese verbindet mit S1, einbauen. Die Tastaturspalte 0 ist Pin 1 des MH74154 oder Pin 15 des unteren DS8205 bei der Schaltung aus JU+TE 2/88.
- In alle Spaltenleitungen vom Tastaturdekoder zur Tastatur sind Dioden, mit der Kathode zum Dekoder, einzufügen, die für die notwendige Entkopplung sorgen, damit z. B. SHIFT gemeinsam mit einer anderen Taste gedrückt werden kann.
- Eventuell müssen die Widerstände am V40098 auf z.B. 3kOhm verringert werden.

### Speicher

- Eine Kapazität von 8kbyte (%E000-%FFFF) ist ausreichend. Bei weniger als 8kbyte muss mindestens der Bereich %E000-E3FF und %F400-%FFFF bestückt sein.

### Kassetteninterface

- Es kann die 4k Interfaceversion von Dr. Hoyer verwendet werden, es geht aber auch weit weniger umständlich, siehe Bild 2
  - Die Portleitungen P31, P32 und P33 sind über etwa 20 kOhm an +5V zu legen. Sie dürfen ihren Pegel beim Laden von Kassette nicht verändern.
-

## Allgemeine Bemerkungen zur Nutzung

Der Cursor ist ein inverser Strich unter der betreffenden Zeichenposition, der nicht blinkt. Nach dem Einschalten oder RESET erscheint auf dem Bildschirm die Meldung ES2.3\_SWB\_1989. Der Cursor steht etwas tiefer. Der Computer befindet sich im Basic-Editor EDI. Jede Tastenbetätigung wird beim Loslassen der Taste registriert. Wird eine Taste länger gedrückt, dann wirkt die Autorepeat-Funktion, d.h. das Zeichen wird in schneller Folge eingegeben bis die Taste losgelassen wird. Die SHIFT- oder CONTROL-Taste ist wie bei einer Schreibmaschine gemeinsam mit einer anderen Taste zu drücken. Jede Eingabe ist mit der RETURN-Taste (von anderen Autoren als ENTER bezeichnet) abzuschließen, dies ist später nicht extra vermerkt.

## Erweiterte Möglichkeiten

Dieses Betriebssystem kann auch auf jedem beliebigen U881 Einchipmikrorechner genutzt werden, wenn der U881 mit dem 7.5Volt-Reset gestartet wird. Der EDI funktioniert dann noch soweit, dass der MON aufgerufen werden kann, der ohne Einschränkungen bei jedem U881 funktioniert. Der U881 Version 004 kann auch mit normalem Reset gestartet werden.

---

## Der Bildschirmeditor (FSE)

Der FSE ist ein Teil des Betriebssystems und realisiert den Dialog mit dem Nutzer. Er wird von EDI und MON benutzt um vom Nutzer Eingaben zu erhalten. Neben der normalen Funktion, das beim Drücken einer Zeichentaste das Zeichen auf dem Bild erscheint, sind noch folgende Editierfunktionen vorhanden (Tastenbezeichnung):

CLS löscht das Bild und setzt Cursor auf Anfang

HOME setzt Cursor auf Bildanfang

SOL setzt Cursor auf den Anfang der Zeile

DBS löschender Rückschritt

DEL Zeichen herauslöschen und Rest von rechts heranrücken

INS Leerzeichen einfügen

RETURN beendet die Bearbeitung der Zeile, in der der Cursor stand

Die Pfeiltasten bewegen den Cursor eine Stelle nach links/rechts oder eine logische Zeile nach oben/unten

Eine logische Zeile kann je nach Programm 16, 22 oder 64 Zeichen lang sein. Es sind also 16, 8 oder 4 logische Zeilen auf dem Bild vorhanden.

Der EDI verwendet 64 Zeichen lange Zeilen, während des Basic-Programmablaufs sind die Zeilen 16 Zeichen lang und MON arbeitet mit 32 Zeichen/Zeile.

---

## Kassetteninterface

Während des Ladens und Speicherns ist das Bild abgeschaltet. Die Aufzeichnung erfolgt nach einem 5kHz Vorton von 5 sek Dauer in Datenblöcken, zwischen denen kurze Pausen sind. Das Laden erfolgt ohne irgendwelche Tonausgaben. Fängt es doch an zu piepen, dann ist

ein Ladefehler aufgetreten. Dieser Fehler kann meist durch das Betriebssystem korrigiert werden, wenn der Nutzer das Band bis einige Blöcke vor die Fehlerstelle spult und dann solange die Taste C drückt, bis auch in den Pausen zwischen den Blöcken der Computer nicht mehr piept. Sollte dies auch nach mehreren Versuchen nicht gelingen, kann mit der Taste B abgebrochen werden.

---

## EDI

Der EDI dient zur Arbeit mit dem Basic des U883. Er versteht folgende 'Kommandos':

N New, löscht das Basic-Programm  
L lädt ein Programm von Kassette  
S speichert ein Programm auf Kassette  
?X druckt Wert der Variable X aus, auch ?A, ?B usw.  
M ruft den MON auf  
E listet das Programm beginnend mit der ersten Zeile zeilenweise auf, zur Fortsetzung Leertaste drücken, sonst eine andere Taste E00125 beginnt mit dem Auflisten bei Zeile 00125, sonst wie E ohne folgende Zahl  
R RUN startet das Programm  
C CONT setzt das Programm nach STOP-Befehl fort  
PO schaltet den Drucker parallel zum Bildschirm an, d. h. jedes Zeichen, das auf den Bildschirm ausgegeben wird, gelangt über die Routine ab %F512 auch auf das dort vorgesehene Ausgabegerät, das meist ein Drucker ist (siehe auch: Sprungtabelle im RAM).  
P schaltet den Drucker wieder aus

00010 oder eine andere 5-stellige Zahl wird als Zeilennummer interpretiert, folgen keine weiteren Zeichen so wird eine evt. vorhandene Zeile mit dieser Nummer gelöscht, sonst wird diese Zeile in das Programm eingefügt. Es kann jede auf dem Bild stehende Basic-Zeile, z.B. durch den E-Befehl aufgelistet, erneut mit RETURN abgeschlossen werden, und sie wird dann ins Programm eingefügt. Der EDI arbeitet korrekt mit allen Zeilennummern, also auch mit solchen, die im L-Byte eine %00 oder %0D haben.

Durch die Ausführung der Befehle L oder S werden die Basic-Variablen A, B und C verändert.

Für die Basic-Befehle werden keine Schlüsselwörter mehr verwendet, sondern nur Buchstaben:

Schlüsselwort (alt)	Buchstabe (neu)
WAIT	W
END	E
RETURN	R
STOP	T
INPUT	I
PROC	O
PRINT	P
PTH	H
GOSUB	S
GOTO	G
LET	L
CALL	C
REM	M

```

TOFF          /
TRAP...TO...  !...,...
IF...THEN...  F...;...
ELSE          >;

```

Damit kommt man aber nach kurzer Gewöhnung gut zurecht.

Als Fehlermeldungen werden ein O : Ende mit END, ein S : Ende mit STOP oder ein E : Error ausgegeben, gefolgt von einer fünfstelligen Zahl, dem Fehlercode (wie in JU+TE beschrieben).

Bei der Verwendung des Basic-Befehls INPUT (Buchstabe I) ist folgendes zu beachten:

Die einzugebende Zahl kann mit allen Funktionen des FSE editiert werden, bevor die Eingabe mit RETURN abgeschlossen wird.

Die Zahl muss aber auf der ersten Position der Zeile (also ganz links) beginnen und nach der Zahl dürfen in derselben Zeile nur noch Leerzeichen stehen.

Das Ausdrucken von beliebigen Programmen ist mit ES2.3 kein Problem: zuerst wird der Drucker mit PO angeschaltet, dann das Programm ganz normal aufgelistet (und dabei automatisch auch auf den Drucker ausgegeben) und zuletzt wird der Drucker mit P wieder ausgeschaltet.

## MON

Der MON ermöglicht die Nutzung und Bearbeitung von Maschinenprogrammen. Er verwendet ausschließlich Hexadezimalzahlen. Wird er aufgerufen, dann sichert er %FD auf dem Stack und %10-1F im RAM auf %F780-F78F. Bei der Rückkehr mit Q werden diese alten Werte wieder hergestellt.

### Befehlsbeschreibung

Erklärung der in der Beschreibung verwendeten Zeichen:

aaaa 2byte Hexzahl (vierstellig)

bc 1byte Hexzahl (zweistellig)

d ASCII-Zeichen

Haaaa	Hexanzeige der Speicherstellen (oder Register, falls Adresse 0000-00FF), es werden 4byte ausgegeben
,aaaa bcbcbcbc	Hexschreiben des Speichers/der Register(0000-00FF), auch bei weniger als 4byte werden Daten eingeschrieben, aber keine neue Adresse ausgegeben
Aaaaa	ASCII -Anzeige des Speichers, die Codes 00-0F werden als Leerzeichen angezeigt
;aaaa dddddddd	ASCII-Schreiben, es werden immer 8byte geschrieben
Q	Rückkehr zum aufrufenden Programm
?aaaa aaaa	gibt aaaa plus aaaa und aaaa minus aaaa aus
Saaaa aaaa	speichert ab Adresse aaaa auf Kassette aaaa Bytes

Laaaa	lädt von Kassette in Speicher ab Adresse aaaa, druckt dann aaaa bc aus, aaaa ist geladene Byteanzahl und bc Fehlermeldung (00: fehlerfrei)
Maaaa aaaa aaaa	verschiebt von Adresse aaaa nach aaaa aaaa bytes im Speicher. auch bei Überschneidung des Quell- und Zielbereichs erfolgt immer korrekte Verschiebung
Jaaaa	startet Maschinenprogramm ab Adresse aaaa mit CALL, das Programm kann mit RET zurückkehren, das dabei benutzte Nutzer-Flagregister ist sonst in %16 gespeichert und der Nutzer-Registerpointer in %17
P	schaltet den Drucker aus
PO	schaltet den Drucker an
Faaaa aaaa bc	füllt Speicher ab Adresse aaaa mit aaaa Bytes bc

Die Befehle A und H werden nach Ausgabe einer Zeile durch die Leertaste fortgesetzt und durch jede andere Taste beendet. Wenn die eingegebenen Zahlen bei den Befehlen nicht korrekt sind, dann wird der Befehl nicht ausgeführt, es erfolgt aber keine Fehlermeldung.

Folgende Befehle sind für Erweiterungen vorgesehen und dürfen nur benutzt werden, wenn die Erweiterung in den Speicher geladen wurde. Bei versehentlichen Eingaben gibt es nicht gleich einen Absturz, da die Erweiterung im RAM nur aufgerufen wird, wenn die Hexzahl aaaa gültig ist

Daaaa	Assemblieren
Exaaaa	Aufruf irgendwelcher Funktionen, x ist das ASCII-Zeichen das als Funktionsname dient.

---

\*\*\* Betriebssysteminformationen \*\*\*

---

### Speicherbelegung

In Klammern: nach Möglichkeit keine direkten Zugriffe

(%F400-F4FF	ASCII-Textspeicher des FSE)
%F500-F57A	System-RAM
(%F57B-F5FF	Kassettenpuffer)
%F600-F6FF	Stack
%F700-F73F	Zeilenspeicher bei Rückkehr aus dem FSE (Routine CHARIN)
%F740-F77F	CONTROL-Belegung der Tasten
%F780-F78F	hier speichert MON (Routine MONITOR) die Inhalte der Register %10-1F

%F790-F79F        belegt durch SAVE und LOAD zum Zwischenspeichern der  
                  Register %04-0F  
%F7A0-F7FF        noch frei  
(%F800-FFFF        Bildwiederholtspeicher)

---

### Registerbelegung

OS: vom ES2.3 in einigen Routinen benutzt

OSD: vom ES2.3 ständig benutzt

%00-03        Ports  
%04-0F        Basic, darf nicht durch Interrupts genutzt werden, da  
                  durch SAVE und LOAD genutzt, aber nicht modifiziert  
%10-1F        Basic, MON  
%20-25        Basic-Variable A,B,C , SAVE und LOAD  
%26-53        Basic-Variable D-Z  
%54            OSD(IRQ-Zähler)  
%55            OSD(maximale logische Zeilenlänge: %10,%20 oder %40)  
%56/57        OSD(BildIRQvektor)  
%58            OSD(Zeilenlänge)  
%59-5A        OS  
%5B            OSD(Kursorposition: %00-FF, bitte keine direkten  
                  Zugriffe, es gibt ja die Routine SCRFUN)  
%5C            OS  
%5D            Drucker an: %4F , sonst aus  
%5E-65        OS  
%66            OSD  
%67            H-Byte Startadresse des Zeichengenerators  
%68-6B        noch frei (aber nur für den Betriebssystemprogrammierer)  
%6C            OSD(Autorepeatflag: 00 aus)  
%6D            OSD(gedr. Taste: 00 keine)  
%6E/6F        OSD(Uhr, wird alle 0.02 sek. incrementiert)  
%70-%7F       frei

---

### Steuerkodes

%00  
%01    Kursor links  
%02    Kursor rechts  
%03    Kursor hoch  
%04    Kursor runter  
%05    HOME, Kursor auf Bildanfang  
%06    SOL, Kursor auf Zeilenanfang  
%07    DEL, Zeichen herauslöschen  
%08    DBS, löschender Rückschritt  
%09    INS, Leerzeichen einfügen  
%0A  
%0B  
%0C    CLS, Bildschirm löschen und Kursor auf Bildanfang  
%0D    RETURN  
%0E  
%0F

---

## Sprungtabelle im Betriebssystem

%0812 KOMMAND  
%0815 CHARIN  
%0818 CHAROUT  
%081B KEY  
%081E WKEY  
%0821 SAVE  
%0824 LOAD  
%0827 SCRFUN  
%082A MONITOR  
%082D frei für Erweiterungen

KOMMAND Einsprung nach Reset

CHARIN liefert in %13 nacheinander die Zeichen einer vom FSE editierten Zeile, als letztes den Kode %0D. Ist %58 auf %FF gesetzt, dann wird der FSE aufgerufen, auch wenn die alte Zeile noch nicht vollständig übergeben wurde, und sofort eine neue bearbeitet. Bei der ersten Rückkehr nach Zeilenbearbeitung steht in %58 die Byteanzahl-1 der Zeile und die Zeile liegt ab %F700 im Speicher, bis zur max. logischen Zeilenlänge durch Leerzeichen ergänzt oder %0D-Kode.

CHAROUT führt das Zeichen aus %15 auf dem Bildschirm aus und gibt es evtl. noch über %F512 aus (siehe dort)

KEY liefert den Ascii-Kode der momentan gedrückten Taste in %6D, wenn keine Taste dann %00

WKEY wartet mit Kursordarstellung auf Tastendruck, liefert Ascii-Kode in %13

SAVE Speicherbereich auf Kassette speichern, in %20/21 Anfangsadresse, in %22/23 Länge übergeben

LOAD Speicherbereich von Kassette laden, in %20/21 Anfangsadresse übergeben, Rückgabewerte: %22/23 Anzahl der geladenen Byte, %24 Status (00 fehlerfrei, %FF Ladefehler)

SCRFUN Spezialfunktionen für den Zugriff auf den Textbildschirm. Momentan existieren 3 Funktionen, die durch den in %53 an SCRFUN übergebenen Wert unterschieden werden.

Wert:

%00 liefert in %4F (Spalte) und %51 (Zeile) die aktuelle Position des Textkursors.

%01 setzt den Textkursor auf die in %4F (Spalte) und %51 (Zeile) übergebene Position. Werte von 0 bis 15 sind sinnvoll und erreichbar, denn es werden nur die Bits 3 2 1 0 verwendet.

%02 liefert in %53 den ASCII-Code des Zeichens auf der in %4F (Spalte) und %51 (Zeile) übergebenen Position.

Nach Aufruf von SCRFUN mit %01 oder %02 in %53 enthalten %4F und %51 nicht mehr die alten Werte. %53 wird immer verändert.

MONITOR ruft den Maschinenmonitor MON auf. Rückkehr aus MON mit Q.

Alle Routinen stellen sich den Registerpointer selbst ein, und stellen den alten Wert vor der Rückkehr wieder her (gilt auch für die Graphiksprungtabelle)

### Graphiksprungtabelle

%17F4 BWKEY  
%17F7 DRAW  
%17FA PTEST  
%17FD PLOT

BWKEY wartet mit Kursordarstellung auf Tastendruck, liefert den Ascii-Code in %5A (also fast wie WKEY)

DRAW verbindet die Punkte, deren Koordinaten in %4B (x) und %4D (y) sowie %4F (x) und %51 übergeben wurden, durch eine Linie. benutzt Register %60-63 und %72-7F

PTEST liefert in %53 die Farbe des Punktes auf Position aus %4F (x) und %51 (y). benutzt Register %60-63

PLOT setzt auf Position aus %4F (x) und %51 (y) einen Punkt. benutzt Register %60-63

Bei allen 3 Routinen wird die Farbe in %53 übergeben, derzeit wird unterschieden, ob das Bit 0 auf 0 oder 1 ist.

0 : weiß und 1 : schwarz

Der Bildschirm ist ein Koordinatensystem mit dem Nullpunkt in der linken oberen Ecke. Die x- Koordinate wächst nach rechts bis maximal 127. Die y-Koordinate wächst nach unten bis maximal 127.

Für die Nutzung unter Basic gilt folgende Zuordnung:

Registeradresse : Basic-variable

%4B : V %4D : W

%4F : X %51 : Y

%53 : Z

---

### Sprungtabelle im RAM

Die richtigen Sprungbefehle müssen vom Anwender hier eingetragen werden, falls diese Möglichkeiten genutzt werden sollen.

Hierhin springen die IRQ:

%F500

%F503  
%F506  
%F509  
%F50C ist nicht benutzt, dieser IRQ springt über %56/57  
%F50F

%F512 wird durch CHAROUT (%0818) aufgerufen wenn in %5D der Wert %4F steht und kann das Byte aus %15 z.B. auf den Drucker ausgeben. Die Routine muss in %15 und %FD die alten Werte zurückgeben und mit RET enden. Eine Routine, die mit Hilfe einer kleinen Zusatzhardware die Ausgabe auf einen Drucker mit Centronic-Interface realisiert, ist im OS schon vorhanden. Diese Routine beginnt auf %1760. Sie kann also mit dem MON-Befehl

,F512 8D1760

(Eintragen eines Sprunges nach %1760) nutzbar gemacht werden. Dabei ist zu beachten, dass der Drucker so eingestellt wird, dass er bei Ausgabe des Codes %0D Zeilenvorschub und Wagenrücklauf ausführt.

---

### Tastatur

Im EPROM sind die Codes für die normale Tastenbetätigung ab %0F00 gespeichert, die Codes für SHIFT stehen ab %0F40. Die CONTROL-Codes können selbst in den RAM ab %F740 eingetragen werden. Es sind 4 mal 16 Byte, die unterste Tastaturzeile zuerst. Nicht belegte Tasten sind im EPROM auf %FF gelassen. Damit kann man seine Tastaturbelegung ohne EPROM-Löschen selbst ergänzen.

---

### Zeichengenerator

Es ist im EPROM ein 8\*8 Punkte Zeichengenerator (ZG) von 512 Byte vorhanden, der das Bitmuster für die ASCII-Kodes %20-5F enthält. Der ZG ist ab %1000 abgelegt, die Startadresse ist also %0F00. Ein vollständiger ZG ist 2kbyte lang und beinhaltet den Codebereich %00-FF. Ein ZG des Nutzers kann an jeder %xx00 Adresse beginnen. Das High-Byte der Startadresse (xx) ist dann in Register %67 einzutragen.

---

### MON

Benutzte Register

%10-12	Hilfsregister
%16	Flagregister des Nutzers
%17	Registerpointer des Nutzers
%18/19	Zahl zum Ausdrucken
%1A/1B	Zeiger auf das bearbeitete Byte im Speicher
%1C/1D	Ergebnis der ASCII->Zahl Wandlung
%1E/1F	Zeiger auf das nächste Byte im ASCII-Zeilenspuffer

Beim Aufruf von MON und bei der Rückkehr werden kurzzeitig auch %60-63 benutzt.

## Routinen

%0C69 HTA16	druckt die 16 Bit aus %18/19 als 4-Ziffer-Hexzahl aus. %18/19 zerstört.
%0C72 HTA8	druckt die 8 Bit aus %19 als 2-Ziffer-Hexzahl aus, %19 zerstört
%0C7B HTA4	druckt die low-4 Bit aus %19 als Hexziffer aus, %19 zerstört
%0C8D PRRET	druckt einen Return-Code aus (%0D)
%0C91 RWCONT	druckt einen Return-Code aus und wartet dann auf Tastendruck, wurde die Leertaste gedrückt dann ist das Z-Flag gesetzt
%0C9B PCAS	druckt das Zeichen aus %15 aus, dann die 16 bit aus %1A/1B (%18/19 zerstört) als Hexzahl und noch ein Leerzeichen
%0CA9 ADRE	wandelt die 4 Ziffern ab Adresse (%1E/1F) nach %1C/1D und %1A/1B, bei Wandlungsfehler wird POP POP RET ausgeführt, sonst Rückkehr mit um fünf erhöhtem %1F

Fehlermeldung von den folgenden 4 Routinen: C-Flag gesetzt, sonst gelöscht. Die Erhöhung von %1F stimmt nur bei gelöschtem C-Flag.

%0CB8 ATH4	wandelt die Ziffer von (%1E/1F) in low-4 Bit von %1D, %1F um eins erhöht
%0CD5 ATH16	wandelt 4 Ziffern ab (%1E/1F) nach %1C/1D, %12 zerstört, %1F um vier erhöht
%0CDC ATH8	wandelt 2 Ziffern ab (%1E/1F) nach %1D, %12 zerstört, %1F um zwei erhöht

Die folgende Routine gehört eigentlich nicht in den MON und wird von diesem auch nicht genutzt, wegen der Gemeinsamkeiten aber hier mit aufgeführt:

%0A52 DA5TH16	wandelt eine 5 Ziffern lange Dezimalzahl (00000 bis 65535) ab (%1E/1F) nach %14/15, %12/13 und %1C/1D zerstört, %1F um fünf erhöht
---------------	--

In all diesen Routinen wird vorausgesetzt, dass der Registerpointer auf %10 steht.

Ab %0C39 steht eine 4 Byte lange Tabelle, die max. 16 Befehlseinträge umfasst. Ein Eintrag ist noch frei.

Aufbau der Befehlseinträge: 1. Byte ASCII-Befehlszeichen, 2./3. Byte Startadresse

Alle Befehle enden mit RET. Beim Aufruf der Befehle zeigt (%1E/1F) auf das Byte nach dem Befehlsbuchstaben im ASCII-Zeilenpuffer. In den Befehlen wird vorausgesetzt, dass der Registerpointer auf %10 steht.

Erweiterungsbefehle:

D     springt nach %C100  
.  
Ex    springt nach %C106, der ASCII-Code von x liegt in %13

Alle 3 Befehle springen erst nach Ausführung von ADRE (%0CA9) in den RAM. Die Befehle müssen mit RET enden.

Falls auf %C100 RAM vorliegt, kann durch folgende Monitorbefehle ein Absturz von D . E bei fehlender Erweiterung selbst bei korrekter Eingabe der Adresse aaaa unmöglich gemacht werden:

,C100 FFFFFFFF  
,C104 FFFF AF

---

### Kassettenformat

(kompatibel zu einem ATARI-Schnelllader)

Fileaufbau: 5 sek Vorton, Blöcke mit Pause von 0.2 sek  
Blocksendung: Vorimpuls (eine Phase mit 500 mikrosek), %85 Bytes  
Blockaufbau: (Adresse im Kassettenpuffer als erste Zahl)  
%F57B     Blocknummer %00 bis %FF, evt. wieder von vorn  
%F57C     Kopie von %F57B  
%F57D     Prüfsumme über Formatbyte und Datenbytes (8 Bit Summe mit dazuaddierten Carrys)  
%F57E     Kopie von %F57D  
%F57F     Formatbyte:        %FA   weniger als 128 Datenbyte, Anzahl auf %F5FF  
                              %FC   128 Datenbyte  
                              %FE   End of File Block, ohne sinnvolle Datenbytes

%F580-F5FF Datenbytes

Byteaufbau:        bit 7 6 5 4 3 2 1 0

Vorton und Frequenz in den Pausen: 5 kHz.

Bitsendung: eine Schwingung mit 2kHz für H-Bit, mit 4kHz für L-Bit

---

### Hinweis für Programmierer

Wenn nur die hier dokumentierten und nicht mit dem Hinweis <keine direkten Zugriffe> versehenen Eigenschaften des ES2.3 genutzt werden, dann ist Kompatibilität zu anderer Tastaturhardware oder späterer Erhöhung der Bildauflösung (mit neuem Betriebssystem) gegeben.

## Tastaturbelegung

NN	1	2	3	4	5	6	7	8	9	0	<	>	DBS	INS	CLS
NN	Q	W	E	R	T	Z	U	I	O	P	-	=	DEL	↑	SOL
CTL	A	S	D	F	G	H	J	K	L	;	+	*	←	HOM	→
SHT	Y	X	C	V	B	N	M	,	.	/	RET	SPA	SPA	↓	RET

nicht belegte Tasten: noch frei

NN : nicht nutzbar

CTL : CONTPOL

SHT : SHIFT

RET : RETURN

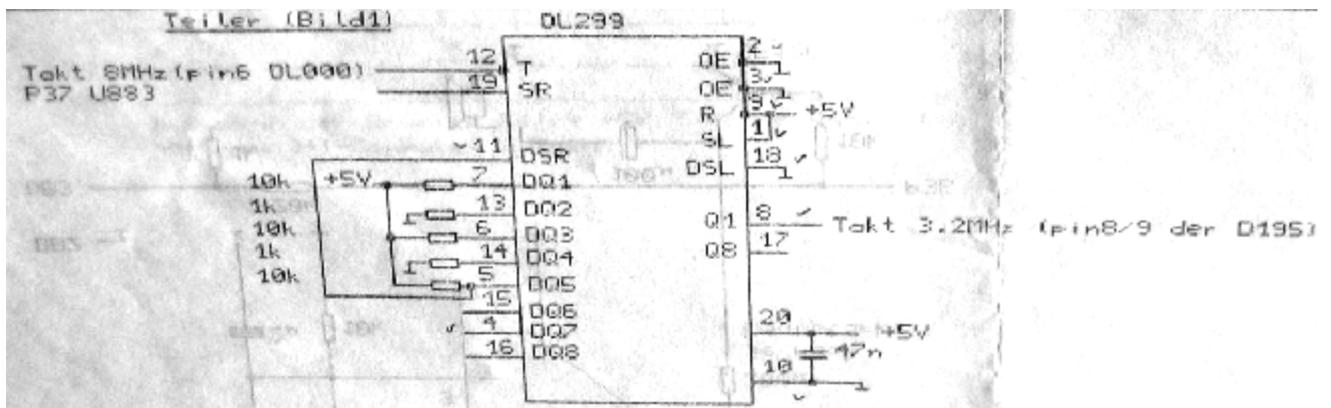
HOM : HOME

SPA : LEERZEICHEN

mit SHIFT:

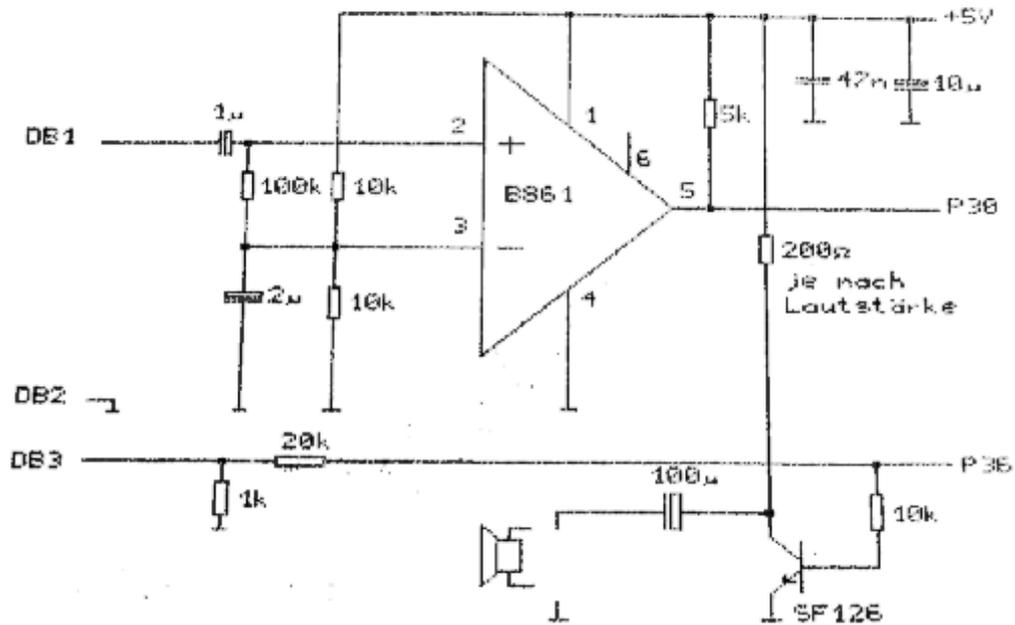
NN	!	"	#	\$	%	&	'	@	(	)					
NN											-				
CTL									:	\	^				
SHT								[	]	?					

## Teiler (Bild 1)

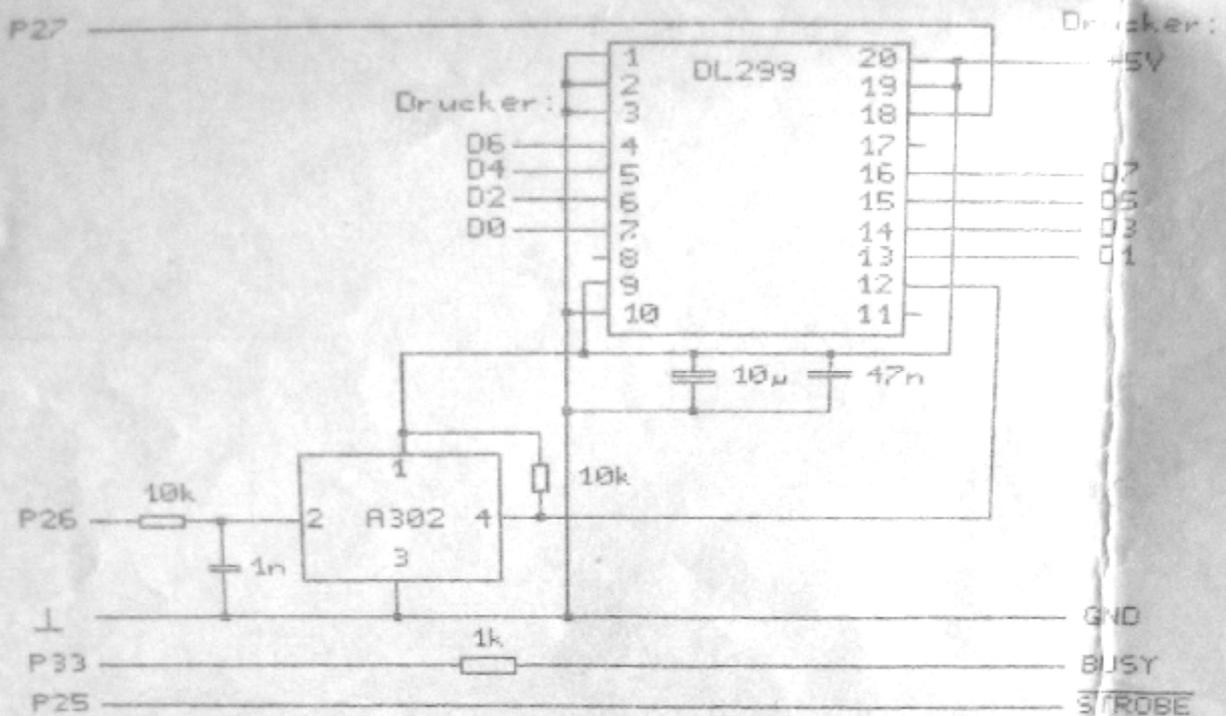


### Kassetteninterface (Bild2)

Diodenbuchse (DB)



### Centronics-Interface am J+T-C.



Nur wenn der Drucker keine +5V liefern kann, die +5V vom J+T-Computer nutzen.