

NDR-Computer als speicherprogrammierbare Steuerung

Es ist unser Ziel, daß Sie möglichst schnell mit der Programmierung beginnen können. Über die Arbeitsweise der Hardware, also des NDR-Computers selbst, wird nur das zum Verständnis des Steuerungsprinzips unbedingt Erforderliche berichtet. Wir setzen außerdem voraus, daß alle Funktionseinheiten des Computers zusammengebaut und betriebsbereit zur Verfügung stehen.

Blockschaltung des NDR-Computers als SPS

Die Blockschaltung in Bild D1.1 enthält alle Funktionseinheiten und Geräte, die wir bereits für die ersten Programmerversuche benötigen. Ihr System sollte wie folgt ausgestattet sein:

Einplatinencomputer	SBC 2 (oder Z80 CPU und ROA 64)
Anschlußbaugruppe Tastatur	KEY
Tastatur	Tast 1 (oder Tast 2)
Anschlußbaugruppe Datensichtgerät	GDP 64
Datensichtgerät (Monitor)	
Ein-/Ausgabebaugruppe	IOE
Experimentier-Ein-/Ausgabe	EINAUS
Busplatine	BUS 1, 2 oder 3
Stromversorgung	POW 5V (oder NE 2)
Netztransformator	
EPROM-SPS	ESPS

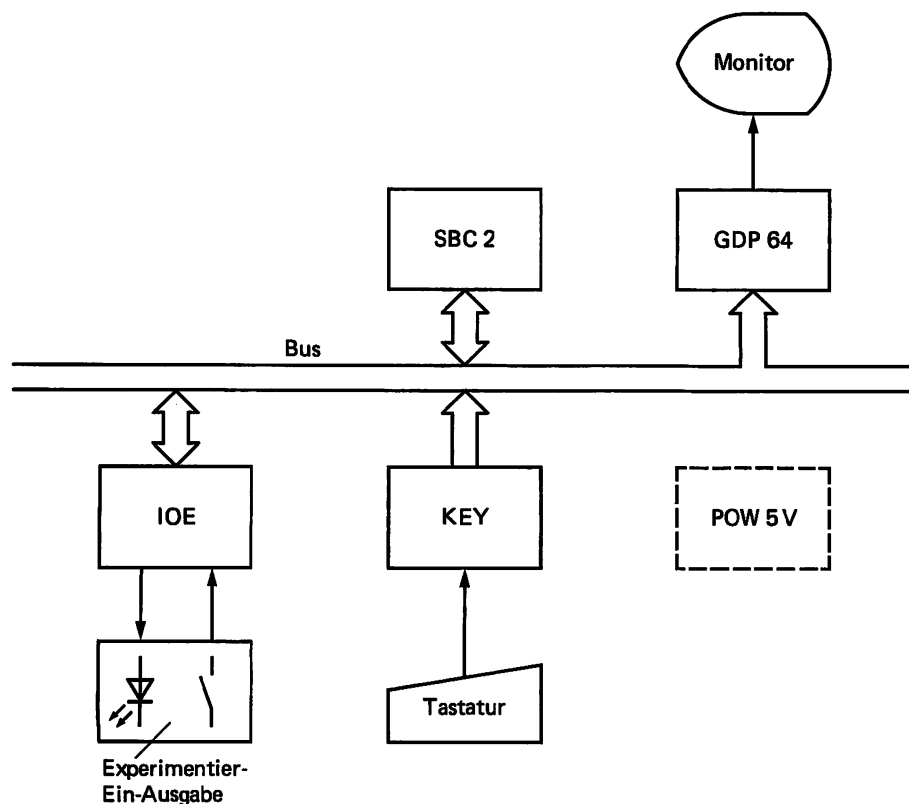
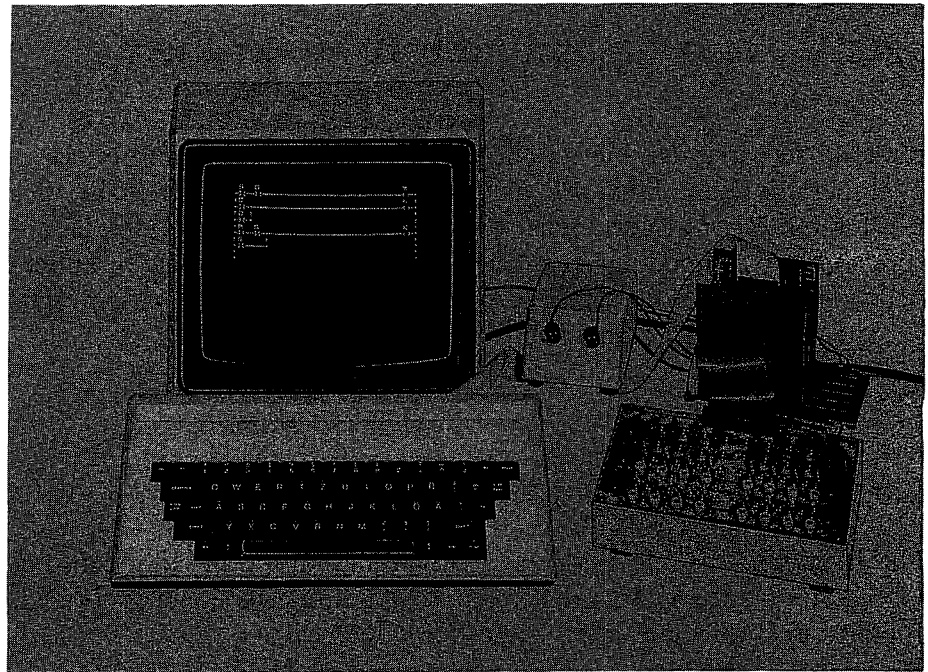


Bild D1.1
Baugruppen des NDR-Computers
als SPS (Minimalsystem).

D

2

Bild D2.1
NDR-Computer als SPS. Anordnung der Baugruppen und Geräte des Minimalsystems.



Das Bild D2.1 zeigt die Anlage des NDR-Computers als SPS. Die Lage von Stift- und Buchsenleiste erlaubt nur eine ganz bestimmte Montage. Die Platinen müssen nach Bild D3.1 in den Führungsnuten sitzen. Achten Sie darauf, daß der erste Anschlußpin auf der linken Seite der Funktionskarte (von der Bestückungsseite aus gesehen) in die von links gezählte erste Buchse der Steckerleiste auf der Busplatine gesteckt sein muß. Falls Sie die Funktionskarte versehentlich um nur eine Buchse nach rechts versetzt einbauen, können nach dem Einschalten Bauteile auf der Karte zerstört werden. Auf den Anschluß unserer Geräte kommen wir noch zu sprechen.

Grundsätzliche Funktion des NDR-Computers als SPS

Mikrocomputer

Die **zentrale Funktionseinheit** ist ein **Mikrocomputer** (Abkürzung: uC). Übernimmt der uC die Steuerungsaufgaben in einem elektronischen System, so bezeichnet man dieses als **uC-System**. Es ist gekennzeichnet durch das Zusammenwirken von uC und **Peripherie**, wobei der uC das steuernde Organ für den Gesamtablauf ist. Zur Peripherie gehören in unserem Fall: Signalgeber, Meßumformer, Stellglieder; aber auch Drucker, Datensichtgeräte, Plotter usw. Einen kompletten uC enthält die Platine **SBC 2** (Bild D4.1). Auf ihr befinden sich der 8-Bit Mikroprozessor (Abkürzung: uP) Z80, ein Taktgenerator und Halbleiterspeicher. Der uP (auch als CPU = Central Processing

Unit bezeichnet) verarbeitet die Daten nach Programmanweisungen. Außerdem steuert er den Funktionsablauf des gesamten uC-Systems. Die CPU holt die Befehle aus dem Programmspeicher und führt sie aus. Wie alle komplexen digitalen Systeme arbeitet auch der uC synchron in einem zyklischen Zeitablauf. Die zur zeitlichen Steuerung des Arbeitsablaufs benötigten Impulse liefert ein Taktgenerator. Er erzeugt Rechteckimpulse mit einer Frequenz von 4 MHz.

Auf der Karte befinden sich zwei verschiedene Speicherarten. Auf den Plätzen J6 und J7 sitzt je ein Festwertspeicher. In diese Speicherbausteine, sogenannte EPROMs (Erasable Programmable Read Only Memory), werden fertig entwickelte Programme eingeschrieben, die sich auch wieder löschen lassen. Während des Betriebs kann der Speicherinhalt jedoch nur gelesen werden. Das auf unserer SBC2-Karte befindliche EPROM vom Typ 2732A o.ä. hat die **Speicherkapazität von 4096 Byte = 4 kByte**. Somit beansprucht unser SPS-Betriebsprogramm, auch Monitorprogramm genannt, eine Speicherkapazität von 8 kByte.

Die Plätze J8 und J9 sind von je einem Schreib-Lese-Speicher (RAM = Random-Access Memory) vom Typ HM 6116 o.ä. belegt. Ein solches RAM hat die **Kapazität von 2048 Byte = 2 kByte**.

Im Festwertspeicher befindet sich das zur Durchführung einer bestimmten Aufgabe erforderliche Programm. Ohne ein derartiges Programm ist der uC nicht arbeitsfähig. In den beiden Fassungen J6 und J7 stecken deshalb die beiden **Programmspeicherbausteine „SPS“**, und zwar in

Fassung J6 Speicherbaustein „SPS 0“
Fassung J7 Speicherbaustein „SPS 1“.

Nur mit dem darin gespeicherten Programm arbeitet der NDR-Computer als SPS (Bild D 2.1).

Falls Ihr NDR-Computer bereits mit der Vollausbau-CPU-Karte und einer ROA64-Speicherkarte ausgerüstet ist, benötigen Sie das SPS-Betriebsprogramm in einem EPROM vom Typ 2764 o.ä. mit einer Speicherkapazität von 8 kByte.

Die Schreib-Lese-Speicher nehmen die variablen Daten auf. Es sind **Arbeitsspeicher**. In unserem Fall wird darin das entwickelte Steuerungsprogramm aufbewahrt. Vorsicht, RAMs sind flüchtige Speicher; beim Abschalten der Stromversorgung geht auch der **Speicherinhalt verloren!**

Die Baugruppe SBC 2 enthält noch eine „Startschaltung“. Bei Inbetriebnahme muß der uC immer in einen bestimmten Anfangszustand gesetzt werden (**RESET-Funktion**). Durch Betätigung eines Tastschalters, den Sie am RESET-Anschluß anbringen sollten, können Sie jederzeit den Anfangszustand herstellen und das System erneut in die Startposition bringen.

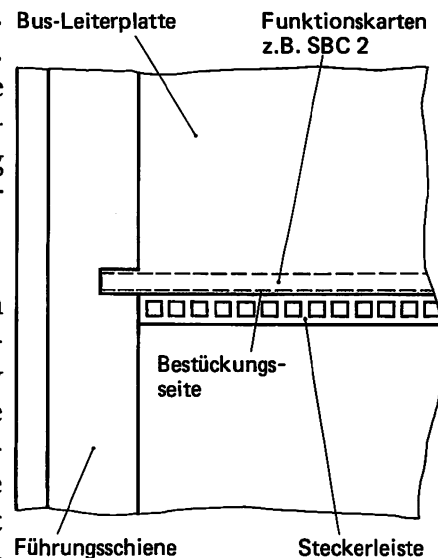
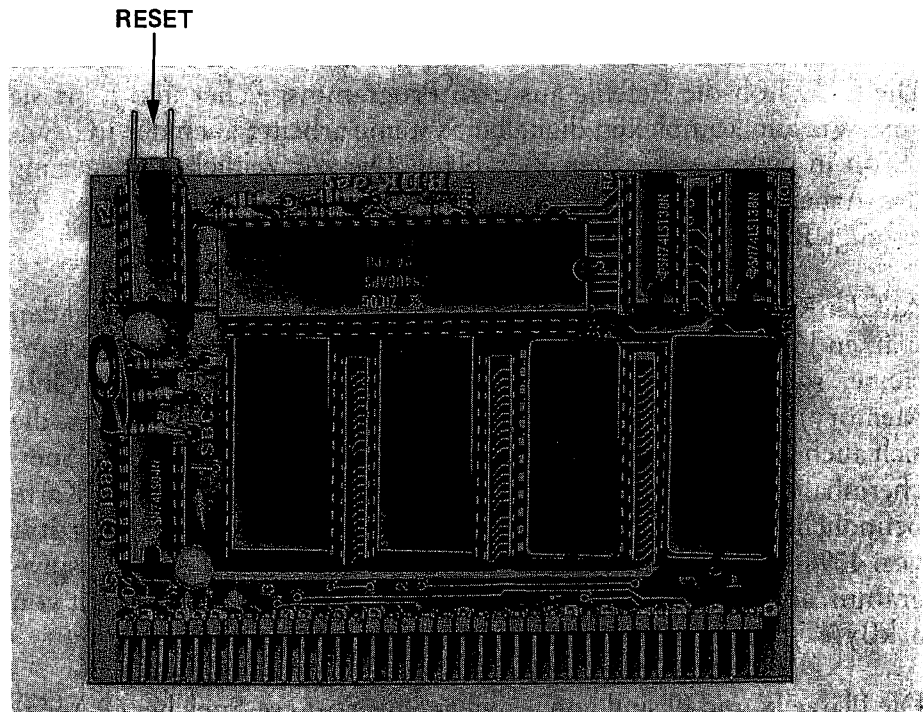


Bild D 3.1
Bus-Platine, Draufsicht. Anordnung einer Funktionsleiterplatte.

D

4

Bild D 4.1
Mikrocomputer-Baugruppe SBC2.
Anordnung der Speicherbausteine.



Busstruktur

Charakteristisch für alle Mikrocomputer ist die Verbindung der verschiedenen Funktionsbaugruppen untereinander über ein **Bussystem**. Busse sind vieladrige, parallel verlaufende Leitungen, an die einzelne Baugruppen des uC-Systems gleichberechtigt angeschlossen sind. Über die Busleitungen erfolgt der **Austausch von Daten** zwischen den einzelnen Bausteinen. Es werden dann Speicherplätze und Ein-/Ausgangskanäle adressiert und Steuersignale übermittelt.

Man unterscheidet deshalb zwischen **Daten-, Adreß- und Steuerbus**. Durch die Busstruktur ist eine nahezu beliebige Erweiterung des Mikrocomputers durch zusätzliche Funktionsgruppen möglich. Je nach Ausbaugrad Ihres Systems sind Sie im Besitz der Baugruppe BUS 1, 2 oder 3.

Tastatur und Tastaturanschluß

Sehen Sie sich bitte auf der Blockschaltung in Bild D 1.1 die Anordnung dieser beiden Baugruppen im Gesamtsystem an. Das von Ihnen entwickelte Programm muß in den uC eingespeichert werden. Es ist vom „Papierspeicher“ in den Arbeitsspeicher (RAM) des uC zu übertragen. Wie Sie noch sehen werden, sind SPS-Programme in einer bestimmten, leicht zu erlernenden **Programmiersprache** abgefaßt. Die über die Tastatur eingegebenen Symbole der Programmiersprache werden vom SPS-Betriebsprogramm in ein elektronisches Bitmuster übersetzt, das von der Elektronik des NDR-Computers verstanden wird. Außer numerischen Zeichen (0 bis 9) sind noch Buchstaben und Sonderzeichen (z. B. & / = ..) darzustellen. Alle Zeichen aus diesem Zeichenvorrat nennt man **alphanumerische Zeichen**.

Der NDR-Computer versteht den sogenannten **ASCII-Code** (American Standard of Information Interchange). Das Codewort besteht aus 7 Bit. Da das System mit einer 8 Bit-Struktur arbeitet, wird das noch verfügbare achte Bit oft für Kontrollzwecke benutzt. Mit 7 Bit sind $2^7 = 128$ verschiedene Zeichen darstellbar.

Neben alphanumerischen Zeichen beinhaltet der ASCII-Code auch noch Steuerzeichen. Theoretisch benötigt man zur Eingabe aller Zeichen 128 Tasten auf der Tastatur. In der Regel wird die Anzahl der erforderlichen Tasten reduziert, indem man eine **Mehrfachbelegung** der Tasten vornimmt. Hierdurch reduziert sich die Zahl der Zeichentasten auf ca. 50. Hinzu kommen noch die Sondertasten. Über die Bedeutung der Tasten werden wir Sie innerhalb der folgenden Anwendungen informieren.

Über ein Flachbandkabel ist die Tastatur mit der Baugruppe **KEY** verbunden (Bild D 2.1). Sie ist das Bindeglied zwischen uC und Tastatur. Bei Betätigung einer Taste wird das zugehörige **Bitmuster über Datenleitungen** zur KEY-Baugruppe übertragen und dort zwischengespeichert. Der uC ruft die Daten aus dem Zwischenspeicher ab und übernimmt sie. Eine Steuerelektronik sorgt dafür, daß die zwischengespeicherten Daten nur einmal gelesen werden.

Datensichtgerät (Monitor) mit Anschlußbaugruppe

Die per Tastatur in den Arbeitspeicher eingegebenen Daten muß der Programmierer auf dem Monitor überprüfen können. Es kann leicht passieren, daß während des Programmierens eine falsche Taste gedrückt wird. Sofort haben wir einen Fehler im Programm und eine Fehlfunktion der SPS ist die Folge.

Auch der Computer hat uns manches mitzuteilen. Seine Bedienung ist teilweise „menügesteuert“. Der Computer teilt uns beispielsweise mit, welche Taste zu drücken ist, um eine bestimmte Funktion auszulösen. Auch ist es für den Programmierer wünschenswert, daß das als **Anweisungsliste** oder **Schaltfunktion** eingegebene Steuerungsprogramm als Grafik, z. B. in Form eines Kontaktplans, wiedergegeben werden kann.

In jedem Fall sind große Datenmengen gleichzeitig darzustellen. Das gebräuchlichste Ausgabegerät hierfür ist der Bildschirm. Zum Anschluß an einen uC eignet sich sowohl ein handelsübliches Fernsehgerät mit einem AV-Eingang als auch ein spezielles Datensichtgerät, das häufig als Monitor bezeichnet wird. Den angeführten Geräten wird das Bildsignal (Videosignal) direkt zugeführt (Bild D 6.1). Fernsehgeräte ohne AV-Eingang bekommen das Bildsignal unter Zwischenschaltung eines HF-Modulators als HF-Signal über die Antennenbuchse.

Das Bindeglied zwischen dem Mikrocomputer SBC 2 und dem Monitor ist die **Anschlußkarte GDP 64**. Das zentrale Bauelement ist ein Grafikprozessor. Mit ihm sind u. a. auf dem Bildschirm 256 x 512 Bildpunkte ansteuerbar. Das auf dem Bildschirm wiederzugebende Bild ist

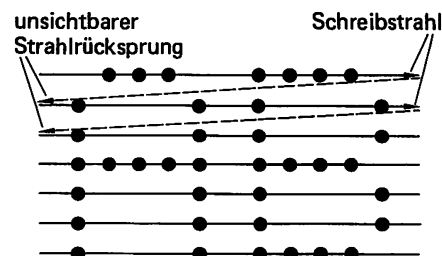
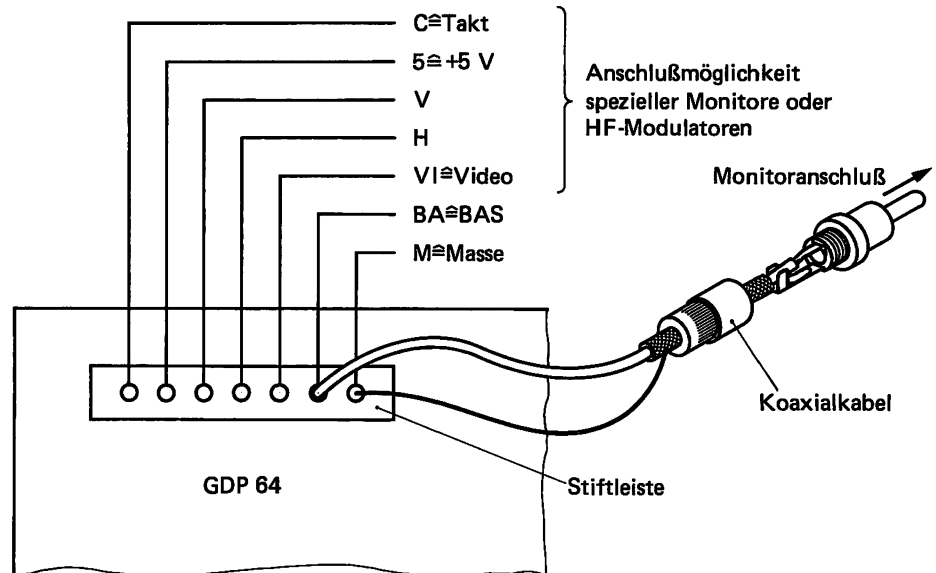


Bild D 5.1
Buchstaben, zusammengesetzt durch Bildpunkte einer 5x7-Punktmatrix.

Bild D 6.1
Funktionskarte GDP64kByte. Anschluß des Monitors, von der Bestückungsseite gesehen. Das BAS-Signal enthält die vollständige Bildinformation sowie die erforderlichen Synchronisationssignale. (BAS = Bild-Austast- und Synchronisationssignal).



in Bildpunkte zerlegt, deren Positionen in einem Bildschirmspeicher abgelegt sind. Der Speicherinhalt wird in Zeitintervallen von 40 ms abgerufen und auf den Bildschirm geschrieben (entspricht 25 Bildern/s). Die Zeichen werden in einer **Matrix von 5x7 Bildpunkten** (Bild D 5.1) dargestellt. Der Schreibvorgang verläuft ähnlich wie bei einem Fernsehempfänger. Wir haben die Signalverarbeitung hier sehr vereinfacht dargestellt.

Für den Anwender ist das aber ausreichend, da wir uns nicht mit der Videotechnik, sondern in erster Linie mit speicherprogrammierten Steuerungen befassen wollen.

Ein-/Ausgabebaugruppe und Experimentier-Ein-/Ausgabe

Können wir im derzeitigen Ausbau das System schon als SPS betreiben? Nein!

Unser auf dem Papier entwickeltes Programm können wir zwar über die Tastatur in den Arbeitsspeicher des Computers geben. Der Speicherinhalt läßt sich auf dem Bildschirm des Monitors darstellen. Sie können damit kontrollieren, ob der Speicherinhalt mit Ihrem Entwurf übereinstimmt. Falls nicht (was vorkommen kann), sind entsprechende Korrekturen möglich.

Es ist jedoch noch nicht möglich, unser Programm zu testen und später den Steuerungsablauf zu simulieren. Das Programm benötigt **Eingabesignale**, die logisch zu verknüpfen sind. Gemäß der Verarbeitung entstehen **Ausgabesignale**, die Stellglieder betätigen oder Anzeigen schalten. Selbstverständlich könnten wir kleine Steuerungen aufbauen und betreiben. Denken Sie doch an Modelle der Fischer-Technik oder an Modelleisenbahnen. Viel einfacher ist es jedoch, die Signaleingaben mit Schaltern vorzunehmen und Signalausgaben durch Leuchtdioden anzuzeigen. Dazu dient die **Experimentier-Ein-/Ausgabe**. Unser System kann 16 Eingabesignale (E0 bis E15) aufnehmen und

16 Ausgabesignale (A0 bis A15) abgeben. Mit unserer Experimentier-Ein-/Ausgabe lassen sich 16 Eingabesignale (E0-E15) über Umschalter und 16 Ausgabesignale über LEDs simulieren. Die Portadressierung erfolgt auf der Baugruppe IOE (30H und 31H).

Achtung: Vor Inbetriebnahme der Ein-/Ausgabe unbedingt das zur Baugruppe gehörende Handbuch durchlesen.

Falls der Aufbau Ihrer Ein-/Ausgabeeinheit nicht so aussieht wie auf dem Bild D.2.1 bzw. auf der letzten Lehrbriefseite, ist es nicht weiter schlimm. Das Prinzip der Ein-/Ausgabesimulation läßt sich auch auf Ihrer Versuchsanordnung durchführen.

Nach Bild D.1.1 ist das Bindeglied zwischen der Experimentier-Ein-/Ausgabe und dem uC SBC 2 die **Baugruppe IOE**. Sie beinhaltet im wesentlichen **zwei Ausgabeports** zu je **8 Bit**. Außerdem sind auf der IOE-Baugruppe noch **zwei Eingabeports**, ebenfalls zu je **8 Bit**, vorhanden. Per Programm werden die Signalanschlüsse für die Ein- bzw. Ausgabe angesprochen.

Stromversorgung

Der NDR-Computer benötigt als Speisespannung eine Gleichspannung von 5 V, die von der **Baugruppe POW 5V** geliefert wird. Sie wird ebenfalls in das BUS-System gesteckt. Eine falsche Verdrahtung oder ein falsches Polen ist damit ausgeschlossen. Die erzeugte Spannung ist stabilisiert. Netzspannungsschwankungen wirken sich kaum auf die Ausgangsspannung aus. Sie müssen nur noch den Wechselspannungsanschluß der Baugruppe POW 5 V mit der Ausgangswechselspannung des Experimentiertransformators verbinden.

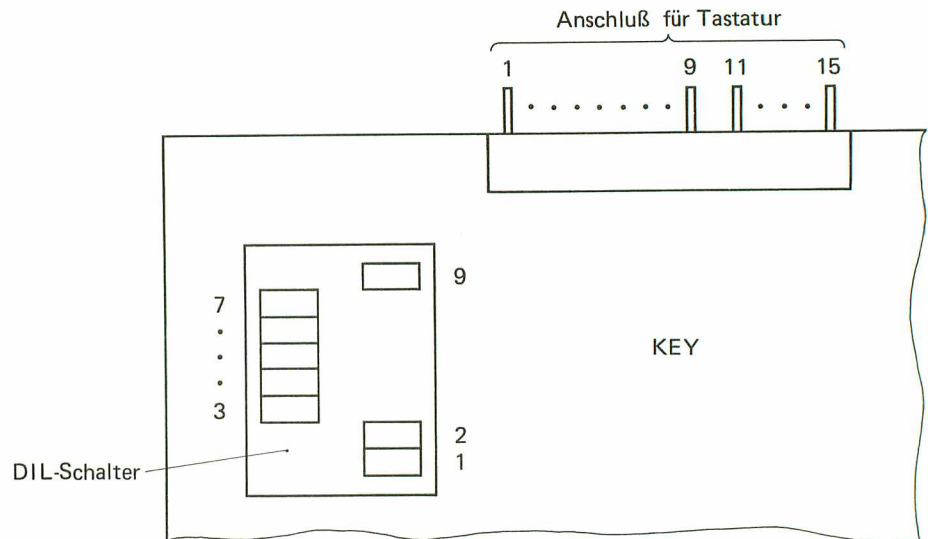
In großen Zügen, zunächst aber völlig ausreichend, haben Sie den Aufbau und die Funktion Ihres Computers kennengelernt. Falls Sie ihn selbst zusammengebaut haben, werden Ihnen manche Hinweise schon bekannt gewesen sein, da die Beschreibungen der einzelnen Baugruppen weitere Informationen enthalten.

Anschluß der Geräte an den NDR-Computer

Wir bleiben zunächst bei unserer Minimalausführung. Die **Funktionsgruppen** sollten Sie nach Bild D.2.1 bereits auf die Busplatine gesteckt haben. Achten Sie bitte darauf, daß alle Platinen in der Führungsnut der Busplatine sitzen (Bild D.3.1). Nur dann sind die richtigen Steckverbindungen hergestellt. Wir brauchen jetzt nur noch den Monitor, die Tastatur und die Experimentier-Ein-/Ausgabe anzuschließen.

Stellen Sie die Verbindung zwischen dem Monitor und der Baugruppe GDP 64 her. Das zu Ihrem Material gehörende **Koaxialkabel** hat auf der einen Seite einen Stecker für den Monitoranschluß, auf der ande-

Bild D 8.1
Funktionskarte KEY, von der
Bestückungsseite gesehen.



ren Seite eine Buchsenleiste für die Verbindung auf der Baugruppe GDP 64 (siehe Bild D 6.1). Stellen Sie die Verbindung so her, daß die Abschirmung des Koaxialkabels mit Pin M der Stiftleiste verbunden wird. Über das Kabel erfolgt die Zuführung des Bildsignals.

Die Tastatur wird mit einem Flachbandkabel an die Baugruppe KEY angeschlossen (Bild D 8.1). Die Steckerleiste befindet sich auf der Baugruppe. Konstruktion von Stecker- und Buchsenleiste ermöglichen nur eine eindeutige Steckverbindung.

Als letzte Baugruppe ist noch unsere Experimentier-Ein-/Ausgabe anzuschließen. Sie wird ebenfalls über eine Flachbandleitung mit der Baugruppe IOE verbunden.

Betrieb des NDR-Computers als SPS

Bevor wir einfache Programme in unsere SPS eingeben und testen, sind noch einige Vorbereitungen erforderlich. Diese wiederholen sich bei jeder neuen Programmeingabe und bei jedem Programmtest. Wir beschreiben deshalb die erforderlichen Maßnahmen an dieser Stelle ausführlich und verweisen später nur noch auf die entsprechenden Punkte.

Tastatur

Nehmen Sie bitte Ihre Tastatur zur Hand. Mit den folgenden Hinweisen sollen Sie sich bereits mit ihr vertraut machen. Nur über die Tastatur haben Sie Zugriff zum Computer. Die folgende Beschreibung bezieht sich auf die Tastatur mit der Bezeichnung „Tast 1“.

Betrachten Sie die folgenden Erläuterungen zunächst als eine theoretische Einführung. Die praktische Erprobung erfolgt einige Abschnitte später. Alle Zeichentasten der Tastatur haben eine **Doppelfunktion**. Wenn Sie eine Buchstabentaste drücken, wird auf dem Bildschirm des Monitors nur ein kleiner Buchstabe erscheinen. Betätigen Sie die **SHIFT-Taste** und die **Buchstabentaste gleichzeitig**, so wird ein großer Buchstabe geschrieben. Sie können somit große und kleine Buchstaben schreiben.

Die Zifferntasten und die Tasten mit Sonderzeichen sind bereits doppelt beschriftet. Wenn man die Zifferntaste allein drückt, erscheint auf dem Bildschirm das untere Tastensymbol. Werden **SHIFT-** und **Zifferntaste gleichzeitig** gedrückt, erscheint das obere Symbol.

Falls Sie einmal eine Taste längere Zeit betätigen, wird die Zeile auf dem Monitor kontinuierlich mit dem betreffenden Zeichen vollgeschrieben. Das ist kein Fehler Ihres Systems. Es ist eine Programmierhilfe, falls einmal eine Zeile mit dem gleichen Zeichen besetzt werden soll. Diese Wirkung wird als „**Repeat-Funktion**“ bezeichnet.

Zeichen für Anweisungen und Steuerbefehle (Kommandos)

Wir müssen dem System mitteilen, daß es bestimmte Funktionen (**Befehle**) ausführen soll. Dafür sind Anweisungen und Steuerbefehle nötig.

Vorab einige kurze Informationen. Im später folgenden Kapitel über Programmiersprachen und Programmierhilfen werden Anweisungen und Steuerbefehle noch ausführlich besprochen. Anweisungen sind Mitteilungen an den Computer. Sie legen u. a. fest

- welche Eingabesignale zu verarbeiten sind,
- wie die Eingabesignale zu verknüpfen sind,
- auf welche Ausgänge die Verknüpfungsergebnisse zu schalten sind.

Diese Anweisungen bilden das **Steuerungsprogramm**. Die Steuerbefehle oder Kommandos beziehen sich auf den Computer. Damit erfolgt ein Eingriff in den Funktionsablauf.

Wir kommen zunächst mit folgenden Anweisungen aus:

Tabelle D9.1: Zusammenstellung einiger Anweisungen

Zeichen	Bedeutung
!	Lade (muß immer am Anfang einer Anweisung bzw. Programmzeile stehen)
&	UND
/	ODER
N	NICHT
=	Zuweisung
PE	Programmende (damit muß jedes Programm abgeschlossen werden)

Einige wichtige Steuerbefehle, die mit Hilfe der CTRL-Taste ausgeführt werden, enthält die folgende Zusammenstellung:

Tabelle D 10.1: Zusammenstellung einiger Steuerbefehle

CTRL	K X	Rückkehr aus dem Editor in das Anfangsmenü, Speichern des eingegebenen Textes (Programms)
CTRL	Ä	Rückkehr von der Kontaktpendarstellung in das Anfangsmenü
CTRL	G	Löschen des Zeichens unter dem Cursor
CTRL	Y	Löschen einer Zeile
CTRL	Z	Löschen des Bildschirms
CTRL	V	Einfügen eines Zeichens

D

10

Befehle zur Cursor-Steuerung

Wenn Sie Ihr System in Betrieb nehmen und das Menü 1 (auch als EDIT-Modus bezeichnet) aufgerufen haben, erscheint auf dem Bildschirm ein kleines blinkendes Quadrat. Es ist eine Orientierungshilfe. Sie zeigt Ihnen, wo das Zeichen erscheint, das mit der Tastatur eingegeben wird. In der Fachsprache nennt man diese Marke „CURSOR“. Mit dem Cursor erfolgt auch die Festlegung, welches Zeichen zu löschen oder wo ein Zeichen einzufügen ist. Zu diesem Zweck muß man den Cursor auf dem Bildschirm verschieben können. Folgende Befehle sind dafür vorgesehen:

Tabelle D 10.2: Cursor-Steuerbefehle

CTRL	E	Bewegung des Cursors um eine Zeile nach oben
CTRL	X oder Taste ↓	Bewegung des Cursors um eine Zeile nach unten
CTRL	D oder Taste →	Bewegung des Cursors um ein Zeichen nach rechts
CTRL	S oder Taste ←	Bewegung des Cursors um ein Zeichen nach links

Beachten Sie folgenden wichtigen Hinweis:

Wenn Sie sich beispielsweise vertippt haben und das falsche Zeichen löschen wollen, so stellen Sie den Cursor auf das zu löschende Zeichen. Drücken Sie die CTRL-Taste und die G-Taste. Das unerwünschte Zeichen verschwindet sofort. CTRL- und Buchstaben-Taste müssen immer gleichzeitig gedrückt werden.

Auf der Tastatur finden Sie eine Taste mit der Bezeichnung „CR“. CR bedeutet „Carriage Return“. Bei Betätigung springt der Cursor zum Anfang der nächsten Zeile.

Ist ein Programm vom „Papier“ über die Tastatur auf den Bildschirm übertragen worden (Menü 1), muß das Programm in den Arbeitsspeicher des uC gebracht werden. Das Programm wird erst mit Betätigung der Tasten „CTRL K“ und anschließend „X“ vom Speicher übernommen.

Inbetriebnahme des Systems

Endlich ist es soweit. Ihren Aufbau haben Sie ja schon auf richtige Leitungsverbindungen überprüft, und die Geräte sind an die Wechselspannung von 220 V geschaltet. Auf dem Bildschirm des Monitors lesen Sie folgendes Anfangsmenü:

SPS Speicherprogrammierbare Steuerung

V 1.2 (C) 1984, Rolf-Dieter Klein

- 1 = Edit
- 2 = Start
- 3 = Kontaktplan
- 4 = speichern
- 5 = laden
- 6 = vergleichen
- 7 = Timer

Damit teilt uns der Computer mit, welche Programmteile des SPS-Betriebsprogramms in den beiden EPROMs enthalten sind. V1.2 deutet den Entwicklungsstand der Programme an. Außerdem sehen Sie noch den Cursor blinken.

Sollte das Anfangsmenü bei Inbetriebnahme nicht sofort erscheinen, dann drücken Sie bitte die RESET-Taste (wir nehmen an, daß Sie die Taste entsprechend dem Bild D 4.1 eingebaut haben). Sie geben damit nochmals die **Startanweisung** aus. Mit den angegebenen Ziffern können Sie bestimmte Funktionen aus dem SPS-Programm auswählen. Nachstehend sind nur die jetzt unmittelbar erforderlichen Programmpunkte kurz erläutert.

Nach dem Drücken einer der Zifferntasten 1 bis 7 ist der Computer bereit, eine der im Anfangsmenü angegebenen Funktionen auszuführen. Es bedeuten:

Taste mit Ziffer 1 (als Menü 1 bezeichnet) drücken:

Edit = Editieren. Editieren ist ein Fremdwort aus der Datenverarbeitung, es bedeutet „**Bearbeiten von Texten**“, d. h. Texte erstellen und ändern. Der Text ist in unserem Fall das Programm. Nach Drücken der Taste 1 können wir unser entwickeltes Programm mit der Tastatur eingeben. Der eingegebene Text erscheint sofort auf dem Bildschirm des Monitors. Haben Sie sich einmal vertippt, dann können Sie das Programm gleich korrigieren (siehe Seite D10).

Zeigt der Bildschirm das komplette und fehlerfreie Programm, wird es in den Arbeitsspeicher des Mikrocomputers gebracht. Betätigen Sie dazu folgende Tasten:

CTRL und K gleichzeitig drücken, dann nur X betätigen:

Das Anfangsmenü erscheint wieder auf dem Bildschirm.

Taste mit Ziffer 3 (als Menü 3 bezeichnet) drücken:

Auf dem Bildschirm erscheint folgende Darstellung:

0 = CRT , 1 = PRT , 2 = EPSON:

D

12

Es bedeuten:

- 0 = CRT Die Ausgabe des Kontaktplans erfolgt auf dem Bildschirm des Monitors.
- 1 = PRT Die Ausgabe des Kontaktplans erfolgt über einen Drucker.
- 2 = EPSON Die Ausgabe des Kontaktplans erfolgt über einen EPSON-Drucker.

Nach Einstieg in das Menü 3 muß noch die Zifferntaste 0 gedrückt werden, damit das eingegebene Programm als Kontaktplan auf dem Bildschirm des Monitors erscheint. Der **Kontaktplan** ist eine bestimmte Form der Programmdarstellung. Das Entwerfen und Lesen von Kontaktplänen lernen Sie noch in einem späteren Kapitel.

Durch Drücken der Tasten **CTRL** und **A** gelangen Sie wieder in das Anfangsmenü.

Taste mit Ziffer 2 (als Menü 2 bezeichnet) drücken:

Jetzt wird es ernst. Das eingegebene Programm wird gestartet. Schalter erzeugen die Zustände 1 oder 0 der Eingabesignale, deren **logische Verknüpfung** nach den eingegebenen Anweisungen vorgenommen wird. Die Verknüpfungsergebnisse erscheinen an den **programmierten Ausgängen**. Dort werden Leuchtdioden (Abkürzung: LED) angesteuert. Eine leuchtende LED entspricht dem Zustand 1, eine nicht leuchtende hat den Zustand 0. Durch Eingabe der verschiedenen Signalkombinationen läßt sich genau überprüfen, ob das Programm die Vorgaben gemäß der Aufgabenstellung erfüllt.

Bei Betätigung der Tasten **CTRL** und **S** und anschließend **CTRL** und **C** wird das laufende Programm abgebrochen, es erfolgt die Rückkehr in das Anfangsmenü.

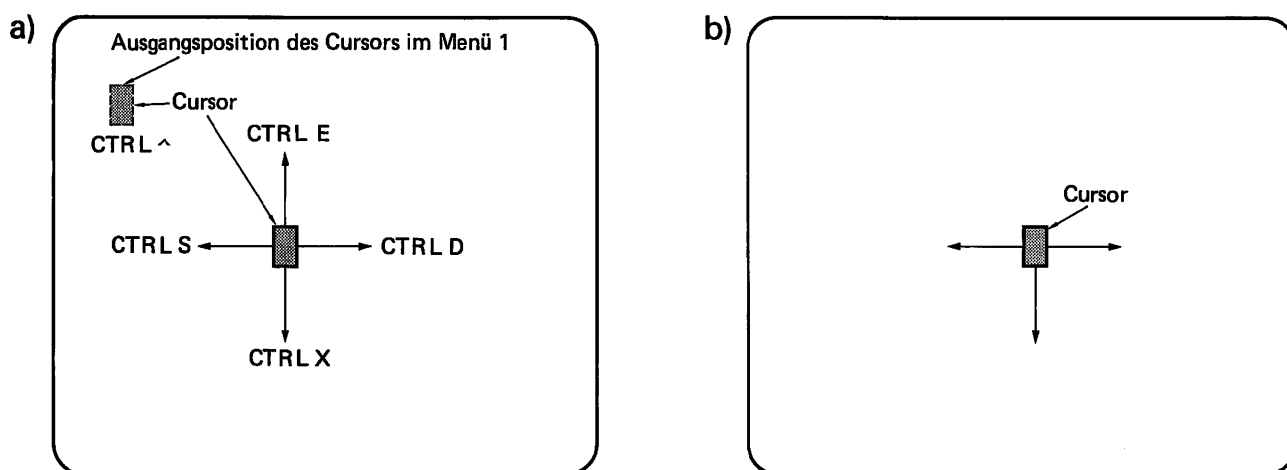
Die Menü-Punkte 4 bis 6 betreffen das Abspeichern des Programms auf die Kassette eines Rekorders, und mit Punkt 7 können wir Zeitglieder programmieren. Diese Punkte besprechen wir erst bei Bedarf.

Spielstunde mit dem NDR-Computer

In den letzten Abschnitten haben Sie schon viele theoretische Informationen über Ihr System erhalten. Es sind wichtige Informationen, ohne diese bringen Sie den Computer nicht zum „Laufen“. Bevor wir logische Verknüpfungen programmieren, sollen Sie erst einmal die Wirkungen einiger Befehle erproben. Sie erhalten von uns einige Anweisungen. Ihre eigene Experimentierfreudigkeit soll dadurch angeregt werden. Denken Sie sich weitere Versuche aus. Die Hardware wird nicht beschädigt, falls Sie einmal eine falsche Taste drücken. Lernen Sie Ihren Computer kennen.

Bild D13.1

Steuerung des Cursors auf dem Bildschirm des Monitors mit CTRL-Anweisungen (a). Steuerung des Cursors auf dem Bildschirm des Monitors mit „Pfeiltasten“ (b).



Beispiel D 13.1

Bewegen Sie bitte mit den Befehlen CTRL D, -S, -X und -E den Cursor über den Bildschirm. Stellen Sie bitte die Wirkung der Steuerbefehle fest und vergleichen Sie die Cursorbewegung wie im Bild D13.1a gezeigt.

Lösung:

Der Bildschirm zeigt uns das **Anfangsmenü**. Die Betätigung der Taste 1 führt in das Menü 1. Der Cursor steht jetzt auf dem Bildschirm am Anfang der obersten Zeile. Die Bezeichnung „!PE“ (Programmende) in der zweiten oder dritten Zeile braucht Sie nicht zu stören. Bei den folgenden Beispielen wird diese Bezeichnung überschrieben.

Beginnen Sie mit der gleichzeitigen Betätigung der Tasten CTRL und D. Bei jedem Tastendruck bewegt sich der Cursor um ein Zeichen nach **rechts**. Nun sollen Sie den Cursor zeilenweise in Richtung des **unteren** Bildschirmrands steuern. Die Bewegung entsteht beim gleichzeitigen Drücken der Tasten CTRL und X. Verschieben Sie anschließend den Cursor zeichenweise nach **links**. Hierfür stehen Ihnen die Tasten CTRL und S zur Verfügung. Zeilenweise nach **oben** wird schließlich der Cursor mit den Tasten CTRL und E gesteuert. Jede Tastenbetätigung bewegt den Cursor nur um ein Zeichen oder eine Zeile weiter.

Verschieben Sie nun bitte den Cursor nach eigener Vorstellung in beliebige Positionen. Zum Schluß soll der Cursor wieder in die Ausgangsposition gebracht werden (an den Anfang der obersten Zeile). Hierfür können Sie die besprochenen Steuerbefehle verwenden. In der Befehlszusammenstellung auf Tafel 3, Seite G 3 finden Sie den Befehl **CTRL A**. Bei gleichzeitiger Betätigung der beiden Tasten springt der Cursor in die Anfangsposition.

Auf der Tastatur finden Sie auch Tasten mit **Pfeilsymbolen**. Wir haben ja schon in der Tabelle D 10.2 angedeutet, daß auch damit eine Cursor-Steuerung möglich ist. Allerdings üben nur die Tasten mit den Zeichen **↓**, **→** und **←** die angegebene Funktion aus. Die Taste mit dem Zeichen **↑** ist zweckentfremdet, sie ist für eine andere Funktion vorgesehen.

D**14**

Beispiel D14.1

Der Cursor befindet sich in der Ausgangsstellung. Betätigen Sie die Tasten in der Reihenfolge **→**, **↓**, **←**. Tippen Sie zunächst die Tasten kurz an, halten Sie anschließend die Taste länger gedrückt. Überprüfen Sie die nachstehend beschriebenen Wirkungen (Bild D 13.1b).

Lösung:

a) Die Tasten werden nur kurz angetippt:

Mit Betätigung der Tasten **→** oder **←** bewegt sich der Cursor immer nur um ein Zeichen nach links oder rechts. Die Taste **↓** steuert den Cursor um eine Zeile vom oberen zum unteren Bildrand.

b) Die Tasten bleiben länger gedrückt:

Taste →: Der Cursor bewegt sich auf einer Zeile kontinuierlich von links nach rechts. Bei Erreichen des rechten Bildrands springt er eine Zeile tiefer und läuft auf dieser wieder über die Bildschirmbreite. Lassen Sie die Taste los, dann bleibt der Cursor an der Stelle stehen, die er zuletzt eingenommen hat.

Taste ←: Der Cursor bewegt sich jetzt kontinuierlich von rechts nach links auf einer Zeile über die Bildschirmbreite. Beim Erreichen des linken Bildrands springt er um eine Zeile nach oben an das rechte Zeilenende und läuft wieder von rechts nach links. Die Bewegung endet mit dem Loslassen der Taste.

Taste ↓: Bei gedrückter Taste bewegt sich der Cursor zeilenweise vom oberen Bildrand in Richtung des unteren Bildrands. Nach Erreichen des unteren Bildrands bleibt er trotz gedrückter Taste stehen.

Mit der Taste **←** können Sie den Cursor wieder zeilenweise in Richtung des oberen Bildrands, also in seine Ausgangsposition steuern. Schneller geht es mit dem Kommando **CTRL** und **A**.

Mit dem folgenden Versuch sollen Sie ermitteln, wieviel Zeichen eine Zeile auf dem Bildschirm aufnehmen kann, und wieviel Zeilen darstellbar sind.

Beispiel D 15.1

Der Cursor muß sich wieder in der Ausgangsposition befinden. Schreiben Sie die oberste Zeile voll mit der Ziffernfolge 0123....89012... . Bringen Sie anschließend den Cursor wieder in die Ausgangsposition. Er muß jetzt am linken Bildrand über der Ziffer 0 stehen. Mit CR wird der Cursor zeilenweise in Richtung des unteren Bildrands bewegt. Dabei sind die Zeilenanfänge mit der gleichen Ziffernfolge zu markieren. Die Ziffer „0“ am Zeilenanfang der obersten Zeile ist gleichzeitig die erste Ziffer für die Zeilenmarkierung.

Lösung:

Durch Zählen der Ziffern kommt man zu folgendem Ergebnis: Auf dem Bildschirm des Monitors können 24 Zeilen zu je 40 Zeichen geschrieben werden.

Mit vorstehendem Versuch haben Sie den Bildschirm vollgeschrieben. Sie müssen die Zeichen wieder löschen, damit andere Informationen Platz haben. Unsere Befehlsliste hält für Sie einen „**elektronischen Radiergummi**“ bereit. Betätigen Sie bitte CTRL und Z. Sie werden sehen, der Bildschirm ist sauber und der Cursor befindet sich wieder in der Ausgangsposition.

Sicher haben Sie schon festgestellt, daß man bei der Zeicheneingabe über die Tastatur doch leicht eine falsche Taste drückt. Wie man den Fehler im Text schnell beseitigt, zeigt Ihnen das nächste Beispiel. Da wir uns noch im Menü 1 befinden, können unmittelbar neue Zeichen eingetastet werden.

Beispiel D 15.2

Geben Sie mit den Buchstabentasten folgende Zeichenkette auf den Bildschirm:

a b c d e e f g i k

Die Zeichenfolge sollte aber dem Alphabet entsprechen. Es sind also zwei Fehler vorhanden. Damit die richtige Zeichenfolge entsteht, muß ein Buchstabe e entfernt werden und zwischen die Buchstaben g und i ist der Buchstabe h zu setzen. Die Korrektur wollen wir gemeinsam besprechen.

Lösung:

- a) Stellen Sie den Cursor über den zweiten Buchstaben „e“. Hat der Cursor die richtige Position eingenommen, dann drücken Sie bitte

die Tasten **CTRL** und **G**. Das unter dem Cursor befindliche Zeichen wird sofort gelöscht, und die rechts vom Cursor stehenden Zeichen rücken um eine Zeichenbreite nach links. Das „elektronische Radieren“ hat keine Lücke hinterlassen.

- b) Bewegen Sie den Cursor an die Stelle, an die der fehlende Buchstabe zu setzen ist. In unserem Fall muß der Cursor auf dem Zeichen „i“ stehen. Nun sind die Tasten **CTRL** und **V** gleichzeitig zu drücken. Alle rechts vom Cursor stehenden Zeichen einschließlich „i“ rutschen um ein Zeichen nach rechts. Unter dem Cursor ist eine Lücke entstanden. Drücken Sie nun die Taste mit dem fehlenden Buchstaben „h“. Die Zeichenfolge ...ghi... ist hergestellt.

Denken Sie sich selbst Beispiele aus. Die bisher beschriebenen Tastenfunktionen sollten Sie möglichst ohne im Text nachzusehen beherrschen.

Wir bleiben im Menü 1. Löschen Sie bitte den Bildschirm mit **CTRL** und **Z**. Ab sofort schreiben wir diese Befehlsart vereinfacht in der Form „**CTRL Z**“. Sie haben ja gelernt, daß damit zwei verschiedene Tasten bezeichnet sind, die man gleichzeitig drücken muß.

Im folgenden Beispiel benutzen wir den Bildschirm als Schreibtafel.

Beispiel D 16.1

Geben Sie bitte den Text von Bild D 16.1 ein. Wir beginnen in der obersten Zeile mit der angegebenen Ziffernfolge. Beachten Sie bitte bei der Benutzung der Buchstabentasten folgende Hinweise:

- Wird nur die Buchstabentaste gedrückt, dann erscheint auf dem Bildschirm der betreffende Buchstabe „klein“ geschrieben,
- beim gleichzeitigen Drücken von **SHIFT**- und **Buchstabentaste** erscheint auf dem Bildschirm der betreffende Buchstabe „groß“ geschrieben.
- Die Tasten „ä, ü, ö“ sind mit anderen Zeichen besetzt.

Bild D 16.1
Texteingabe im Menü 1, Beispiel
D 16.1. Anordnung des Textes auf
dem Bildschirm des Monitors.

	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9
1. Zeile																				
2. Zeile	STEUERUNGEN																			
3. Zeile																				
4. Zeile	verbindungs-										speicher-									
5. Zeile	programmiert																			
6. Zeile	fest und										frei und									
7. Zeile	austauschbar										austauschbar									

Vielleicht werden Sie sich bei der Texteingabe verschreiben. Zur Korrektur einer falschen Eingabe können Sie das bereits Gelernte anwenden. Den eingegebenen Text und die Textanordnung sollen Sie nun folgendermaßen ändern:

- Zwischen der oberen Zahlenreihe (Zeile 1) und dem Text „STEUERUNGEN“ (Zeile 2) ist eine Freizeile einzufügen.
- Die Ziffern in der obersten Zeile (Zeile 1) sind zu löschen.
- Alle Freizeilen im Text sind zu löschen, der Text soll in Zeile 1 beginnen.

Lösung:

- Stellen Sie den Cursor an den linken Rand von Zeile 2. Drücken Sie danach die Tasten **CTRL** und **N**. Sie werden folgendes beobachten: Der gesamte Text einschließlich „STEUERUNGEN“ rutscht eine Zeile nach unten. Zeile 2 wird frei. Sie haben eine Zwischenzeile eingefügt, die mit Zeichen ausgefüllt werden kann.
Sollten Sie einmal bei der Texteingabe eine Zeile vergessen haben, dann können Sie an der betreffenden Stelle mit „**CTRL N**“ eine Leerzeile einfügen und den vergessenen Text nachtragen.
- Sie haben beispielsweise einen falschen Text eingegeben, die Zeichen einer Zeile sind zu löschen. Setzen Sie den Cursor bitte in die Zeile, die Sie löschen wollen. Überprüfen Sie noch einmal, ob es auch die richtige Zeile ist. Danach ist der Befehl „**CTRL Y**“ auszuführen. Die Zeichen der markierten Zeile verschwinden (für immer). Der gesamte darunter stehende Text rutscht um eine Zeile nach oben, die leer gewordene Zeile ist wieder besetzt.
- Mit dem Befehl „**CTRL Y**“ können Sie auch die Leerzeilen löschen. Stellen Sie einfach den Cursor in eine Leerzeile. Nach Eingabe des Befehls rutscht der Text um eine Zeile nach oben, die Leerzeile wird ausgefüllt. Löschen Sie nach diesem Schema die Leerzeilen in unserem eingegebenen Text.

Bleiben Sie weiter im Menü 1. Löschen Sie bitte alle Zeichen auf dem Bildschirm. Wir wollen nun eine kleine Schaltung programmieren und testen. In diesem Beispiel geben wir Ihnen das Programm vor.

Beispiel D 17.1

Das Schema der Schaltung, deren Programm Sie nachfolgend in die SPS eingeben und testen sollen, zeigt uns das Bild D 18.1. E0 bis E3 sind die Eingangsvariablen, die die Werte 0 bzw. 1 annehmen können. Die Werte der Ausgangsvariablen A0 bis A3 sind von der Signalverknüpfung durch die Schaltung und den Signalzuständen der Eingangsvariablen abhängig. Da auf dem Bildschirm noch das Menü 1 dargestellt ist, können Sie das Programm direkt eintasten (Tabelle D 18.1).

Bild D18.1
Blockschaltung zum Programmier-
beispiel D17.1.

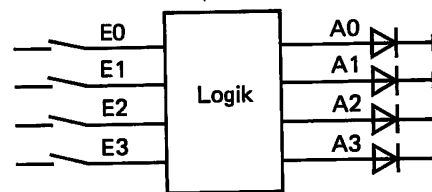
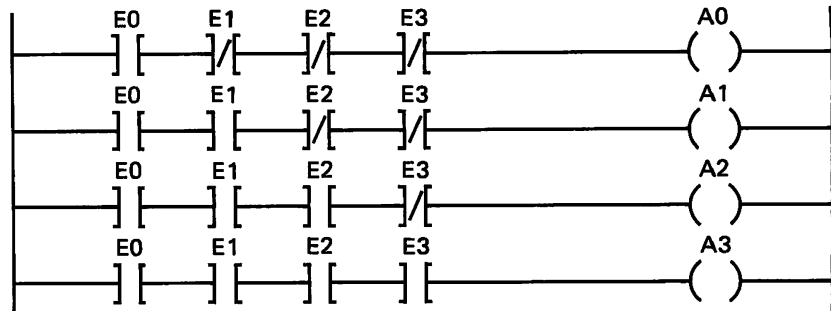


Bild D18.2
Kontaktplan zum Beispiel D17.1.



Mit der Tastenbetätigung „CTRL K“ und anschließend „X“ geben Sie das Programm in den Arbeitsspeicher. Auf dem Bildschirm erscheint das Anfangsmenü. Nun werden folgende Operationen durchgeführt:

- Das eingegebene Programm ist auf dem Bildschirm des Monitors als Kontaktplan auszugeben.
- Überprüfen Sie durch Betätigung der Eingabeschalter E0 bis E3 die Funktionstabelle der Schaltung in Tabelle D18.1.

In der Funktionstabelle sind alle möglichen Signalkombinationen der Eingangsvariablen aufgeführt. 4 Variablen ergeben 16 verschiedene Signalkombinationen. In Abhängigkeit dieser sind die Signalzustände der Ausgangsvariablen angegeben.

Tabelle D18.1: Funktionstabelle und Programm zu Beispiel D17.1

E3	E2	E1	E0	A0	A1	A2	A3
0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0
0	0	1	0	0	0	0	0
0	0	1	1	0	1	0	0
0	1	0	0	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	0	0	0	0	0
0	1	1	1	0	0	1	0
1	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
1	0	1	0	0	0	0	0
1	0	1	1	0	0	0	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	0	0
1	1	0	1	0	0	0	0
1	1	1	1	0	0	0	1

$\neg E0 \& \neg E1 \& \neg E2 \& \neg E3 = A0$
 $\neg E0 \& E1 \& \neg E2 \& \neg E3 = A1$
 $\neg E0 \& E1 \& E2 \& \neg E3 = A2$
 $\neg E0 \& E1 \& E2 \& E3 = A3$
 $\neg PE$

- a) **Bildschirmausgabe:** Da Sie sich im Anfangsmenü befinden, ist die Taste mit der Ziffer 3 zu drücken. Die Bildschirmausgabe wird durch Betätigung der Taste mit der Ziffer 0 gewählt. Auf dem Bildschirm muß der Kontaktplan nach Bild D18.2 erscheinen. Das Ende der Bildschirmausgabe und die Rückkehr ins Anfangsmenü bewirkt der Befehl „CTRL A“.
- b) Jetzt kommt der große Augenblick. Wir testen mit dem **NDR-Computer als SPS** das erste Programm. Betätigen Sie die Taste mit der Ziffer 2. Auf dem Bildschirm des Monitors sehen Sie weiterhin das Anfangsmenü, nur der Cursor ist nicht mehr vorhanden. Ein Zeichen dafür, daß die SPS sämtliche Anweisungen zyklisch abarbeitet. Geben Sie nun die **Signalkombinationen** laut Funktionstabelle in die Steuerung ein. Am Ausgang müssen die aufgeführten Signalkombinationen entstehen. Die wichtigsten Kombinationen sind nachstehend noch einmal aufgelistet.

E3	E2	E1	E0	A0	A1	A2	A3
0	0	0	1	1	0	0	0
0	0	1	1	0	1	0	0
0	1	1	1	0	0	1	0
1	1	1	1	0	0	0	1

Bei allen anderen Eingangskombinationen haben die Ausgangsvariablen den Wert 0. Mit **CTRL S** und **CTRL C** erfolgt der Abbruch des Programms und die Rückkehr zum Anfangsmenü.

Zusammenfassung

Ein Mikrocomputer übernimmt als zentrale Funktionseinheit die Steuerungsaufgaben in einem uC-System. Ein Mikrocomputer besteht im wesentlichen aus der CPU und aus dem **Speicher**. An dieser Stelle sei nochmals erwähnt, daß Ihr Mikrocomputersystem von unseren Darstellungen etwas abweichen kann.

Entweder haben Sie eine einzige Leiterplatte zur Verfügung, auf der die CPU zusammen mit dem Speicher untergebracht ist. Diese Baugruppe hat die Bezeichnung **SBC2** (Abkürzung für **Single Board Computer**), also für einen (Mikro-) Computer auf einer einzigen Leiterplatte.

Oder sowohl die CPU (zusammen mit einigen Hilfsschaltungen) als auch der Speicher sind jeweils auf einer eigenen Leiterplatte aufgebaut. Die CPU-Baugruppe hat in diesem Fall die Bezeichnung **CPU Z80**, wobei Z80 die Typenbezeichnung für eine weit verbreitete CPU ist. – Die Speicherbaugruppe hat die Bezeichnung **ROA64K**; sie ist für einen Speicher mit 64kByte Fassungsvermögen ausgelegt.

Das ist schon ein recht umfangreicher Speicher, der bei den Versuchen in diesem Lehrgang bei weitem nicht ausgenutzt wird. Aus diesem

Grunde muß die ROA64K-Karte — wenn Sie eine solche besitzen — auch nicht voll mit integrierten Speicherbausteinen (ICs) bestückt sein.

Eine weitere wichtige Baugruppe ist bei unserer SPS-Anlage die **IOE-Baugruppe** (IO-Einheit) zusammen mit der Experimentier-Ein-/Ausgabe. Die IO-Einheit besorgt die Weitergabe von Informationen von der Experimentier-Ein-/Ausgabe an die CPU.

Die IOE steckt zusammen mit anderen Mikrocomputerbaugruppen auf dem Bus. Unter einem Bus versteht man in der Mikroprozessortechnik ein Bündel von Datenleitungen, an das mehrere Rechnerbausteine gleichzeitig angeschlossen sind.

Unser NDR-Computer erhält seine Speisespannung von 5 V Gleichspannung entweder von der Baugruppe **POW 5 V** oder dem Netzgerät **NE1**. Das Netzgerät NE1 ist dann notwendig, wenn Sie Ihr Computersystem später weiter ausbauen wollen. Folgende Spannungen stehen dann zur Verfügung: +5 V, +/–12 V und +28 V.

Programmierverfahren für den NDR-Computer als SPS

Im letzten Abschnitt haben Sie die verschiedenen Programmierverfahren für industrielle SPS kennengelernt. Nur verhältnismäßig aufwendige und teure Programmiergeräte erlauben die Programmeingabe als Kontakt- und Funktionsplan oder als Anweisungsliste und in mathematischer Funktion.

Ihren NDR-Computer müssen Sie als Lehrmodell betrachten. Die Programmeingabe ist mit diesem System nur als **Anweisungsliste** oder als **mathematische Funktion** (eine Folge von Anweisungen) möglich. Zur Vorbereitung des Anwenderprogramms werden uns die Kontakt- oder Funktionsplandarstellungen eine große Hilfe sein. Sie erleichtern das Aufstellen der Anweisungsliste. Auch zur Analyse der gestellten Aufgabe werden die Pläne benötigt.

Wie Sie bereits wissen, kann das Programm im Menü 1 als Anweisungsliste gespeichert werden. Auf dem Bildschirm des Monitors können Sie die Eingaben kontrollieren und gegebenenfalls korrigieren. Nach dem Laden in den Programmspeicher und nach erneuter Ausgabe des Programms im Menü 1 erscheint die Darstellung auf dem Bildschirm als Folge von mathematischen Anweisungen. Die Anweisungsliste wurde in diese Darstellung umgewandelt.

Trotzdem werden wir oft die Anweisungsliste aufstellen. Betrachten Sie diese Maßnahme als notwendige Übung. In industriellen Steuerungen hat diese Form der Programmerstellung eine große Bedeutung. Unsere Anweisungsliste entspricht nicht den Formalismen nach DIN 19 239. Wir verwenden zur Aufstellung der Liste die mathematischen Symbole. Es ist auch üblich, einen **Mnemo-Code** zu benutzen. Eine Gegenüberstellung von mathematischen Symbolen und Code-Zeichen finden Sie in Lehrbrief 1 auf Tafel 4.

Die unmittelbare Programmeingabe erfolgt beim NDR-Computer als SPS mit Anweisungen.

Das gespeicherte Programm kann im Menü 3 als Kontaktplan auf dem Bildschirm ausgegeben werden. Damit ist ein Vergleich mit dem entworfenen und dem programmierten Plan möglich. Der wiedergegebene Kontaktplan entspricht den Anweisungen bzw. der Anweisungsliste.

In den nachfolgenden Programmierbeispielen werden wir meistens nach folgendem Schema vorgehen:

Die Aufgabenstellung erfolgt nach

- einer verbalen Beschreibung oder
- der Funktionstabelle oder
- der Schaltfunktion oder
- der Schaltung.

Zur Programmvorbereitung sind ein

- Kontaktplan oder
- Funktionsplan

anzufertigen. Das Programm ist

- als Anweisungsliste oder
- in Form von Anweisungen aufzustellen.

Beispiel D 22.1

Es soll kontrolliert werden, ob die Leuchtdioden auf der Experimentier-Ein-/Ausgabe in Ordnung sind. Eine UND-Verknüpfung von zwei Eingängen soll vier Leuchtdioden der Ausgabeseite ansprechen. Es sollen z. B. bei E0 und E1 = A0=A1=A2=A3 die entsprechenden Leuchtdioden aufleuchten. Testen Sie auf diese Art alle Leuchtdioden der Ein- und Ausgabeseite.

Lösung

Das Bild D 22.1 zeigt die Anweisungsfolge. Geben Sie die Anweisungen in Ihre SPS ein und starten Sie anschließend das Programm. Bei Betätigung der Eingabeschalter E0 UND E1 ODER E8 UND E9 müssen die vier Leuchtdioden A0, A1, A2 und A3 aufleuchten – bei Betätigung von E2 UND E3 ODER E10 UND E11 leuchten die Dioden A4, A5, A6 und A7 usw.

Die Ausgabe des Kontaktplans erfolgt über mehrere Bildschirmseiten. Die Tastenbetätigung CTRL C bzw. CTRL E ermöglicht das Scrollen der Bildschirmdarstellung nach oben oder nach unten (siehe Tafel 3 auf Seite G 3).



Bild D 22.1
Anweisungsfolge zu Beispiel
D 22.1.

Speicherfunktionen

Bevor wir Speicherfunktionen selbst programmieren, geben wir Ihnen einen Überblick des kommenden Lehrstoff. Aufgabe und Funktion des Speicherglieds haben Sie schon im Fachgebiet „Digitaltechnik“ kennengelernt. Schaltungen mit diesen Eigenschaften sind nun zu programmieren, also softwaremäßig nachzubilden. Wir sprechen nicht mehr von einem Speicherglied, sondern von einer **Speicherfunktion**. Die Speicherfunktion gehört zu den Grundfunktionen einer Programmiersprache für die SPS.

Zunächst werden wir für verschiedene RS-Speicherglieder den Kontaktplan entwickeln und das Programm in Form von Anweisungen aufstellen. Erweitert wird die Programmierung später noch durch die **Funktionsplandarstellung**. Man kann den Funktionsplan sehr leicht aus der vorgegebenen Schaltung entwickeln.

Nun zu unserem Lehrstoff.

Speicherschaltungen gab es schon, als noch Relais und Schütze in der Steuerungstechnik die schaltenden Elemente waren. Anwendung fanden Schaltungen mit „**Selbsthaltung**“. Die Arbeitsweise dieser Schaltungen bekommen Sie gleich erläutert. Wir werden ihre Funktion durch ein Programm nachbilden.

Größere Bedeutung haben Programme für die RS-Speicherfunktion. Das Programm muß die Eigenschaften des elektronischen RS-Speicherglieds nachbilden. In den meisten Fällen kommt die statische Steuerung des Kippvorgangs zur Anwendung. Es gibt aber auch Aufgaben, die nur mit dynamischer Ansteuerung lösbar sind. Den Kippvorgang lösen die sogenannten **Taktsignale** aus. Spezielle Programme, als „**Flankenmerker**“ bezeichnet, reagieren auf diese Taktsignale und erzeugen ein Steuersignal.

Das gespeicherte Signal steht am programmierten Ausgang zur Verfügung. Falls wir alle Ausgänge unseres Computers in Speicherfunktionen einbeziehen, können **16 RS-Speicherfunktionen** programmiert werden. Oft wird das gespeicherte Signal nur zur internen Signalverarbeitung benötigt. Für diese Anwendung ist die Benutzung eines **Signal-Ausgabespeichers** zur Bildung der Speicherfunktion zu schade. In umfangreichen Aufgaben werden die Ausgänge dringend für andere Signale benötigt.

Man kann auch mit einem Merker eine Speicherfunktion programmieren. Das im Merker gespeicherte Signal steht aber nur intern, also innerhalb der Schaltung, zur Verfügung. Der Merkerzustand kann genau wie der Zustand eines Ausgangs abgefragt und in eine Signalverknüpfung einbezogen werden. Beispielsweise könnte die Anweisung für eine UND-Funktion lauten:

```
!E3&A4&M8=A1
!PE
```

Selbsthalteschaltung als Speicherfunktion

Die in Schützenschaltungen übliche Schaltung für eine Speicherfunktion ist, wie bereits erwähnt, die **Selbsthalteschaltung** der Schütze. In Bild D 24.1 sehen Sie dafür zwei Beispiele. Betrachten wir zunächst die Stromlaufpläne. Zum Schütz A0, dessen Schaltzustand zu speichern ist, gehört der Schließer A0. Dieser liegt **parallel zum EIN-Taster E0**. Bei Betätigung des Tasters E0 spricht das Schütz an und die Kontakte des Schließers A0 werden geschlossen. Über diese fließt nunmehr der Haltestrom für die Schützspule. Nach dem Öffnen des Tasters E0 bleibt das Schütz erregt, der Zustand EIN ist gespeichert.

Durch Unterbrechen des Haltestroms wird das Schütz wieder in den Zustand AUS gesteuert. Hierfür sind zwei Schaltungsvarianten möglich.

Bei der Lösung mit **vorrangigem Ausschalten (Rücksetzdominanz)** ist der AUS-Taster E1 (ein Öffner) in Reihe zur Parallelschaltung der Kontakte A0 und E0 angeordnet (Bild D 24.1a). Werden E0 und E1 gleichzeitig betätigt, wird das Schütz in den Schaltzustand AUS gesteuert. Bei **vorrangigem Einschalten (Setzdominanz)** liegt der AUS-Taster E1 in Reihe mit Kontakt A0. Werden nun E0 und E1 gleichzeitig betätigt, wird das Schütz in den Schaltzustand EIN gesteuert. Mit Hilfe der Stromlaufpläne können Sie sich selbst die elektrische Funktion der beiden Schaltungen erarbeiten. Nach den Stromlaufplänen lassen sich auch unmittelbar die Kontaktpläne aufstellen.

Im Kontaktplan von Bild D 24.1a wird das Ergebnis der ODER-Verknüpfung von E0 mit A0 in den Merker M0 gesetzt. Merkerinhalt und

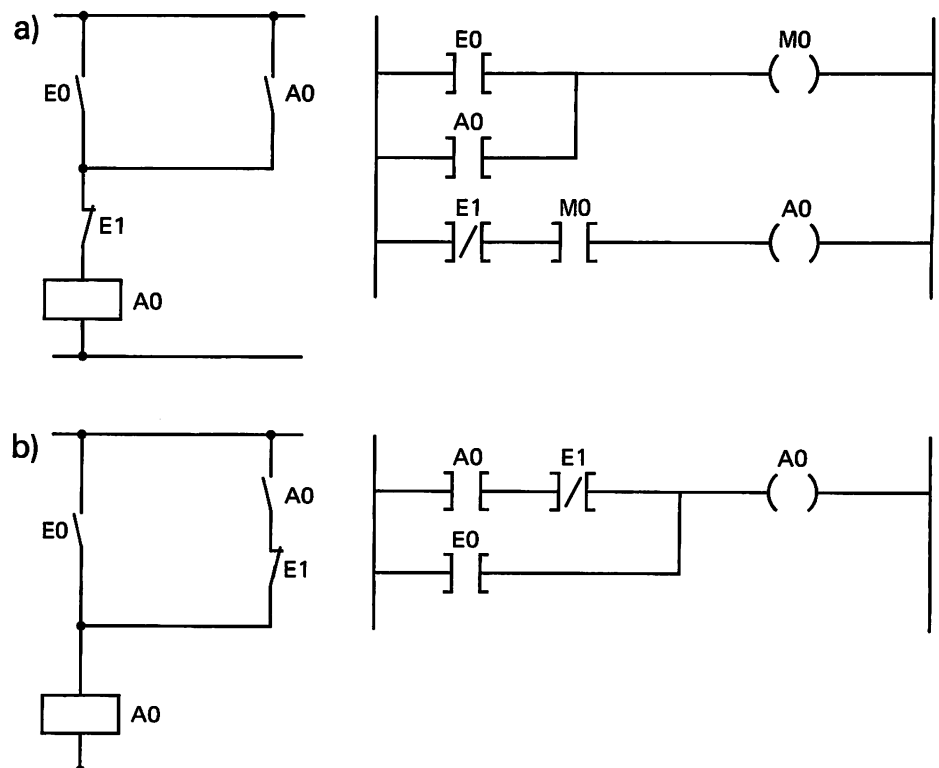


Bild D 24.1
Schaltungen mit speicherndem
Verhalten durch Selbsthaltung
a) Schaltung mit Rücksetz-
dominanz,
b) Schaltung mit Setzdominanz.

Zustand von E1 bestimmen den Zustand der Ausgangsvariablen A0. Der Stromlaufplan der Schaltung von Bild D 24.1b lässt sich direkt, d.h. ohne die Verwendung eines Merkers, als Kontaktplan wiedergeben.

Das Programm der Schaltung zu Bild D 24.1a lautet:

```
!A0/E0=M0
!NE1&M0=A0
!PE
```

Geben Sie bitte das Programm in Ihren Computer ein. Prüfen Sie anschließend mit Ihrem Ein-/Ausgabegerät die nachstehende Funktionstabelle.

Tabelle D25.1: Funktionstabelle der programmierten Selbsthalteschaltung nach Bild D24.1a. E0=EIN-Signal, E1=AUS-Signal.

E1	E0	A0	
0	0	0	
0	1	1	Speicher setzen
0	0	1	
1	0	0	Speicher rücksetzen
0	1	1	
1	1	0	

Die Zeile 6 der Funktionstabelle zeigt, daß das Rücksetzsignal dominiert. Ein 1-Signal an beiden Eingängen ergibt den Signalzustand 0 am Ausgang.

Programm der Schaltung zu Bild D 24.1b:

```
!A0&NE1/E0=A0
!PE
```

Testen Sie nun auch dieses Programm durch Überprüfung der folgenden Funktionstabelle.

Tabelle D25.2: Funktionstabelle der programmierten Selbsthalteschaltung nach Bild D24.1b

E1	E0	A0	
0	0	0	
0	1	1	Speicher setzen
0	0	1	
1	0	0	Speicher rücksetzen
0	0	0	
0	1	1	
1	1	1	

Die Zeile 7 der Funktionstabelle zeigt, daß bei dieser Schaltung die Setzbedingung dominiert. Obwohl beide Eingänge den Zustand 1 führen, bleibt der Setzzustand bestehen.

Nun kommen wir zu einer weiteren, umfangreichen Gruppe der Speicherfunktionen. Es sind die Programme, mit denen die Schalteigenschaften elektronischer Speicherglieder nachgebildet werden.

RS-Speicherfunktion

Zur Programmierung der Speicherfunktion und Bildung der Anweisung stehen uns die beiden Operationen S und R zur Verfügung (siehe Tafel 4, Lehrbrief 1).

S bedeutet: Speicherfunktion „Setzen“ bei Verknüpfungsergebnis 1, Verknüpfungsergebnis 0 hat keine Wirkung.

R bedeutet: Speicherfunktion „Rücksetzen“ bei Verknüpfungsergebnis 1, Verknüpfungsergebnis 0 hat keine Wirkung.

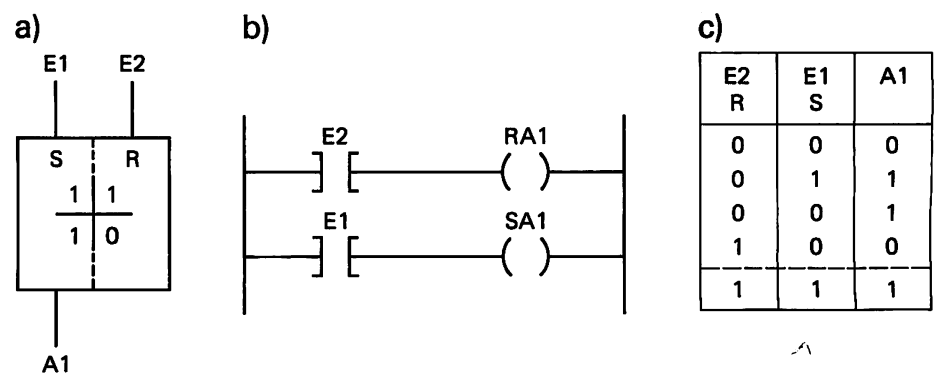
Beide Operationen können in Verbindung mit den Operanden A und M zur Anwendung kommen. Wir können also das gespeicherte Signal einem Ausgang, aber auch einem Merker zuweisen. Beide Anwendungen werden wir besprechen.

RS-Speicherfunktion mit Setz- oder Rücksetzdominanz

RS-Speicherfunktion mit Setzdominanz

Schaltzeichen, Kontaktplan und Funktionstabelle finden Sie in Bild D 26.1. Wie Sie sicher selbst bemerken, hat der Kontaktplan keinen Bezug zu einer bisher benutzten Schaltung oder zu einem bekannten Plan.

Bild D 26.1
RS-Speicherfunktion mit Setzdominanz.
a) Schaltzeichen,
b) Kontaktplan,
c) Funktionstabelle.



Das Lesen des Kontaktplans wird folgendermaßen vorgenommen:

Mit der Anweisung E1=1 wird dem Ausgang A1 der Zustand 1 zugewiesen. Diese Zuweisung wird symbolisiert durch SA1, die dem Setzen des Ausgangs A1 entspricht. Bei E1=0 und E2=0 bleibt der gesetzte Zustand bestehen.

Mit der Anweisung E2=1 erfolgt das Rücksetzen. Dem Ausgang A1 wird dadurch Zustand 0 zugewiesen. Dieser Vorgang ist symbolisiert durch die Zuweisung RA1, die dem Rücksetzen des Ausgangs A1 entspricht.

Die Speicherfunktion hat nur den Signalausgang A1, der negierte Ausgang A1* ist nicht vorhanden. Diese Auslegung gilt für alle RS-Speicherfunktionen. In unserer SPS können die Operanden die Parameter 0 bis 15 haben.

Nun wollen wir die Dominanz besprechen. Nach dem Schaltzeichen soll $A1=1$ sein, wenn **beide Eingänge** den Signalzustand 1 haben ($E1=E2=1$). Erinnern Sie sich bitte an die Arbeitsweise der SPS. Wie schon mehrfach erklärt, werden die Anweisungen nacheinander ausgeführt. Im Kontaktplan sind die Anweisungen in der richtigen Reihenfolge (von oben nach unten gelesen) eingetragen. Demnach lautet bei $E1=E2=1$ die erste Anweisung „Rücksetzen“. Diese kommt jedoch gar nicht zur Ausführung, da unmittelbar als nächste Anweisung das „Setzen“ folgt. Im momentanen Bearbeitungszyklus wird die Anweisung „Setzen“ ausgeführt. Erst im nachfolgenden Zyklus kann die Anweisung korrigiert werden, falls sich die Zustände der Eingangsvariablen geändert haben sollten. Allgemein gilt für alle RS-Speicherfunktionen:

Haben beide Eingänge der RS-Speicherfunktion **gleichzeitig** den Signalzustand 1, dann wird die Speicherfunktion vorrangig gesetzt oder vorrangig rückgesetzt.

Es gilt: Die **zuletzt** programmierte Anweisung wird von der SPS **vorrangig** bearbeitet.

Das Programm der RS-Speicherfunktion in Bild D 26.1 hat demnach Setzdominanz.

Das Programm der RS-Speicherfunktion mit **Setzdominanz** lautet:

!E2=RA1
!E1=SA1
!PE

Anweisungslisten für die RS-Speicherfunktion mit Setzdominanz:

Mnemo-Code	NDR-Computer
U E 2	!E2
R A 1	=RA1
U E 1	!E1
S A 1	=SA1
= BE	!PE

Sehen Sie sich nun in der Funktionstabelle D 28.1 nochmals die Zuordnung der Signalwerte an.

Tabelle D 28.1: Funktionstabelle der RS-Speicherfunktion mit Setzdominanz

E2	E1	A1
R	S	
0	0	0
0	1	1
0	0	1
1	0	0
0	0	0
1	1	1

Speicher setzen

Speicher rücksetzen

Setzdominanz

Im Gegensatz zum „Hardware“-RS-Speicherglied führt bei der „Software“-RS-Speicherfunktion der Signalzustand $E1=E2=1$ ($S=R=1$) zu keinem undefinierten Speicherzustand. Die programmierte Funktion hat Setz- oder Rücksetzdominanz.

Überprüfen Sie bitte mit Ihrer SPS, ob das Programm die in der Funktionstabelle vorgegebenen Werte liefert.

RS-Speicherfunktion mit Rücksetzdominanz

Nach den Ausführungen im letzten Abschnitt können Sie die RS-Speicherfunktion mit Rücksetzdominanz selbst programmieren.

Aufgabe D 28.1

RS-Speicherfunktion mit Rücksetzdominanz

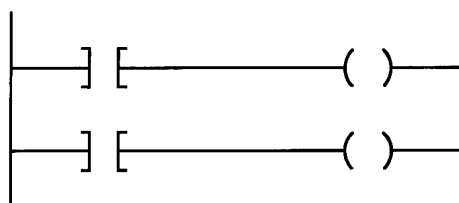
- a) Vervollständigen Sie in Bild D 28.1 den Kontaktplan und die Funktionstabelle.

Festlegungen: E1 Setzsignal
E2 Rücksetzsignal
A1 Ausgangssignal

Aufzustellen ist:

- b) Eine Anweisungsliste für den NDR-Computer und
c) das Programm in Form von Anweisungen.

a)



b)

E2	E1	A1
0	0	
0	1	
0	0	
1	0	
0	0	
1	1	

Bild D 28.1
RS-Speicherfunktion mit Rücksetzdominanz, Aufgabe D 28.1.

- d) Geben Sie Ihr Programm in den Computer ein. Die aufgestellte Funktionstabelle ist experimentell zu überprüfen.

Die Lösung dieser Aufgabe finden Sie auf Seite F16.

In den meisten Anwendungen werden die den Kippvorgang steuernden Signale nicht unmittelbar zur Verfügung stehen. Vielfach entstehen die Steuersignale durch **Signalverknüpfungen**. Ein einfaches Beispiel wollen wir gemeinsam lösen. Den aufgezeigten Lösungsweg können Sie sinngemäß auf alle derartigen Schaltungen anwenden.

Beispiel D 29.1

RS-Speicherfunktion mit Eingangsbeschaltung und Setzdominanz.

Unser Bild D 29.1a zeigt das Schaltzeichen des zu programmierenden RS-Speicherglieds. Das Setzsignal entsteht durch eine **UND-Verknüpfung** der Eingangsvariablen E1 und E2. Rückgesetzt wird das Speicherglied von den Variablen E3 oder E4. Programmieren Sie nun das Speicherglied.

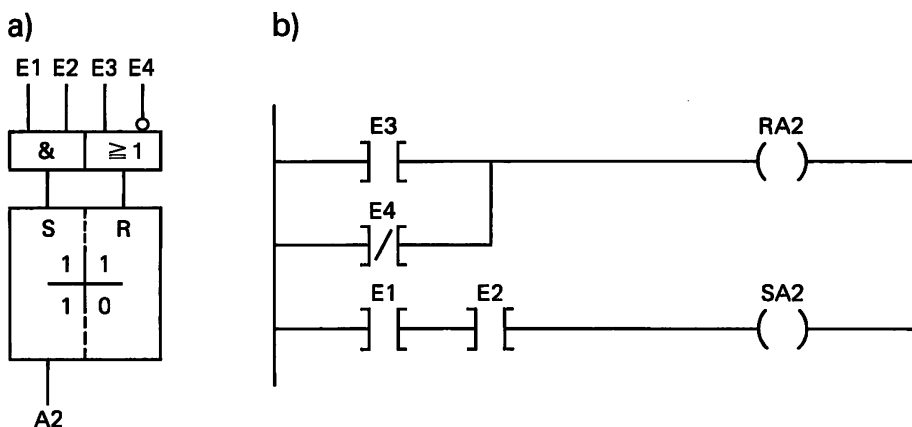


Bild D 29.1
RS-Speicherfunktion mit Eingangsbeschaltung und Setzdominanz.
a) Schaltung,
b) Kontaktplan.

Lösung:

Der Ausgang A2 kann unmittelbar von den Verknüpfungsergebnissen in den Setz- oder Rücksetzzustand gebracht werden. Den Kontaktplan hierfür zeigt das Bild D 29.1b. Die Setzbedingung kommt zur Herstellung der geforderten Setzdominanz **nach** der Rücksetzbedingung. Man kann die Verknüpfungsergebnisse der Eingangsvariablen auch in Merker setzen, falls sie noch für weitere Aufgaben benötigt werden.

Nach dem Kontaktplan können Sie das Programm durch folgende Anweisungen ausdrücken:

```
!E3/NE4=RA2
!E1&E2=SA2
!PE
```

Mit Merker:

$\neg E3/\neg E4 = M1$

$\neg M1 = RA2$

$\neg E1 \& E2 = M2$

$\neg M2 = SA2$

$\neg PE$

Es gibt Anwendungsfälle, bei denen die Dominanz nicht durch entsprechende Anordnung der Anweisungen im Programmablauf zum gewünschten Ziel führt. Schwierigkeiten entstehen nach der bisherigen Methode, wenn Speicherglieder gegeneinander zu verriegeln sind. Derartige **Verriegelungen** werden noch bei der Programmierung einfacher Schaltwerke besprochen. Die Dominanz ist in solchen Fällen direkt zu programmieren.

Wie man das machen kann, zeigt die Zusammenstellung in Bild D 30.1. Das Prinzip ist sehr einfach. Wir betrachten es anhand der RS-Speicherfunktion mit Rücksetzdominanz. Aus einer konjunktiven Verknüpfung von E1 mit dem negierten Rücksetzsignal E2 wird das Setzsignal gewonnen. Das Speicherglied wird nur bei $E1=1$ und $E2=0$ gesetzt. Bei $E2=1$ wird es unabhängig vom Zustand des Signals E1 zurückgesetzt somit dominiert Signal E2.

Nach dem gleichen Prinzip wird die Setzdominanz erzwungen. Die im Schaltzeichen gezeigte Verknüpfungsmethode kommt auch in den Kontaktplänen zum Ausdruck. Nach den Kontaktplänen sind die angegebenen Programme aufgestellt.

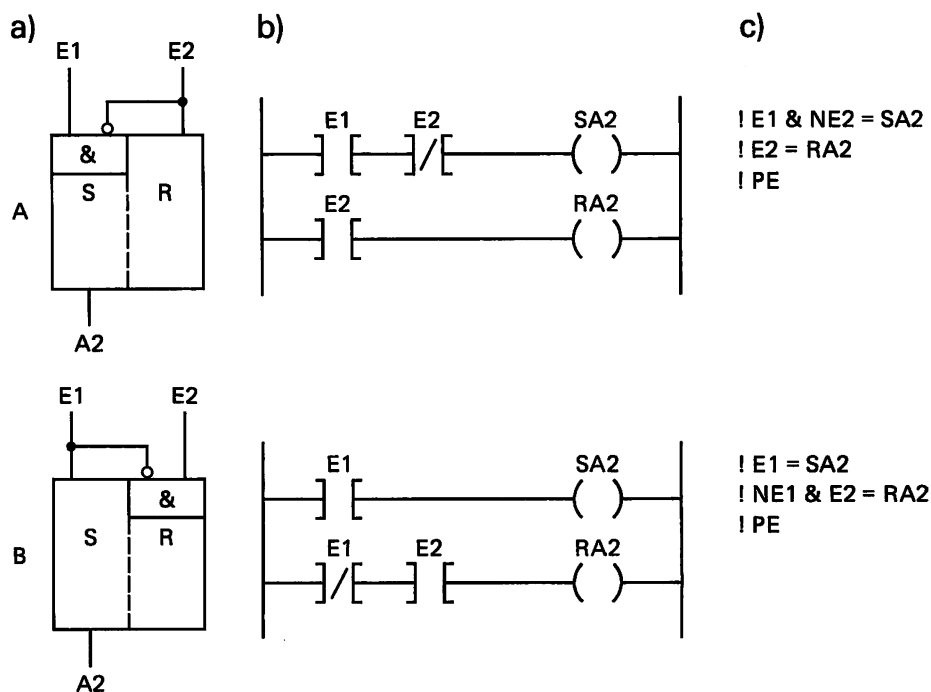


Bild D 30.1
RS-Speicherfunktion mit programmierter Dominanz
A Rücksetzdominanz,
B Setzdominanz.

a) Schaltzeichen,
b) Kontaktplan,
c) Programm.

Aufgabe D 30.1

Untersuchen Sie bitte experimentell, ob die Programme in Bild D 30.1 die Dominanzbedingungen erfüllen. Benutzen Sie die in den Pro-

grammen angegebenen Operanden. Für die nachstehend angegebenen Signalkombinationen der Eingangsvariablen sind die Ausgangsvariablen zu ermitteln.

a) RS-Speicherfunktion mit Rücksetzdominanz

E2	E1	A2
0	1	
0	0	
1	1	

b) RS-Speicherfunktion mit Setzdominanz

E2	E1	A2
0	1	
0	0	
1	1	

Die Lösung dieser Aufgabe finden Sie auf Seite F17.

Merker mit speicherndem Verhalten

Die RS-Speicherfunktionen mit Ausgangsspeicher als speicherndem Element haben wir nun erschöpfend besprochen. Wie bereits in der Einführung angedeutet, kann auch der Merker als Speicherfunktion programmiert werden.

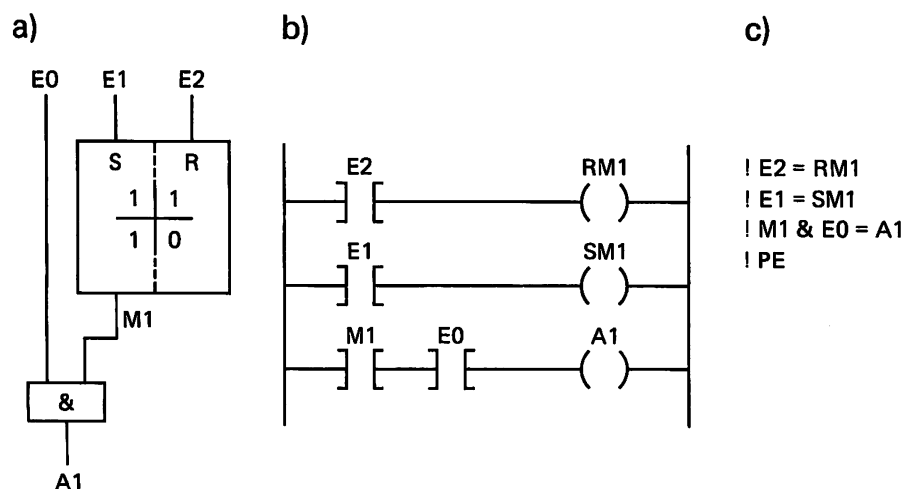


Bild D 31.1
Merker mit speicherndem Verhalten. Einfache RS-Speicherfunktion.
a) Schaltzeichen,
b) Kontaktplan,
c) Programm

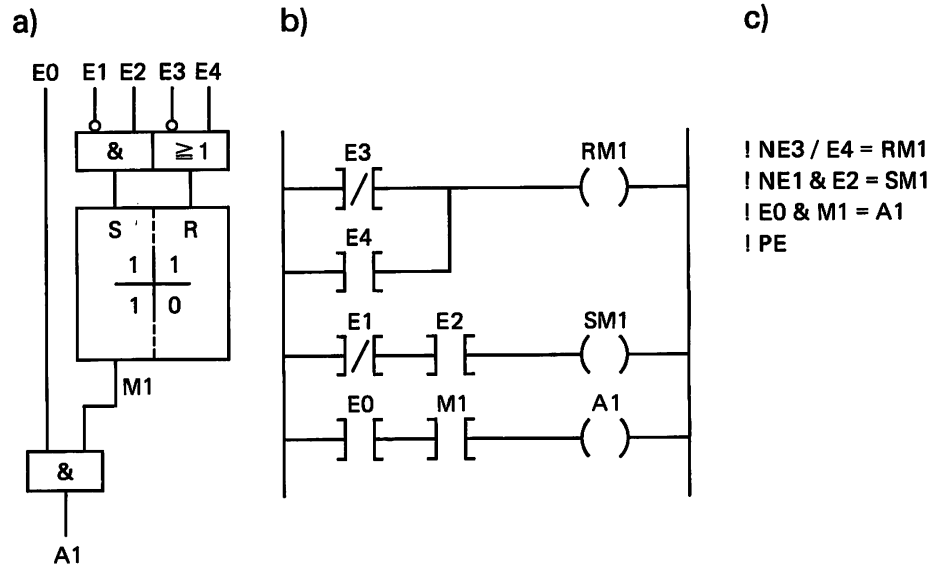


Bild D 32.1
Merker mit speicherndem Verhalten. RS-Speicherfunktion mit Eingangsbeschaltung.
a) Schaltzeichen,
b) Kontaktplan,
c) Programm.

Die bisher aufgestellten Programme haben auch für Merker Gültigkeit. In den Programmen ist der Operand A durch den Operand M zu ersetzen. Falls Sie den im Merker gespeicherten Zustand überprüfen wollen, muß der Merkerinhalt einem Ausgang zugewiesen werden. In Bild D 31.1 und D 32.1 sind einige Beispiele wiedergegeben. E0 ist als Freigabesignal anzusehen. Erst bei $E0=1$ wird das im Merker gespeicherte Signal auf den programmierten Ausgang geschaltet.

Verzögerungsfunktion

Wir betrachten als nächstes Schaltungen und Programme, mit denen der Signalsprung $0 \rightarrow 1$ oder $1 \rightarrow 0$ eines Schaltsignals verzögert zur Auswirkung kommt, d. h. die Signalsprünge werden mit einer einstellbaren Verzögerungszeit vom Eingang einer Schaltung auf deren Ausgang übertragen. Das Bild D 33.2 zeigt die Darstellung als Signal-Zeit-Plan. Dieses Zeitverhalten wollen wir durch ein Programm nachbilden.

Der Übergang vom Wert 0 zum Wert 1 der Variablen am Ausgang erfolgt nach einer Verzögerungszeit „ t_1 “. Dieser Vorgang wird auch als **Einschaltverzögerung** bezeichnet.

Der Übergang vom Wert 1 zum Wert 0 der Variablen am Ausgang erfolgt nach einer Verzögerungszeit „ t_2 “. Dieser Vorgang wird auch als **Ausschaltverzögerung** bezeichnet.

Die Verzögerungszeiten sind gewöhnlich zwischen 0 (keine Verzögerung) und einem Maximalwert stetig einstellbar. In der Schaltungstechnik gilt für Verzögerungsglieder das Schaltzeichen nach Bild D 33.1. Für t_1 und t_2 können auch die tatsächlichen Verzögerungszeiten eingetragen werden.

Verschiedentlich wird das Verzögerungsglied in Schaltungsbeschreibungen auch als **Zeitglied** bezeichnet. Durch Kombinationen von Verzögerungsgliedern mit Verknüpfungs- und Speichergliedern lassen sich alle in der Schaltungstechnik benötigten zeitabhängigen Schaltungen herstellen.

Die folgenden Abschnitte enthalten wichtige Programme mit Verzögerungsfunktionen, die wir als **Zeitfunktionen** bezeichnen wollen.

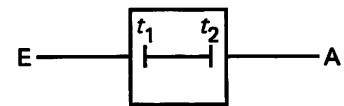


Bild D 33.1
Schaltzeichen des Verzögerungsglieds.

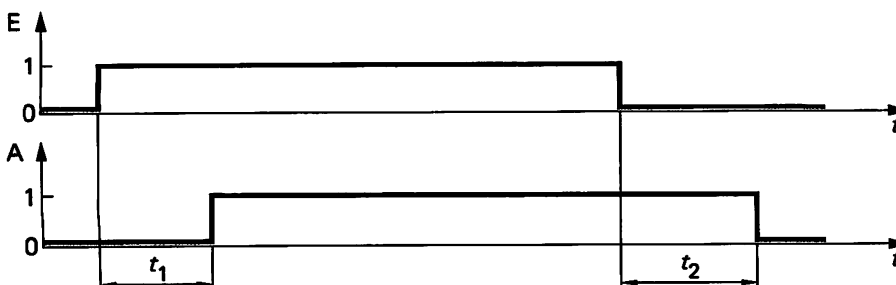


Bild D 33.2
Verzögerungsglied mit
 t_1 Einschaltverzögerung und
 t_2 Ausschaltverzögerung.

Kontaktplan und Programm einer Zeitfunktion

Den Kontaktplan, wie er auf dem Bildschirm des Monitors vom NDR-Computer als SPS erscheint, zeigt das Bild D 34.1. Aus dem Plan können Sie unmittelbar die Anweisungen des Programms ablesen.

Programm der Zeitfunktion:

```
!E0=T0
!T0=A0
!PE
```

D

34

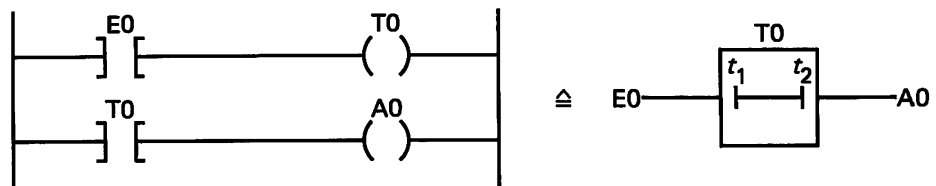


Bild D 34.1
Kontaktplan einer Zeitfunktion.

Mit „T0“ wird das programmierbare Zeitglied bezeichnet. Der NDR-Computer verfügt über 16 Zeitglieder „T0“ bis „T15“. Die vorstehende Darstellung können Sie wie folgt auslegen:

Nimmt die Eingangsvariable E0 den Zustand 1 an, dann beginnt die vorgewählte Zeit t_1 des Zeitglieds T0 abzulaufen. Nach Ablauf der eingestellten Verzögerungszeit nimmt das Zeitglied den Zustand 1 an. Es wird im Beispiel auf Ausgang A0 geschaltet. Bei $E0 \rightarrow 0$ schaltet Ausgang A0 nach Ablauf der Verzögerungszeit t_2 ebenfalls in Zustand 0.

Diese allgemeine Formulierung müssen wir nun auf die Programmierung der Zeitfunktion unserer SPS umsetzen.

Zeitfunktion der SPS des NDR-Computers

Im NDR-Computer werden die Zeitfunktionen als „Timer“ bezeichnet. Wie bereits angeführt, sind 16 Timer mit den Operanden T0 bis T15 vorhanden. Die getrennte Einstellung der Einschaltverzögerung (t_1) und Ausschaltverzögerung (t_2) erfolgt im Menü 7 Ihres NDR-Computers.

Als Beispiel wollen wir die Verzögerungszeiten für Timer T0 einstellen. Vollziehen Sie bitte die folgenden Schritte an Ihrem Gerät gleich nach.

Nach Inbetriebnahme Ihres Systems sehen Sie im Anfangsmenü die Bezeichnung „Timer“ angeführt. Nach Betätigung von Taste „7“ meldet sich auf dem Bildschirm des Monitors Timer T0 mit:

Timer 0 Einschalt-Verzögerung: 50

Der Wert 50 stellt eine fest vorprogrammierte Einschaltverzögerung für alle Timer (T0 bis T15) dar. Sie beträgt 50 Zeiteinheiten. Eine **Zeiteinheit**, Abkürzung ZE, hat den Wert von 20 ms. Somit beträgt die vorgegebene Einschaltverzögerung:

$$t_1 = 50 \text{ Zeiteinheiten} \times 20 \text{ ms pro Zeiteinheit} = 1000 \text{ ms} = 1 \text{ s}$$

Falls erforderlich, können Sie die vorgegebene Einschaltverzögerung von 50 Zeiteinheiten ändern. Sie brauchen nur den Wert 50 mit dem neuen Zahlenwert zu überschreiben.

Beispiel D 35.1

Wieviel Zeiteinheiten sind für eine Einschaltverzögerung von $t_1=6 \text{ s}$ erforderlich?

Lösung:

$$t_1 = 6 \text{ s} = 6000 \text{ ms}$$

$$\begin{aligned} \text{Anzahl der Zeiteinheiten} &= \frac{t_1}{20 \text{ ms/Zeiteinheit}} = \frac{6000 \text{ ms}}{20 \text{ ms/Zeiteinheit}} \\ &= 300 \text{ Zeiteinheiten (Abk. ZE)} \end{aligned}$$

Überschreiben Sie bitte die 50 mit 300. Die Verzögerungswirkung wird später noch in einem Beispiel getestet.

Nun ist noch die Ausschaltverzögerung zu programmieren. Betätigen Sie bitte die Taste CR. Auf dem Bildschirm des Monitors erscheint:

Timer 0 Einschaltverzögerung: 300

Timer 0 Ausschaltverzögerung: 0

Der Computer fragt nach dem Wert der Ausschaltverzögerung t_2 . Sie ist mit 0 Zeiteinheiten vorgegeben. Auch hier hat eine Zeiteinheit den Zeitwert von 20 ms. Bei vorgegebener Ausschaltverzögerung kann man die Zeiteinheiten wieder wie im obigen Beispiel berechnen.

Wir nehmen eine Ausschaltverzögerung von 2 s an. Diesem Wert entsprechen 100 Zeiteinheiten. Geben Sie den Wert ein. Damit ist Timer T0 programmiert.

Mit Betätigung der Taste CR werden die weiteren Timer in der Reihenfolge T1, T2... aufgerufen — die Zeitwerte können eingegeben werden. Ein Programm benötigt jedoch sehr selten alle Timer. Nach Programmierung der letzten Zeitfunktion und der Betätigung der Taste CR können Sie durch Betätigung der Tasten „CTRL“ und „C“ in das Anfangsmenü zurückkehren.

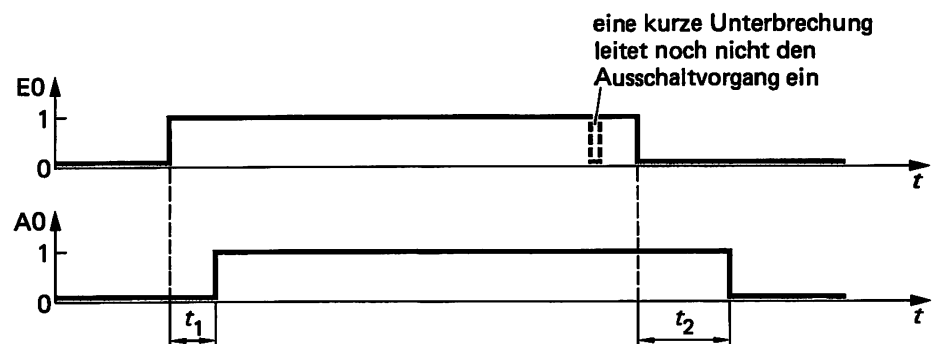
Vor dem Einstieg in die Praxis müssen wir Ihnen noch einige Hinweise geben. Betrachten Sie bitte das Signal-Zeit-Diagramm im Bild D 36.1.

Damit die Zeitfunktion ordnungsgemäß abläuft, muß das Eingangssignal $E0$ mindestens solange den Zustand 1 beibehalten, bis die Ausgangsvariable den Wert 1 angenommen hat. Wenn $E0$ vorher den Wert 0 annimmt, kommt die Zeitfunktion nicht zur Wirkung. Somit muß die Zeitdauer von $E0=1 > t_1$ sein. Weiterhin ist zu beachten, daß eine kurze Signalunterbrechung ($E0 = 1..0..1$) nicht sofort den Vorgang der Ausschaltverzögerung einleitet.

Nun wird es wieder Zeit, die Theorie in der Praxis zu erproben. An Hand einiger Beispiele werden wir Ihnen verschiedene Möglichkeiten der Anwendung zeigen.

D**36**

Bild D 36.1
Signal-Zeit-Diagramm einer Zeitfunktion.



Beispiel D 36.1

Es ist ein Programm zur Demonstration der Ein- und Ausschaltverzögerung mit Timer T0 aufzustellen. Die Verzögerungszeiten sollen jeweils 100 Zeiteinheiten (= 2 s) betragen. Das den Schaltvorgang auslösende Eingangssignal $E0$ ist am Ausgang $A0$ anzuzeigen, das verzögerte Signal soll am Ausgang $A1$ zur Verfügung stehen.

- Zeichnen Sie den Signal-Zeit-Plan und
- den Kontaktplan.
- Das Programm ist aufzustellen und in die SPS einzugeben.
- Testen Sie anschließend das Programm und vergleichen Sie die Ergebnisse mit Teil a).

Lösung:

- Den Signal-Zeit-Plan zeigt das Bild D 37.1. Gegenüber dem Eingangssignal $E0$ (dargestellt durch $A0$) ist das Ausgangssignal $A1$ um 100 Zeiteinheiten verzögert.
- Den Kontaktplan gibt das Bild D 37.2 wieder.

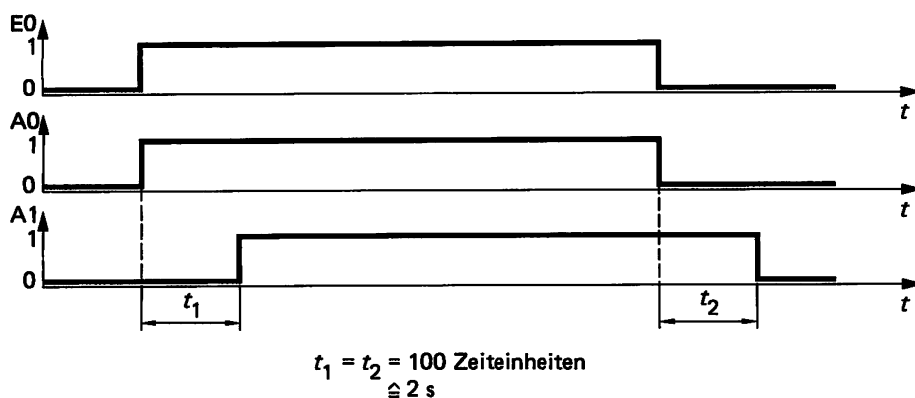


Bild D 37.1
Signal-Zeit-Diagramm zum Beispiel D 36.1.
 $t_1 = t_2 = 100 \text{ ZE} = 2 \text{ s}$

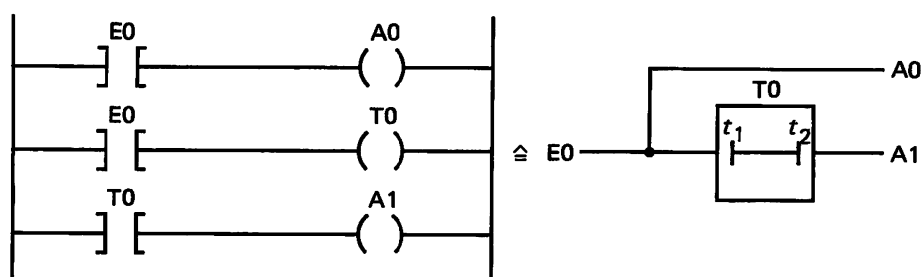


Bild D 37.2
Kontaktplan zum Beispiel D 36.1.

c) Anweisungen des Programms:

```
!E0=A0
!E0=T0
!T0=A1
!PE
```

Gehen Sie bei der Eingabe zunächst in das Menü 7. Dort ist der Timer T0 entsprechend der Beschreibung zu programmieren. Mit „CTRL“ und „C“ kommen Sie zurück in das Anfangsmenü. Nunmehr ist Menü 1 zu wählen und das Programm einzugeben. Nach dem Speichern des Programms befindet sich die SPS wieder im Anfangsmenü.

d) Der Test erfolgt im Menü 2. Die Betätigung des Eingabeschalters E0 wird unmittelbar mit Leuchtdiode A0 angezeigt. LED A1 schaltet zeitverzögert ein und aus. Beim Ausschalten von E0 schaltet sich die Leuchtdiode A1 mit derselben Zeitverzögerung aus.

Aufgabe D 37.1

Timer T1 ist so zu programmieren, daß er keine Einschaltverzögerung ($t_1=0$) und nur eine Ausschaltverzögerung von 200 Zeiteinheiten hat. E0 ist die den Schaltvorgang auslösende Eingangsvariable. Sie soll am Ausgang A0 sichtbar gemacht werden. Das verzögerte Signal ist dem Ausgang A1 zuzuordnen. Der Timer soll nach Bild D 38.1 einmal mit einem Eingangssignal längerer Zeitdauer und einmal mit einem kurzen Signalimpuls angesteuert werden. Es ist zu überprüfen, ob in beiden Fällen die programmierte Ausschaltverzögerung wirksam ist.

- Stellen Sie das Programm in Form von Anweisungen auf und
- testen Sie das Programm.

Die Lösung dieser Aufgabe finden Sie auf Seite F 32.

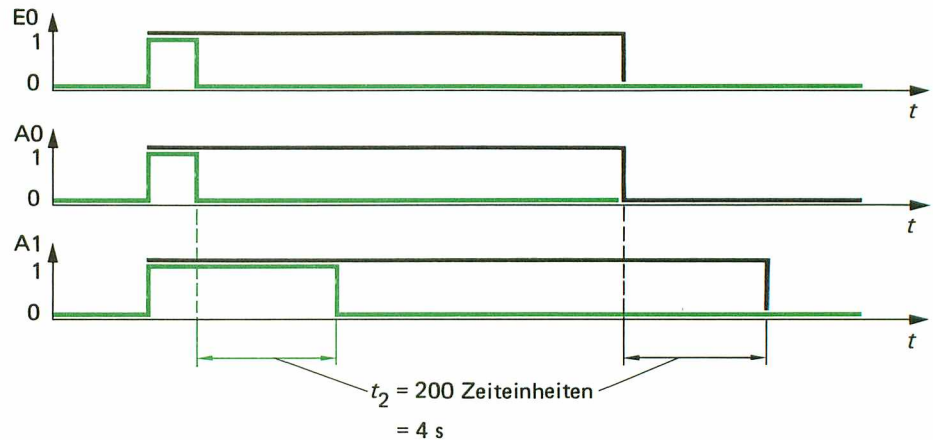


Bild D 38.1
Signal-Zeit-Diagramm zur Auf-
gabe D 37.1.

D

38

Aufgabe D 38.1

Timer T0 soll nur eine Einschaltverzögerung erhalten. Seine Ansteuerung erfolgt mit einem Verknüpfungsergebnis nach Bild D 38.2. Es ist eine Verzögerungszeit von $t_1 = 3,2 \text{ s}$ einzustellen.

- Ergänzen Sie bitte den in Bild D 38.3 vorgegebenen Signal-Zeit-Plan.
- Zeichnen Sie den Kontaktplan auf.
- Das Programm ist aufzustellen und wie im Beispiel D 36.1 erläutert in die SPS einzuspeichern. Überprüfen Sie, ob Ihr Kontaktplan b) mit dem des Programms (Menü 3) übereinstimmt.

(Hinweis: die Zeitfunktion kann unmittelbar mit dem Ergebnis der Verknüpfungsfunktion angesteuert werden).

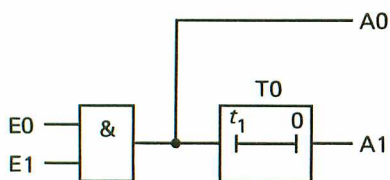


Bild D 38.2
Ansteuerung des Timers T0, Auf-
gabe D 38.1.

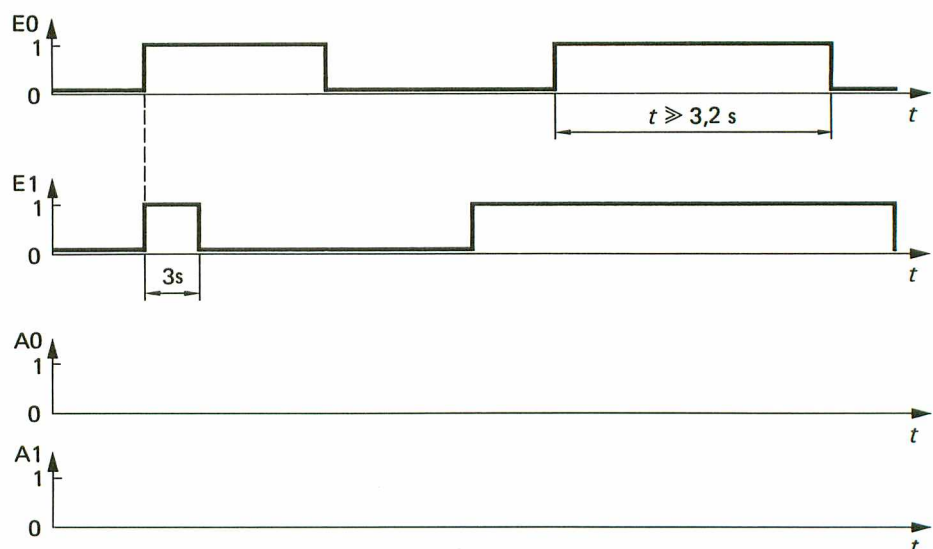


Bild D 38.3
Signal-Zeit-Diagramm zur Auf-
gabe D 38.1.

- Das Programm ist durch Betätigung von Schalter E0 und E1 zu testen. Dabei können Sie feststellen, ob Ihr Signal-Zeit-Plan mit den praktischen Ergebnissen übereinstimmt.

Die Lösung dieser Aufgabe finden Sie auf Seite F 33.

In den bisherigen Beispielen und Aufgaben wurden mit der Zeitfunktion nur Eingangssignale E oder deren Verknüpfungsergebnisse verzögert. Die Ansteuerung der Zeitfunktion kann aber auch durch Merker oder Speicherfunktionen erfolgen, wobei ein Merker auch nach der Zeitfunktion gesetzt werden kann.

!E0&E1=M0 Verknüpfungsergebnis in Merker M0 setzen
 !M0=A0 Merkerinhalt auf A0 ausgeben
 !M0=T0 bei M0=1 Zeitfunktion T0 starten
 !T0=A1 verzögertes Signal auf Ausgang A1 geben
 !PE

Eine wichtige Anwendung der Zeitfunktion wollen wir bereits in diesem Abschnitt besprechen. Zur Steuerung mancher Vorgänge sind **Taktsignale** erforderlich. Ein Anwendungsbeispiel, das Sie alle kennen, ist die Verkehrsampel. Das Weiterschalten der Verkehrsampelphasen erfolgt mit einem Taktsignal. Außerdem gibt es auch optische Warnmelder, die Lichtsignale mit einer bestimmten Blinkfrequenz erzeugen.

Die Hardware-Lösung zur Erzeugung von Taktsignalen kennen Sie bereits, es ist die **astabile Kippstufe**. Wir wollen nun ein Programm zur Erzeugung von Rechtecksignalen mit einer bestimmten Frequenz aufstellen.

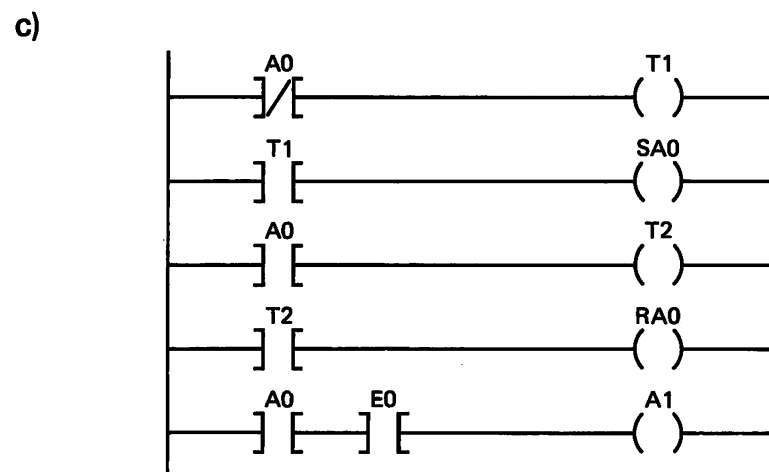
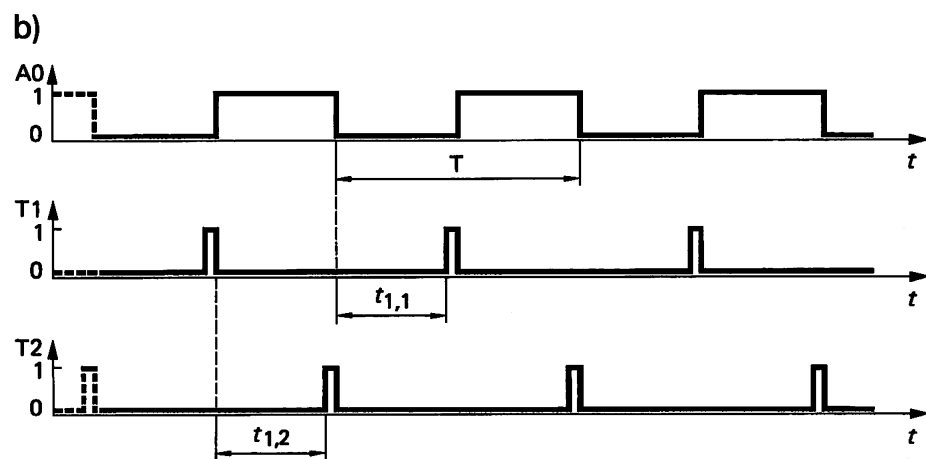
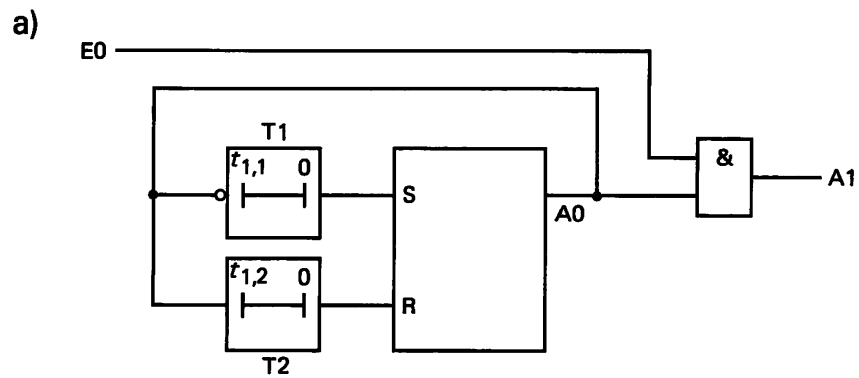
Taktgenerator mit zwei Zeitfunktionen

Benötigt werden zwei Zeit- und eine Speicherfunktion. Den Funktionsplan zeigt das Bild D 40.1a. Die Zeitfunktionen dürfen nur eine Einschaltverzögerung t_1 erhalten. Mit dieser wird die Kipp- bzw. Taktfrequenz festgelegt. Als Frequenz bezeichnet man die Zahl der vollständigen Rechteckschwingungen in einer Sekunde.

Betrachten wir zunächst den Funktionsplan. Der Ausgangszustand sei $A0=0$. Das 0-Signal wird invertiert und startet das Zeitglied T1. Erst nach der Zeit $t_{1,1}$ geht der Ausgang von T1 in Zustand 1 und setzt die Speicherfunktion in Zustand $A0=1$. Das 1-Signal schaltet den Ausgang von T1 sofort auf 0 und startet Zeitglied T2. Nach Ablauf der Verzögerungszeit $t_{1,2}$ nimmt der Ausgang von T2 den Wert 1 an. Dieser setzt die Speicherfunktion zurück auf $A0=0$. Damit geht auch der Ausgang von T2 wieder in Zustand 0. Der Anfangszustand ist wieder hergestellt und mit $A0=0$ wird die Zeitfunktion T1 erneut gestartet. Dieser Vorgang wiederholt sich periodisch. Bei $t_{1,1}=t_{1,2}$ haben 1-Signal und 0-Signal am Ausgang A0 die gleiche Zeitdauer.

Dauer einer Taktperiode T: $T = t_{1,1} + t_{1,2}$

Taktfrequenz f: $f = 1/T$



d)

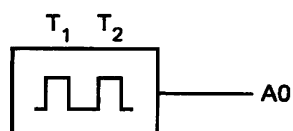


Bild D 40.1
Taktgenerator mit zwei Zeitfunktionen.
a) Funktionsplan,
b) Signal-Zeit-Diagramm,
c) Kontaktplan,
d) Schaltzeichen.

Wenn Sie die Zeit in der Einheit Sekunden einsetzen, erhalten Sie die Frequenz in der Einheit Hertz (in unserem Fall ist: Frequenz = Zahl der Rechteckschwingungen in einer Sekunde).

Aus dem Funktionsplan (Bild D 40.1a) ergibt sich der Kontaktplan in Bild D 40.1c und daraus das folgende Programm. An Ausgang A0 steht das Taktsignal immer zur Verfügung, mit E0=1 kann es auch auf den Ausgang A1 geschaltet werden. E0 wird als Freigabesignal bezeichnet.

Programm:

```
!NA0=T1
!T1=SA0
!A0=T2
!T2=RA0
!A0&E0=A1
!PE
```

Aufgabe D 41.1

Als weitere Aufgabe sollen Sie das eben gezeigte Programm prüfen. Die in diesem Programm aufgeführten Timer und Operanden sind zu verwenden. Beide Timer sollen die gleiche Einschaltverzögerung $t_1 = 1\text{ s}$ erhalten.

- Ermitteln Sie rechnerisch die zu erwartende Frequenz des Taktsignals.
- Überprüfen Sie bitte mir Ihrem NDR-Computer die Funktion des Programms.

Die Lösung finden Sie auf Seite F 33.

Zusammenfassung

Die Zeitfunktion der SPS wird als Timer bezeichnet. Der NDR-Computer hat 16 programmierbare Timer mit den Operanden T0 bis T15, deren Einschalt- und Ausschaltverzögerung getrennt einstellbar sind. Die Zeiteinheit der Timer beträgt 20 ms. Die Verzögerungszeit in ms ergibt sich aus dem Produkt Anzahl der Zeiteinheiten mal 20 ms. Die Anzahl der Zeiteinheiten ergibt sich aus dem Quotienten Verzögerungszeit durch 20 ms.

Bei einer Einschaltverzögerung erfolgt der Signalsprung des Ausgangssignals von 0 auf 1 nach einer Verzögerung gegenüber dem Signalsprung des Eingangssignals. Eine Ausschaltverzögerung liegt vor, wenn beim Ausgangssignal der Sprung von 1 auf 0 verzögert gegenüber dem Eingangssignal vor sich geht.

Mit einer Zeitfunktion lassen sich Eingangssignale oder deren Verknüpfungsergebnisse sowie Merker oder Speicherfunktionen verzögern. Zur Steuerung mancher Vorgänge sind Taktsignale erforder-

lich, die per Hardware häufig von einer astabilen Kippschaltung erzeugt werden. Taktsignale werden auch per Programm realisiert. Die Taktsignale haben in der Regel eine ganz bestimmte Frequenz. Als Frequenz bezeichnet man die Anzahl der Schwingungen in einer Sekunde.

Externe Speicherverfahren

Die von Ihnen geschriebenen und in den Arbeitsspeicher der SPS (RAM) eingegebenen Programme gehen beim Abschalten der Speisespannung verloren. Man könnte den Arbeitsspeicher mit batteriegepufferten CMOS-RAMs ausrüsten. Beim Abschalten der Speisespannung übernimmt dann sofort eine Batterie die Stromversorgung des Speichers. Im Ruhezustand ist die Stromaufnahme dieser Speicher vernachlässigbar klein. Leider sind diese Speicherbausteine mit großer Speicherkapazität noch sehr teuer. Auch sind sie für unsere Anwendung wenig geeignet, da sich mit der NDR-SPS die Speicherzellen nicht per Programm adressieren lassen.

Man verwendet daher zur Ablage von Programmen und Daten **nicht-flüchtige Speicherverfahren**. Bei diesen entsteht beim Abschalten oder beim Ausfall der Speisespannung kein Datenverlust. Die in Mikrocomputer-Systemen benutzten Massenspeicher verwenden durchweg magnetische Aufzeichnungsverfahren. Gebräuchliche Speichermedien sind Magnetbandkassette und Diskette (Floppy Disk). Beide gibt es in verschiedenen technischen Ausführungen.

Unsere SPS ist für den Betrieb mit Magnetbandkassette als Massenspeicher ausgelegt und vorbereitet. Wir besprechen deshalb nur dieses Speichermedium.

Kassettenaufzeichnung

Die Kassettenrekorder der Unterhaltungselektronik mit den dazu gehörenden Tonkassetten stellen die einfachste und billigste Form von Magnetbandkassetten speichern dar. Zur Aufzeichnung digitaler Informationen auf einem derartigen Magnetband gibt es verschiedene Verfahren. Die meisten speichern die Informationen bitseriell, d. h. die Bits eines Codeworts werden nacheinander, also seriell übertragen und aufgezeichnet.

Innerhalb der Mikroelektronik werden die (meist 8) Bits eines Codeworts immer gleichzeitig, also parallel transferiert. Zur Ausgabe in den Speicher (also zum Schreiben) ist deshalb eine **Parallel-Serien-Wandlung** erforderlich. Umgekehrt muß beim Lesen vom Speicher und der Eingabe in die Elektronik eine **Serien-Parallel-Wandlung** erfolgen.

Zur Wandlung gibt es zwei grundsätzliche Möglichkeiten. Man kann einmal den μC mit einem bestimmten Programm veranlassen, an einem I/O-Port die Daten seriell auszugeben bzw. aufzunehmen. Das Verfahren funktioniert nur bis zu einer Geschwindigkeit von ca. 1200 Bits/s zufriedenstellend. Die zweite Möglichkeit, von der in unserem Gerät Gebrauch gemacht wird, besteht darin, für diese Aufgabe einen speziellen Hardware-Baustein einzusetzen. Der serielle Datentransport wird mit einem **Taktsignal** synchronisiert. Von der Taktfrequenz ist die Zahl der pro Sekunde übertragenen Bits abhängig. Man bezeichnet diesen Wert als **Baudrate**.

Ein Aufzeichnungsverfahren, das unter der Bezeichnung „Kansas-City-Standard“ bekannt ist, stellt ein Bit vom Zustand 1 mit 8 Schwingungen der Frequenz 2 400 Hz und das 0-Bit mit 4 Schwingungen der Frequenz 1 200 Hz dar.

Unser Gerät verwendet ein Aufzeichnungsverfahren mit der Bezeichnung „Phase Encoding“ (Abkürzung PE) bzw. „Richtungstaktschrift“. Wir beschreiben das Verfahren nicht in allen Einzelheiten, da Sie nur das Grundprinzip verstehen sollen.

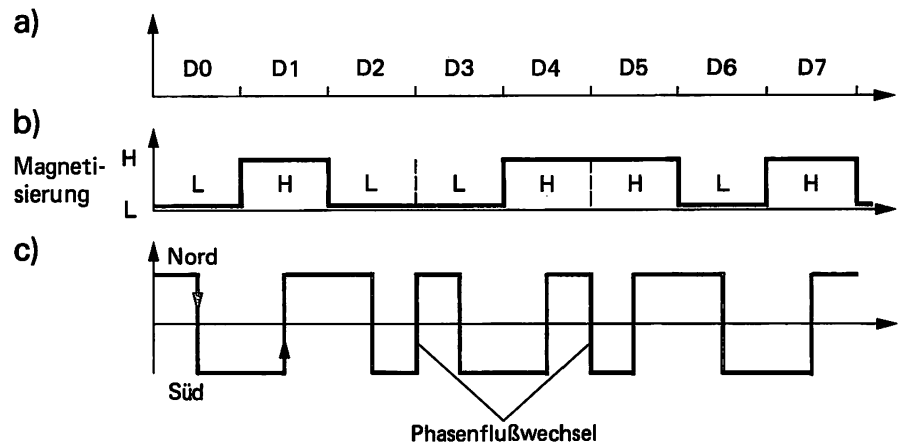


Bild D 44.1
Richtungstaktschrift.
a) Struktur eines 8-Bit Datenworts,
b) Datenwort D0...D7
01001101,
c) Richtungstaktschrift (Signalverlauf nach der Modulation).

Bei der Richtungstaktschrift wird zur Codierung eines Bits auf einem kleinen Bereich des Magnetbands (dieser Bereich wird als „Spurelement“ bezeichnet) eine Nord- und eine Südmagnetisierung vorgenommen (Bild D 44.1). Das bedeutet, daß in jedem Spurelement ein Wechsel der magnetischen Flußrichtung entsprechend dem binären Signalzustand (Binärzustand) erfolgt. Dem Flußwechsel von Nord nach Süd wird beispielsweise der Binärzustand L (0), dem umgekehrten Flußwechsel der Binärzustand H (1) zugeordnet. Wenn die Information aus alternierenden Bits besteht (Bitfolge ..L-H-L-...), ergeben sich die Flußwechsel automatisch (Bild D 44.1). Bei gleichbleibenden Bitzuständen (...L-L-L-... bzw. ...H-H-H-...) sind dagegen sogenannte **Phasenflußwechsel** einzuführen. Die Umformung der Bitdarstellung b) in den entsprechenden Flußverlauf c) ist per Software oder mit Hardware lösbar. Danach muß noch der unipolare Signalverlauf in einen bipolaren Signalverlauf umgewandelt und dem Schreibkopf des Kassettenrekorders zugeführt werden. Die bipolare Umwandlung ist erforderlich, weil sich auf einem Magnetband keine Gleichspannungen, sondern nur Wechselspannungen aufzeichnen lassen. Für die beschriebene Codierung finden Sie auch die Bezeichnung NRZ (No Return to Zero), weil bei gleichbleibend aufeinanderfolgenden L- bzw. H-Bits ebenfalls ein magnetischer Flußwechsel erfolgt.

Eingerahmt wird ein Datenwort mit **Start- und Stop-Bits**. Diese kennzeichnen Anfang und Ende eines Datenworts. Dieser Modulationsvorgang muß selbstverständlich mit einem Taktsignal synchronisiert werden. Die Frequenz des Taktsignals beträgt 1 200 Hz. Sie entspricht der Baudrate, d. h. es können in jeder Sekunde 1 200 Bits seriell übertragen werden.

Nun wollen wir noch den Lesevorgang betrachten. Das aus der Modulation resultierende Signalgemisch, das beim Lesen des Magnetbands entsteht und am Rekorderausgang abgenommen wird, muß wieder in den ursprünglichen Signalzustand gebracht werden. Dieser Vorgang wird als **Demodulation** bezeichnet. Die folgenden Erläuterungen beziehen sich auf das Bild D 45.1. Zunächst wird zu jeder Flanke des gelesenen Signals (a) mittels flankengetriggelter monostabiler Kippstufen ein Nadelimpuls (b) erzeugt. Diese Impulse triggern eine weitere monostabile Kippstufe, die Signale bestimmter Zeitdauer (d) erzeugt ($t=0,75T$ =Periodendauer der Frequenz von 1200 Hz). Mit der steigenden Flanke des Signals (d) werden die gelesenen Signale (a) abgetastet. Die Abtastung ergibt das ursprüngliche Signal.

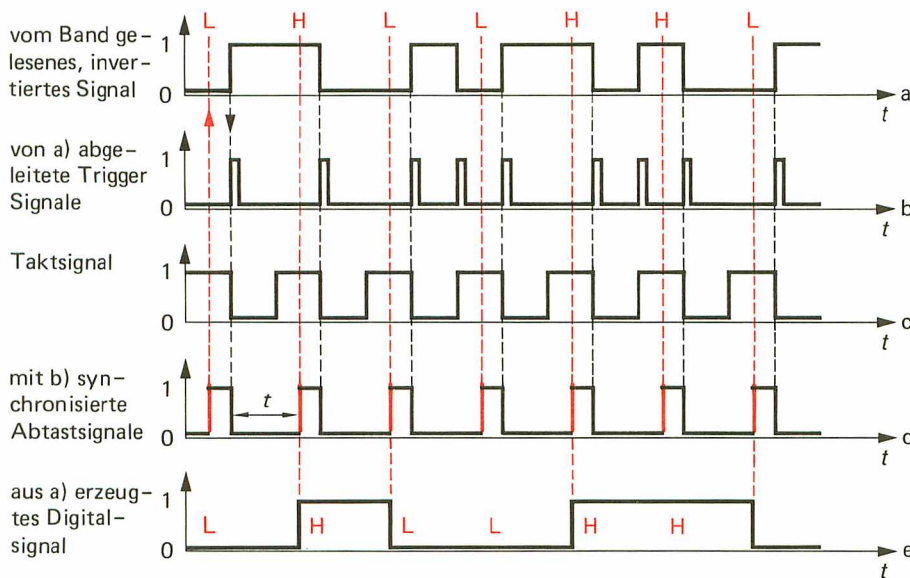


Bild D 45.1
Lesen der Richtungstaktschrift
(Demodulation).

Kassetteninterface CAS

Rufen Sie bitte das Anfangsmenü auf und lesen Sie dort die bisher noch nicht benutzten Anweisungen:

- 4 speichern
- 5 laden
- 6 vergleichen

Operation: „speichern“

1. Zählwerk des Rekorders auf einen bestimmten Anfangswert stellen und notieren.
2. Schalter S2 auf CAS in Stellung „speichern“ bringen (Schalter geschlossen).
3. Durch Betätigen der Taste „4“ Operation „speichern“ wählen.
4. Programmnamen eingeben: z.B. Dualzähler.

5. Kassettenrekorder auf Aufnahme schalten und starten (lassen Sie vor Ausführung des nächsten Schritts den Rekorder erst richtig anlaufen).

6. Start des Speichervorgangs: Taste „CR“ drücken.

Nach beendeter Übertragung, die je nach Programmlänge einige Sekunden dauert, meldet sich die SPS wieder mit dem Anfangsmenü. Schalten Sie den Rekorder ab und notieren Sie sich wieder den Zählerstand.

Achtung!

Die ersten Zentimeter am Bandanfang (oder Bandende) haben keine magnetisierbare Schicht. Falls Sie das Band vor der Aufnahme bis an den Anfang zurückgespult haben, darf erst nach Ablauf dieses „Vorspanns“ die Taste „CR“ gedrückt und mit der Speicherung begonnen werden.

Operation: „laden“

1. Bandspeicher mit Hilfe des Rekorderzählwerks (und der Notizen) auf Programmanfang stellen.
2. Schalter S2 auf CAS in Stellung „laden“ bringen (Schalter geöffnet).
3. Rekorder auf Wiedergabe schalten und starten.
4. Start des Ladevorgangs: Taste „5“ drücken.

Auch der Ladevorgang dauert selbst bei einem langen Programm nur wenige Sekunden. Zuerst erscheint auf dem Bildschirm der Programmname, z.B. Dualzähler. Danach meldet sich wieder das Anfangsmenü als Zeichen dafür, daß das Programm vom Bandspeicher in den Arbeitsspeicher (RAM) geladen wurde. Vergessen Sie nicht, den Rekorder auszuschalten.

Operation: „vergleichen“

1. Bandspeicher mit Hilfe des Rekorderzählwerks auf Programmanfang stellen.
2. Schalter S2 auf CAS in Stellung „laden“ bringen (Schalter geöffnet).
3. Rekorder auf Wiedergabe schalten und starten.
4. Start des Vergleichsvorgangs: Taste „6“ drücken.

Zuerst erscheint auf dem Bildschirm der Programmname und danach das Anfangsmenü. Damit wird gemeldet, daß das vom Bandspeicher gelesene Programm mit dem Programm im Arbeitsspeicher übereinstimmt. Der Speichervorgang wurde fehlerfrei ausgeführt. Der Rekorder kann ausgeschaltet werden.