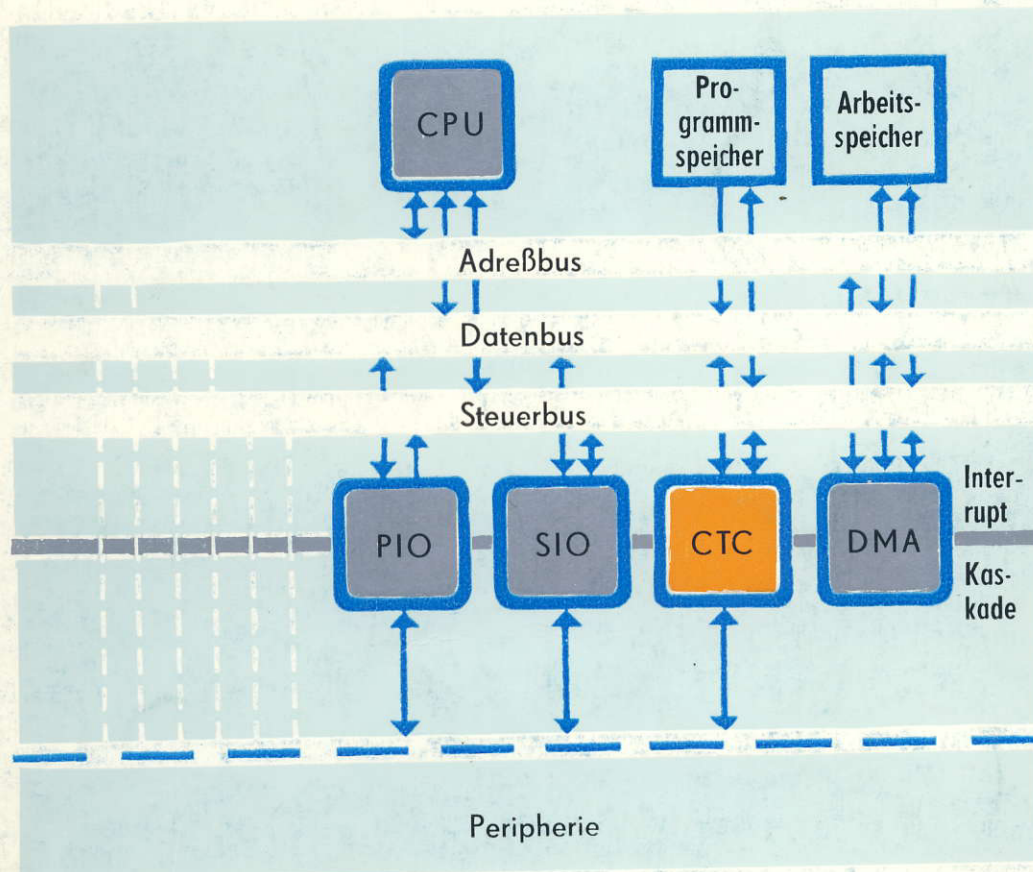




Mikroprozessorsystem

der II. Leistungsklasse

– Technische Beschreibung –



Schaltkreis für Zähler- und Zeitgeberfunktion
CTC U 857 D

**Schaltkreis für Zähler - und Zeitgeberfunktion
CTC U 857 D**



veb funkwerk erfurt

im veb kombinat mikroelektronik

Die technischen Angaben dieses Handbuches tragen reinen Informationscharakter. Verbindliche technische Liefer- und Reklamationsgrundlage sind ausschließlich die Typstandards. Die vorliegende Dokumentation gibt keine Auskunft über Liefermöglichkeiten und beinhaltet keine Verbindlichkeiten zur Produktion.

Inhaltsverzeichnis

	Seite
1. Einführung	4
2. Aufbau der U 857 D	5
2.1. Aufbau eines CTC-Kanals	6
2.1.1. Format des Kanalsteuer-Registers	7
2.2. Die Interrupt-Steuerlogik	8
3. U 857 D-Anschlußbeschreibung	10
4. Betriebs- und Operationsarten der U 857 D	13
4.1. Betriebsart "Zähler" der U 857 D	14
4.2. Betriebsart "Zeitgeber" der U 857 D	15
5. Programmierung der U 857 D	16
5.1. Programmierung der Kanalsteuer-Register	17
5.2. Programmierung der Zeitkonstanten-Register	19
5.3. Programmierung des Interruptvektor-Registers	20
5.4. Ablauf der Programmierung der U 857 D	21
6. Interruptbehandlung in der U 857 D	22
6.1. Der Interrupt-Aannahme-Zyklus	23
6.2. Start der Interrupt-Service-Routine durch den U 880 D	24
6.3. Rückkehr vom Interrupt	25
6.4. Die Interrupt-Prioritätskette	26
7. Zeitprobleme in der Interrupt-Behandlung	28
7.1. Maßnahmen zur Entschärfung der Zeitprobleme	29
8. Zeitverhalten der U 857 D	31
8.1. CTC-Schreib-Zyklus	31
8.2. CTC-Lese-Zyklus	32
8.3. Zeitverhalten eines Zähler-Kanals	32
8.4. Zeitverhalten eines Zeitgeber-Kanals	33
8.5. Zeitverläufe im Interrupt-Aannahme-Zyklus	34
8.6. Zeitverläufe bei der Rückkehr vom Interrupt	36
9. Applikationsbeispiel	36
10. Technische Daten der U 857	43
10.1. Zuverlässigkeitswerte	43
10.2. Ein- und Ausgänge	44
10.3. Elektrische Kennwerte	47
10.4. Gehäuse	51

1. Einführung

Die U 857 D ist der Zähler/Zeitgeber-Schaltkreis (CTC) im U 880-Mikroprozessorsystem (2. Leistungsklasse).

Zum System gehören folgende LSI-Schaltkreise:

- U 880 D - CPU (Zentrale Verarbeitungseinheit)
- U 855 D - PIO (Parallele Ein/Ausgabe)
- U 856 D - SIO (Serielle Ein/Ausgabe)
- U 857 D - CTC (Zähler/Zeitgeber)

Alle Schaltkreise werden in n-Kanal-Silicon-Gate-Technologie hergestellt und haben mit Ausnahme der U 857 D (28polig) ein 40poliges DIL-Plastgehäuse. Der Aufbau der Schaltkreise realisiert ein einheitliches Systemkonzept, das folgende wesentliche Merkmale aufweist:

- Signalzuführung über drei Systembusse (Adressbus, Datenbus und Steuerbus)
- einheitliches Interruptregime mit der Möglichkeit der Prioritätsverkettung der Peripherieschaltkreise
- hochgradige Softwareprogrammierbarkeit der schaltkreisspezifischen Eigenschaften
- gemeinsamer Systemtakt als Einphasentakt mit einer maximalen Frequenz von 2,5 MHz
- einheitliche Spannungsversorgung von +5 V

Beim Aufbau von Mikrorechnern wird das System durch Standard-Festwertspeicher (ROM, PROM), statische oder dynamische Standard-Schreib-Lesespeicher (RAM) und ggf. durch Bustreiber und Adreßdekoder arbeitsfähig.

Die CTC U 857 D weist die folgenden charakteristischen Merkmale auf:

- Sie besitzt vier voneinander unabhängige software-programmierbare 8-Bit-Zähler/16-Bit-Zeitgeber Kanäle.
- Jeder Kanal ist wahlweise als Zähler oder Zeitgeber programmierbar.
- Aktuelle Werte (Zähler- bzw. Zeitgeberzustände) sind über den Datenbus abrufbar.
- In der Betriebsart Zeitgeber verwendet jeder Kanal einen Vorteiler, mit dem er die durch 16 oder 256 geteilten Systemtakte als Zeitbasis verwendet.
- Ein als Zeitgeber programmierter Kanal kann wahlweise durch positiven oder negativen Triggerimpuls gestartet werden.
- In der Betriebsart Zähler ist die aktive Zählflanke wählbar.
- Die 4 Kanäle bilden 4 unabhängige interruptfähige Einheiten.
- Aus dem Inhalt des Interrupt-Vektorregisters werden bei Interrupts durch Variation von 2 Bits vier Kanal-Interruptvektoren erzeugt.
- Nulldurchgänge der Kanäle können Interrupts anfordern, d. h. periphere Zustände können Interrupts auslösen.
- Die 4 Kanäle sind intern zu einer Interruptprioritätskette verschaltet, wobei die Prioritätsreihenfolge festliegt.
- In der Betriebsart Zeitgeber beträgt die Zeitraster-Intervall-Auflösung entweder 16 Systemtaktperioden ($6,4 \mu\text{s}$ bei 2,5 MHz) oder 256 Systemtaktperioden ($102,4 \mu\text{s}$ bei 2,5 MHz) entsprechend dem Vorteilerfaktor.
- Die maximale Zählfrequenz in der Betriebsart Zähler beträgt $f_c/2$, d. h. 1,25 MHz bei einer Systemtaktfrequenz $f_c = 2,5$ MHz.
- Die Kanäle sind zur Erweiterung des Zählumfanges und gegenseitigen Triggerung kaskadierbar.
- Alle Ein- und Ausgänge des CTC sind TTL-kompatibel.
- Die 3 Ausgänge der Kanäle 0 - 2 (ZC/TO_n) sind für den Anschluß von Darlington-Transistoren ausgelegt.
- Der Ausgang von Kanal 3 (ZC/TO₃) ist wegen Beschränkung auf ein 28-PIN Gehäuse nicht herausgeführt.

Der integrierte Schaltkreis U 857 D ist ein Vier-Kanal-Zähler/Zeitgeberbaustein, dessen Betriebsart und Operationsparameter durch Laden von Registern programmierbar, d. h. in bestimmten Grenzen variabel sind. Neun 8 Bit-Register (4 Kanalsteuerregister, 4 Zeitkonstantenregister und 1 Interruptvektorregister) bestimmen die Betriebsarten, das Interruptverhalten, die Zeitgebergrundtakte, die Trigger-Flanken, die Trigger-Art, die Zeitkonstanten (Zählumfänge) der Kanäle und den Interruptvektor. In diesen Registern werden bestimmte Belegungen einzelner Bits als direkte Operationsanweisungen aufgefaßt - wie zum Beispiel "Abbruch der Operation", (reset) "Zeitkonstante laden folgt", "dieses Datenwort ist ein Kanalsteuerwort", und "dieses Datenwort ist ein Interruptvektor". Jeder der vier Kanäle kann unabhängig von den anderen Kanälen programmiert, gelesen und gestartet werden. Zusammenhänge zwischen den Kanälen bestehen nur durch ihre festgelegte Prioritätsreihenfolge bei Interrupts, d. h. sie können nur als Einheit von 4 Kanälen einer Interrupt-Prioritätskette in ein System einbezogen werden.

Der U 857 ist ein TTL-kompatibler Schaltkreis und arbeitet in positiver Logik. Die Anschlußbezeichnungen tragen Eigennamen-Charakter. Ein Querstrich (z. B. bei \overline{CS}) ist ein Hinweis auf den aktiven Pegel (z. B. die Leitung \overline{CS} ist bei Low-Pegel aktiv und IEI ist bei High-Pegel aktiv). Jedoch wird ein weggenommener Querstrich bei Logikdefinitionen auch für passive, d. h. High-Pegel verwendet (\overline{RD} wird als RD bezeichnet, z. B. im logischen Schaltbild). Aus dem weiteren Text ist jedoch die exakte Bedeutung ersichtlich. Für den Bausteinaktivierungseingang (Anschluß 16) sind die Bezeichnungen \overline{CE} und \overline{CS} gebräuchlich. Die Bedeutung ist in diesem Falle die gleiche. Der Systemtakt (Anschluß 15) wird mit dem Kurzzeichen C, die Periode mit t_c und die Taktzyklen bezüglich den Maschinenzyklen der CPU mit $T_1 - T_N$ und T_w, T_w^* bezeichnet.

2. Aufbau der U 857 D

Die einzelnen Funktionseinheiten der U 857 D sind im Bild 1 dargestellt. Vier unabhängige Zähler/Zeitgeber-Kanäle, eine interne Steuerlogik, eine Interruptsteuerlogik zur Realisierung der Vektor-Interrupt-Behandlung im System U 880 D sowie eine U 880 Systembus-Schnittstelle sind in der U 857 D integriert. Die vier Zähler/Zeitgeber-Kanäle werden durch die Zahlen 0 bis 3 gekennzeichnet. Die U 880 D-Systembus-Schnittstelle erlaubt ein direktes Zusammenschalten von CPU und CTC, ohne Verwendung zusätzlicher externer Logik. Nur Portadressen-Dekoder und/oder Bustreiber sind in der Praxis meistens erforderlich. Der CTC hat die Möglichkeit, für jeden seiner Kanäle einen eigenen Interruptvektor zu erzeugen - und so kanalspezifische Interrupt-Service-Routinen (ISR) zu adressieren. Die vier Kanäle können als vier aufeinanderfolgende Glieder in die U 880 D-Interrupt-Prioritätskette einbezogen werden - wobei Kanal 0 höchste und Kanal 3 niedrigste Priorität hat.

Durch Ein-/Ausgabebefehle der CPU bei Aktivierung des CTC's wird durch die binäre Belegung der Steuerleitungen KS1 und KSO immer nur jeweils einer der vier CTC-Kanäle adressiert. Bild 8 zeigt die Zuordnung der Kanäle zur Belegung der Leitungen KS1 und KSO. Meistens werden diese Leitungen mit den Adreßleitungen A_1 und A_0 der CPU verbunden, wodurch die Port-Adressen der vier Kanäle, entsprechend der Reihenfolge der Kanalnummern hintereinander liegen.

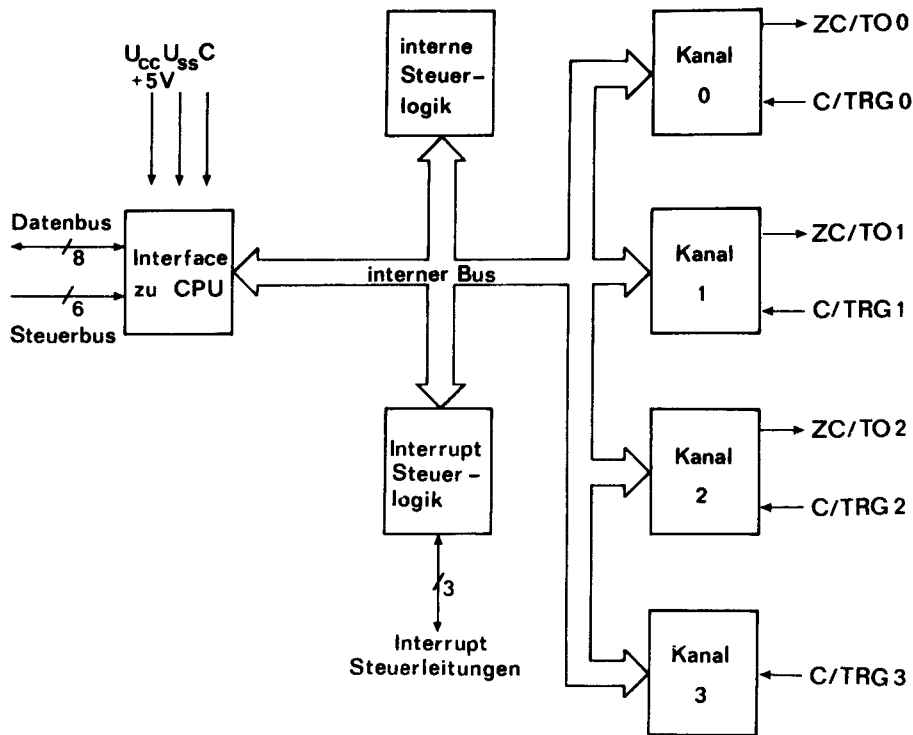


Bild 1: Blockschaltbild der U 857 D

Der Datenverkehr erfolgt über den Datenbus, d. h. über ihn werden CTC-Register beschrieben oder gelesen. Die Richtung des Datenverkehrs wird durch die Steuerleitung \overline{RD} bestimmt, das Ziel durch die Kanal-Portadressierung. Dadurch können die entsprechenden Register der Kanäle gezielt geladen werden. Neben der Kanal-Port-Adressierung ergibt sich das Ziel der übertragenen Daten auch aus deren Format, d. h. der Inhalt bestimmter Bits entscheidet über die Zielregister, bzw. auch der Inhalt vorangegangener Datentransfers.

2.1. Aufbau eines CTC-Kanals

Der Aufbau einer der vier Zähler/Zeitgeber-Kanäle ist in Bild 2 dargestellt. Ein Kanal besteht aus 2 Registern, 2 Zählern und der Steuerlogik. Ein 8 Bit-Zeitkonstanten-Register (ZKR) und ein 8 Bit-Kanalsteuerregister enthalten die Instruktionen und Parameter des Kanals.

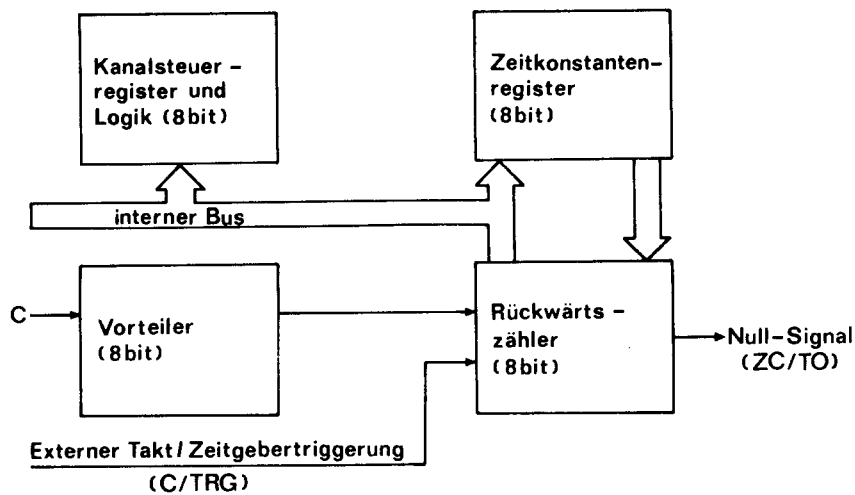


Bild 2: BlockschaItbild eines CTC-Kanals

Hauptbestandteil ist der lesbare 8 Bit-Rückwärtszähler, dessen Zählumfang zwischen 1 und 256 variierbar ist. Dieser Zähler zählt entweder die Impulsfolge am externen Takt-Eingang oder den Takt am Ausgang des 8 Bit-Vorteilers. Dadurch unterscheiden sich auch die beiden Betriebsarten Zähler oder Zeitgeber. Jeweils bei nulldurchgang wird kurzzeitig die Null-Signal-Leitung (ZC/TO_n) aktiviert und die Zeitkonstante übernommen. Der Zählerstand, d. h. die Anzahl der Schritte bis Null ist jederzeit auslesbar. Der Vorteiler ist ein 8 Bit-Zähler, der den Systemtakt C entweder durch 16 oder durch 256 teilt und in der Betriebsart Zeitgeber den Rückwärtszähler mit diesem vorgeteilten Systemtakt taktet. Der Teilerfaktor ist durch die CPU über das Kanalsteuerregister (Bit D5) programmierbar. In der Betriebsart Zähler wird der Vorteiler nicht verwendet.

Das Zeitkonstantenregister wird nach der Ladung des Konstantenregisters mit einer ganzzahligen Zeitkonstante zwischen 1 und 256 beschrieben. Es setzt den Rückwärtszähler zu Beginn einer Zähloperation und bei jedem nulldurchgang auf den programmierten Wert. Wird während einer laufenden Operation der Inhalt des Zeitkonstantenregisters geändert, so vollendet der Rückwärtszähler den Vorgang bis zum nulldurchgang und übernimmt die neue Konstante.

2.1.1. Format des Kanalsteuer-Registers

Dieses Register wird durch die CPU über die Kanal-Port-Adresse geladen, wobei Bit D0 logisch 1 sein muß. Der Inhalt dieses Registers bestimmt die Betriebsart und die Parameter des Kanals. Bild 3 zeigt allgemein die Bedeutung der 8 Bits des Kanalsteuer-Registers.

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt- freigabe 0: gesperrt 1: freigegeben	Betriebsart- auswahl 0: Zeitgeber 1: Zähler	Zeitgeber- Vorteiler 0: Faktor 16 1: Faktor 256	Trigger- flanke 0: negativ 1: positiv	Zeitgeber- triggerart 0: C/TRG passiv 1: C/TRG aktiv	Zeitkonstanten- ladung 0: folgt nicht 1: folgt	Rückstellen 0: nicht rücksetzen 1: rücksetzen	1

Bild 3: Format des Kanalsteuer-Registers

Wird der Inhalt von Bit D1 logisch 1, so entspricht das einem Rückstell-Befehl an den Kanal, die laufende Operation wird abgebrochen. Der Kanal wartet in Abhängigkeit von Bit D2 entweder auf eine Zeitkonstante (D2=1) oder auf ein neues Kanalsteuerwort. Bit D2=1 entspricht allgemein dem Befehl: nächstes Steuerwort ist Zeitkonstante. Die übrigen Bits legen die Betriebsart (D6), Operationsparameter (D5, D4, D3) und die Interrupt-fähigkeit (D7) des Kanals fest. Ausführlich wird das Kanalsteuerwort im Abschnitt 5.1. beschrieben.

2.2. Die Interrupt-Steuerlogik

Durch die Interrupt-Steuerlogik werden Anforderungen, Bestätigungen, Zwischenspeicherungen und Rücksetzungen von durch Zählernulldurchgängen erzeugten Interrupts behandelt. Jeder CTC-Kanal kann eine Interruptanforderung stellen, indem er die gemeinsame \overline{INT} -Leitung aktiviert. Voraussetzungen für eine Interruptanforderung sind die Interruptfähigkeit des Kanals (ist programmierbar durch Bit D7 des Kanalsteuerwortes, D7=1), ein aktives Signal auf der Leitung IEI des CTC's und daß kein Kanal mit niedrigerer Kanalnummer gleichzeitig eine Interruptanforderung stellt oder sich im Interruptzustand befindet (wird bedient mit noch nicht beendeter Interrupt-Service-Routine).

Auslösemoment einer Interruptanforderung ist eine Nulldurchgangsbedingung, d. h., daß der Kanal auf Null zählt. Zur Festlegung der Priorität der gemeinsamen über die Leitung \overline{INT} durch die Kanäle zur CPU gesendeten Interruptanforderungen sind die Kanäle intern durch jeweils zwei prioritätsbestimmende Kaskadierungsleitungen in der Reihenfolge der Kanalnummern verkettet (Bild 4).

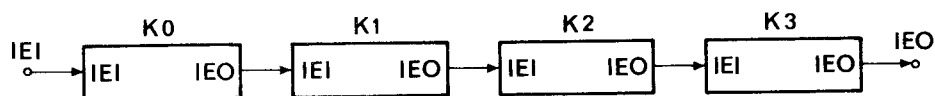


Bild 4: Interne Prioritätskette des CTC's

Durch diese Verkettung ist die Prioritätsreihenfolge der vier Kanäle festgelegt - Kanal 0 hat höchste, Kanal 3 niedrigste Priorität.

Ein Kanal kann die INT-Leitung nur dann aktivieren, wenn seine Interruptfähigkeit nicht durch vorangegangene Kanäle gesperrt wird. Mit der Aktivierung der $\overline{\text{INT}}$ -Leitung sperrt ein Kanal gleichzeitig die Interruptfähigkeit der nachfolgenden Kanäle, was nach außen durch ein passives Signal auf der Leitung IEO dokumentiert wird.

Der CTC kann als Einheit von vier Gliedern in den Aufbau einer Prioritätskette im System U 880 einbezogen werden.

Jeder Kanal kann durch einen Interrupt einzeln bedient werden, indem er durch seinen Interruptvektor die CPU veranlaßt, einen bestimmten Programmteil im Programmspeicher gezielt aufzurufen. Liegt eine Interruptanforderung auf der $\overline{\text{INT}}$ -Leitung an, wird durch die CPU anstelle des nächsten Befehlslese-Zyklus ein Interrupt-Bestätigungs-Zyklus (INTA), durchgeführt. Dieser Zyklus wird durch die Leitungen $\overline{\text{M1}}$ und $\overline{\text{IORQ}}$ dem CTC mitgeteilt, der daraufhin den Inhalt seines Interruptvektor-Registers auf den Datenbus gibt. Dabei werden durch die Interruptsteuerlogik die Datenbits D2 und D1 mit der binären 2 Bit-Adresse des interruptanfordernden Kanals belegt (entsprechend KS1 und KS0). Das Interruptvektor-Register muß in der Initialisierungsphase über die Portadresse von Kanal 0 geladen worden sein.

Das Format des Interruptvektor-Registers und die Modifikation bei Aussendung eines Kanal-Interruptvektor-Datenwortes sind im Bild 5 dargestellt.

Durch Bit DC=0 unterscheidet der CTC das zu ladende Steuerwort vom Kanalsteuerwort.

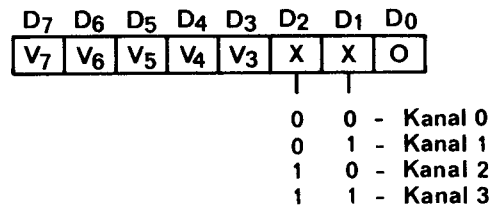


Bild 5: Format des Interruptvektor-Registers

Der Interruptvektor dient der CPU zum indirekten Aufruf eines Speicherplatzes (Interrupt-Mode 2, IM2) oder im selteneren Fall (Interrupt-Mode 0, IMO) als 1-Byte-Befehl nach dem Bestätigungszyklus (INTA). Im ersten Fall (IM2) bildet die CPU aus dem Inhalt ihres I-Registers die oberen 8 Bit und aus dem Interruptvektor des interruptanfordernden Kanals die unteren 8 Bits eines 16 Bit-Zeigers (Bild 6). Dieser Zeiger zeigt auf das erste (Low-Byte) von zwei aufeinanderfolgenden Bytes in einer Speichertabelle, in der die Adresse der Interrupt-Service-Routine des Kanals gespeichert ist (Interrupttabelle). Die CPU liest diese Startadresse und führt einen Unterprogrammaufruf zu dieser Adresse durch.

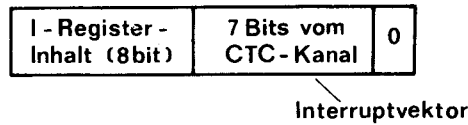


Bild 6: 16-Bit-Zeiger zur Interrupttabelle

Die Interrupt-Service-Routinen der CPU müssen mit dem Befehl RETI-Rücksprung vom Interrupt abschließen. Der CTC dekodiert die Datenfolge dieses Befehls (ED4D /Hex.) und erkennt dadurch den Interrupt-Abschluß der Bedienroutine des entsprechenden Kanals und gibt die Interrupt-Prioritätskette wieder frei. Die RETI-Erkennung wird nur dem Kanal zugeordnet, der sich als einziger Kanal im Interruptzustand (IEI aktiv, IEO passiv) befindet. Dadurch können verschachtelte Interruptanforderungen eindeutig bedient werden. Der CTC besitzt intern ein sogenanntes "Interrupt wird bedient"-Flipflop, das kanalorientiert ist. Dieses wird durch RETI zurückgesetzt.

3. U 857 D-Anschlußbeschreibung

Der CTC U 857 D wird in einem 28poligen Standard-Dual-In-Line-Plastgehäuse gefertigt. Bild 7 zeigt die Anschlußbelegung und das logische Schaltbild.

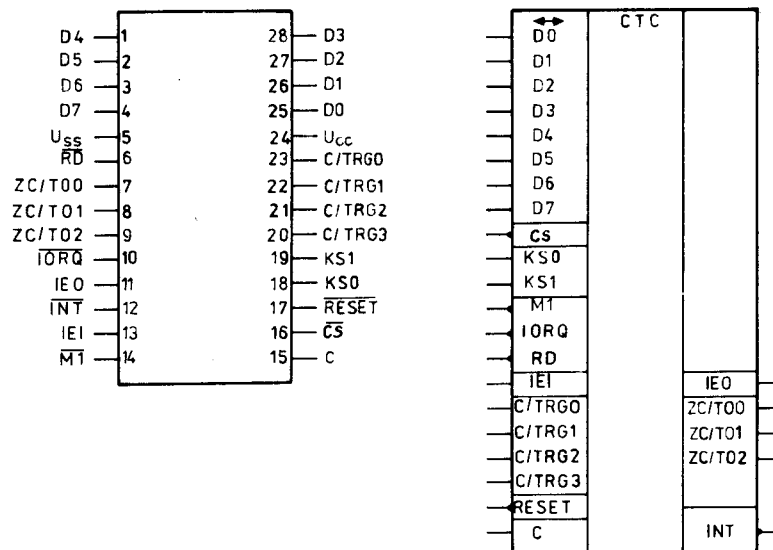


Bild 7: Anschlußbelegung und logisches Schaltbild

Die Anschlußleitungen des CTC haben folgende Funktion:

D7-DO U 880 D-Datenbus; bi-direktionale Tri-State-Leitungen; 8 Bit.
Dieser Bus wird für die Übertragung von Daten- und Steuerwörtern zwischen der CPU und dem CTC benutzt. Er besteht aus 8 Leitungen, wobei DO das niederwertigste (LSB) und D7 das höchstwertigste (MSB) Bit ist. Steuerwörter, Zeitkonstanten, Interruptvektoren und Zählerstände werden auf dem Datenbus übertragen.

CS Steuerleitung "Bausteinauswahl" (Chip Select); Eingang; Low aktiv, auch als \overline{CE} bezeichnet. Low-Pegel auf dieser Leitung ermöglicht dem CTC Steuerwörter, Interruptvektoren oder Zeitkonstanten-Datenwörter während Ein-/Ausgabe-Schreib-Zyklen der CPU zu übernehmen, oder den Inhalt der Zähler (Zählerstände) während E/A-Lese-Zyklen der CPU auf dem Datenbus auszugeben. In den meisten Anwendungsfällen wird dieses Signal durch Dekodierung der niederwertigen 8 Adreß-Leitungen der CPU gewonnen, wobei 2 Adreßbits zur Selektierung der Kanäle verwendet werden.

KS1/KS0 Kanal Selekt; Eingänge; High aktiv,
Durch die Belegung mit einer 2 Bit-Adresse wählen diese Steuerleitungen einen der vier CTC-Kanäle zur Kommunikation mit der CPU aus, d. h. sie dienen der Kanaladressierung bei Ein/Ausgabe-Zyklen der CPU.

	KS1	KS0
Kanal 0	0	0
Kanal 1	0	1
Kanal 2	1	0
Kanal 3	1	1

Bild 8: Kanalauswahl

Meistens werden diese Steuerleitungen mit den Leitungen A_1 und A_0 des CPU-Adressbusses belegt. Die Portadressen der vier Kanäle liegen dann in der Reihenfolge der Kanalnummern hintereinander.

$\overline{M1}$ Maschinenzyklus 1 der CPU; Eingang; Low aktiv,
Diese Steuerleitung wird von der CPU bei Befehlslesezyklen (zusammen mit \overline{RD} aktiv) und bei Interrupt-Bestätigungs-Zyklen (zusammen mit \overline{IORQ} aktiv) aktiviert. Beide CPU-Zyklen verwendet der CTC in der Interrupt-Behandlung. Das Auftreten von Interrupts (H/L-Übergang an \overline{INT}) wird für die Dauer von $\overline{M1}$ intern verhindert, damit sich die Prioritätskette sicher stabilisieren kann.

$\overline{\text{IORQ}}$	<p>Eingabe/Ausgabe-Anforderung (Input/Output Request); Eingang; Low aktiv Diese Steuerleitung wird von der CPU bei Ein/Ausgabezyklen und bei Interrupt-Bestätigungs-Zyklen (zusammen mit $\overline{\text{M1}}$ aktiv) aktiviert. Aktives $\overline{\text{IORQ}}$ ist Voraussetzung zur Datenübertragung zwischen CPU und CTC, d. h. zur Beschreibung der CTC-Register, zur Ausgabe der Interruptvektoren und zum Lesen der Zählerstände. Eine Ausnahme ist die Datenbusüberwachung zur RETI-Dekodierung.</p>
$\overline{\text{RD}}$	<p>Lese-Leitung (Read); Eingang; Low aktiv Diese Steuerleitung wird von der CPU bei Speicher-Lese-Zyklen ($\overline{\text{MREQ}}$ aktiv) und bei Ein/Ausgabe-Lese-Zyklen ($\overline{\text{IORQ}}$ aktiv) aktiviert. Der CTC verwendet diese Leitung zur Erkennung der Datenübertragungs-Richtung, bei aktivem $\overline{\text{RD}}$ in Richtung CPU und bei passivem $\overline{\text{RD}}$ in Richtung CTC. ($\overline{\text{IORQ}}$ und $\overline{\text{CS}}$ müssen aktiv sein.) Es existiert keine Schreibleitung. Dieses Signal wird intern erzeugt ($\overline{\text{WR}} = \text{RD} \cdot \overline{\text{IORQ}} \cdot \overline{\text{CE}} \cdot \text{M1}$). Ausnahme ist wieder die Datenbusübernahme zur RETI-Dekodierung (Befehlsholezyklus), Daten werden von der CPU und der CTC gelesen!</p>
$\overline{\text{RESET}}$	<p>Rücksetz-Leitung; Eingang; Low aktiv. Ein kurzzeitiger aktiver Pegel auf dieser Leitung, stoppt alle Operationen der Kanäle und setzt sie in einen definierten Anfangszustand. Nach Anschalten der Betriebsspannung ist dies notwendig (power-on-reset). Definierter Zustand nach kurzzeitiger aktiver $\overline{\text{RESET}}$-Leitung wird folgendermaßen charakterisiert:</p> <ul style="list-style-type: none"> - der Datenbus ist hochohmig - die Interruptprioritätskette ist durchgeschaltet (IEI = IEO) - die Interruptfähigkeit ist gesperrt (D7 des Kanalsteuerregisters ist zurückgesetzt) - das Interruptvektor-Register wird nicht verändert - die Kanäle erwarten Kanalsteuerwörter oder einen Interruptvektor <p>Das Signal muß mindestens 3 Taktperioden lang aktiv sein.</p>
C	<p>Systemtakt (clock); Eingang; Einphasen 5-V-Takt Diese Leitung dient zur Synchronisation aller Operationen zwischen CTC und CPU. Bei Änderung der Periode innerhalb der Grenzwerte ist die Funktionsfähigkeit des Systems gewährleistet.</p>
$\overline{\text{INT}}$	<p>Interrupt-Anforderung (Interrupt Request) Ausgang; Open Drain; Low aktiv. Diese Leitung wird aktiv, wenn ein CTC-Kanal, der zur Interruptfähigkeit programmiert wurde, eine Nulldurchgangsbedingung seines Zählers erfüllt. Die Erkennung, welcher der Kanäle diese Leitung aktiviert, erfolgt durch die Kaskadierungsleitungen IEI und IEO. $\overline{\text{INT}}$ ist ein Open Drain-Ausgang, also zusammenschaltbar mit weiteren Open-Drain-Ausgängen.</p>
IEI	<p>Interruptfreigabe-Leitung; Eingang; High aktiv, Diese prioritätsbestimmende Kaskadierungsleitung ist logisch Kanal 0 zugeordnet, der die höchste Priorität besitzt. Aktiver Pegel auf dieser Leitung signalisiert den Kanälen Interruptmöglichkeit. Sind die logischen Pegel von IEI und IEO identisch, ist kein Kanal im Interrupt-Zustand (IEI = IEO), sind sie unterschiedlich (IEI = High, IEO = Low), wird ein</p>

Kanal interruptmäßig bedient. Low-Pegel auf dieser Leitung sperrt die Interruptfähigkeit aller CTC-Kanäle.

IEO Interruptfreigabe-Ausgang, Ausgang, High aktiv.
Diese prioritätsbestimmende Kaskadierungsleitung ist logisch Kanal 3 zugeordnet, der niedrigste Priorität besitzt. Aktiver Pegel auf dieser Leitung signalisiert einem folgenden Schaltkreis Interruptfreigabe, d. h. IEI des CTC's ist aktiv und kein Kanal wird interruptmäßig bedient. Passiver Pegel auf IEO ist nur durch RETI-Dekodierung aktivierbar. (Im CTC oder bei höherpriorisierten Bausteinen)

C/TRG 0-3 Externer Takt/Zeitgebertrigger-Leitungen 0-3
(Externer Clock/Timer-Trigger); Eingänge; wahlweise High oder Low aktiv.
Je nach Betriebsart des Kanals werden diese peripheren Kanal-Leitungen als externe Takteingänge von Zähler-Kanälen oder als Zeitgeber-Trigger-eingänge von Zeitgeber-Kanälen verwendet. Die aktiven Flanken sind durch den Anwender wählbar. Die Triggerung von Zeitgeber-Kanälen kann wahlweise entfallen (Software-Triggerung).

ZC/TO 0-2 Nulldurchgangs-Signal/Zeitgeber-Leitungen 0-2
(Zero Count/Timeout 0-2); Ausgänge; High aktiv.
Durch Anschlußbegrenzung auf 28 Leitungen hat Kanal 3 keinen Ausgang. Auf diesen peripheren Kanalleitungen senden die CTC-Kanäle unabhängig von ihrer Betriebsart bei Nulldurchgang ihrer Rückwärtszähler High-Impulse. Zählerkanäle sind über diese Leitungen (verbunden mit C/TRG eines anderen Kanals) kaskadierbar. Zeitgeber-Kanäle führen auf diesen Leitungen kontinuierliche Impulsfolgen (bei gleichbleibender Zeitkonstante + Betriebsart) mit der Periode

$$t_c \cdot p \cdot T_c$$

wobei t_c : Systemtaktperiode
 p : Vorteilerfaktor (16 oder 256)
 T_c : Zeitkonstante (zwischen 1 und 256)

zu beachten ist, daß die Impulse nur eine Breite von minimal 200 ns und maximal 600 ns haben.

4. Betriebs- und Operationsarten der U 857 D

Nach Anlegen der Betriebsspannung ist ein kurzzeitiges aktives RESET-Signal (Power-on-reset) notwendig, um den CTC in einen definierten Anfangszustand zu bringen. Daraufhin erwartet der CTC für seine Kanäle und die Interrupt-Steuerlogik Arbeitsinstruktionen, die er durch Ausgabebefehle der CPU über den Datenbus erhält und in die ausgewählten Register lädt. Die Zielregister dieser 8 Bit-Worte werden durch die Kanaladresse, durch das Format der Worte (Bit D0 unterscheidet Kanalsteuerwort und Interruptvektor) und durch den Inhalt des vorangegangenen Kanalsteuerwortes (nächstes Steuerwort ist Zeitkonstante) bestimmt: Der eigentliche Arbeitsbefehl an einen Kanal wird in das Kanalsteuerregister geladen. Der Inhalt dieser Register enthält die Betriebsartenauswahl (Zähler: D6=1, Zeitgeber: D6= 0), Operationsparameter innerhalb dieser Betriebsarten (Triggerflanken/D4, Zeitgeber-Triggerart/D3, Zeitgeber-Vorteilerfaktor/D5) und allgemeine betriebsartunabhängige Befehle (Interruptfähigkeit/D7, nächstes Steuerwort ist Zeitkonstante/D2, Rückstellbefehl). Das Kanalsteuerregister eines Kanals hat keinen Einfluß auf die anderen drei Kanäle. Der Zählumfang eines Kanals (seines Rückwärtszählers) wird durch den Inhalt seines Zeitkonstantenregisters bestimmt, das einmalig oder wiederholt durch eine Ladeforderung im vorangegangenen Kanalsteuerwort von der CPU in das entsprechende Register geladen werden kann.

Das Zeitkonstanten-Datenwort kann einen ganzzahligen Wert zwischen 1 und 256 annehmen, wobei das 8 Bit-Wort OO/Hex einer Zeitkonstanten von 256 entspricht. Bei Nulldurchgang des Kanals (bzw. seines Rückwärtszählers) beginnt die Zähloperation mit dem Inhalt des Zeitkonstantenregisters erneut, egal ob inzwischen ein neues Zeitkonstanten-Datenwort geladen wurde. Eine Zeitkonstantenänderung wird erst bei Nulldurchgang wirksam. Der Zählerstand OOH wird nicht wirksam und durch den Wert der Zeitkonstanten ersetzt. Der Ladevorgang des Zeitkonstantenregisters kann auch zum Start einer Operation verwendet werden (Software-Triggerung). Das Zeitkonstantenregister eines Kanals hat keinen Einfluß auf die anderen drei Kanäle. Ohne Ladung des Zeitkonstantenregisters ist keine Operation möglich. Die Arbeitsweise der Rückwärtszähler entspricht in beiden Betriebsarten der eines Ringzählers mit programmierbarem Zählumfang. Jeweils bei Nulldurchgang wird der Wert für die folgende Zählrunde (Zeitkonstante) automatisch in den Rückwärtszähler übernommen, am Ausgang ZC/TO ein High-Impuls gesendet (minimale Pulsbreite = 200 ns) und die Interruptsteuerlogik bei entsprechenden Bedingungen aktiviert. In beiden Betriebsarten sind durch Ein/Ausgabe-Lese-Befehle der CPU momentane Zählerstände abfragbar, d. h. die Anzahl der noch erforderlichen Dekrementierungen des Rückwärtszählers bis zum Nulldurchgang. Dadurch kann der CTC auch zum Polling-Betrieb genutzt werden, d. h. zur Software-Abfrage oder Suche eines bestimmten Zwischenzählerschrittes. In beiden Betriebsarten wird durch Ausgabe-Befehl der CPU zum Kanalsteuerregister mit $D_1=1$ der entsprechende Kanal zurückgesetzt, d. h. der laufende Zählvorgang wird abgebrochen und der Kanal erwartet ein neues Steuerwort und/oder eine neue Zeitkonstante. Je nach Interrupt-Modus der CPU, entweder zur Adressierung der richtigen Interrupt-Service-Routine (Interrupt-Mode 2, IM2) oder zur Bereitstellung einer CPU-Instruktion nach Interruptbestätigung (Interrupt Mode 0, IMO, unwahrscheinlicher Anwendungsfall) muß der CPU vom CTC ein 8 Bit-Datenwort, bezeichnet als Interruptvektor, zugeleitet werden. Vor Auftreten der ersten Interrupt-Anforderung muß der jeweils gewünschte Vektor durch ein Ein-/Ausgabe-Schreibbefehl der CPU über die Port-Adresse von Kanal 0 geladen worden sein. Bit $DO=0$ definiert dieses Steuerwort als Interruptvektor. Die Bits D2 und D1 des Steuerwortes "Interruptvektor" werden bei bestätigtem Interrupt mit der binären 2 Bit-Kanaladresse des interruptanfordernden, bestätigten Kanals belegt, wodurch jedem Kanal ein Interruptvektor zugeordnet ist. Die niederwertigsten 3 Bits der Kanal-Interruptvektoren sind entsprechend dem Kanal konstant. Unter der Voraussetzung, daß die CTC-Kanäle interruptfähig sind ($IEI = \text{High}$, Kanalsteuerbit $D7=1$), kann jeder Kanal bei jedem Nulldurchgang eine Interruptanforderung stellen. Interrupts werden jedoch nur in entsprechender Prioritätsreihenfolge behandelt (Kanal 0 ... Kanal3), d. h. niederpriorisierte Kanäle werden, wenn sie gerade von höherpriorisierten Kanälen gesperrt werden, gespeichert und nach Bedienung des höherpriorisierten Kanals abgearbeitet.

4.1. Betriebsart "Zähler" der U 857 D

Hat Bit D6 des Kanalsteuerregisters den Wert "1", ist der Kanal als "Zähler" programmiert, d. h. der Kanal ist ein Zähler-Kanal. Bild 9 zeigt den Aufbau eines Zähler-Kanals des CTC's. Ein Zähler-Kanal zählt die extern an seinem Eingang C/TRG anliegende Impulsfolge. Die maximale Zählfrequenz beträgt 1,25 MHz. Gesteuert wird ein Zähler-Kanal durch Ausgabebefehle der CPU. Das Laden des Zeitkonstanten-Registers startet den Zählbetrieb bei der Initialisierung. Bit D4 des Kanalsteuerregisters bestimmt die auslösende Triggerflanke am Eingang C/TRG ($D4=0$: negative Triggerflanke, $D4=1$: positive Triggerflanke).

Der Rückwärtszähler übernimmt die aktiven Flanken am Eingang C/TRG synchron zu den steigenden Flanken des Systemtaktes t_c . Die Impulsfolge am Eingang C/TRG kann vollkommen asynchron empfangen werden, jedoch muß die auslösende Flanke eine Mindest-Setzzeit t_s (SK) vor der steigenden Flanke von t_c anliegen, um garantiert mit dieser Flanke übernommen zu werden. Dadurch ist bei Asynchronbetrieb zwischen nulldurchgangs-auslösender Flanke am Eingang C/TRG und L/H-Flanke am Ausgang ZC/TO ein Verzögerungsfehler möglich, der sich aus der o. g. Setzzeit, der Taktperiode und der Verzögerungszeit am Ausgang ZC/TO ($t_{DH}(CK)$) ergibt:

$$t_s(CK)_{\min} + t_c + t_{DH}(ZC)$$

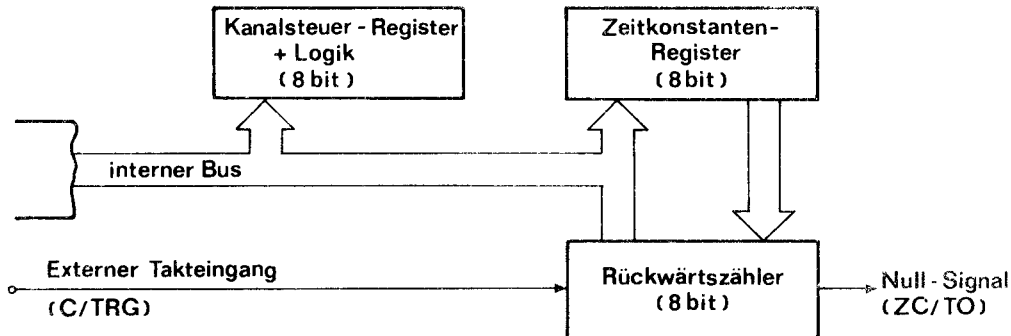


Bild 9: Zähler-Kanal der U 857

Maximal kann diese Zeit 810 ns betragen (bei 2,5 MHz Takt). Für die Impulsfolge am Eingang C/TRG wird eine minimale High- und Low-Pulsbreite von je 120 ns und TTL-Pegel gefordert. Soll die Zählfolge 8 Bits überschreiten, können mehrere Kanäle (theoretisch beliebig viele) in Reihe geschaltet werden, d. h. der Zählerausgang ZC/TO des einen Kanals wird mit dem Zählereingang C/TRG des folgenden Kanals verknüpft. Wenn es die erforderliche Zählergebnis-Reaktionszeit und die Struktur des Mikrorechner-Programms zulässt, ist der Zählumfang softwaremäßig beliebig erweiterbar (über Software-Zählung von Interruptfolgen). Kanal 3 ist nur softwaremäßig auswertbar (durch fehlenden Ausgang ZC/TO).

4.2. Betriebsart "Zeitgeber" der U 857 D

Hat das Kanalsteuerbit D6 den Wert "0", ist der Kanal als "Zeitgeber" programmiert, d. h. der Kanal ist ein Zeitgeber-Kanal. Bild 10 zeigt den Aufbau eines Zeitgeber-Kanals des CTC's.

Ein Zeitgeber-Kanal zählt den Systemtakt t_c , d. h. er teilt ihn mit seinem programmierbaren Vorteiler durch 16 (Kanalsteuerbit D5=0) oder durch 256 (Kanalsteuerbit D5=1) und mit seinem Rückwärtszähler durch einen Wert zwischen 1 und 256. Dadurch ergibt sich am Ausgang ZC/TO eine Impulsfolge mit einem ganzzahligen Teilverhältnis zur 16- oder 256fachen Systemtaktperiode. Eine synchrone Interruptfolge ist programmierbar (Kanalsteuerbit D7=1). Bei gleichbleibender Zeitkonstante ergibt sich am Zählerausgang ZC/TO eine exakte periodische Impulsfolge aus dem Produkt:

$$t_c \cdot p \cdot TC,$$

wobei t_c die Systemtaktperiode, p der Vorteilerwert (16 oder 256) und TC die vorprogrammierte Zeitkonstante ist.

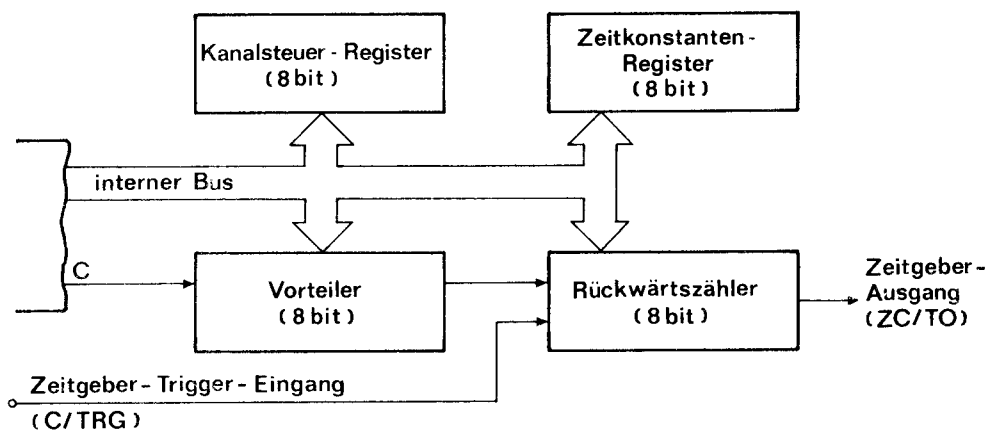


Bild 10: Zeitgeber-Kanal der U 857

Abhängig vom Kanalsteuerbit D3 wird der Zeitgebervorgang entweder durch den Ladezeitpunkt des Zeitkonstantenregisters (Softwaretriggerung, D3=0) oder durch eine aktive Flanke am Eingang C/TRG (Hardwaretriggerung, D3=1) gestartet. Die aktive Flanke am Eingang C/TRG ist durch Kanalsteuerbit D4 wählbar (D4=0: negative Flanke, D4=1: positive Flanke). Bei Softwaretriggerung beginnt die Zeitmessung (erstes Dekrementieren des Vorteilers) mit der steigenden Flanke von T2 des CPU-Maschinen-Zyklus, der auf den Ausgabebefehl zur Ladung des Zeitkonstantenregisters folgt.

Bei Hardwaretriggerung beginnt die Zeitmessung mit der zweiten oder dritten steigenden Systemtaktflanke nach der Triggerflanke am Eingang C/TRG, die der Ladung des Zeitkonstantenregisters bzw. des Kanalsteuerregisters (falls Kanalsteuerbit D2=0) folgt. Für die aktive Flanke am Eingang C/TRG wird eine Mindest-Setzzeit ($t_s(\text{TR})$) vor Auftreten der steigenden Systemtaktflanke gefordert, um schon mit der darauffolgenden steigenden Systemtaktflanke den Vorteiler zu dekrementieren. Bei Asynchronbetrieb kann dadurch ein Zeitmeßfehler am Ausgang ZC/TO auftreten, der sich aus der Systemtaktperiode t_c , der minimalen Triggereingangssetzzeit $t_s(\text{TR})$ und der Verzögerungszeit $t_{\text{DH}}(\text{ZC})$ am Ausgang ZC/TO ergibt:

$$t_s(\text{TR}) + t_c + t_{\text{DH}}(\text{ZC})$$

Maximal kann diese Zeit 810 ns betragen (bei 2,5 MHz Takt).

Zeitgeber-Kanäle sind nicht kaskadierbar, ihr Zählumfang beträgt maximal 16 Bit. Man kann jedoch einen Zeitgeber-Kanal durch Zähler-Kanäle aufrüsten, indem der Ausgang ZC/TO des Zeitgebers mit dem Eingang C/TRG eines folgenden Zählerkanals verbunden wird usw. Kanal 3 besitzt keinen Zeitgeberausgang ZC/TO und muß softwaremäßig behandelt werden.

5. Programmierung der U 857 D

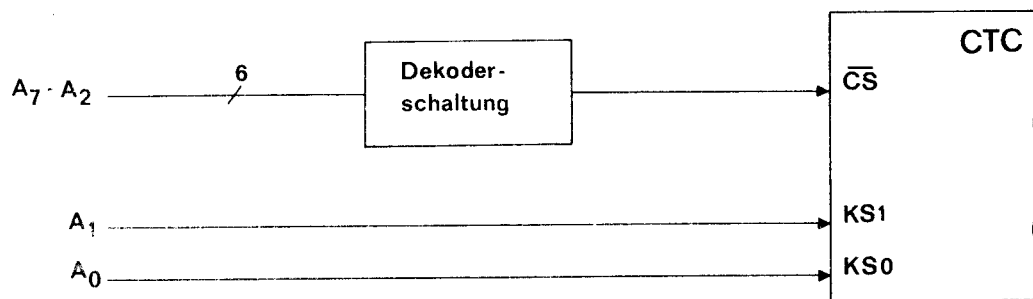
Zur Initialisierung der Kanäle des CTC als Zähler oder Zeitgeber müssen durch die CPU die entsprechenden Datenwörter in die Kanalsteuer- und Zeitkonstantenregister der Kanäle geladen werden. Zusätzlich muß, wenn mindestens einer der Kanäle durch gesetztes Kanalsteuerbit D7 zur Anforderung von Interrupts programmiert wurde, ein Interruptvektor in

das Interruptvektor-Register des CTC's geladen werden. Infolge der automatischen Kanal-Interruptvektorerzeugung genügt ein Datenwort "Interruptvektor" für alle vier Kanäle. Die Initialisierung des CTC's ist eine einfache Ausgabe-Befehlsfolge der CPU an die Portadressen der Kanäle, wobei das Format und das Format des vorangegangenen Datenwortes die Zielregister in den Kanälen bestimmen (DO=1: Kanalsteuerregister, DO=0: Interruptvektorregister und Kanalsteuerbit D2=1: nächstes Datenwort ist Zeitkonstante). Zum erweiterten Initialisierungsprogramm bei Vektor-Interruptbetrieb des CTC's gehören das Laden des I-Registers der CPU und der Aufbau der Startadressenliste der Interrupt-Service-Routinen (siehe Punkt 6.).

Die Portadressen der vier Kanäle ergeben sich aus der Belegung der Leitungen \overline{CS} , KS1 und KSO, wobei \overline{CS} den Baustein und KS1/KSO den Kanal selektieren. In den meisten Anwendungsfällen werden die Leitungen KS1/KSO mit den Adreßleitungen A1/A0 der CPU belegt, so daß die vier Kanäle eines CTC-Bausteins vier aufeinanderfolgende Portadressen erhalten. Die Portadressen ergeben sich aus den niederwertigen 8 Adreßbits der CPU. Die Wertigkeitsreihenfolge der Bits entspricht der Reihenfolge der Indexpzahlen (A0 ist LSB, A7 ist MSB).

Eine eindeutige Portadresse wird dekodiert, wenn der Ausgang des Dekoders nur dann aktiv ist, wenn die Adreßleitungen definierte logische Zustände besitzen.

Beispiel: A1 und A0 der CPU werden mit KS1 und KSO des CTC's zur Kanalselektion verbunden. Die oberen 6 Adreßbits werden zur Bausteinauswahl (Aktivierung der \overline{CS} -Leitung der CTC) benutzt.



Dekoderschaltung: $\overline{CS} = \overline{A7} \cdot A6 \cdot \overline{A5} \cdot A4 \cdot A3 \cdot A2$

Portadressen:	A7	A6	A5	A4	A3	A2	A1	A0	Hex-Adresse
Kanal 0	0	1	0	1	1	1	0	0	5CH
Kanal 1	0	1	0	1	1	1	0	1	5DH
Kanal 2	0	1	0	1	1	1	1	0	5EH
Kanal 3	0	1	0	1	1	1	1	1	5FH

Bild 11: Portadressen-Beispiel

Sind bei Ein/Ausgabe-Befehlen der CPU die niederwertigen 8 Adreßbits mit einer der Portadressen belegt, so werden durch den CTC-Kanal bei E/A-Schreibbefehlen Daten vom Datenbus übernommen und bei E/A-Lesebefehlen momentane Zählerstände auf den Datenbus ausgegeben.

5.1. Programmierung der Kanalsteuer-Register

Ein Datenwort wird als Kanalsteuerwort erkannt, wenn Bit DO dieses Wortes den Wert 1 hat und kein Zeitkonstanten-Datenwort erwartet wird (durch vorangegangenes Kanalsteuerwort mit D2=1). Die Kanalsteuerbits sind exakte Instruktionen für den Kanal, deren Bedeutung in der folgenden Auflistung detailliert beschrieben ist:

D7	D6	D5	D4	D3	D2	D1	D0
Interrupt-freigabe	Betriebsart-auswahl	Zeitgeber-Vorteiler-Faktor	Trigger-flanke	Zeitgeber-Triggerart	Zeitkonstante laden	Rückstellen (reset)	1

Bild 12: Das Kanalsteuerwort

Bit 7=1 Interruptfreigabebit gesetzt. Der Kanal wird befähigt, eine Interruptanforderungsfolge zu erzeugen (Interrupt Enabled). Ein Interrupt wird angefordert, wenn der Rückwärtszähler einen Nulldurchgang erreicht. Ist dieses Bit in einem der Kanäle gesetzt, so ist es erforderlich, einen Interruptvektor vor Beginn der Operation in den CTC zu laden. Interruptanforderungen werden in den Betriebsarten Zähler oder Zeitgeber gleichermaßen behandelt. Erreicht ein Kanal eine interne Interruptbedingung, hat aber momentan keine Priorität, so wird die Forderung gespeichert.

Bit 7=0 Interruptfreigabebit ist zurückgesetzt. Der Kanal kann keine Interruptanforderung stellen (Interrupt Disabled). Eine anhängige Interruptanforderung wird gelöscht.

Bit 6=1 Betriebsartauswahlbit: "Zähler". Der entsprechende Kanal ist ein Zähler-Kanal. Er wird mit jeder Triggerflanke des am Eingang C/TRG anliegenden Taktsignals dekrementiert. Kanalsteuerbit 4 definiert die Triggerflanke. Der Vorteiler wird nicht benutzt.

Bit 6=0 Betriebsartauswahlbit: "Zeitgeber". Der entsprechende Kanal ist ein Zeitgeber-Kanal. Der Kanal wird mit dem Systemtakt t_c dekrementiert. Ein Zeitgeberkanal besteht aus zwei 8 Bit-Zählern, dem Vorteiler und dem Rückwärtszähler. Am Zeitgeberausgang ZC/TO entsteht bei gleichbleibender Zeitkonstante TC (zwischen 1 und 256) und gleichbleibendem Vorteilerwert p (16 oder 256) eine kontinuierliche Impulsfolge mit der Periode

$$t_c \cdot p \cdot TC$$

Eine Interruptfolge der gleichen Periode ist programmierbar.

Bit 5=1 Zeitgebervorteilerbit: Faktor 256. Nur für Zeitgeber-Kanäle von Bedeutung. Der Systemtakt wird vom Vorteiler durch 256 geteilt.

Bit 5=0 Zeitgebervorteilerbit: Faktor 16. Nur für Zeitgeber-Kanäle von Bedeutung. Der Systemtakt wird vom Vorteiler durch 16 geteilt.

Bit 4=1 Triggerflankenbit: positive Flanke aktiv. Zeitgeber-Kanal beginnt die Operation nach positiver Flanke am Eingang C/TRG (bei Hardware-Triggerung, D3=1). Zähler-Kanal wird mit positiver Flanke dekrementiert.

Bit 4=0 Triggerflankenbit: negative Flanke aktiv. Zeitgeber-Kanal beginnt die Operation nach negativer Flanke am Eingang C/TRG (bei Hardware-Triggerung, D3=1). Zähler-Kanal wird mit negativer Flanke dekrementiert.

Bit 3=1 Zeitgebertriggerbit: Triggereingang C/TRG aktiv. Nur für Zeitgeber-Kanäle von Bedeutung. Zeitgeberoperation wird durch aktive Flanke am Eingang C/TRG gestartet. Der Eingang C/TRG wird mit der steigenden Systemtaktflanke T2 des CPU-Maschinenzyklusses aktiviert, der auf das Laden der Zeitkonstanten folgt. Der Zeitgeber-Kanal wird 2 oder 3 Taktperioden nach erfolgter aktiver Flanke am Triggereingang dekrementiert, je nach eingehaltener Setzeit $t_p(TR)$ (Hardware-Triggerung).

Bit 3=0 Zeitgebertriggerbit: Triggereingang C/TRG passiv. Nur für Zeitgeber-Kanäle von Bedeutung. Zeitgeberoperation beginnt mit der steigenden Systemtaktflanke T2 des CPU-Maschinenzyklusses, der auf das Laden der Zeitkonstanten folgt (Software-Triggerung).

Bit 2=1 Zeitkonstantenladebit: Ladung folgt. Das nächste Datenwort, das in den Kanal geschrieben wird, ist die Zeitkonstante und wird im Zeitkonstantenregister abgespeichert; unabhängig vom Format dieses Datenwortes. Eine Zeitkonstanten-Neuladung kann bei laufender Operation erfolgen und wird bei Nulldurchgang wirksam. Dazu muß Kanalsteuerbit 2 vorher gesetzt werden, auch bei laufender Operation möglich.

Bit 2=0 Zeitkonstantenladebit: keine Ladung folgt. Diese Bedingung ist nur sinnvoll, wenn bei laufender Operation Kanalsteuerbits geändert werden, ohne eine neue Zeitkonstante zu benötigen. Ein Kanal arbeitet nicht ohne korrekte Zeitkonstante, d. h. notwendige Bedingung zum Operationsstart ist immer ein Ladevorgang zum Zeitkonstantenregister, also ein gesetztes Zeitkonstantenladebit in der Initialisierungsphase.

Bit 1=1 Rückstellbit, gesetzt. Kanal stoppt die laufende Operation. Die Register werden nicht verändert. Dieses Bit wirkt nur unmittelbar nach seiner Ladung. Ist das Zeitkonstantenladebit gesetzt, beginnt die Operation wieder nach Ladung der Zeitkonstante, ansonsten ist eine Neuinitialisierung des Kanals erforderlich.

Bit 1=0 Rückstellbit, passiv. Kanal wird nicht beeinflusst.

Die einzelnen Kanalsteuerbits können nur geschlossen als Kanalsteuerwort (Bild 12 zeigt dessen Format) geladen werden. Falls nicht gerade eine Zeitkonstante erwartet wird, ist eine Änderung der Kanalsteuerbits bei laufender Operation möglich, und wird unmittelbar wirksam. Der Inhalt der Kanalsteuerregister ist durch die CPU nicht auslesbar. Bei Neuladung des Kanalsteuerregisters ohne Kanalrücksetzung werden nur Bit-Änderungen wirksam, d. h. Kanalsteuerbits, die keine Änderung erfahren bewirken keinen Störeffekt durch ihre Neuladung. Das Kanalsteuerwort wird vom Anwender entsprechend den gewünschten Betriebsbedingungen des Kanals zusammengestellt, und dann im Initialisierungsprogramm mit einem Ausgabebefehl an die Portadresse des Kanals verknüpft.

5.2. Programmierung der Zeitkonstanten-Register

Das Zeitkonstantenregister kann nur als Folge der Ladung des Kanalsteuerregisters des entsprechenden Kanals geladen werden. Voraussetzung ist ein gesetztes Zeitkonstantenladebit (Bit 2) im vorangegangenen Kanalsteuerwort. Beide Datenwörter werden durch Ausgabebefehle der CPU an die entsprechende Kanal-Portadresse über den Datenbus in den CTC geladen. Erst nach einer minimalen Ladefolge: Kanalsteuerwort - Zeitkonstanten-Datenwort kann ein Kanal mit der Operation beginnen. Das Zeitkonstanten-Datenwort kann einen ganzzahligen Wert zwischen 1 und 256 annehmen. Wenn alle 8 Bits dieses Wortes 0 sind, wird es als 256 interpretiert. Das Zeitkonstantenregister kann während laufender Operation geändert werden, wobei die Änderung erst bei Nulldurchgang wirksam wird. Die Wertigkeit der Zeitkonstantenbits ZK7 - ZK0 entspricht der der Datenbits D7 - D0.

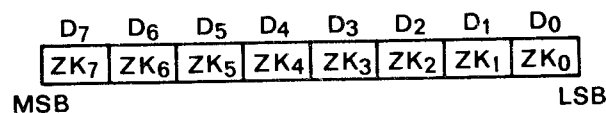


Bild 13: Das Zeitkonstanten-Register

Der Wert des Zeitkonstantenwortes entspricht der Anzahl der Zählschritte bis zum Null-durchgang des Rückwärtszählers. Der Zählerstand "Null" wird durch den Wert der Zeitkonstanten ersetzt, d. h. der "nullte" Zählschritt entspricht dem ersten der folgenden Zählrunde.

5.3. Programmierung des Interruptvektor-Registers

Das Interruptvektor-Register des CTC's wird durch einen Ausgabebefehl der CPU an die Portadresse von Kanal 0 über den Datenbus geladen. Durch die Bedingung Bit 0=0 wird das Datenwort als Interruptvektor interpretiert und vom Kanalsteuerwort unterschieden. Das Interruptvektor-Register braucht nur bei Anwendung von Interrupts geladen werden. Es sollte vor dem Kanalsteuerregister von Kanal 0 geladen werden, um nicht als Zeitkonstante interpretiert zu werden. Die Bits D2 und D1 des Interruptvektor-Datenwortes sind beim Laden gleichgültig. Wenn ein Kanal einen Interruptvektor über den Datenbus ausgibt, werden diese Bits durch die Interruptsteuerlogik mit dem 2 Bit-Binär-Kode des interruptanfordernden, von der CPU bestätigten Kanals belegt. Dadurch sind die unteren 3 Bits der Kanal-Interruptvektoren je Kanal festgelegt (Bit 0=0!). Die Startadressen der Interrupt-Service-Routinen der vier Kanäle liegen in der Startadressenliste hintereinander, d. h. sie bilden eine Interrupttabelle.

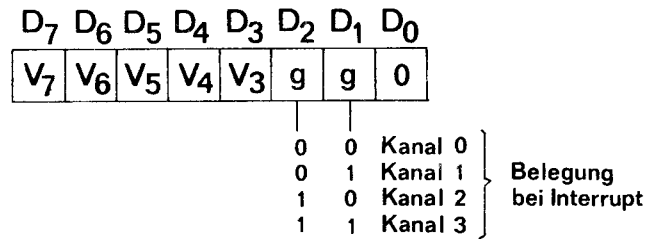


Bild 14: Aufbau des Interruptvektor-Datenwortes

Um die Startadressen zu lesen, muß die CPU jeweils kanalspezifische 16 Bit-Adressen bilden, deren höherwertigere 8 Bits dem I-Register der CPU entnommen werden, und deren niederwertigere 8 Bits die interruptanmeldenden, bestätigten Kanäle durch ihre Kanal-Interruptvektoren liefern.

Diese 16 Bit-Adressen zeigen jeweils auf das erste, zweier aufeinanderfolgenden Bytes, die die Startadressen der Interrupt-Service-Routinen (ISR) enthalten (Low-Byte vor High-Byte!). Die CPU führt Unterprogrammaufrufe zu diesen Startadressen durch.

Eine Neuladung des Interruptvektorregisters ist bei laufender Operation möglich und wird im erstmöglichen nachfolgenden Interrupt-Annahme-Zyklus wirksam.

5.4. Ablauf der Programmierung der U 857 D

Folgende Fakten müssen bereitstehen:

Hardware-Konfiguration des CTC - Portadressen

- Beschaltung als Zeitgeber oder Zähler
- Problemspezifische Betriebsbedingungen

Daraus ermittelt man die Datenwörter (Hexadezimalzahlen) für das Kanalsteuerwort, die Zeitkonstante und den Interruptvektor falls erforderlich.

Initialisierungsprogramm: Voraussetzung ist definierter Anfangszustand (Power-On-Reset)

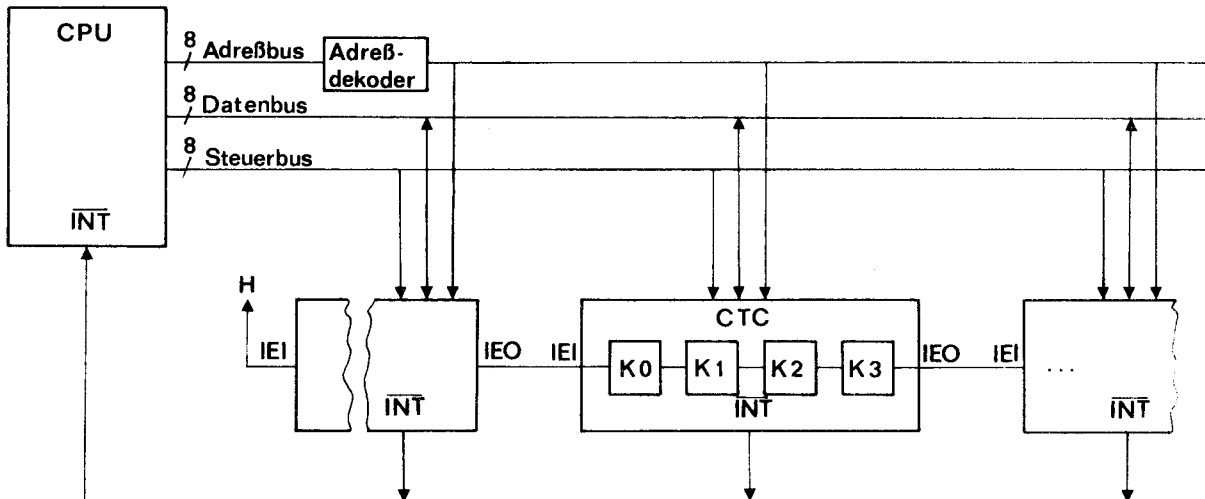
1. LD A,INTV ; Lade CPU-Register mit 8 Bit Wort: "Interruptvektor"
2. OUT KO ; Ausgabe des CPU-Registers über Portadresse von Kanal 0
3. LD A,KWKO ; Lade CPU-Register mit 8 Bit Wort "Kanalsteuerwort Kanal 0"
4. OUT KO ; Ausgabedes CPU-Registers über Portadresse von Kanal 0
5. LD A,ZKKO ; Lade CPU-Register mit 8 Bit Wort "Zeitkonstante für Kanal 0"
6. OUT KO ; Ausgabe des CPU-Registers über Portadresse von Kanal 0

Schritt 3-6 kann bei Bedarf für weitere Kanäle entsprechend wiederholt werden. Schritt 1-2 ist nur bei Interruptbetrieb erforderlich. Bei Interruptbetrieb muß noch das I-Register der CPU geladen, das Interruptflipflop (IFF1) der CPU gesetzt (Enable Interrupt, EI), die Startadressenliste der ISR aufgebaut und die Interrupt-Service-Routinen (Unterprogramme) programmiert werden. Ansonsten kann nach Schritt 6 zur Auswertung - eine Zählerstandsabfrage (Polling) - oder z. B. auch eine Port-Abfrage der Nullsignal-Leitungen (Beschaltung z. B. an eine PIO-Portleitung) erfolgen. Die Möglichkeiten können je nach Anwendungsfall genutzt werden. Die Schritte 3-6 sind für jeden Kanal mindestens erforderlich.

6. Interruptbehandlung in der U 857 D

Der CTC ist für den leistungsfähigsten Interrupt-Antwort-Modus des U 880 D, dem Interrupt-Modus 2 (IM2), konzipiert. Der Zweck eines Interrupts (Unterbrechung) besteht darin, daß periphere Kanäle oder Ports im System U 880 die CPU zwingen können, das laufende Programm zu unterbrechen, gezielt Unterprogramme zur Behandlung des Interrupts (Interrupt-Service-Routinen, ISR) aufzurufen und Rücksprünge aus den ISR in die unterbrochenen Programme durchzuführen. Die dazu notwendigen CPU-Operationszyklen, d. h. Interrupt-Akzeptanz-Zyklen (INTA), ISR-Aufruf-Zyklen und die ISR-Abschluß-Zyklen (RETI-Zyklen) werden in diesem Abschnitt detailliert beschrieben.

Die Interruptbehandlung im System U 880 D ist in allen peripheren Schaltkreisen (U 855, U 856, U 857) annähernd identisch, nur Interrupt-Auslösekriterien sowie die Anzahl der interruptfähigen Baugruppen (Kanäle oder Ports) sind unterschiedlich. Auslösekriterium eines CTC-Kanal-Interrupts ist ein Nulldurchgang seines Rückwärtszählers. Voraussetzung eines Interrupts ist die Interruptfähigkeit des Kanals, programmierbar durch Setzen des Interruptfreigabebits. Die Nulldurchgangsbedingung aktiviert die $\overline{\text{INT}}$ -Leitung bei Priorität des Kanals, d. h. entweder sofort, wenn kein weiterer höherpriorisiertes Interrupt behandelt wird, oder nach Abschluß höherpriorisierter Interrupts. Eine interne Interruptauslösung wird bis zur Interrupt-Antwort (INTA) der CPU aufgespeichert. Pro Kanal kann nur ein Interrupt gespeichert bleiben. Gelöscht werden kann ein gespeicherter Interrupt durch Rücksetzen des Interruptfreigabebits. Voraussetzung einer Interrupt-Antwort des U 880 D ist dessen gesetztes Interruptfreigabeflipflop IFF1, programmierbar durch den Befehl "Enable Interrupt" (EI). Innerhalb der peripheren Schaltkreise im System U 880 D sowie zwischen ihnen ist die Möglichkeit vorhanden, mehrere gleichzeitig auftretende Interrupts in einer hardwaremäßig festlegbaren Reihenfolge behandeln zu lassen. Dazu besitzen die peripheren Schaltkreise sowie ihre Kanäle oder Ports eine interruptprioritätsbestimmende Logikkettenstruktur (Daisy Chain Priority Interrupt), kurz als Prioritätskette bezeichnet. Innerhalb dieser Kette wird jeder Interrupt in seiner Annahme, seiner Bearbeitung und seinem Abschluß eindeutig der auslösenden Schaltung (Kanal oder Port) zugeordnet. Die Prioritätsreihenfolge innerhalb der Schaltkreise ist nicht veränderbar. So ist sie z. B. im CTC durch die Kanalnummern definiert; Kanal 0 hat höchste, Kanal 3 niedrigste Priorität. Die Priorität der Schaltkreise untereinander ist abhängig von der Stellung in der Kette, d. h. von der hardwaremäßigen Beschaltung ihrer prioritätsbestimmenden Interruptsteuerleitungen IEI und IEO. Eine Prioritätskette ist nach beiden Richtungen offen, d. h. sie kann erweitert werden. Nur die höchstpriorisierte Interruptfreigabe-Leitung IEI muß aktiv sein, oder kann zur Interruptsperrung der gesamten Kette genutzt werden. Am Ende jeder Instruktion fragt die CPU die INT-Leitung ab. Bei Aktivität wird ein Interrupt-Akzeptanz-Zyklus (INTA) eingeschoben. Dabei erwartet die CPU vom momentan höchstpriorisierten interruptanmeldenden Kanal oder Port ein 8 Bit-Wort, den kanal- oder portspezifischen Interruptvektor. Mit Hilfe dieses Vektors erfolgt dann der gezielte Unterprogrammaufruf der ISR, die auf jedem beliebigen Speicherplatz beginnen kann. Die Startadresse der ISR muß in einer Speichertabelle abgelegt sein. Um die Startadresse zu lesen (2 Byte) bildet die CPU aus dem Inhalt ihres I-Registers (High-Byte) und aus dem von der Peripherie gelieferten Interruptvektor einen 16 Bit-Zeiger, der den ersten von zwei Speicherplätzen adressiert, auf dem die untere Hälfte und auf dem zweiten die obere Hälfte der Startadresse der ISR steht. Durch zwei Speicherlese-Zyklen liest die CPU diese zwei Speicherplätze (Low Byte vor High Byte), bildet aus den Daten die Startadresse der ISR und führt einen Unterprogrammaufruf zu dieser Startadresse durch. Zum Abschluß der ISR muß die CPU den Befehl RETI (Rücksprung vom Interrupt) ausführen. Dieser Befehl wird durch die peripheren Schaltkreise im System U 880 D ebenfalls dekodiert und dient zur gezielten Rücksetzung von Interruptzuständen. Die Prioritätskette wird freigegeben. Innerhalb der Prioritätskette wird der RETI-Befehl eindeutig dem bedienten Kanal oder Port zugeordnet. Voraussetzung für weitere oder verschachtelte Interrupts ist die Neusetzung des Interruptfreigabe-Flipflops der CPU (EI), welches jeweils im INTA zurückgesetzt wird. Bei verschachtelten Interrupts muß es noch vor Abschluß der ISR gesetzt werden.



Prioritätsreihenfolge im CTC: Kanal 0
 Kanal 1
 Kanal 2
 Kanal 3

Bild 15: Prinzipielle Beschaltung des CTC bei Interruptbetrieb

Dies erfolgt durch den Befehl "Enable Interrupt" (EI), der bei linearer Interruptfolge meist vor dem RETI-Befehl steht und bei verschachtelten Interrupts zu Beginn der ISR, d. h. höherwertige Interrupts können dann niederwertigere ISR unterbrechen und nach RETI weiterführen. Die Verschachtelung kann mehrere Ebenen erfassen, die Bearbeitung erfolgt nach dem Prinzip "Zuletzt begonnen - zuerst abschließen", also nach dem Stack-Prinzip.

6.1. Der Interrupt-Akzeptanz-Zyklus

Die Bezeichnungen Interrupt-Bestätigungs-Zyklus, Interrupt-Quittungs-Zyklus, Interrupt-Acknowledge-Cycle, oder kurz INTA sind ebenfalls gebräuchlich. Am Ende des letzten Maschinenzyklusses eines Befehls mit der ansteigenden Flanke des letzten Taktzyklusses wird die Sammelleitung \overline{INT} durch die CPU kontrolliert. Liegt eine Interrupt-Anmeldung oder -Anforderung an, vollendet die CPU den Befehl und beginnt danach mit dem INTA-Zyklus. Der INTA wird durch den CTC, wie auch von den anderen E/A-Schaltkreisen des U 880 D-Systems durch gleichzeitige aktive Pegel auf den Steuerleitungen \overline{IORQ} und $\overline{M1}$ erkannt. Zuerst wird $\overline{M1}$ aktiv, wodurch das Auftreten weiterer Interrupts im System U 880 D verhindert wird. Für die Dauer von $\overline{M1}$ -Low wird das Auftreten (H-L-Flanke an der \overline{INT} -Leitung) von Interrupts in den peripheren Schaltkreisen intern verhindert.

Die Zeit zwischen der fallenden Flanke von $\overline{M1}$ und der fallenden Flanke von \overline{IORQ} ist für die Prioritätsklärung unter den interruptanmeldenden Kanälen oder Ports reserviert. Dazu werden noch zwei WAIT-Zyklen automatisch durch die CPU eingeschoben, wobei zusätzliche WAIT's erlaubt sind. Wenn $\overline{M1}$ und \overline{IORQ} beide aktiv sind, ist dies ein Zeichen für den Interruptanmelder mit der momentan höchsten Interruptpriorität (IEI=High, IEO=Low), daß sein Interrupt von der CPU anerkannt wurde. Der anerkannte Interruptanmelder (Kanal oder Port) gibt dabei einen Interruptvektor auf den Datenbus, der zur indirekten Adressierung der ISR dient. Jeder CTC-Kanal sendet seinen spezifischen Interruptvektor aus. Bit D0 der Interruptvektoren aller peripheren Schaltkreise im System U 880 ist immer "0", woraus sich für die Startadressenliste jeweils eine "gerade" Adresse ergibt. Beim CTC werden im INTA die Bits D2 und D1 durch die binäre 2 Bit-Adresse des angenommenen Interrupt-Anmelde-Kanals belegt. Bild 17 zeigt das Format der Interruptvektoren der 4 Kanäle.

für Kanal 0	D7	D6	D5	D4	D3	D2	D1	D0
	V7	V6	V5	V4	V3	0	0	0
für Kanal 1	D7	D6	D5	D4	D3	D2	D1	D0
	V7	V6	V5	V4	V3	0	1	0
für Kanal 2	D7	D6	D5	D4	D3	D2	D1	D0
	V7	V6	V5	V4	V3	1	0	0
für Kanal 3	D7	D6	D5	D4	D3	D2	D1	D0
	V7	V6	V5	V4	V3	1	1	0

Bild 16: Format der Kanal-Interruptvektoren.

V7-V3 werden dem Interruptvektor-Register entnommen, das in der Initialisierungsphase als Steuerwort Interrupt-Vektor geladen wurde (siehe Abschnitt 2.2). Durch das Format der Kanal-Interruptvektoren liegen die Startadresse der ISR der Kanäle in der Speichertabelle (Interrupttabelle) hintereinander.

6.2. Start der Interrupt-Service-Routine durch den U 880 D

Nach Erhalt des Interruptvektors führt die CPU automatisch eine Befehlsfolge zum indirekten Programmaufruf der Interrupt-Service-Routine durch. Der aktuelle PC wird gekellert (2 Speicher-Schreib-Zyklen), die Startadresse der ISR aus dem Speicher gelesen (2 Speicher-Lese-Zyklen) und die Befehle der ISR bearbeitet, wobei zu Beginn des 1. Befehls (OP-Code Fetch) die Adressleitungen der CPU mit der aus dem Speicher gelesenen Startadresse belegt sind. Die Adresse zum Lesen der Startadresse wird aus dem I-Register der CPU (obere Hälfte) und aus dem von der Peripherie gelieferten Interruptvektor (untere Hälfte) gebildet. Mit dieser und der folgenden Adresse werden Low-Bytes und High-Bytes der ISR-Startadresse gelesen (Bild 17). Dadurch kann die ISR auf jedem beliebigen Speicherplatz beginnen.

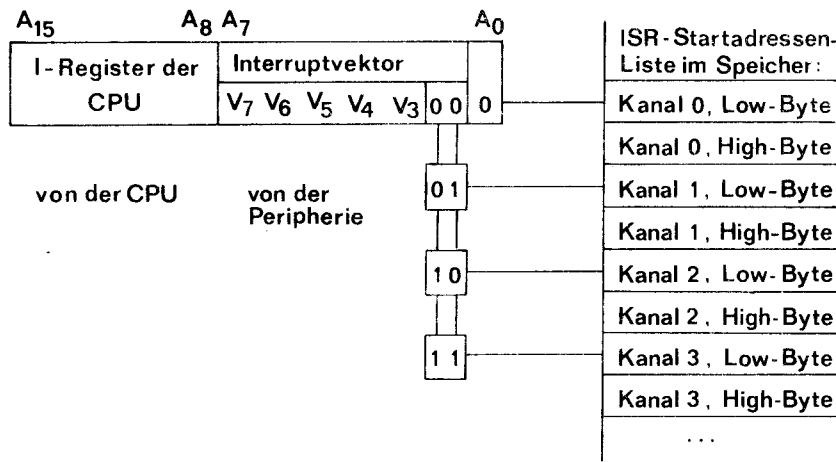


Bild 17: Unterprogrammaufruf der ISR

6.3. Rückkehr von Interrupt

Am Schluß jeder ISR muß der Befehl "Return from Interrupt" (RETI) stehen. Diesen Befehl dekodieren die peripheren Schaltkreise, so auch der CTC. Im Kanal, dessen ISR gerade beendet wird, wird durch RETI der interne Interruptzustand zurückgesetzt und damit intern der Interrupt wieder freigegeben. Nach außen wird die Prioritätskette wieder freigegeben, d. h. niederpriorisierte Kanäle können Interrupts anfordern (IEO geht auf High). Der Kanal, dem der RETI-Befehl gilt, fühlt sich deshalb angesprochen, da er als einziger in der Prioritätskette an seinem IEI-Anschluß High- und auf seinem IEO-Anschluß Low-Pegel führt. Alle anderen Kanäle oder Ports haben an beiden Anschlüssen entweder High-Pegel, wenn sie prioritätsmäßig höher liegen, oder Low-Pegel, wenn sie niedrigere Priorität haben. Damit der Kanal mit laufender ISR auch dann erkannt wird, wenn von höherpriorisierten Kanälen oder Ports Interrupts angemeldet, aber nicht anerkannt sind, senden diese nach der Dekodierung des ersten Befehls-Bytes von RETI (EdH) vorübergehend IEO=High. Dieses High-Signal muß den Kanal mit der laufenden ISR noch mit dem Befehlslesezklus des 2. RETI-Bytes (4dH) erreichen, damit dieser den RETI-Befehl seinem Interrupt zuordnet. Erst nach der Dekodierung des 2. RETI-Bytes (4dH) wird der Interrupt-Zustand beendet und die Prioritätskette durchgeschaltet (IEI=IEO). Der höherpriorisierte Interrupt-Anmelder sperrt dabei wieder die nachfolgende Kette durch IEO=Low.

6.4. Die Interrupt-Prioritätskette

Jeder periphere Schaltkreis im System U 880 sowie intern jeder Kanal oder jedes Port, besitzt zwei prioritätsbestimmende Kaskadierungsleitungen:

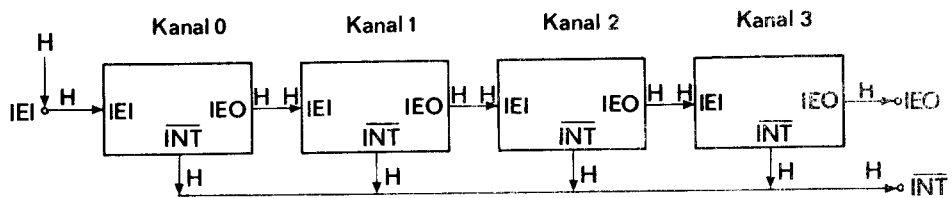
IEI und IEO (siehe Abschnitt 3.). Durch die Kaskadierung der Schaltkreise (Kanäle oder Ports) mittels dieser Interrupt-Steuerleitungen entsteht eine Interrupt-Prioritätsbestimmende Logikkettenstruktur, kurz Prioritätskette, wobei die Priorität durch den Platz in dieser Kette festgelegt wird oder ist (Daisy Chain Priority Interrupt).

Jeweils das niederpriorisierte Kettenglied (Schaltkreis oder intern Port oder Kanal) ist durch seine Leitung IEI mit der Leitung IEO des höherpriorisierten Kettengliedes verbunden und umgekehrt. Die logischen Pegel auf diesen Leitungen sind für jedes Kettenglied ausschlaggebend für Interrupt-Aktivitäten in allen Interrupt-Zyklen, d. h. der Zustand des Kettengliedes wird auf diesen Leitungen repräsentiert.

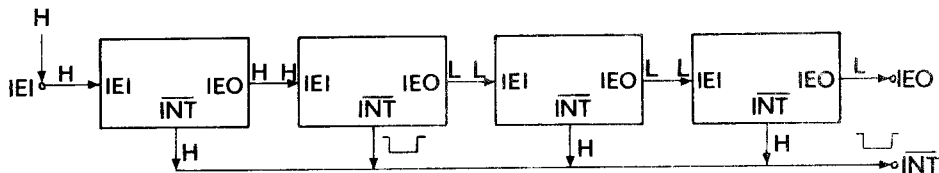
Für jedes Kettenglied gilt:

- IEI=Low verursacht unbedingt IEO=Low, d. h. ein höherpriorisiertes Kettenglied hat Interrupt angemeldet oder wird bedient.
- Interruptanforderung ist nur bei IEI=High möglich
- Bei Interruptanmeldung wird IEO=Low.
- Nach Interrupt-Annahme bleibt IEO=Low bis zur RETI-Dekodierung.
- Ein Interruptvektor wird nur bei IEI=High und IEO=Low ausgegeben.
- RETI wird nur bei IEI=High und IEO=Low akzeptiert.
- IEO geht kurzzeitig von Low auf High, wenn das 1. Byte von RETI (EdH) dekodiert ist und IEI=High gilt (nur bei angeforderten und noch nicht angenommenen Interrupts zum ISR-Abschluß niederwertigerer Glieder).
- IEI=IEO gilt, wenn unmittelbar kein Interrupt angemeldet oder bedient wird.

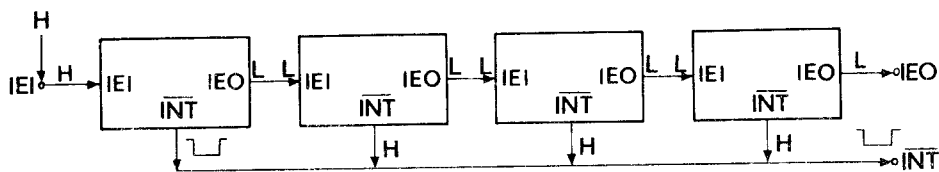
Die 4 Kanäle des CTC bilden 4 festverdrahtete Glieder einer Prioritätskette, wobei IEI von Kanal 0 und IEO von Kanal 3 herangeführt sind. Der CTC kann also nur als Einheit von 4 Gliedern in eine Prioritätskette einbezogen werden. Bild 18 zeigt ein Beispiel der Prioritätskaskadierung anhand der 4 CTC-Kanäle. Die Prioritätsreihenfolge unter den Kanälen ist durch die Kanalnummern definiert, Kanal 0 hat höchste, Kanal 3 niedrigste Priorität.



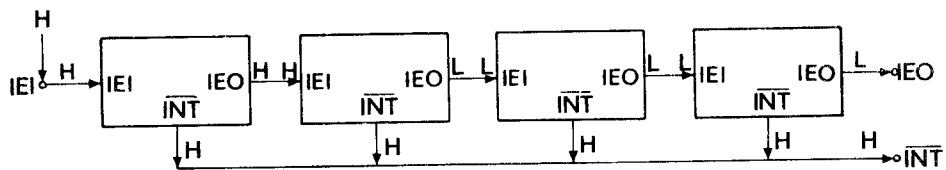
Kein Kanal meldet Interrupt an oder wird bedient.



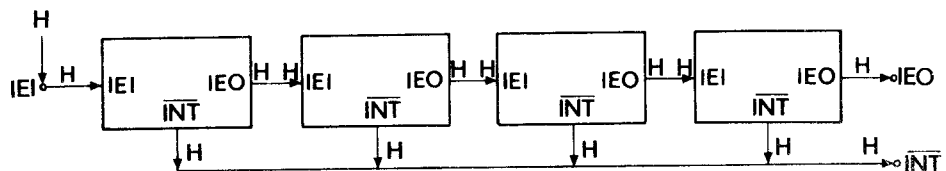
Kanal 1 sendet eine Interruptanforderung aus, sie wird bestätigt.
Die Prioritätskette wird durchgeschaltet.



Kanal 0 fordert einen Interrupt an, die Bedienung von Kanal 1 wird unterbrochen, Kanal 0 wird bedient (INTA).



Kanal 0 hat RETI dekodiert und gibt die Kette wieder frei.
Die Bedienung von Kanal 1 wird fortgeführt.



Kanal 1 hat RETI dekodiert und gibt die Kette wieder frei.
Kein Kanal wird mehr bedient.

Bild 18: Beispiel der Prioritätskaskadierung durch vier CTC-Kanäle

7. Zeitprobleme in der Interrupt-Behandlung

Ein E/A-Schaltkreis reagiert auf den INTA mit der Aussendung des Interruptvektors. Nur der Schaltkreis, der an IEI High-Pegel empfängt und auf IEO Low-Pegel führt, gibt einen Interruptvektor aus und geht in den Interrupt-Zustand ("Interrupt in Behandlung" - Zustand). In einer Prioritätskette darf im INTA mit der H/L-Flanke von \overline{IORQ} , nur ein Schaltkreis die o. g. Pegel an den Leitungen IEO/IEI führen. Ebenso müssen zur gezielten RETI-Dekodierung diese Pegel anliegen (IEI=High, IEO=Low). Dadurch, daß jeder Schaltkreis die IEI/IEO-Durchschaltung verzögert ($t_{DL(IO)}$, $t_{DH(IO)}$) kann es in bestimmten Anwendungsfällen zu Zeitproblemen kommen, wobei die Anzahl der möglichen Kettenglieder beschränkt wird.

- Die Stabilisierungszeit, die der Kette zur Verfügung steht, ist im Extremfall die Zeit zwischen H/L-Flanke von $\overline{M1}$ und H/L-Flanke von \overline{IORQ} im INTA:

$$t_{m1} = 2t_c + t_{w(CH)} + t_f - 80ns$$

Dieser Extremfall tritt auf, wenn ein Schaltkreis am Ende der Kette den INTA auslöste (relativ niedrigere Priorität) und wenn kurz vor der H/L-Flanke von $\overline{M1}$ ein Schaltkreis am Anfang der Kette (relativ höhere Priorität) einen Interrupt anfordert. Der Low-Pegel am höher priorisierten Schaltkreis muß bis zur H/L-Flanke von \overline{IORQ} den hinteren IEI-Eingang erreichen. Wenn dies nicht geschehen würde, können in zwei Schaltkreisen Interrupt-Zustände auftreten, und zwei Interruptvektoren würden gegeneinander "kämpfen"!

In diesem Fall gilt für ein ordnungsgemäßes Funktionieren einer Kette ohne zusätzliche Logik die folgende Bedingung:

$$t_{m1} > t_{DM(IO)} + (N-2) \cdot t_{DL(IO)} + t_s(IEI)$$

t_{m1} :	Zeit die $\overline{M1}$ vor \overline{IORQ} im INTA stabil ist
$t_{DM(IO)}$:	IEO-Verzögerungszeit von der H/L-Flanke von $\overline{M1}$
$t_{DL(IO)}$:	IEO-Verzögerungszeit zur H/L-Flanke von IEI
$t_s(IEI)$:	Setzzeit vor der H/L-Flanke von \overline{IORQ} im INTA

Diese Bedingung wird nur für $N \leq 4$ erfüllt (bei maximaler Systemtaktfrequenz). Maßnahmen zur Lösung dieses Problems sind im Abschnitt 7.1. dargelegt.

- Ein ähnliches Problem kann bei der gezielten RETI-Dekodierung auftreten. Bausteine, die einen Interrupt angemeldet haben, jedoch noch nicht bestätigt wurden, senden nach der Dekodierung von ED/Hex (erstes RETI-Byte) vorübergehend IEO=H. Dieses High muß einen möglichen, in der Interrupt-Bearbeitung befindlichen, niederpriorisierten Baustein noch vor der 4D/Hex (zweites RETI-Byte) - Dekodierung erreichen, damit dieser IEI=High und IEO=Low führt und den RETI-Befehl seiner ISR zuordnet. Die Zeit des Durchlaufs des L/H-Übergangs von IEI nach IEO beschränkt wiederum die Anzahl der möglichen Kettenglieder. Die Kette hat 4 Taktperioden Zeit, um sich zu stabilisieren.

Daraus ergibt sich folgende Bedingung:

$$4t_c \geq t_{DR(IO)} + (N-2) t_{DH(IO)} + t_{SH(IEI)}$$

$t_{DR(IO)}$:	IEO-Verzögerungszeit bei ED-Dekodierung
$t_{DH(IO)}$:	IEO-Verzögerungszeit nach L/H-Flanke an IEI
$t_{SH(IEI)}$:	Setzzeit für IEI vor L/H-Flanke von C

Die Zeiten $t_{DR(IO)}$ und $t_{SH(IEI)}$ sind in der TGL nicht definiert. Im Extremfall, bei maximal möglichen Werten für diese Zeiten wird diese Bedingung für maximal $N \leq 6$ erfüllt, d. h. 6 Schaltkreise in der Kette erfüllen bei maximaler Systemtaktfrequenz diese Bedingung. Wird diese Bedingung nicht eingehalten, kann das Problem auftreten, daß ein Bauelement aus dem "Interrupt in Behandlung" - Zustand nicht herauskommt und sich und die folgende Kette für weitere Interrupts sperrt.

7.1. Maßnahmen zur Entschärfung der Zeitprobleme

Soll die Anzahl der Schaltkreise über die erwähnten Grenzen erhöht werden, müssen zusätzliche Schaltungsmaßnahmen getroffen oder Zeitbeschränkungen im gesamten System in Kauf genommen werden. Die Probleme der H/L-Durchschaltung im INTA sind durch verschieden schnelle Zusatzlogiken lösbar. Ein einfacher Schaltungsvorschlag (Bild 19) ermöglicht die Erweiterung der Prioritätskette auf ca. 25 E/A-Bausteine (mit TTL-Logik).

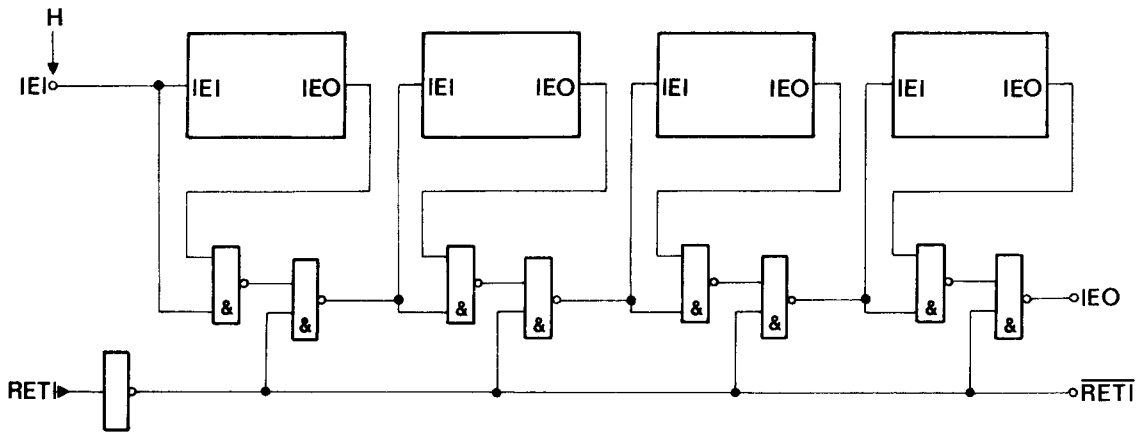


Bild 19: Schnelle Prioritätslogik

Bild 20 zeigt einen universellen Lösungsvorschlag, der die Anzahl der möglichen Schaltkreise in der Kette auf den vollen direkt adressierbaren E/A-Adressbereich erhöht.

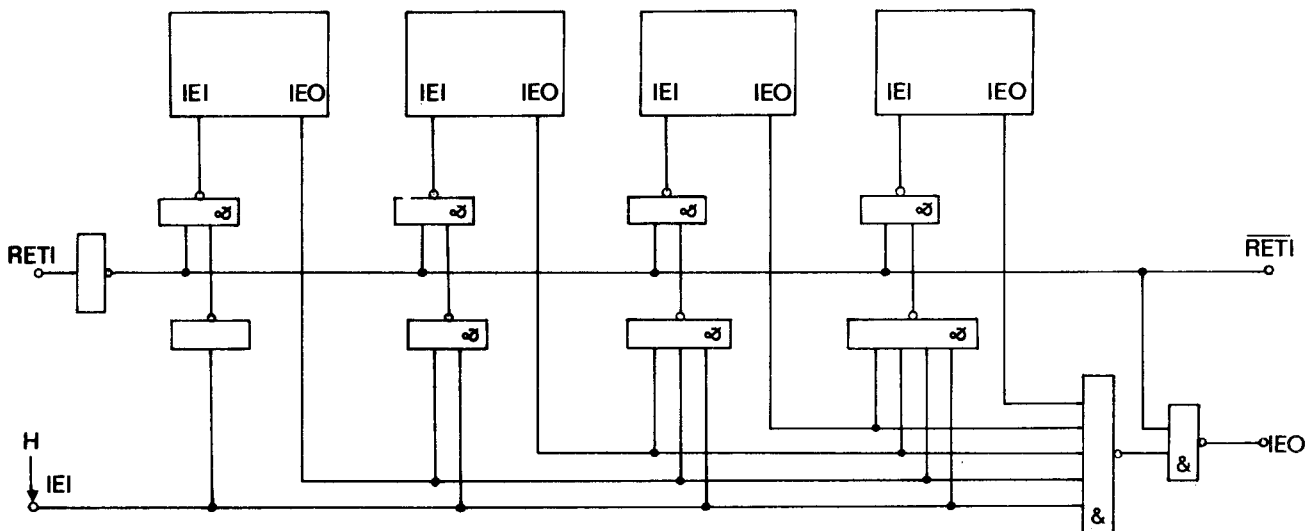


Bild 20: Universelle schnelle Prioritätslogik

Beide Schaltungen benötigen ein sogenanntes RETI-Signal, das durch Dekodierung des ersten RETI-Bytes Ed/Hex ermittelt wird (Bild 21). Diese Dekodierung ist etwas aufwendig, da Ed/Hex als 1. Byte eines Befehls erkannt werden muß.

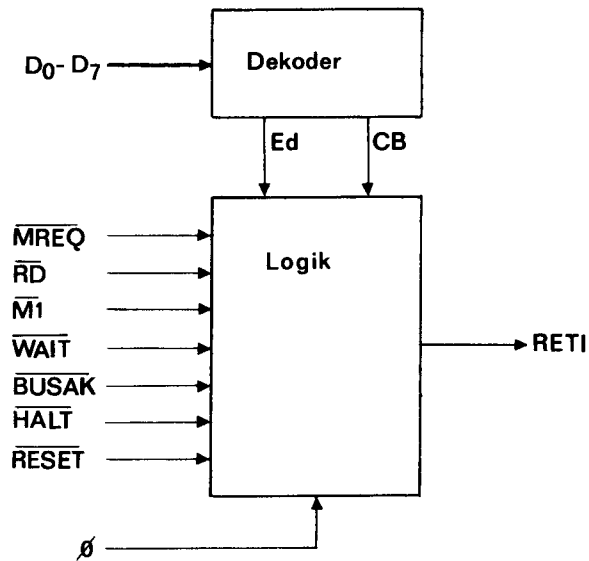


Bild 21: Prinzipielle RETI-Dekodierung

Es gibt noch eine Reihe anderer Lösungen, die jedoch Softwarebeschränkungen einschließen. Der Abstand zwischen H/L- $\overline{M1}$ und H/L- \overline{IORQ} im INTA ist nicht durch WAIT's verlängerbar. Eine Beschleunigung der H/L-Durchschaltung verlangt generell zusätzliche Logik. Das RETI-Problem könnte durch WAIT's gelöst werden (während der OP-Code-Fetchs von Ed/Hex) jedoch müßte Ed/Hex ebenfalls extern dekodiert werden. Es ist also für Systeme mit mehr als 4 bzw. 6 E/A-Schaltkreisen generell eine der oben aufgezeigten Lösungen äquivalente Schaltung erforderlich.

8. Zeitverhalten der U 857 D

In diesem Abschnitt werden die zeitlichen Beziehungen der entsprechenden CTC-Leitungen untereinander und zum Systemtakt für folgende Operationsarten dargestellt:

CTC-Schreib-Zyklus:	Eingabe eines Datenwortes in den CTC
CTC-Lese-Zyklus:	Ausgabe eines Datenwortes durch den CTC
CTC-Zähler-Zyklus:	Zählung des externen Taktes an C/TRG
CTC-Zeitgeber-Zyklus:	Triggerung eines Zeitgeber-Kanals
Interrupt-Bestätigungs-Zyklus:	Ausgabe des Interruptvektors vom CTC
Interrupt-Abschluß-Zyklus:	RETI-Dekodierung und Freigabe der Prioritätskette

8.1. CTC-Schreib-Zyklus

Bild 22 zeigt das Zeitverhalten bei einem Schreibzyklus. Dieser Zyklus ist beim Laden eines Kanalsteuerwortes, eines Interruptvektors und einer Zeitkonstante anzuwenden. Die CPU U 880 D erzeugt Zeitabläufe bei Ausgabe-Befehlen auf den entsprechenden Steuerleitungen, die den geforderten Bedingungen der U 857 D genügen.

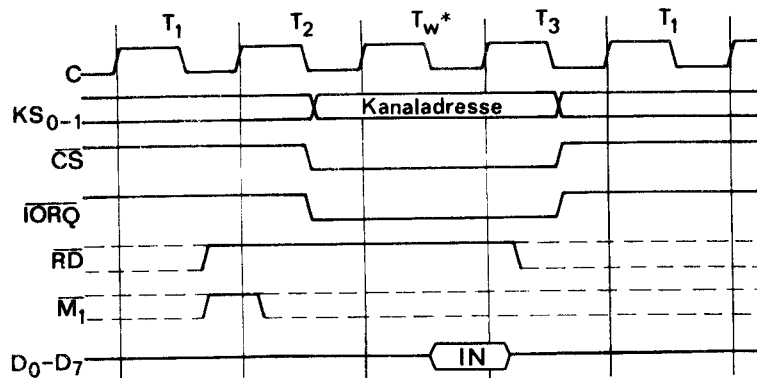


Bild 22: CTC-Schreib-Zyklus

Die Steuerleitung \overline{WR} der CPU ist beim CTC nicht vorhanden und wird durch die logische Verknüpfung $\overline{WR}^* = CS \cdot \overline{IORQ} \cdot RD \cdot \overline{M1}$ intern erzeugt.

$\overline{M1}$ darf nicht aktiv sein, um den Schreib-Zyklus vom Interruptbestätigungszyklus (INTA) zu unterscheiden. Während T2 muß die 2 bit-Binäradresse des zu beschreibenden Kanals (KS1, KS0) sowie die Daten auf dem Datenbus bereitgestellt werden. Die CPU fügt aus Stabilisierungsgründen automatisch einen WAIT-Zyklus T_w^* ein. Weitere WAIT-Zyklen sind nicht erlaubt. Die Daten werden durch den CTC mit der steigenden Systemtaktflanke T3 in das entsprechende Register übernommen.

8.2. CTC-Lese-Zyklus

Bild 23 zeigt das Zeitverhalten bei einem Lesezyklus. In beiden Betriebsarten (Zähler/Zeitgeber) kann der Momentanwert der Rückwärtszähler zu jedem beliebigen Zeitpunkt ausgelesen, d. h. auf den Datenbus ausgegeben werden (die CPU liest: CTC-Lese-Zyklus) (Bild 23). Während T2 beginnt die CPU im Lesezyklus mit der Aktivierung von \overline{RD} , \overline{IORQ} und \overline{CS} . $\overline{M1}$ darf nicht aktiv sein, im Unterschied zum INTA-Zyklus. Zur Auswahl des Kanals muß die binäre Kanaladresse (KS1/KS0) bereitgestellt sein. Die CPU fügt einen WAIT-Zyklus automatisch ein, weitere sind nicht erlaubt. Der entsprechende Zählerstand des Kanals, d. h. der vor der steigenden Systemtaktflanke von T2 aktuelle Wert wird vor der steigenden Flanke von T3 auf den Datenbus ausgegeben. Der aktuelle Zählerstand ist die Binärzahl (8 Bit) der notwendigen Zählschritte des Rückwärtszählers bis zum Nulldurchgang.

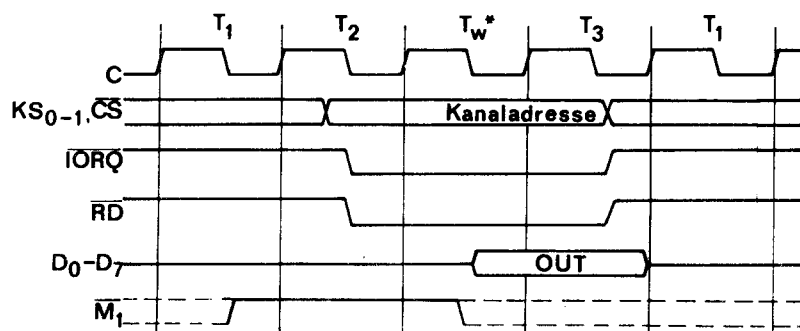


Bild 23: CTC-Lese-Zyklus

8.3. Zeitverhalten eines Zähler-Kanals

Eine aktive Flanke (ist programmierbar, im Bild 24 wurde eine positive Flanke gewählt) am Eingang C/TRG veranlaßt ein Dekrementieren des Rückwärtszählers; synchron zur steigenden Systemtaktflanke. Die aktive Flanke muß eine Setzzeit $t_{SH(CK)}$ vor der steigenden Systemtaktflanke anliegen, um garantiert mit dieser Flanke übernommen zu werden. Für das Taktsignal am Eingang C/TRG werden minimale Pulsbreiten gefordert. Wenn der Rückwärtszähler vom Zählerstand "1" herunterzählt, wird am Ausgang ZC/TO nach einer Verzögerungszeit $t_{DH(CK)}$ nach der steigenden Systemtaktflanke ein High-Impuls ausgegeben.

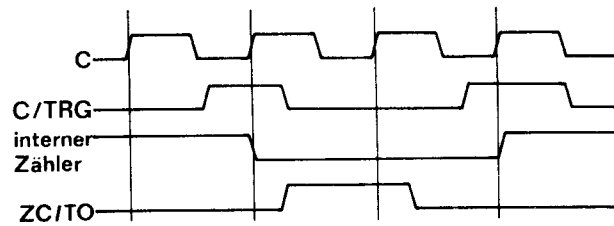


Bild 24: Zeitverhalten eines Zähler-Kanals

8.4. Zeitverhalten eines Zeitgeber-Kanals

Bild 25 zeigt das Zeitverhalten beim Start eines Zeitgeber-Kanals. Der Triggereingang ist schon durch Laden der Zeitkonstanten aktiviert worden, eine aktive Flanke (ist programmierbar, im Bild 26 wurde eine positive Flanke gewählt) am Eingang C/TRG startet die Zeitgeberoperation (Dekrementieren des Vorteilers) mit der zweiten (Setzzeit $t_{S(TR)}$ wird eingehalten) oder dritten erfolgten steigenden Systemtaktflanke. Wie bei Zähler-Kanälen kann der Triggerimpuls asynchron zum Systemtakt empfangen werden und muß eine minimale Pulsbreite haben. Zwischen auslösender Flanke am Eingang C/TRG und steigender Systemtaktflanke wird eine minimale Setzzeit $t_{S(CK)}$ gefordert. Dadurch ist eine Verzögerung des Zeitgeberstarts um eine Taktperiode möglich.

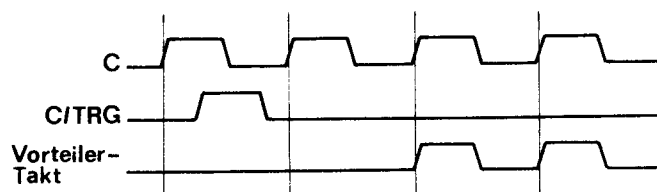


Bild 25: Zeitgeberstart durch C/TRG-Signal

Wenn der Rückwärtszähler vom Zählerstand "1" herunterzählt, wird am Ausgang ZC/TO nach einer Verzögerungszeit $t_{DH(ZC)}$ nach der steigenden Systemtaktflanke ein High-Impuls erzeugt, dessen Impulsbreite zwischen 200 ms und 600 ms liegt.

Bei der Initialisierung erfolgt die Aktivierung des C/TRG-Einganges durch Laden der Zeitkonstanten. Mit der steigenden Systemtaktflanke T2 des CPU-Maschinenzyklusses, der auf das

Laden der Zeitkonstante folgt, wird der Triggereingang aktiviert, d. h. eine aktive Flanke wird dann durch den Zeitgeberkanal anerkannt. Ohne Verwendung des Triggereingangs wird mit der zweiten steigenden Systemtaktflanke nach Laden der Zeitkonstante, bzw. mit steigender Flanke von T2 die Zeitgeberoperation gestartet. Setzzeiten sind dabei uninteressant, da der Zeitgeber den Systemtakt als Basis verwendet.

8.5. Zeitabläufe im Interrupt-Akzeptanzzyklus

Bild 26 zeigt die Zeitabläufe der CPU im INTA. (Siehe auch Technische Beschreibung U 880 D). Kritisch ist die Zeit zwischen der fallenden Flanke von $\overline{M1}$ und der fallenden Flanke von \overline{IORQ} , in der die Peripherie die Prioritätsklärung abschließen muß. Durch die Verzögerungszeiten beim Durchschalten der Prioritätskette wird die Anzahl der möglichen Kettenglieder praktisch beschränkt. Weitere WAIT's sind erlaubt. Sind $\overline{M1}$ und \overline{IORQ} aktiv, wird vom anerkannten Interruptanmelder ein Interruptvektor zur individuellen Adressierung der ISR ausgegeben.

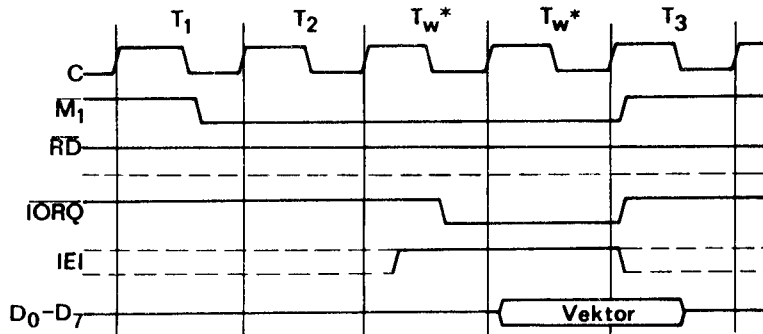


Bild 26: Interrupt-Akzeptanzzyklus der CPU

Bild 27 zeigt den gesamten Zeitverlauf bei Vektorinterrupt (IM2), von der Interruptanforderung bis zum ISR-Aufruf. Zusätzliche WAIT's sind nicht eingeführt.

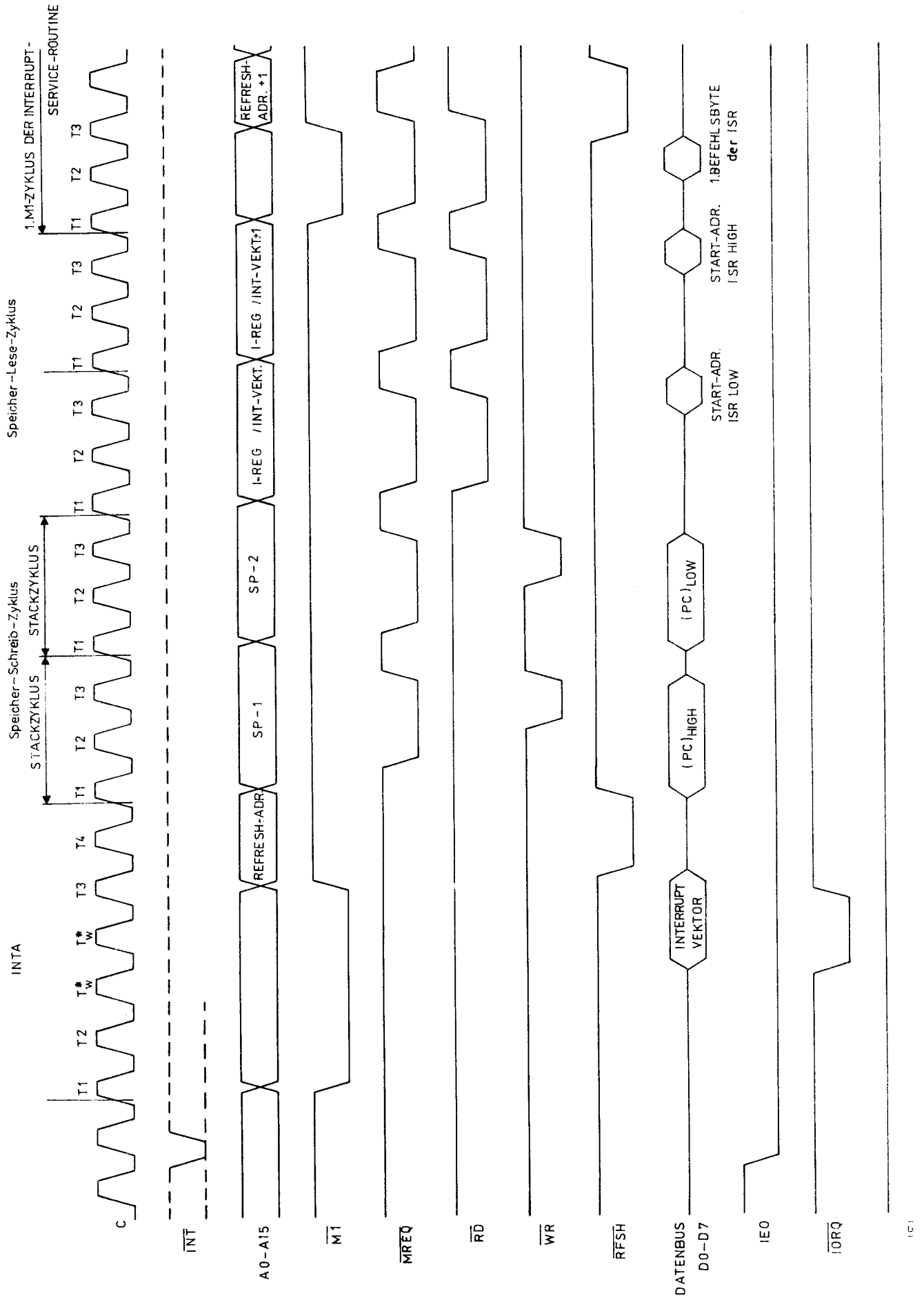


Bild 27: Zeitverläufe beim Aufruf einer ISR

8.6. Zeitverläufe bei der Rückkehr vom Interrupt

Der RETI-Befehl ist ein 2-Byte Befehl und wird durch die CPU mit 2 Befehlsholezyklen gelesen. Diese Befehlsholezyklen müssen durch den CTC überwacht werden, wobei die Daten auf dem Datenbus durch den CTC dekodiert werden. Bild 28 zeigt den Zeitverlauf der beiden RETI-Befehlsholezyklen, wobei die Daten ED-4D/Hex von der CPU aus dem Programmspeicher gelesen werden.

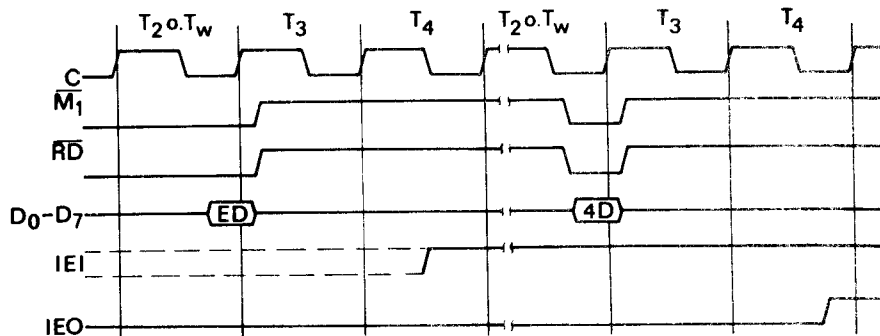


Bild 28: Rückkehr vom Interrupt

Für die CPU ist dieser RETI-Befehl identisch mit dem RET-Befehl (Return), sie führt einen Unterprogramm-Rücksprung aus, d. h. sie springt in das vom Interrupt unterbrochene Programm zurück. Steht der Befehl "Enable Interrupt" (EI) direkt vor RETI, so ist die CPU nach Abschluß des RETI-Befehls wieder fähig auf Interrupts zu antworten. Das Interruptfreigabe-Flipflop der CPU wird 2 Befehlsholezyklen nach EI wieder gesetzt.

9. Applikationsbeispiel

Anhand eines Anwendungsbeispiels soll die Programmierung und der Einsatz der U 857 D veranschaulicht werden.

Problem allgemein: An einer Arbeitsmaschine sollen Arbeitsschritte gezählt, angezeigt und bei bestimmten Schrittzahlen Funktionen ausgelöst werden.

Problem speziell: Ein Arbeitszyklus umfaßt 64 Schritte, jeder Schritt soll angezeigt werden. Bei jedem 12. Schritt soll ein akustisches Signal erfolgen, bei jedem 32. Schritt soll eine Meßwerterfassung und nach 64 Schritten soll eine Ausgabe (Drucker) der Meßwerte und ein Zyklus-Neustart erfolgen. Die Arbeitsschrittfrequenz liegt im 1 Hz-Bereich.

Problemlösung: Ziel ist der Einsatz einer Minimalkonfiguration des Systems U 880 D inklusive CTC.

Der CTC wird für folgende Funktionen benutzt:

Kanal 0: Zeitgeber für Multiplexanzeige der Zählsschritte;
Software-Anzeige durch Interrupt
Frequenz: ca. 1 kHz.

Kanal 1: Zähler der Arbeitsschritte, Auswertung durch Zählerstandsabfrage (Polling) und Interrupt.

Kanal 2: Zeitgeber, Ausgabe des akustischen Signals in Abhängigkeit von Zählerstand 12 von Kanal 1.

Kanal 3: Zeitgeber zur Abschaltung von Kanal 2 durch Interruptfolge.

Programmierung von Kanal 0: Kanal 0 soll eine Interruptfolge zur Softwaresteuerung der Anzeige liefern (Multiplexanzeige). Die Frequenz soll etwa 1 kHz betragen. Bild 29 zeigt das Kanalsteuerwort für Kanal 0.

$$\begin{array}{cccccccc}
 D_7 & D_6 & D_5 & D_4 & D_3 & D_2 & D_1 & D_0 \\
 \hline
 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
 \hline
 \end{array} = A5/\text{Hex}$$

Bild 29: Kanalsteuerwort Kanal 0

Kanal 0 ist interruptfähig (D7), ist ein Zeitgeberkanal (D6), der Verteilerfaktor beträgt 256 (D5), er wird durch Laden der Zeitkonstanten getriggert (D3) und das nachfolgende Steuerwort wird als Zeitkonstante interpretiert (D2).

Die Zeitkonstante ermittelt sich aus der Formel:

$$T_c = \frac{1 \text{ ms}}{t_c \cdot p}$$

und beträgt ca. 62/Hex (Bild 30).

$$\begin{array}{cccccccc}
 ZK_7 & ZK_6 & ZK_5 & ZK_4 & ZK_3 & ZK_2 & ZK_1 & ZK_0 \\
 \hline
 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
 \hline
 \end{array} = 62/\text{Hex}$$

Bild 30: Zeitkonstante Kanal 0

Beide Steuerworte werden im Initialisierungsprogramm geladen und die durch die Interrupts aufgerufene Interrupt-Service-Routine bedient die Anzeige. Interruptvektor, Interrupttabelle, Laden des I-Registers der CPU sowie die ISR sind dazu noch erforderlich (siehe Programmierung von Kanal 1).

Programmierung von Kanal 1: Kanal 1 soll die Arbeitsschritte Zählen (Zählerkanal), die in über den entsprechenden Geber und die Schnittstelle zugeführt werden. Das jeweilige Zählergebnis soll abgefragt und ausgewertet werden. Nach jeweils 64 Zählschritten soll ein Interrupt zur Bedienung eines Druckers und zum Neustart erzeugt werden (Nulldurchgang). Bild 31 zeigt das Kanalsteuerwort, Bild 32 die Zeitkonstante.

$$\begin{array}{cccccccc}
 D_7 & D_6 & D_5 & D_4 & D_3 & D_2 & D_1 & D_0 \\
 \hline
 1 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
 \hline
 \end{array} = C5/\text{Hex}$$

Bild 31: Kanalsteuerwort Kanal 1

ZK ₇	ZK ₆	ZK ₅	ZK ₄	ZK ₃	ZK ₂	ZK ₁	ZK ₀	= 40/Hex = 64
0	1	0	0	0	0	0	0	

Bild 32: Zeitkonstante Kanal 1

Der Zählerkanal (D6) ist interruptfähig (D7), zählt negative Flanken (D4) und das Laden der Zeitkonstanten folgt (D2). Beide Steuerworte werden im Initialisierungsprogramm geladen. Der Programmstart muß synchron zum Arbeitsmaschinenstart erfolgen. Zählerstandsabfrage (Polling), ISR und Eintragung in die Interrupttabelle sind erforderlich. Das Anzeige-Interrupt von Kanal 0 hat höhere Priorität, kann also die Bedienung von Kanal 1 hinauszögern. Praktisch ist dies unwesentlich.

Programmierung Kanal 2 und 3: In Abhängigkeit von einem bestimmten Zählerstand von Kanal 1 sollen Kanal 2 und 3 gleichzeitig gestartet werden. Kanal 2 liefert eine kontinuierliche Impulsfolge (1 kHz) als akustisches Signal, Kanal 3 setzt Kanal 2 nach einigen Sekunden zurück (akustisches Signal aus). Kanal 2 ist nicht interruptfähig, ist er ein Zeitgeberkanal und liefert am Ausgang ZC/T02 eine kontinuierliche Impulsfolge. Bild 33 zeigt das Kanalsteuerwort von Kanal 2.

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	= 45/Hex
0	0	1	0	0	1	0	1	

Bild 33: Kanalsteuerwort Kanal 2

Die Zeitkonstante für ein 1 kHz-Signal entspricht der von Kanal 0: 62/Hex (Bild 31)

Kanal 3 liefert eine Interruptfolge mit maximaler Periode. Durch Software-Zählung der Interrupts wird der Zeitgeber erweitert.

Bild 34 zeigt das Kanalsteuerwort für Kanal 3.

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀	= A5/Hex?
1	0	1	0	0	1	0	1	

Bild 34: Kanalsteuerwort Kanal 3

Als Zeitkonstante wird 00/Hex verwendet, was 256 entspricht. Somit meldet Kanal 3 ca. alle 26 ms einen Interrupt an. Nach 50 Interrupts sollen Kanal 2 und Kanal 3 zurückgesetzt werden (siehe Programm). Die ISR und deren Startadressen müssen erstellt bzw. eingetragen werden, die Initialisierung der Kanäle 2 und 3 erfolgt im Startprogramm zur akustischen Signalzeugung.

Programm zum Applikationsbeispiel

Hauptprogramm: Es beginnt, wenn durch $\overline{\text{RESET}}$ gestartet auf Adresse 0000/Hex des Speicherbereiches:

```
START: LD SP,SPA;      Stack-Pointer laden, 2 Byte-RAM-Adresse, notwendig für Stackoperationen
        LD HL,INTAB;   Startadresse der Interrupttabelle (Kanal 0)
        LD A,H
        LD I,A;        I-Register der CPU mit High-Byte von INTAB laden
        LD A,L;        Interruptvektor für CTC entspricht
        OUT CTC0;      Low-Byte von INTAB, laden
```

Initialisierung und Start von Kanal 0: Zeitgeber Anzeige

```
LD A,0A5H;      Kanalsteuerwort Kanal 0: A5/Hex
OUT CTC0;       laden
LD A, 62 H;     Zeitkonstante Kanal 0: 62/Hex
OUT CTC0;       laden
```

Initialisierung und Start von Kanal 1: Zähler Arbeitsschritte

```
LD A,0C5H;      Kanalsteuerwort Kanal 1: C5/Hex
OUT CTC1;       laden
LD A, 40H;      Zeitkonstante Kanal 1: 40/Hex
OUT CTC1;       laden

IM2;           Interrupt-Mode 2: Vektor-Interrupt
EI;            Interruptfreigabe
```

Abfrageschleife (endlos) nach Zählerständen von Kanal 1:

ABFRK1:

```
IN CTC1;        Zählerstand Kanal 1 im Akku
CALL ANZVO;     Anzeige von A vorbereiten, Anzeigespeicher laden; ANZVO wird nicht weiter erläutert
CP 34H;         Zählerstand 12 aktuell (64 - 12 = 52 = 34/Hex)
CAZ AKSIG;      AKSIG wenn Zählerstand 12;
                AKSIG löst akustisches Signal aus (Verwendung von Kanal 2 und 3)
CP 20H;         Zählerstand 32 aktuell ?
                (64 - 32 = 32 = 20/Hex)

CAZ MESS;       MESS wenn Zählerstand 32;
                MESS erfaßt aktuelle Meßwerte (z. B. über A/D-Wandler) und bereitet sie zum Ausdrucken durch PRINT vor. MESS wird nicht weiter erläutert.

IR ABFRK1;     Abfrageschleife erneut; Endlosschleife
```

Nach Start und Initialisierung verbleibt das Programm in der Abfrageschleife, nur bestimmte Zählerstände oder Interrupts lösen andere Aktivitäten aus. Zur Interruptbehandlung muß eine Interrupttabelle erstellt werden, in der die Startadressen aller verwendeten Interrupt-Service-Routinen stehen.

```

INTAB:  ISROLow;      Startadresse der ISR von
        ISROHigh;     Kanal 0
        ISR1Low;      Startadresse der ISR von
        ISR1High;     Kanal 1
        00           nicht benötigt (Kanal 2)
        00
        ISR3Low;      Startadresse der ISR von
        ISR3High;     Kanal 3

```

Auf diesen Speicherplätzen müssen die Adressen der ISR's eingetragen werden (Low-Byte vor High-Byte).

Interrupt-Service-Routine zur Anzeigeansteuerung; ISRO besitzt höchste Priorität.

```

ISRO:   PUSH REG;      Register retten
        CALL ANZ;      UP zur Anzeigeansteuerung
        Anzeigewerte aus Anzeige-Speicher
        POP REG;       Register retten
        EI;            Interrupt freigeben
        RETI;          Rückkehr aus Anzeige-Interrupt

```

ANZ ist abhängig von der verwendeten Ansteuerlogik.

Interrupt-Service-Routine zur Druckeransteuerung. Nach 64 Zählschritten von Kanal 1 ausgelöst:

```

ISR1:   PUSH REG;      Register retten
        CALL PRINT;    UP zur Ausdrückung von Meßwerten; Meßwerte wurden durch UP: ME55
        vorbereitet.
        POP REG;
        EI;
        RETI;

```

Der Befehl EI kann auch zu Beginn der ISR stehen, um den Anzeige-Interrupt nicht zu sperren. PRINT muß dazu interrupttransparent sein. PRINT ist abhängig vom verwendeten Drucker und der dazugehörigen Ansteuerlogik.

Unterprogrammteil mit der ISR von Kanal 3 zur Auslösung eines akustischen Signals. Die Initialisierung von Kanal 2 und 3 erfolgt in diesem Programm.

```

AKSIG:  PUSH AF;
        LD A, 45 H;    Kanalsteuerwort Kanal 2: 45/Hex
        OUT CTC2;     laden
        LD A, 62 H;    Zeitkonstante Kanal 2: 62/Hex
        OUT CTC2;     laden

```

```

LD A,0A5H;      Kanalsteuerwort Kanal 3: '5/Hex
OUT CTC3;       laden
LD A, 00H;      Zeitkonstante Kanal 3: 00/Hex
OUT CTC3;       laden

LD A,00H;      Zurücksetzen des Interruptzählers
LD (INTZ),A;
POP AF;        Akku retten
RET;

```

Kanal 2 gibt ein 1 kHz-Signal aus, Kanal 3 erzeugt eine Interruptfolge.

Interrupt-Service-Routine von Kanal 3

```

ISR3:  PUSH AF      Akku retten
        LD A,(INTZ);
        INC A;      Zähler erhöhen (RAM-Zelle)
        LD (INTZ),A  In A steht Anzahl der Interrupts
        CP 50;      50 Interrupts?
        JRNZ ISR31;  nein: ISR31
        LD A, 03H;   Kanalsteuerwort "reset"
        OUT CTC2;    für Kanal 2 laden
        OUT CTC3;    für Kanal 3 laden
ISR31:  POP AF;      Akku retten
        EI;          Abschluß der ISR
        RETI

```

Nach 50 Interrupts von Kanal 3 wird Kanal 2 und 3 zurückgesetzt.

Mit diesem prinzipiellen Programm ist das Problem gelöst. Die Unterprogramme zur Meßwert-
erfassung, Meßwertausdruckung und Anzeigeansteuerung sind hier nicht im Detail beschrieben,
sie sind auch zum Verständnis des CTC's uninteressant.
Bild 35 zeigt den prinzipiellen Schaltungsaufbau zum Problem.

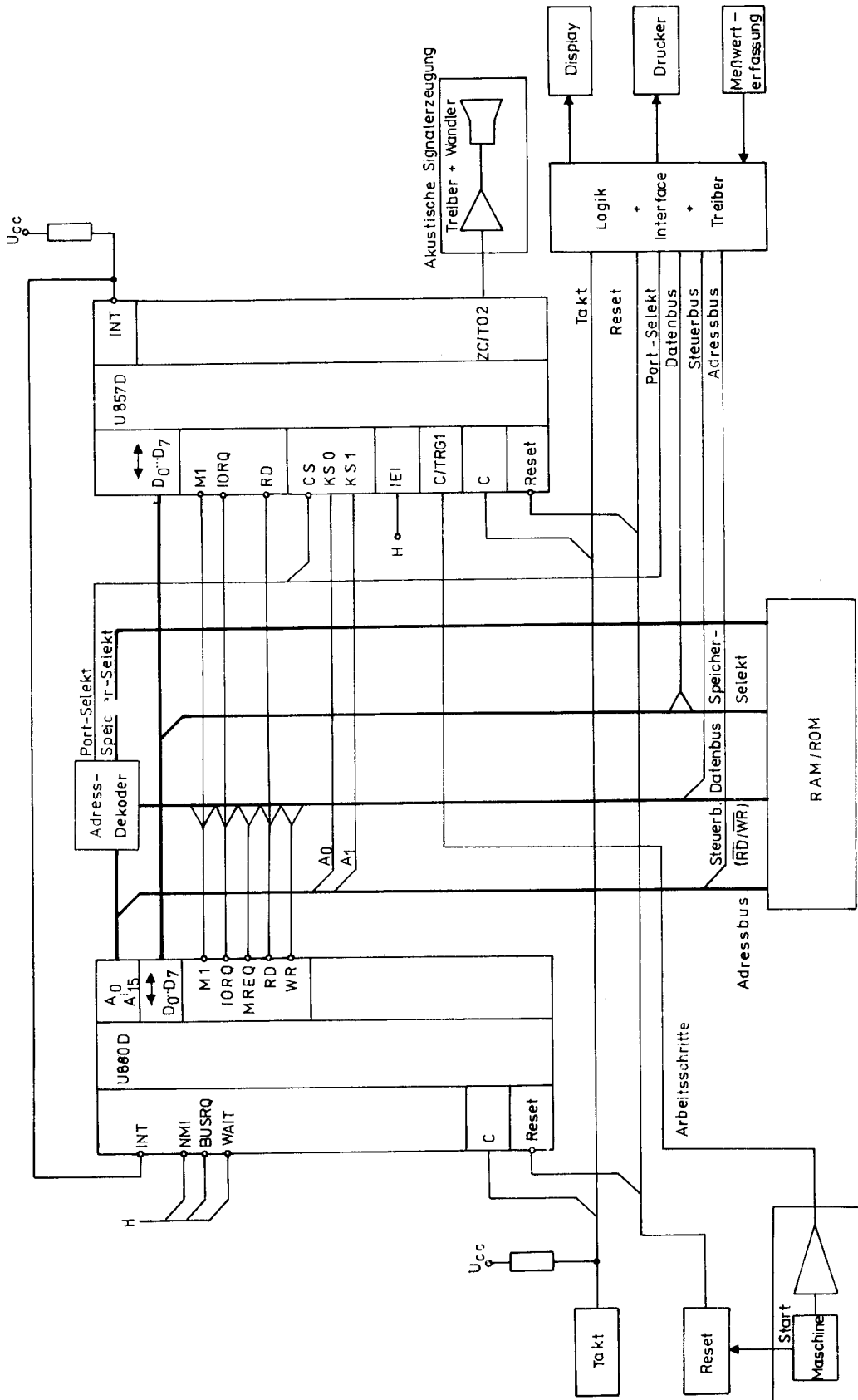


Bild 35: Prinzipielle Schaltung zum Applikationsbeispiel

10. Technische Daten der U 857 D

Gesetzliche Grundlage für alle technischen Daten des Bauelementes U 857 D bildet der Fachbereichsstandard TGL 37002. Abnahmeregeln, Prüfverfahren, Transport und Lagerung sind in der TGL 24 951 festgelegt.

10.1. Zuverlässigkeitswerte

Prüfzuverlässigkeit

$$\text{Prüfausfallrate } \lambda_{PO,G} \leq 3 \cdot 10^{-5} \cdot \text{h}^{-1}$$

Betriebszuverlässigkeit (garantiert)

Bei mittlerer elektrischer Belastung (Betriebsspannung 4,75 V bis 5,25 V und Umgebungstemperatur $\vartheta_a \leq 50^\circ\text{C}$) normaler klimatischer Beanspruchung (TGL 24 951) und vernachlässigbare, mechanische Beanspruchung wird eine Betriebsausfallrate

$$\lambda_{BG} \leq 3 \cdot 10^{-6} \cdot \text{h}^{-1}$$

bezogen auf die durch den Schaltkreis verursachten Funktionsausfälle von Geräten und Anlagen, vom Hersteller garantiert.

Betriebszuverlässigkeit (erwartet)

Unter den gleichen Einsatzbedingungen wie sie bei der garantierten Betriebsausfallrate gefordert werden, kann eine Betriebsausfallrate von

$$\lambda_{BE} \leq 1 \cdot 10^{-6} \cdot \text{h}^{-1}$$

bezogen auf die durch den Schaltkreis verursachten Funktionsausfälle von Geräten und Anlagen, erwartet werden (bezogen auf die Bauelementegesamtheit).

10.2. Ein- und Ausgänge

Reine Eingänge sind die Anschlüsse \overline{RD} , \overline{IORQ} , $\overline{M1}$, \overline{RESET} , C, \overline{CS} , KS1, KSO, IEI und C/TRGO - C/TRG3.



Bild 36: Eingänge der U 857

Die Schaltung der Eingänge zeigt Bild 36.

Der Transistor T1 ist ein Enhancement-Transistor, der wegen $U_{GS} = 0 \text{ V}$ stets gesperrt ist und mit seinem Durchbruch das Gate des Basistransistors vor Zerstörung schützt.

Ausgänge sind die Anschlüsse IE0, ZC/T00-2 und INT.

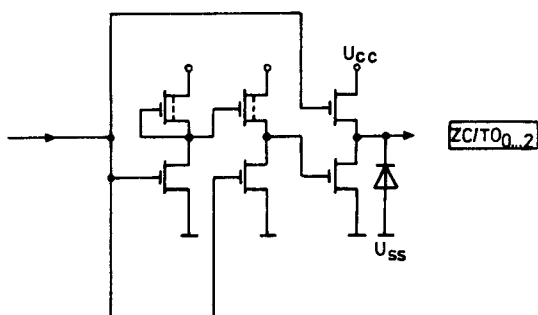


Bild 37: Schaltung der Ausgänge ZC/T00, ZC/T01, ZC/T02

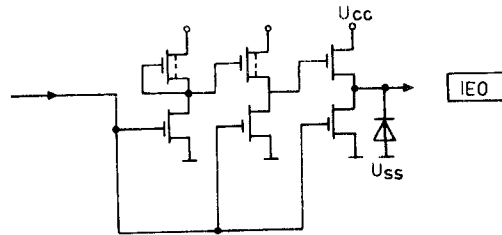


Bild 38: Schaltung des Ausgangs IEO

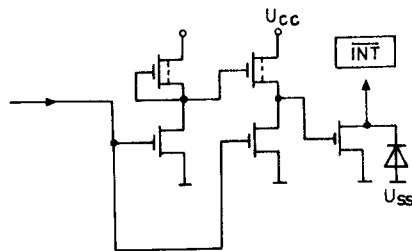


Bild 39: Schaltung des Ausgangs $\overline{\text{INT}}$

Die Ausgänge ZC T00-2 können Darlington-Transistoren treiben.
 Der Ausgang $\overline{\text{INT}}$ ist ein Open-Drain-Ausgang und extern mit einem Widerstand gegen U_{CC} zu beschalten (Sammelschiene $\overline{\text{INT}}$).

Tri-State-Ein- und Ausgänge

Tri-State, oder Dreizustands-Ein- und Ausgänge sind die Anschlüsse D_0 bis D_7 . Die zugehörige Schaltung zeigt Bild 40.

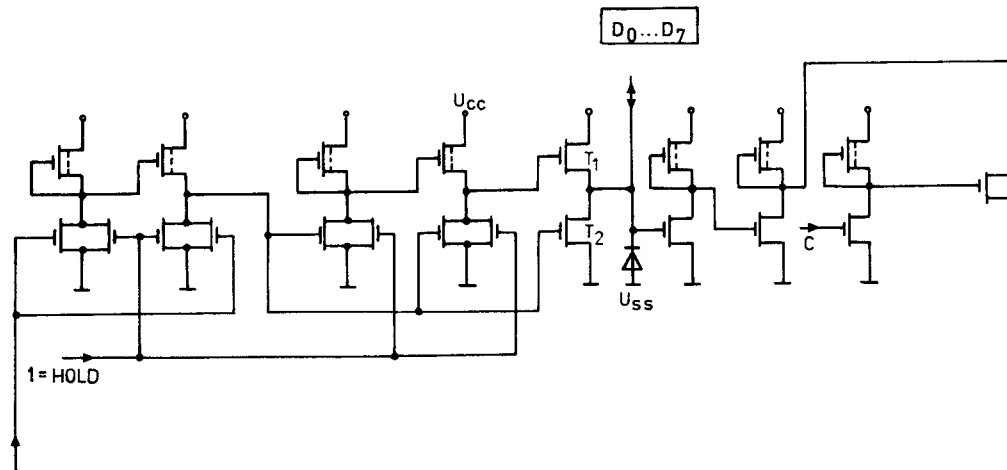


Bild 40: Schaltung der Tri-State-Ein- und Ausgänge $D_0 - D_7$

Mit dem Signal HOLD auf logisch 1 (H-Pegel) werden die Ausgangstreibertransistoren T1 und T2 in den hochohmigen Zustand gebracht; die Schaltung arbeitet als Eingang. Ist HOLD = 0, arbeitet die Schaltung als Ausgang, die Transistoren T1 und T2 schalten entweder nach U_{cc} oder nach U_{ss} durch.

10.3. Elektrische Kennwerte

Grenzwerte

bei $\mathcal{J}_a = 0$ bis 70°C , alle Spannungen bezogen auf $U_{ss} = 0$ V

Kenngröße	Kurzzeichen	Einheit	Kleinstwert	Größt- wert	Bemerkung
Betriebsspannung	U_{CC}	V	-0,5	7	
Eingangsspannung	U_I		-0,5	7	
Betriebstemperaturbereich	\mathcal{J}_a		0 bis	70	
Lagerungstemperaturbereich	\mathcal{J}_{stg}	$^\circ\text{C}$	-55 bis	125	
Verlustleistung	P	W	-	0,7	bei $\mathcal{J}_a = 25^\circ\text{C}$

Statische Kennwerte

Kenngröße	Kurzzeichen	Einheit	Kleinstwert	Größt- wert	Bemerkung
Betriebsspannung	U_{CC}		4,75	5,25	$\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
Eingangsspannung	U_{IL}		-0,5	0,8	$\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
	U_{IH}	V	2	U_{CC}	$\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
Takteingangsspannung	U_{ILC}		-0,5	0,45	$\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
	U_{IHC}		$U_{CC} - 0,2$	U_{CC}	
Ausgangsspannung	U_{OL}		-	0,4	bei $I_{OL} = 1,8$ mA $\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
	U_{OH}		2,4	-	bei $I_{OH} = -0,25$ mA $\mathcal{J}_a = 0^\circ\text{C} \dots 70^\circ\text{C}$
Eingangsreststrom	I_{LI}		-	10	
Reststrom des Tri-State-Ausgangs im hochohmigen Zustand	I_{LO}	μA	-	10	bei $U_I = 0$ V und $5,25$ V
Reststrom des Datenbusses bei Eingabe	I_{LD}		-	10	$\mathcal{J}_a = 0^\circ\text{C}$ und 70°C
Reststrom des $\overline{\text{INT}}$ -Einganges im hochohmigen Zustand	I_{LINT}		-	10	- " -
Taktkapazität	C_{CP}	pF	-	25	bei $\mathcal{J}_a = 25^\circ\text{C}$
Eingangskapazität	C_I		-	7	und $f = 0,5 \dots 2$ MHz
Ausgangskapazität	C_O		-	14	
Stromaufnahme	I_{CC}	mA	-	80	bei $U_{CC} = 5,25$ V und $\mathcal{J}_a = 25^\circ\text{C}$
Darlington-Treiberstrom für die Signale ZC/TO0, ZCTO1, und ZC/TO2	I_{OHD}	mA	-1,5		bei $U_{CC} = 4,75$ V, $U_{OH} = 1,5$ V $R_{ext} = 390 \Omega$ und $\mathcal{J}_a = 25^\circ\text{C}$

Dynamische Kennwerte

Signal	Kenngröße	Kurzzeichen	Einheit	Kleinwert	Größtwert
-	Taktperiode	t_c		400	1)
-	High-Breite des Taktes	$t_w(CH)$		170	2000
-	Low-Breite des Taktes	$t_w(CL)$		170	2000
-	Anstiegs- und Abfallzeiten des Taktes	t_r, t_f		-	30
-	Alle Haltezeiten für spezifizierte Setzzeiten	t_H		0	
DO bis D7	Datensetzzeit bis zur L/H-Flanke von C während Schreib- oder $\overline{M1}$ -Zyklus	$t_{SC(D)}$		60	
KSO, KS1	Kanalauswahlzeit bis zur L/H-Flanke von C während Lese- oder Schreibzyklus	$t_{SC(KS)}$		160	
\overline{CS}	Chipfreigabezeit bis zur L/H-Flanke von C während Lese- oder Schreibzyklus	$t_{SC(CS)}$	ns	160	-
\overline{IORQ}	\overline{IORQ} -Setzzeit bis zur L/H-Flanke von C während Lese- oder Schreibzyklus	$t_{SC(IR)}$		250	
$\overline{M1}$	$\overline{M1}$ -Setzzeit bis zur L/H-Flanke von C während INTA- oder $\overline{m1}$ -Zyklus	$t_{SC(M1)}$		210	
\overline{RD}	\overline{RD} -Setzzeit bis zur L/H-Flanke von C während Schreib- oder INTA-Zyklus	$t_{SWC(RD)}$		115	
\overline{RD}	\overline{RD} -Setzzeit bis zur L/H-Flanke von C während Lese oder $\overline{M1}$ -Zyklus	$t_{SRC(RD)}$		240	
	Taktperiode (Betriebsart: Zähler)	$t_c(CK)$		$2t_c$	
	Takt-Setzzeit bis zur L/H-Flanke von C für einen anliegenden Zählimpuls (Betriebsart: Zähler)	$t_s(SK)$		210	-
C/ TRG0 bis C/ TRG3	Trigger-Setzzeit bis zur L/H-Flanke von C für die Freigabe des Vorteilers auf die folgende steigende Flanke von C (Betriebsart: Zeitgeber)	$t_s(TR)$	ns	210	
	Takt- und Trigger-Anstiegs- und Abfallzeiten (in beiden Betriebsarten)	t_r, t_f		-	50
	High-Pulsbreite von Takt- und Triggersignal	$t_w(CTH)$		200	
	Low-Pulsbreite von Takt- und Triggersignal	$t_w(CTL)$		200	-

1) $t_c = t_w(CH) + t_w(CL) + t_r + t_f$

Anmerkung: Das RESET-Signal muß mindestens 3 Takt-Zyklen aktiv sein.

Verzögerungszeiten

bei $U_{cc} = 4,75 \text{ V}$; $U_{IL} = 0,8 \text{ V}$, $U_{IH} = 2 \text{ V}$, $U_{ILC} = 0,45 \text{ V}$, $U_{IHC} = 4,55 \text{ V}$, $C_L = 100 \text{ pF}$ und $\vartheta_a = 70 \text{ }^\circ\text{C}$

Signal	Kenngröße	Kurzzeichen	Einheit	Kleinsterwert	Größtwert	Bemerkung
DO-D7	Verzögerung des Datenausganges von der L/H-Flanke von C während Lesezyklus	$t_{DR}(D)$	ns		490	bei $U_{cc} = 4,75 \text{ V}$
	Datenausgangsverzögerung von der H/L-Flanke von IORQ während INTA-Zyklus	$t_{DI}(D)$			350	bei $U_{cc} = 4,75 \text{ V}$
	Verzögerung bis zum Floaten des Busses	$t_F(D)$			240	bei $U_{cc} = 4,75$
IEO	IEO-Verzögerungszeit von der H/L-Flanke von IBI	$t_{DL}(IO)$			200	bei $U_{cc} = 4,75$
	IEO-Verzögerungszeit von der L/H-Flanke von IBI	$t_{DH}(IO)$			230	
	IEO-Verzögerungszeit von der H/L-Flanke von M1 während RETI	$t_{DM}(IO)$			310	
\overline{INT}	\overline{INT} -Verzögerungszeit vor der L/H-Flanke von C/TRG (Betriebsart: Zähler)	$t_{DCT}(IT)$			$2t_c + 210$	bei $U_{cc} = 4,75 \text{ V}$
	\overline{INT} -Verzögerungszeit von der L/H-Flanke von C (Betriebsart: Zeitgeber)	$t_{DC}(IT)$			$2t_c + 210$	bei $U_{cc} = 4,75 \text{ V}$
ZC/T00 bis ZC/T02	ZC/TO-Verzögerungszeit von der L/H-Flanke von C, ZC/TO High	$t_{OH}(ZC)$	ns		200	
	ZC/TO-Verzögerungszeit von der H/L-Flanke von C, ZC/TO Low	$t_{DL}(ZC)$			200	

Die Zeitmessungen erfolgen, wenn nicht anders angegeben, bei folgenden Spannungen:

	H	L
Takt	4,2 V	0,8 V
Ausgang	2 V	0,8 V
Eingang	2 V	0,8 V
FLOAT	$\Delta U = \pm$	0,5 V

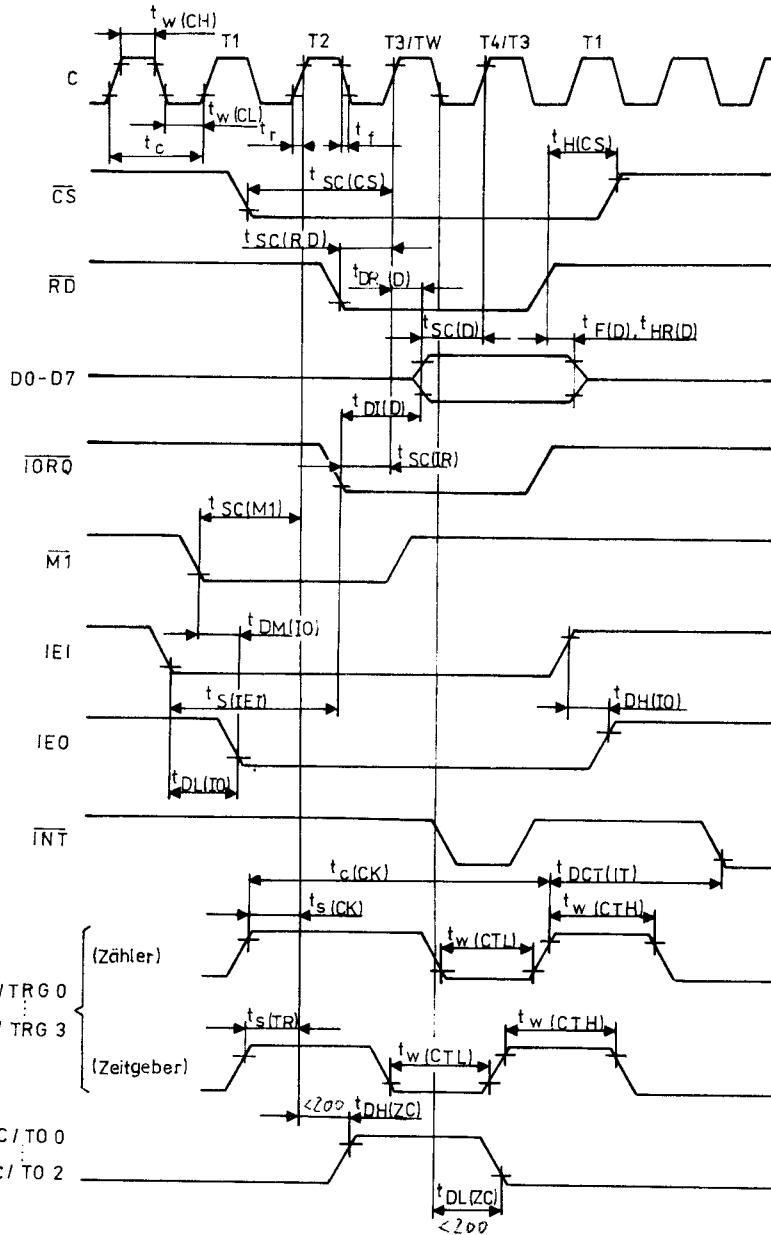


Bild 41: Zeitverhalten der U 857 D

10.4. Gehäuse der U 857 D

Die U 857 D wird in einem 28poligen Plastikgehäuse nach TGL 26 713 (Bauform 21.2.3.2.28) ausgeliefert (Bild 42). Die Masse beträgt ca. 4 g.

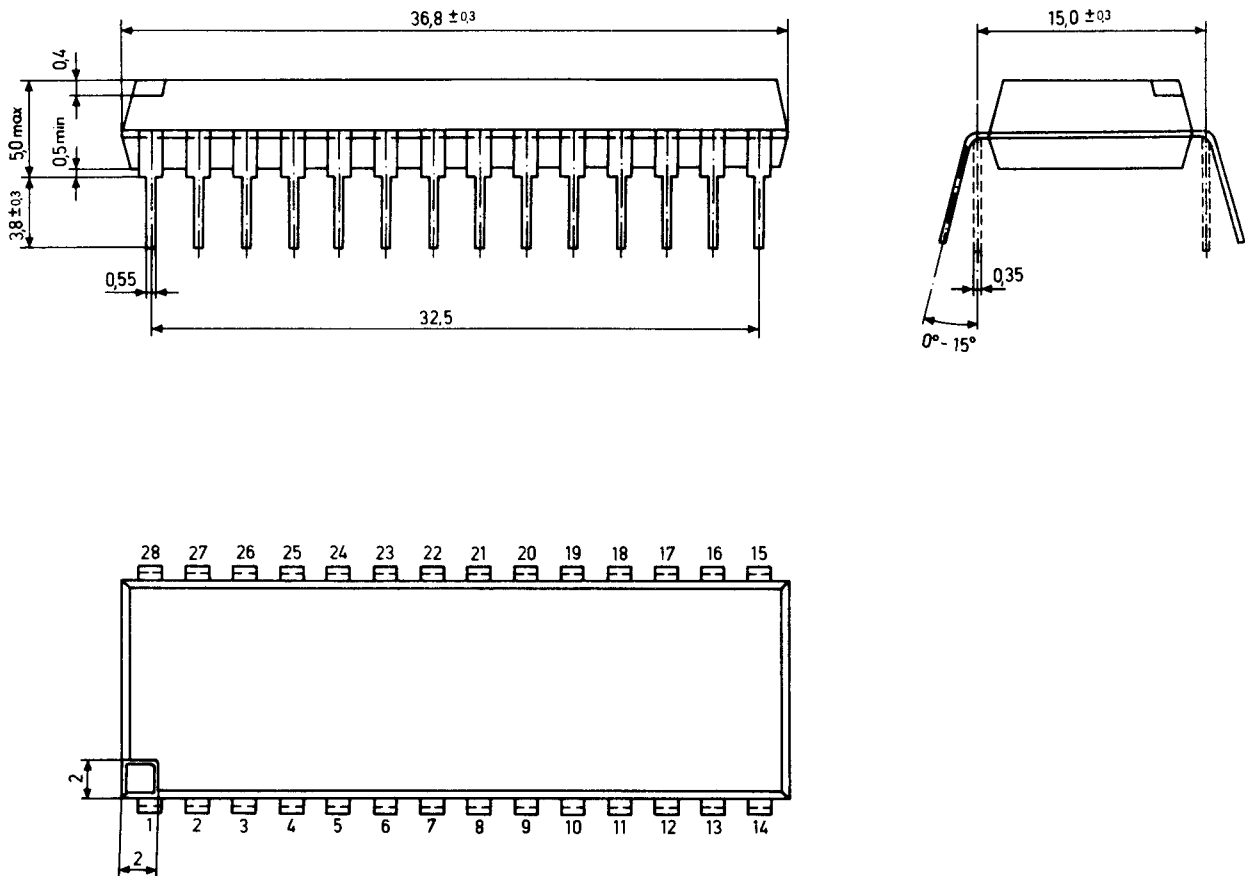


Bild 42: Gehäuse der U 857 D

Rs 1/82-1 V 71 339 Ko



elektronik
export·import

VOLKSEIGENER AUSSENHANDELSBETRIEB DER
DEUTSCHEN DEMOKRATISCHEN REPUBLIK
DDR 1026 BERLIN ALEXANDERPLATZ
HAUS DER ELEKTROINDUSTRIE

VEB FUNKWERK ERFURT

im VEB Kombinat Mikroelektronik

DDR - 501 Erfurt, Rudolfstraße 47

Telefon: 5 80

Telex: 061 306

Kabel: FUNKWERK ERFURT