

# ITT 3030, System- und Servicehandbuch

---

```
*****
*
*           I T T   3 0 3 0
*   SYSTEM- UND SERVICEHANDBUCH
*   -----
*
*****
```

## Verzeichnis der Abschnitte (Registernummern)

- 1 Allgemeine Gesichtspunkte
- 2 Grundsystem
- 3 CPU
- 4 Speicher
- 5 Terminalschnittstellen
- 6 interne Schnittstellen
- 7 externe Schnittstellen
- 8 Massenspeicher
- 9
- 10
- 11
- 12
- 13
- 14 Applikationshinweise
- 15 Softwarehinweise

© 1982 by Standard Elektrik Lorenz AG, Pforzheim, Germany  
Produktion: IWT Verlag GmbH, Vaterstetten  
Autor: Bernd Pol, Stuttgart  
Druck: Holdenrieds Druck- und Verlags-GmbH Füssen

Alle Rechte, auch die der Übersetzung, vorbehalten. Kein Teil des Werkes darf in irgendeiner Form (Druck, Fotokopie, Mikrofilm oder einem anderen Verfahren) ohne schriftliche Genehmigung der Standard Elektrik Lorenz AG., Pforzheim reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.  
Technische Änderungen vorbehalten!

## Inhalt (Stand Oktober 1982)

-----  
Die verschiedenen Abschnitte sind am Fuß jeder Seite nach folgendem Muster durchnummeriert:

R - A - S

Dabei ist

- R die Registernummer (1 bis 15),
- A die Abschnittsnummer innerhalb dieses Registers,
- S die Seitennummer in dem Abschnitt.

So gehört beispielsweise das Blatt mit der Seitennummer 5-1.1-24 unter Registerblatt 5 (Terminalschnittstellen) in Abschnitt 1.1 (Video-Tastatur-Schnittstelle, Kombinationskarte 1). Es ist dort Seite 24. Im vorliegenden Inhaltsverzeichnis ist dieser Abschnitt als 5.1.1 verzeichnet.

-----

- 1 Allgemeine Gesichtspunkte
  - 1.1 Systemübersicht
- 3 CPU
  - 3.1 CPU I
- 4 Speicher
  - 4.1 Schreib-Lese-Speicher, 64 KBytes (64K-RAM)
- 5 Terminalschnittstellen
  - 5.1 Video-Tastatur-Schnittstelle
    - 5.1.1 Kombinationskarte 1
  - 5.2 Monitoranpassung
    - 5.2.1 SW-Monitoradapter
  - 5.3 Tastaturanpassung
    - 5.3.1 Tastaturadapter I
  - 5.4 Tastatur
    - 5.4.1 Grund- und Zusatztastatur I

```
*****
*
*           I T T   3 0 3 0
*       S Y S T E M - U N D   S E R V I C E - H A N D B U C H
*       -----
*           A b s c h n i t t   1
*       A l l g e m e i n e   G e s i c h t s p u n k t e
*
*****
```

## 1 Systemübersicht

### 1.1 Konzept

Der ITT 3030 ist ein konsequent modular aufgebautes Mikrocomputersystem, das einfache Erweiterbarkeit mit bequemem und raschem Service verbindet. Es läßt sich, ausgehend von einer Basis-Zentraleinheit, durch Einschub oder Austausch von Modulen lückenlos erweitern. Die einfache Zugänglichkeit dieser Module gestattet es darüberhinaus, vor Ort einen Fehler durch Modultausch in wenigen Minuten zu lokalisieren und zu beheben. So gewährleistet das ITT-3030-Konzept Zukunftssicherheit, geringe Wartungskosten und raschen Service.

### 1.2 Aufbau

Das ITT-3030-System umfaßt

- eine Zentraleinheit, die den eigentlichen Mikrocomputer, ein Netzteil und bis zu zwei Minidiskettenlaufwerke beherbergt,
- eine separate, frei bewegliche Tastatur,
- weitere periphere Zusatzgeräte nach Bedarf, wie Monitor, Drucker, Festplattenspeicher u.a.

Beim gegenwärtigen Stand kann die Zentraleinheit folgende Module aufnehmen (vgl. mit dem "Blockschaltbild ITT 3030"):

- eine CPU-Karte mit einem 8- oder 16-Bit-Mikroprozessor,
- eine RAM-Speicherkarte mit 64 oder mit 256 KBytes Kapazität,
- eine Video- und Tastatur-Schnittstelle entweder für Schwarzweißbetrieb oder für Farb- und Grafikbetrieb und
- verschiedene Monitoradapter,

- eine Echtzeituhr,
- einen Arithmetikbaustein,
- eine serielle Doppel- oder Vierfachschnittstelle im RS232-Standard (V.24-Standard),
- eine parallele Schnittstelle mit zwei 8-Bit-Kanälen,
- eine Schnittstellenkarte für den IEC-Bus (IEEE488-Bus),
- eine Schnittstellenkarte zum Anschluß von auf dem S-100-Busstandard beruhenden weiteren Zusatzkarten
- eine PROM-RAM-BASIC-Karte mit 16 KBytes RAM-Speicher, der mit einer weiteren
- Speicherkarte mit 16 KBytes RAM auf 32 KBytes erweitert werden kann. Alternativ zu diesem ROM-RAM-BASIC-Modul lassen sich
- bis zu zwei Minidiskettenlaufwerke mit 256 bzw 512 KBytes Kapazität und eine dazu passende Steuerkarte (zum Betrieb von drei Minidiskettenlaufwerken geeignet) einsetzen. Weiteren externen Speicherplatz ermöglicht
- eine Schnittstellenkarte zum Ansteuern eines externen 8-Zoll-Diskettenlaufwerks und
- eine Schnittstellenkarte zur Ansteuerung von zwei Festplattenlaufwerken von je 5 MBytes Speicherkapazität.

Die zentralen Systemkarten (CPU, Speicher usw.) sind im Europaformat aufgebaut und können je nach Funktion zu Zweiermoduln zusammengefaßt sein (so trägt beispielsweise die CPU-Karte auch die Karte mit dem RAM-Speicher). Die meisten Schnittstellenkarten besitzen Sonderformate.

Die verschiedenen Moduln stehen über ein gemeinsames Verdrahtungsfeld ("motherboard") in der Bodenwanne der Zentraleinheit miteinander in Verbindung. Ein weiteres Verdrahtungsfeld trägt der Haltebügel für die Diskettenlaufwerke, der somit zur mechanischen und zur elektrischen Verbindung der Laufwerke mit der Zentraleinheit dient.

**Wichtig:** Die verschiedenen Steckplätze auf dem Verdrahtungsfeld sind nicht gleichwertig, d.h. es ist eine bestimmte Anordnung der Karten vorgeschrieben. Ein Überblick dazu findet sich in der beiliegenden Ansicht der geöffneten Zentraleinheit. Im Einzelfall gibt die Beschreibung des jeweiligen Moduls Auskunft über den ihm zugeordneten Steckplatz.

Laufwerke nimmt die Zentraleinheit auf, ein drittes kann extern angefügt werden).

Zusätzlich sind derzeit möglich:

- ein 8-Zoll-Diskettenlaufwerk mit voller IBM-3740-Kompatibilität,
- maximal zwei Festplattenlaufwerke (5 1/4 Zoll) mit Speicherkapazitäten von 5 bzw. 10 MBytes.

## **1.4 Andere Peripherieeinheiten**

Über die S-100-Schnittstellenkarte ist der Ausbau mit weiterer Peripherie aus einem umfassenden Marktangebot von Systemkarten auf Basis des S-100-Busses (nach der IEEE-696-Norm) in einem Zusatzgehäuse möglich.

## **1.5 Gehäuse**

### **1.5.1 Zentraleinheit**

Das Gehäuse der Zentraleinheit besteht aus zwei Teilen, einer Bodenwanne und einer Haube, die vorne durch Kugelschnappverschlüsse und hinten mit Bolzen miteinander verbunden sind. Dadurch läßt sich die Zentraleinheit ohne besonderes Werkzeug öffnen:

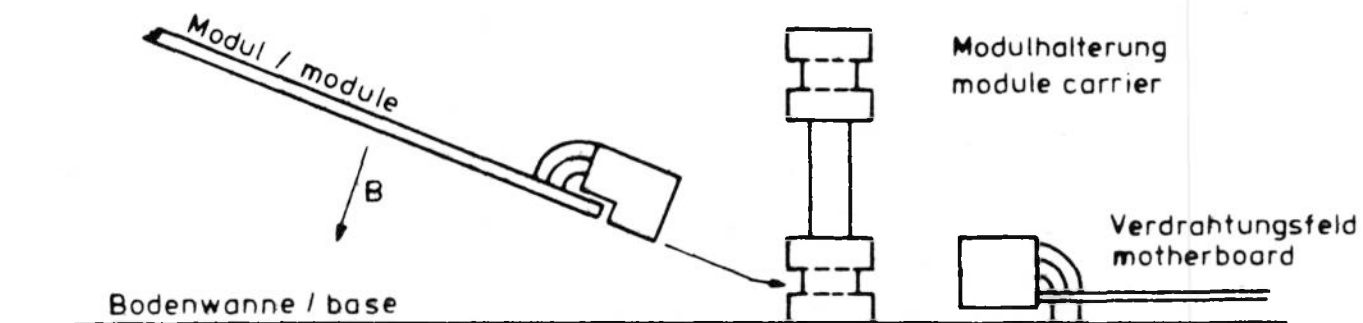
- Fassen Sie das Gehäuse mit beiden Händen an den beiden vorderen Ecken zwischen der dunklen Bodenwanne und der hellen Haube.
- Lösen Sie die Haube durch einen kurzen kräftigen Ruck.
- Lösen Sie die hinteren Bolzenverschlüsse, indem Sie die Haube etwas nach hinten schieben. Sie kann dann frei abgenommen werden.

### **1.5.2 Tastatur**

Das Tastaturgehäuse besteht aus einer Oberschale und einer mit ihr durch Kugelschnappverschlüsse verbundenen Bodenplatte. Legen Sie zum Öffnen die Tastatur auf den Kopf, so daß das Verbindungskabel auf der linken Seite austritt. Greifen Sie dann mit den Zeigefingern in die beiden Löcher im Ihnen zugewandten senkrechten Teil der Bodenplatte und lösen Sie diese vorne mit einem kurzen kräftigen Ruck nach oben. An der (in dieser Lage) hinteren Kante wird die Platte durch Bolzen gehalten, aus denen sie sich einfach herausziehen läßt.

## 1.6 WICHTIGE HINWEISE

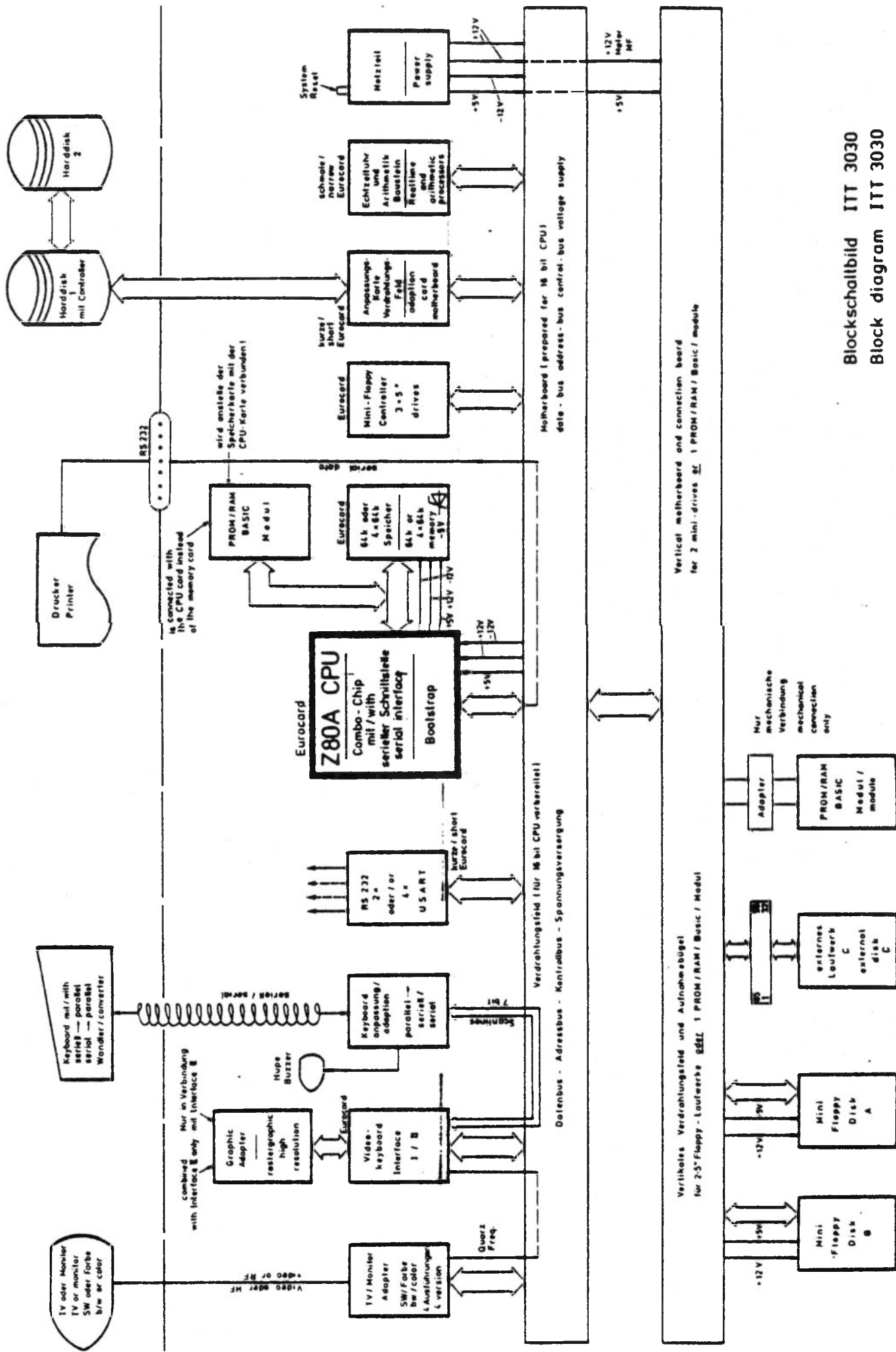
- Das Gehäuse des Netzteils ist **keine** System-Masse. Die System-Masse ist auf dem Verdrahtungsfeld zugänglich.
- Die vorne in der Zentraleinheit liegenden Moduln (CPU, Diskettensteuerung, Video-Tastatur-Schnittstelle usw.) werden an der Steckerleiste durch besondere Führungspfosten entlastet. Sie dürfen daher auf keinen Fall gewaltsam von oben eingedrückt werden, sondern müssen wie in der folgenden Abbildung gezeigt schräg "eingeschwenkt" werden.



Die Schaltkarte ist dabei gleichzeitig nach unten und nach vorne zu schieben. Achten Sie darauf, daß die Karte richtig in den Führungen der Modulhalterung zu liegen kommt. Ist die Karte vollständig gesteckt, so muß sie noch an der Vorderseite durch einen leichten Druck nach unten arretiert werden. Zum Entnehmen der Karte wird sie an der jeweiligen Frontplatte nach vorne gezogen, bis die vordere Arretierung freigegeben ist. Darauf läßt sie sich schräg nach oben aus der Bodenwanne entfernen.

- Die Speicherkarte muß nach dem Aufsetzen auf die CPU an deren Frontplatte verschraubt werden.
- Das Verdrahtungsfeld auf dem Haltebügel für die Diskettenlaufwerke steht mit dem Hauptverdrahtungsfeld in der Bodenschale über eine Steckverbindung ganz rechts (neben dem Netzteilköfig) in Kontakt. Achten Sie hier auf die richtige Steckverbindung.

- Im Unterteil der Diskettenlaufwerke befindet sich ein Schaltgestänge zur Betätigung eines Mikroschalters. Achten Sie beim Einsetzen der Laufwerke darauf, daß dieses Schaltgestänge in seiner Führung bleibt.
- Transportieren Sie den Computer über längere Distanzen grundsätzlich **ohne** Laufwerke!

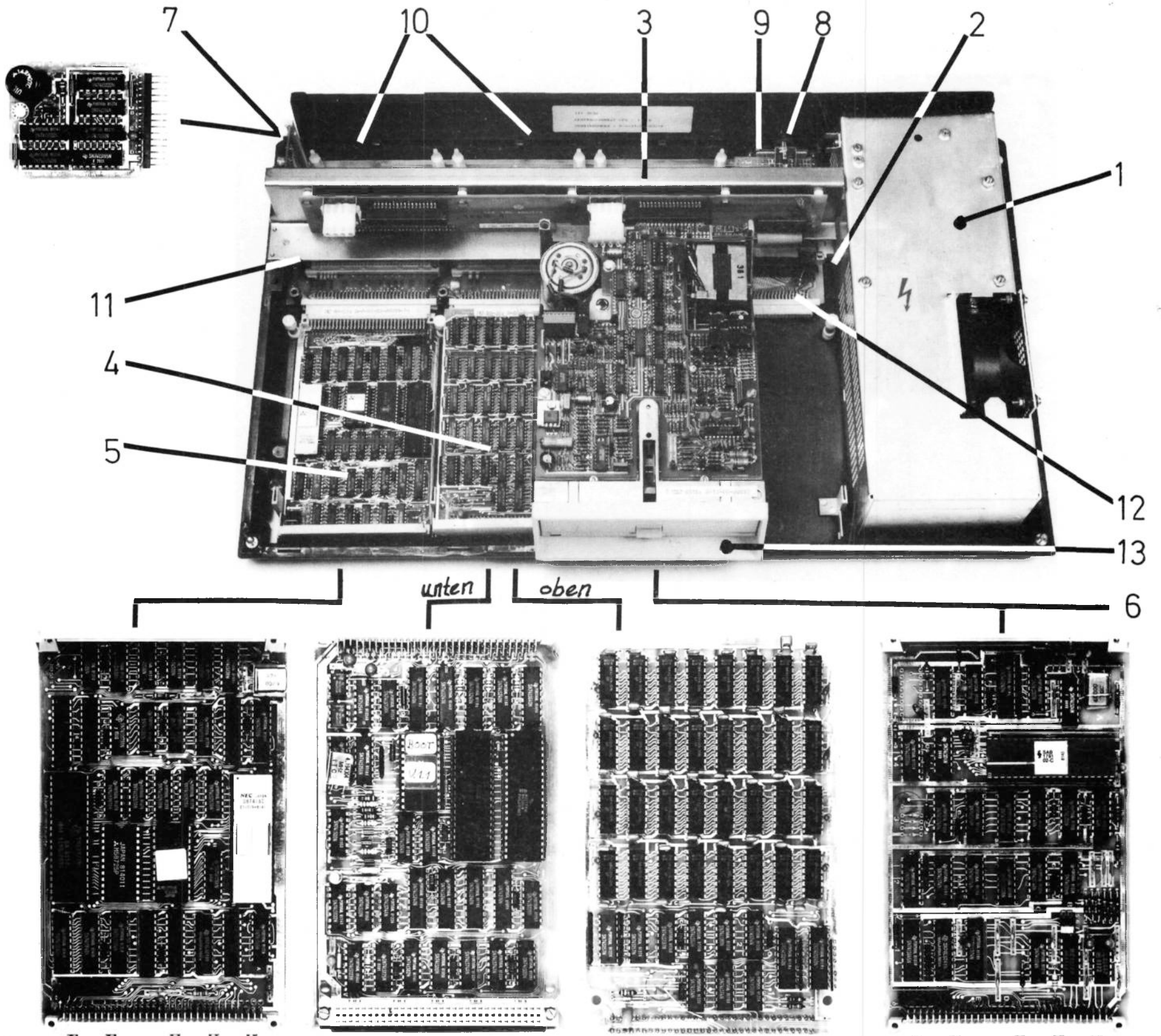


Blöckschaltbild ITT 3030  
Block diagram ITT 3030



# ITT 3030, System- und Servicehandbuch

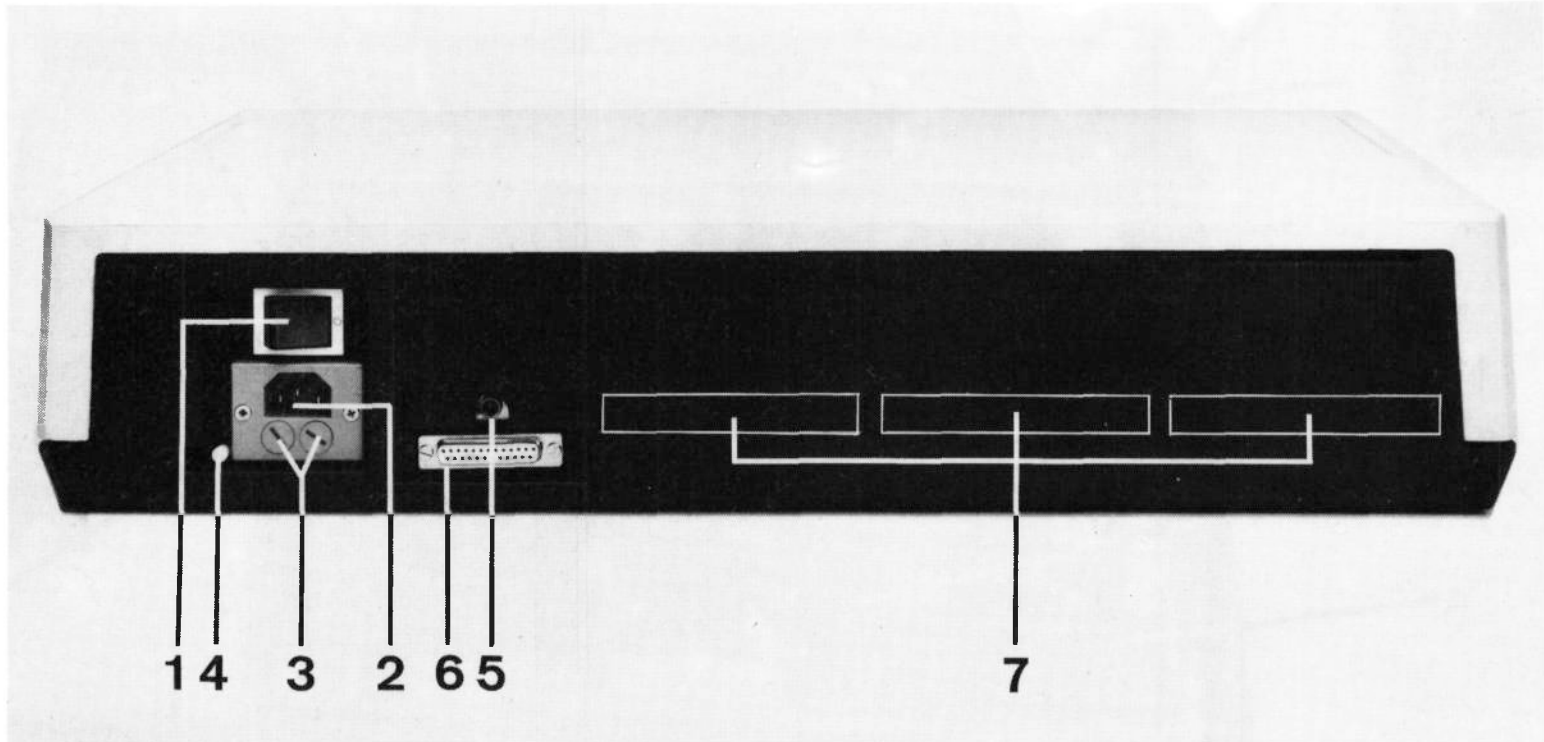
(Foto: Zentraleinheit mit abgenommener Haube)  
(Bild 1 - 3)



- 1: Netzteil
- 2: Verdrahtungsfeld
- 3: Haltebügel für die Diskettenlaufwerke
- 4: CPU- und Speicherkarte
- 5: Video-Tastatur-Schnittstelle
- 6: Mini-Floppy-Controller
- 7: Tastaturanpassung

- 8: Monitoranpassung
- 9: serielle Standardschnittstelle
- 10: Erweiterungsöffnungen in der Gehäuserückwand
- 11: Tastaturanschluß
- 12: Steckplatz für Echtzeituhr und Arithmetikbaustein
- 13: Minifloppy - Laufwerk

(Foto: Gehäuserückwand mit Anschlüssen)  
(Bild 1 - 4)



- 1: Netzschalter mit Lampe
- 2: Netzstecker
- 3: Sicherungen
- 4: Rücksetztaste (RESET)
- 5: Monitoranschluß
- 6: Anschluß der seriellen Standardschnittstelle
- 7: Gehäuseöffnungen für Erweiterungen

```
*****
*
*           I T T   3 0 3 0
*   SYSTEM- UND SERVICEHANDBUCH
*   -----
*           Abschnitt 3
*           C P U
*
*****
```

## 1 CPU I

### 1.0 Inhalt

1.1 Konzept .....	3-1-3
1.2 Schnittstellen .....	3-1-4
1.2.1 Die Systemschnittstelle (Stecker A) .....	3-1-4
1.2.2 Die Speicherschnittstelle (Stecker B) .....	3-1-4
1.2.3 Die serielle Schnittstelle (Stecker A) .....	3-1-6
1.2.4 Die Zählerschnittstelle (Stecker A) .....	3-1-6
1.3 Programmunterbrechungen (interrupts) .....	3-1-7
1.3.1 Die Z-80-Unterbrechungsarten .....	3-1-7
1.3.2 Die Unterbrechungskette (daisy chain) .....	3-1-8
1.4 Die Funktionsblöcke .....	3-1-10
1.4.1 Prozessor und Taktgenerator .....	3-1-10
1.4.1.1 Prozessor .....	3-1-10
1.4.1.2 Taktgenerator .....	3-1-11
1.4.2 Prozessorsteuerung .....	3-1-12
1.4.2.1 Automatische Wartezyklen (WAIT) .....	3-1-12
1.4.2.2 Die Rücksetzlogik (RESET) .....	3-1-14
1.4.3 Der Kombinationsbaustein MK3886 .....	3-1-15
1.4.3.1 RAM .....	3-1-15
1.4.3.1.1 Adressen .....	3-1-15
1.4.3.1.2 Schreibschutz .....	3-1-15
1.4.3.2 Zähler .....	3-1-17
1.4.3.2.1 Betriebsarten .....	3-1-17
1.4.3.2.2 Adressen der Zählerregister .....	3-1-17
1.4.3.2.3 Zähler A .....	3-1-18
1.4.3.2.4 Zähler B .....	3-1-21
1.4.3.2.5 Besonderheiten der Zählerschnittstelle ..	3-1-24

1.4.3.3	Die serielle Schnittstelle .....	3-1-25
1.4.3.3.1	Adressen .....	3-1-25
1.4.3.3.2	Datenaufbereitung .....	3-1-26
1.4.3.3.3	Der Baudratengenerator .....	3-1-26
1.4.3.3.4	Programmierung .....	3-1-27
1.4.3.3.5	Statusinformationen .....	3-1-28
1.4.3.3.6	Asynchroner Datenverkehr .....	3-1-29
1.4.3.3.6.1	Empfang asynchroner Daten .....	3-1-29
1.4.3.3.6.2	Senden asynchroner Daten .....	3-1-30
1.4.3.4	Programmunterbrechungen (interrupts) .....	3-1-32
1.4.3.4.1	Adressen .....	3-1-33
1.4.3.4.2	Programmierung .....	3-1-33
1.4.3.4.3	Unterbrechungsmasken (interrupt masks)...	3-1-35
1.4.3.4.4	Unterbrechungsbehandlung beim MK3886 ....	3-1-35
1.4.3.4.5	INT-Status .....	3-1-36
1.5	Adreßlogik auf der Karte .....	3-1-37
1.5.1	Adressierung der karteninternen E-A-Tore (ports).	3-1-37
1.5.2	Umschaltung der Speicherseiten .....	3-1-38
1.5.3	Die karteninterne Speicherseite 8 .....	3-1-38
1.6	Zusammenfassung der Adressen und Befehle .....	3-1-40
1.6.1	Allgemeines .....	3-1-40
1.6.2	Kombinationsbaustein MK3886 .....	3-1-42
1.6.2.1	Speicher .....	3-1-42
1.6.2.2	Zähler .....	3-1-43
1.6.2.3	Serielle Schnittstelle .....	3-1-45
1.6.2.4	Unterbrechungssteuerung (interrupt control)..	3-1-47
1.7	Steckerbelegung .....	3-1-49
1.7.1	Stecker A .....	3-1-49
1.7.2	Stecker B .....	3-1-52
1.7.4	Serielle Standardschnittstelle .....	3-1-54
1.8	Technische Daten .....	3-1-55
1.9	Bilder .....	3-1-56
1.9.1	Bestückung .....	3-1-56
1.9.2	Anschlußbelegungen .....	3-1-57
1.9.3	Blockschaltbild des Kombinationsbausteins MK3886	3-1-58
1.9.4	Blockschaltbild der CPU-Karte .....	3-1-59

## 1.1 Konzept

Die Karte CPU I ist eine auf dem 8-Bit-Mikroprozessor Z80A beruhende Mehrfunktionseinheit mit Kaltstart-PROM (bootstrap), Unterbrechungssteuerung (interrupt), zwei 8-Bit-Zählern (timer) und einer seriellen RS-232-Schnittstelle (V.24). Über eine separate 96-polige Steckerleiste wird eine Speicherkarte angesprochen werden, die auf die CPU-Karte aufgesteckt wird und so mit ihr eine mechanische Einheit eingeht.

Dazu umfaßt die Karte folgende Funktionsblöcke:

- einen 8-Bit-Mikroprozessor Z80A;
- einen Kombinationsbaustein (Combo-Chip) MK3886 mit Unterbrechungssteuerung (3 interne und 3 externe Unterbrechungsvektoren),  
256 Bytes RAM,  
zwei Zählern und  
einer seriellen Ein-Ausgabe-Schnittstelle;
- 2 KBytes EPROM 2716 mit Kaltstart- und Treiberprogrammen;
- Adressierungslogik zur Umschaltung zwischen karteninternem und kartenexternem Speicherbereich, bis zu 8 verschiedenen externen Speicherseiten zu 64 KBytes;
- Rücksetz- und Wartelogik zur CPU-Steuerung;
- verschiedene Schnittstellentreiber.

## 1.2 Schnittstellen

Es stehen über die Steckerleisten A (Systemstecker) und B (Speicherstecker) vier Schnittstellen zur Verfügung:

1. die Systemschnittstelle mit Daten-, Adreß- und Steuerbus (Stecker A)
2. die Speicherschnittstelle (Stecker B)
3. die serielle Ein- und Ausgabe (auf Stecker A)
4. die Zählerausgänge (auf Stecker A)

### 1.2.1 Die Systemschnittstelle (Stecker A)

Der Datenverkehr mit den anderen Karten auf dem Systembus erfolgt über einen bidirektionalen 8-Bit-Datenbus, einen 16-Bit-Adreßbus und verschiedene Steuerleitungen. Daten- und Adreßbus arbeiten mit negativer Logik (0 = H, 1 = L). Über eine besondere Steuerleitung  $\overline{\text{SR/KOMB}}\overline{\text{I}}$  kann der Adreßraum in zwei Seite zu je 64 KBytes aufgeteilt werden, von denen eine internen Systemzwecken (Aktivierung des Speichers im Video/Tastatur-Kombinationsinterface) dient und die andere den allgemein zugänglichen Speicherbereich abdeckt.

### 1.2.2 Die Speicherschnittstelle (Stecker B)

Der Hauptspeicher des Systems wird von der CPU-Karte über eine besondere Schnittstelle angesprochen, die außer Daten-, Steuer- und 16-Bit-Adreßbus vier weitere Adreßleitungen (ADR 16 bis ADR 19) bereitstellt. Dies ermöglicht theoretisch die Adressierung von 1 M ( $2^{20}$ ) Speicherplätzen in 16 Seiten zu je 64 KBytes Tiefe, doch werden davon nur 8 Seiten genutzt, da ADR 19 zur Unterscheidung zwischen dem internen und dem externen Speicher-raum dient. Es gilt folgende Festlegung:

**ADR 19 = 0:** Interner Speicherraum (definiert als Seite 8)  
(Kaltstart-PROM, karteninterner RAM-Bereich, Bildschirmspeicher)

**ADR 19 = 1:** Externer Speicherraum (Seite 0 bis 7)  
(Systemspeicher)

Zwei weitere Steuersignale dienen - in Abhängigkeit von ADR 19 (s.u.: Adreßlogik) - dazu, die Speicherkarte zu aktivieren:

$SR/\overline{KOMBI} = 0$ : Aktiviert den Speicher auf dem Kombinationsinterface und verriegelt den Systemspeicher.

$SR/\overline{KOMBI} = 1$ : Gibt den Systemspeicher frei.

$\overline{VERR} = 0$ : Verriegelt die Speicherschnittstelle (im Adreßbereich 0 bis C000H der internen Seite 8).

$\overline{VERR} = 1$ : Gibt die Speicherschnittstelle frei.

Die Umschaltung zwischen den Speicherbänken erfolgt durch einen Ausgabebefehl (OUT) über Tor F6 (bzw. FE) hexadezimal, wobei die Datenbits D4 bis D7 die Seite wie folgt bestimmen (beachten Sie die umgekehrte Zuordnung der Bits):

**Seitenumschaltung (OUT F6H):** D7 bestimmt ADR 16  
 D6 bestimmt ADR 17  
 D5 bestimmt ADR 18  
 D4 bestimmt ADR 19

Das ergibt für die Seitenumschaltung folgende Datenwerte (niederwertige Bits als 0 angenommen):

Seite	D7 ADR 16	D6 ADR 17	D5 ADR 18	D4 ADR 19	OUT-Daten
0	0	0	0	1	10H
1	1	0	0	1	90H
2	0	1	0	1	50H
3	1	1	0	1	D0H
4	0	0	1	1	30H
5	1	0	1	1	B0H
6	0	1	1	1	70H
7	1	1	1	1	F0H
8	x	x	x	0	z.B. 00H

Man kann aus der internen Seite 8 ohne besondere Umschaltung auf die Adressen C000 bis FFFF der gerade aktivierten externen Speicherseite (0 bis 7) zugreifen. Das ermöglicht einen Austausch gemeinsamer Daten.

Zur Aufrechterhaltung des Inhalts der dynamischen Speicherbausteine ist das **Auffrischsignal RFSH** des Z80-Prozessors auf den Speicherschnittstelle (Stecker B), nicht aber auf den Systembus (Stecker A) geführt.

### 1.2.3 Die serielle Schnittstelle (Stecker A)

Am Stecker A (d.h. auf dem Systembus) liegen die folgenden Standardsignale der seriellen RS-232-Schnittstelle an:

$\overline{\text{RxD}}$ ,  $\overline{\text{TxD}}$ : Daten (negative Logik)  
RTS, CTS, DSR, DTR: Anforderungs- und Quittungssignale

Die Datenströme werden über die serielle Schnittstelle im Kombinationsbaustein MK3886 geleitet (s.u., Abschnitt 1.4.3.3). Die Anforderungs- und Quittungssignale müssen programmgesteuert über Ein- und Ausgabepore erfaßt bzw. bereitgestellt werden. Dabei gilt folgende Zuordnung:

#### Serielle Anforderungs- und Quittungssignale

Ausgabe (Tor F3, Z80-Befehl OUT 0F3H): Bit 0 -> RTS  
Bit 1 -> DTR  
Eingabe (Tor F0, Z80-Befehl IN 0F0H): DSR -> Bit 4  
CTS -> Bit 5

### 1.2.4 Die Zählerschnittstelle (Stecker A)

Die Ausgänge der beiden Zähler im Multifunktionsbaustein MK 3886 sind (in negativer Logik) auf den Systembus geführt:

$\overline{\text{ZCA}}$ : Zähler A = Null  
 $\overline{\text{ZCB}}$ : Zähler B = Null

Setzen und Starten der Zähler erfolgt über Software. (Zähler A kann auch über die Unterbrechungsleitung  $\overline{\text{INT}}\overline{\text{O}}$  gesteuert werden.) Zähler B liefert darüber hinaus den Sende- bzw. Empfangstakt für die serielle Schnittstelle. Taktquelle für die Zähler ist der halbierte Systemtakt  $\phi$ .



### 1.3 Programmunterbrechungen (interrupts)

Die CPU-Karte verfügt über mehrere Möglichkeiten, ein laufendes Programm zu unterbrechen. (Einzelheiten sind einer ausführlichen Z80-Beschreibung zu entnehmen.)

- Eine **nicht maskierbare Unterbrechungsanforderung** erfolgt über den  $\overline{\text{NMI}}$ -Anschluß 61 am Stecker A (Systembus). Dieser Anschluß führt auf den  $\overline{\text{NMI}}$ -Eingang des Z80-Prozessors und löst einen Aufruf des Programms ab Adresse 66H in der gerade aktiven Speicherseite aus.  $\overline{\text{NMI}}$  hat die höchste Priorität in der Unterbrechungslogik des Systems.
- Die **allgemeine Unterbrechungsanforderung**  $\overline{\text{INT}}$  an die Z80-CPU ist über Anschluß 66 von Stecker A auf den Systembus geführt. Sie kann zum Auslösen einer Programmunterbrechung in einer der drei Unterbrechungsbetriebsarten des Z80-Prozessors eingesetzt werden.
- Ferner stehen über den Kombinationsbaustein MK3886 **drei weitere Unterbrechungsanforderungen** zur Verfügung, die mit  $\overline{\text{INT}}_0$ ,  $\overline{\text{INT}}_1$  und  $\overline{\text{INT}}_2$  bezeichnet sind und an Anschluß 62 bis 64 von Stecker A angefordert werden können.

Die Unterbrechungsanforderungen  $\overline{\text{NMI}}$  und  $\overline{\text{INT}}$  werden auf der CPU-Karte mit je einem Widerstand auf die Versorgungsspannung + 5 V gezogen und müssen so über Treiber mit offenen Kollektoren angesteuert werden. Das ergibt die Möglichkeit, Anforderungen von verschiedenen Quellen im System über Verdrahtetes ODER (wired OR) zusammenzuschalten.

#### 1.3.1 Die Z-80-Unterbrechungsarten

Der Z80-Prozessor kennt drei verschiedene Betriebsarten (interrupt modes) zur Behandlung einer Unterbrechungsanforderung.

**Modus 0:** Dies sind 8080-kompatible Unterbrechungen, bei denen die unterbrechende Einheit auf Bestätigung der Anforderung ein Befehlsbyte auf den Datenbus legt. Dieses Byte - üblicherweise ein RST-Befehl - bestimmt die auszuführende Unterbrechungsroutine.

**Modus 1:** Ähnlich wie bei der nicht maskierbaren Unterbrechung  $\overline{\text{NMI}}$  wird hier eine feste Speicherstelle (Adresse 38H) aufgerufen, wo die Unterbrechungsroutine beginnt. Diese Betriebsart gestattet die schnellste Bearbeitung eindeutig festliegender Programmunterbrechungen.

**Modus 2:** Die unterbrechende Einheit legt bei Bestätigung der Anforderung eine 8-Bit-Adresse auf den Datenbus. Diese bildet die niederwertige Hälfte der Adresse einer 16-Bit-Speicherstelle, in der die Adresse der anzuwählenden Unterbrechungsroutine steht. Die höherwertige Hälfte wird dem Unterbrechungsregister I (interrupt register) der Z80-CPU entnommen, das vorher passend vom Programm geladen werden mußte.  
(Bit 0 der niederwertigen Adreßhälfte muß immer den Wert Null haben.)

### 1.3.2 Die Unterbrechungskette (daisy chain)

Computersysteme auf Z80-Basis haben die Möglichkeit, durch Anordnung der Karten im System die Prioritäten von Unterbrechungsanforderungen zu bestimmen. Dazu werden besonders eingerichtete Bausteine benötigt, mit denen sich eine **Unterbrechungskette** (interrupt daisy chain) aufbauen läßt. Diese Bausteine besitzen zwei zusätzliche Anschlüsse, IEI (interrupt enable in, Eingang der Unterbrechungsaktivierung) und IEO (interrupt enable out, Ausgang der Unterbrechungsaktivierung). Ein besonderes Signal zur Aktivierung der Unterbrechungsbestätigung wird über diese Anschlüsse jeweils von IEO der einen zu IEI der auf dem Bus folgenden Karte durchgeschleift.

Bestätigt der Prozessor (durch gleichzeitiges Aktivieren der Synchronisiersignale M1 und IOREQ) eine Unterbrechungsanforderung, dann dekodiert die erste Einheit in der Kette, deren IEI-Eingang auf H-Pegel liegt, diese Meldung. Hat sie die Unterbrechung nicht angefordert, dann legt sie ihren IEO-Ausgang ebenfalls auf H-Pegel, was den IEI-Eingang der nächstfolgenden Karte in der Kette hochzieht. Diese Karte dekodiert ebenfalls die Unterbrechungsbestätigung des Z80-Prozessors und gibt gegebenenfalls das Aktivierungssignal über ihren IEO-Ausgang an ihren Nachfolger weiter. Findet in diesem Prozeß eine der in der Unterbrechungskette angeordneten Karten heraus, daß sie eine Programmunterbrechung angefordert hat, dann läßt sie ihren IEO-Ausgang auf L-Pegel liegen und blockiert so alle in der Kette später eventuell vorliegenden Anforderungen.

Die solcherart aktivierte Karte meldet sich dann dem gewählten Unterbrechungsmodus gemäß an den Z80-Prozessor und wird von diesem geeignet (durch Aufruf des zugeordneten Programms) bedient. Ein besonderer Endbefehl (RETI, return from interrupt) bewirkt

dann, daß der Prozessor das Ende der Bedienung an die betreffende Karte meldet, die daraufhin ihren IE0-Ausgang freigibt (auf H-Pegel legt). Es können dann eventuell weiter hinten in der Kette vorliegende Unterbrechungsanforderungen erfüllt werden.

Im ITT-3030-System sind durch diese Kette die folgenden Unterbrechungs-Prioritäten vorgegeben:

**Unterbrechungskette (daisy chain) im ITT3030**

**höchster Vorrang:** CPU-Karte  
Adaptersteckplatz IV (ganz links)  
Adaptersteckplatz V  
Adaptersteckplatz VI

**niedrigster Vorrang:** Arithmetikbaustein und Echtzeituhr

**WICHTIG:** Achten Sie darauf, daß die Kette nicht unterbrochen wird. Belegen Sie die Steckplätze in der angegebenen Folge. Gegebenenfalls müssen die IEI- und IE0-Anschlüsse eines freien Steckplatzes mit einer Brücke verbunden werden, um die Kette zu schließen.

## 1.4 Die Funktionsblöcke

### 1.4.1 Prozessor und Taktgenerator

#### 1.4.1.1 Prozessor

Als zentrales Rechen- und Steuerwerk auf der CPU-I-Karte dient der 8-Bit-Mikroprozessor Z80A, dessen Befehlssatz und Registerstruktur aufwärtskompatibel zum 8080- bzw. 8085-Prozessor ist. Er verarbeitet 8-Bit-Wörter und kann über 16 Adreßleitungen unmittelbar 65536 (64 K) Speicherstellen adressieren. Darüber hinaus ist ein Ansprechen von je 256 Ein- bzw. Ausgabeteilen (I/O ports) möglich.

(Die Ein-Ausgabeteile werden in dieser Dokumentation anhand ihrer hexadezimalen Adresse unterschieden. So ist z.B. "Tor EA" das Ein- oder Ausgabeteil auf der hexadezimalen Adresse 0EAH.)

Adreß- und Datenleitungen sind über invertierende 8-Bit-Bustreiber (unidirektional für die Adressen, bidirektional für die Datenleitungen) an den System- bzw. Speicherbus geschaltet. Der Speicherbus wird zusätzlich über ein Vierfach-Latch mit den vier Seitenadreibits ADR 16 bis ADR 19 versorgt. Dieses Latch wird über Ausgabeteil Nummer F6 getriggert und übernimmt dabei den Wert der Datenleitungen D 4 bis D 7 wie oben dargestellt (s. Abschnitt 1.2.2).

Die vom Z80-Prozessor erzeugten Steuersignale  $\overline{WR}$ ,  $\overline{RD}$ ,  $\overline{MREQ}$ ,  $\overline{IORQ}$ ,  $\overline{MI}$ ,  $\overline{BUSAK}$ ,  $\overline{RFSH}$  und  $\overline{HALT}$  werden zusammen mit dem Systemtakt  $\phi$  ebenfalls über invertierende Treiber auf den System- bzw. Speicherbus gelegt und sind dort ständig verfügbar.

Wichtig für den Systembetrieb ist das RFSH-Signal, das zum Auffrischen dynamischer Speicher benutzt wird. Der Z80-Prozessor enthält einen internen 7-Bit-Zähler, der in jedem Befehlsübernahmezyklus (M1-Zyklus) um Eins weitergezählt und mit dem dritten Taktimpuls ( $T_3$ ) in diesem Zyklus für zwei Taktperioden auf die sieben niedrigwertigen Adreibits gelegt wird. Diese Tatsache wird durch das  $\overline{RFSH}$ -Signal angezeigt. Dynamische Speicherkarten können sich diese Information zunutze machen und so den Speicherinhalt in jeweils 128 Zyklen auffrischen (genauere Informationen finden sich in der Beschreibung der Speicherkarte). Da der längste Z80-Befehl 23 Taktzyklen umfaßt, ist bei einem Sy-

stemtakt von 4,194 MHz ein komplettes Auffrischen des Speichers in maximal 0,7 Millisekunden (ohne Wartezyklen beim Datenzugriff) möglich.

#### 1.4.1.2 Taktgenerator

Der Prozessor arbeitet mit einem Takt  $\phi$  von 4,194 MHz, der durch einen aus zwei Invertern aufgebauten, quarzstabilisierten Oszillator erzeugt wird. Eine besondere Impulsformerstufe ist dem Oszillator nachgeschaltet und gewährleistet, daß die Anstiegs- und Abfallzeiten der Taktflanken unter 30 Nanosekunden liegen.

Der so gewonnene Takt versorgt unmittelbar den Z80A-Prozessor sowie (über einen invertierenden Treiber) den System- bzw. Speicherbus und wird noch einmal halbiert (2,097 MHz) den Zählern im Kombinationsbaustein MK3886 zugeführt. Weiter dient der Takt als Bezug beim automatischen Einschub von Wartezyklen zum Speicher- bzw. E-A-Zugriff.

## 1.4.2 Prozessorsteuerung

### 1.4.2.1 Automatische Wartezyklen (WAIT)

Der Z80-Prozessor unterscheidet zwischen dem Zugriff auf den Speicher und dem auf Ein-Ausgabebus. Dabei läßt sich der Speicherzugriff noch untergliedern in einen Befehlsübernahme- und einen normalen Datenzyklus. Für jede dieser drei Zugriffsarten

- **Befehl übernehmen**  
(M1 und MREQ aktiv),
- **Speicherdaten übernehmen oder schreiben**  
(MREQ aktiv) und
- **E-A-Daten übernehmen oder ausgeben**  
(IORQ aktiv)

läßt sich zur Synchronisation mit langsamen Einheiten - softwaregesteuert - jeweils ein Wartezyklus einschieben.

Zu diesem Zweck muß dem Z80-Prozessor bei Beginn des Zugriffszyklus über den  $\overline{\text{WAIT}}$ -Anschluß ein Wartesignal gegeben werden. Der Prozessor untersucht den Zustand dieses Eingangs jeweils mit dem zweiten Taktzyklus im Zuge des Speicher- bzw. E-A-Zugriffs und schiebt solange Wartezyklen (von je einem Taktzyklus Dauer) ein, wie dieser Eingang auf Null liegt. Das Wartesignal  $\overline{\text{WAIT}}$  kann dabei entweder von der angesprochenen Einheit erzeugt werden (über Anschluß 48 an Stecker A), oder es wird auf der CPU-Karte bereitgestellt.

Dies geschieht synchron mit dem Systemtakt  $\phi$  durch je zwei Flipflops (ein Paar für den normalen Datenverkehr und ein Paar für die im Timing davon abweichende Befehlsübernahme). Diese Flipflops erzeugen am  $\overline{\text{WAIT}}$ -Eingang einen Warteimpuls von einem Taktzyklus Dauer, wenn die zugehörigen Anforderungssignale IORQ, MREQ bzw. M1 aktiv und an die Vorbereitungseingänge der Flipflops durchgeschaltet sind.

Das Durchschalten dieser Anforderungssignale an den jeweiligen  $\overline{\text{WAIT}}$ -Impuls-Generator wird durch drei Flipflops festgelegt, die durch Ausgabebefehle softwaregesteuert ein- oder ausgeschaltet werden. Dabei gilt folgende Zuordnung:

### Vorprogrammierung von Wartezyklen

Tor	schaltet Wartezyklus bei	Befehl
F0	Befehlsübernahme	OUT 0F0H
F1	Speicherzugriff	OUT 0F1H
F2	Zugriff auf ein Ein-Ausgabeter	OUT 0F2H

wobei Bit 0 im jeweils ausgegebenen Datenbyte den Zustand der Wartelogik bestimmt:

**D0 = 1:** Die Wartelogik wird eingeschaltet.

**D0 = 0:** Die Wartelogik wird ausgeschaltet.

**WICHTIG:** Nach dem Neueinschalten des Systems und nach jedem Rücksetzen durch RESET ist die Wartelogik für alle drei Zugriffsarten eingeschaltet.

**Kartenexterne Wartezyklen** müssen über Treiber mit offenen Kollektoren über Leitung 48 des Systembusses angefordert werden. Das ermöglicht eine Zusammenschaltung der  $\overline{W\bar{A}I\bar{T}}$ -Signale von verschiedenen Systemeinheiten als Verdrahtetes ODER (wired-OR).

#### 1.4.2.2 Die Rücksetzlogik (RESET)

Im Prinzip kann der Z80-Prozessor durch einen beliebig, aber mindestens drei Taktzyklen langen 0-Impuls am  $\overline{R\overline{E}\overline{S}\overline{E}\overline{T}}$ -Eingang rückgesetzt werden. Das löscht die Unterbrechungsaktivierung (interrupt enable), den Programmzähler PC, den Unterbrechungszeiger I, den Auffrischzähler R und aktiviert den Unterbrechungsmodus Null. Wesentlich vom Systemstandpunkt ist dabei jedoch, daß für die Dauer des  $\overline{R\overline{E}\overline{S}\overline{E}\overline{T}}$ -Signals Adreß- und Datenbus abgeschaltet und alle vom Prozessor ausgegebenen Steuersignale inaktiv werden. Das bewirkt vor allem, daß der Systemspeicher kein Auffrischsignal mehr erhält und so die dort festgehaltene Information verlorenzugehen droht, dauert der  $\overline{R\overline{E}\overline{S}\overline{E}\overline{T}}$ -Impuls zu lange an. (Der Speicher muß mindestens alle 2 Millisekunden aufgefrischt werden.)

Aus diesem Grund wird auf der CPU-Karte ein eigenes Rücksetzsignal erzeugt. Mit der positiven Rückflanke des  $\overline{R\overline{E}\overline{S}\overline{E}\overline{T}}$ -Eingangs (Anschluß 46 am Stecker A) wird ein Flipflop gesetzt und so die Rücksetzlogik aktiviert. Der nächste  $\overline{M\overline{T}}$ -Impuls der CPU, der den Beginn eines neuen Taktzyklus ankündigt, triggert über ein weiteres Flipflop einen monostabilen Multivibrator, der einen etwa 1,3 Millisekunden langen Rücksetzimpuls erzeugt. Dieser Impuls ist im System über Anschluß 47 von Stecker A als  $\overline{R\overline{S}\overline{T}}$ -Signal verfügbar.

Ein weiteres RC-Glied (56 KOhm, 10 uF) sorgt beim Neueinschalten der Stromversorgung dafür, daß die RST-Leitung mindestens 200 Millisekunden aktiv, d.h. auf Nullpegel gehalten wird. Das gewährleistet einen definierten Ausgangszustand des Systems.



### 1.4.3 Der Kombinationsbaustein MK3886

Eine Anzahl wichtiger Funktionen auf der CPU-Karte werden von einem Kombinationsbaustein ausgeführt. Dieser "Combo-Chip" MK3886 enthält:

- 256 Bytes RAM, davon 64 Bytes schreibschützbare,
- zwei 8-Bit-Zähler mit programmierbarem Vorteiler,
- eine serielle Ein- und Ausgabeschnittstelle und
- die Logik für drei zeigergesteuerte Unterbrechungsebenen mit
- Prioritätslogik im "daisy-chain"-Betrieb gemäß dem üblichen Z80-System-Standard.

Die verschiedenen Funktionen sind über besondere, als Ein-Ausgabebetore wirkende Steuerregister programmier- und testbar. Insgesamt stehen 10 E-A-Tore zur Verfügung.

#### 1.4.3.1 RAM

Der Kombinationsbaustein MK3886 enthält 256 Bytes statischen Schreib-Lesespeicher (RAM). Die unteren 64 Bytes davon können schreibgeschützt werden. Dies geschieht automatisch beim Rücksetzen des Bausteins (in erster Linie also beim Einschalten der Stromversorgung), oder der Schreibschutz kann durch Programmieren eines Steuerregisters gesetzt bzw. aufgehoben werden.

##### 1.4.3.1.1 Adressen

**Speicher:**

RAM 1000H...10FFH (auf der internen Seite 8)  
davon schützbare 1000H...103FH

**Ein-Ausgabebetore:**

Schreibschutzregister 0EAH (Ausgabe)  
Statusregister 0E5H (Eingabe)

##### 1.4.3.1.2 Schreibschutz

Der Schreibschutz der ersten 64 RAM-Bytes im MK3886-Kombinationsbaustein wird durch einen Ausgabebefehl der Z80-CPU beeinflusst.

**Schreibschutz im MK3886-RAM (OUT 0EAH)**

<b>Funktion</b>	<b>Ausgabedaten</b>
Schreibschutz aufheben	66H
einen Wert schreiben, dann schützen	55H
Schreibschutz setzen	alle übrigen Werte

Der Schreibschutzzustand des MK3886-RAM-Bereichs kann über das Statusregister des Bausteins (Eingabetor E5) abgefragt werden:

**RAM-Schreibschutzinformation (IN 0E5H)**

- Bit 5 = 1: Die ersten 64 Bytes sind schreibgeschützt.
- Bit 5 = 0: Die ersten 64 Bytes können verändert werden.

### 1.4.3.2 Zähler

Zwei Abwärtszähler mit 8 Bit Breite und programmierbarem Vorteiler (d.h. einem den vorgeschalteten rein internen Zähler) stehen im Kombinationsbaustein MK3886 zur Verfügung. Sie werden über einen gemeinsamen Anschluß mit dem halben Systemtakt (2.097 MHz) versorgt und sind über Software setz- und startbar. Zähler A kann zusätzlich über den Unterbrechungseingang  $\overline{INT} \overline{O}$  angesteuert werden. Beide Zähler können so programmiert werden, daß sie beim Wert 0 eine Programmunterbrechung auslösen.

Die Ausgänge sind auf Anschluß 95 (Zähler A,  $\overline{ZCA}$ ) bzw. 96 (Zähler B,  $\overline{ZCB}$ ) von Stecker A geführt und zeigen den Nulldurchgang des betreffenden Zählers an. Dabei liegt der  $\overline{ZCA}$ -Ausgang (Zähler A) immer dann auf Null, wenn das Zählregister den Wert Null hat. Der  $\overline{ZCB}$ -Ausgang (Zähler B) wird dagegen bei jedem Nulldurchgang umgeschaltet, produziert bei ständigem Durchzählen also ein Signal mit Taktverhältnis 1:1. Diesen Unterschieden entsprechen die für die Zähler möglichen Betriebsarten.

#### 1.4.3.2.1 Betriebsarten

Zähler A	Zähler B
Zeitgeber (interval timer)	Zeitgeber (interval timer)
Impulsbreitenmessung	Baudratengenerator
Ereigniszähler	

Die jeweilige Betriebsart wird durch ein Steuerregister pro Zähler bestimmt.

#### 1.4.3.2.2 Adressen der Zählerregister

Die Steuerregister werden durch Z80-Ausgabebefehle (OUT) programmiert. Je ein weiteres Ausgabebitor wirkt als Zählvorgabe: Sein Inhalt wird nach jedem Nulldurchgang in das Zählregister geladen, so daß ein kontinuierliches Arbeiten mit minimalem Programmaufwand möglich ist. Der jeweilige Zählerstand kann jederzeit über ein Eingabebitor gelesen werden, ohne den Zählvorgang zu beeinflussen. Diese Register befinden sich unter den folgenden E-A-Adressen:

### Ein-Ausgabeadressen der Zählerschnittstellen

Funktion	Zähler A	Zähler B
Steuerung	0E0H	0E2H
Zählvorgabe	0E1H	0E3H, 0E4H
Zählerstand	0E1H	0E3H

#### 1.4.3.2.3 Zähler A

Zähler A läßt sich in den folgenden Betriebsarten einsetzen.

- **Zeitgeber:**  
Zählt die Taktperioden (bei 2,097 MHz) zur Erzeugung exakter Zeitintervalle.
- **Impulsbreitenmessung:**  
Mißt die Dauer eines über  $\overline{INT}\overline{O}$  zugeführten positiven oder negativen Signalimpulses in Takteinheiten (bei 2,097 MHz).
- **Ereigniszähler:**  
Zählt die über die  $\overline{INT}\overline{O}$ -Leitung zugeführten Impulse, wobei auf die positive oder auf die negative Taktflanke getriggert werden kann.

Der Zähler erfaßt 8 Bits, wobei die Zählrate durch einen programmierbaren Vorteiler mit den Faktoren 2, 5 und 20 beeinflussbar ist. Der **Vorteiler** ist ein interner Zähler, der dem eigentlichen Zählregister vorgeschaltet ist. Er zählt eine bestimmte Anzahl der zu erfassenden Impulse ab und gibt erst dann einen Impuls an das eigentliche Zählregister ab. Dies bewirkt eine Teilung der Zählfrequenz durch den für den Vorteiler festgelegten Zählerstand, weshalb man vom **Vorteilfaktor** spricht. So halbiert beispielsweise der Vorteilfaktor 2 die Frequenz des zu zählenden Impulsstroms (oder verlängert das Zählintervall auf das Doppelte), oder man erreicht beim Vorteilfaktor 100, daß immer nur jeder Hundertste Impuls erfaßt wird. Der MK-3886-Kombinationsbaustein ermöglicht für Zähler A eine Verlängerung des Zählintervalls in acht Schritten bis hin zum Zweihundertfachen der Zählvorgabe.

Die Programmierung erfolgt über das Ausgabeter auf Adresse 0E0H. Dabei haben die Bits in dem ausgegebenen Byte folgende Bedeutungen:

Programmformat für Zähler A (OUT 0E0H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+-----+-----+-----+-----+-----+-----+-----+-----+							
! Vorteilfaktor	! Zähler-	! Zähler-	! Zähler-	! INT0-	! INT0-	! Betriebs-	! Start !
! 20 !	! 5 !	! 2 !	! unterbr.!	! Aktiv.!	! Flanke !	! art	! Stopp !
+-----+-----+-----+-----+-----+-----+-----+-----+							

Bit	Bezeichnung	Funktion																																				
7	Vorteilfaktor 20	Die erzielbaren Vorteilfaktoren sind: <table border="1"> <thead> <tr> <th>Bit 7</th> <th>Bit 6</th> <th>Bit 5</th> <th>Faktor</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>2</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>5</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>10</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>20</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>40</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>100</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>200</td></tr> </tbody> </table> Dabei hat der Faktor 1 eine besondere Bedeutung bei der Auswahl der Zählerbetriebsart (siehe Bit 1)	Bit 7	Bit 6	Bit 5	Faktor	0	0	0	1	0	0	1	2	0	1	0	5	0	1	1	10	1	0	0	20	1	0	1	40	1	1	0	100	1	1	1	200
Bit 7	Bit 6		Bit 5	Faktor																																		
0	0		0	1																																		
0	0		1	2																																		
0	1		0	5																																		
0	1		1	10																																		
1	0		0	20																																		
1	0		1	40																																		
1	1	0	100																																			
1	1	1	200																																			
6	Vorteilfaktor 5																																					
5	Vorteilfaktor 2																																					
4	Unterbrechung	1 = Zähler A erzeugt eine Unterbrechungsanforderung, wenn der Stand 0 erreicht worden ist 0 = Zähler A erzeugt keine Unterbrechungsanforderung an die CPU																																				
3	$\overline{\text{INT0}}$ -Aktivierung	1 = unterbricht das Programm mit der aktiven INT0-Flanke (siehe Bit 2) 0 = sperrt die Unterbrechungsleitung INT0 und löscht etwa vorliegende Anforderungen (Zählmodus für $\overline{\text{INT0}}$ )																																				
2	$\overline{\text{INT0}}$ -Flanke	1 = triggert auf abfallende Flanke 0 = triggert auf ansteigende Flanke																																				

1 Betriebsart	1 = Impulsbreitenmessung (Vorteilfaktor immer ungleich 1)
	0 = Ereigniszähler, wenn Vorteilfaktor gleich 1
	0 = Zeitgeber, wenn Vorteilfaktor ungleich 1
0 Start/Stopp	1 = der Zähler läuft an
	0 = der Zähler stoppt und der Vorteiler wird zurückgesetzt

Der Zähler wird **gesetzt** durch Ausgabe eines Bytes über Tor E1 (OUT OE1H), wobei der Zählvorgang unmittelbar beeinflusst wird (für eine verzögerte Vorgabe ohne Beeinflussung des laufenden Zählvorgangs siehe Zähler B). Es wird bis Null heruntergezählt und dann mit dem jeweils vorgegebenen Wert wieder von vorne angefangen. Das **längste Zählintervall** erhält man bei Vorgabe des Werts Null: Hier werden 256 Schritte gezählt. Das **kürzeste Zählintervall** ist durch den Wert 1 gegeben.

Der **Zählerstand** kann jederzeit durch einen Eingabebefehl von Tor E1 gelesen werden (IN OE1H), ohne die Zählfunktion zu beeinflussen.

Das **Zählergebnis** steht auf zweierlei Weise zur Verfügung.

- Als Signal  $\bar{Z}\bar{C}\bar{A}$ :

Wenn das Zählregister den Wert Null erreicht, dann wird der Ausgang  $\bar{Z}\bar{C}\bar{A}$  (zero count timer A) auf Nullpegel gezogen. Ist der Inhalt des Zählregisters ungleich Null, so hat  $\bar{Z}\bar{C}\bar{A}$  den Wert 1. Die Dauer des Nullimpulses am Ausgang hängt vom jeweils eingestellten Vorteilfaktor ab und bewegt so sich zwischen dem Ein- und Zweihundertfachen des jeweiligen Zählimpulses (Takt bzw.  $\bar{I}\bar{N}\bar{T}\bar{O}$ ).

- Als Unterbrechungsanforderung an den Prozessor:

Ist über Steuerbit 4 = 1 (beim Befehl OUT OE0H) der Unterbrechungsmodus für Zähler A gewählt, dann wird mit der abfallenden Flanke von  $\bar{Z}\bar{C}\bar{A}$  das Unterbrechungsflipflop im MK3886-Baustein gesetzt und - vorausgesetzt, sie ist im allgemeinen Unterbrechungsmodus nicht gesperrt (s.u.) - eine Programmunterbrechung vom Z80-Prozessor angefordert.

#### 1.4.3.2.4 Zähler B

Zähler B läßt sich nur als **Zeitgeber** (intervall timer) einsetzen. Die Impulsdauern können dabei gleichmäßig sein oder durch Neuvorgabe der Zählgrenzen je nach Bedarf umprogrammiert werden. Im ersten Fall läßt sich Zähler B als **Baudratengenerator** einsetzen und kann so beispielsweise die interne serielle Schnittstelle des MK3886-Bausteins versorgen.

Der Zähler erfaßt 8 Bits und zählt den halbierten Takt der CPU-Karte (2,097 MHz). Der Versorgungstakt des MK3886-Bausteins von 2,097 MHz wird bei Zähler B immer vorgeteilt, wobei der Faktor in vier Schritten zwischen 2 und 20 wählbar ist. Das ermöglicht **Baudraten zwischen 409,6 Hz und 1048,5 kHz** (bzw. bei weiterer Teilung in der seriellen Schnittstelle mit dem Faktor 16 bis hinunter zu 25,6 Hz; siehe dazu die Besprechung der seriellen Schnittstelle).

Die Impulsperiode T am Zählerausgang  $\bar{Z}\bar{C}\bar{B}$  ergibt sich bei gleichmäßiger Teilung nach der Formel (bei einer Periode von 0,477 ms des Versorgungstakts von 2,097 MHz):

$$T = 2 \times 0,477 \times V \times Z \quad (\text{in ms})$$

Dabei ist:

V der Vorteilfaktor  
Z die Zählvorgabe

Die Frequenz f (in kHz) des an  $\bar{Z}\bar{C}\bar{B}$  ergibt sich entsprechend beim vorgegebenen Versorgungstakt von 2,097 MHz zu:

$$f = \frac{2097}{2 \times V \times Z} \quad (\text{in kHz})$$

Der Zähler wird über das Ausgabeter E2 wie folgt programmiert:

Programmformat für Zähler B (OUT 0E2H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
! Vorteilfaktor !		nicht		! Start !		Zähler- !	
! V 1 !		V 2 !		belegt !		Stopp !	
				unterbr. !		belegt !	

Bit	Bezeichnung	Funktion															
7,6	Vorteilfaktor	Die erzielbaren Vorteilfaktoren sind:															
		<table border="1"> <thead> <tr> <th>V 1</th> <th>V 2</th> <th>Faktor</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>2</td> </tr> <tr> <td>0</td> <td>1</td> <td>5</td> </tr> <tr> <td>1</td> <td>0</td> <td>10</td> </tr> <tr> <td>1</td> <td>1</td> <td>20</td> </tr> </tbody> </table>	V 1	V 2	Faktor	0	0	2	0	1	5	1	0	10	1	1	20
V 1	V 2	Faktor															
0	0	2															
0	1	5															
1	0	10															
1	1	20															
3	Start/Stop	1 = der Zähler läuft an 0 = der Zähler stoppt und der Vorteiler wird zurückgesetzt															
2	Unterbrechung	1 = Zähler B erzeugt eine Unterbrechungsanforderung, wenn der Stand 0 erreicht worden ist 0 = Zähler B erzeugt keine Unterbrechungsanforderung an die CPU															

Zum **Setzen** gibt es für Zähler B zwei Möglichkeiten.

- sofortiges Setzen des Zählregisters:

Dies erfolgt über Ausgabeter E3. Ein Ausgabebefehl (OUT 0E3H) über dieses Tor schreibt das vom Prozessor ausgegebene Byte sowohl in das Zählregister als auch in das Zählvorgaberegister ein. Damit wird das Zählintervall unmittelbar beeinflusst.

- Setzen der Zählvorgabe allein:

Hier wird Ausgabeter E4 benutzt. Mit dem Z80-Befehl OUT 0E4H wird der vom Prozessor ausgegebene Wert lediglich in das Zählvorgaberegister übertragen, ohne den eigentlichen Zählvorgang zu beeinflussen. Erst beim nächsten Nulldurchgang wird das Zählregister hier mit dem neuen Wert geladen. Auf diese Weise lassen sich exakt definierte variable Impulsströme erzeugen.

In beiden Fällen wird der Inhalt des Zählregisters nach Null heruntergezählt und dann mit der jeweiligen Zählvorgabe von neuem begonnen. Das **längste Zählintervall** erhält man bei Vorgabe des Werts Null: Hier werden 256 Schritte gezählt. Das **kürzeste Zählintervall** ist durch den Wert 1 gegeben.



Der **Zählerstand** kann jederzeit durch einen Eingabebefehl von Tor E3 gelesen werden (IN 0E3H), ohne die Zählfunktion zu beeinflussen.

Das **Zählergebnis** steht auf zweierlei Weise zur Verfügung.

- Als Signal  $\overline{ZCB}$ :

Wenn das Zählregister den Wert Null erreicht, dann wird der Ausgang  $\overline{ZCB}$  (zero count timer B) **umgeschaltet**. Auf diese Weise lassen sich genau definierte Impulsströme erzeugen.

- Als Unterbrechungsanforderung an den Prozessor:

Ist über Steuerbit 2 = 1 (im Befehl OUT 0E2H) der Unterbrechungsmodus für Zähler B gewählt, dann wird mit Erreichen des Zählerstands Null das Unterbrechungsflipflop im MK3886-Baustein gesetzt und - vorausgesetzt, sie ist im allgemeinen Unterbrechungsmodus nicht gesperrt (s.u.) - eine Programmunterbrechung vom Z80-Prozessor angefordert.

#### 1.4.3.2.5 Besonderheiten der Zählerschnittstelle

- Bei beiden Zählern wird normalerweise mit einem **Vorteilfaktor** größer als 1 gearbeitet. Lediglich beim Einsatz von Zähler A als Ereigniszähler wird der eingehende Impulsstrom unmittelbar gezählt (d.h. der Vorteilfaktor ist hier gleich 1).
- Der Stand des **Vorteilers** kann nicht gelesen werden. Der Vorteil wird geladen mit Beginn des Zählvorgangs und nach jedem nulldurchgang. Er wird auf Null gesetzt, wenn der jeweilige Zähler angehalten wird. Das bewirkt eine gewisse Ungenauigkeit bei der Impulsbreitenmessung und evtl. auch beim Zeitgereinsatz (wenn der Zähler zwischendurch angehalten wird).
- Der Einsatz der Unterbrechungsanforderung an den Prozessor beim nulldurchgang des Zählers erlaubt es, programmgesteuert **beliebig lange Intervalle** zu erzeugen bzw. zu messen. Unmittelbar lassen sich bei Zähler A Impulse von maximal 51 200 Taktzyklen Dauer (24,4 ms bei 2,097 MHz), bei Zähler B Impulse von 5 120 Taktzyklen Dauer (2,4 ms bei 2,097 MHz) erzeugen oder erfassen.
- Die **Unterbrechungsanforderung** an den Prozessor beim Zähler-nulldurchgang kann zusätzlich bei der Programmierung des allgemeinen Unterbrechungsmechanismus ausmaskiert werden. Dies geschieht über Ausgabebitor E9, wobei folgende Zuordnung gilt:

##### Allgemeine Unterbrechungsmasken (OUT 0E9H)

Zähler A, Unterbrechung aktiviert bei Bit 4 = 1  
Unterbrechung ausmaskiert bei Bit 4 = 0  
Zähler B, Unterbrechung aktiviert bei Bit 5 = 1  
Unterbrechung ausmaskiert bei Bit 5 = 0

Die Zuordnung der Unterbrechungsvektoren zu diesen Funktionen ist unten (in Abschnitt 1.4.3.4.2) wiedergegeben.

### 1.4.3.3 Die serielle Schnittstelle

Der Kombinationsbaustein MK3886 verfügt des weiteren über eine serielle Schnittstelle, die zur Ein- wie zur Ausgabe programmierbar ist. (Ein gleichzeitiges Senden und Empfangen ist jedoch nicht möglich.) Der Datenverkehr kann synchron oder asynchron geschehen, wobei 8 verschiedene Wortlängen zwischen 4 und 16 Bits (einschließlich Synchronisierbits) programmierbar sind.

Die Ein- und Ausgabedaten werden auf der Karte über RS-232C-Treiber geleitet und sind auf Stecker A am Anschluß 78 (Eingabedaten) bzw. Anschluß 79 (Ausgabedaten) verfügbar. Die RS-232-Quittungssignale müssen durch gesonderte Ein- und Ausgabebefehle erfaßt werden. (s.o., Abschnitt 1.2.3).

Die Daten werden im Kombinationsbaustein mit Hilfe eines 16-Bit-Schieberegisters aufbereitet, das vom Prozessor als zwei 8 Bit breite Ein- bzw. Ausgabepore angesprochen werden kann. Dabei kann die Eingabeoperation so programmiert werden, daß der Baustein auf die Anfangsflanke des Synchronisierimpulses (des Startbits) wartet, bevor mit der Auswertung der Daten begonnen wird. Dies erleichtert die Handhabung asynchron übertragener Information.

Dieser seriellen Schnittstelle im MK3886-Baustein ist eine eigene Möglichkeit zur Programmunterbrechung zugeordnet, weitere Kontrollmöglichkeiten bietet ein jederzeit lesbares Statusregister im Baustein.

#### 1.4.3.3.1 Adressen

Die serielle Schnittstelle wird über drei Ein- und Ausgabepore auf den folgenden Adressen angesprochen.

##### Ein-Ausgabeadressen der seriellen Schnittstelle

Funktion	Eingabetor	Ausgabepore
Steuerung	----	0E5H
Status	0E5H	----
Schieberegister		
höherwertiges Byte	0E6H	0E6H
niederwertiges Byte	0E7H	0E7H

#### 1.4.3.3.2 Datenaufbereitung

Die Umwandlung der durch den Prozessor parallel angelieferten Daten in einen seriellen Bitstrom und umgekehrt erfolgt mit Hilfe eines 16-Bit-Schieberegisters auf den Ein-Ausgabebenen E6 (höherwertiges Byte) und E7 (niederwertiges Byte). Nach Maßgabe des durch den Baudratengenerator bereitgestellten Versorgungstakts werden die Daten wie folgt aufbereitet:

**Eingabedaten** werden von links eingeschoben, beginnend mit Bit 15 des Schieberegisters.

**Ausgabedaten** werden nach rechts ausgeschoben, beginnend mit Bit 0 des Schieberegisters.

Es werden immer so viele Bits bearbeitet, wie im Steuerwort vorgegeben worden ist (zwischen 4 und 16). Synchronisierbits (Start- und Stoppbits) müssen ausdrücklich durch das Programm bereitgestellt bzw. ausgewertet werden. Das Ende des jeweiligen Konvertierungsvorgangs (alle im Befehl vorgegebenen Bits bearbeitet) kann durch eine Programmunterbrechung oder durch Abfragen des Statusregisters ermittelt werden.

#### 1.4.3.3.3 Der Baudratengenerator

Der Versorgungstakt wird von Zähler B des Kombinationsbausteins bereitgestellt. Eine Versorgung von außerhalb der CPU-Karte ist nicht vorgesehen. Das von ihm erzeugte Taktsignal wird in den Takteingang der seriellen Schnittstelle eingespeist.

Dort kann dieser Versorgungstakt entweder unmittelbar (ohne weitere Verteilung) verwendet werden oder wahlweise noch einen zur seriellen Schnittstelle gehörenden Verteiler mit dem Teilfaktor 16 durchlaufen. Bei einem Systemtakt von 4,194 MHz lassen sich die wichtigsten Baudraten für den asynchronen Datenverkehr wie folgt programmieren.

Zählvorgabe für Zähler B für die Standardbaudraten  
(Fehler kleiner als 3%)

Baudrate	Vorteilfaktor in Zähler B							
	Versorgungstakt unmittelbar genutzt (nur Sendebetrieb)				Versorgungstakt vorgeteilt mit 16 (Senden und Empfangen)			
	2	5	10	20	2	5	10	20
110	---	---	---	---	---	119	60	30
300	---	---	---	175	109	44	22	11
600	---	---	175	87	55	22	11	---
1200	---	175	87	44	27	11	---	---
2400	218	87	44	22	14	---	---	---
4800	109	44	22	11	7	---	---	---
9600	55	22	11	---	---	---	---	---
19200	27	11	---	---	---	---	---	---

#### 1.4.3.3.4 Programmierung

Die Funktion der seriellen Schnittstelle wird über Ausgabeter E5 gesteuert. Die Bits in dem vom Prozessor beim Befehl OUT 0E5H ausgegebenen Wort haben dabei die folgenden Bedeutungen.

#### Programmformat für die serielle Schnittstelle (OUT 0E5H)

#### Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+	+	+	+	+	+	+	+
! Vort.-	! Senden	! Flanken-	! nicht	! Unter-	! Wortlänge	!	!
! faktor	! Empf.	! detektor	! belegt	! brechg.!	L2 ! L1 ! L0 !	!	!
+	+	+	+	+	+	+	+

Bit	Bezeichnung	Funktion
7	Vorteilfaktor	1 = Versorgungstakt wird nicht geteilt 0 = Verteilung mit Faktor 16 (notwendig beim asynchronen Empfangsbetrieb)
6	Senden/Empfangen	1 = Die Schnittstelle sendet Daten 0 = Die Schnittstelle empfängt Daten

5	Flankendetektor	<p>1 = Die Arbeit beginnt mit der nächsten positiven <math>\overline{RxD}</math>-Flanke (Anschl. 78 an Stecker A)</p> <p>0 = Die Arbeit beginnt sofort nach Laden des Steuerregisters</p>																																				
3	Unterbrechung	<p>1 = Der Baustein fordert eine Programmunterbrechung beim Prozessor an, sobald die angegebene Bitanzahl ein- bzw. ausgeschoben worden ist.</p> <p>0 = Es erfolgt keine Unterbrechungsanforderung.</p>																																				
2...0	Wortlänge	<p>Die Wortlänge kann wie folgt vorgegeben werden (Gesamtzahl der Bits einschließlich Synchronisierbits):</p> <table border="0" style="margin-left: 40px;"> <thead> <tr> <th style="text-align: left;">L 2</th> <th style="text-align: left;">L 1</th> <th style="text-align: left;">L 0</th> <th style="text-align: left;">Wortlänge</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>4</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>7</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>8</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>9</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>10</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>11</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>12</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>16</td></tr> </tbody> </table>	L 2	L 1	L 0	Wortlänge	0	0	0	4	0	0	1	7	0	1	0	8	0	1	1	9	1	0	0	10	1	0	1	11	1	1	0	12	1	1	1	16
L 2	L 1	L 0	Wortlänge																																			
0	0	0	4																																			
0	0	1	7																																			
0	1	0	8																																			
0	1	1	9																																			
1	0	0	10																																			
1	0	1	11																																			
1	1	0	12																																			
1	1	1	16																																			

#### 1.4.3.3.5 Statusinformationen

Der Baustein stellt zwei Informationen über den Zustand der seriellen Schnittstelle im Statusregister über Eingabetor E5 bereit.

##### Status der seriellen Schnittstelle (IN 0E5H)

Bit	Bezeichnung	Funktion
7	Bitzählerstand	<p>1 = Der Bitzähler steht auf Null (d.h. die Arbeit ist beendet).</p> <p>0 = Die Ausgabe läuft noch.</p>
6	Wortende-Synchronisation	<p>Diese Information wird normalerweise beim synchronen Datenverkehr zum exakten Erkennen der Wortgrenzen benötigt. (Ein synchroner Datenverkehr erfordert jedoch eine Änderung der CPU-Karte, da hier der Sendetakt mit auf den Zählerbaustein geführt werden müßte.)</p>

- 1 = Der Bitzähler steht auf Null und der Schiebetak (intern am Register) hat H-Pegel.  
(D.h. die Wortgrenze ist erreicht)
- 0 = Die Wortgrenze ist nicht exakt erreicht.

#### 1.4.3.3.6 Asynchroner Datenverkehr

Die wichtigste Form, die serielle Schnittstelle einzusetzen, besteht im asynchronen Datenverkehr mit externen Einheiten über die RS-232-Schnittstelle des Systems. Asynchroner Datenverkehr beinhaltet, daß die jeweiligen Daten in kleinen Abschnitten von in der Regel 6 bis 8 Bit Länge übertragen und von Synchronisierbits umrahmt werden: einem Startbit, das den Beginn des Datenworts anzeigt und einem oder mehr Stoppbits, die es dem Empfänger erlauben, eine zwischendurch möglicherweise verschobene Synchronisation mit dem Datenstrom wieder herzustellen. Es ist vereinbart, daß ein Startbit den (logischen) Wert Null, Stoppbits den Wert Eins haben. Das entspricht auf der invertierten RS-232-Schnittstelle auf dem Systembus (Stecker A, Signale  $\bar{RxD}$  für den Empfang,  $\bar{TxD}$  zum Senden) einem positiven Impulspegel für das Start- und negativem Pegel für das oder die Stoppbits.

##### 1.4.3.3.6.1 Empfang asynchroner Daten

Zum Empfang eines asynchron übertragenen Datenbits muß

- der Anfang des Startbits erkannt werden,
- sichergestellt werden, daß tatsächlich ein Startbit vorliegt,
- der Datenstrom ab dem Startbit in der vereinbarten Länge möglichst in der Mitte jeder "Bitzelle" abgetastet und in das Schieberegister eingeschoben werden und schließlich
- das so übernommene Wort parallel vom Prozessor aus dem Schieberegister ausgelesen und ausgewertet werden.

Die ersten drei Aufgaben nimmt die serielle Schnittstelle im Kombinationsbaustein wahr - vorausgesetzt, die **Flankenerkennung** und der **Vorteilfaktor 16** sind aktiviert. Der Ablauf ist wie folgt:

- Die Schnittstelle wartet, bis die (positive) Anfangsflanke des Startbits am  $\bar{RxD}$ -Anschluß eingetroffen ist.
- Ist diese Flanke eingetroffen, dann läuft der Vorteilzähler an und überwacht für die nächsten 7 Impulse des Versorgungstakts den  $\bar{RxD}$ -Pegel. Bleibt dieser stabil, so kann angenommen wer-

- den, daß es sich tatsächlich um ein Startbit und nicht um einen Störimpuls handelt.
- Mit dem achten Impuls des Versorgungstakts befindet sich die Schnittstelle in der Mitte der Bitzelle. Ab hier wird mit jedem 16. Impuls des Versorgungstakts der gerade an  $\bar{R}\bar{x}\bar{D}$  anliegende Pegel (invertiert!) in das Schieberegister eingeschoben, und zwar für die im Steuerwort programmierte Anzahl von Bits (einschließlich des Startbitwerts).
  - Ist die geforderte Anzahl von Bits übernommen, dann meldet der Kombinationsbaustein dies (sofern aktiviert und nicht ausmaskiert) durch eine Unterbrechungsanforderung an den Prozessor und setzt gleichzeitig im Statusregister Bit 7 auf Eins.
  - Mit Eintreffen der nächsten Anfangsflanke des nächsten Startbits beginnt dieser Vorgang von vorne. Dabei wird der alte Registerinhalt nach links geschoben. Das bedeutet bei einer Datenrate von 4800 Baud und einem Stoppbit, daß der Prozessor 208 Mikrosekunden Zeit hat, das empfangene Datenwort aus dem Register zu lesen. Es steht dort linksbündig (d.h. beim höchstwertigen Bit 16 beginnend) zur Verfügung.

#### 1.4.3.3.6.2 Senden asynchroner Daten

Das Senden eines Worts gestaltet sich einfacher, da hier keine ankommende Flanke entdeckt werden muß. Folglich kann hier wahlweise mit oder ohne 16er-Vorteilung gearbeitet werden. Die Flankenerkennung sollte ausgeschaltet sein, um einen störungsfreien Betrieb zu gewährleisten.

Der Sendevorgang beginnt hier unmittelbar, nachdem das niederwertige Byte in Tor E7 eingeschrieben worden ist. Dazu wird gemäß dem vorgegebenen Versorgungstakt der Registerinhalt nach rechts, d.h. zum niederwertigen Bit hin in den seriellen Ausgang geschoben, durchläuft eine Wandlerstufe und steht dann im RS-232-Standard an Anschluß  $\bar{T}\bar{x}\bar{D}$  (Stecker A, Anschluß 79) zur Verfügung. Der Sendevorgang stoppt, wenn alle im Befehl angegebenen Bits ausgeschoben worden sind.

Zum seriellen Übertragen von Daten müssen also folgende Schritte ausgeführt werden:

1. Über Ausgabeter E5 Sendebetrieb, Vorteilfaktor und Wortlänge wählen. Die Flankenerkennung muß abgeschaltet werden
2. Wenn nötig, das höherwertige Byte in Tor E6 einschreiben.
3. Das niederwertige Byte in Tor E7 einschreiben. (Das Wort muß linksbündig, mit dem Startbitwert in Bit 0, angeordnet sein.)



4. Wenn alle Bits bearbeitet worden sind, meldet sich der Baustein über die der seriellen Schnittstelle zugeordnete Unterbrechungsanforderung (s.u.) und setzt gleichzeitig Statusbit 7 in Eingabetor E5 auf 1. So läßt sich im Unterbrechungs- oder Abfragebetrieb (polling) ein kontinuierlicher Sendevorgang aufrechterhalten. Zum Senden des jeweils nächsten Worts genügt es, mit Schritt 2 weiterzumachen, d.h. das Steuerregister E5 braucht nicht neu geladen zu werden.

#### 1.4.3.4 Programmunterbrechungen (interrupts)

Der Kombinationsbaustein MK3886 besitzt verschiedene Möglichkeiten zur Erzeugung von Unterbrechungsanforderungen an den Z80-Prozessor. Er kann drei externe Unterbrechungsanforderungen (über die Systembusleitungen  $\overline{\text{INT}}_0$ ,  $\overline{\text{INT}}_1$  und  $\overline{\text{INT}}_2$ ) und vier interne Unterbrechungssituationen (Zähler A und B, Senden und Empfangen über die serielle Schnittstelle) handhaben. Dies geschieht nach dem folgenden Prioritätsschema:

##### Vorrangsstufen der Programmunterbrechungen

- |                             |                              |
|-----------------------------|------------------------------|
| <b>höchster Vorrang:</b>    | 1. $\overline{\text{INT}}_0$ |
|                             | 2. Nulldurchgang Zähler A    |
|                             | 3. Nulldurchgang Zähler B    |
|                             | 4. Wortende Empfangen        |
|                             | 5. Wortende Senden           |
|                             | 6. $\text{INT}1$             |
| <b>niedrigster Vorrang:</b> | 7. $\overline{\text{INT}}_2$ |

(Beachten Sie, daß  $\text{INT}1$  gegenüber  $\text{INT}0$  und  $\text{INT}2$  invertiert ist.)

Jede der Unterbrechungsanforderungen kann unabhängig von den anderen ausmaskiert werden. Es werden Unterbrechungs-Betriebsarten des Z80-Prozessors unterstützt:

##### Unterbrechungs-Betriebsarten des Kombinationsbausteins

- Modus 0: RESTART-Betrieb, 8080-kompatibel  
Übergibt bei Annahme der Unterbrechung an den Prozessor eine 8-Bit-Information, die zum Aufruf einer von acht vorgegebenen Adressen zwischen 0 und 63 führt.
- Modus 2: Zeigergesteuerter Unterbrechungsbetrieb (vectored interrupt)  
Hier wird bei Bestätigung an den Prozessor das niederwertige Byte eines Unterbrechungszeigers übergeben. Das höherwertige stammt aus dem Unterbrechungsregister (interrupt register) I des Z80-Prozessors. Unter der so übergebenen Adresse findet sich dann die Adresse der Unterbrechungsroutine selbst.

Der Baustein ist weiter über die Anschlüsse IEI (interrupt enable input, Stecker A, Anschluß 24) und IEO (interrupt enable output, Stecker A, Anschluß 25) in die Z80-Unterbrechungskette ("daisy chain" der Z80-Unterbrechungseinheiten) eingeschaltet (s.o., Abschnitt 1.3.2).

Da der Kombinationsbaustein auf der CPU-Karte sitzt, hat er in der Unterbrechungskette die höchste Priorität. Dabei muß jedoch beachtet werden, daß der MK3886-Baustein ein etwas anderes Verhalten zeigt, als sonst in der Unterbrechungskette üblich ist (s.u.).

#### 1.4.3.4.1 Adressen

Der Unterbrechungsbetrieb des Kombinationsbausteins MK3886 auf der CPU-I-Karte wird über die folgenden Ausgabebitore bestimmt:

##### Ein-Ausgabeadressen für die Unterbrechungssteuerung

Funktion	Ausgabebitor
Steuerung	0E8H
Ausmaskierung	0E9H

#### 1.4.3.4.2 Programmierung

Die Art, in der die Unterbrechungsanforderungen vom MK3886-Baustein gehandhabt werden, muß wie folgt über Ausgabebitor E8 programmiert werden.

##### Programmformat für die Unterbrechungssteuerung (OUT 0E8H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+-----+-----+-----+-----+-----+-----+-----+-----+							
!	U 7	!	U 6	!	U 5	!	U 4
! Unterbrechungszeiger ! Betriebs-				! Aktivierung ! nicht		!	
! U 7 ! U 6 ! U 5 ! U 4 !				art		! INT1 ! INT2 ! belegt !	
+-----+-----+-----+-----+-----+-----+-----+-----+							

Bit	Bezeichnung	Funktion
7...4	Unterbrechungs- zeiger	Diese vier Bits bilden die höherwertige Hälfte des 8-Bit-Worts, das im zeigergesteuerten Unterbrechungsbetrieb auf den Datenbus gelegt wird. Die niederwertige Hälfte setzt sich wie folgt zusammen:

U 3	U 2	U 1	U 0	Unterbrechung
1	1	1	0	$\overline{INT0}$
1	1	0	0	Zähler A
1	0	1	0	Zähler B
1	0	0	0	Empfangen
0	1	1	0	Senden
0	1	0	0	INT1
0	0	1	0	$\overline{INT2}$

Beim Einsatz des RESTART-Unterbrechungsbetriebs (Modus 0) haben diese Bits keine Funktion. Hier werden folgende Adressen aufgerufen:

Unterbrechung	RST-Adresse
$\overline{INT0}$	56 (38H)
Zähler A	48 (30H)
Zähler B	40 (28H)
Empfangen	32 (20H)
Senden	24 (18H)
INT1	16 (10H)
$\overline{INT2}$	8 (08H)

3	Betriebsart	1 = RESTART-Betrieb 0 = zeigergesteuerter Betrieb
2	Aktivierung INT1	1 = INT1 erzeugt Unterbrechungsanforderung 0 = keine Anforderungserzeugung
2	Aktivierung $\overline{INT2}$	1 = $\overline{INT2}$ erzeugt Unterbrechungsanforderung 0 = keine Anforderungserzeugung

### 1.4.3.4.3 Unterbrechungsmasken (interrupt masks)

Der MK3886-Baustein besitzt neben der Aktivierung einer jeden Möglichkeit zur Programmunterbrechung (INT-Leitungen, Zähler, serielle Schnittstelle) noch die Möglichkeit, jede dieser Anforderungen zu sperren. Das erleichtert die Programmierung in vielen Fällen, da dadurch der allgemeine Stand des jeweiligen Befehls nicht berührt wird. Dieses Sperren von Unterbrechungsanforderungen geschieht durch Setzen einer Unterbrechungsmaske über Ausgabeter E9.

Die Unterbrechungsmaske arbeitet dergestalt, daß in dem ausgegebenen Byte jeder Unterbrechungsmöglichkeit ein Bit zugeordnet ist, das mit der jeweiligen Anforderung UND-verknüpft wird. Hat das Bit den Wert 1, so ist die Anforderung freigegeben, hat es den Wert 0, dann ist sie gesperrt. Im einzelnen gilt folgende Zuordnung:

#### Unterbrechungsmasken im Kombinationsbaustein

Bit	maskiert
6	serielle Schnittstelle
5	Zähler B
4	Zähler A
3	$\overline{\text{INT0}}$
2	INT1
1	$\overline{\text{INT0}}$

(Bits 7 und 0 sind nicht belegt.)

### 1.4.3.4.4 Unterbrechungsbehandlung beim MK3886

Der MK3886-Baustein behandelt die Z80-übliche Unterbrechungskette in etwas anderer Form als sonst üblich. Hat er eine Unterbrechungsanforderung ausgegeben und die Bestätigung durch den Prozessor empfangen, dann legt er zunächst wie gewohnt eine 8-Bit-Meldung auf den Datenbus und veranlaßt so den Aufruf der entsprechenden Bedienungsroutine.

Unmittelbar anschließend jedoch gibt er die Kette frei, indem er seinen IEO-Ausgang auf H-Pegel zieht. D.h. er wartet nicht wie die anderen Z80-Einheiten auf einen RETI-Befehl zu diesem Zweck.

Das bedeutet, daß seine Bedienung jederzeit durch weiter hinten in der Kette liegende Karten unterbrochen werden kann.

Das hat eine besondere Konsequenz bei der Anlage der Unterbrechungs-routinen für den Kombinationsbaustein. Würden sie wie üblich mit RETI abschließen, dann würde eine eventuell weiter hinten in der Kette liegende Unterbrechungsanforderung von der betreffenden Karte als bereits erledigt betrachtet. Störungen im Systembetrieb wären die Folge. **Die Unterbrechungsprogramme zur Bedienung des MK3886-Bausteins müssen daher mit einem normalen "RET"-Befehl (also nicht mit "RETI") enden.**

#### 1.4.3.4.5 INT-Status

Im Statusregister auf Eingabetor E5 des MK3886-Bausteins können Informationen über den Stand der Unterbrechungsanforderungen  $\overline{\text{INT}}_0$  bis  $\overline{\text{INT}}_2$  erfragt werden. Es gilt folgende Zuordnung:

##### INT-Status-Informationen (IN 0E5H)

Bit	(invertierter Pegel auf)
0	$\overline{\text{INT}}_0$
1	$\overline{\text{INT}}_1$
2	$\overline{\text{INT}}_2$

(Beachten Sie, daß die hier vorgefundenen Bitwerte gegenüber den  $\overline{\text{INT}}_x$ -Anschlüssen an Stecker A invertiert sind.)

## 1.5 Adreßlogik auf der CPU-Karte

Durch die Seiteneinteilung des von der CPU-Karte adressierbaren Speicherplatzes wird eine besondere Adreßlogik benötigt, die folgende Aufgaben erfüllt:

- Adressierung der karteninternen Ein-Ausgabepore,
- Anwahl der externen Speicherseiten 0 bis 7,
- Anwahl der internen Speicherseite 8 und dort
- Sperren des allgemeinen Speicherbereichs für Adressen kleiner als C00H sowie
- Unterscheidung zwischen karteninternem und kartenexternem Adreßraum (d.h. zwischen Speicher und Kombinationsinterface).

### 1.5.1 Adressierung der karteninternen E-A-Tore (I/O ports)

Die karteninternen Ein-Ausgabepore liegen im Adreßbereich von E0 und FF. Die Tore von E0 bis EF sind für den Kombinationsbaustein MK3886 reserviert. Die von F0 bis FF dienen zur Speicherumschaltung, zur Aktivierung der Wartelogik sowie zum Quittungsbetrieb bei serieller Ein- bzw. Ausgabe von Daten (die Signale RTS, DTR, DSR, CTS). Einzelheiten dazu finden sich oben in den betreffenden Abschnitten.

Die Auswahl der karteninternen Ein-Ausgabepore besorgt im wesentlichen ein Dekoder. Er wird freigegeben, wenn sowohl  $\overline{WR}$  als auch  $\overline{IORQ}$  aktiv sind (d.h. bei einem Z80-OUT-Befehl) und erfaßt die drei niederwertigen Adreßbits A0, A1 und A2 und die UND-Verknüpfung aus A4, A5, A6, A7, ist also nur bei den Adressen F0 bis FF aktiv.

**Wichtig:** Adreßbit A3 wird nicht mit dekodiert. D.h. es gelten die folgenden Beziehungen:

Funktion	wird erreicht durch
Wartezyklus Befehlskode	OUT 0F0H und OUT 0F8H
Wartezyklus Speicher	OUT 0F1H und OUT 0F9H
Wartezyklus Ein-Ausgabe	OUT 0F2H und OUT 0FAH
Quittungssignale RTS, DTR	OUT 0F3H und OUT 0FBH
Seitenumschaltung	OUT 0F6H und OUT 0FEH

Die E-A-Adressen 0F4H, 0F5H, 0F7H, 0FCH, 0FDH und 0FFH sind dagegen (im Prinzip) für andere Zwecke frei. Der **Systembustreiber** wird allerdings nur freigegeben, wenn eine E-A-Einheit angespro-

chen wird, die außerhalb des internen Adreßbereichs liegt, d.h. nur bei den E-A-Toren von von 00 bis DF.

### 1.5.2 Umschaltung der Speicherseiten

Die Anwahl der Speicherseiten geschieht durch einen Z80--Ausgabebefehl über Tor F6. Die dabei benötigten Datenwerte sind oben in Abschnitt 1.2.2 wiedergegeben. Wichtig ist, daß das erweiterte Adreßbit ADR 19 zwischen internem und externem Speicher unterscheidet. ADR 19 = 0 aktiviert den internen Speicherraum.

### 1.5.3 Die karteninterne Speicherseite 8

Der karteninterne Adreßraum auf Seite 8 ist wie folgt eingeteilt:

Adresse	erfaßt
ab 0000H	Kaltstart- und Treiber-PROM
ab 1000H	RAM-Bereich im MK3886-Baustein
ab 3000H	Speicher auf dem Kombinationsinterface
ab C000H	allgemeiner Systemspeicher

Davon befindet sich das Kombinationsinterface nicht auf der Karte, sondern muß über den Systembus angesprochen werden.

Mit ADR 19 = 0 wird u.a. der Dekoder auf Kartenposition C1 aktiviert. Er dekodiert die Adreßbits A12 und A13 unmittelbar und die NAND-Verknüpfung aus A14 und A15. Das ergibt eine unvollständige Dekodierung mit folgenden **Aktivierungen**:

	Kaltstart-PROM	MK3886-RAM	Kombinationsinterface
<b>Ad-</b>	0000...0FFF	1000...10FF	3000...3FFF
<b>res-</b>	4000...4FFF	5000...50FF	7000...7FFF
<b>sen</b>	8000...8FFF	9000...90FF	B000...BFFF

Damit wird es in späteren Ausbaustufen im Kombinationsinterface möglich, den vollen Adreßbereich von 0000 bis 0C000H zu nutzen, ohne die Kompatibilität mit früheren Versionen aufgeben zu müssen.

Der **Zugriff auf das Kombinationsinterface** (Video- und Tastaturzugriff) aus Seite 8 heraus wird durch das Signal **SR/ $\overline{\text{KOMBI}}$**  geregelt, das in diesem Fall auf Nullpegel liegt. Bei sonstigem Speicherzugriff hat es den Wert 1.



Ein Zugriff auf den Systemspeicher wird über das **Verriegelungssignal  $\overline{V\overline{E}R\overline{R}}$**  verhindert, das aktiv ist, solange nicht die Adreßbits A14 und A15 beide den Wert 1 haben, d.h. bei allen Adressen kleiner als C000H.

Der **Systembustreiber** der CPU-I-Karte wird bei Speicheradressierung in folgenden Fällen gesperrt:

- Der Bus ist einer anderen Einheit freigegeben worden (DMA-Betrieb).
- Die interne Seite ist (bei Adressen unter C000H) ausgewählt, außer wenn dabei das Kombinationsinterface adressiert wurde.

## 1.6 Zusammenfassung der Adressen und Befehle

### 1.6.1 Allgemeines

#### Seitenumschaltung (OUT F6H):

D7 bestimmt ADR 16  
 D6 bestimmt ADR 17  
 D5 bestimmt ADR 18  
 D4 bestimmt ADR 19

Das ergibt für die Seitenumschaltung folgende Datenwerte (niederwertige Bits als 0 angenommen):

Seite	D7 ADR 16	D6 ADR 17	D5 ADR 18	D4 ADR 19	OUT-Daten
0	0	0	0	1	10H
1	1	0	0	1	90H
2	0	1	0	1	50H
3	1	1	0	1	D0H
4	0	0	1	1	30H
5	1	0	1	1	B0H
6	0	1	1	1	70H
7	1	1	1	1	F0H
8	x	x	x	0	z.B. 00H

#### Gliederung der internen Seite 8

	Kaltstart-PROM	MK3886-RAM	Kombinationsinterface
Ad-	0000...0FFF	1000...10FF	3000...3FFF
res-	4000...4FFF	5000...50FF	7000...7FFF
sen	8000...8FFF	9000...90FF	B000...BFFF

### Unterbreungskette (daisy chain) im ITT3030

**höchster Vorrang:** CPU-Karte  
Adaptersteckplatz IV (ganz links)  
Adaptersteckplatz V  
Adaptersteckplatz VI

**niedrigster Vorrang:** Arithmetikbaustein und Echtzeituhr

**WICHTIG:** Achten Sie darauf, daß die Kette nicht unterbrochen wird. Belegen Sie die Steckplätze in der angegebenen Folge. Gegebenenfalls müssen die IEI- und IEO-Anschlüsse eines freien Steckplatzes mit einer Brücke verbunden werden, um die Kette aufrechtzuerhalten.

### Vorprogrammierung von Wartezyklen

Tor	schaltet Wartezyklus bei	Befehl
F0	Befehlsübernahme	OUT 0F0H
F1	Speicherzugriff	OUT 0F1H
F2	Zugriff auf ein Ein-Ausgabeter	OUT 0F2H

wobei Bit 0 im jeweils ausgegebenen Datenbyte den Zustand der Wartelogik bestimmt:

D0 = 1: Die Wartelogik wird eingeschaltet.  
D0 = 0: Die Wartelogik wird ausgeschaltet.

**WICHTIG:** Nach dem Neueinschalten des Systems und nach jedem Rücksetzen durch RESET ist die Wartelogik für alle drei Zugriffsarten eingeschaltet.

## 1.6.2 Kombinationsbaustein MK3886

### 1.6.2.1 Speicher

#### Adressen

Speicher:

RAM 1000H...10FFH (auf der internen Seite 8)  
davon schützbar 1000H...103FH

Ein-Ausgabepore:

Schreibschutzregister 0EAH (Ausgabe)

Statusregister 0E5H (Eingabe)

#### Schreibschutz im MK3886-RAM (OUT 0EAH)

Funktion

Ausgabedaten

Schreibschutz aufheben

66H

einen Wert schreiben, dann schützen

55H

Schreibschutz setzen

alle übrigen Werte

1.6.2.2 Zähler

Ein-Ausgabeadressen der Zählerschnittstellen

Funktion	Zähler A	Zähler B
Steuerung	0E0H	0E2H
Zählvorgabe	0E1H	0E3H, 0E4H
Zählerstand	0E1H	0E3H

Programmformat für Zähler A (OUT 0E0H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+-----+-----+-----+-----+-----+-----+-----+-----+							
! Vorteilfaktor !	Zähler-	INT0-	INT0-	Betriebs-	Start !		
! 20 !	! 5 !	! 2 !	unterbr.!	Aktiv.!	Flanke !	art	! Stopp !
+-----+-----+-----+-----+-----+-----+-----+-----+							

Programmformat für Zähler B (OUT 0E2H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+-----+-----+-----+-----+-----+-----+-----+-----+							
! Vorteilfaktor !	nicht	! Start !	Zähler-	nicht			
! V 1 !	! V 2 !	belegt	! Stopp !	unterbr.!	belegt	!	
+-----+-----+-----+-----+-----+-----+-----+-----+							

Die erzielbaren Vorteilfaktoren sind:

V 1	V 2	Faktor
0	0	2
0	1	5
1	0	10
1	1	20

### Berechnung der Zählvorgaben für Zähler B

Die Impulsperiode T am Zählerausgang  $\bar{Z}\bar{C}\bar{B}$  ergibt sich bei gleichmäßiger Teilung nach der Formel (bei einer Periode von 0,477 ms des Versorgungstakts von 2,097 MHz):

$$T = 2 \times 0,477 \times V \times Z \quad (\text{in ms})$$

Dabei ist:

V der Vorteilfaktor

Z die Zählvorgabe

Die Frequenz f (in kHz) des an  $\bar{Z}\bar{C}\bar{B}$  ergibt sich entsprechend beim vorgegebenen Versorgungstakt von 2,097 MHz zu:

$$f = \frac{2097}{2 \times V \times Z} \quad (\text{in kHz})$$

### Allgemeine Unterbrechungsmasken (OUT OE9H)

Zähler A, Unterbrechung aktiviert bei Bit 4 = 1  
Unterbrechung ausmaskiert bei Bit 4 = 0  
Zähler B, Unterbrechung aktiviert bei Bit 5 = 1  
Unterbrechung ausmaskiert bei Bit 5 = 0

1.6.2.3 Serielle Schnittstelle

Serielle Anforderungs- und Quittungssignale

Ausgabe (Tor F3, Z80-Befehl OUT 0F3H): Bit 0 -> RTS  
 Bit 1 -> DTR  
 Eingabe (Tor F0, Z80-Befehl IN 0F0H): DSR -> Bit 4  
 CTS -> Bit 5

Ein-Ausgabeadressen der seriellen Schnittstelle

Funktion	Eingabetor	Ausgabeter
Steuerung	----	0E5H
Status	0E5H	----
Schieberegister		
höherwertiges Byte	0E6H	0E6H
niederwertiges Byte	0E7H	0E7H

Zählvorgabe für Zähler B für die Standardbaudraten  
 (Fehler kleiner als 3%)

Baudrate	Vorteilfaktor in Zähler B							
	Versorgungstakt unmittelbar genutzt (nur Sendebetrieb)				Versorgungstakt vorgeteilt mit 16 (Senden und Empfangen)			
	2	5	10	20	2	5	10	20
110	---	---	---	---	---	119	60	30
300	---	---	---	175	109	44	22	11
600	---	---	175	87	55	22	11	---
1200	---	175	87	44	27	11	---	---
2400	218	87	44	22	14	---	---	---
4800	109	44	22	11	7	---	---	---
9600	55	22	11	---	---	---	---	---
19200	27	11	---	---	---	---	---	---

*vorteilfaktor  
 $V_1, V_2$  in  
 Steuerregister  
 von Zähler B*

**Programmformat für die serielle Schnittstelle (OUT OE5H)**

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
! Vort.-	! Senden	! Flanken-	! nicht	! Unter-	! Wortlänge		
! faktor	! Empf.	! detektor	! belegt	! brechg.!	L2	L1	L0

Die Wortlänge kann wie folgt vorgegeben werden (Gesamtzahl der Bits einschließlich Synchronisierbits):

L 2	L 1	L 0	Wortlänge
0	0	0	4
0	0	1	7
0	1	0	8
0	1	1	9
1	0	0	10
1	0	1	11
1	1	0	12
1	1	1	16

**Status der seriellen Schnittstelle (IN OE5H)**

Bit	Bezeichnung	Funktion
7	Bitzählerstand	1 = Der Bitzähler steht auf Null (d.h. die Arbeit ist beendet). 0 = Die Ausgabe läuft noch.
6	Wortende-Synchronisation	Diese Information wird normalerweise beim synchronen Datenverkehr zum exakten Erkennen der Wortgrenzen benötigt. (Ein synchroner Datenverkehr erfordert jedoch eine Änderung der CPU-Karte, da hier der Sendetakt mit ausgewertet werden muß.) 1 = Der Bitzähler steht auf Null und der Schiebetakt (intern am Register) hat H-Pegel. (D.h. die Wortgrenze ist erreicht) 0 = Die Wortgrenze ist nicht exakt erreicht.



### 1.6.2.4 Unterbrechungssteuerung (interrupt control)

#### Vorrangsstufen der Programmunterbrechungen

höchster Vorrang:	1. $\overline{INT0}$
	2. Nulldurchgang Zähler A
	3. Nulldurchgang Zähler B
	4. Wortende Empfangen
	5. Wortende Senden
	6. INT1
niedrigster Vorrang:	7. $\overline{INT2}$

#### Ein-Ausgabeadressen für die Unterbrechungssteuerung

Funktion	Ausgabektor
Steuerung	0E8H
Ausmaskierung	0E9H

#### Programmformat für die Unterbrechungssteuerung (OUT 0E8H)

Daten beim OUT-Befehl:

7	6	5	4	3	2	1	0
+-----+-----+-----+-----+-----+-----+-----+-----+							
! Unterbrechungszeiger !		Betriebs-		! Aktivierung !		! nicht !	
! U 7 ! U 6 ! U 5 ! U 4 !		art		! INT1 ! INT2 !		! belegt !	
+-----+-----+-----+-----+-----+-----+-----+-----+							

Unterbrechungszeiger	U 3	U 2	U 1	U 0	Unterbrechung
	1	1	1	0	$\overline{INT0}$
	1	1	0	0	Zähler A
	1	0	1	0	Zähler B
	1	0	0	0	Empfangen
	0	1	1	0	Senden
	0	1	0	0	INT1
	0	0	1	0	$\overline{INT2}$

Beim Einsatz des RESTART-Unterbrechungsbetriebs (Modus 0) haben diese Bits keine Funktion. Hier werden folgende Adressen aufgerufen:

Unterbrechung	RST-Adresse
$\overline{\text{INT}}_0$	56 (38H)
Zähler A	48 (30H)
Zähler B	40 (28H)
Empfangen	32 (20H)
Senden	24 (18H)
INT1	16 (10H)
$\overline{\text{INT}}_2$	8 (08H)

#### Unterbrechungsmaske im Kombinationsbaustein

Bit	maskiert
6	serielle Schnittstelle
5	Zähler B
4	Zähler A
3	$\overline{\text{INT}}_0$
2	INT1
1	$\overline{\text{INT}}_0$

(Bits 7 und 0 sind nicht belegt.)

#### INT-Status-Informationen (IN 0E5H)

Bit	(invertierter Pegel auf)
0	$\overline{\text{INT}}_0$
1	$\overline{\text{INT}}_1$
2	$\overline{\text{INT}}_2$

(Beachten Sie, daß die hier vorgefundenen Bitwerte gegenüber den  $\overline{\text{INT}}_x$ -Anschlüssen an Stecker A invertiert sind.)

## 1.7 Steckerbelegung

### 1.7.1 Stecker A

#### Reihe a

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	+ 12 V	Versorgungsspannung			
4	0 V	Masserückleitung, + 12 V			
5	- 12 V	Versorgungsspannung			
6	0 V	Masserückleitung, - 12 V			
7	M1	Kenner für Maschinenzyklus 1	A	positiv	
8	SR/ $\overline{\text{KOMBI}}$	Umschaltung Speicher-Kombi.	A	positiv	
9	$\overline{\text{ADR}}_0$	Adreßbit 0	A	negativ	
10	$\overline{\text{ADR}}_1$	Adreßbit 1	A	negativ	
11	$\overline{\text{ADR}}_2$	Adreßbit 2	A	negativ	
12	$\overline{\text{ADR}}_3$	Adreßbit 3	A	negativ	
13	$\overline{\text{ADR}}_4$	Adreßbit 4	A	negativ	
14	$\overline{\text{ADR}}_5$	Adreßbit 5	A	negativ	
15	$\overline{\text{ADR}}_6$	Adreßbit 6	A	negativ	
16	$\overline{\text{ADR}}_7$	Adreßbit 7	A	negativ	
17	$\overline{\text{ADR}}_8$	Adreßbit 8	A	negativ	TS
18	$\overline{\text{ADR}}_9$	Adreßbit 9	A	negativ	TS
19	$\overline{\text{ADR}}_{10}$	Adreßbit 10	A	negativ	TS
20	$\overline{\text{ADR}}_{11}$	Adreßbit 11	A	negativ	TS
21	$\overline{\text{ADR}}_{12}$	Adreßbit 12	A	negativ	TS
22	$\overline{\text{ADR}}_{13}$	Adreßbit 13	A	negativ	TS
23	$\overline{\text{ADR}}_{14}$	Adreßbit 14	A	negativ	TS
24	$\overline{\text{ADR}}_{15}$	Adreßbit 15	A	negativ	TS
25	$\overline{\text{DAT}}_0$	Datenbit 0	B	negativ	TS
26	$\overline{\text{DAT}}_1$	Datenbit 1	B	negativ	TS
27	$\overline{\text{DAT}}_2$	Datenbit 2	B	negativ	TS
28	$\overline{\text{DAT}}_3$	Datenbit 3	B	negativ	TS
29	$\overline{\text{DAT}}_4$	Datenbit 4	B	negativ	TS
30	$\overline{\text{DAT}}_5$	Datenbit 5	B	negativ	TS
31	$\overline{\text{DAT}}_6$	Datenbit 6	B	negativ	TS
32	$\overline{\text{DAT}}_7$	Datenbit 7	B	negativ	TS

positive Logik: 1=H, 0=L

negative Logik: 1=L, 0=H

Richtung: E = Eingang, A = Ausgang, B = bidirektional

Treiber: TS = tri state

OC = open collector

Reihe b

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
33	+ 5 V	Versorgungsspannung			
34	0 V	Masserückleitung, + 5 V			
35	+ 12 V	Versorgungsspannung			
36	0 V	Masserückleitung, + 12 V			
37	- 12 V	Versorgungsspannung			
38	0 V	Masserückleitung, - 12 V			
39					
40	HALT	Meldung: CPU ist angehalten	A	positiv	
41	$\overline{\text{MEMR}}$	Speicher lesen (Anforderung)	A	negativ	OC
42	$\overline{\text{MEMW}}$	Speicher schreiben (Anf.)	A	negativ	OC
43	$\overline{\text{IOR}}$	Eingabeter lesen (Anf.)	A	negativ	OC
44	$\overline{\text{IOW}}$	Ausgabeter schreiben (Anf.)	A	negativ	OC
45					
46	$\overline{\text{RES\!ET}}$	externes Rücksetzsignal	E	negativ	OC
47	$\overline{\text{RST}}$	internes Rücksetzsignal	A	negativ	
48	$\overline{\text{WAIT}}$	Warteanforderung an die CPU	E	negativ	OC
49	RD	Lesebefehl (Synchronisation)	A	positiv	
50	$\overline{\text{BUSRQ}}$	Busanforderung an die CPU	E	negativ	TS
51	$\overline{\text{BUSAK}}$	Busfreigabe durch die CPU	A	negativ	
52	WR	Schreibbefehl (Synchronis.)	A	positiv	
53	$\overline{\phi}$	Systemtakt	A	negativ	
54	IORQ	E-A-Tor-Anforderung	A	positiv	
55	MREQ	Speicheranforderung	A	positiv	
56	$\overline{\text{INT}}$	Unterbrechungsanforderung	E	negativ	OC
57	IEI	Unterbrechungskette, Eingang	E	positiv	
58	IEO	Unterbrechungskette, Ausgang	A	negativ	
59					
60					
61	$\overline{\text{NMI}}$	nicht maskierbare Unterbr.	E	negativ	OC
62	$\overline{\text{INT0}}$	Unterbrechungszeiger 0	E	negativ	
63	$\overline{\text{INT1}}$	Unterbrechungszeiger 1	E	negativ	
64	$\overline{\text{INT2}}$	Unterbrechungszeiger 2	E	negativ	

positive Logik: 1=H, 0=L

Treiber: TS = tri state

negative Logik: 1=L, 0=H

OC = open collector

Richtung: E = Eingang, A = Ausgang, B = bidirektional

Reihe c

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
65	+ 5 V	Versorgungsspannung			
66	0 V	Masserückleitung, + 5 V			
67	+ 12 V	Versorgungsspannung			
68	0 V	Masserückleitung, + 12 V			
69	- 12 V	Versorgungsspannung			
70	0 V	Masserückleitung, - 12 V			
71					
72					
73					
74	RTS	RS-232, Sendeanforderung	A	(positiv)	
75	CTS	RS-232, Bestätigung vom Sender	E	(positiv)	
76	DTR	RS-232, Bereitschaftsmeldung	A	(positiv)	
77	DSR	RS-232, Bereitschaftsbestätig.	E	(positiv)	
78	$\bar{R}x\bar{D}$	RS-232, Empfangsdaten	E	negativ	
79	$\bar{T}x\bar{D}$	RS-232, Sendedaten	A	negativ	
80					
81					
82					
83					
84					
85					
86					
87					
88					
89					
90					
91					
92					
93					
94					
95	$\bar{Z}C\bar{A}$	Zähler A = 0	A	negativ	
96	$\bar{Z}C\bar{B}$	Zähler B = 0	A	negativ	

---

**positive Logik:** 1=H, 0=L      **in Klammern:** Pegel kann durch Software bestimmt werden.  
**negative Logik:** 1=L, 0=H  
**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

## 1.7.2 Stecker B

### Reihe a

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	+ 12 V	Versorgungsspannung			
4	0 V	Masserückleitung, + 12 V			
5	- 12 V	Versorgungsspannung			
6	0 V	Masserückleitung, - 12 V			
7					
8					
9	$\overline{ADR}^0$	Adreßbit 0	A	negativ	
10	$\overline{ADR}^1$	Adreßbit 1	A	negativ	
11	$\overline{ADR}^2$	Adreßbit 2	A	negativ	
12	$\overline{ADR}^3$	Adreßbit 3	A	negativ	
13	$\overline{ADR}^4$	Adreßbit 4	A	negativ	
14	$\overline{ADR}^5$	Adreßbit 5	A	negativ	
15	$\overline{ADR}^6$	Adreßbit 6	A	negativ	
16	$\overline{ADR}^7$	Adreßbit 7	A	negativ	
17	$\overline{ADR}^8$	Adreßbit 8	A	negativ	TS
18	$\overline{ADR}^9$	Adreßbit 9	A	negativ	TS
19	$\overline{ADR}^{10}$	Adreßbit 10	A	negativ	TS
20	$\overline{ADR}^{11}$	Adreßbit 11	A	negativ	TS
21	$\overline{ADR}^{12}$	Adreßbit 12	A	negativ	TS
22	$\overline{ADR}^{13}$	Adreßbit 13	A	negativ	TS
23	$\overline{ADR}^{14}$	Adreßbit 14	A	negativ	TS
24	$\overline{ADR}^{15}$	Adreßbit 15	A	negativ	TS
25	$\overline{DAT}^0$	Datenbit 0	B	negativ	TS
26	$\overline{DAT}^1$	Datenbit 1	B	negativ	TS
27	$\overline{DAT}^2$	Datenbit 2	B	negativ	TS
28	$\overline{DAT}^3$	Datenbit 3	B	negativ	TS
29	$\overline{DAT}^4$	Datenbit 4	B	negativ	TS
30	$\overline{DAT}^5$	Datenbit 5	B	negativ	TS
31	$\overline{DAT}^6$	Datenbit 6	B	negativ	TS
32	$\overline{DAT}^7$	Datenbit 7	B	negativ	TS

positive Logik: 1=H, 0=L

Treiber: TS = tri state

negative Logik: 1=L, 0=H

OC = open collector

Richtung: E = Eingang, A = Ausgang, B = bidirektional

Reihe c

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik Treiber
65	+ 5 V	Versorgungsspannung		
66	0 V	Masserückleitung, + 5 V		
67	+ 12 V	Versorgungsspannung		
68	0 V	Masserückleitung, + 12 V		
69	- 12 V	Versorgungsspannung		
70	0 V	Masserückleitung, - 12 V		
71				
72				
73	ADR16	Adreßbit 16	A	positiv
74	ADR17	Adreßbit 17	A	positiv
75	ADR18	Adreßbit 18	A	positiv
76	ADR19	Adreßbit 19	A	positiv
77				
78				
79	$\overline{VERR}$	Speicherverriegelung	A	negativ
80				
81	M1	Kenner für Maschinenzklus 1	A	positiv
82	$\overline{\phi}$	Systemtakt	A	negativ
83	MREQ	Speicheranforderung	A	positiv
84	RFSH	Speicher-Auffrischsignal	A	positiv
85	RDSP	Speicher-Lesesynchronisation	A	positiv
86	WR	Schreibsynchronisation	A	positiv
87	IORQ	E-A-Tor-Anforderung	A	positiv
88	HALT	Meldung: CPU ist angehalten	A	positiv
89	SR/ $\overline{KOMBI}$	Umschaltung Speicher-Kombi.	A	positiv
90				
91				
92				
93				
94				
95				
96				

-----  
**positive Logik:** 1=H, 0=L

**negative Logik:** 1=L, 0=H

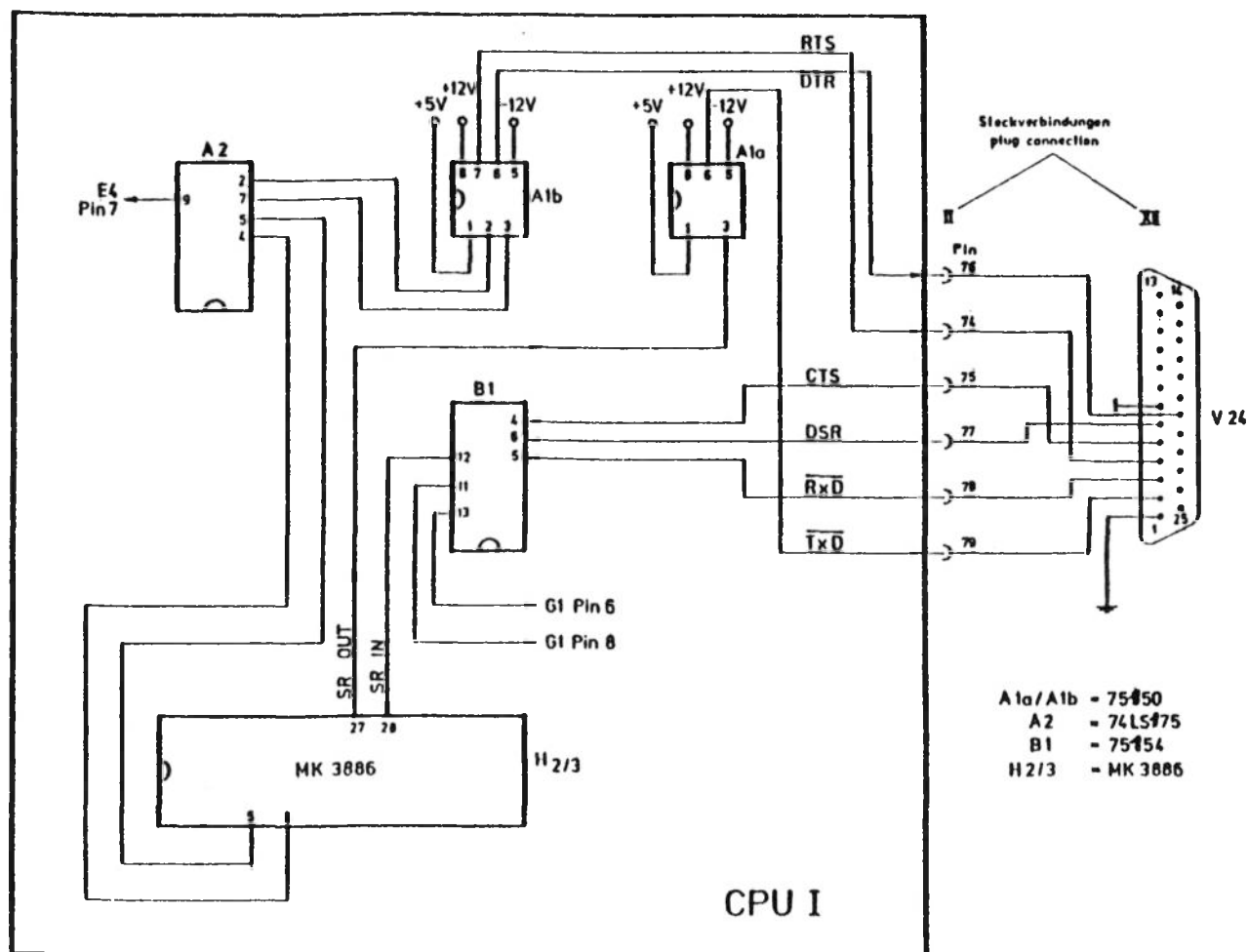
**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

### 1.7.3 Serielle Standardschnittstelle

Dies ist der auf der Gehäuserückseite unter dem Monitoranschluß gelegene D-Stecker. Er ist als RS-232C-"Master" geschaltet und weist folgende Belegung auf:

Anschluß	Bezeichnung	Bedeutung	Richtung	Logik	Treiber
1	0 V	Schutzerde			
2	$\overline{\text{TxD}}$	RS-232, Sendedaten	A	negativ	
3	$\overline{\text{RxD}}$	RS-232, Empfangsdaten	E	negativ	
4	RTS	RS-232, Sendeanforderung	A	(positiv)	
5	CTS	RS-232, Bestätigung vom Sender	E	(positiv)	
6	DSR	RS-232, Bereitschaftsbestätig.	E	(positiv)	
7	0 V	Signalerde (Masseverbindung)			
20	DTR	RS-232, Bereitschaftsmeldung	A	(positiv)	

positive Logik: 1=H, 0=L in Klammern: Pegel kann durch Software bestimmt werden.  
 negative Logik: 1=L, 0=H  
 Richtung: E = Eingang, A = Ausgang, B = bidirektional





## 1.8 Technische Daten

**Stromaufnahme:**

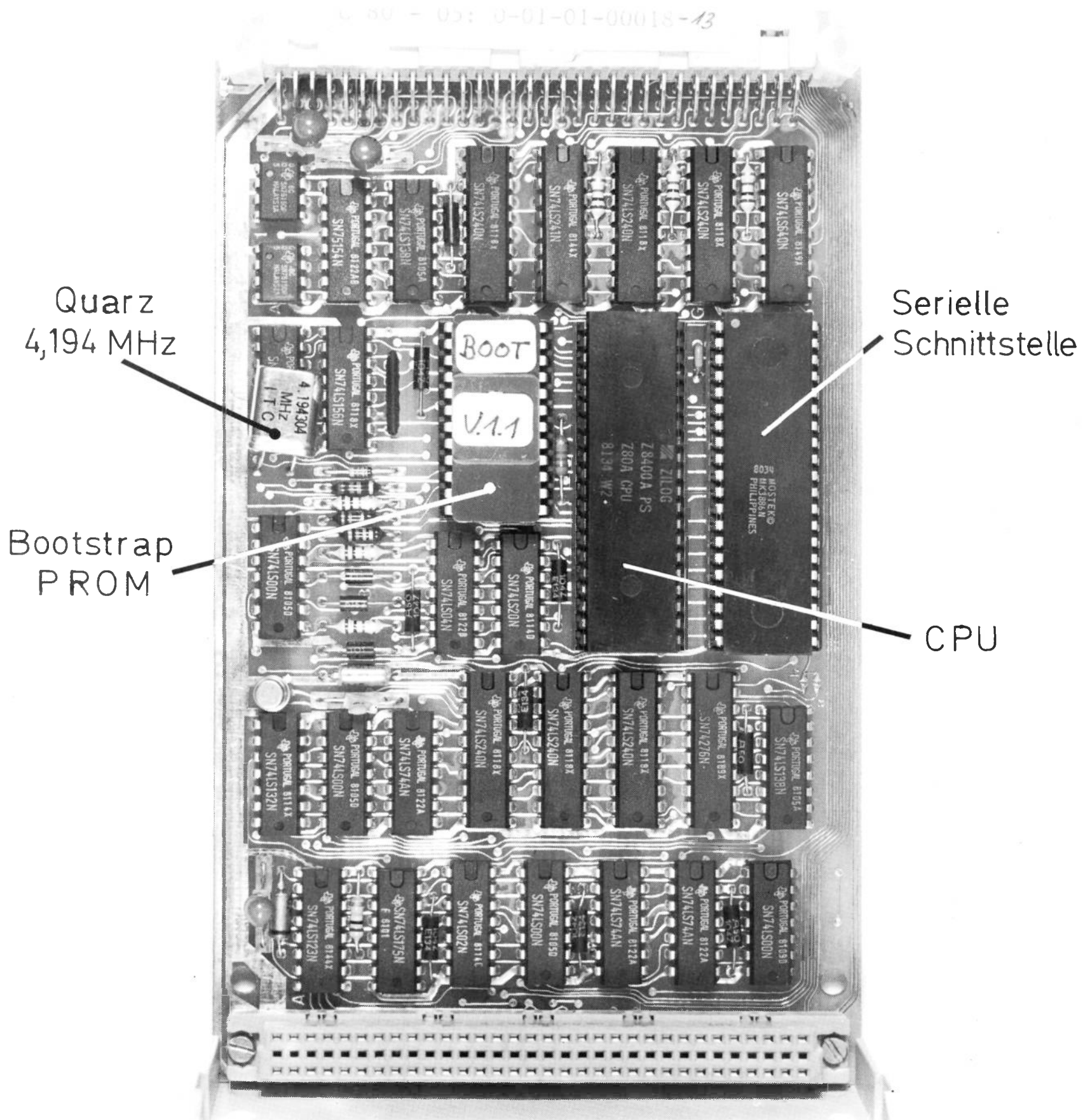
+ 5 V	$\pm 5\%$	$\leq 660$ mA
+ 12 V	$\pm 5\%$	$\leq 17$ mA
- 12 V	$\pm 5\%$	$\leq 16$ mA

### Systemtakt $\phi$ :

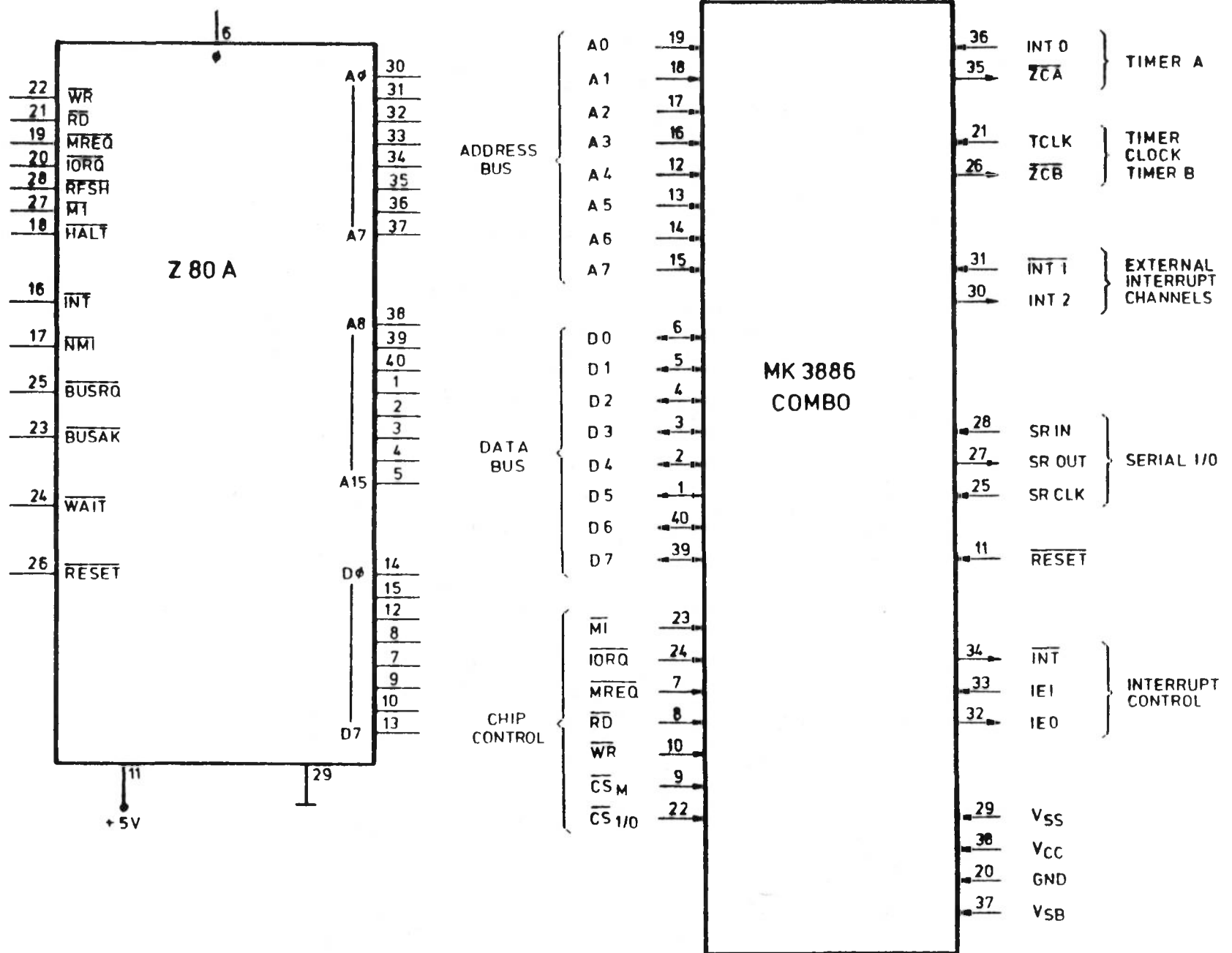
Oszillator: 4,194 MHz  $\pm 10^{-4}$   
Flankensteilheit:  $\leq 30$  ns

1.9 Bilder

1.9.1 Bestückung CPU I

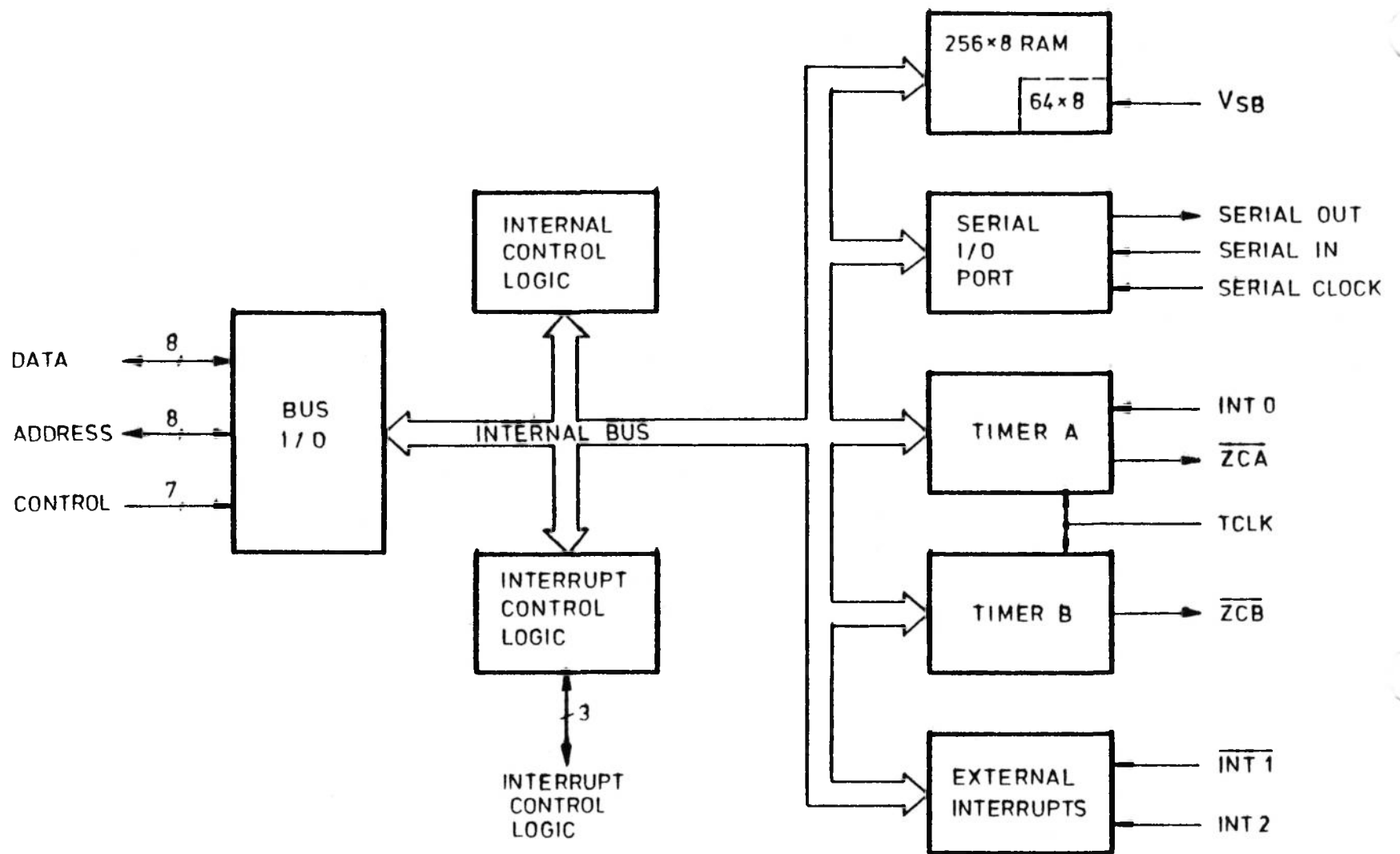


1.9.2 Anschlußbelegungen



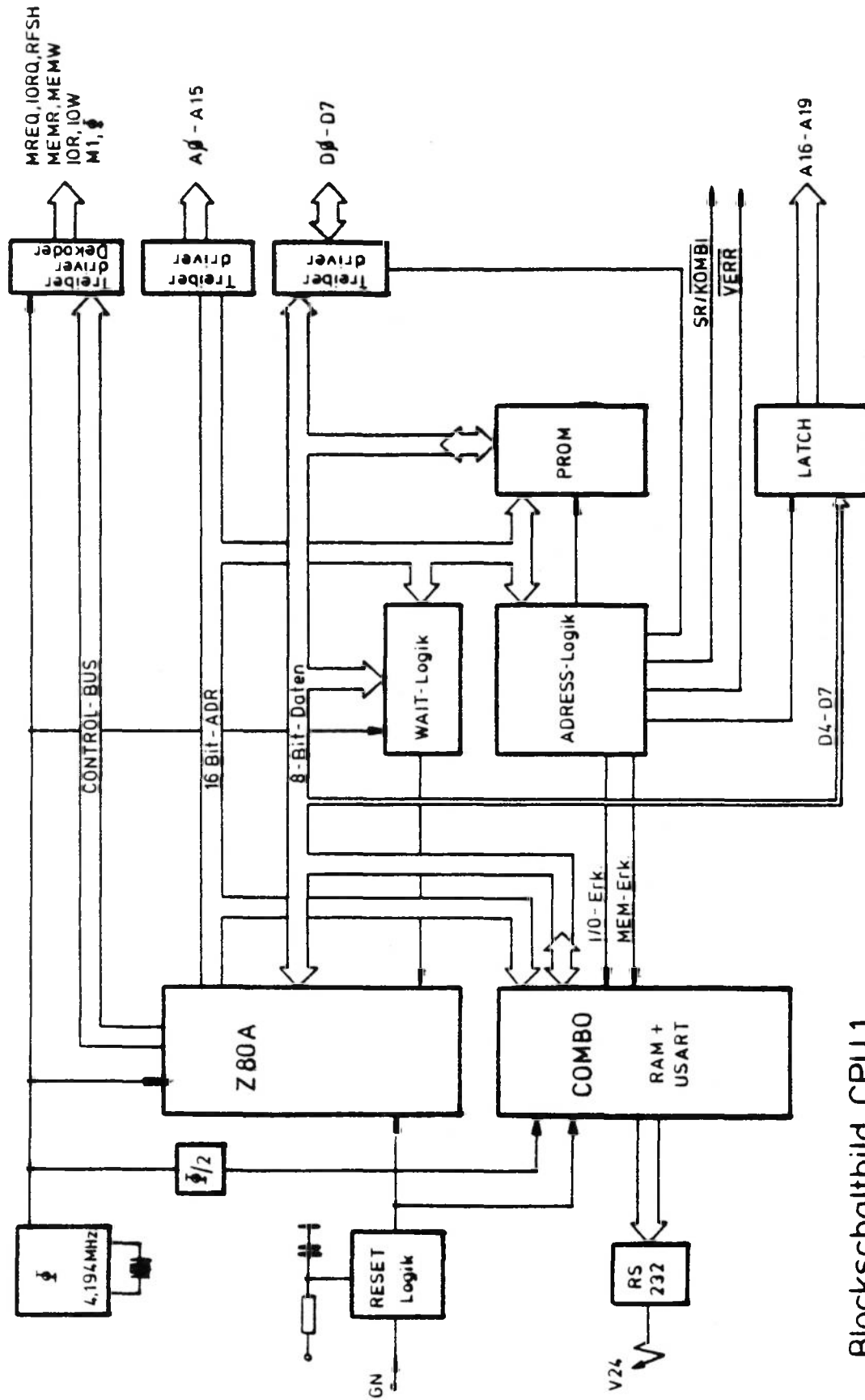
MK 3886 COMBO

1.9.3 Blockschaltbild des Kombinationsbausteins  
MK 3886

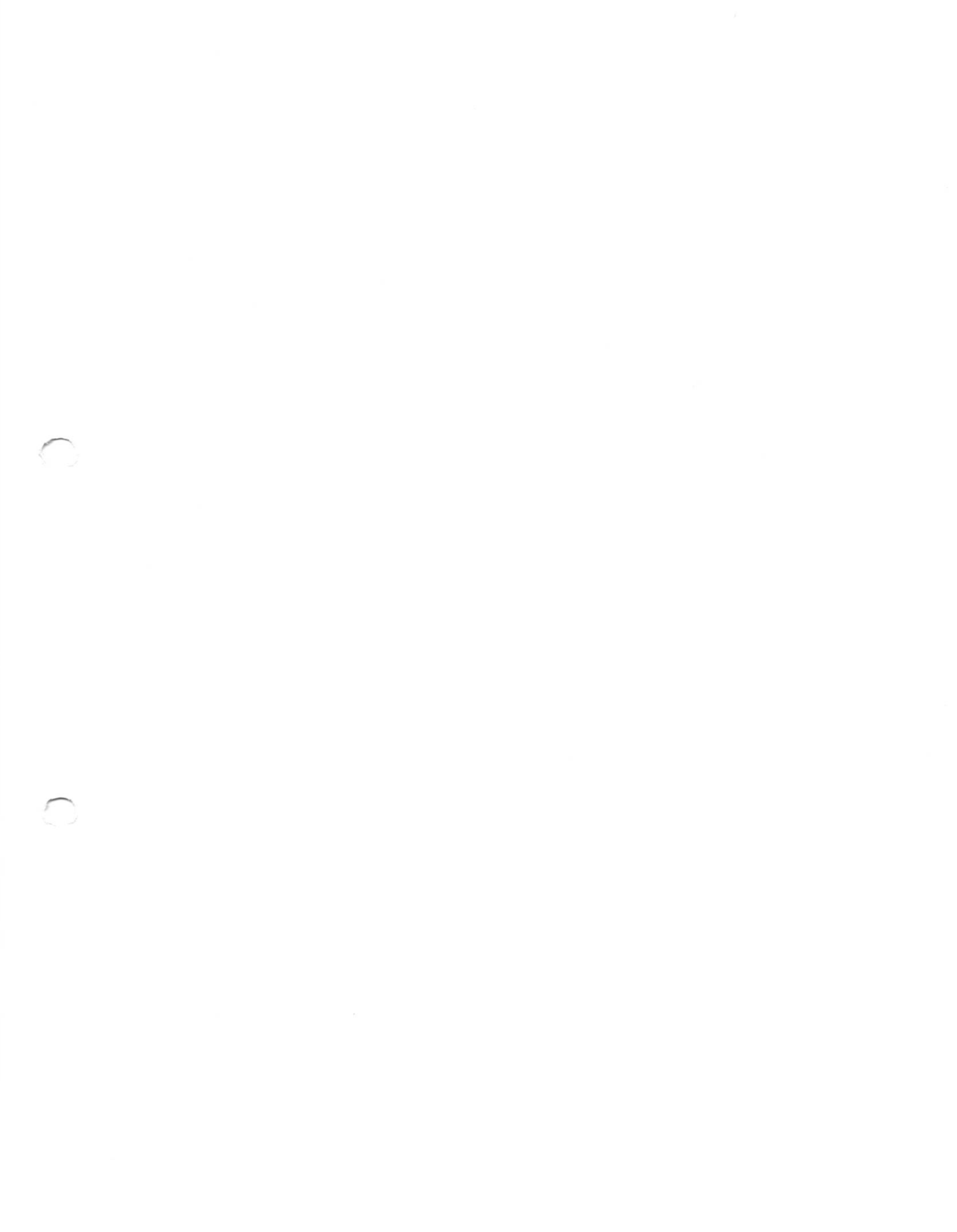


Blockschaltbild MK 3886  
Blockdiagram MK 3886

1.9.4 Blockschaltbild der CPU-Karte



Blockschaltbild CPU 1  
blockdiagram CPU 1



```
*****
*
*           I T T   3 0 3 0
*       S Y S T E M - U N D S E R V I C E H A N D B U C H
*       -----
*           A b s c h n i t t   4
*           S p e i c h e r
*
*****
```

## 1 Schreib-Lese-Speicher, 64 KBytes (64K-RAM)

### 1.0 Inhaltsverzeichnis

1.1 Konzept .....	4-1-2
1.2 Die CPU-Schnittstelle .....	4-1-3
1.3 Der Speicherbereich .....	4-1-4
1.3.1 Adressierung der Speicherbausteine .....	4-1-4
1.3.2 Auffrischen des Speicherinhalts (refresh) .....	4-1-5
1.4 Die Adressierlogik .....	4-1-6
1.4.1 Speicheraktivierung .....	4-1-6
1.4.2 $\overline{R\bar{A}\bar{S}}$ und $\overline{C\bar{A}\bar{S}}$ .....	4-1-6
1.5 Die Datenbustreiber .....	4-1-8
1.6 Auffrischen des Speicherinhalts (refresh) .....	4-1-9
1.7 Erzeugung der Versorgung mit - 5 V .....	4-1-10
1.8 Steckerbelegung .....	4-1-11
1.9 Technische Daten .....	4-1-13
1.10 Bilder .....	4-1-14
1.10.1 Bestückung .....	4-1-14
1.10.2 Blockschalbild .....	4-1-15

## 1.1 Konzept

Die Karte 64K-RAM ist ein Schreib-Lese-Speicher mit einer Kapazität von 64 KBytes zu je 8 Bit. Er wird über Stecker B direkt auf die CPU-Karte aufgesteckt und von dieser angesteuert.

Die Karte umfaßt die folgenden **Funktionsblöcke**:

- vier Speicherbänke zu 16 KBytes, aufgebaut aus dynamischen Speicherbausteinen mit einer Organisation von 16 K x 1 Bit und 200 ns Zugriffszeit
- Adressierlogik zur RAS- und CAS-Adressierung
- Auffrischlogik
- bidirektionale Datenbustreiber
- Erzeugung der Versorgungsspannung von - 5 V für die Speicherbausteine



## 1.2 Die CPU-Schnittstelle

Die Schnittstelle zur CPU-Karte ist auf einen Stecker gelegt, bei dem nur die a- und c-Reihe belegt sind. Sie trägt folgende Signale.

- **a-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen  
Adreßbus 16 Bit (invertiert):  $\bar{A}D\bar{R}0 \dots \bar{A}D\bar{R}15$   
bidirektionaler Datenbus 8 Bit (invertiert):  $\bar{D}A\bar{T}0 \dots \bar{D}A\bar{T}7$

- **c-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen  
Steuersignale  $\bar{\phi}$ , M1,  
MREQ,  $\bar{V}E\bar{R}\bar{R}$ , SR/ $\bar{K}o\bar{m}b\bar{i}$ ,  
RDSP, WR

Signale zum Prüfen der Speicherfunktion (Prüfstellen)

PSt - 5 V,  
PSt  $\bar{R}A\bar{S}0$ , PSt  $\bar{R}A\bar{S}1$ , PSt  $\bar{R}A\bar{S}2$ , PSt  $\bar{R}A\bar{S}3$ ,  
PSt  $\bar{C}A\bar{S}$

### 1.3 Der Speicherbereich

Der Speicherbereich setzt sich aus vier Bänken zu je 16 KBytes zusammen. Jede Bank ist aus 8 dynamischen Speicherbausteinen mit einer Organisation von 16 K x 1 Bit aufgebaut.

#### 1.3.1 Adressierung der Speicherbausteine

Diese Speicherbausteine besitzen ein 16-poliges Gehäuse mit 4 Versorgungsanschlüssen (0, + 12 V, + 5 V, - 5 V), je einem Datenein- und -ausgang ( $D_{in}$ ,  $D_{out}$ ) und einer Schreibaktivierung  $\bar{WE}$ . Es verbleiben noch 9 Anschlüsse, weshalb die zur Adressierung von 16 KBytes notwendigen 14 Adreßbits im Multiplexverfahren zu je 7 Bits angelegt werden müssen.

Bausteinintern dient die erste Hälfte zur "Zeilenadressierung" (row addresses), die zweite zur "Spaltenadressierung" (column addresses) der Speichermatrix. Diese Adreßbestandteile werden im Baustein in besonderen Registern festgehalten, wozu es je eines Strobesignals bedarf. So stehen zur Adressierung (und Aktivierung) zwei Anschlüsse zur Verfügung, der Zeilenadrestrobe  $\bar{RAS}$  (row address strobe) und der Spaltenadrestrobe  $\bar{CAS}$  (column address strobe).

Der Zugriff auf den Baustein erfolgt so im wesentlichen in drei Schritten:

- Die erste 7-Bit-Adreßhälfte wird angelegt und dem Baustein mit einem Nullpegel am  $\bar{RAS}$ -Eingang mitgeteilt. Dies bewirkt zweierlei. Zum einen wird diese Adreßhälfte intern zwischengespeichert, und zum anderen wird der Baustein aktiviert. D. h. der  $\bar{RAS}$ -Eingang wirkt zugleich als Chip-Select.
- Ist die Information zuverlässig übernommen, dann kann die zweite Adreßhälfte angelegt, und der Spaltenadrestrobe  $\bar{CAS}$  durch Nullpegel aktiviert werden. Durch  $\bar{CAS}$  wird einerseits die zweite 7-Bit-Adreßhälfte im Baustein festgehalten. Andererseits aktiviert  $\bar{CAS}$  (bei Leseoperationen) den Ausgabetreiber für den  $D_{out}$ -Anschluß bzw. (bei Schreiboperationen) den Eingangspuffer am  $D_{in}$ -Anschluß.
- Nach Verstreichen der (ab  $\bar{RAS}$  zählenden) Zugriffszeit können die beiden Strobesignale wieder entfernt werden. Der Baustein ist damit deaktiviert und arbeitet im Ruhebetrieb (stand-by mode).

Wichtig ist dabei, daß zur Arbeit beide Strobes aktiv sein müssen, und daß die Reihenfolge - erst  $\overline{R\overline{A}\overline{S}}$ , dann  $\overline{C\overline{A}\overline{S}}$  - eingehalten wird.

### 1.3.2 Auffrischen des Speicherinhalts (refresh)

Da in dynamischen Speicherbausteinen die Information in Form elektrischer Ladungen festgehalten wird, die im Lauf der Zeit abfließen, muß sie regelmäßig wieder aufgefrischt werden. Dieses Auffrischen geschieht bei jedem Speicherzugriff für jeweils eine ganze Zeile auf einmal. Um den Speicherinhalt zu bewahren müssen daher innerhalb einer angemessenen Zeit (hier innerhalb von 2 Millisekunden) alle 128 Zeilenadressen einmal anliegen.

Der Auffrischvorgang ist eine rein interne Angelegenheit. Deshalb genügt es, nur den Zeilenstrobe  $\overline{R\overline{A}\overline{S}}$  zu aktivieren. Hier wird die zu der anlegenden 7-Bit-Adresse gehörende Zeile der internen Speichermatrix aufgefrischt. Der (nur durch  $\overline{C\overline{A}\overline{S}}$  aktivierte) Ausgabetreiber bleibt dabei abgeschaltet, was die Schaltungsauslegung des Systems wesentlich vereinfacht.

## 1.4 Die Adressierlogik

Die Adressierlogik hat damit folgende Aufgaben:

- Sie muß eine der vier Speicherbänke auswählen.
- Sie muß die verbleibenden 14 Bits von  $\overline{ADR}_0$  bis  $\overline{ADR}_{13}$  in zwei 7-Bit-Hälften aufteilen, nacheinander an die Speicherchips anlegen und die  $\overline{RAS}$ - bzw.  $\overline{CAS}$ -Signale erzeugen.
- Ferner ist eine eventuelle Verriegelung der Speicherkarte durch das Signal  $\overline{VERR}$  zu beachten.

### 1.4.1 Speicheraktivierung

Die Speicherbänke werden abhängig von den beiden höchstwertigen Adreßbits  $\overline{ADR}_{14}$  und  $\overline{ADR}_{15}$  durch einen Dekoder ausgewählt. Er liefert die Grundinformation, welche der vier Bänke durch  $\overline{RAS}$  aktiviert werden kann. Dieser Dekoder ist gegebenenfalls durch das Sperrsignal  $\overline{VERR}$  verriegelt und damit ein Zugriff auf die Karte unterbunden.

Eine weitere Verriegelung erfolgt beim Schreibzugriff durch das WR-Signal, dem nur stattgegeben wird, wenn zugleich das Auswahl-signal  $SR/\overline{Kombi}$  auf H-Pegel liegt, wenn also ausdrücklich der Speicher adressiert wird.

### 1.4.2 $\overline{RAS}$ und $\overline{CAS}$

Die niederwertigen vierzehn Adreßbits von  $\overline{ADR}_0$  bis  $\overline{ADR}_{13}$  werden über Multiplexer in eine höherwertige und eine niederwertige 7-Bit-Hälfte aufgespalten, die nacheinander auf die sieben Adreßleitungen der Speicherbänke gelegt werden.

Das jeweilige  $\overline{RAS}$ -Signal wird durch Anlegen der Speicheranforderung MREQ an die Karte erzeugt. Die Umschaltung auf die zweite Adreßhälfte und die Aktivierung von  $\overline{CAS}$  erfolgt in Abhängigkeit vom Systemtakt  $\phi$ .

Ist MREQ aktiv (H-Pegel), dann werden mit der nächsten abfallenden Flanke des (invertierten) Systemtakts  $\overline{\phi}$  die Eingangsmultiplexer auf die zweite Adreßhälfte umgeschaltet. Mit der folgenden (ansteigenden)  $\overline{\phi}$ -Systemtaktflanke wird dann ein für alle Speicherbänke wirkendes  $\overline{CAS}$ -Signal erzeugt.

Die Signale  $\bar{R}\bar{A}\bar{S}\bar{0}$  bis  $\bar{R}\bar{A}\bar{S}\bar{3}$  und  $\bar{C}\bar{A}\bar{S}$  sind am Kartenstecker auf Reihe c zu Testzwecken verfügbar (siehe die Steckerbelegung im Anhang).

### **1.5 Die Datenbustreiber**

Bei einer Leseoperation werden noch die Datenbustreiber in Richtung auf den Systembus umgeschaltet, wenn die allgemeine Speicheranforderung MREQ und das Lesesignal RDSP aktiv sind. In allen anderen Fällen sind diese Treiber auf Empfang von Systembusdaten geschaltet und somit gegenüber dem System in hochohmigem Zustand.

### 1.6 Auffrischen des Speicherinhalts (refresh)

Das Auffrischen des Speicherinhalts wird durch die Kombination der Signale RFSH und MREQ im jeweils ersten Befehlszyklus (dritter Takt im M1-Zyklus) ausgelöst. Wie oben dargestellt, sollte hierbei lediglich die Zeilenadresse über RAS übernommen werden. Weiter sind alle vier Speicherbänke auf einmal zu erfassen.

Das Auffrischsignal RFSH aktiviert alle vier  $\overline{RAS}$ -Signale und damit alle vier Speicherbänke. Die jeweils aufzufrischende Zeile in der Speichermatrix der Bausteine wird dabei von den niederwertigen sieben Bits der gerade anliegenden Adresse (d.h. von  $\overline{ADR}_0$  bis  $\overline{ADR}_6$ ) bestimmt.

In den ersten beiden Takten des Befehlszyklus wurde das M1-Signal auf der Karte festgehalten. Durch diese Information wird jetzt im Auffrischvorgang sowohl die Umschaltung der Adreßmultipler als auch das  $\overline{CAS}$ -Signal gesperrt, so daß der Auffrischvorgang wie verlangt allein mit  $\overline{RAS}$  ausgeführt werden kann.

### **1.7 Erzeugung der Versorgung mit - 5 V**

Die für die Speicherbausteine benötigte -5V-Versorgung wird aus der -12V-Kartenversorgung gewonnen. Sie ist auf Anschluß 72 am Kartenstecker herausgeführt und kann dort überprüft werden.



## 1.8 Steckerbelegung

### Reihe a

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	+ 12 V	Versorgungsspannung			
4	0 V	Masserückleitung, + 12 V			
5	- 12 V	Versorgungsspannung			
6	0 V	Masserückleitung, - 12 V			
7					
8					
9	$\overline{ADR}^0$	Adreßbit 0	E	negativ	
10	$\overline{ADR}^1$	Adreßbit 1	E	negativ	
11	$\overline{ADR}^2$	Adreßbit 2	E	negativ	
12	$\overline{ADR}^3$	Adreßbit 3	E	negativ	
13	$\overline{ADR}^4$	Adreßbit 4	E	negativ	
14	$\overline{ADR}^5$	Adreßbit 5	E	negativ	
15	$\overline{ADR}^6$	Adreßbit 6	E	negativ	
16	$\overline{ADR}^7$	Adreßbit 7	E	negativ	
17	$\overline{ADR}^8$	Adreßbit 8	E	negativ	TS
18	$\overline{ADR}^9$	Adreßbit 9	E	negativ	TS
19	$\overline{ADR}^{10}$	Adreßbit 10	E	negativ	TS
20	$\overline{ADR}^{11}$	Adreßbit 11	E	negativ	TS
21	$\overline{ADR}^{12}$	Adreßbit 12	E	negativ	TS
22	$\overline{ADR}^{13}$	Adreßbit 13	E	negativ	TS
23	$\overline{ADR}^{14}$	Adreßbit 14	E	negativ	TS
24	$\overline{ADR}^{15}$	Adreßbit 15	E	negativ	TS
25	$\overline{DAT}^0$	Datenbit 0	B	negativ	TS
26	$\overline{DAT}^1$	Datenbit 1	B	negativ	TS
27	$\overline{DAT}^2$	Datenbit 2	B	negativ	TS
28	$\overline{DAT}^3$	Datenbit 3	B	negativ	TS
29	$\overline{DAT}^4$	Datenbit 4	B	negativ	TS
30	$\overline{DAT}^5$	Datenbit 5	B	negativ	TS
31	$\overline{DAT}^6$	Datenbit 6	B	negativ	TS
32	$\overline{DAT}^7$	Datenbit 7	B	negativ	TS

positive Logik: 1=H, 0=L

negative Logik: 1=L, 0=H

Richtung: E = Eingang, A = Ausgang, B = bidirektional

Treiber: TS = tri state

OC = open collector

Reihe c

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
65	+ 5 V	Versorgungsspannung			
66	0 V	Masserückleitung, + 5 V			
67	+ 12 V	Versorgungsspannung			
68	0 V	Masserückleitung, + 12 V			
69	- 12 V	Versorgungsspannung			
70	0 V	Masserückleitung, - 12 V			
71					
72					
73	ADR16	Adreßbit 16	E	positiv	
74	ADR17	Adreßbit 17	E	positiv	
75	ADR18	Adreßbit 18	E	positiv	
76	ADR19	Adreßbit 19	E	positiv	
77					
78					
79	$\overline{VERR}$	Speicherverriegelung	E	negativ	
80					
81	M1	Kenner für Maschinenzyklus 1	E	positiv	
82	$\overline{\phi}$	Systemtakt	E	negativ	
83	MREQ	Speicheranforderung	E	positiv	
84	RFSH	Speicher-Auffrischsignal	E	positiv	
85	RDSP	Speicher-Lesesynchronisation	E	positiv	
86	WR	Schreibsynchronisation	E	positiv	
87	IORQ	E-A-Tor-Anforderung	E	positiv	
88	HALT	Meldung: CPU ist angehalten	E	positiv	
89	SR/ $\overline{KOMB\bar{I}}$	Umschaltung Speicher-Kombi.	E	positiv	
90					
91					
92					
93					
94					
95					
96					

-----  
**positive Logik:** 1=H, 0=L

**Treiber:** TS = tri state

**negative Logik:** 1=L, 0=H

OC = open collector

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

## 1.9 Technische Daten

### Stromaufnahme

+ 5 V	$\frac{+}{-}$	5 %:	$\leq$	300 mA
+ 12 V	$\frac{+}{-}$	5 %:	$\leq$	200 mA
- 12 V	$\frac{+}{-}$	20 %:	$\leq$	50 mA

(zur Erzeugung von - 5 V)

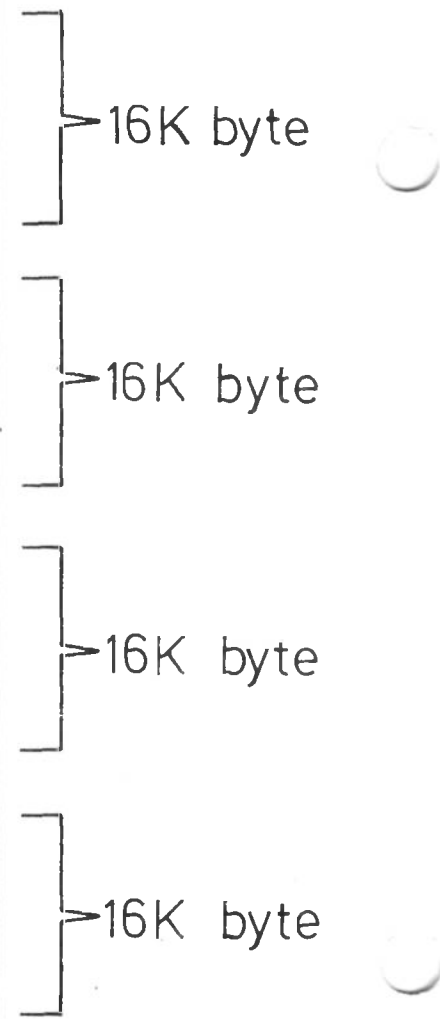
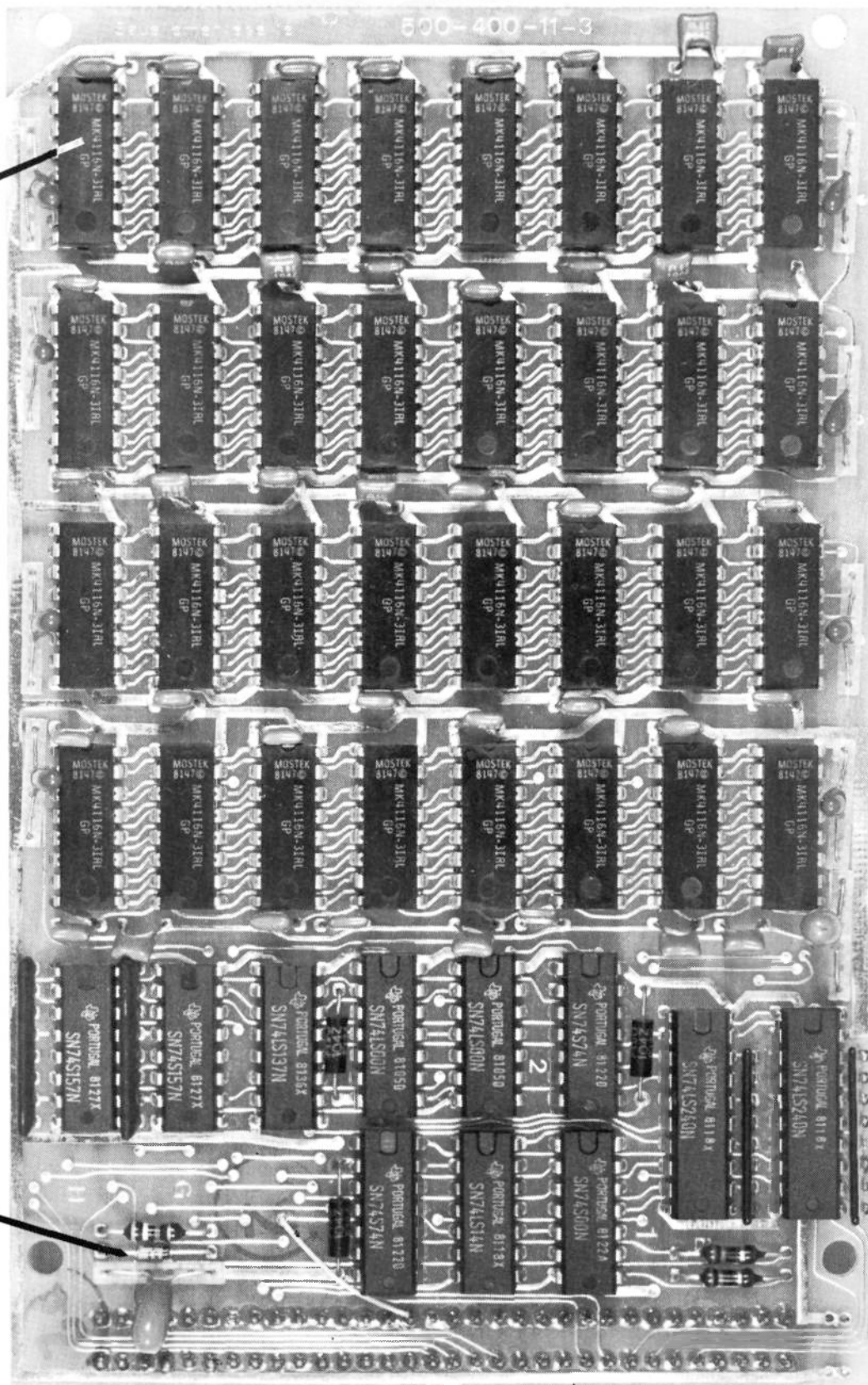
1.10 Bilder

1.10.1 Bestückung

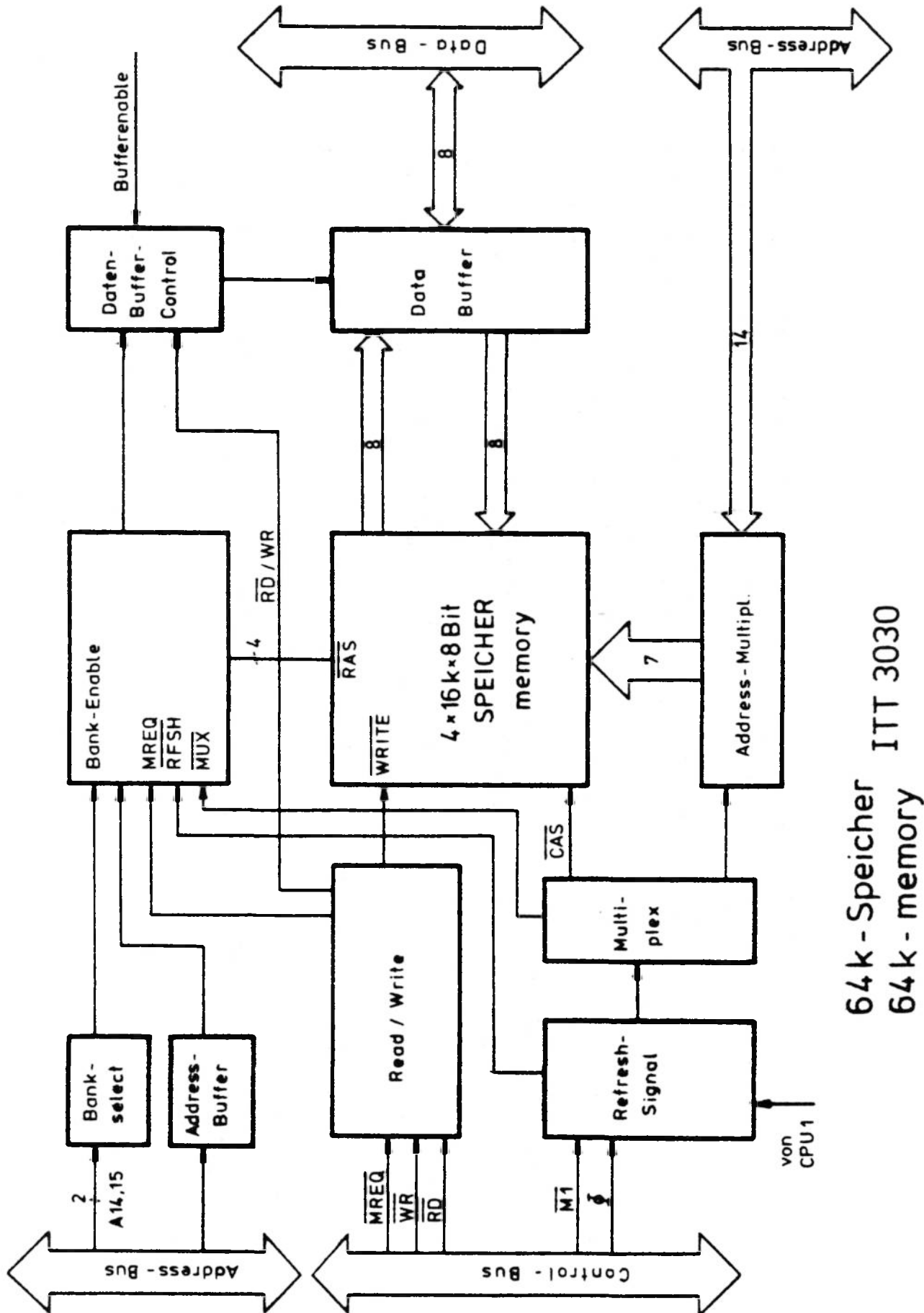
64 KB-Speicherzusatz

16K RAM  
1Bit-  
organisiert

-5V  
Erzeugung



1.10.2 Blockschaltbild



64 k - Speicher  
 64 k - memory  
 ITT 3030



```
*****
*
*           I T T   3 0 3 0
*   SYSTEM- UND SERVICEHANDBUCH
*   -----
*           Abschnitt 5
*   Terminalschnittstellen
*
*****
```

## 1 Video-Tastatur-Schnittstelle

### 1.1 Kombinationskarte I

#### 1.1.0 Inhalt

1.1.1 Konzept .....	5-1.1-3
1.1.2 Schnittstellen .....	5-1.1-4
1.1.2.1 Die Systembusschnittstelle .....	5-1.1-4
1.1.2.2 Die Bildschirmschnittstelle .....	5-1.1-4
1.1.2.3 Die Tastaturschnittstelle .....	5-1.1-5
1.1.3 Die Bildschirmsteuerung .....	5-1.1-6
1.1.3.1 Der Video-Steuerbaustein .....	5-1.1-7
1.1.3.1.1 Adressen .....	5-1.1-7
1.1.3.1.2 Programmierung des Steuerbausteins .....	5-1.1-8
1.1.3.1.2.1 Besonderheiten .....	5-1.1-10
1.1.3.1.2.2 Beispiel (24x80-Anzeige) .....	5-1.1-11
1.1.3.2 Der Bildwiederholpeicher .....	5-1.1-12
1.1.3.2.1 Adressierung des Bildwiederholpeichers .	5-1.1-13
1.1.3.2.1.1 Systemzugriff auf den Speicher .....	5-1.1-14
1.1.3.2.1.2 Interner Speicherzugriff .....	5-1.1-15
1.1.3.2.2 Synchronisation des Speicherzugriffs ....	5-1.1-16
1.1.3.2.3 Festlegen der Zahl der Abtastzeilen ....	5-1.1-17
1.1.3.2.4 Lupe .....	5-1.1-17
1.1.3.3 Der Zeichengenerator .....	5-1.1-19
1.1.3.3.1 Organisation des Zeichengenerators .....	5-1.1-19
1.1.3.3.2 Umsetzung des Zeichenkodes .....	5-1.1-19
1.1.3.3.3 Zeichensätze .....	5-1.1-20
1.1.3.3.3.1 Graphikzeichen .....	5-1.1-20
1.1.3.3.3.2 Standard-Zeichensätze .....	5-1.1-23
1.1.3.4 Erzeugung des Videosignals .....	5-1.1-24

1.1.4 Die Tastatursteuerung .....	5-1.1-25
1.1.4.1 Funktionsprinzip .....	5-1.1-25
1.1.4.2 Entprellung .....	5-1.1-26
1.1.4.3 Zeichenpuffer .....	5-1.1-27
1.1.4.4 Mehrfachbetätigung (rollover) .....	5-1.1-27
1.1.4.5 Programmierung des 8278-Bausteins .....	5-1.1-28
1.1.4.5.1 Die 8278-Register .....	5-1.1-28
1.1.4.5.2 8278-Befehle .....	5-1.1-28
1.1.4.5.2.1 Betriebsart des 8278-Bausteins setzen	5-1.1-29
1.1.4.5.2.2 FIFO lesen .....	5-1.1-30
1.1.4.5.2.3 Löschbefehl .....	5-1.1-30
1.1.4.5.3 Das Statusregister .....	5-1.1-30
1.1.4.5.4 Betriebshinweise .....	5-1.1-31
1.1.4.6 Fehlerbehandlung .....	5-1.1-32
1.1.5 Signaleinheiten .....	5-1.1-34
1.1.6 Der Kartenstatus .....	5-1.1-35
1.1.7 Technische Daten .....	5-1.1-36
1.1.8 Zusammenfassung der Adressen und Befehle .....	5-1.1-37
1.1.8.1 Bildschirmsteuerung .....	5-1.1-37
1.1.8.2 Tastatursteuerung .....	5-1.1-46
1.1.8.3 Allgemeine Karteneigenschaften .....	5-1.1-50
1.1.9 Steckerbelegung .....	5-1.1-51
1.1.10 Bilder .....	5-1.1-54
1.1.10.1 Bestückung .....	5-1.1-54
1.1.10.2 Anschlußbelegung .....	5-1.1-55
1.1.10.3 Innenschaltung der Tastatursteuerung .....	5-1.1-56
1.1.10.4 Blockschaltbild .....	5-1.1-57



### 1.1.1 Konzept

Die Video-Tastatur-Kombination faßt zwei Baugruppen zusammen: eine Bildschirmsteuerung und eine Tastatursteuerung. Diese Baugruppen liefern verallgemeinerte Signale bzw. werten verallgemeinerte Signale aus, die noch von Adapterkarten an die jeweils betriebene Peripherie angepaßt werden müssen (siehe Abschnitt 5.2 und 5.3).

Die Kombinationskarte umfaßt im einzelnen:

- die Bildschirmsteuerung mit
  - = programmierbarer Zeitsteuerung (VTAC 5027 bzw. TMS 9927),
  - = Bildwiederholtspeicher mit Ausgabelatch,
  - = Adressierlogik für den Bildspeicher,
  - = Zeichengenerator und
  - = Videosignalerzeugung mit Taktaufbereitung,
- die Tastatursteuerung auf Basis des Steuerbausteins 8278,
- die Ansteuerung von Signaleinheiten (Hupe, Tastatur-LED),
- die allgemeine Adreßlogik.

## 1.1.2 Schnittstellen

Über den Systemstecker werden drei Schnittstellen herausgeführt:

- der Systembus (Reihe a und b),
- die Bildschirmsteuerung (Reihe c) und
- die Tastatursteuerung (ebenfalls Reihe c).

Die Signalleitungen auf Reihe c stehen auf dem Verdrahtungsfeld mit den Steckplätzen VII (ganz links: Tastaturanpassung) bzw. XI (ganz rechts: Bildschirmanpassung) in Verbindung.

### 1.1.2.1 Die Systembusschnittstelle

Die Schnittstelle zum Systembus trägt die Versorgungsspannungen, Adreß- und Datensignale sowie einige Steuersignale des Systems. Im einzelnen sind das auf der

- **a-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen

Adreßbus 16 Bit (invertiert):  $\overline{ADR}_0 \dots \overline{ADR}_{15}$

bidirektionaler Datenbus 8 Bit (invertiert):  $\overline{DAT}_0 \dots \overline{DAT}_7$

Kartenfreigabe:  $SR/\overline{Kombi}$

Kennung für einen Befehlsübernahmezyklus: M1

- **b-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen

Synchronisiersignale:  $\overline{MEMR}$ ,  $\overline{MEMW}$ ,  $\overline{MREQ}$ ,  $\overline{IOR}$ ,  $\overline{IOW}$ ,  $\overline{IORQ}$

systeminternes Rücksetzsignal:  $\overline{RST}$

Unterbrechungsvektor:  $\overline{INT}_0$

Die Kombinationskarte belegt im Speicherraum des System den 4 KBytes umfassenden Bereich von 3000H bis 3FFFH. Dieser Speicherbereich wird jedoch nur auf ausdrückliche Anforderung über das Signal  $SR/\overline{Kombi}$  freigegeben. Dabei wird der entsprechende Teil des Systemspeichers abgeschaltet. Normalerweise ist dies nur auf Seite 8 des Gesamtspeichers möglich (siehe dazu die CPU-Beschreibung in Abschnitt 3).

### 1.1.2.2 Die Bildschirmschnittstelle

Die Bildschirmschnittstelle trägt zu- und abgehende Signale wie folgt:

- **zugehende Signale:**

Quarztakt zur Videosignalaufbereitung: QT

Statussignale zur Erkennung der Anpassung:  $\overline{STAT}_0$ ,  $\overline{STAT}_1$ ,  $\overline{STAT}_2$

**- abgehende Signale:**

Synchronisiersignale:  $\overline{HSYN}$ ,  $\overline{VSYN}$ ,  $\overline{CSYN}$

Bildsignal: Video

**1.1.2.3 Die Tastaturschnittstelle**

Über die Tastaturschnittstelle werden die folgenden Signale übertragen:

- Signale zur Abfrage der Tastaturmatrix: M0 ... M1
- Takt zur Tastaturabfrage: KCL
- Rückleitung von der Tastatur:  $\overline{RL}$
- Ansteuerung der Signalisiergeräte: LED, Hupe

### 1.1.3 Die Bildschirmsteuerung

Die Bildschirmsteuerung beinhaltet zwei Schnittstellen, eine zur CPU (über den Systembus) und eine zur Anpassungskarte an das jeweilige Ausgabegerät. Über die CPU-Schnittstelle kann die darzustellende Information (ASCII-kodiert) in den Bildwiederhol-speicher eingeschrieben und aus ihm bei Bedarf auch wieder ausgelesen werden. Bildschirmseitig bereitet ein hochintegrierter Steuerbaustein (VTAC 5027 oder TMS 9927) die im Bildspeicher festgehaltene Information mit Hilfe eines 128 Zeichenbilder enthaltenden Zeichengenerator-EPROMs auf und erzeugt daraus die Ansteuersignale für die Ausgabe des Bilds auf einem Rastersichtgerät (Synchronisations- und Bildsignale). Diese Signale werden dann noch von einer besonderen Karte an das jeweilige Ausgabegerät angepaßt (siehe Abschnitt 5.2 im Handbuch).

Die Bildgröße kann über den Steuerbaustein programmiert werden. Standardwerte sind 24 Zeilen zu 80 Zeichen, 16 Zeilen zu 32 Zeichen und 16 Zeilen zu 64 Zeichen. (Dies hängt auch von dem Videotakt QT ab, der vom Monitoradapter geliefert wird.) Die Zeichen selbst werden in einem Feld von 12 Zeilen zu 8 Punkten wiedergegeben; durch Verändern zweier Brücken kann dies auf der Karte umgestellt werden auf 16 Zeilen zu 8 Punkten. Die Zeichendarstellung erfolgt wahlweise hell auf dunklem Grund oder invertiert dazu dunkel auf hellem Grund. Dies wird durch das höchstwertige Bit der im Bildspeicher festgehaltenen Zeichen bestimmt (0 = normale, 1 = invertierte Darstellung).

### 1.1.3.1 Der Video-Steuerbaustein

Kern der Bildschirmsteuerung ist ein hochintegrierter Steuerbaustein vom Typ VTAC 5027 bzw. TMS 9927. Dieser Baustein führt drei Aufgaben aus:

- Er erzeugt den horizontalen und vertikalen **Synchronisationsimpuls** und ihre Kombination für das BAS-Signal. Zusätzlich werden die Austastlücken durch ein besonderes Blank-Signal gemeldet.
- Er erzeugt die Adressen zur Umsetzung der Information im Bildwiederholtspeicher in ein Videosignal, und zwar für das jeweils vom Elektronenstrahl anzuzeigende Zeichen
  - = die **Zeilennummer**,
  - = die **Nummer des Zeichens** in dieser Zeile und
  - = die **Nummer der Abtastzeile** in diesem Zeichen.
 Die beiden ersten Nummern bilden die Adresse für den Bildwiederholtspeicher, und die Abtastzeile bestimmt die Umsetzung des Zeichens in Punkte anhand der im Zeichengenerator-EPROM festgehaltenen Information (s.u.).
- Schließlich liefert er ein Signal, wenn im Zuge der Schirmabtastung die **Kursorposition** erreicht worden ist.  
**Diese Eigenschaft zur Kursorpositionierung wird auf der Kombinationskarte derzeit jedoch nicht genutzt.**

#### 1.1.3.1.1 Adressen

Alle **Funktionen des Steuerbausteins** sind über Ausgabebitore programmierbar. Sie befinden sich auf den Adressen 20H bis 2FH wie folgt:

Funktion	Tor	Z80-Befehl
Zeitsteuerung		
stoppen	2A	OUT 2AH
starten	2E	OUT 2EH
Steuerregister	20...26	OUT 20H ... OUT 26H
Schirmbild rollen	2B	OUT 2BH

(Die nicht angegebenen Adressen betreffen die auf der Kombinationskarte nicht genutzten Kursor- und automatischen Ladefunktionen des Steuerbausteins.)

### 1.1.3.1.2 Programmierung des Steuerbausteins

Stoppen und Starten der Zeitsteuerung sowie das Aufwärtsrollen des Bildschirms werden bereits durch die Ausgabeoperation selbst ausgelöst. Es genügt, irgendetwas in die angegebenen Ausgabepore zu schreiben. Der jeweils ausgegebene Wert ist hier unwesentlich.

Die über die Tore 20 bis 26 programmierbaren Steuerregister müssen dagegen mit genau definierten Werten geladen werden. Dabei sind folgende Funktionen genauer bestimmbar:

- die Anzahl der Rasterzeilen auf dem Schirm
- die Anzahl der leeren Rasterzeilen am Bildanfang
- der Rastermodus (mit oder ohne Zeilensprungverfahren)
- die gesamte Zeilenlänge,
- die Breite des Zeilensynchronimpulses
- der Abstand des Zeilensynchronimpulses vom rechten Bildrand
- die Verzögerung der Austastsignale (und des intern gesteuerten Cursors) gegenüber der Horizontaladresse (skew factor)
- die Anzahl von Rasterzeilen pro Zeichen
- die Anzahl von Zeichen pro Zeile
- die Zahl der auf dem Schirm angezeigten Zeilen
- die Nummer der letzten auf dem (vollen) Schirm angezeigten Zeile

Im einzelnen erfüllen die Steuerregister folgende Funktionen:

#### Programmierung der Video-Steuerung (VTAC 5027/TMS9927)

Register- nummer	Bit- nummer	programmierte Funktion
20	7...0	Zeilenlänge, berechnet nach:

Videotakt

-----  
Zeichenbreite x Rasterzeilen x Bildfrequenz

dabei ist vorgegeben (europäische Norm):

Zeichenbreite = 8  
 Rasterzeilen = 312 (ohne Zeilensprung)  
                   625 (mit Zeilensprung)  
 Bildfrequenz = 50 (ohne Zeilensprung)  
                   25 (mit Zeilensprung)

21            7            Rastermodus  
                           mit Zeilensprung = 1  
                           ohne Zeilensprung = 0  
                   6...3        Breite des Zeilensynchronisierimpulses  
                           in Einheiten der Zeichenbreite  
                   2...0        Abstand des rechten Bildrand zum Zeilensyn-  
                           chronisierimpuls, gemessen in Einheiten der  
                           Zeichenbreite

22            6...3        Rasterzeilen pro Zeichen (- 1)  
                   2...0        Anzahl (darstellbarer) Zeichen pro Zeile  
                           verschlüsselt nach:

Bit 2	Bit 1	Bit 0	Zeichenzahl
0	0	0	20
0	0	1	32
0	1	0	40
0	1	1	64
1	0	0	72
1	0	1	80
1	1	0	96
1	1	1	132

23            7, 6        Verzögerungsfaktor der Austastsignale und  
                           des Cursors gegenüber der normalen Schirm-  
                           position (in Einheiten der Zeichenbreite),  
                           verschlüsselt nach:

Bit 7	Bit 6	Austastsignal	Kursor
verzögert um Zeichen:			
0	0	0	0
0	1	1	0
1	0	2	1
1	1	2	2

5...0        Anzahl der Zeichenzeilen (- 1)

24            7...0      Anzahl der Rasterzeilen pro Schirmbild  
verschlüsselt nach:

mit Zeilensprung:

$$\frac{\text{Rasterzeilen} - 513}{2}$$

ohne Zeilensprung:

$$\frac{\text{Rasterzeilen} - 256}{2}$$

25            7...0      Anzahl der leeren Rasterzeilen am Bild-  
anfang (zwischen dem Anfang des Bildsyn-  
chronisierimpulses und der ersten Daten-  
zeile)

26            5...0      Nummer der letzten auf dem (vollen) Schirm  
angezeigten Zeichenzeile. Die Zählung be-  
ginnt dabei mit 0.

### 1.1.3.1.2.1 Besonderheiten

Bei der Programmierung des Steuerbausteins ist zu beachten:

- Die Programmierung hängt wesentlich von dem Videotakt ab, mit dem der Steuerbaustein versorgt wird. Dieser wird der Kombinationskarte von der Monitoranpassungskarte geliefert, was bedeutet, daß beim Austausch der Anpassungskarten in der Regel auch der Bildschirmsteuerbaustein neu programmiert werden muß. Aus diesem Grund befinden sich auf den Anpassungskarten Kodierungsbrücken, deren Lage durch das Statustor auf der Kombinationskarte abgefragt werden kann (s.u.).
- Die Arbeit im Zeilensprungverfahren verdoppelt die vertikale Auflösung des Bilds. (Es können doppelt so viele Textzeilen dargestellt werden.) Wegen der relativ geringen Gesamtbildfrequenz von 25 Hz kommt es dabei jedoch zu unangenehmen Flimmererscheinungen, die nur bei einem Monitor mit langsamen Phosphor (z.B. P39) vertretbar sind. Bei den in der Regel verwendeten schnellen bis mittelschnellen Phosphoren empfiehlt sich daher, auf das Zeilensprungverfahren zu verzichten. Der Bild-



wiederholtspeicher auf der Kombinationskarte faßt demzufolge auch gerade 24 Zeilen zu je 80 Zeichen.

- Die exakte Zeilen- und Bildfrequenz hängt von zwei Parametern ab, der Gesamtlänge der Zeile und der Anzahl von Rasterzeilen pro Schirm. Dabei ist der erste der bestimmende Faktor, während der zweite zum "Feinabgleich" dient. Das Bild wird normalerweise vom Monitor in einem Bereich von 3 bis 4 Zeichenbreiten um den Idealwert der Zeilenbreite noch sicher gefangen, dagegen machen sich bereits leichte Abweichungen von der 50-Hz-Bildfrequenz durch brummartige Überlagerungserscheinungen im Bild bemerkbar (das Bild "tanzt"). Man muß gegebenenfalls etwas mit beiden Parametern experimentieren, um ein ruhig stehendes Bild zu erhalten.
- Die Zählung von Zeilen und Zeichen beginnt immer bei 0. Dies ist vor allem für den folgenden Punkt wichtig:
- Die Lage der Zeichenzeilen auf dem Schirm wird durch den Wert von Register 26 bestimmt. Es legt fest, welche Zeile ganz unten auf dem Schirm angezeigt wird. Durch Verändern dieses Werts läßt sich das Bild auf- oder abwärtsrollen (scrolling). Dabei werden die unten herausgeschobenen Zeilen oben am Bild angefügt und umgekehrt.

**Beispiel:** Wenn die letzte Zeile auf dem Schirm die Nummer 10 hat, dann bewirkt eine 9 in Register 26 ein Abwärtsrollen des Bilds um eine Zeile, während der Wert 11 das Bild um eine Zeile nach oben rollt.

Dieses Verfahren des Auf- oder Abrollens durch die Hardware des Steuerbausteins hat den Vorteil, daß es sehr schnell und ohne Störung des Bilds (kein Schnee o.ä.) vor sich geht. Nachteilig ist in manchen Anwendungsfällen, daß dabei der gesamte Schirminhalt verschoben wird (man kann so beispielsweise nicht den Text in einem "Fenster" auf dem Schirm verschieben).

#### 1.1.3.1.2.2 Beispiel (24x80-Anzeige)

Bei einem Videotakt von 12,8112 MHz, wie er von dem Standard-Monitoradapter (s. Abschnitt 5.2.1) geliefert wird und der Arbeit ohne Zeilensprungverfahren, erhält man mit den folgenden Werten eine ruhige Anzeige von 24 Zeilen zu 80 Zeichen mit Zeile 0 als erster, Zeile 23 als letzter Schirmzeile:

Register- nummer	Wert (hexadezimal)	Z80-Programmbefehle (8080-kompatibel)
20	65	LD A,6DH OUT 20H,A
21	2E	LD A,2EH OUT 21H,A
22	5D	LD A,5DH OUT 22H,A
23	57	LD A,57H OUT 23H,A
24	1D	LD A,1DH OUT 24H,A
25	12	LD A,12H OUT 25H,A
26	17	LD A,17H OUT 26H,A

Nutzt man die Z80-Möglichkeiten besser aus, dann erhält man den gleichen Effekt durch das folgende Programmstück:

```

SET8024: LD HL,VAL8024 ; HL --> Initialisierungstabelle
          LD BC,7*256+1FH ; B = Registerzahl, C = Adresse-1
DOSET:  INC C ; Ausgaberegister bestimmen
          OUTI ; Tabellenwert ausgeben & weiter
          JR NZ,DOSET ; Schleife, solange B ungleich 0
          RET ; sonst zurück zur Aufrufstelle
VAL8024: DEFB 65H ; Initialisierung, Register 20
          DEFB 2EH ; Initialisierung, Register 21
          DEFB 5DH ; Initialisierung, Register 22
          DEFB 57H ; Initialisierung, Register 23
          DEFB 1DH ; Initialisierung, Register 24
          DEFB 12H ; Initialisierung, Register 25
          DEFB 17H ; Initialisierung, Register 26
    
```

### 1.1.3.2 Der Bildwiederholpeicher

Die auf dem Schirm darstellbare Information wird in einem 2 Kbytes tiefen Schreib-Lese-Speicher in Form von ASCII-Zeichen festgehalten. Der Steuerbaustein liest die dort stehenden Byte-werte zeichenweise Zeile für Zeile aus. Dabei wird jeder Zeichenwert in einem 8-Bit-Puffer zwischengespeichert. Die Ausgänge des Puffers adressieren dann zusammen mit dem vom Zeichengenerator erzeugten Abtastbits R0 bis R3 den Zeichengenerator (siehe Abschnitt 1.1.3.3).

### 1.1.3.2.1 Adressierung des Bildwiederholerspeichers

Der Bildwiederholerspeicher muß von zwei Seiten aus adressiert werden:

- durch die CPU oder andere Systemeinheiten zum Einschreiben der darzustellenden Information,
- durch den Bildschirm-Steuerbaustein zum zeilen- und zeichenweisen Auslesen der Information im vorgegebenen Rastertakt.

Diese Anforderungen bringen einige teilweise einander widersprechende Aufgaben mit sich. Das betrifft zum einen den Speicherzugriff und zum anderen die Organisation der Information im Speicher. So ist im einzelnen zu berücksichtigen:

- Zur Adressierung eines Zeichens stehen (bei einer Organisation von 80 Zeichen pro Zeile im europäischen Bildschirmformat von 312 Zeilen fünfzigmal pro Sekunde) nur etwa 0,8 Millisekunden zur Verfügung.
- um ein ungestörtes Bild zu erhalten, muß der Steuerbaustein den Bildwiederholerspeicher demnach für die Dauer eines Bilds (vom Ende der einen vertikalen Austastlücke bis zum Anfang der nächsten) ununterbrochen adressieren können.
- Auf der anderen Seite muß man vom Systembus aus auf den Speicher zugreifen können. Dazu bleiben zwei Möglichkeiten: Entweder man greift sofort wenn die CPU es verlangt auf den Speicher zu und nimmt dafür eine Störung des Bilds in Kauf, oder man wartet mit dem Zugriff, bis gerade eine vertikale Austastlücke vorliegt. Im letzteren Fall wird ein Mechanismus benötigt, durch den der Zugriff auf die vertikale Austastlücke synchronisiert werden kann. Dadurch wiederum verlangsamt sich die Ausgabeoperation u. U. beträchtlich.
- Bei einer Bildschirmorganisation von 80 x 24 Zeichen werden 1920 Zeichenplätze im Bildwiederholerspeicher belegt. Diese füllen nahezu die gesamte Kapazität von 2048 verfügbaren Bytes aus. Das bedeutet, daß die Zeilen zusammenhängend abgespeichert werden müssen: Alle (oder nahezu alle) 80 Zeichen beginnt eine neue Zeile. Da sich das nicht mit der Zeilen-Zeichen-Adressierung durch den Steuerbaustein verträgt, ist eine passende Adreßumsetzung von der zeilenweisen zur zusammenhängenden Darstellung notwendig.
- Dies betrifft auch den Zugriff über den Systembus. Hier könnte man im Prinzip auch einen zusammenhängenden Speicher adressieren, doch muß dann die jeweilige Zeilen-Spalten-Adresse vor jedem Zugriff zeitaufwendig umgerechnet werden. Es ist günsti-

ger, wenn man auch diese Adreßumsetzung durch passende Hardware ausführen läßt.

Auf der Video-Tastatur-Kombinationskarte I hat man diese Probleme wie folgt gelöst:

### 1.1.3.2.1.1 Systemzugriff auf den Speicher

Vom System her erscheint der Bildwiederholpeicher als ein RAM von 4 KBytes Tiefe ab Adresse 3000H, das nach Zeile und Zeichenposition im folgenden Format adressiert wird:

#### Adreßformat zum Zugriff auf den Bildwiederholpeicher

A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
!	0	!	0	!	1	!	1	!							!
+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+

! 0 ! 0 ! 1 ! 1 !      Zeilennummer      !      Zeichenposition      !

Diese effektive 12-Bit-Adresse wird auf der Karte über einen Multiplexer auf eine 11-Bit-Adresse reduziert und mit dieser der Speicher angesprochen. Die Adreßzuordnung stellt sich dabei wie folgt dar:

**Interne Zeichenadressen im Bildwiederholpeicher**

Zeilen- nummer	Zeichenposition	
	0 bis 63	64 bis 79
0	000 ... 03F	600 ... 60F
1	040 ... 07F	640 ... 64F
.	.	.
.	.	.
.	.	.
15	3C0 ... 3FF	7D0 ... 7DF
16	400 ... 43F	620 ... 62F
17	440 ... 47F	660 ... 66F
.	.	.
.	.	.
.	.	.
23	5C0 ... 5FF	7E0 ... 7EF
24	600 ... 63F	
.	.	
.	.	
.	.	
31	7C0 ... 7FF	

**Wichtig:**

Der Speicherbereich für die Zeichen Nummer 64 bis 79 in Zeile 0 bis 23 und der für die Zeichen Nummer 0 bis 63 in Zeile 24 bis 31 ist identisch. Das bedeutet, daß die größtmöglichen Darstellungsformate entweder **32 Zeilen zu 64 Zeichen** oder **24 Zeilen zu 80 Zeichen** sind.

**1.1.3.2.1.2 Interner Speicherzugriff**

Die vom Steuerbaustein ausgegebenen Zeilen- und Spaltenadressen können ebenfalls nicht unmittelbar verwertet werden. Auch sie müssen erst in eine 11-Bit-Adresse umgesetzt werden. Das geschieht über einen besonders verschalteten, 16 Bit breiten Multiplexer, der auch die Adressen der Abtastzeile innerhalb einer Zeichenzeile verarbeitet.

Der Multiplexer bietet zwei Zusatzmöglichkeiten:

- Bestimmen der **vertikalen Zeichenauflösung**, d.h. der Anzahl von Abtastzeilen pro Zeichenzeile. Über Brücken auf der Karte kann zwischen 12 und 16 Abtastzeilen pro Zeichen gewählt werden.

- **Vergrößern der Schirmbilddarstellung ("Lupe")**, eine über Software steuerbare Funktion, bei der jedes Zeichen auf dem Schirm mit doppelter Größe dargestellt wird. (Dies ist die eigentliche Multiplexerfunktion: Die Adreßumsetzung wird durch die Lupe geeignet umgeschaltet, so daß jede Zeile und jeder Punkt im Zeichengenerator zweimal auf den Schirm geschrieben werden.)

Diese beiden Möglichkeiten werden weiter unten ausführlicher dargestellt.

### 1.1.3.2.2 Synchronisation des Speicherzugriffs

Prinzipiell ist die Karte so ausgelegt, daß der Systemzugriff auf den Bildwiederholungspeicher Vorrang vor dem internen Zugriff hat. So kann beispielsweise die CPU ohne warten zu müssen ein Zeichen ausgeben oder abfragen, welches Zeichen auf einer bestimmten Stelle auf dem Schirm steht. Da dies jedoch die normale Abtastung im Zuge der Schirmbildauffrischung unterbricht, erscheint für die Dauer dieses Zugriffs ein Störimpuls ("Schnee") auf dem Bildschirm. Dieser unmittelbare Zugriff lohnt sich also nur dann, wenn ein Bild sehr schnell verändert werden soll, und man Störungen des Bilds in Kauf nehmen kann. In allen anderen Fällen sollte man den Systemzugriff in die Bild- oder Zeilenaustastlücke legen.

Zu diesem Zweck sind die vom Bildschirmsteuerbaustein bereitgestellten Synchronisierungssignale, mit denen der Schirm dunkelgetastet wird, über ein Eingabeterminal auf der Karte abfragbar.

#### Video-Synchronisierungsinformation (IN 35H)

Bit	Bezeichnung	Funktion
6	VSYN	1, wenn die vertikale Austastlücke (Bild-austastlücke) erreicht wurde
7	BL	1, wenn die horizontale (Zeilen-) oder die vertikale Austastlücke (Bild-austastlücke) erreicht worden ist

Der Prozessor fragt vor Zugriff auf den Bildwiederholungspeicher eine dieser Synchronisierungsinformationen ab und greift erst dann auf den Speicher zu, wenn die jeweilige Lücke vorliegt. Der Zugriff geht schneller, wenn man die allgemeine Austastinformation (BL) nutzt, jedoch ist die Zeilenaustastlücke in der Regel

für einen Zugriff ohne Störung des Bilds zu klein. Man wird von der CPU aus daher vorzugsweise auf die Bildaustastlücke (VSYN) synchronisieren und die andere Möglichkeit rascheren, DMA-ähnlichen Verfahren überlassen.

**Beispiel:** Es sei das in Register C vorliegende Zeichen ohne Störung des Schirmbilds an die durch Register HL adressierte Stelle auf dem Schirm auszugeben. Das läßt sich mit folgendem Unterprogramm erreichen (wobei angenommen ist, daß die interne Speicherseite 8 bereits ausgewählt ist; das Programmstück wird sich demzufolge normalerweise im Bereich oberhalb der Adresse 0C000H befinden):

```

VIDOUT: IN  A,(35H)      ; Status abfragen
        BIT  6,A         ; ist VSYN (Bit 6) gesetzt?
        JR   Z,VIDOUT   ; nein: weiterwarten
        LD   (HL),C     ; sonst Zeichen ausgeben
        RET              ; und zurück zum Aufruf
    
```

### 1.1.3.2.3 Festlegen der Zahl der Abtastzeilen

Auf der Kartenrückseite befinden sich (hält man Stecker nach unten) ganz links vier vorkaschierte Brücken, die von "a" bis "d" bezeichnet sind. Sie legen fest, wieviele Abtastzeilen pro Zeichenzeile verwendet werden.

#### Brücken zur Festlegung der Zahl der Abtastzeilen

Abtastzeilen pro Zeichen	Brücken
12	b, d geschlossen a, c offen
16	b, d offen a, c geschlossen

### 1.1.3.2.4 Lupe

Die Lupenfunktion (Verdoppelung der Zeichengröße auf dem Schirm) wird über Ausgabebitor 34 gesteuert. Bit 0 im ausgegebenen Wert bestimmt dabei die Zeichengröße.

**Lupenfunktion (OUT 34H)**

Bit 0 = 1: Lupe eingeschaltet

Bit 0 = 0: Lupe ausgeschaltet

**Wichtig:** Die Lupe wirkt auf den Schirm als Ganzes, nicht nur auf ein einzelnes Zeichen auf dem Schirm. Das bedeutet, daß die Zahl der auf dem Schirm darstellbaren Zeichen geringer wird. Das Format 80 x 24 wird zu 40 x 12, wobei das obere linke Schirmviertel angezeigt wird (falls Zeile 0 ganz oben auf dem Schirm steht).

**Vorsicht beim Auf- und Abrollen des Schirmbilds, wenn die Lupe eingeschaltet ist!** Durch die Umsetzung der Adressen für den Bildwiederholtspeicher verhält sich das Schirmbild recht kompliziert, wenn mit der Hardware-Rollfunktion (Register 26 oder 2B des Steuerbausteins) gearbeitet wird. Am sichersten setzt man Zeile 0 nach oben und rollt das Bild dann bei Bedarf über Software durch Umspeichern der Information im Bildwiederholtspeicher.



### 1.1.3.3 Der Zeichengenerator

Im Bildwiederholtspeicher werden nur die Zeichenkodes, nicht die Zeichenbilder selbst festgehalten. Das beschleunigt die Arbeit mit dem Schirmbild von der Systemseite (z.B. der CPU) aus, erfordert aber, daß die Information aus dem Bildwiederholtspeicher besonders aufbereitet wird, um ein sinnvolles Videosignal zu bekommen.

Diese Funktion erfüllt der Zeichengenerator. Er besteht auf der Kombinationskarte I aus einem 2-K-EPROM, das die Bilder von 128 Zeichen trägt. Jedes Zeichen ist in einer 16x8-Bit-Matrix festgehalten, die das Punktmuster des Zeichenbilds bestimmt, wie es auf dem Schirm erscheinen soll. Ein gesetztes Bit in dieser Matrix (Wert Eins) ergibt bei der Umsetzung einen hellen, ein gelöschtes (Wert Null) einen dunklen Punkt auf dem Schirm.

#### 1.1.3.3.1 Organisation des Zeichengenerators

Diese Zeichenbilder sind im EPROM fortlaufend wie folgt angeordnet:

- Ein Zeichen beginnt immer an einer Sechzehnergrenze (niederwertige Adreßbits 0 bis 3 = 0000).
- Jedes Byte (8 Bits) entspricht einer Abtastzeile des Zeichens, wobei Bit 0 den ganz links, Bit 7 den ganz rechts stehenden Punkt im Zeichenbild bestimmt.

**Wichtig:** Beachten Sie, daß die Bitmuster im Zeichengenerator-EPROM dadurch spiegelverkehrt zu den Punkten auf dem Schirm liegen.

- Die sechzehn für ein Zeichen möglichen Abtastzeilen stehen auf aufeinanderfolgenden Adressen im EPROM (so nimmt beispielsweise ein großes "E" den Adreßraum von 450H bis 45FH im Zeichengenerator-EPROM ein). Normalerweise werden bei einem 80x24-Bildschirmformat davon aber nur die ersten 12 Bytes (die 12 oberen Abtastzeilen pro Zeichen) genutzt.

#### 1.1.3.3.2 Umsetzung des Zeichenkodes

Die Umsetzung vom im Bildwiederholtspeicher festgehaltenen Zeichenkode zum für das Videosignal benötigten Zeichenbild vollzieht sich auf der Kombinationskarte wie folgt:

- Der aus dem Bildwiederholtspeicher ausgelesene Zeichenkode wird in einem 8-Bit-Latch zwischengespeichert, um die weitere Verarbeitung zu stabilisieren.
- Die sieben niederwertigen Ausgänge dieses Zwischenspeichers adressieren die sieben höherwertigen der elf Adreßeingänge des Zeichengenerator-EPROMs (Adreßbit 4 bis 10) und legen so den Anfang des gewünschten Zeichenbilds fest.  
(Das höchstwertige Bit 7 gibt an, ob das Zeichen hell auf dunklem Grund (Bit 7 = 0) oder dunkel auf hellem Grund (Bit 7 = 1) dargestellt werden soll.)
- Über die vier Zeichenabtastbits, die (nach der Umsetzung durch den Multiplexer) vom Video-Steuerbaustein bereitgestellt werden, wird die jeweils benötigte Abtastzeile im Zeichenbild festgelegt. Sie sind dazu an die vier niederwertigen Adreßbits (0 bis 3) des Zeichengenerator-EPROMs geführt.
- Das mit dieser 11-Bit-Adresse angegebene Byte wird ausgelesen und in ein Schieberegister eingeschrieben, aus dem die Information dann Bit für Bit nach rechts herausgeschoben und zur Konstruktion des Videosignals verwendet wird.

### 1.1.3.3.3 Zeichensätze

Die Zeichenbilder im Zeichengenerator-EPROM sind in der Folge des ASCII-Kodes (ISO-7-Bit-Kodes) angeordnet. Dabei sind den Steuerkodes von 00H bis 1FH Graphikzeichen zugeordnet, denen von 20H bis 7FH die eigentlichen Textzeichen (Zeichen, Ziffern und Buschstaben) folgen.

#### 1.1.3.3.3.1 Graphikzeichen

Das Zeichengenerator-EPROM auf der Kombinationskarte I gestattet auf dem Schirm graphische Darstellungen mit einer Auflösung von 72 Zeilen zu 160 Punkten. Dazu wird jede Zeichenposition auf dem Schirm in drei Zeilen zu je zwei Punkten untergliedert. Um einen Punkt auf dem Schirm darzustellen, muß das dazu passende Zeichen (also immer sechs Punktpositionen auf einmal) geschrieben werden.

Dabei sind alle denkbaren Punktkombinationen zu berücksichtigen. Bei sechs Punkten pro Zeichenposition entspricht das 64 verschiedenen Zeichenbildern. Das Zeichengenerator-EPROM bietet zwar nur 32 Graphikzeichen Platz, doch lassen sich die restlichen 32 Zeichen durch Helligkeitsumkehr daraus ableiten. Die

Zeichen sind so angeordnet, daß jedem Punkt ein bestimmtes Adreßbit wie folgt entspricht.

**Bitzuordnung an die Graphikzeichen**

```

+---+---+
! 0 ! 1 !
+---+---+
! 2 ! 3 !
+---+---+
! 4 !   ! <-- nicht unmittelbar
+---+---+                kodiert
    
```

Der nicht unmittelbar kodierte sechste Punkt wird durch Helligkeitsumkehr erfaßt, wobei sich dann die Bitwertigkeiten in Position 1 bis 4 umkehren müssen (0 = Punkt setzen, 1 = Punkt löschen). Dies geschieht durch Setzen von Bit 7 im Zeichenkode, so daß die 64 möglichen Graphikzeichen wie folgt kodiert sind.

**Zeichenkodes zur Graphikwiedergabe**

(Ein "0" gibt einen hellen, ein "." einen dunklen Punkt wieder.)

..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
.C	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
9F	9E	9D	9C	9B	9A	99	98	97	96	95	94	93	92	91	90
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
8F	8E	8D	8C	8B	8A	89	88	87	86	85	84	83	82	81	80

**Amerkung:** Die dadurch erreichte Auflösung auf dem Schirm ist nicht symmetrisch. Das Schirmbild hat ein Höhen-zu-Breiten-Verhältnis von 3:4, was bei einer Auflösung von 72x160 Punkten ergibt, daß einem Schritt in waagrechtlicher Richtung 3/5 Schritte in senkrechter Richtung entsprechen. D.h. das Punktverhältnis beträgt (x = Breite, y = Höhe):

$$x:y = 3:5$$

(Man erhält also ein Quadrat, wenn man einen Block mit 5 Punkten Breite und 3 Punkten Höhe zeichnet.)

#### 1.1.3.4 Erzeugung des Videosignals

Das aus dem Zeichengenerator gewonnene Zeichenbild muß noch Bit für Bit in ein Videosignal umgewandelt werden. Dazu muß der Schieberegisterinhalt mit einer durch den anzusteuern Monitor vorgegebenen Frequenz nach rechts (mit den niederwertigen Bits voran) ausgeschoben, mit der Helltast-Information aus Bit 7 des Zeichenkodes verknüpft und mit den Synchronisierungssignalen versehen werden.

Dies geschieht in zwei Stufen. In der ersten wird der von der Monitoradapterkarte gelieferte Quarztakt geeignet aufbereitet und dem Video-Steuerbaustein, dem Schieberegister sowie der Video-Aufbereitungslogik zugeführt.

Die zweite Stufe ist die Videosignalaufbereitung selbst. Sie verknüpft folgende Informationen:

- die **Zeicheninformation**, wie sie vom Schieberegister bereitgestellt wird,;
- das **Invertierungsbit** (Zeichenkode, Bit 7), das mit der Zeicheninformation EXOR-verknüpft wird und dadurch deren logischen Wert dann umkehrt, wenn eine Invertierung des Zeichenbilds gefordert wurde;
- die kombinierte **Austastinformation** (das BL-Signal des Video-Steuerbausteins), die einen Nullpegel des Videosignals in der Austastlücke gewährleistet.

Das daraus gewonnene Videosignal hat TTL-Pegel und ist mit "positiver Logik" definiert (1 = Helltasten, 0 = Dunkeltasten). Es wird über Anschluß 86 des Kartensteckers ausgegeben und an die Monitoradapterkarte weitergeleitet. Diese erzeugt aus der Videoinformation und den Synchronisationssignalen  $\overline{H}\overline{S}\overline{Y}\overline{N}$  (horizontale Synchronisation, Anschluß 83),  $\overline{V}\overline{S}\overline{Y}\overline{N}$  (vertikale Synchronisation, Anschluß 84) und  $\overline{C}\overline{S}\overline{Y}\overline{N}$  (kombiniertes Synchronisationssignal, Anschluß 85) das für den jeweiligen Monitor benötigte BAS- oder sonstige Ansteuersignal (siehe Abschnitt 5.2).

### 1.1.4 Die Tastatursteuerung

Die Tastatursteuerung auf der Kombinationskarte I wird im wesentlichen von dem programmierbaren Schnittstellenbaustein 8278 oder einem entsprechend programmierten 8048-Baustein durchgeführt. (Der 8278-Baustein ist ein fest vorprogrammierter 8048-Einchipcomputer.) Dies ermöglicht die Ansteuerung einer Tastaturmatrix aus maximal 128 Tasten. Der ebenfalls auf dem Baustein untergebrachte Teil zur Ansteuerung einer Siebensegmentanzeige wird auf der Kombinationskarte I nicht genutzt.

Die vom 8278-Baustein verarbeiteten Signale müssen noch durch eine Anpassungskarte der verwendeten Tastatur entsprechend aufbereitet werden. Diese Karte befindet sich auf dem Verdrahtungsfeld ganz links auf Steckplatz VII. Sie wird in Abschnitt 5.3 beschrieben.

#### 1.1.4.1 Funktionsprinzip

Die Tastatur besteht aus maximal 128 Tasten, die (elektrisch) in einer Matrix von 16 Zeilen und 8 Spalten angeordnet sind. (Dies ist eine Frage der Verdrahtung des Tastaturfelds. Die physische Tastaturanordnung ist davon unabhängig.) Auf jedem Kreuzungspunkt von Zeilen und Spalten kann ein Tastenschalter angeordnet sein, der bei Betätigen die betreffende Zeilen- und Spaltenleitung miteinander verbindet. (Dabei sind verschiedene Technologien möglich, z.B. Schalter- oder kapazitive Tasten. Die Anpassung an die verwendete Technologie ist eine Aufgabe der Adapterkarte.) Aufgabe der Tastatursteuerung ist es, eine solche Verbindung von Zeilen- und Spaltenleitung zuverlässig zu erkennen und in den der betreffenden Taste zugeordneten Zeichenkode umzusetzen.

Dazu dienen sieben Steuerleitungen, M0 bis M6 (die als MSA 0 bis MSA 6 auf Steckeranschluß 65 bis 71 gelegt sind). Sie haben unterschiedliche Aufgaben:

- M0, M1 und M2 erfassen die acht Spaltenleitungen.
- M3 bis M6 erfassen die 16 Zeilenleitungen.

Dazu kommt eine Rückleitung RL (invertiert als  $\bar{R}\bar{L}$  auf Steckeranschluß 72 gelegt), die meldet, ob die gerade angesteuerte Taste offen oder geschlossen ist.

Schließlich gibt es noch ein Taktsignal KCL (auf Steckeranschluß 95 gelegt), das angibt, wann eine neue Taste untersucht wird.

Die Abfrage erfolgt im Prinzip so:

- Der Steuerbaustein legt einen Kode auf die Zeilenleitungen M3 bis M6.
- Dieser wird in der Anpassungselektronik geeignet dekodiert und die zugehörige Zeilenleitung angesteuert.
- Ist eine Taste in dieser Zeile gedrückt, dann wird das Signal auf eine der Spaltenleitungen weitergegeben.
- In der Zeit, in der eine Zeilenleitung angesteuert wird, gibt der Steuerbaustein nacheinander (im KCL-Takt) die Codes für alle acht Spaltenleitungen aus.
- Diese Spaltenkodes steuern in der Tastaturelektronik einen Multiplexer an, der die betreffende Spaltenleitung auf die Rückleitung RL durchschaltet.
- Jede in der betreffenden Zeile gedrückte Taste wird so über die Rückleitung vom Steuerbaustein erfaßt.
- Dieser Vorgang wiederholt sich für alle 16 Zeilen, so daß alle 128 möglichen Tasten in einem Abfragezyklus erfaßt werden.

Wichtig ist dabei, daß die Tastatursteuerung ohne Schwierigkeiten auch mehrere gleichzeitig gedrückte Tasten (in der Reihenfolge der Matrixabfrage) erfassen kann.

#### **1.1.4.2 Entprellung**

Die Dauer des Abfragezyklus ist von der Taktversorgung des Steuerbausteins abhängig. Auf der Kombinationskarte I wird die Tastaturmatrix in 10,7 Millisekunden einmal abgefragt.

Diese definierte Zyklusdauer wird zur Tastenentprellung eingesetzt. Wurde eine gedrückte Taste entdeckt, dann wird dies zunächst nur vermerkt und im nächsten Abfragezyklus untersucht, ob die betreffende Taste immer noch gedrückt ist. Wenn dies nicht der Fall ist, dann handelt es sich um einen Prellimpuls: Die Taste wird nicht berücksichtigt. Ist die Taste dann immer noch gedrückt, so ist die Prellphase normalerweise bereits überschritten, und man kann davon ausgehen, daß es sich um eine absichtlich gedrückte Taste handelt.

### 1.1.4.3 Zeichenpuffer

Die Matrixadresse (M0 bis M6) einer derart entprellten Taste wird im Steuerbaustein in einem Zeichenpuffer abgelegt. dabei handelt es sich um ein First-in-first-out-Register (FIFO), bei dem die zuerst eingeschriebene Information auch zuerst wieder ausgelesen wird. Das FIFO bietet 8 Speicherplätze und kann auf Befehl ausgelesen oder auch gelöscht werden. Sollen mehr als acht Zeichen in den Puffer eingeschrieben werden, so wird ein Fehler gemeldet. Keine ausdrückliche Fehlermeldung erfolgt jedoch, wenn versucht wird, ein Zeichen aus dem leeren FIFO zu lesen (hier wird das zuletzt ausgelesene Zeichen erneut übergeben).

### 1.1.4.4 Mehrfachbetätigung (rollover)

Der Tastatursteuerbaustein 8278 erlaubt verschiedene Antworten auf die Situation, in der mehrere Tasten zugleich gedrückt sind:

- **Mehrtastentrennung (N-key rollover)**

Hier wird jede (neu gedrückte) Taste unabhängig von allen anderen behandelt und nach Entprellung in das FIFO eingeschrieben. Dies ist möglich, weil durch die Abfragemethode jede einzelne Taste für sich erkannt werden kann.

Das gestattet die Bedienung der Tastatur auch durch Ungeübte mit voller Geschwindigkeit, da die vorhergehende Taste nicht losgelassen sein muß, bevor die nächste erkannt werden kann.

- **Mehrtastenausblendung (N-key lockout)**

Der 8278-Baustein kann so programmiert werden, daß er immer dann einen Fehler meldet, wenn in einem Abtastzyklus mehr als eine gedrückte Taste vorgefunden wird. In diesem Fall wird der Puffer gesperrt bis die Fehlerflagge ausdrücklich gelöscht worden ist.

Damit lassen sich solche Anwendungen verwirklichen, in denen es wichtig ist, daß immer eine Taste nach der anderen gedrückt wird.

Kommt es auf die genaue Reihenfolge der Tastendrucke an, dann gibt es eine weitere Möglichkeit zur Auswertung mittels Software. Hier kann der Baustein so programmiert werden, daß eine Taste zweimal in den Puffer eingeschrieben wird: Einmal, wenn sie gedrückt, und einmal wenn sie losgelassen worden ist. Damit läßt sich beispielsweise eine Zweitastentrennung erreichen (2-key rollover), oder man kann Spezialeffekte programmieren.



### 1.1.4.5 Programmierung des 8278-Bausteins

Die meisten Funktionen des Tastaturbausteins 8278 sind programmierbar; sie müssen vor seinem ersten Einsatz (nach dem Einschalten des Systems o.ä.) erst vorgegeben werden. Im wesentlichen wird dabei erfaßt:

- die Tastaturfunktion (Zeichen ein- oder zweimal übergeben),
- die Fehlererkennung bei Mehrfachbetätigung,
- die Unterbrechungsanforderung (bei Vorliegen eines Zeichens im FIFO oder bei einem Fehler),
- Löschen der Fehlerflagge,
- Löschen des FIFO-Registers und
- Auslesen des FIFO-Registers.

#### 1.1.4.5.1 Die 8278-Register

Der Tastatursteuerbaustein 8278 enthält drei von außen zugängliche Register, die für die Tastatursteuerung wesentlich sind:

- ein Befehlsregister,
- ein Statusregister und
- den FIFO-Puffer, der allerdings nur über ein besonders Datenbusregister zugänglich ist.

Diese Register sind als Ein-Ausgabepore unter folgenden Adressen erreichbar:

#### Adressen der 8278-Register zur Tastatursteuerung

Funktion	E-A-Adresse	Z80-Befehl
Datenbusregister lesen	10H	IN A, (10H)
Status lesen	11H	IN A, (11H)
Befehl ausgeben	11H	OUT (11H), A

#### 1.1.4.5.2 8278-Befehle

Im Rahmen der Tastatursteuerung kann man dem 8278-Baustein drei verschiedene Arten von Befehlen übergeben:

- Betriebsart setzen,
- FIFO lesen und
- Löschbefehle (Fehlerflagge, FIFO).

Die jeweilige Befehlsart ist im übergebenen Befehlsbyte in den drei höchstwertigen Bits wie folgt verschlüsselt.

**Befehlsarten für den 8278-Baustein (Tastatursteuerung)**

Bit 7	Bit 6	Bit 5	Befehlsart
0	0	0	Betriebsart setzen
0	1	0	FIFO lesen
1	0	1	Löschbefehl

**1.1.4.5.2.1 Betriebsart des 8278-Bausteinssetzen**

**Format des Befehlsbytes (OUT 11H):**

7	6	5	4	3	2	1	0
+	+	+	+	+	+	+	+
!	0	!	0	!	0	!	7-Seg.-
!	!	!	!	!	!	!	Tasten-
!	!	!	!	!	!	!	erkenn.!
+	+	+	+	+	+	+	+

Bit	Bezeichnung	Bedeutung
4 und 1	7-Seg.-Anzeige	Anzeigensteuerung (hier nicht benutzt)
3	Fehlerbehandlung	0 = Fehler bei Mehrfachbe- tätigung von Tasten mel- den 1 = keinen Fehler in diesem Fall melden (N-key roll- over)
2	Unterbrechungs- anforderung	0 = Unterbrechung bei Zei- chen im FIFO und bei Fehler anfordern (über INT0, Anschluß 62) 1 = keine Unterbrechung an- fordern
0	Tastenerkennung	0 = Normalbetrieb, jede Taste nur einmal melden 1 = jede Taste einmal melden, wenn sie gedrückt wurde, und einmal, wenn sie los- gelassen wurde

#### 1.1.4.5.2.2 FIFO lesen

**Befehlsbyte:** 40H

**Wirkung:** Das unterste FIFO-Register wird in das Datenbusregister übertragen und der FIFO-Inhalt einen Platz nach unten geschoben. Der Zähler (im Statusregister) wird heruntergezählt.

Man erhält den FIFO-Inhalt anschließend durch Auslesen des Datenbusregisters.

**Fehler:** Wenn das FIFO leer ist, wird das zuletzt ausgelesene Zeichen übergeben. Der Zähler bleibt auf Null, und es wird kein Fehler gemeldet.

#### 1.1.4.5.2.3 Löschbefehl

Es gibt im Rahmen der Tastatursteuerung zwei Löschbefehle:

<b>Befehl</b>	<b>Befehlsbyte</b>
Fehlerflagge (ERROR) löschen	0CEH
FIFO-Inhalt löschen	0CFH

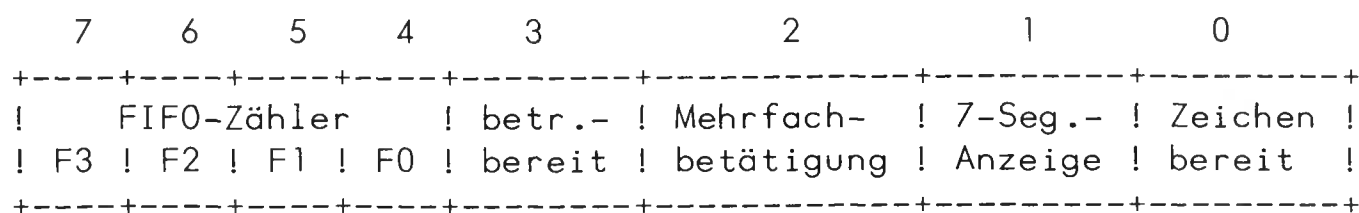
#### 1.1.4.5.3 Das Statusregister

Der 8278-Steuerbaustein bietet folgende Statusinformationen, was die Tastatursteuerung angeht:

- FIFO ist nicht leer,
- Mehrfachbetätigung von Tasten,
- Baustein ist nicht abfragebereit und
- Anzahl der im FIFO abgelegten Tastenkodes.

Diese Information ist wie folgt in Eingaberegister 11 abgelegt:

**Format des Statusbytes (IN 11H):**



Bit	Bezeichnung	Bedeutung
7...4	FIFO-Zähler	0000 = FIFO ist leer 0001...0111 = Zeichen im FIFO 1111 = FIFO-Überlauf
3	betriebsbereit	1 = 8278 ist beschäftigt 0 = 8278 ist betriebsbereit, d.h. Daten sind gültig
2	Mehrfachbetätigung	1 = Mehrere Tasten sind zugleich gedrückt (sperrt FIFO bis Fehlerflagge ausdrücklich gelöscht wurde; s. Löschbefehl; nur wenn Betriebsart Bit 3 gelöscht ist!) 0 = Keine Mehrfachbetätigung oder Mehrtastentrennung gewählt
1	7-Seg.Anzeige	(nicht benutzt)
0	Zeichen bereit	1 = (mindestens) ein Zeichen liegt im FIFO vor 0 = kein Zeichen ist verfügbar

**1.1.4.5.4 Betriebshinweise**

Nach jeder Übergabe eines neuen Befehls ist der 8278-Baustein einige Zeit mit dessen Abarbeitung beschäftigt. Die in dieser Zeit aus dem Datenbusregister abgerufenen Daten sind undefiniert. Erst wenn der betreffende Befehl abgearbeitet worden ist, können von dort gültige Daten übernommen werden. Man muß nach jeder Befehlsübergabe also zunächst das Bereitschaftsbit (Bit 3) im Statusregister testen und warten, bis dieses den Wert Null hat, bevor man auf das Datenbusregister zugreift.

Dies gilt im Fall der Tastatursteuerung für den FIFO-Lesebetrieb. Ist allerdings der Befehl vom 8278-Baustein erst einmal

verarbeitet, dann können so oft neue FIFO-Daten ausgelesen werden, wie dies durch Bit 0 (Zeichen bereit) im Statusregister angegeben wird.

Um eine Folge von Tasten zu übernehmen, empfiehlt sich also folgende Vorgehensweise:

#### Algorithmus zur Zeichenübernahme von der Tastatur

```

Befehl "FIFO lesen" ausgeben,
Wiederholen:
    Statusregister lesen.
    Bis Statusbit 3 (betriebsbereit) = 0.
Wiederholen:
    Wiederholen:
        Statusregister lesen.
        Bis Statusregister (Zeichen bereit) = 1,
        Datenbusregister (d.h. FIFO) lesen.
    Bis genug Zeichen übernommen sind.
    
```

#### 1.1.4.6 Fehlerbehandlung

Wenn ein Fehler auftritt, wird im 8278-Baustein eine Fehlerflagge gesetzt und, falls so programmiert, über den Unterbreuchungszeiger  $\overline{INT0}$  (Steckeranschluß 62) eine Programmunterbrechung angefordert. Dies erfolgt jedoch nicht in allen Fällen. Es gibt folgende Fehlersituationen:

#### Fehlersituationen bei der Tastatursteuerung

Fehler	Kennzeichen	bei Betriebsart
Mehrfachbetätigung	Statusbit 2 = 1	Befehlsbit 3 = 0
FIFO-Überlauf	Status 7...4 = 1111	immer
FIFO-Unterlauf	Status 7...4 = 0000	<b>Keine besondere Fehlermeldung!</b>
Baustein nicht bereit	Statusbit 3 = 1, Datenbusregister ist ungültig	<b>Keine besondere Fehlermeldung!</b>

Die Fehlerflagge wird gelöscht durch:

- Ausgabe eines Löschbefehls (siehe Abschnitt 1.4.4.5.2.3)
- einen Hardware-RESET-Befehl.

**Achtung:** Der 8278-Baustein gehört nicht zur Z80-Familie und kann daher keine Z80-spezifischen Programmunterbrechungen handhaben (er dekodiert keinen RETI-Befehl). Die KombinationskarteI ist daher nicht in die Z80-Unterbrechungskette eingegliedert.

Soll die Unterbrechung über den Zeiger  $\overline{INT0}$  genutzt werden, dann muß das davon aufgerufene Unterbrechungsprogramm mit einem normalen RET-Befehl und nicht mit dem Befehl RETI enden.

### 1.1.5 Signaleinheiten

Am Busstecker der Kombinationskarte I sind zwei Signale (LED: Anschluß 94, Hupe: Anschluß 95) verfügbar, mit denen sich externe Signaleinheiten steuern lassen. Diese Signale werden zur Tastaturanpassungskarte auf Steckplatz VII (ganz links auf dem Verdrahtungsfeld) geführt und dort ausgewertet. Sie haben folgende Funktion:

- **LED** steuert eine Signallampe an, die normalerweise in der Tastatur untergebracht ist und dort als Feststelleranzeige für die Großbuchstabenumschaltung (shift lock) dient.
- **Hupe** steuert einen Tonsignalgeber an, zumeist einen einfachen Oszillator mit nachgeschaltetem Piezo-Schallwandler auf der Tastaturanpassungskarte.

Diese beiden Signale werden über die folgenden Ausgabebitore angesteuert:

#### Adressen der Signaleinheiten

Funktion	Ausgabebitor	Z80-Befehl
Hupe	32	OUT (32H),A
LED	33	OUT (33H),A

Dabei bestimmt Bit 0 im ausgegebenen Byte die Funktion:

- Bit 0 = 1: Hupe bzw. LED einschalten
- Bit 0 = 0: Hupe bzw. LED ausschalten

### 1.1.6 Der Kartenstatus

Auf der Kombinationskarte befindet sich ein allgemeines Statustor, das als Eingabeeinheit unter Adresse 35H erreicht wird. Es bietet folgende Informationen zur Bildschirmsteuerung.

#### Allgemeines Statustor (IN 35H)

Bit	Bezeichnung	Funktion
7	BL	(blank) allgemeines Austastsignal (Zeilen- und Bildsynchronisation)
6	VSYN	(vertical synchronisation) Bild-austastsignal
5...3	STAT5...STAT3	Statusinformationen von der Tastatur: Gibt Auskunft über die eingesetzte Tastatur (siehe Abschnitt 5.3)
2...0	STAT2...STAT0	Statusinformation von der Monitoranpassung: Gibt Auskunft über die verwendete Taktfrequenz (Signal QT, siehe Abschnitt 5.2)



### 1.1.7 Technische Daten

**Stromversorgung:**

+	5 V	$\pm 5\%$	<	700 mA
+	12 V	$\pm 5\%$	<	90 mA
-	12 V	$\pm 5\%$	<	80 mA

## 1.1.8 Zusammenfassung der Adressen und Befehle

### 1.1.8.1 Bildschirmsteuerung

#### Funktionen des Steuerbausteins

Funktion	Tor	Z80-Befehl
Zeitsteuerung		
stoppen	2A	OUT 2AH
starten	2E	OUT 2EH
Steuerregister	20...26	OUT 20H ... OUT 26H
Schirmbild rollen	2B	OUT 2BH

(Die nicht angegebenen Adressen betreffen die auf der Kombinationskarte nicht genutzten Cursor- und automatischen Ladefunktionen des Steuerbausteins.)

**Programmierung der Video-Steuerung (VTAC 5027/TMS9927)**

Register- nummer	Bit- nummer	programmierte Funktion																																				
20	7...0	Zeilenlänge, berechnet nach:  <div style="text-align: center;">Videotakt</div> <p>-----                      Zeichenbreite x Rasterzeilen x Bildfrequenz</p> dabei ist vorgegeben (europäische Norm):  Zeichenbreite = 8 Rasterzeilen = 312 (ohne Zeilensprung) 625 (mit Zeilensprung) Bildfrequenz = 50 (ohne Zeilensprung) 25 (mit Zeilensprung)																																				
21	7	Rastermodus mit Zeilensprung = 1 ohne Zeilensprung = 0																																				
	6...3	Breite des Zeilensynchronisierimpulses in Einheiten der Zeichenbreite																																				
	2...0	Abstand des rechten Bildrand zum Zeilensyn- chronisierimpuls, gemessen in Einheiten der Zeichenbreite																																				
22	6...3 2...0	Rasterzeilen pro Zeichen (- 1) Anzahl darstellbarer) Zeichen pro Zeile verschlüsselt nach:																																				
		<table border="0" style="margin-left: 40px;"> <thead> <tr> <th>Bit 2</th> <th>Bit 1</th> <th>Bit 0</th> <th>Zeichenzahl</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>20</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>32</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>40</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>64</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>72</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>80</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>96</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>132</td></tr> </tbody> </table>	Bit 2	Bit 1	Bit 0	Zeichenzahl	0	0	0	20	0	0	1	32	0	1	0	40	0	1	1	64	1	0	0	72	1	0	1	80	1	1	0	96	1	1	1	132
Bit 2	Bit 1	Bit 0	Zeichenzahl																																			
0	0	0	20																																			
0	0	1	32																																			
0	1	0	40																																			
0	1	1	64																																			
1	0	0	72																																			
1	0	1	80																																			
1	1	0	96																																			
1	1	1	132																																			
23	7, 6	Verzögerungsfaktor der Austastsignale und des Cursors gegenüber der normalen Schirm- position (in Einheiten der Zeichenbreite), verschlüsselt nach:																																				

Bit 7	Bit 6	Austastsignal verzögert um Zeichen:	Kursor
0	0	0	0
0	1	1	0
1	0	2	1
1	1	2	2

5...0 Anzahl der Zeichenzeilen (- 1)

24 7...0 Anzahl der Rasterzeilen pro Schirmbild  
verschlüsselt nach:

mit Zeilensprung:

$$\frac{\text{Rasterzeilen} - 513}{2}$$

ohne Zeilensprung:

$$\frac{\text{Rasterzeilen} - 256}{2}$$

25 7...0 Anzahl der leeren Rasterzeilen am Bild-  
anfang (zwischen dem Anfang des Bildsyn-  
chronisierimpulses und der ersten Daten-  
zeile)

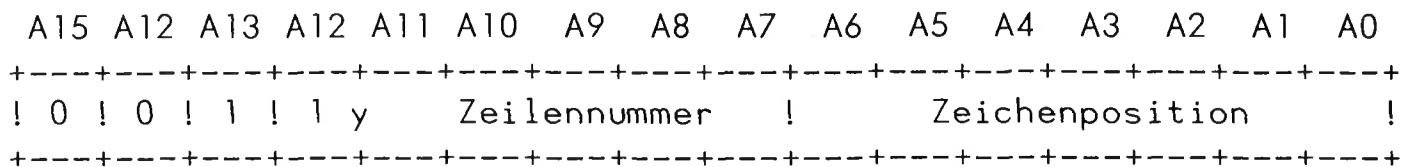
26 5...0 Nummer der letzten auf dem (vollen) Schirm  
angezeigten Zeichenzeile. Die Zählung be-  
ginnt dabei mit 0.

### Beispiel (24x80-Anzeige)

Bei einem Videotakt von 12,8112 MHz, wie er von dem Standard-Monitoradapter (s. Abschnitt 5.2.1) geliefert wird und der Arbeit ohne Zeilensprungverfahren, erhält man mit den folgenden Werten eine ruhige Anzeige von 24 Zeilen zu 80 Zeichen mit Zeile 0 als erster, Zeile 23 als letzter Schirmzeile:

Register- nummer	Wert (hexadezimal)	Z80-Programmbefehle (8080-kompatibel)
20	65	LD A,6DH OUT 20H,A
21	2E	LD A,2EH OUT 21H,A
22	5D	LD A,5DH OUT 22H,A
23	57	LD A,57H OUT 23H,A
24	1D	LD A,1DH OUT 24H,A
25	12	LD A,12H OUT 25H,A
26	17	LD A,17H OUT 26H,A

**Adreßformat zum Zugriff auf den Bildwiederholtspeicher**



Diese effektive 12-Bit-Adresse wird auf der Karte über einen Multiplexer auf eine 11-Bit-Adresse reduziert und mit dieser der Speicher angesprochen. Die Adreßzuordnung stellt sich dabei wie folgt dar:

**Interne Zeichenadressen im Bildwiederholtspeicher**

Zeilen- nummer	Zeichenposition	
	0 bis 63	64 bis 79
0	000 ... 03F	600 ... 60F
1	040 ... 07F	640 ... 64F
.	.	.
.	.	.
.	.	.
15	3C0 ... 3FF	7D0 ... 7DF
16	400 ... 43F	620 ... 62F
17	440 ... 47F	660 ... 66F
.	.	.
.	.	.
.	.	.
23	5C0 ... 5FF	7E0 ... 7EF
24	600 ... 63F	
.	.	
.	.	
.	.	
31	7C0 ... 7FF	

**Wichtig:**  
 Der Speicherbereich für die Zeichen Nummer 64 bis 79 in Zeile 0 bis 23 und der für die Zeichen Nummer 0 bis 63 in Zeile 24 bis 31 ist identisch. Das bedeutet, daß die größtmöglichen Darstellungsformate entweder **32 Zeilen zu 64 Zeichen** oder **24 Zeilen zu 80 Zeichen** sind.

### Video-Synchronisierungsinformation (IN 35H)

Bit	Bezeichnung	Funktion
6	VSYN	1, wenn die vertikale Austastlücke (Bild-austastlücke) erreicht wurde
7	BL	1, wenn die horizontale (Zeilen-) oder die vertikale Austastlücke (Bildaustastlücke) erreicht worden ist

#### Beispiel:

```
VIDOUT: IN  A,(35H)      ; Status abfragen
          BIT  6,A        ; ist VSYN (Bit 6) gesetzt?
          JR   Z,VIDOUT   ; nein: weiterwarten
          LD   (HL),C     ; sonst Zeichen ausgeben
          RET              ; und zurück zum Aufruf
```

### Brücken zur Festlegung der Zahl der Abtastzeilen

Abtastzeilen pro Zeichen	Brücken
12	b, d geschlossen a, c offen
16	b, d offen a, c geschlossen

### Lupenfunktion (OUT 34H)

Bit 0 = 1: Lupe eingeschaltet

Bit 0 = 0: Lupe ausgeschaltet

**Wichtig:** Die Lupe wirkt auf den Schirm als Ganzes, nicht nur auf ein einzelnes Zeichen auf dem Schirm. Das bedeutet, daß die Zahl der auf dem Schirm darstellbaren Zeichen geringer wird. Das Format 80 x 24 wird zu 40 x 12, wobei das obere linke Schirmviertel angezeigt wird (falls Zeile 0 ganz oben auf dem Schirm steht).

**Vorsicht beim Auf- und Abrollen des Schirmbilds, wenn die Lupe eingeschaltet ist!** Durch die Umsetzung der Adressen für den Bildwiederholtspeicher verhält sich das Schirmbild recht kompliziert, wenn mit der Hardware-Rollfunktion (Register 26 oder 2B des Steuerbausteins) gearbeitet wird. Am sichersten setzt man Zeile 0 nach oben und rollt das Bild dann bei Bedarf über Software durch Umspeichern der Information im Bildwiederholtspeicher.



**Zeichenkodes zur Graphikwiedergabe**

(Ein "0" gibt einen hellen, ein "." einen dunklen Punkt wieder.)

..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
..	..	..	..	..	..	..	..	..	..	..	..	..	..	..	..
00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0	.0
9F	9E	9D	9C	9B	9A	99	98	97	96	95	94	93	92	β	90
..	0.	.0	00	..	0.	.0	00	..	0.	.0	00	..	0.	.0	00
..	..	..	..	0.	0.	0.	0.	.0	.0	.0	.0	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
8F	8E	8D	8C	8B	8A	89	88	87	86	85	84	83	82	81	80

**Amerkung:** Die dadurch erreichte Auflösung auf dem Schirm ist nicht symmetrisch. Das Schirmbild hat ein Höhen-zu-Breiten-Verhältnis von 3:4, was bei einer Auflösung von 72x160 Punkten ergibt, daß einem Schritt in waagrechtlicher Richtung 3/5 Schritte in senkrechter Richtung entsprechen. D.h. das Punktverhältnis beträgt (x = Breite, y = Höhe):

$$x:y = 3:5$$

(Man erhält also ein Quadrat, wenn man einen Block mit 5 Punkten Breite und 3 Punkten Höhe zeichnet.)

Nationale Zeichensätze

Kode (hex)	Zeichen			Kode (hex)	Zeichen			Kode (hex)	Zeichen		
	dt.	engl.	frz.		dt.	engl.	frz.		dt.	engl.	frz.
20	Leerschritt			40	§	Ø	à	60	`	`	`
21	!	!	!	41	A	A	A	61	a	a	a
22	"	"	"	42	B	B	B	62	b	b	b
23	#	£	#	43	C	C	C	63	c	c	c
24	\$	\$	\$	44	D	D	D	64	d	d	d
25	%	%	%	45	E	E	E	65	e	e	e
26	&	&	&	46	F	F	F	66	f	f	f
27	'	'	'	47	G	G	G	67	g	g	g
28	(	(	(	48	H	H	H	68	h	h	h
29	)	)	)	49	I	I	I	69	i	i	i
2A	*	*	*	4A	J	J	J	6A	j	j	j
2B	+	+	+	4B	K	K	K	6B	k	k	k
2C	,	,	,	4C	L	L	L	6C	l	l	l
2D	-	-	-	4D	M	M	M	6D	m	m	m
2E	.	.	.	4E	N	N	N	6E	n	n	n
2F	/	/	/	4F	O	O	O	6F	o	o	o
30	0	0	0	50	P	P	P	70	p	p	p
31	1	1	1	51	Q	Q	Q	71	q	q	q
32	2	2	2	52	R	R	R	72	r	r	r
33	3	3	3	53	S	S	S	73	s	s	s
34	4	4	4	54	T	T	T	74	t	t	t
35	5	5	ß	55	U	U	U	75	u	u	u
36	6	6	6	56	V	V	V	76	v	v	v
37	7	7	7	57	W	W	W	77	w	w	w
38	8	8	8	58	X	X	X	78	x	x	x
39	9	9	ß	59	Y	Y	Y	79	y	y	y
3A	:	:	:	5A	Z	Z	Z	7A	z	z	z
3B	;	;	;	5B	Ä	[	°	7B	ä	{	é
3C	<	<	<	5C	Ö	\	ç	7C	ö		ù
3D	=	=	=	5D	Ü	]	&	7D	ü	}	è
3E	>	>	>	5E	^	^	^	7E	ß	~	“
3F	?	?	?	5F	_	_	_	7F	⊠	⊠	⊠

### 1.1.8.2 Tastatursteuerung

#### Adressen der 8278-Register zur Tastatursteuerung

Funktion	E-A-Adresse	Z80-Befehl
Datenbusregister lesen	10H	IN A, (10H)
Status lesen	11H	IN A, (11H)
Befehl ausgeben	11H	OUT (11H), A

#### Befehlsarten für den 8278-Baustein (Tastatursteuerung)

Bit 7	Bit 6	Bit 5	Befehlsart
0	0	0	Betriebsart setzen
0	1	0	FIFO lesen
1	0	1	Löschbefehl

#### Format des Befehlsbytes (OUT 11H):

7	6	5	4	3	2	1	0
+	+	+	+	+	+	+	+
!	0	!	0	!	0	!	0
!	!	!	!	!	!	!	!
!	!	!	!	!	!	!	!
+	+	+	+	+	+	+	+

Bit	Bezeichnung	Bedeutung
4 und 1	7-Seg.-Anzeige	Anzeigensteuerung (hier nicht benutzt)
3	Fehlerbehandlung	0 = Fehler bei Mehrfachbe- tätigung von Tasten mel- den 1 = keinen Fehler in diesem Fall melden (N-key roll- over)
2	Unterbrechungs- anforderung	0 = Unterbrechung bei Zei- chen im FIFO und bei Fehler anfordern (über INT0, Anschluß 62) 1 = keine Unterbrechung an- fordern
0	Tastenerkennung	0 = Normalbetrieb, jede Taste nur einmal melden 1 = jede Taste einmal melden, wenn sie gedrückt wurde, und einmal, wenn sie los- gelassen wurde

### FIFO lesen

**Befehlsbyte:** 40H

**Wirkung:** Das unterste FIFO-Register wird in das Datenbusregister übertragen und der FIFO-Inhalt einen Platz nach unten geschoben. Der Zähler (im Statusregister) wird heruntergezählt.  
Man erhält den FIFO-Inhalt anschließend durch Auslesen des Datenbusregisters.

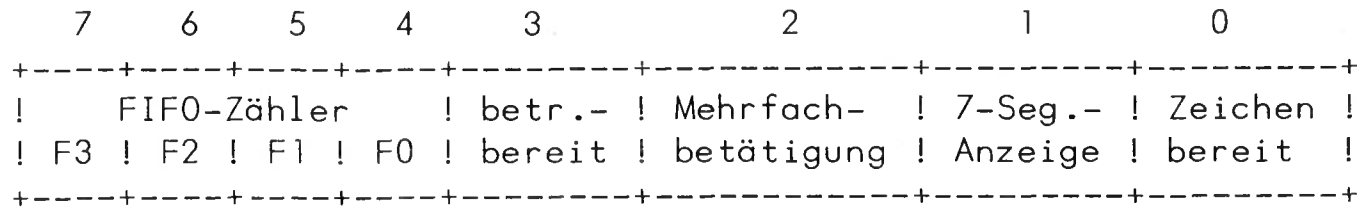
**Fehler:** Wenn das FIFO leer ist, wird das zuletzt ausgelesene Zeichen übergeben. Der Zähler bleibt auf Null, und es wird kein Fehler gemeldet.

### Löschbefehl

Es gibt im Rahmen der Tastatursteuerung zwei Löschbefehle:

<b>Befehl</b>	<b>Befehlsbyte</b>
Fehlerflagge (ERROR) löschen	0CEH
FIFO-Inhalt löschen	0CFH

**Format des Statusbytes (IN 11H):**



Bit	Bezeichnung	Bedeutung
7...4	FIFO-Zähler	0000            FIFO ist leer 0001...0111 = Zeichen im FIFO 1111            FIFO-Überlauf
3	betriebsbereit	1 = 8278 ist beschäftigt 0 = 8278 ist betriebsbereit, d.h. Daten sind gültig
2	Mehrfachbetätigung	1 = mehrere Tasten sind zu- gleich gedrückt (sperrt FIFO bis Fehlerflagge ausdrücklich gelöscht wurde; s. Löschbefehl; nur wenn Betriebsart Bit 3 gelöscht ist!) 0 = Keine Mehrfachbetäti- gung oder Mehrtasten- trennung gewählt
1	7-Seg.Anzeige	(nicht benutzt)
0	Zeichen bereit	1 = (mindestens) ein Zeichen liegt im FIFO vor 0 = kein Zeichen ist ver- fügbar

**Algorithmus zur Zeichenübernahme von der Tastatur**

```

Befehl "FIFO lesen" ausgeben,
Wiederholen:
    Statusregister lesen.
Bis Statusbit 3 (betriebsbereit) = 0.
Wiederholen:
    Wiederholen:
        Statusregister lesen.
        Bis Statusregister (Zeichen bereit) = 1,
        Datenbusregister (d.h. FIFO) lesen.
Bis genug Zeichen übernommen sind.

```

### Fehlersituationen bei der Tastatursteuerung

Fehler	Kennzeichen	bei Betriebsart
Mehrfachbetätigung	Statusbit 2 = 1	Befehlsbit 3 = 0
FIFO-Überlauf	Status 7...4 = 1111	immer
FIFO-Unterlauf	Status 7...4 = 0000	<b>Keine besondere Fehlermeldung!</b>
Baustein nicht bereit	Statusbit 3 = 1, Datenbusregister ist ungültig	<b>Keine besondere Fehlermeldung!</b>

Die Fehlerflagge wird gelöscht durch:

- Ausgabe eines Löschbefehls (siehe Abschnitt 1.4.4.5.2.3)
- einen Hardware-RESET-Befehl.

### 1.1.8.3 Allgemeine Karteneigenschaften

#### Adressen der Signaleinheiten

Funktion	Ausgabeter	Z80-Befehl
Hupe	32	OUT (32H),A
LED	33	OUT (33H),A

Dabei bestimmt Bit 0 im ausgegebenen Byte die Funktion:

- Bit 0 = 1: Hupe bzw. LED einschalten
- Bit 0 = 0: Hupe bzw. LED ausschalten

#### Allgemeines Statustor (IN 35H)

Bit	Bezeichnung	Funktion
7	BL	(blank) allgemeines Austastsignal (Zeilen- und Bildsynchronisation)
6	VSYN	(vertical synchronisation) Bild-austastsignal
5...3	STAT5...STAT3	Statusinformationen von der Tastatur: Gibt Auskunft über die eingesezte Tastatur (siehe Abschnitt 5.3)
2...0	STAT2...STAT0	Statusinformation von der Monitoranpassung: Gibt Auskunft über die verwendete Taktfrequenz (Signal QT, siehe Abschnitt 5.2)

### 1.1.9 Steckerbelegung

#### Reihe a

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	+ 12 V	Versorgungsspannung			
4	0 V	Masserückleitung, + 12 V			
5	- 12 V	Versorgungsspannung			
6	0 V	Masserückleitung, - 12 V			
7	M1	Kenner für Maschinenzklus 1	E	positiv	
8	SR/KOMB $\bar{I}$	Umschaltung Speicher-Kombi.	E	positiv	
9	$\bar{A}D\bar{R}^0$	Adreßbit 0	E	negativ	
10	$\bar{A}D\bar{R}^1$	Adreßbit 1	E	negativ	
11	$\bar{A}D\bar{R}^2$	Adreßbit 2	E	negativ	
12	$\bar{A}D\bar{R}^3$	Adreßbit 3	E	negativ	
13	$\bar{A}D\bar{R}^4$	Adreßbit 4	E	negativ	
14	$\bar{A}D\bar{R}^5$	Adreßbit 5	E	negativ	
15	$\bar{A}D\bar{R}^6$	Adreßbit 6	E	negativ	
16	$\bar{A}D\bar{R}^7$	Adreßbit 7	E	negativ	
17	$\bar{A}D\bar{R}^8$	Adreßbit 8	E	negativ	TS
18	$\bar{A}D\bar{R}^9$	Adreßbit 9	E	negativ	TS
19	$\bar{A}D\bar{R}^{10}$	Adreßbit 10	E	negativ	TS
20	$\bar{A}D\bar{R}^{11}$	Adreßbit 11	E	negativ	TS
21	$\bar{A}D\bar{R}^{12}$	Adreßbit 12	E	negativ	TS
22	$\bar{A}D\bar{R}^{13}$	Adreßbit 13	E	negativ	TS
23	$\bar{A}D\bar{R}^{14}$	Adreßbit 14	E	negativ	TS
24	$\bar{A}D\bar{R}^{15}$	Adreßbit 15	E	negativ	TS
25	$\bar{D}A\bar{T}^0$	Datenbit 0	B	negativ	TS
26	$\bar{D}A\bar{T}^1$	Datenbit 1	B	negativ	TS
27	$\bar{D}A\bar{T}^2$	Datenbit 2	B	negativ	TS
28	$\bar{D}A\bar{T}^3$	Datenbit 3	B	negativ	TS
29	$\bar{D}A\bar{T}^4$	Datenbit 4	B	negativ	TS
30	$\bar{D}A\bar{T}^5$	Datenbit 5	B	negativ	TS
31	$\bar{D}A\bar{T}^6$	Datenbit 6	B	negativ	TS
32	$\bar{D}A\bar{T}^7$	Datenbit 7	B	negativ	TS

-----  
**positive Logik:** 1=H, 0=L

**Treiber:** TS = tri state

**negative Logik:** 1=L, 0=H

OC = open collector

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional



**Reihe b**

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
33	+ 5 V	Versorgungsspannung			
34	0 V	Masserückleitung, + 5 V			
35	+ 12 V	Versorgungsspannung			
36	0 V	Masserückleitung, + 12 V			
37	- 12 V	Versorgungsspannung			
38	0 V	Masserückleitung, - 12 V			
39					
40					
41	$\overline{MEMR}$	Speicher lesen (Anforderung)	E	negativ	OC
42	$\overline{MEMW}$	Speicher schreiben (Anf.)	E	negativ	OC
43	$\overline{IOR}$	Eingabetor lesen (Anf.)	E	negativ	OC
44	$\overline{IOW}$	Ausgabeter schreiben (Anf.)	E	negativ	OC
45					
46					
47	$\overline{RST}$	internes Rücksetzsignal	E	negativ	
48					
49					
50					
51					
52					
53					
54	IORQ	E-A-Tor-Anforderung	E	positiv	
55	MREQ	Speicheranforderung	E	positiv	
56					
57					
58					
59					
60					
61					
62	$\overline{INTO}$	Unterbrechungszeiger	A	negativ	
63					
64					

-----  
**positive Logik:** 1=H, 0=L

**negative Logik:** 1=L, 0=H

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

**Treiber:** TS = tri state

OC = open collector

Reihe c

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
65	M0	Matrixabtastung, Leitung 0	A	positiv	
66	M1	Matrixabtastung, Leitung 1	A	positiv	
67	M2	Matrixabtastung, Leitung 2	A	positiv	
68	M3	Matrixabtastung, Leitung 3	A	positiv	
69	M4	Matrixabtastung, Leitung 4	A	positiv	
70	M5	Matrixabtastung, Leitung 5	A	positiv	
71	M6	Matrixabtastung, Leitung 6	A	positiv	
72	$\overline{RL}$	Tastatur-Rückmeldung	E	negativ	
73					
74	STAT2	Statussignal 2	E	positiv	
75	STAT1	Statussignal 1	E	positiv	
76	STAT0	Statussignal 0	E	positiv	
77					
78					
79					
80					
81					
82					
83	$\overline{HSYN}$	horizontale Synchronisation	A	negativ	
84	$\overline{VSYN}$	vertikale Synchronisation	A	negativ	
85	$\overline{CSYN}$	kombinierte Synchronisation	A	negativ	
86	VIDEO	Bildsignal	A	positiv	
87					
88					
89					
90					
91					
92	QT	Quarztakt für das Videosignal	E		
93					
94	LED	Ansteuerung der Signallampe	A	positiv	
95	KCL	Abfragetakt für die Tastatur	A	positiv	
96	HUPE	Ansteuerung der Signalhupe	A	positiv	

-----  
**positive Logik:** 1=H, 0=L

**Treiber:** TS = tri state

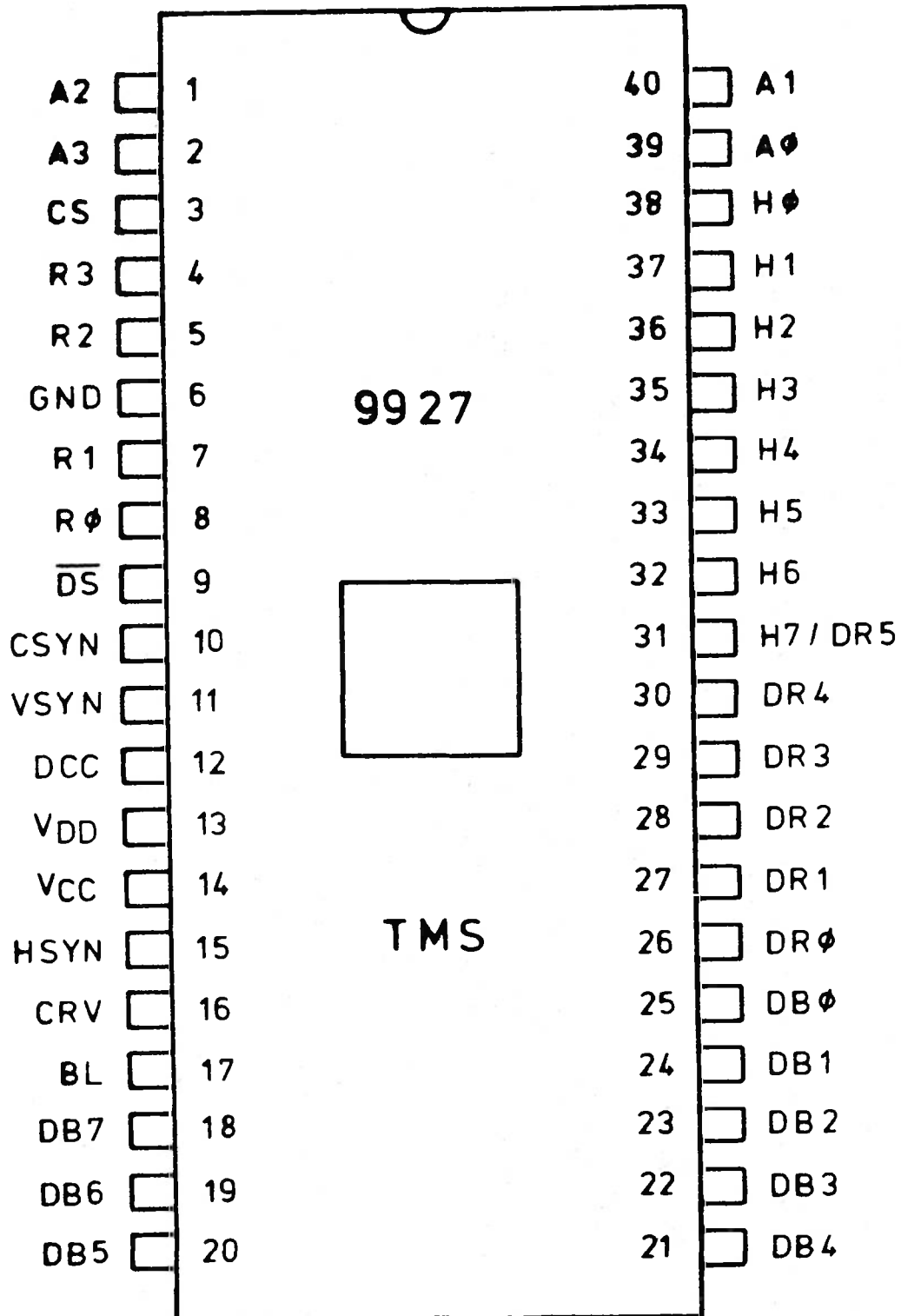
**negative Logik:** 1=L, 0=H

OC = open collector

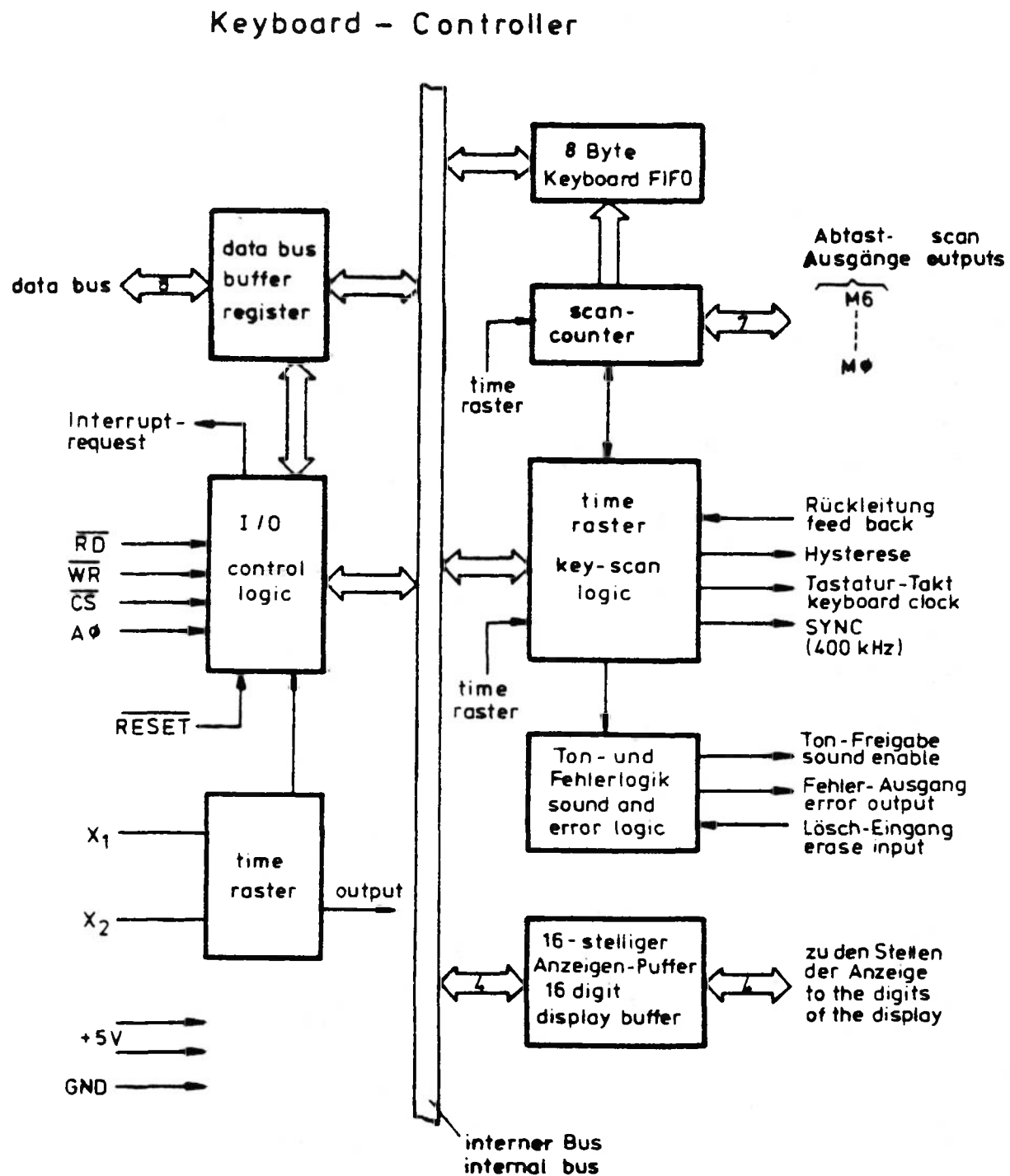
**Richtung:** E = Eingang, A = Ausgang, B = bidirektional



1.1.10.2 Anschlußbelegung

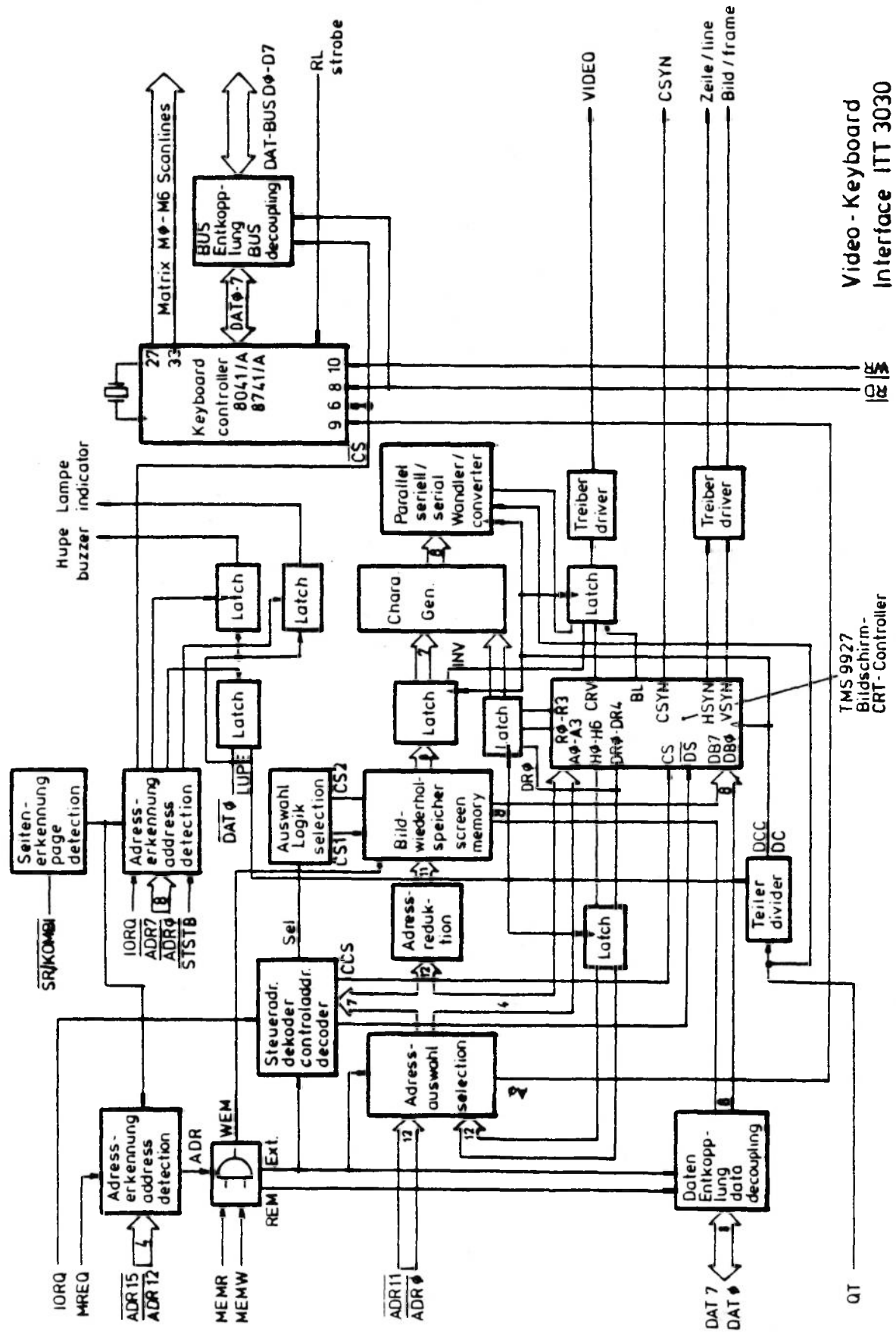


1.1.10.3 Innenschaltung der Tastatursteuerung



Blockschaltbild 8741/8041  
blockdiagram 8741/8041

1.1.10.4 Blockschaltbild



Video-Keyboard Interface ITT 3030

```
*****
*
*           I T T   3 0 3 0
*   SYSTEM- UND SERVICEHANDBUCH
*   -----
*           Abschnitt 5
*   Terminalschnittstellen
*
*****
```

## 2 Monitoranpassung

### 2.1 SW-Monitoradapter

#### 2.1.0 Inhalt

2.1.1 Konzept .....	5-2.1-2
2.1.2 Schnittstelle .....	5-2.1-3
2.1.3 Mischstufe .....	5-2.1-4
2.1.4 Quarzoszillator .....	5-2.1-4
2.1.5 Statuskodierung .....	5-2.1-4
2.1.6 Technische Daten .....	5-2.1-5
2.1.7 Steckerbelegung .....	5-2.1-6
2.1.8 Schaltbild .....	5-2.1-7

### 2.1.1 Konzept

Der Schwarz-Weiß-Monitoradapter hat die Aufgabe, die von der Video-Tastatur-Schnittstelle gelieferten Video- und Synchronisationssignale in ein BAS-Signal zur Ansteuerung eines einfarbigen Videomonitors umzusetzen. Darüberhinaus muß er den Quarztakt zur Videosignalaufbereitung an die Video-Tastatur-Schnittstelle liefern.

Dazu umfaßt der Monitoradapter folgende Baugruppen:

- eine Mischstufe zur Erzeugung des BAS-Signals,
- einen Quarzoszillator ( $12,8112 \text{ MHz} \pm 10^{-4}$ ) und
- drei Kodierbrücken zur Verschlüsselung der Taktfrequenz.



### 2.1.2 Die Schnittstelle

Der Monitoradapter wird auf das Verdrahtungsfeld ganz rechts auf Steckplatz XI (die querliegende schmale Buchsenleiste) aufgesteckt. Dieser Steckplatz ist mit dem Busstecker I (Video-Tastaturschnittstelle, vorne links) verbunden und trägt 17 Signale, von denen die folgenden neun vom Monitoradapter ausgewertet bzw. bereitgestellt werden:

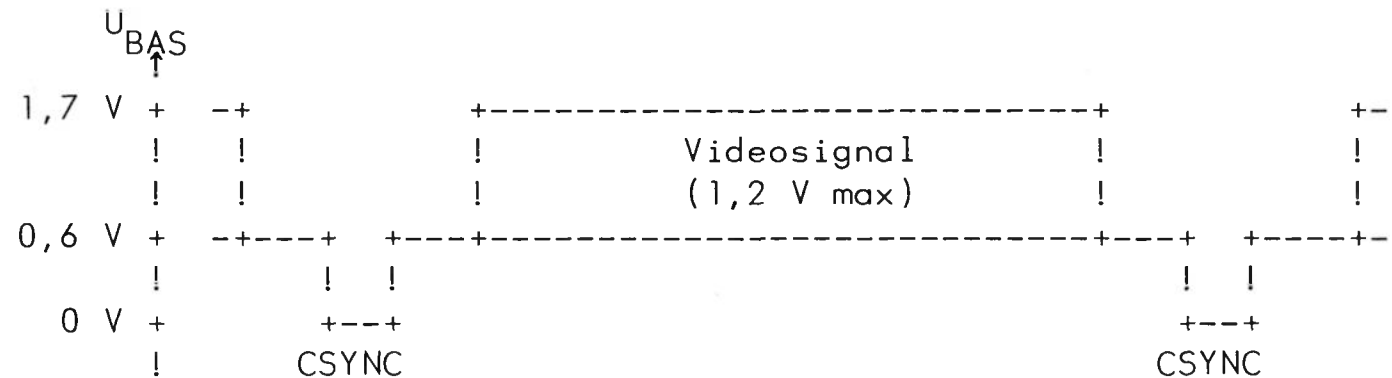
- die Versorgungsspannung + 5 V und eine Masseleitung,
- das kombinierte Synchronisiersignal  $\overline{CSYN}\overline{C}$  (Zeilen- und Bildaustastung; TTL-Pegel),
- das Videosignal VIDEO, das in TTL-Pegel die Bildinhalts-Information trägt,
- der Quarztakt  $\overline{QT}$  zur Videosignalaufbereitung in der Video-Tastatur-Schnittstelle,
- drei Statussignale STAT0...STAT2, mit denen die Videotaktfrequenz kodiert angegeben wird (hier:  $111_2$ ) und
- das BAS-Signal (zu Testzwecken).

### 2.1.3 Die Mischstufe

Die Mischstufe besteht aus zwei Emitterfolgern mit gemeinsamem Emitterzweig. Sie werden gepuffert über TTL-Inverter mit  $\overline{CSYN}\overline{C}$  bzw. VIDEO versorgt. Das BAS-Signal wird am gemeinsamen Emitterwiderstand abgenommen und auf den Busstecker sowie auf eine BAS-Buchse, den externen Monitoranschluß, geführt.

Die Widerstände in der Mischstufe sind so eingestellt, daß sich folgende Signalspannungen ergeben:

- BAS-Signal bei Vollaussteuerung:  $1,7 \text{ V} \pm 15 \%$
- Synchronisiersignal:  $0 \dots 0,6 \text{ V} \pm 15 \%$
- Videosignal:  $0,6 \dots 1,7 \text{ V} \pm 15 \%$



### 2.1.4 Quarzoszillator

Ein aus Invertern aufgebauter quarzgesteuerter Oszillator erzeugt eine Frequenz von  $12,8112 \text{ MHz} \pm 10^{-4}$ . Dieser Takt wird über einen weiteren Inverter ausgekoppelt und über Anschluß 13 als Signal  $\overline{QT}$  an die Video-Tastatur-Schnittstelle weitergeleitet.

### 2.1.5 Statuskodierung

Da die Programmierung der Videosteuerung vom Versorgungstakt QT abhängt, wird über drei Leitungen eine Kodierung bereitgestellt. Sie ist über die Brücken a, b und c auf der Karte einstellbar. Diese befinden sich auf der Leiterbahnseite links unten neben dem Busstecker. Sie sind bei der hier verwendeten Taktfrequenz alle offen, leiten also das Signal  $111_2$  über die Statusausgänge STAT0...STAT2 (Anschluß 14...16) an die Video-Tastatur-Schnittstelle weiter und können dort über ein Statustor abgefragt werden (siehe Abschnitt 5.1).

### 2.1.6 Technische Daten

Stromversorgung:	+ 5 V < 30 mA			
Video-Versorgungstakt:	12,8112 MHz	$\pm 10^{-4}$		
Videokodierung (Status):	111 <sub>2</sub>			
BAS-Signal:	0 ... 1,7 V	$\pm 15\%$		maximal
	0 ... 0,6 V	$\pm 15\%$		Synchronisierung
	0,6 ... 1,7 V	$\pm 15\%$		Bildinformation (Videosignal)

### 2.1.7 Steckerbelegung

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3					
4					
5					
6					
7					
8	$\overline{CS}\overline{Y}\overline{N}\overline{C}$	Kombiniertes Synchronsignal	E	negativ	
9	VIDEO	Videosignal (Bildinhalt)	E	positiv	
10					
11					
12					
13	QT	Quarztakt	A		
14	$\overline{ST}\overline{A}\overline{T}\overline{0}$	Statussignal 0	A	positiv	
15	$\overline{ST}\overline{A}\overline{T}\overline{1}$	Statussignal 1	A	positiv	
16	$\overline{ST}\overline{A}\overline{T}\overline{2}$	Statussignal 2	A	positiv	
17	BAS	BAS-Signal zum Monitor			

-----  
**positive Logik:** 1=H, 0=L

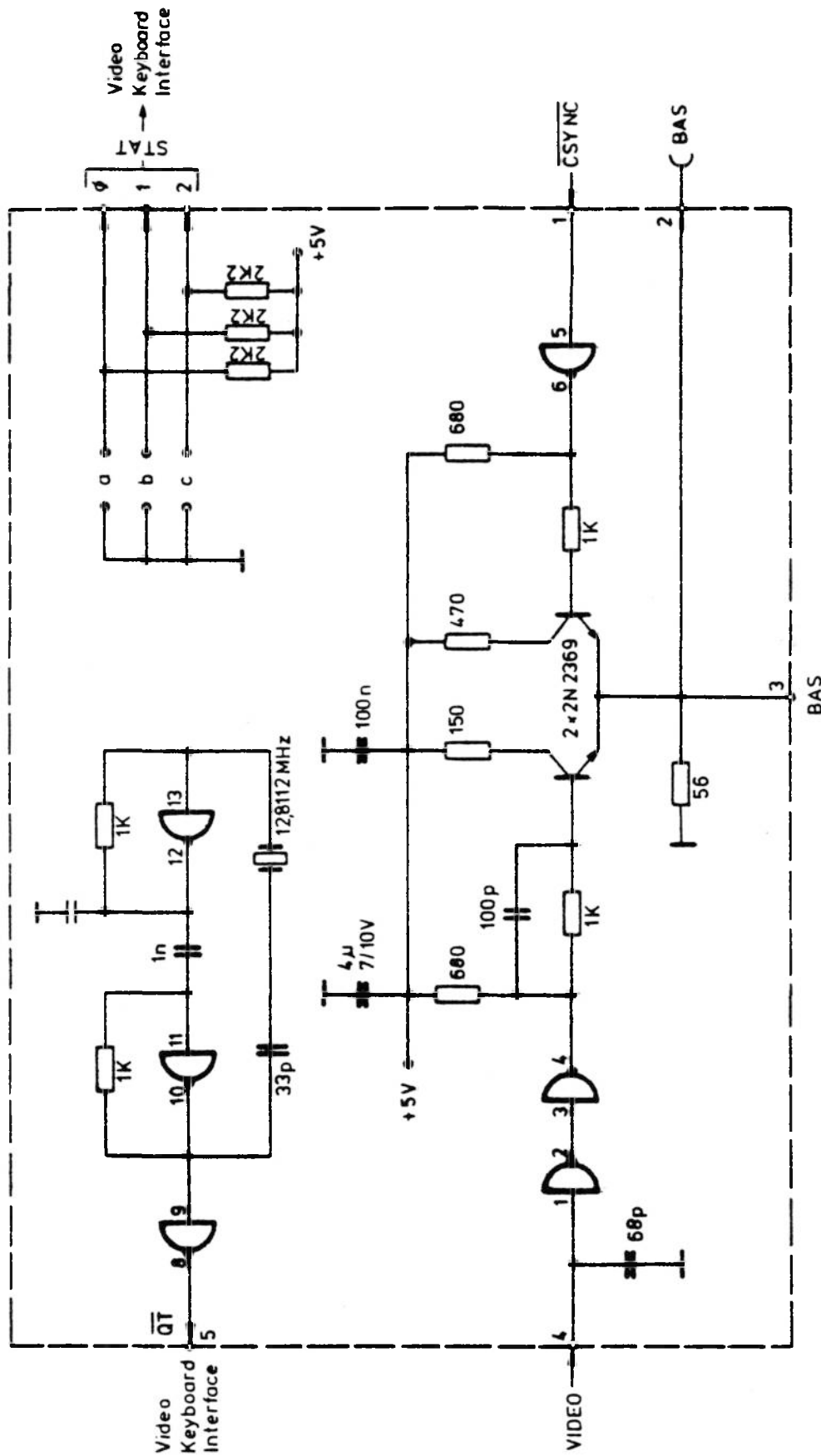
**negative Logik:** 1=L, 0=H

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

**Treiber:** TS = tri state

OC = open collector

2.1.8 Schaltbild



s/w Monitor - Adapter  
b/w monitor - adapter

```
*****
*
*           I T T   3 0 3 0
*   SYSTEM- UND SERVICEHANDBUCH
*   -----
*           Abschnitt 5
*       Terminalschnittstellen
*
*****
```

### 3 Tastaturanpassung

#### 3.1 Tastaturadapter I

##### 3.1.0 Inhalt

3.1.1 Konzept .....	5-3.1-2
3.1.2 Schnittstellen .....	5-3.1-3
3.1.3 Der Wandler für die Abtastsignale .....	5-3.1-4
3.1.4 Die Hupe .....	5-3.1-5
3.1.5 Technische Daten .....	5-3.1-6
3.1.6 Steckerbelegung .....	5-3.1-7
3.1.7 Bestückung .....	5-3.1-8

### 3.1.1 Konzept

Der Tastaturadapter I hat die Aufgabe, die sieben von der Video-Tastatur-Schnittstelle gelieferten Abtastleitungen M0...M6 für die Tastaturmatrix in serielle Form zu bringen und so an die Tastaturelektronik weiterzugeben. Weiter enthält er eine Hupe, die ebenfalls durch ein Signal von der Video-Tastatur-Schnittstelle aus angesteuert wird.

Damit umfaßt der Tastaturadapter I folgende Baugruppen:

- den parallel-seriell-Wandler für die Abtastsignale und
- die Hupenschaltung.

### 3.1.2 Schnittstellen

Der Tastaturadapter wird auf das Verdrahtungsfeld ganz links auf Steckplatz VII (die querliegende schmale Buchsenleiste hinter der Haltebügelverriegelung) aufgesteckt. Dieser Steckplatz ist einerseits mit dem Busstecker I (Video-Tastatur-Schnittstelle, vorne links) andererseits mit dem Tastaturanschluß auf Steckplatz VIII (die querliegende schmale Steckerleiste vor der Haltebügelverriegelung) verbunden. Die Schnittstelle zur Video-Tastatur-Karte trägt 12 Signale, die zum Tastaturanschluß zwei. Im Einzelnen liegen hier an:

- Video-Tastatur-Schnittstelle:
  - = die Versorgungsspannung + 5 V und eine Masseleitung,
  - = die Abtastleitungen M0...M6 für die Tastaturmatrix,
  - = der Abfragetakt KCL für die Tastatur,
  - = die systeminterne Rücksetzleitung  $\bar{R}\bar{S}\bar{T}$  und
  - = die Hupenansteuerung HUPE.
- Tastaturanschluß-Schnittstelle:
  - = eine serielle Datenleitung  $\bar{D}\bar{A}\bar{T}\bar{A}$  und
  - = eine Taktleitung  $\bar{C}\bar{L}$ .



### 3.1.3 Der Wandler für die Abtastsignale

Die Video-Tastatur-Schnittstelle liefert für jeden Schnittpunkt zwischen Zeilen- und Spaltenleitungen auf der Tastaturmatrix ein Abtastsignal auf den Leitungen M0...M6, das seriell an die Tastaturelektronik weitergeleitet werden soll. Die Gültigkeit dieses Abtastsignals wird durch die fallende Flanke des Tastaturtakts KCL festgelegt. Damit hat die Tastaturadapterkarte die Aufgabe, die sechs Abtastsignale zwischen zwei fallenden Flanken des KCL-Signals auf die serielle Datenleitung zu legen.

Dies geschieht mit drei Schaltungsgruppen:

- einem Schieberegister zur Datenwandlung,
- einem Taktgenerator und
- einer Ansteuerlogik.

Im einzelnen geschieht folgendes:

- Mit der fallenden Flanke von KCL (Steckeranschluß 11) wird ein Oszillator (von ca. 770 kHz) gestartet und gleichzeitig ein 16er-Binärzähler auf den Wert 6 gesetzt.
- Der erste Oszillatortakt übernimmt die parallel anliegende Abtastinformation von M0...M6 (Anschlüsse 3 bis 9) in das Schieberegister (das höchstwertige Schieberegisterbit 7 wird dabei auf Null gezogen) und schaltet mit seiner Rückflanke die Ansteuerlogik auf Schiebetrieb um.
- Die nächsten 8 Takte schieben die Information nach rechts (mit den niederwertigen Bits voran) aus. Sie wird über einen invertierenden Puffer auf die Datenleitung  $\overline{DATA}$  gelegt.
- Nach diesen neun Takten stoppt der Zähler beim Übergang von 15 nach 0 den Oszillator und setzt das Schieberegister in parallelen Ladebetrieb zurück. In diesem Zustand wartet die Schaltung auf die nächste fallende KCL-Flanke, die einen neuen Zyklus einleitet.

Wichtig ist dabei, daß die Abtastadresse in negativer Logik (1=L, 0=H) über Anschluß 1 ( $\overline{DATA}$ ) zur Tastatur gesendet wird. Der Schiebetakt wird ebenfalls invertiert als Bezugssignal  $\overline{CL}$  (Steckeranschluß 2) an die Tastatur weitergegeben.

### 3.1.4 Die Hupe

Ein zweiter Start/Stopp-Oszillator wird über das Signal HUPE (Anschluß 15) angesteuert. Eine logische Eins gibt den Oszillator frei, der über einen Puffer ein Signal von ca. 770 kHz an einen Schallwandler weitergibt. Mit einer Null wird die Hupe abgeschaltet.

Die Lautstärke des Hupensignals kann mit einem Trimpotentiometer (an der Kartenoberkante neben der Piezohupe) eingestellt werden.

### 3.1.5 Technische Daten

<b>Stromversorgung:</b>	+ 5 V < 80 mA
<b>Tastaturtakt:</b>	770 kHz $\pm$ 30 %
<b>Hupe:</b>	770 kHz $\pm$ 30 % Lautstärke einstellbar

### 3.1.6 Steckerbelegung

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	M0	Matrixabtastung, Leitung 0	E	positiv	
4	M1	Matrixabtastung, Leitung 1	E	positiv	
5	M2	Matrixabtastung, Leitung 2	E	positiv	
6	M3	Matrixabtastung, Leitung 3	E	positiv	
7	M4	Matrixabtastung, Leitung 4	E	positiv	
8	M5	Matrixabtastung, Leitung 5	E	positiv	
9	M6	Matrixabtastung, Leitung 6	E	positiv	
10					
11	KCL	Abfragetakt für die Tastatur	E	positiv	
12	$\overline{DATA}$	serielle Abtastdaten	A	negativ	
13	$\overline{CL}$	serieller Übertragungstakt	A	negativ	
14	$\overline{RST}$	internes Rücksetzsignal	E	negativ	
15	HUPE	Hupenansteuerung	E	positiv	

-----  
**positive Logik:** 1=H, 0=L

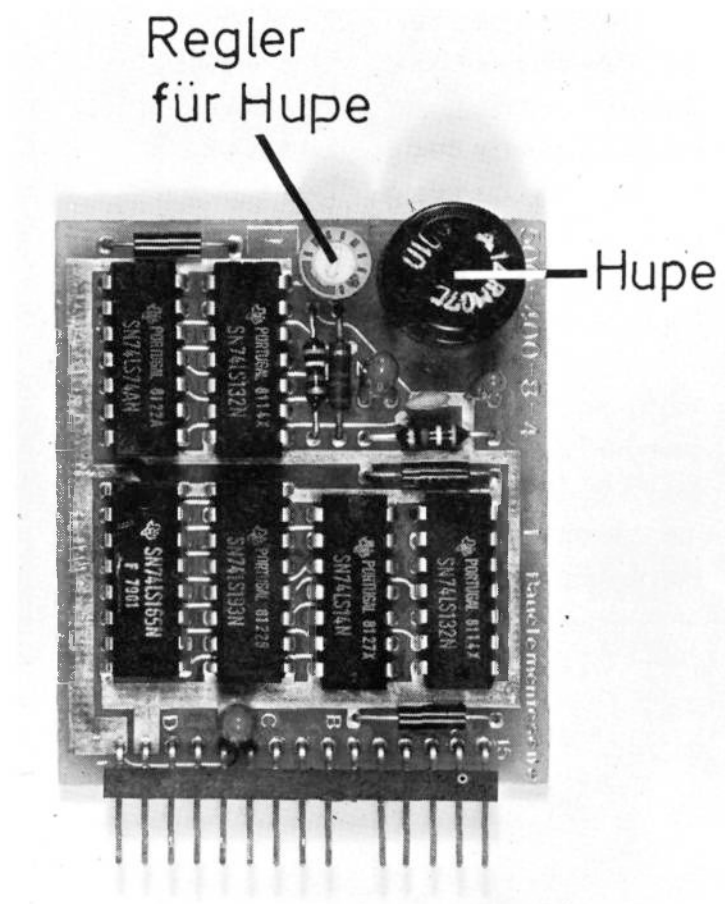
**Treiber:** TS = tri state

**negative Logik:** 1=L, 0=H

OC = open collector

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional

### 3.1.7 Bestückung



```
*****
*
*           I T T   3 0 3 0
*       SYSTEM- UND SERVICEHANDBUCH
*       -----
*           Abschnitt 5
*       Terminalschnittstellen
*
*****
```

## 4 Tastatur

### 4.1 Grund- und Zusatztastatur I

#### 4.1.0 Inhalt

4.1.1 Konzept .....	5-4.1-2
4.1.2 Die Systemschnittstelle .....	5-4.1-3
4.1.3 Die Tastaturelektronik .....	5-4.1-4
4.1.3.1 Der parallel-seriell-Wandler .....	5-4.1-4
4.1.3.2 Die Matrixansteuerung .....	5-4.1-4
4.1.3.3 Die Schaltermatrix .....	5-4.1-5
4.1.4 Mechanischer Aufbau .....	5-4.1-8
4.1.4.1 Öffnen des Tastaturgehäuses .....	5-4.1-8
4.1.4.2 Grundtastatur und Verbindungskabel .....	5-4.1-8
4.1.4.3 Die Zusatztastatur .....	5-4.1-8
4.1.4.4 Der Kabelanschluß in der Zentraleinheit .....	5-4.1-9
4.1.5 Technische Daten .....	5-4.1-10
4.1.6 Steckerbelegung .....	5-4.1-11
4.1.7 Gesamtübersicht über die Tastaturabfrage .....	5-4.1-12

#### 4.1.1 Konzept

Die Grund- und Zusatzastatur besteht aus zwei mechanisch getrennten Einheiten, die jeweils eine Schaltermatrix tragen. Die Schaltermatrix muß von einer Ansteuerelektronik in der ITT-3030-Zentraleinheit abgefragt werden. Eine Dekodierelektronik setzt die seriell übermittelten Abfragesignale in die Ansteuerung der Zeilen- und Spaltenleitungen der Schaltermatrix um und übermittelt der Zentraleinheit Informationen darüber, ob der Tastenschalter auf dem abgefragten Matrixschnittpunkt offen oder geschlossen ist.

Die Tasten sind in keiner Weise bewertet, insbesondere gilt das für die Großbuchstabenumschaltung (SHIFT) und die Steuertaste (CTRL). Es wird lediglich gemeldet, ob die unter dem jeweiligen Ansteuerkode befindliche Taste gedrückt ist oder nicht. Die Bewertung der Tasten muß in der Zentraleinheit erfolgen.

Gegenwärtig werden die beiden Tastaturen in drei Tastenanordnungen geliefert, die der deutschen, der englischen und der französischen Tastaturnorm entsprechen. Dies betrifft jedoch nur die Tastenkennzeichnung, nicht die elektrische Anordnung der Tasten in der Schaltermatrix, die für alle Tastaturauslegungen gleich ist.

Die Tastatur umfaßt folgende Baugruppen:

- die Grundtastatur mit
  - = der Schaltermatrix für 66 Tasten, angeordnet als Standard-Schreibmaschinentastatur und eine Funktionstastenzeile,
  - = einer Anzeige-LED in der "SHIFT-LOCK"-Taste,
  - = einem seriell-parallel-Wandler für die Abfragedaten,
  - = der Ansteuerelektronik zur Abfrage der Schaltermatrix und
  - = einer Schnittstelle zur Zusatzastatur;
- die Zusatzastatur mit der Schaltermatrix für 32 weitere Tasten, angeordnet als ein Cursorsteuerblock, ein Zehner-Eingabefeld und zwei Felder für Funktionstasten;
- das Tastaturgehäuse, bestehend aus einer Oberschale, welche die Tastaturplatinen aufnimmt und einer mit Kugelschnappverbindungen aufgesteckten Bodenplatte.

#### 4.1.2 Die Systemschnittstelle

Die Tastatur ist über ein dehnbares Spiralkabel mit der Zentraleinheit verbunden. Dieses Kabel ist auf beiden Seiten, in der Zentraleinheit und auf der Grundtastatur, steckbar und trägt folgende Signale:

- die Versorgungsspannung + 5 V und zwei Masseleitungen,
- die seriellen Daten zur Matrixabtastung ( $\overline{D\overline{A}\overline{T}\overline{A}}$ ),
- den Übertragungstakt  $\overline{C\overline{L}}$  für diese Daten,
- die Rückleitung  $\overline{R\overline{L}}$ , die den Zustand der abgefragten Taste meldet und
- die Ansteuerleitung LED für die Leuchtdiode.



### 4.1.3 Die Tastaturelektronik

Die Tastaturelektronik besteht aus drei Grundelementen:

- dem seriell-parallel-Wandler,
- der Matrixansteuerung und
- der Schaltermatrix selbst.

Dazu kommt noch die Ansteuerung für die auf der Tastatur untergebrachte Leuchtdiode.

#### 4.1.3.1 Der parallel-seriell-Wandler

Die über die Datenleitung  $\overline{D\overline{A}\overline{T}\overline{A}}$  (Anschluß 7) mit negativer Logik (0 = H, 1 = L) eintreffenden Matrixadressen werden mit der positiven Flanke des Taktsignals  $\overline{C\overline{L}}$  (Anschluß 8) mit dem niederwertigen Bit (M0) voran in ein 8-Bit-Schieberegister eingeschoben.

Dabei werden nur die sieben zur Matrixadressierung benötigten Bits (M0...M7, siehe Abschnitt 5.1 und 5.3) erfaßt. Die eingeschobene Information steht nach jedem Schiebetakt unmittelbar an den parallelen Schieberegisterausgängen zur Verfügung.

#### 4.1.3.2 Die Matrixansteuerung

Die vier höherwertigen Bits (M3...M6) der Matrixadresse werden an zwei 3-zu-8-Dekodierer geschaltet. Das stellt sechzehn Signale zur Ansteuerung der Zeilenleitungen der Tastaturmatrix bereit. Die Leitungen werden jeweils mit einem L-Pegel angesteuert. Sie erfassen sowohl die Grund- als auch die Zusatztastatur.

Die drei niederwertigen Bits (M0...M2) der Matrixadresse steuern einen 8-zu-1-Multiplexer an, der jeweils eine der Spaltenleitungen auf die Rückleitung  $\overline{R\overline{L}}$  durchschaltet. Dabei erfassen

- die Spaltenleitungen 0 bis 4 die Grundtastatur und
- die Spaltenleitungen 5 bis 7 die Zusatztastatur.

Die Rückleitung  $\overline{R\overline{L}}$  wird über einen invertierenden Treiber auf Anschluß 4 des Kabelsteckers gelegt. Sie hat L-Pegel, wenn die gerade angesteuerte Taste gedrückt ist und ansonsten H-Pegel.

**WICHTIG:** Die Tasteninformation liegt asynchron auf der Rückleitung an, d.h. es wird immer eine andere Taste dekodiert, sobald ein Bit in das Schieberegister eingeschoben worden ist. Es ist Aufgabe der Ansteuerlogik in der Zentraleinheit, zu bestimmen, wann der Pegel auf  $\bar{R}\bar{L}$  der gerade mit M0...M6 adressierten Taste entspricht (d.h. wann alle Ansteuerbits übertragen worden sind).

#### 4.1.3.3 Die Schaltermatrix

Die Tastenschalter sind in einer 8x16-Matrix angeordnet, wobei die Grundtastatur 64 der 80 Kreuzungspunkte in den ersten 5 Spalten und die Zusatztastatur 32 der 48 Kreuzungspunkte in den letzten 3 Spalten belegt. (Die Zusatztastatur wertet dabei nur Spalte 6 und 7 aus, die letzte Spalte wird nicht erfaßt.)

Die Tasten sind nach Spalten und Zeilen in folgender Form durchnummeriert:

<Spaltennummer>.<Zeilennummer>

So bedeutet beispielsweise die Angabe 3.11 die elfte Taste in der dritten Spalte.

Diese elektrische Zuordnung in der Schaltermatrix entspricht jedoch nicht der physischen Anordnung der Tasten auf der Tastatur. So liegen auf der Grundtastatur die Tasten von Matrixspalte 1 in der obersten Reihe (der Funktionstastenreihe), während die Tastenreihen im eigentlichen Schreibmaschinentastatenfeld von oben nach unten den Matrixspalten 2, 3, 4 und 5 zugeordnet sind. Die Wagenrückauftaste gehört dabei zu Matrixspalte 3 und die Leertaste zu Matrixspalte 5.

Auf der Zusatztastatur bilden die Tasten der Matrixspalte 6 im wesentlichen den linken Tastenblock und die linke Funktionstastenreihe, während der rechte Block und die rechte Funktionstastenreihe zu Matrixspalte 7 gehören. Matrixspalte 8 ist nicht belegt.

Die Zuordnung der Matrixadressen zu den einzelnen Tasten, wie sie von der Tastaturelektronik ausgewertet werden, ist bedingt durch die Dekodierung komplizierter. Hier ergibt sich folgendes Bild.

**Zuordnung von elektrischen und logischen Matrixadressen**  
 (hexadezimale Werte, nicht belegte Tasten: --)

**Grundtastatur**

Taste	Adr.	Taste	Adr.	Taste	Adr.	Taste	Adr.	Taste	Adr.
1.1	20H	2.1	21H	3.1	22H	4.1	23H	5.1	24H
1.2	28H	2.2	29H	3.2	2AH	4.2	2BH	5.2	2CH
1.3	30H	2.3	31H	3.3	32H	4.3	33H	5.3	34H
1.4	38H	2.4	39H	3.4	3AH	4.4	3BH	5.4	3CH
1.5	00H	2.5	01H	3.5	02H	4.5	03H	5.5	04H
1.6	08H	2.6	09H	3.6	0AH	4.6	0BH	5.6	0CH
1.7	10H	2.7	11H	3.7	12H	4.7	13H	5.7	14H
1.8	18H	2.8	19H	3.8	1AH	4.8	1BH	5.8	1CH
1.9	60H	2.9	61H	3.9	62H	4.9	63H	5.9	64H
1.10	68H	2.10	69H	3.10	6AH	4.10	6BH	5.10	6CH
1.11	70H	2.11	71H	3.11	72H	4.11	73H	5.11	74H
1.12	78H	2.12	79H	3.12	7AH	4.12	7BH	5.12	7CH
1.13	40H	2.13	41H	3.13	42H	4.13	43H	5.13	44H
1.14	48H	2.14	--	3.14	--	4.14	--	5.14	--
1.15	--	2.15	--	3.15	--	4.15	--	5.15	--
1.16	--	2.16	--	3.16	--	4.16	--	5.16	--

**Zusatztastatur**

Taste	Adr.	Taste	Adr.
6.1	25H	7.1	26H
6.2	2DH	7.2	2EH
6.3	35H	7.3	36H
6.4	3DH	7.4	3EH
6.5	05H	7.5	06H
6.6	0DH	7.6	0EH
6.7	15H	7.7	18H
6.8	1DH	7.8	1EH
6.9	65H	7.9	66H
6.10	6DH	7.10	6EH
6.11	75H	7.11	76H
6.12	7DH	7.12	7EH
6.13	45H	7.13	46H
6.14	4DH	7.14	4EH
6.15	55H	7.15	56H
6.16	5DH	7.16	5EH

**WICHTIG:** Es handelt sich bei diesen Tastenadressen um die Codes, die an die Ansteuerelektronik in der Zentraleinheit geliefert werden. Es ist Aufgabe der Zentraleinheit durch Hard- oder Software diese Codes in die für die Computersteuerung benötigten Tastenkodes (beispielsweise nach dem ISO-7-Bit-Standard) umzuwandeln.

#### **4.1.4 Mechanischer Aufbau**

Die Tastaturen sind in einem eigenen Gehäuse getrennt von der Zentraleinheit untergebracht und mit dieser über ein dehnbares Spiralkabel verbunden.

##### **4.1.4.1 Öffnen des Tastaturehäuses**

Die Bodenplatte des Gehäuses ist mit Kugelschnappverschlüssen mit der Oberschale verbunden und kann so ohne Werkzeug geöffnet werden.

Man dreht dazu die Tastatur um, so daß man den Kabelaustritt links vor sich liegen hat. Dann greift man mit beiden Zeigefingern in die Löcher am senkrechten Rand der Bodenmulde (in dieser Lage vorne am Tastaturboden) und löst mit einem kurzen kräftigen Ruck nach oben die Schnappverschlüsse. Die Bodenplatte läßt sich dann leicht aus den hinten liegenden Arretierungsbolzen entfernen.

##### **4.1.4.2 Grundtastatur und Verbindungskabel**

Die Grundtastatur liegt jetzt links im Gehäuse. Sie ist mit sechs Schrauben fest mit der Oberschale verbunden und trägt links vorne den Stecker für das Verbindungskabel zur Zentraleinheit. Dieser Stecker ist kodiert und läßt sich nur in einer Position aufsetzen. Die Zugentlastung des Kabels wird dadurch erreicht, daß man die Schraube der Kabelschelle in das dafür vorgesehene Loch unmittelbar hinter dem Gehäusedurchbruch steckt.

##### **4.1.4.3 Die Zusatztastatur**

Die Zusatztastatur wird rechts an die Grundtastatur gesteckt. Sie wird durch Bolzen in der Oberschale justiert und von der Bodenplatte durch Distanzstreifen im Gehäuse arretiert. Sie läßt sich ohne Werkzeug einsetzen und entnehmen:

- Entfernen Sie vor dem ersten Einsetzen die Abdeckplatten in den Aussparungen der Oberschale.
- Schieben Sie dann leicht schräg von oben die Stecker von Grund- und Zusatztastatur zusammen. Achten Sie dabei darauf,

- daß die Tastatur richtig unter die linken Justier- und Arretierbolzen gleitet.
- Stellen Sie sicher, daß die Löcher in den linken Ecken der Zusatz tastatur ohne zu klemmen auf die dort in der Oberschale befindlichen Justierbolzen passen.
  - Setzen Sie dann die Bodenplatte in die hinteren Arretierbolzen ein und verriegeln Sie sie durch einen kräftigen Druck in den vordern Kugelschnappverschlüssen. Die Zusatz tastatur wird dadurch sicher an ihrem Platz gehalten.

#### **4.1.4.4 Der Kabelanschluß in der Zentraleinheit**

Das Tastaturkabel wird mit Steckplatz VIII des Verdrahtungsfelds verbunden. Dies ist die querliegende schmale Steckerleiste vor der Haltebügelverriegelung. Gehen Sie zum Anschluß der Tastatur wie folgt vor.

- Öffnen Sie das Gehäuse der Zentraleinheit und nehmen Sie gegebenenfalls die beiden Diskettenlaufwerke heraus.
- Stecken Sie das Tastaturkabel durch den länglichen Ausschnitt im Boden der Zentraleinheit links vorne und ziehen Sie das gerade Stück bis zum Beginn der Kabelspirale in das Gehäuse herein.
- Legen Sie das Kabel in den nach links führenden Ausschnitt im Gehäuseboden ein und setzen Sie das Gehäuse wieder auf den Tisch.
- Lösen Sie den Haltebügel für die Diskettenlaufwerke, indem Sie die Haltebügelverriegelung nach vorne schwenken (sie kann u.U. unter den Bügel geschoben und so nicht zu sehen sein). Schwenken Sie den Haltebügel nach vorne.
- Stecken Sie den Kabelstecker auf die schmale Steckerleiste vor der Haltebügelverriegelung auf. (Er ist kodiert und läßt sich daher nur in einer Position stecken.)
- Setzen Sie den Haltebügel wieder auf und verriegeln Sie ihn.
- Führen Sie dann das Kabel zur Zugentlastung zwischen den drei Nocken vorne links an der Gehäusewand hindurch.
- Setzen Sie schließlich die Diskettenlaufwerke wieder ein und schließen Sie das Gehäuse.

#### 4.1.5 Technische Daten

**Stromversorgung:** + 5 V  $\pm$  5 %  $\leq$  50 mA

Alle Signale haben TTL-Pegel.

#### 4.1.6 Steckerbelegung

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
1	+ 5 V	Versorgungsspannung			
2	0 V	Masserückleitung, + 5 V			
3	0 V	Masseleitung			
4	$\overline{RL}$	Rückmeldung von der Tastatur	A	negativ	
5	(abgezwickelt als Kodierung)				
6	LED	Ansteuerung der Signal-LED	E	positiv	
7	$\overline{DATA}$	serielle Abtastdaten	E	negativ	
8	$\overline{CL}$	serieller Übertragungstakt	E	negativ	

-----  
**positive Logik:** 1=H, 0=L

**Treiber:** TS = tri state

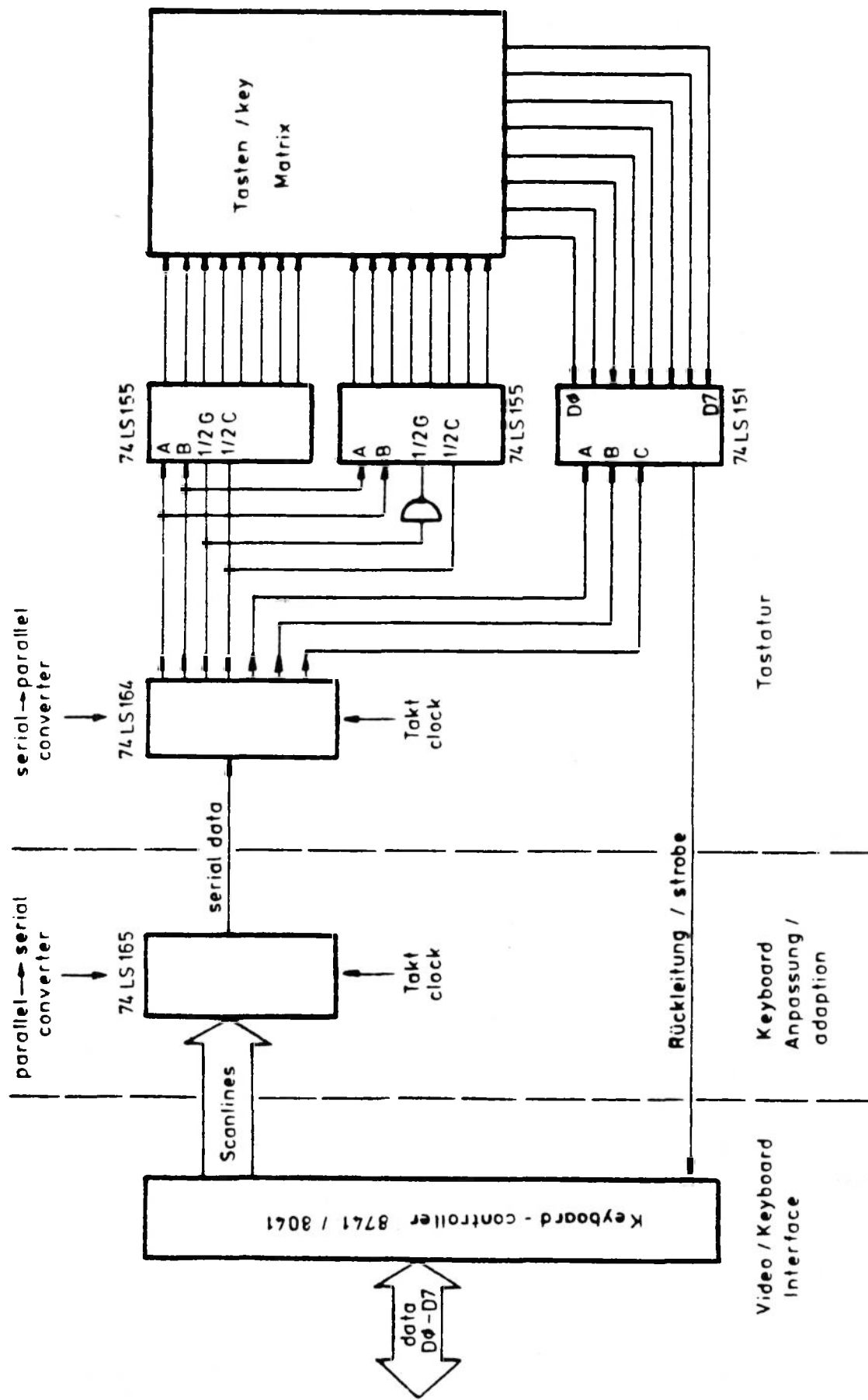
**negative Logik:** 1=L, 0=H

OC = open collector

**Richtung:** E = Eingang, A = Ausgang, B = bidirektional



4.1.7 Gesamtübersicht der Tastaturabfrage



Tastatur - Abfrage ITT 3030  
Keyboard - scanning ITT 3030

\*\*\*\*\*  
\* I T T 3 0 3 0 \*  
\* SYSTEM- UND SERVICEHANDBUCH \*  
\* ----- \*  
\* Abschnitt 6 \*  
\* Interne Schnittstellen \*  
\* \*\*\*\*\*

## 1 Verdrahtungsfeld

### 1.0 Inhalt

1.1	Konzept .....	6-1-2
1.2	Verzeichnis der Steckplätze .....	6-1-3
1.3	Verbindungsplan Stecker II (CPU) .....	6-1-4
1.4	Verbindungsplan Stecker VII (Tastatur-Adapter) .....	6-1-6
1.5	Verbindungsplan Stecker VIII (Tastatur-Stecker) .....	6-1-7
1.6	Verbindungsplan Stecker IX (Peripherie-Adapter) .....	6-1-8
1.7	Verbindungsplan Stecker X (Minifloppy Signalstecker) .....	6-1-10
1.8	Verbindungsplan Stecker XI (Video-Adapter) .....	6-1-11
1.9	Verbindungsplan Stecker XII (V24 Signalstecker) .....	6-1-12
1.10	Verbindungsplan Stecker XIII (Stromversorgung) .....	6-1-13
1.11	Lage der Steckverbinder .....	6-1-15

## 1.1 Konzept

Ein wichtiger Faktor für die Servicefreundlichkeit und die Möglichkeit, Systemerweiterungen einfach durchzuführen, ist der Aufbau des ITT 3030 als Steckkartensystem. Durch liegende Anordnung der Steckkarten im CPU-Gehäuse wird eine sehr flache Gehäuseform erreicht.

Die Verbindung der einzelnen Karten untereinander sowie der Anschluß einiger peripherer Einheiten erfolgt über das Verdrahtungsfeld, eine Leiterplatte, auf der die Steckverbinder montiert sind. Alle Verbindungen zwischen den Steckverbindern sind als Leiterbahnen ausgeführt.

Das Verdrahtungsfeld trägt keine aktiven Bauteile, da die Pufferung und Entkopplung der Systemsignale auf den jeweiligen Steckkarten erfolgt.

Die Verbindungen zwischen den Steckerleisten sind unterschiedlich, je nach Funktion der Baugruppe, die der entsprechende Steckplatz aufnimmt. Aus diesem Grund dürfen Steckkarten nur in die jeweils dafür vorgesehenen Positionen gesteckt werden.

Alle Steckverbinder werden mit römischen Zahlen gekennzeichnet. Ihre Anordnung entnehmen Sie bitte der Abbildung (siehe 1.7).

Zentraler Steckplatz ist die Position II, die die CPU-Karte aufnimmt. Von diesem Steckverbinder gehen praktisch alle Systemsignale aus und werden auf die anderen Steckplätze verteilt. Die übrigen Steckplätze nehmen Schnittstellenkarten und Steuerkarten für Ein-/Ausgabegeräte auf. Bei diesen Steckplätzen sind zusätzlich entsprechende Verbindungen zu Steckverbindern für die Peripheriegeräte vorhanden (z.B. Signale zu den Diskettenlaufwerken, zur Tastatur usw.).

## 1.2 Verzeichnis der Steckplätze

Steckplatz	Baugruppe
I	Video-Tastatur Schnittstelle
II	CPU
III	Minifloppy Schnittstelle
IV	Peripherie Adapter
V	Peripherie Adapter
VI	Peripherie Adapter
VII	Tastatur Adapter
VIII	Tastatur Anschluß
IX	Peripherie Adapter
X	Minifloppy Signalstecker
XI	Video Adapter
XII	V24 Signalstecker
XIII	Stromversorgung

1.3 Verbindungsplan Stecker II (CPU)

Reihe a

Stift	verbunden mit: Stecker	mit: Stift	Signal	
1			+ 5 V	
2			∅ V	
3			+ 12 V	
4			∅ V	
5			- 12 V	
6			∅ V	
7	I, III-VI	7	M1	Kenner f. Maschinen Zyklus 1
8	I, III-VI	8	SR/ <u>KOMBI</u>	Selektion RAM/Kombi-Baustein
9	I, III-VI	9	<u>ADR0</u>	Adreßleitung
10	I, III-VI	10	<u>ADR1</u>	Adreßleitung
11	I, III-VI	11	<u>ADR2</u>	Adreßleitung
12	I, III-VI	12	<u>ADR3</u>	Adreßleitung
13	I, III-VI	13	<u>ADR4</u>	Adreßleitung
14	I, III-VI	14	<u>ADR5</u>	Adreßleitung
15	I, III-VI	15	<u>ADR6</u>	Adreßleitung
16	I, III-VI	16	<u>ADR7</u>	Adreßleitung
17	I, III-VI	17	<u>ADR8</u>	Adreßleitung
18	I, III-VI	18	<u>ADR9</u>	Adreßleitung
19	I, III-VI	19	<u>ADR10</u>	Adreßleitung
20	I, III-VI	20	<u>ADR11</u>	Adreßleitung
21	I, III-VI	21	<u>ADR12</u>	Adreßleitung
22	I, III-VI	22	<u>ADR13</u>	Adreßleitung
23	I, III-VI	23	<u>ADR14</u>	Adreßleitung
24	I, III-VI	24	<u>ADR15</u>	Adreßleitung
25	I, III-VI	25	<u>DAT0</u>	Datenleitung
26	I, III-VI	26	<u>DAT1</u>	Datenleitung
27	I, III-VI	27	<u>DAT2</u>	Datenleitung
28	I, III-VI	28	<u>DAT3</u>	Datenleitung
29	I, III-VI	29	<u>DAT4</u>	Datenleitung
30	I, III-VI	30	<u>DAT5</u>	Datenleitung
31	I, III-VI	31	<u>DAT6</u>	Datenleitung
32	I, III-VI	32	<u>DAT7</u>	Datenleitung

Reihe b

Stift	verbunden mit: Stecker	Stift	Signal	
7	I, III-VI	7	MI	Signal: CPU Befehlsaufnahme
33			+ 5 V	
34			∅ V	
35			+ 12 V	
36			∅ V	
37			- 12 V	
38			∅ V	
39	XIII	39	UHR	Zeittakt für Uhr
40	I, III-VI	40	HALT	Meldung: CPU ist angehalten
41	I, III-VI	41	<u>MEMR</u>	Anforderung: Speicher lesen
42	I, III-VI	42	<u>MEMW</u>	Anforderung: Speicher schreiben
43	I, III-VI	43	<u>IOR</u>	Anforderung: Port lesen
44	I, III-VI	44	<u>IOW</u>	Anforderung: Port schreiben
45	I, III-VI	45	Reserve	
46	XIII	7	<u>Reset</u>	externes Rücksetzsignal
47	I, III-VI	47	<u>RST</u>	internes Rücksetzsignal
48	I, III-VI	48	<u>WAIT</u>	Warteanforderung an die CPU
49	I, III-VI	49	<u>RD</u>	Lesebefehl (Synchronisation)
50	I, III-VI	50	<u>BUSRQ</u>	Busanforderung an die CPU
51	I, III-VI	51	<u>BUSAK</u>	Busfreigabe durch die CPU
52	I, III-VI	52	<u>WR</u>	Schreibbefehl (Synchronisation)
53	I, III-VI	53	<u>Φ</u>	Systemtakt
54	I, III-VI	54	<u>IORQ</u>	Anforderung Ein-/Ausgabeport
55	I, III-VI	55	<u>MREQ</u>	Anforderung Speicher
56	I, III-VI	56	<u>INT</u>	Maskierbare Unterbrechung
57			IEI	Eingang Unterbrechungskette
58	IV	57	IEO	Ausgang Unterbrechungskette
59	I, III-VI	59	Reserve	
60	I, III-VI	60	Reserve	
61	XIII	38	<u>NMI</u>	nicht maskierbare Unterbrechung
62	I, III-VI	62	<u>INT0</u>	Unterbrechungsvektor 0
63	I, III-VI	63	<u>INT1</u>	Unterbrechungsvektor 1
64	I, III-VI	64	<u>INT2</u>	Unterbrechungsvektor 2

1.4 Verbindungsplan Stecker VII (Tastatur-Adapter)

Stift	verbunden mit: Stecker	Stift	Signal	
1			+ 5 V	
2			0 V	
3	I	65	MSA0	Matrix-Abtastung
4	I	66	MSA1	Matrix-Abtastung
5	I	67	MSA2	Matrix-Abtastung
6	I	68	MSA3	Matrix-Abtastung
7	I	69	MSA4	Matrix-Abtastung
8	I	70	MSA5	Matrix-Abtastung
9	I	71	MSA6	Matrix-Abtastung
10				Steckercodierung
11	I	95	KCL	Abfragetakt Tastatur
12	VIII	7	<u>DATA</u>	serielle Abtastdaten
13	VIII	8	<u>CL</u>	serieller Übertr.Takt
14	I	47	<u>RST</u>	internes Rücksetzsignal
15	I	96	Hupe	Ansteuerung Signalgeber

1.5 Verbindungsplan Stecker VIII (Tastatur-Stecker)

Stift	verbunden mit: Stecker	Stift	Signal	
1			+ 5 V	
2			∅ V	
3			∅ V	
4	I	72	$\overline{RL}$	Rückmeldung v. Tastatur
5				Steckercodierung
6	I	94	LED	Ansteuerung SHIFT-LOCK LED
7	VII	12	$\overline{DATA}$	serielle Abtastdaten
8	VII	13	$\overline{CL}$	serieller Übertr.Takt
9	I	77	STAT3	Statussignal
10	I	81	STAT4	Statussignal
11	I	82	STAT5	Statussignal
12			∅ V	



1.6 Verbindungsplan Stecker IX (Peripherie-Adapter)

Reihe a

Stift	verbunden mit: Stecker	Stift	Signal	
1	VI	25	<u>DAT0</u>	Datenleitung
2	VI	26	<u>DAT1</u>	Datenleitung
3	VI	27	<u>DAT2</u>	Datenleitung
4	VI	28	<u>DAT3</u>	Datenleitung
5	VI	29	<u>DAT4</u>	Datenleitung
6	VI	30	<u>DAT5</u>	Datenleitung
7	VI	31	<u>DAT6</u>	Datenleitung
8	VI	32	<u>DAT7</u>	Datenleitung
9	VI	56	Reserve	
10	VI	61	<u>NM 1</u>	nicht maskierbare Unterbr.
11	VI	64	<u>INT 2</u>	Unterbrechungsvektor 2
12			- 12 V	
13			+ 5 V	
14			∅ V	
15			∅ V	
16			+ 12 V	

Reihe b

17	VI	8	<u>SR/KOMBI</u>	Umschaltg. Speicher/Kombibaust.
18	VI	10	<u>ADR 1</u>	Adreßleitung
19	VI	12	<u>ADR 3</u>	Adreßleitung
20	VI	14	<u>ADR 5</u>	Adreßleitung
21	VI	16	<u>ADR 7</u>	Adreßleitung
22	VI	42	<u>MEMW</u>	Anforderung: Speicher schreiben
23	VI	44	<u>IOW</u>	Anforderung: Port schreiben
24	VI	48	<u>WAIT</u>	Warteanforderung an CPU
25	VI	51	<u>BUSAK</u>	Busfreigabe durch die CPU
26	VI	53	<u>∅</u>	Systemtakt
27	VI	63	<u>INT 1</u>	Unterbrechungsvektor 1
28			- 12 V	
29			+ 5 V	
30			∅ V	
31			∅ V	
32			+ 12 V	

Reihe c

Stift	verbunden mit: Stecker	Stift	Signal	
33	VI	7	M1	Kenner Masch. Zyklus 1
34	VI	9	$\overline{\text{ADR } 0}$	Adreßleitung
35	VI	11	$\overline{\text{ADR } 2}$	Adreßleitung
36	VI	13	$\overline{\text{ADR } 4}$	Adreßleitung
37	VI	15	$\overline{\text{ADR } 6}$	Adreßleitung
38	VI	41	$\overline{\text{MEMR}}$	Anforderung: Speicher lesen
39	VI	43	$\overline{\text{IOR}}$	Anforderung: Port lesen
40	VI	47	$\overline{\text{RST}}$	internes Rücksetzsignal
41	VI	50	$\overline{\text{BUSRQ}}$	Busanforderung an CPU
42	VI	58	$\overline{\text{IEI}}$	Eingang Unterbrechungskette
43	VI	62	$\overline{\text{INT } 0}$	Unterbrechungsvektor 0
44			- 12 V	
45			+ 5 V	
46			0 V	
47			0 V	
48			+ 12 V	

1.7 Verbindungsplan Stecker X (Minifloppy Signalstecker)

Stift	verbunden mit: Stecker	Stift	Signal	
1	III	66	<u>Side Sel</u>	Anwahl Diskettenseite
2	III	65	<u>MOT ON</u>	Einschaltung Motor
3	III	68	<u>Write Data</u>	Schreibdaten
4	III	67	<u>Write Gate</u>	Schreibstromfreigabe
5	III	70	<u>Write Protect</u>	Diskette schreibgeschützt
6	III	69	<u>DIRIN</u>	Positionierrichtung
7	III	72	<u>Read Data</u>	Lesedaten
8	III	71	<u>RDY 1</u>	Laufwerk 1 bereit
9	III	75	<u>RDY 3</u>	Laufwerk 2 bereit
10	III	74	<u>RDY 2</u>	Laufwerk 3 bereit
11	III	81	<u>Sel 1</u>	Anwahl Laufwerk 1
12	III	76	<u>TRACK 00</u>	Meldung: Kopf auf Spur 0
13	III	83	<u>Step</u>	Schritimpuls
14	III	82	<u>Sel 2</u>	Anwahl Laufwerk 2
15	III	85	<u>Sel 3</u>	Anwahl Laufwerk 3
16	III	84	<u>Head Load</u>	Kopf aufsetzen
17	III	87	<u>DOOR 2</u>	Klappenverriegelung Lw. 2
18	III	86	<u>DOOR 1</u>	Klappenverriegelung Lw. 1
19	III	92	<u>IND</u>	Index-Impuls
20			+ 5 V	
21			+ 5 V	
22			0 V	
23			0 V	
24			0 V	
25			+ 12 V M	Motor-Spannungsversorgung
26			+ 12 V M	Motor-Spannungsversorgung

1.8 Verbindungsplan Stecker XI (Video-Adapter)

Stift	verbunden mit: Stecker	Stift	Signal	
1			+ 5 V	
2			∅ V	
3	I	78	BVID	Video-Signal, blau
4	I	79	GVID	Video-Signal, grün
5	I	80	RVID	Video-Signal, rot
6	I	83	<u>HSYN</u>	Horizontal-Synchronsignal
7	I	84	<u>VSYN</u>	Vertikal-Synchronsignal
8	I	85	<u>CSYN</u>	kombiniertes Synchronsignal
9	I	86	Video	Video-Signal
10	I	87	b	
11	I	88	g	
12	I	89	r	
13	I	92	QT	Quarztakt für Videosignal
14	I	76	STAT ∅	Statussignal ∅
15	I	75	STAT 1	Statussignal 1
16	I	74	STAT 2	Statussignal 2
17			BAS	BAS-Signal

1.9 Verbindungsplan Stecker XII (V24 Signalstecker)

Stift	verbunden mit: Stecker	mit: Stift	Signal	
1			PGND	Gerätemasse
2	II	79	$\overline{\text{TXD}} \emptyset$	Sendedaten
3	II	78	$\overline{\text{RXD}} \emptyset$	Empfangsdaten
4	II	74	RTS $\emptyset$	Sendeanforderung
5	II	75	CTS $\emptyset$	Sendebereitschaft
6	II	77	DSR $\emptyset$	Bereitschaftsbestätigung
7	II		SGND	Signalmasse
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20	II	76	DTR $\emptyset$	Bereitschaftsmeldung
21				
22				
23				
24				
25				

1.6 Verbindungsplan Stecker XIII (Stromversorgung)

Reihe a

Stift	verbunden mit: Stecker	mit: Stift	Signal	
1				
2				
3				
4				
5				
6				
7	II	46	<u>RESET</u>	externes Rücksetzsignal
8			+ 12 V	
9			- 12 V	
10			+ 12 V M	Motorspannung für Minifloppy
11			+ 12 V M	Motorspannung für Minifloppy
12			+ 5 V	
13			+ 5 V	
14			∅ V	
15			∅ V	
16			∅ V	

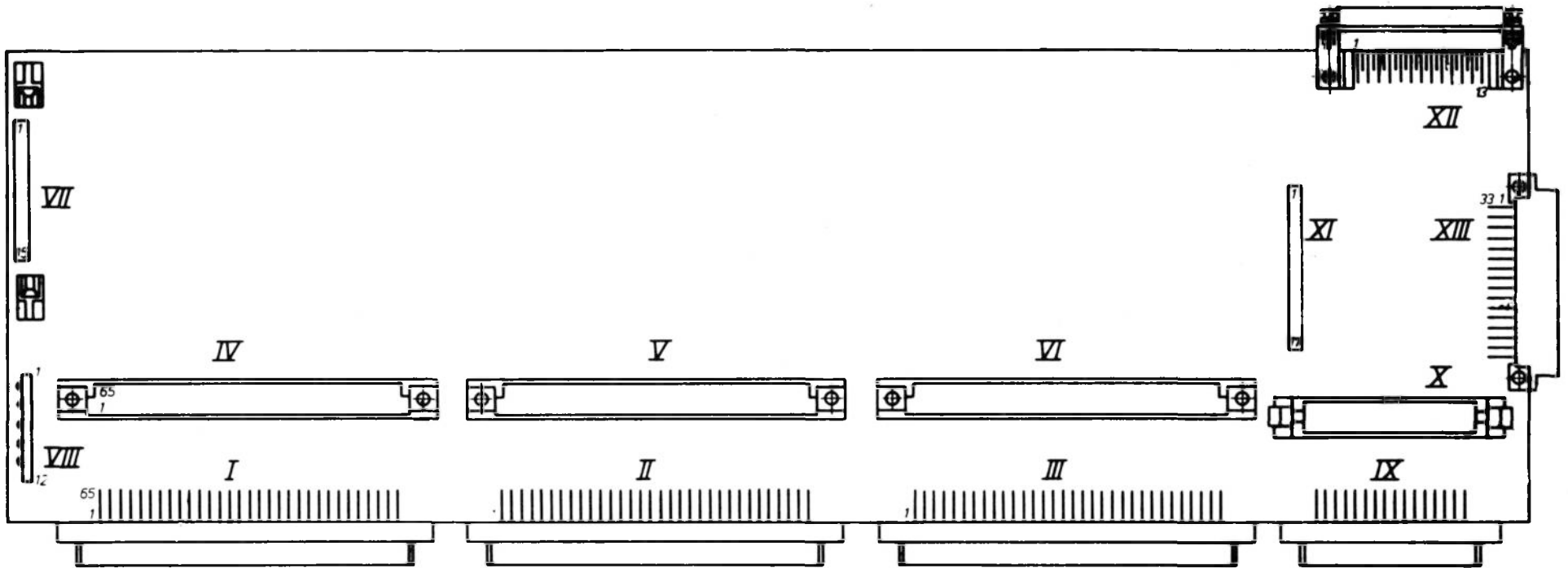
Reihe b

17				
18				
19				
20				
21				
22				
23				
24			+ 12 V	
25			- 12 V	
26			+ 12 V M	Motorspannung für Minifloppy
27			+ 12 V M	Motorspannung für Minifloppy
28			+ 5 V	
29			+ 5 V	
30			∅ V	
31			∅ V	
32			∅	

Reihe c

Stift	verbunden mit:		Signal	
	Stecker	Stift		
33				
34				
35				
36				
37				
38	II	61	NMI	nicht maskierbare Unterbrechung
39	II	39	UHR	Zeittakt für Uhr
40			+ 12 V	
41			- 12 V	
42			+ 12 V M	Motorspannung für Minifloppy
43			+ 12 V M	Motorspannung für Minifloppy
44			+ 5 V	
45			+ 5 V	
46			Ø V	
47			Ø V	
48			Ø	

1.7 Lage der Steckverbinder





```
*****
*
*           I T T   3 0 3 0
*       SYSTEM- UND SERVICEHANDBUCH
*       -----
*           Abschnitt 6
*       Interne Schnittstellen
*
*****
```

## 2 Disketten-Schnittstellen

### 2.1 Minifloppy-Controllerkarte

#### 2.1.0 Inhalt

2.1.1 Konzept .....	6-2-2
2.1.2 Schnittstellen .....	6-2-3
2.1.2.1 Systembusschnittstelle .....	6-2-3
2.1.2.2 Floppybusschnittstelle .....	6-2-4
2.1.3 Funktionsgruppen .....	6-2-4
2.1.4 Adreßbelegung .....	6-2-6
2.1.5 Register der Minifloppy-Controllerkarte .....	6-2-6
2.1.5.1 Karten-Status .....	6-2-6
2.1.5.2 Karten-Befehlsregister .....	6-2-7
2.1.6 Register des Formatter/Controller-Bausteins 1791 ...	6-2-7
2.1.7 Abgleich der Datentrennstufe (PLL) .....	6-2-8
2.1.8 Steckerbelegung .....	6-2-10
2.1.9 Technische Daten .....	6-2-13
2.1.10 Abbildungen .....	6-2-14
2.1.10.1 Bestückung .....	6-2-14
2.1.10.2 Blockschalbild .....	6-2-15

### 2.1.1 Konzept

Die Minifloppy-Controllerkarte stellt die Verbindung zwischen der CPU und den als Massenspeicher verwendeten Minidiskettenlaufwerken her. Die Hauptaufgaben dieser Baugruppe sind:

- Selektion der angeschlossenen Minidiskettenlaufwerke (maximal drei).
- BUS-Entkopplung
- Zwischenspeicherung der Ein-/Ausgabe-Adressen und Steuersignale.
- Positionieren der Schreib-/Leseköpfe auf die gewünschte Spur.
- Parallel-Serienwandlung der Schreibdaten und Serien-Parallelwandlung der Lesedaten.
- Schreiben bzw. Lesen eines Diskettensektors.
- Schreiben bzw. Lesen einer Diskettenspur (diese Funktion wird normalerweise zum Formatieren der Disketten verwendet).
- Zeitsteuerung aller Funktionen und Signalaufbereitung mit Schreibtakterzeugung und Trennung von Takt- und Datensignalen beim Lesen (Datenseparation).
- Prüfung der Lesedaten auf Fehler und Erzeugung von Fehlerprüfcodes beim Schreiben.

Der zentrale Baustein der Minifloppy-Controllerkarte ist eine integrierte Schaltung (FD1791 A, Formatter/Controller) die zusammen mit den übrigen Bausteinen der Karte die oben angegebenen Funktionen ausführt.

Die zwischen CPU und Controller ausgetauschte Information kann wie folgt gegliedert werden:

- Befehle an die Laufwerke (Selektion, Motor ein/aus usw.).
- Befehle an den Formatter/Controller-Baustein (Spur suchen, Sektor lesen, Sektor Schreiben usw.).
- Statusinformationen der Controllerkarte und der Laufwerke (Bereitmeldung der Laufwerke, Schreibschutz usw.).
- Statusinformationen des Formatter/Controller-Bausteins (Fehlermeldungen, Datenanforderung usw.).
- Schreib- und Lesedaten (8 Bit parallel).

## 2.1.2 Schnittstellen

Über den Systemstecker werden zwei Schnittstellen herausgeführt:

- der Systembus (Reihe a und b) und
- der Floppybus (Reihe c).

Die Minifloppy-Controllerkarte wird im CPU-Gehäuse in die Steckposition III (erste 96-polige VG-Leiste von rechts) des Verdrahtungsfeldes gesteckt. Die Signalleitungen der Reihe c sind mit dem Stecker X auf dem Verdrahtungsfeld verbunden. Von dort wird über ein Flachkabel die Verbindung zum Floppy-Verdrahtungsfeld hergestellt.

### 2.1.2.1 Systembusschnittstelle

Die Schnittstelle zum Systembus trägt die Versorgungsspannungen, Adreß- und Datensignale sowie einige Steuersignale des Systems. Im Einzelnen sind das auf der

**- a-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen  
 Adreßbus 8 Bit (invertiert):  $\overline{ADR0} \dots \overline{ADR7}$   
 bidirektionaler Datenbus 8 Bit (invertiert):  $\overline{DAT0} \dots \overline{DAT7}$

**- c-Reihe:**

Versorgungsspannungen + 5 V, + 12 V, - 12 V und Masseleitungen  
 Steuersignale:  $\overline{IOR}$ ,  $\overline{IOW}$ ,  $\overline{IORQ}$ ,  $\overline{BUSAK}$ ,  $\overline{INTV}$   
 systeminternes Rücksetzsignal:  $\overline{RST}$

Die Minifloppy-Controllerkarte wird als Ein-/Ausgabekanal angesprochen. Sie belegt im Normalfall die Adressen 50H bis 55H (eine Änderung der Kanaladresse 5xH ist durch Umlöten von Brücken auf der Karte möglich).

- Head-Load (HLD)/Step Logik: Wenn durch den Formatter/Controller Baustein ein Befehl zum Positionieren oder Aufsetzen des Schreib-/Lesekopfes gegeben wird, müssen mechanische Reaktionszeiten beachtet werden. Durch monostabile Kippstufen werden die erforderlichen Verzögerungen (40 ms für Spurwechsel 60 ms für Kopfaufsetzen) erzeugt. Der Ablauf dieser Zeiten kann sowohl durch die CPU als auch durch den Formatter/Controllerbaustein (über den HLT-Eingang) abgefragt werden.
- Formatter/Controller Baustein (1791): dieser Baustein übernimmt u.a. die komplexen Steuerfunktionen für die Laufwerke, die Parallel/Serienwandlung der Schreib- und die Serien/Parallelwandlung der Lesesignale, Erzeugung von Fehlerprüf-codes beim Schreiben und Fehlerprüfung der Lesedaten sowie das blockweise Schreiben und Lesen der Daten. Die Ausgangssignale dieses Bausteins werden über einen
  - Treiber auf die Diskettenlaufwerke übertragen.
  - Die Datentrennstufe führt die Zerlegung der Lesesignale in Daten und Taktinformation durch. Sie besteht aus zwei Funktionsgruppen:
    - dem Phasenregelkreis (PLL), mit dem die Referenzfrequenz für Öffnungszeitpunkt und -dauer der Daten- und Taktfenster der Bitfolgefrequenz der Lesedaten nachgeführt wird.
    - einer Logik zum Erkennen von Fehltakten. Werden vier aufeinanderfolgende Fehltakte erkannt, dann wird die Zuordnung zwischen Takt- und Datenfenster umgekehrt. Wenn keine Daten gelesen werden (Steuersignal Read Gate - RG nicht aktiv), dann wird die PLL durch ein Taktsignal (8 MHz) auf ihre Sollfrequenz gezwungen. Außer durch das Signal RG kann diese Umschaltung auch über die Steuerleitung RG J durch die CPU erfolgen. Das ist erforderlich, da der Formatter/Controller-Baustein die Leitung RG nicht aktiviert, wenn er die Funktion "Spur lesen" ausführt (siehe 2.1.5.2 Karten-Befehlsregister).
- Kompensations-Logik: abhängig von der zu schreibenden Spur und der sich damit ändernden Datendichte wird die Schreibkompensation aktiviert. Bei 48 tpi Laufwerken erfolgt das ab Spur 16 und bei 96 tpi Laufwerken ab Spur 40. Je nach zu schreibendem Bitmuster setzt der Controller-Baustein die Signale EARLY und LATE, mit denen ein Multiplexer geschaltet wird, der den Schreibimpuls 125 ns früher oder später als normal zum Laufwerk schaltet. Durch einen entsprechenden Befehl kann durch die CPU die Kompensation abgeschaltet werden (siehe 2.1.5.2 Karten-Befehlsregister).

### 2.1.4 Adreßbelegung

Die Basis- oder Kanaladresse der Minifloppy-Controllerkarte ist normalerweise 5xH. Eine Änderung dieser Adresse kann durch Auftrennen und Umlöten der Brücken S1 ... S4 erfolgen. Die Brücken befinden sich auf der Bauteilseite der Karte in der Reihe 3 unten (Karte mit VG-Stecker nach rechts orientiert). Bei Bedarf kann hier ein Vierfach-DIP Schalter eingesetzt werden. (Vorher kaschierte Verbindungen trennen!)

Im Einzelnen gilt folgende Adreßbelegung (bei Basisadresse 5x):

- 50H - 1791 Status/Befehls-Register
- 51H - 1791 Spur-Register
- 52H - 1791 Sektor-Register
- 53H - 1791 Daten-Register
- 54H - Karten-Status/Karten-Befehlsregister

### 2.1.5 Register der Minifloppy-Controllerkarte

#### 2.1.5.1 Karten-Status

Durch den Befehl IN 54H kann der Status der Minifloppy-Controllerkarte durch die CPU gelesen werden. Dabei haben die einzelnen Bits folgende Bedeutung:

<u>Bit</u>	<u>Bedeutung</u>
7	Data Request ( $\overline{DRQ}$ - invertiertes 1791-Signal)
6	Interrupt Request (INTRQ - 1791-Signal)
5	Head Load ( $\overline{HLD}$ - invertiertes 1791-Signal)
4	Ready 3 (Laufwerk 3 bereit)
3	Ready 2 (Laufwerk 2 bereit)
2	Ready 1 (Laufwerk 1 bereit)
1	Schreibschutz (die Diskette im selektierten Laufwerk ist schreibgeschützt)
0	HLT (Haltsignal bei Kopfaufsetzen und Spurwechsel)

### 2.1.5.2 Karten-Befehlsregister

Durch den Befehl OUT 54H kann durch die CPU ein Befehl an die Minifloppy-Controllerkarte übertragen werden. Dabei haben die einzelnen Bits folgende Bedeutung:

<u>Bit</u>	<u>Bedeutung</u>
7	$\overline{\text{SEL1}}$ - Laufwerk 1 selektieren
6	$\overline{\text{SEL2}}$ - Laufwerk 2 selektieren
5	$\overline{\text{SEL3}}$ - Laufwerk 3 selektieren
4	$\overline{\text{MOTOR}}$ - Motor ein
3	$\overline{\text{DOOR}}$ - Klappenverriegelung Laufwerk 1 und 2 (nicht bei allen Laufwerken möglich)
2	$\overline{\text{SIDESEL}}$ - Auswahl der Diskettenseite
1	$\overline{\text{KOMP}}$ - Schreibkompensation ein/aus
0	RG J - Umschalten der Daten-Trennstufe auf Lesen

### 2.1.6 Register des Formatter/Controller-Bausteins 1791

Die in diesem Abschnitt enthaltenen Adreßangaben basieren auf einer Basisadresse der Minifloppy-Controllerkarte von 5xH.

Der Baustein 1791 enthält vier Register, die durch die CPU gelesen oder beschrieben werden können (weitere Register dienen internen Zwecken und sind für die Programmierung ohne Bedeutung):

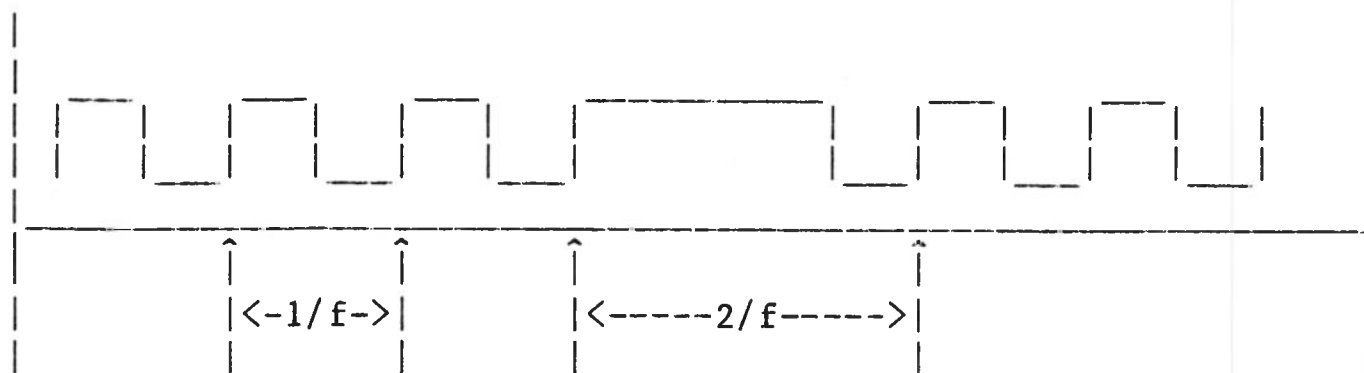
- Status/Befehls-Register: durch den Befehl IN 50H kann durch die CPU der Status des Bausteins gelesen werden. Die Bedeutung der einzelnen Bits hängt von dem jeweiligen Befehl ab, den der Baustein ausführt.  
Durch den Befehl OUT 50H wird ein Befehl an den Baustein übertragen. Je nach auszuführender Funktion müssen andere Register des Bausteins vorher mit entsprechenden Daten geladen werden. Mögliche Befehle sind: Kopf auf Spur 0, Spur suchen, eine Spur fortschalten, eine Spur nach innen, eine Spur nach außen, Sektor lesen, Sektor schreiben, Lesen der Adreßinformation eines Sektors, Spur schreiben, Spur lesen und laufenden Befehl abbrechen.
- Spur-Register (Adresse 51H): dieses Register enthält die aktuelle Spur. Bei jeder Kopfbewegung wird der Inhalt des Registers entsprechend geändert. Vor dem Schreiben oder Lesen wird der Inhalt dieses Registers mit dem Spurkenner auf der

- Diskette verglichen und bei Abweichung ein Fehler signalisiert.
- Sektor-Register (Adresse 52H): dieses Register enthält die Adresse des nächsten zu schreibenden oder zu lesenden Sektors. Vor dem Schreiben oder Lesen wird der Inhalt dieses Registers mit dem Sektorkenner auf der Diskette verglichen und bei Abweichung ein Fehler signalisiert.
  - Daten-Register (Adresse 53H): über dieses Register erfolgt beim Schreiben und Lesen von Diskettendaten die Übertragung zwischen CPU und Controller. Durch entsprechende Programmierung muß dafür gesorgt werden, daß die Datenbytes rechtzeitig in das Register geladen bzw. aus dem Register gelesen werden. Vor einem Befehl "Spur suchen" muß in das Datenregister die gewünschte Spur geschrieben werden.

Genauere Informationen über die verschiedenen Befehle und die Bedeutung der einzelnen Register finden Sie im Datenblatt des 1791 Bausteins.

### 2.1.7 Abgleich der Datentrennstufe (PLL)

Für den Abgleich der PLL-Stufe ist ein Rechtecksignal erforderlich, das nachfolgendem Diagramm entspricht (Impulsfolge: 2 Perioden mit je  $1/f$ , 1 Periode mit  $2/f$ ):



Ein derartiges Signal kann aus einer Grundfrequenz  $f$  mittels Zählern gewonnen werden.

Bei einer Grundfrequenz von 333 kHz an Stift 72 des VG-Steckers (Read Data) muß der Gleichspannungspegel am Anschluß 94 (PLL-Pegel) 3,95 V ( $\pm 0,1$  V) betragen.

Ein Abgleich kann durch Öffnen und Schließen der Brücken J5 ... J7 erfolgen. Diese Brücken befinden sich auf der Bauteileseite der Karte oben zwischen den Bauteilereihen 1 und 2 (Karte mit VG-Stecker nach rechts orientiert). J5 schaltet dem frequenzbestimmenden Kondensator der PLL eine Kapazität von 5,6 pF parallel, die Brücken J6 und J7 jeweils 10 pF.

Nach erfolgtem Abgleich des Pegels muß der Regelbereich mindestens von 208,5 kHz (- 1%) bis 312,7 kHz (+ 1%) betragen. Die Regelbereichsgrenzen werden wie folgt geprüft:

Grundfrequenz  $f$  in Stufen (278, 333, 417 kHz) einstellen. Die Frequenz an Pin 15 der IS 74LS393, die sich in der Mitte der Baugruppe befindet, muß jeweils  $0,75 \times f \pm 1\%$  (entspr. 208,5, 250, 312,7 kHz) betragen.



Reihe b

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
33	+ 5 V	Versorgungsspannung			
34	∅ V	Masserückleitung + 5 V			
35	+ 12 V	Versorgungsspannung			
36	∅ V	Masserückleitung + 12 V			
37	- 12 V	Versorgungsspannung			
38	∅ V	Masserückleitung - 12 V			
39					
40					
41					
42					
43	$\overline{\text{IOR}}$	Eingabekanal lesen	E	negativ	
44	$\overline{\text{IOW}}$	Ausgabekanal schreiben	E	negativ	
45	$\overline{\text{INT}} \text{ V}$	Unterbrechungs-Vektor	E	negativ	
46					
47	$\overline{\text{RST}}$	internes Rücksetzsignal	E	negativ	
48					
49					
50					
51	$\overline{\text{BUSAK}}$	Busfreigabe durch die CPU	E	negativ	
52					
53					
54	IORQ	Anforderung Ein-/Ausgabe	E	positiv	
55					
56					
57					
58					
59					
60					
61					
62					
63					
64					

positive Logik: 1=H, ∅=L

negative Logik: 1=L, ∅=H

Richtung: E = Eingang, A = Ausgang, B = bidirektional

Treiber: TS = tri state

OC = open collector

Reihe c

An- schluß	Bezeich- nung	Bedeutung	Rich- tung	Logik	Treiber
65	<u>MOTON</u>	Motor-Start-Signal	A	negativ	
66	<u>SIDSEL</u>	Diskettenseiten-Anwahl	A	negativ	
67	<u>WrtGate</u>	Schreibstrom-Freigabe	A	negativ	
68	<u>WrtData</u>	Schreibdaten	A	negativ	
69	<u>DIRIN</u>	Schrittmot.-Richtg. z. Mitte	A	negativ	
70	<u>WrtProt</u>	Schreibschutz	E	negativ	
71	<u>RDY 1</u>	Laufwerk 1 bereit	E	negativ	
72	<u>RdData</u>	Lesedaten	E	negativ	
73					
74	<u>RDY 2</u>	Laufwerk 2 bereit	E	negativ	
75	<u>RDY 3</u>	Laufwerk 3 bereit	E	negativ	
76	<u>TRK 00</u>	Spur Null Position	E	negativ	
77					
78					
79					
80					
81	<u>SEL 1</u>	Anwahl Laufwerk 1	A	negativ	
82	<u>SEL 2</u>	Anwahl Laufwerk 2	A	negativ	
83	<u>Step</u>	Schrittmotorimpuls	A	negativ	
84	<u>HeadLd</u>	Kopf aufsetzen	A	negativ	
85	<u>SEL 3</u>	Anwahl Laufwerk 3	A	negativ	
86	<u>DOOR 1</u>	Klappenverriegelung Lw. 1	A	negativ	
87	<u>DOOR 2</u>	Klappenverriegelung Lw. 2	A	negativ	
88					
89					
90	PLL-B	Prüfanschluß für PLL	A		
91					
92	<u>Index</u>	Index-Puls	E	negativ	
93	PLL-A	Prüfanschluß für PLL	A		
94	PLL-Peg.	Prüfanschluß für PLL	A	analog	
95	0 V	Massepotential			
96					

positive Logik: 1=H, 0=L

Treiber: TS = tri state

negative Logik: 1=L, 0=H

OC = open collector

Richtung: E = Eingang, A = Ausgang, B = bidirektional

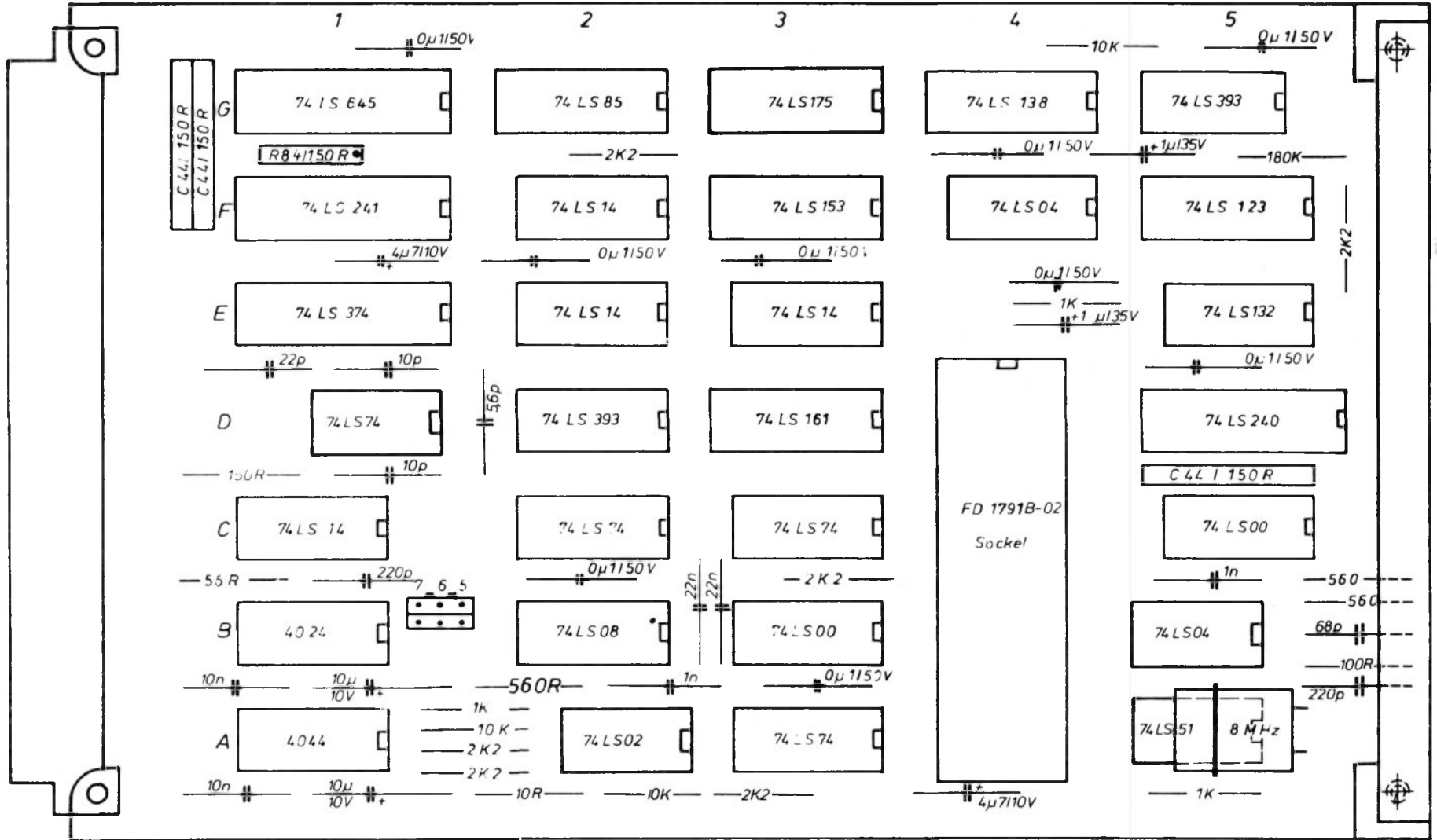
### 2.1.9 Technische Daten

**Stromaufnahme:**        + 5 V  $\pm$  5% :  $\leq$  350 mA  
                             + 12 V  $\pm$  5% :  $\leq$  12 mA

**Oszillator:**            4 MHz  $\pm$   $10^{-4}$

2.1.10 Abbildungen

2.1.10.1 Bestückung



2.1.10.2 Blockschaltbild

