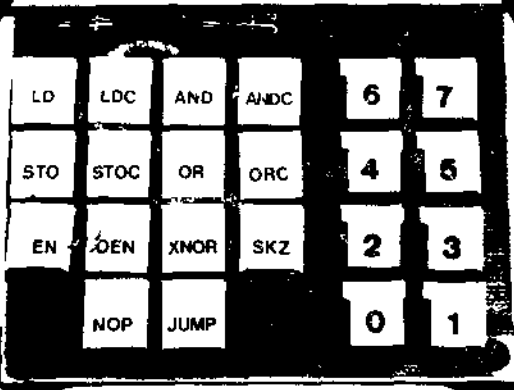
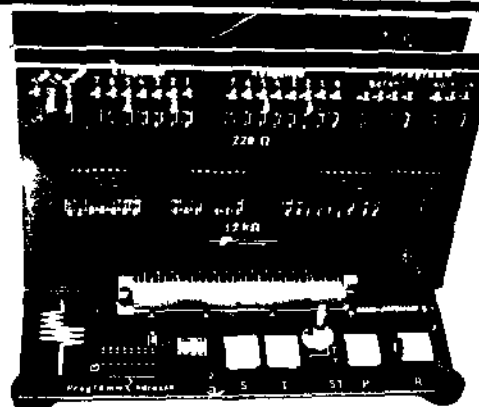
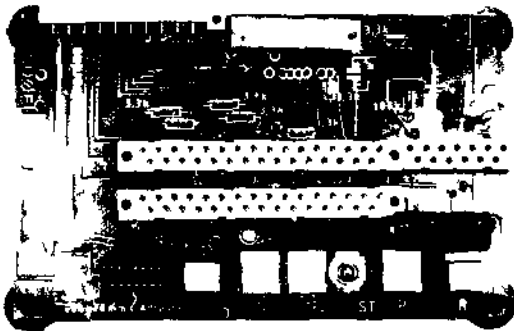
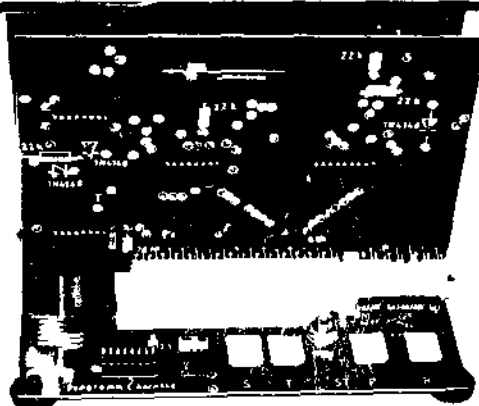
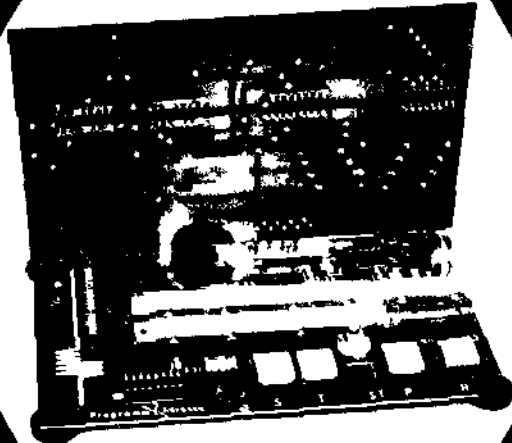


WDR

FAST ALLES UBER DEN
WDR-1-Bit-Computer



**VOLKER LUDWIG
KLAUS PASCHENDA U.A.**

Inhaltsverzeichnis

1.	Vorbemerkungen	1
2.	Der Aufbau des WDR-1-Bit-Computers (Hardware)	8
2.1.	Der prinzipielle Aufbau	8
2.2.	Die Funktionseinheiten	13
2.2.1.	Der Mikroprozessor MC 14500	13
2.2.2.	Der Eingabebaustein CD 4051	21
2.2.3.	Der Ausgabebaustein MC 14599	23
2.2.4.	Die Eingabetastatur	24
2.2.5.	Der Programmspeicher 2112	26
2.2.6.	Der Programmzähler CD 4029	28
2.2.7.	Die Anzeigeeinheit	30
2.2.8.	Die Takteinheit	31
2.2.9.	Die Stromversorgung	35
2.2.10.	Der 6-fach-Inverter CD 4049	36
2.2.11.	Das ODER-Glied	36
2.3	Das Zusammenwirken der Funktionseinheiten	37
3.	Der Zusammenbau	43
3.1.	Bausatz und Werkzeug	43
3.2.	Die Anzeigeplatine	45
3.3.	Die Grundplatine	50
3.4.	Die Speicherplatine	53
3.5.	Die Prozessorplatine	55
3.6.	Die Tastatur	56
4 .	Die Programmierung des WDR-1-Bit-Computers (Software)	59
4.1.	Der Befehlssatz des Mikroprozessors	59
4.2.	Das Eingeben von Programmen über DIL-Schalter	61
4.3.	Einfache Programme	63
4.3.1.	Programme zur Eingabe von Daten	63
4.3.2.	Programme zur Ausgabe von Daten	65
4.4.	Es wird bequemer: Die Tastatur	66
4.5.	Weitere Programme	66
4.5.1.	Logische Verknüpfungen	66
4.5.2.	Programmverzweigungen	70
4.5.3.	Programme zur Darstellung von Flipflops	73
4.5.4.	Programme zu arithmetischen Funktionen	82
4.5.5.	Simulationsprogramme	95
4.6.	Programme zur Steuerung angeschlossener Peripherie	102
4.6.1.	Motorsteuerung	104
4.6.2.	Sortieranlage	105
4.6.3.	Steuerung eines Roboters	108
4.6.4.	Ampelsteuerung	111
4.6.5.	Morsezeichengenerator	113
4.6.6.	Siebensegmentanzeige	114
4.6.7.	Bohrer	115
4.6.8.	Plotter	116
	Anhang	119

Volker Ludwig DDOEU
Klaus Paschenda DG5YCK

unter Mitarbeit von

Heinz Schepers
Hermann-Josef Terplane
Dr. Klaus Grannemann
Burkhard John
Hermann Komar
Ludwig Meinersen

FAST ALLES ÜBER DEN WDR-1-BIT-COMPUTER

Einige Hinweise in eigener Sache

Die Darstellungen von Schaltungen und Verfahren erfolgen ohne Rücksicht auf bestehende Patente, da sie nur für Amatuer- und Lehrzwecke gedacht sind. Ausdrücklich untersagt wird die gewerbliche Nutzung.

Sämtliche Rechte an Text und Bildern bleiben vorbehalten. Die Vervielfältigung einzelner Teile zu Lehrzwecken ist unter Verweis auf die Quelle erlaubt.

Trotz sorgfältiger Prüfung aller Darstellungen, Schaltungen und Verfahren sind Fehler nicht auszuschließen. Daher kann weder eine Garantie für Fehlerlosigkeit gegeben werden noch die juristische Verantwortung oder irgendeine Haftung übernommen werden.

Titelbild: Klaus Paschenda

Vertrieb: DATANorf
Am alten Bach 14a
4040 Neuss 21

Neuss und Recklinghausen 1986

1. VORBEMERKUNGEN

SO IST DER WDR-1-BIT-COMPUTER ENTSTANDEN

Nach dem Aufbau von über 50 Computern auf Z-80-Basis mit Teilnehmern an der Volkshochschule Düsseldorf durch Burkhard John und Volker Ludwig zeigten die Erfahrungen, daß weniger als 10% der Teilnehmer solche komplexen Computer verstanden hatten. Somit konnte man einen solchen Computerbau Schülern der Sekundarstufe 1 erst recht nicht zumuten.

Auf der Suche nach einem einfacheren System stieß man auf den Motorola-Baustein 14500. Der Computer, von seinen beiden Konstrukteuren nach dem Gesichtspunkt der didaktischen Vereinfachung konstruiert, wurde von Schülern erfolgreich aufgebaut und programmiert.

Nach Schlagzeilen in der Presse über den erfolgreichen Computerbau durch Schüler beauftragte der Redakteur des WDR-Schulfernsehens, Alexander Rimmelman, die Konstrukteure des Lerncomputers, Drehbücher über sechs Folgen mit dem Titel "Bit und Byte - Wir bauen einen Computer" zu schreiben.

SO IST DIESES BUCH ENTSTANDEN

Nach ersten Veröffentlichungen über den Lerncomputer wurde der Themenbereich als Diplomarbeit an der Fachhochschule Münster an Heinz Schepers und Hermann-Josef Terglane vergeben. Diese Arbeit wurde im Autorenteam zu dem vorliegenden Buch überarbeitet, wobei weitere Veröffentlichungen eingeflossen sind.

DAS AUTORENTEAM

Volker Ludwig und Klaus Paschenda
 Heinz Schepers und Hermann-Josef Terglane
 Dr. Klaus Grannemann
 Burkhard John
 Hermann Komar
 Ludwig Meinersen

EIN STÜCK GESCHICHTE

Den Beginn des modernen "Computer-Zeitalters" markieren die von Konrad Zuse gebaute Z1, Z2 und Z3, die von Howard Aiken konstruierte Rechenmaschine Mark I und die von J.P. Eckert und J.W. Manchley konzipierte Rechanlage ENIAC.

Die Z3 war der erste über Lochkarten gesteuerte elektromagnetische Rechner; Schaltelemente waren Relais. Die von Zuse initiierte Entwicklung konnte in Deutschland infolge des Krieges aber erst Jahre später wieder aufgegriffen werden. Anders in den USA, hier wurden die neuen Ideen sofort von der Industrie umgesetzt.

Der Bau der Mark I wurde beispielsweise durch den damaligen Chef von IBM, Thomas J. Watson, ermöglicht; er steckte eine Million Dollar in das Projekt.

ENIAC war die erste elektronische Großrechenanlage. Sie enthielt zirka 18000 Röhren und 1500 Relais, wog 30t, nahm einen ganzen Saal in Anspruch und verbrauchte 174kW. Ihre Programmierung erfolgte über Schalttafeln, die je nach Programm unterschiedlich verdrahtet wurden.

LINEARE LOCHSTREIFEN

Auch die Z3 wurde wie alle damaligen Rechner extern gesteuert. Ihr Programm war auf Lochstreifen codiert. Der Lochstreifen wurde abgespult, so daß der Anlage sämtliche Operationen nacheinander eingegeben wurden und sie das Programm linear abarbeiten konnte. Verzweigungen, logische Abfragen waren nicht möglich.

Moderne Computer weisen hingegen eine flexible Speicherprogrammierung auf. Dabei werden die einzelnen Befehle codiert, mit einer "Adresse" versehen und in einem Speicher abgelegt.

Die Codierung erfolgt üblicherweise in binärer Form. Der Rechner kann nun die Anweisungen linear abarbeiten; er kann aber auch logische Entscheidungen, eine Auswahl treffen, wenn bestimmte Bedingungen erfüllt oder nicht erfüllt sind. Einzelne Programmteile können so übersprungen, ausgelassen werden, im Programm können Verzweigungen enthalten sein.

Die grundlegenden Kennzeichen speicher-programmierter Rechner, die der Mathematiker von Neumann als erster in seinem Entwurf eines Universalrechners, der "Princeton-Maschine" formulierte, wollen wir am Beispiel dieser Maschine erläutern.

WIRKSAME WERKE

Der grundlegende Aufbau der meisten heutigen Rechenanlagen entspricht dem Konzept der Princeton-Maschine. Sie läßt sich durch folgende Beschreibung kennzeichnen:

Die Von-Neumann-Maschine besteht prinzipiell aus fünf Funktionseinheiten, sogenannten "Werken" oder "Einheiten", und zwar aus dem

- Rechen- und Logikwerk
- dem Leitwerk
- dem Speicherwerk
- dem Eingabewerk und
- dem Ausgabewerk.

Die ersten zwei Werke faßt man unter dem Begriff "Zentraleinheit" zusammen. Das Rechen- und Logikwerk führt arithmetische und logische Operationen durch. Das Leitwerk, auch Steuerwerk genannt, steuert den Austausch von Daten und Befehlen zwischen dem Rechen- und Logikwerk und den anderen Funktionseinheiten, es sichert einen geordneten Programmablauf. Wie dies im einzelnen abläuft, werden wir an späterer Stelle an unserem Prozessor beispielhaft erklären.

Die Übertragungswege, über die Daten zwischen den Funktionseinheiten und der Zentraleinheit ausgetauscht werden, nennt man Busse.

Alle Daten und Befehle, die von der Maschine verarbeitet werden sollen, müssen binär codiert sein. Sie werden als Folge von Nullen und Einsen dargestellt, weil sich dies physikalisch leicht realisieren läßt.

ZAHLREICHE ZELLEN

Im Unterschied zu Rechnern mit externer Programmsteuerung werden hier das Programm und die Daten in einem gemeinsamen Speicher abgelegt, in sogenannten "Zellen". Damit jederzeit auf die Zelleninhalte zugegriffen werden kann, wird jeder einzelnen Zelle eine bestimmte "Adresse", eine Nummer, zugeordnet. Die Zellen werden dazu fortlaufend durchnummeriert.

Ein solcher Speicher ist mit einem Regal vergleichbar, das eine gewisse Anzahl von Schließfächern enthält. Jedes Fach ist durch eine im Dualsystem dargestellte Zahl eindeutig gekennzeichnet. Ist beispielsweise die Zahl 0011 im Fach mit der Adresse 0010 gespeichert, kann der Speicherinhalt jederzeit abgerufen werden. Adresse und Inhalt einer Speicherzelle sind also zu unterscheiden.

Da die Befehle eines Programmes immer in aufeinanderfolgenden Zellen gespeichert werden, muß nur die Adresse des ersten Befehls bekannt sein, um das Programm starten und abzuarbeiten zu können. Die Adresse des nächsten Befehls liefert ein Programmzähler. Er erhöht die alte Adresse jeweils um Eins. Dies entspricht einem linearen Programmablauf.

Diese sequentielle Abarbeitung eines Programmes kann durch "Sprungbefehle" durchbrochen werden. Es gibt bedingte und unbedingte Sprungbefehle. Ein bedingter Sprung wird nur dann ausgeführt, wenn eine bestimmte Bedingung erfüllt ist. Andernfalls zählt der Programmzähler wie üblich weiter. Liegt hingegen ein unbedingter Sprung vor, wird der Programmzähler automatisch mit der neuen Adresse geladen.

Diese Eigenschaften, die für die Von-Neumannsche Struktur speicher-programmierter Rechner charakteristisch sind, werden von dem 1-Bit-Computer alle realisiert.

AUS EINER INNOVATION WIRD EINE NEUE INDUSTRIELLE REVOLUTION

Die zweite industrielle Revolution geht auf die Erfindung des Transistors im Jahre 1948 zurück. Dieses Bauelement hat nach und nach die Elektronenröhre ersetzt. Der Ablöseprozess verlief zunächst eher langsam. Erst mit dem Übergang zum Silizium als Grundmaterial begann eine explosionsartige Entwicklung, wie sie in der Technik bisher ohne Beispiel ist. Etwa um 1960 wurde nämlich die Möglichkeit erkannt, daß man, statt einzelne Transistoren herzustellen und zu verdrahten, auch auf einem kleinen Silizium-Chip mehrere mikroskopisch kleine Komponenten zusammenfassen ("integrieren") kann. Dies war der Beginn der Mikroelektronik.

Man vermutet, daß die Mikroelektronik den Motor zur dritten industriellen Revolution darstellt. Es gibt kaum einen Bereich, der nicht von der Entwicklung der Mikroelektronik beeinflusst wird. Von der Kommunikations- und Datentechnik bis zur Unterhaltungselektronik, von der Prozeßautomatisierung bis zur Medizin- und Verfahrenstechnik, Überall wird der technische Fortschritt durch die Mikroelektronik geprägt.

Ob nun zweite oder dritte industrielle Revolution, offensichtlich vollzieht sich momentan ein sprunghafter Wandel in Industrie, Dienstleistungsgewerbe und Handel. Treibende Kraft dieses Wandels ist die Mikroelektronik mit den damit verbundenen neuen Techniken und Produkten.

Weil dieser Wandel einschneidende Auswirkungen auf das berufliche, das gesellschaftliche und private Leben aller, insbesondere aber der jungen Menschen haben wird, muß auch die Schule auf diese Herausforderung reagieren.

DER NEUE BILDUNGSaufTRAG

Es gehört heute "zum Bildungsauftrag der Schule, allen Schülern frühzeitig - nach Möglichkeit bereits in der Sekundarstufe I - entsprechende Grundlagen in geeigneter Form zu vermitteln."

Über die konkrete Ausgestaltung, die geeignete Form, dieses Bildungsauftrages liegen sich die Pädagogen und Didaktiker noch im Clinch, im Computer-Clinch.

Schaut man sich in den einzelnen Bundesländern um, zeigt sich, daß es nur für die Sekundarstufe II in allen Bundesländern ein eigenständiges Unterrichtsfach "Informatik" gibt. Über eine Grundbildung Informatik in der Sekundarstufe I, die immerhin von ca. 70% aller Schüler besucht wird, denken die meisten Kultusminister noch nach. Ausnahme sind die Länder Bayern und Baden-Württemberg sowie Berlin, in denen es schon fixierte Vorstellungen gibt.

UNSER RAHMENKONZEPT

Gegenstand unseres Konzeptes ist die Einführung einer informationstechnischen Bildung für alle Schüler und Jugendlichen. Analog zu unserem Schul- bzw. Bildungssystem ist es gegliedert in :

- * eine informationstechnische Grundbildung für alle Schüler
- * eine vertiefende informationstechnische Bildung in Form der Informatik
- * eine berufsbezogene informationstechnische Bildung.

In der Sekundarstufe I soll - entsprechend dieser Empfehlung - die informationstechnische Grundbildung im Rahmen bestehender Unterrichtsfächer vermittelt werden. Infrage kommen dabei solche Fächer, "die induktiven, deduktiven und statistischen Methoden Vorrang einräumen und sich formalisierter Verfahren und Darstellungen bedienen."

Lange Rede, kurze Interpretation: In erster Linie sind damit wohl Physik, Mathematik und Technik gemeint.

Ein eigenständiges Fach Informatik bleibt der gymnasialen Oberstufe vorbehalten, was nicht ausschließt, "daß Informatikangebote auch schon in den Wahlpflichtunterricht des Sekundarbereichs I aufgenommen werden."

Um dies für die Sekundarstufe I ansatzweise zu lösen, werfen wir einen Blick in die Richtlinien für den Informatikunterricht der gymnasialen Oberstufe (Richtlinien für die gymnasiale Oberstufe in Nordrhein-Westfalen, Informatik, Hrsg.: Der Kultusminister des Landes NRW, Köln 1981, S. 29). Dort finden wir die folgenden vier Lernbereiche:

- Lernbereich I:
Algorithmik
- Lernbereich II:
Daten und Datenstrukturen
- Lernbereich III:
Hard- und Softwaresysteme
- Lernbereich IV:
Realisierung, Probleme und Auswirkungen der praktischen Datenverarbeitung.

WIE SIEHT DAS DIDAKTISCHE KONZEPT DIESES PROJEKTES AUS?

Entmystifizierung des Computers ist hier kein bloßes Schlagwort, hier wird ernst damit gemacht, und zwar handgreiflich. Es wird gebohrt, gelötet, verdrahtet und getestet. Nicht graue Theorie über das Innenleben einer CPU wird vermittelt, sondern handfestes Wissen, das Schülern der Sekundarstufe I zumutbar ist. Ein üblicher 8-Bit-Prozessor ist halt sehr viel komplexer aufgebaut als ein 1-Bit-Prozessor. Um beispielsweise die grundsätzlichen Komponenten eines Z-80 kennenzulernen, bedarf es schon eines umfangreichen Vokabulars:

Computer Control Unit, Program Control Unit, Programmadresszähler, Adreß Register, Akkumulator, Zwischenspeicher, Index-Register, Stackpointer, Flag Register, Takt, Steuer-Bus, Daten-Bus, Adreß-Bus, usw. . Die Erklärung, wie diese Komponenten nun zusammenspielen, wollen wir uns, und vor allem den Schülern, ersparen.

Didaktische Reduktion ist hier gefordert. Dem entspricht die Wahl eines 1-Bit Prozessors, des MC 14500 von Motorola.

WAS KANN DER WDR-1-BIT-COMPUTER?

Im wesentlichen erfüllt er Steueraufgaben. Er kann eine Ampelanlage, einen Roboter, einen Aufzug und andere Geräte steuern. Er kann aber auch zur physikalischen Modellbildung herangezogen werden, wenn Boolesche bzw. Schaltalgebra Unterrichtsthema ist. Die wichtigsten logischen Verknüpfungen, die NICHT-, UND- sowie die ODER-Verknüpfung, sind in der CPU unter anderem realisiert.

Er stellt einen vollwertigen Mikroprozessor dar, dessen Funktionsweise Schülern auf Grund seines relativ einfachen Aufbaus erklärt werden kann und die auch verstanden wird. Der Prozessor 14500 schließt die Lücke zwischen der einfachen, aber teuren, weil fest verdrahteten Schaltungstechnik und den preiswerten, aber schwer verständlichen Mikroprozessoren. Er ist sozusagen das "Missing link" zwischen diesen beiden Extremen.

Natürlich kann er dafür auch sehr viel weniger als seine größeren Brüder, sein Repertoire umfaßt nur 16 Befehle. Die beherrschen Schüler dann aber auch alle!

Der WDR-1 Bit-Computer ist modular aufgebaut, er besteht aus verschiedenen Funktionseinheiten, auf die an späterer Stelle ausführlich eingegangen wird (Siehe: Kapitel 2) Der modulare Aufbau hat einen Nachteil: Der Computer bzw. der Bausatz wird dadurch teurer. Er wurde aber aus didaktischen bzw. methodischen Gründen gewählt: Der Computer kann so im arbeitsteiligen Gruppenunterricht von den Schülern selbst gebaut werden.

Programmiert wird der Computer in Maschinensprache, mit Tastatur in Assembler. Auch dies ist sinnvoll, wenn die grundsätzliche Arbeitsweise eines Computers im Vordergrund steht. Die wenigsten Schüler werden jemals im Berufsleben eigene professionelle Programme schreiben müssen, weder in Basic, Pascal, Logo, Elan oder sonst einer Sprache. Durch die Maschinensprache erfahren sie aber, wie Computer Daten wirklich verarbeiten.

DIE KONZEPTION DES WDR 1-BIT-COMPUTERS

Ein Computer ist eine funktionsfähige Steuereinheit aus einem Prozessor und Zusatzbausteinen, je nach Bedarf. Ein Prozessor besteht aus einer vollständig programmgesteuerten Rechnerschaltung. Einen besonders einfachen Prozessor stellt der 1-Bit Mikroprozessor MC 14500 dar. Es ist ein 1-Bit-Prozessor, weil er zu einem Zeitpunkt nur 1 Bit verarbeiten kann. 1 Bit, das ist die Information "ja" oder "nein" bzw. "1" oder "0", verwirklicht durch 0 Volt bei "0" und 5 Volt bei "1". Von einem Mikrocomputer spricht man, wenn der Prozessor ein Mikroprozessor ist, der Prozessor also in einer winzigen, mikroskopisch kleinen integrierten Schaltung realisiert ist und die Zusatzbausteine ebenfalls integrierte Schaltungen sind. Moderne Mikrocomputer arbeiten z.B. mit 8 oder 16 Bit-Mikroprozessoren. Sie sind so komplex und kompakt gebaut, daß sie sich für den Bau und die elementare Programmierung eines Computers in der Schule nicht eignen.

Der WDR-1 Bit-Computer dagegen stellt für Schüler der Sekundarstufe I und II das didaktisch Einfache dar, da er auf seine wesentlichen Teile reduziert worden ist. Bei einer weiteren Verringerung von Funktionsteilen würde er als Computer nicht mehr richtig arbeiten können. Für den schulischen Bedarf wurden die Verbindungen zwischen den integrierten Schaltungen im Gegensatz zu den professionellen Mikrocomputern nicht miniaturisiert, sondern so weit vergrößert, daß der Computer für den Schüler wieder greifbar wird. Der WDR-1-Bit-Computer ist ein Computer zum Anfassen, deshalb fehlt ihm auch ein Gehäuse. Eine weitere Vereinfachung stellt sein modularer Aufbau in 4 steckbare Funktionseinheiten dar. Dies bedeutet für den Schüler eine zusätzliche Verringerung des Komplexitätsgrades beim Begreifen des Prinzips eines Mikrocomputers. Die genannte Vereinfachung sowie der modulare Aufbau ermöglichen so einen sehr sicheren Nachbau des WDR-1-Bit-Computers durch Schüler unter Anleitung des Lehrers ohne Verwendung spezieller Hilfsmittel. Der Schüler soll sogar Löcher für die Aufnahme der Bauteile in die Platine selbst bohren. Bei dieser Tätigkeit gewinnt er Vertrauen beim Zusammenbauen eines für ihn zunächst noch unbekanntes Computers mit Hilfe einer ihm bekannten Fertigkeit.

Die Verwendung von nur 14 Befehlen (16 sind möglich) erleichtert zusätzlich das Erlernen der Grundprinzipien der Programmierung eines Computers, da herkömmliche Computer einen Befehlsvorrat von einigen 100 Befehlen aufweisen, deren Kenntnis und Anwendung sehr viele Anwender überfordern. Mit diesen 14 Befehlen können mittels einfachster Programme alle den Computer aufbauenden Elemente simuliert werden. So können seine logischen Bausteine wie UND, ODER, NICHT oder seine Speicher und seine Zähler dargestellt werden: Der Computer erklärt sich selbst.

Es liegt auf der Hand, daß ein für die Schule konzipierter, auf das didaktisch Einfache zurückgeführter Mikrocomputer im Preis-Leistungsverhältnis nicht mit modernen Mikrocomputern konkurrieren kann, zumal er auch in seinem Aufbau bewußt groß gehalten worden ist. Wegen seiner Einfachheit ermöglicht er jedoch schon in der Sek. I einen schülergemäßen und elementaren Einstieg in den Aufbau, die Wirkungsweise und die Programmierung von Mikrocomputern.

WELCHE LERNZIELE KÖNNEN IN EINEM UNTERRICHTSPROJEKT ERREICHT WERDEN?

Wir haben diese Frage gestellt, obwohl wir sie nur "grob" beantworten wollen. Eine Auflistung aller Feinziele wäre lang und wenig sinnvoll. Die Schülergruppe, die die Anzeigeplatine zusammenbaut, wird andere Ziele erreichen als die Gruppe, die sich die Speicherplatine vorgenommen hat. Auf den jeweiligen Schwerpunkt kommt es also an. Begnügen wir uns deshalb mit einigen wenigen (ergänzungsbedürftigen) Grobzielen.

- * Das unseres Erachtens wichtigste Ziel: Die Schüler sollen durch den Bau eines eigenen Computers, am Bau ihres Computers regen Anteil nehmen, "motiviert" sein! In allen bisherigen Unterrichtsprojekten war dies der Fall, obwohl einzelne Platinen nicht immer auf Anhieb funktionierten.
- * Weiter sollen die Schüler ein gewisses handwerkliches Geschick beim Bau erlangen. Dieses Ziel gilt sicherlich weniger für Gymnasiasten, um so wichtiger ist es für Hauptschüler. Für diese Gruppe steht es in engem Zusammenhang mit dem ersten Ziel. Wie weit der Lehrer auf die einzelnen Bauteile des Computers, auf Widerstände, Dioden, Kondensatoren und Chips eingehen soll, muß der Lehrer vor Ort entscheiden. Hier kann nur allgemein gefordert werden, daß die Schüler das Verhalten der wichtigsten Bauelemente des 1-Bit-Computers kennen, insbesondere das des Mikroprozessors kennenlernen.
- * Ein Verständnis des Zusammenwirkens der einzelnen Bauteile und der Funktionsgruppen des Computers ist das nächste Ziel. Dazu gehört ein Vergleich der Grobstruktur des 1-Bit-Computers mit größeren Systemen.
- * Um schließlich programmieren zu können, muß man die Befehle kennen, ihre Codierung und ihre Wirkungsweisen, speziell die vorhandenen logischen Verknüpfungen.

Obwohl sich noch weitere Grobziele auflisten lassen, z.B. Binärsystem, Peripherie etc. heißt unsere letzte Forderung, die Schüler sollen ein Grundverständnis der Komponenten der CPU und ihres Zusammenwirkens erlangen. Dies heißt, sie sollen exemplarisch die Architektur eines Rechners kennenlernen.

DER WDR-COMPUTER-SCHEIN

Wer den 1-Bit-Computer erfolgreich baut, der erhält einen (oder auch mehrere) Computerscheine! Nach Beendigung des erfolgreichen Aufbaus und der Programmierung des WDR-1-Bit-Computers bekommen die Schüler einen Computerschein der Klasse 1 entsprechend 1 Bit. Der Computerschein ist so ausgelegt, daß er auch für andere Klassen ausgegeben werden kann. Der Computerschein ist im Anhang als Kopiervorlage vorgestellt, er kann aber auch farbig in Klassenstärke von der Redaktion des Schulfernsehens des WDR, Appelloplatz 1, 5000 Köln 1 angefordert werden.

2. DER AUFBAU DES WDR-1-BIT-COMPUTERS (HARDWARE)

In diesem Abschnitt soll auf den technischen Aufbau, d.h. auf die Bauteile und ihre Verdrahtung eingegangen werden. Der englische Fachausdruck dafür ist "Hardware". Zunächst wird der prinzipielle Aufbau beschrieben. Abb. 2 zeigt das Blockschaltbild des WDR-1-Bit-Computers. Es soll als Beilage zum Kapitel 2.1 dienen.

2.1. DER PRINZIPIELLE AUFBAU

Der WDR-1-Bit-Computer besitzt vier Eingangsleitungen, die entweder durch Schalter oder über einen Expansionsstecker durch angeschlossene Geräte beschaltet werden können. Auf diesem Wege kommen Signale in den Computer hinein.

Weiter gibt es acht Ausgangsleitungen, die ebenfalls an den Expansionsstecker geführt sind. Über diese Leitungen können Ergebnisse oder Entscheidungen des Computers an angeschlossene Geräte weitergegeben werden, die dann die Informationen in Steuervorgänge umsetzen. Drei dieser Ausgänge sind zusätzlich mit drei weiteren Eingangsleitungen verbunden, so daß der Computer auf seine alten Ergebnisse zurückgreifen kann, d.h., das System hat drei Datenspeicher.

Der Mikroprozessor besitzt ein Ergebnisregister, welches immer das Ergebnis, das durch die Ausführung eines Programmschrittes gewonnen wird, enthält. Dieses Ergebnisregister ist mit noch einer Eingangsleitung verbunden, damit die Information des Ergebnisregisters wieder gelesen werden kann.

Der Computer besitzt also acht Eingangs- und acht Ausgangsleitungen. Zur Kontrolle der Informationen dieser Leitungen sind sie mit jeweils einer Leuchtdiode verbunden, die zusammen einen Teil der Anzeige darstellen.

Die Eingangsleitungen sind mit einem Eingabebaustein und die Ausgangsleitungen mit einem Ausgabebaustein verbunden. Der Eingabebaustein schickt über die Datenleitung die Information einer der acht Eingänge an den Mikroprozessor. Genauso erhält der Ausgabebaustein über dieselbe Datenleitung eine Information vom Mikroprozessor, die er weiter an einen der acht Ausgänge schickt. Über eine weitere Leitung kann der Ausgabebaustein den Auftrag erhalten, diese Information wieder zu löschen. Dazu muß man einen Löschtaster betätigen, der auch mit dem Mikroprozessor verbunden ist und auch dessen Information löscht. Damit gesichert ist, ob der Eingabebaustein eine Information an den Mikroprozessor senden soll oder ob der Ausgabebaustein eine Information vom Mikroprozessor aufnehmen soll, gibt es eine Schreib-/Leseleitung, die den Mikroprozessor mit den beiden Bausteinen verbindet. Beim Auftrag "Lesen" arbeitet nur der Eingabebaustein und beim Auftrag "Schreiben" arbeitet nur der Ausgabebaustein.

Über die Eingabetastatur können die Befehle eingegeben werden. Von den sechzehn möglichen Befehlen sind nur vierzehn ausgeführt, um das System übersichtlicher zu machen. Im Gegensatz zu den üblichen Computern sind die vierzehn Befehle fest verdrahtet, d.h., für jeden Befehl gibt es eine Taste. Dies wäre bei einem Befehlssatz von einigen hundert Anweisungen, wie sie größere Computer haben, nicht mehr zweckmäßig.

Damit der Prozessor die vierzehn im System verwirklichten Befehle unterscheiden kann, werden vier Leitungen benötigt.

Je nachdem, ob eine Leitung "0" oder "1" führt, sind $2^4=16$ Möglichkeiten vorhanden, wie folgende Darstellung im Dualsystem es zeigt: 0000; 0001; 0010; 0011; 0100; 0101; 0110; 0111; 1000; 1001; 1010; 1011; 1100; 1101; 1110; 1111.

Die meisten Befehle beziehen sich auf einen der acht Eingänge oder einen der acht Ausgänge, die somit zu Adressen werden. Daher gibt es acht weitere Tasten, die die Nummern der Ein- und Ausgänge enthalten. Diese Ein-/Ausgabeadressen sind also ebenfalls fest verdrahtet. Für sie werden drei Leitungen zur Unterscheidung der Adressen benötigt ($2^3=8$): 000; 001; 010; 011; 100; 101; 110; 111. Sie bedeuten in dieser Reihenfolge: 0; 1; 2; 3; 4; 5; 6; 7. Dieses Binärsystem wird auch Dualsystem genannt. Jede Stelle besitzt eine Wertigkeit. Die letzte Stelle, d.h. das rechte Bit, hat die Wertigkeit 1. Nach vorne verdoppelt sich die Wertigkeit pro Stelle. Sie ist wichtig bei der Umrechnung einer Dualzahl in eine Dezimalzahl. Eine "0" bedeutet kein Wert vorhanden und eine "1" bedeutet Wert vorhanden. Die vorhandenen Werte werden addiert. Die Dualzahl 101 z.B. ist die Dezimalzahl 5: $1*4+0*2+1*1=5$.

Eine vierte Leitung existiert, ist aber nicht mit der Eingabetastatur verbunden. Sie kann zur Erweiterung der Ein-/Ausgabeadressen dienen, denn mit vier Leitungen können sechzehn Ein-/Ausgabeadressen unterschieden werden.

Die Eingabetastatur enthält also vierzehn Tasten für die Befehle und acht Tasten für die Ein-/Ausgabeadressen. Mit Hilfe dieser zweiundzwanzig Tasten und den sieben Leitungen können nun Programme erstellt werden. Wie bei den Ein- und Ausgängen sind die sieben Leitungen mit jeweils einer Leuchtdiode verbunden. Diese zeigen an, um welchen Befehl und um welche Ein-/Ausgabeadresse es sich bei den einzelnen Programmschritten handelt.

Ein Programm bekommt erst seine große Bedeutung, wenn man die einzelnen Programmschritte speichern kann. Da ein Programmschritt aus einem Befehl, durch vier Leitungen codiert, und einer Ein-/Ausgabeadresse, durch drei Leitungen codiert, besteht, müssen die Programmspeicher sieben Informationen parallel, d.h. gleichzeitig, aufnehmen.

Die Speicherbausteine in dem WDR-1-Bit-Computer sind jeweils 256 x 4-Bit-Speicher. Die zweite Zahl, d.h. die "4", bezeichnet die Anzahl der Bits, die der Programmspeicher zu einem Zeitpunkt parallel aufnehmen kann. Bei sieben Informationen brauchen wir somit zwei Speicherbausteine, so daß wir einen 256 x 8-Bit-Speicher erhalten. Die erste Zahl, d.h. die "256", bezeichnet die Anzahl der Bits, die der Programmspeicher auf einem der acht Eingänge seriell, d.h. nacheinander unter verschiedenen Programmadressen, aufnehmen kann. Sie gibt also die maximale Anzahl der Programmschritte an, die gespeichert werden können. In diesem Fall darf ein Programm nicht mehr als 256 Programmschritte haben.

Der eine Programmspeicher beinhaltet die Befehle und der andere Programmspeicher enthält die Ein-/Ausgabeadressen. Das Einlesen erfolgt mit Hilfe eines Schreib-Tasters. Der Befehlsspeicher ist mit dem Mikroprozessor verbunden und der Adressenspeicher ist mit dem Eingabe- und dem Ausgabebaustein verbunden.

Damit die einzelnen Programmschritte in der richtigen Reihenfolge gespeichert sind und bei Bedarf in eben dieser

Reihenfolge ausgegeben werden können, benötigen wir einen Programmzähler. Jeder Programmschritt hat eine eigene Programmadresse. Wir müssen 256 verschiedene Adressen anwählen können. Dies ist mit einem 8-Bit-Binärzähler zu realisieren ($2^8=256$). Beide Programmspeicher sind mit diesem Programmzähler über acht Leitungen verbunden. In dem 1-Bit-Computer sind zwei 4-Bit-Binärzähler hintereinander geschaltet.

Normalerweise beginnt der 8-Bit-Zähler bei der Adresse 0 ($0 \cdot 128 + 0 \cdot 64 + 0 \cdot 32 + 0 \cdot 16 + 0 \cdot 8 + 0 \cdot 4 + 0 \cdot 2 + 0 \cdot 1 = 0$) und zählt im Dualsystem hoch bis zur Adresse 255 ($1 \cdot 128 + 1 \cdot 64 + 1 \cdot 32 + 1 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 1 \cdot 1 = 255$). Danach fängt er wieder bei 0 an. Man kann jedoch mit Hilfe eines Tasters jederzeit zum Zählerstand 0 zurückkehren. Dieser Tastendruck kann aber auch durch einen Programmschritt, bzw. durch einen Befehl, ersetzt werden, den man z.B. am Ende eines Programmes setzt.

Damit der Programmzähler zählen kann, braucht er einen Takt. Diesen Takt bekommt er vom Mikroprozessor. Durch einen Taktwahlschalter kann die Schnelligkeit des Taktes eingestellt werden. Dabei gibt es drei Stufen:

In der Stellung "Schnell-Takt" kommt der Takt allein vom Mikroprozessor.

In der Stellung "Langsam-Takt" wird das Taktsignal eines Timers wirksam. Über ein Potentiometer (einstellbarer Widerstand) kann der Takt noch stufenlos variiert werden.

In der Stellung "Hand-Takt" kommt nur dann ein Taktimpuls, wenn man einen dafür vorgesehenen Taster betätigt.

Eine Leuchtdiode zeigt uns die Taktimpulse an. Außer dem Programmzähler ist noch der Eingabebaustein mit dem Takt verbunden.

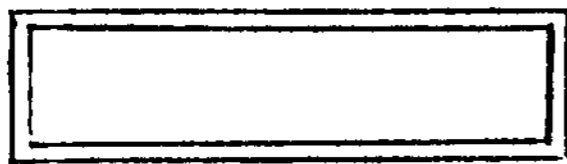
Die Anzeigeeinheit besteht aus vier Bausteinen und 27 Leuchtdioden. Fünfzehn Leuchtdioden zeigen den Informationsgehalt der Ein- und Ausgänge, vier den Befehlscode, drei die Nummer der Ein- oder Ausgabelitung im Dualsystem, eine den Inhalt des Ergebnisregisters und eine den Takt an. Eine Leuchtdiode ist dann noch mit der Datenleitung verbunden.

Es wurde hier bewußt auf englische Fachausdrücke der Datenverarbeitung verzichtet, um dem Anfänger den Eintritt in die Mikroprozessortechnik zu erleichtern.

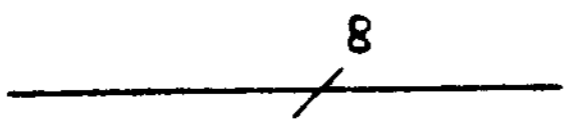
Abb. 1: Erläuterungen zu den Abb. 2 und 3:



Funktionseinheiten mit integrierten Bausteinen



Bedienungselemente



8 Leitungen

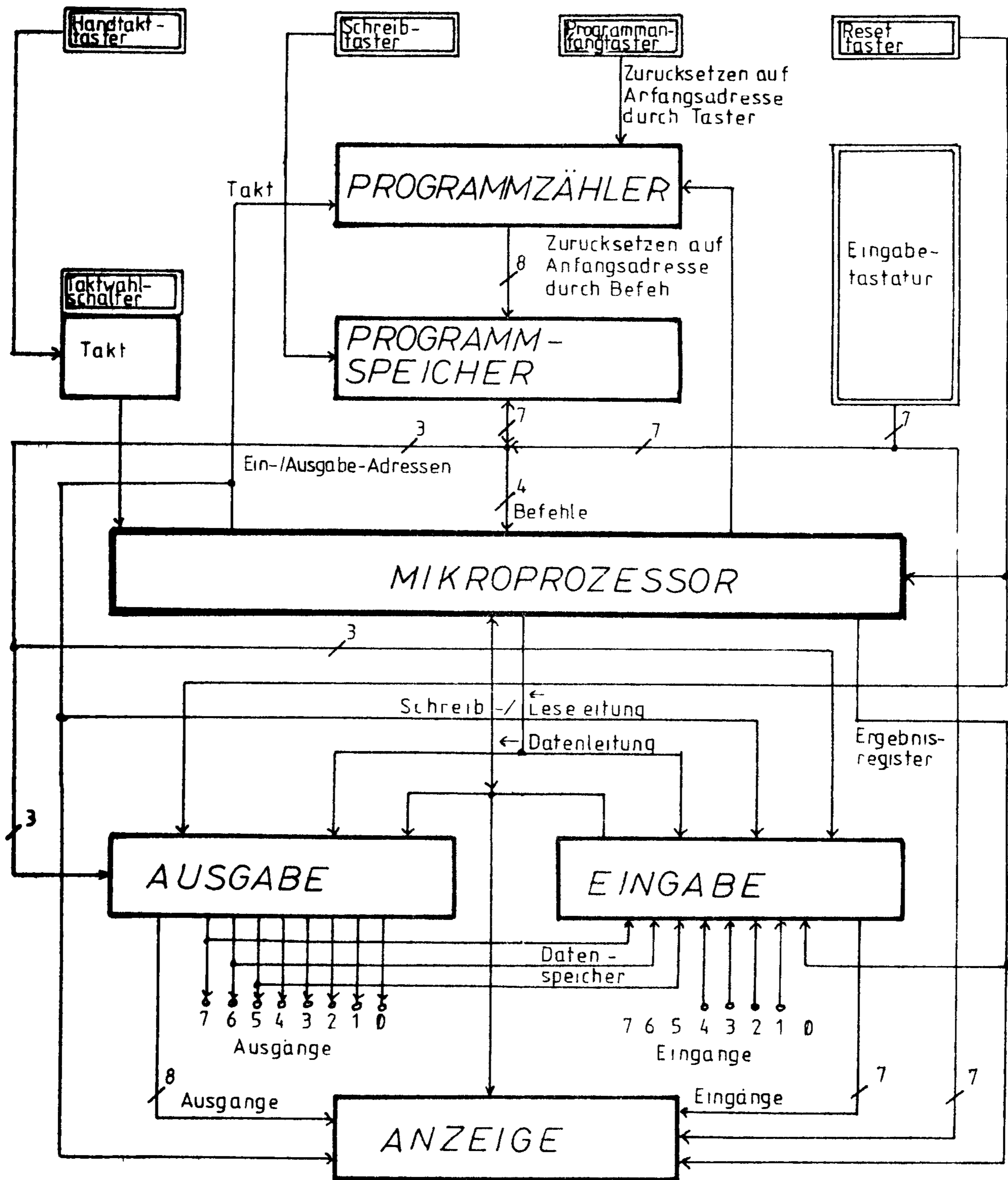


Abb. 2: Blockschaltbild des WDR-1-Bit-Computers

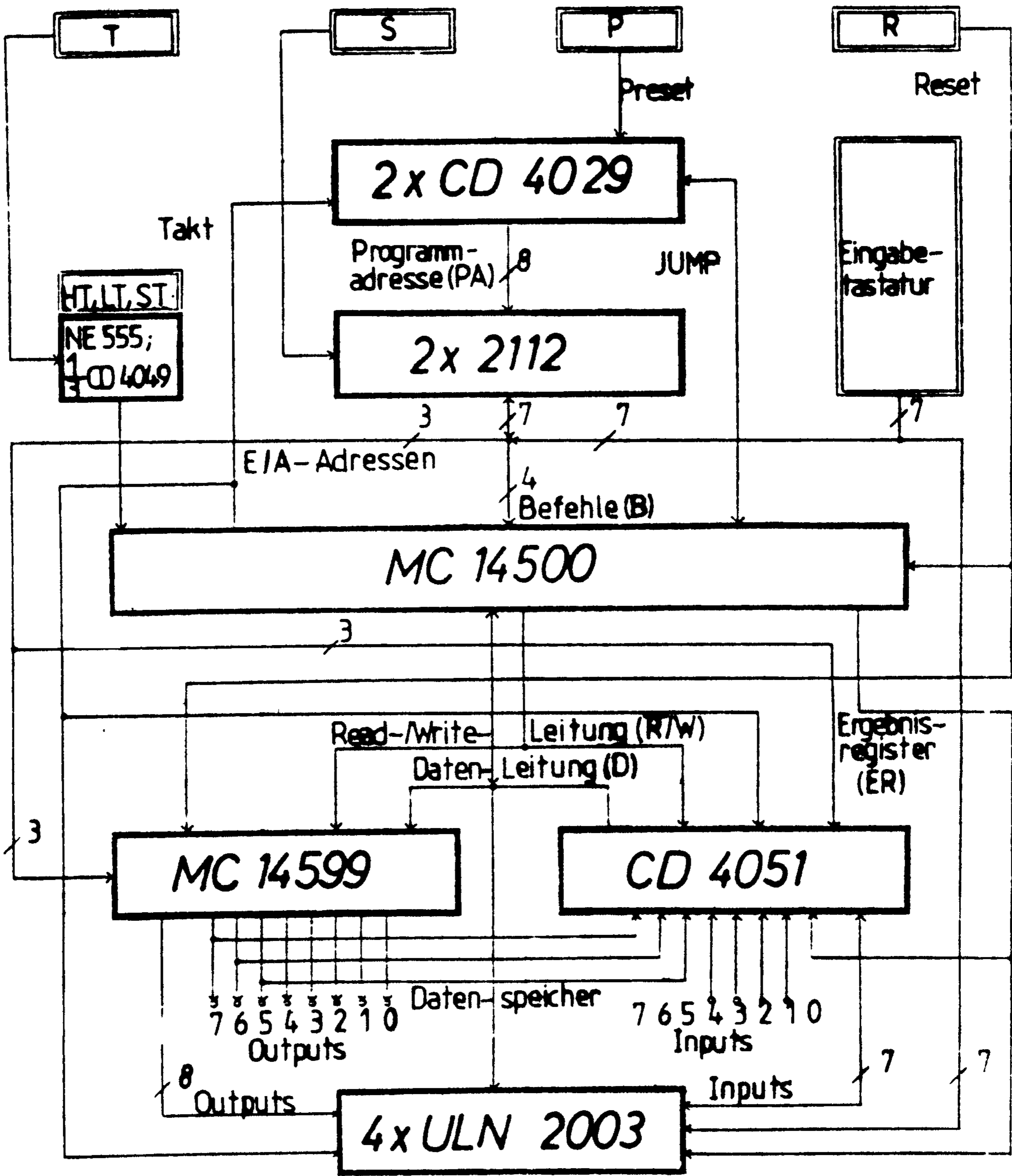


Abb. 3: Blockschaltbild des WDR-1-Bit-Computers (technisch)

Abb. 3 zeigt das gleiche Blockschaltbild wie Abb. 2. Hier sind nur teilweise die deutschen Ausdrücke durch die englischen Fachausdrücke ersetzt. Die dafür verwendeten Abkürzungen stehen in Klammern dahinter. Außerdem sind in Abb. 3 die Funktionseinheiten durch die Typenbezeichnungen, d.h. durch die Typennummern, der integrierten Bausteine ersetzt.

Durch zwei Blockschaltbilder wird dem Anfänger der Zugang zu den englischen Fachausdrücken erleichtert. Zusätzlich kann er schnell nachschauen, welche Aufgabe z.B. der Baustein mit der Typennummer 4051 hat.

2.2. DIE FUNKTIONSEINHEITEN

2.2.1. DER MIKROPROZESSOR MC 14500

Der wichtigste Baustein in einem Computersystem ist der Mikroprozessor. Er wird auch zentrale Datenverarbeitungseinheit (engl. Central Processing Unit - CPU) genannt. Die CPU MC 14500 ist erheblich einfacher aufgebaut als ihre 8-Bit-Kollegen. Abb. 4 zeigt die Pinbelegung des MC 14500.

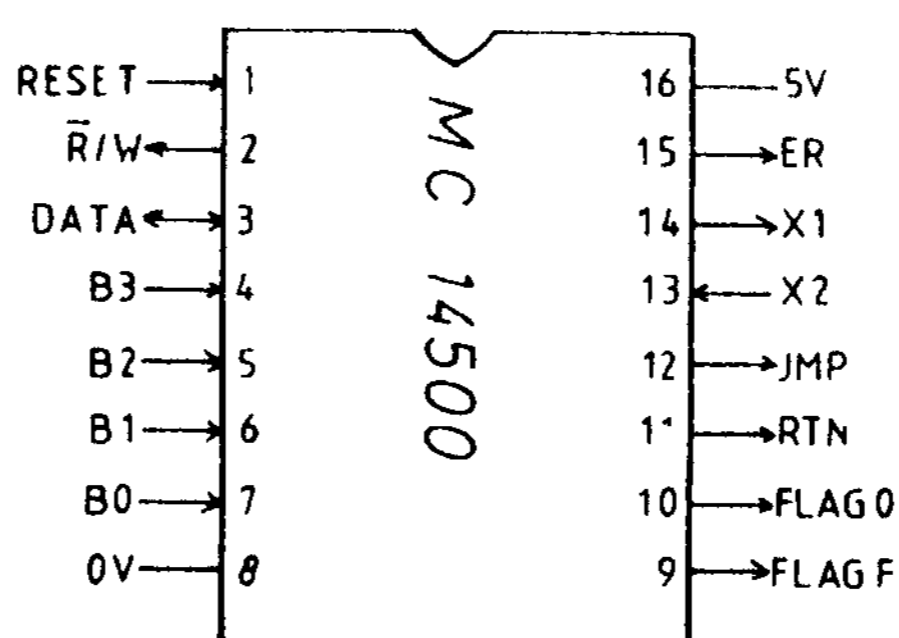


Abb. 4: Pinbelegung des MC 14500




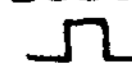
Bedeutung der einzelnen Beine:

- Pin 1: RESET Rücksetzeingang:
Normalerweise ist dieses Bein über einen Widerstand mit 0 V verbunden. Verbindet man es aber mit Hilfe eines Tasters (Reset-Taster) mit 5 V, dann werden alle Informationen in der CPU gelöscht, und diese geht in die Ruhstellung über.
- Pin 2: \bar{R}/W Schreib /Leseleitung:
Ausgangsleitung, die angibt, ob die CPU gerade eine Information über ihre Datenleitung hereinholt ("Lesen") oder ob sie über diese Leitung Daten ausgibt ("Schreiben"). Im ersten Fall ist die Spannung an diesem Bein 0 V, während bei der Ausgabe 5 V anliegen. Der Querbalken über dem R ist ein Negationszeichen und bedeutet hier "0"-aktiv. (Kein Querbalken bedeutet dann "1"-aktiv.) Er soll hier ausdrücken, daß die CPU bei 0 V "liest" und bei 5 V "schreibt".
- Pin 3: DATA Datenleitung:
Leitung, über die die Informationen sowohl beim Lesen als auch beim Schreiben von dem Eingabebaustein bzw. zu dem Ausgabebausteins des Computers geleitet werden. Da dieser Vorgang auf einer Leitung in beide Richtungen erfolgt, hat die Leitung auch den Namen "Bidirektionaler Datenbus".
- Pin 4: B3,
- Pin 5: B2,
- Pin 6: B1,
- Pin 7: B0 Befehlsleitungen:
Anschluß von Leitungen, die die Befehle vom Programmspeicher an die CPU heranzuführen.
- Pin 8: 0 V Masseanschluß
- Pin 9: FLAG F Kennzeichen für den NOP F-Befehl:
Ausgabe eines 5 V Impulses (Steuersignal) aus der CPU, wenn der Befehl NOP F ausgeführt wird. Sonst liegen hier 0 V an.
Dieses Bein ist im WDR-1 Bit-Computer nicht beschaltet.
- Pin 10: FLAG O Kennzeichen für den NOP O-Befehl:
Ausgabe eines 5 V-Impulses aus der CPU, wenn der Befehl NOP O ausgeführt wird. Sonst liegen auch hier 0 V an. Dieses Bein ist auch nicht beschaltet.
- Pin 11: RTN ReTurN, Zurückkehren:
Hier tritt ein 5 V Impuls beim Befehl RTN auf:
Dieses Bein ist auch nicht beschaltet.
- Pin 12: JMP JuMP, Springen:
Hier tritt ein 5 V-Impuls beim Befehl JMP auf. Dieser wird zum Rücksetzen des Programmzählers benutzt:

- Pin 13: X2 Takteingang für den Taktoszillator der CPU:
Hier wird von außen ein Takt eingepreßt.
- Pin 14: X1 Taktausgang des Taktoszillators der CPU:
Bausteine des Computers, die einen Takt zur Synchronisation benötigen, sind mit diesem Taktausgang verbunden.
- Pin 15: ER Ergebnisregister:
Ausgang, an den das Ergebnisregister der CPU geführt ist. Dort steht immer das Ergebnis, das durch die Ausführung des letzten Programmschrittes gewonnen wird.
- Pin 16: 5 V Betriebsspannung

Zum besseren Verständnis des 1-Bit-Computers soll der innere Aufbau der CPU an einem Prinzipschaltbild (Abb. 5) erläutert werden. Dafür muß zunächst der Befehlssatz der CPU kennengelernt werden. Die CPU kann sechzehn verschiedene Befehle ausführen. Jeder hat ein Codewort, das durch eine vierstellige Zahl im Dualsystem ausgedrückt wird. Hier soll nur kurz auf die Wirkung der einzelnen Befehle eingegangen werden. Eine ausführliche Beschreibung erfolgt im Kapitel 4.1.

Der Befehlssatz der CPU:

Befehlscode	Befehl	Befehlsart	Wirkung des Befehls
B3 B2 B1 B0			
0 0 0 0	NOP 0	Steuerbefehl	FLAG 0 = 
0 0 0 1	LD	Eingabebefehl	ER = DATA
0 0 1 0	LDC	Eingabebefehl	ER = $\overline{\text{DATA}}$
0 0 1 1	AND	Rechenbefehl	ER = ER * DATA
0 1 0 0	ANDC	Rechenbefehl	ER = ER * $\overline{\text{DATA}}$
0 1 0 1	OR	Rechenbefehl	ER = ER + DATA
0 1 1 0	ORC	Rechenbefehl	ER = ER + $\overline{\text{DATA}}$
0 1 1 1	XNOR	Rechenbefehl	ER = ER * DATA + $\overline{\text{ER * DATA}}$
1 0 0 0	STO	Ausgabebefehl	DATA = ER
1 0 0 1	STOC	Ausgabebefehl	DATA = $\overline{\text{ER}}$
1 0 1 0	IEN	Freigabebefehl	IEN-Register = DATA
1 0 1 1	OEN	Freigabebefehl	OEN-Register = DATA
1 1 0 0	JMP	Steuerbefehl	JMP FLAG = 
1 1 0 1	RTN	Steuerbefehl	RTN FLAG = 
1 1 1 0	SKZ	Programmverzweigungsbefehl	Falls ER = 0: übernächster Befehl Falls ER = 1: nächster Befehl
1 1 1 1	NOP F	Steuerbefehl	FLAG F = 

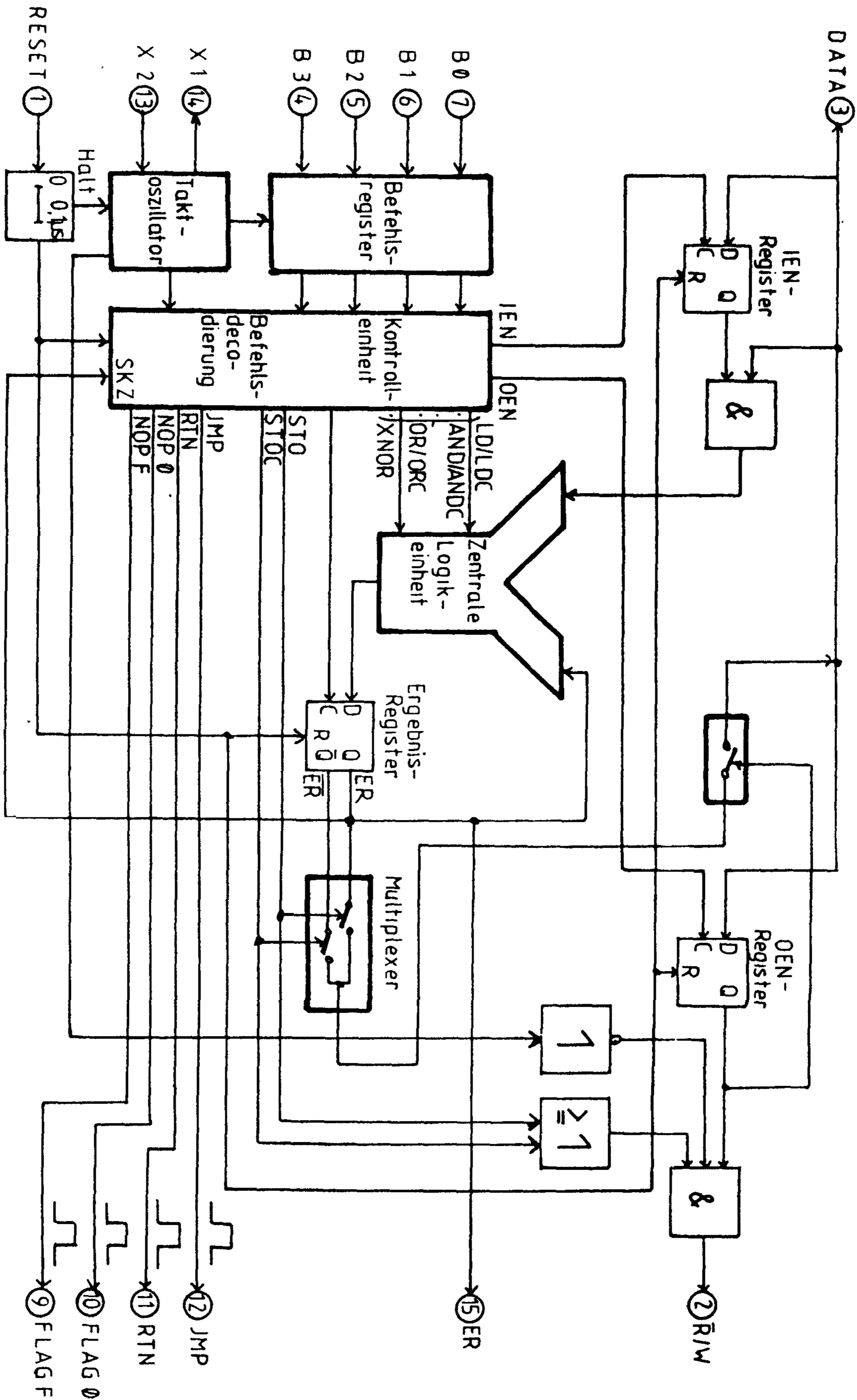


Abb. 5: Prinzipschaltbild des Mikroprozessors MC 14500

In der letzten Spalte stehen die Wirkungen der Befehle als arithmetische Anweisungen. Eine arithmetische Anweisung enthält von links nach rechts:

1. den Namen einer Speicherstelle (Register oder Leitung), die das Ergebnis aufnimmt,
2. das Zuordnungszeichen (kein Gleichheitszeichen !) "=",
3. einen arithmetischen Ausdruck aus Namen der Speicherstellen und Rechenzeichen ("*" bedeutet logische UND-Verknüpfung, "+" bedeutet logische ODER-Verknüpfung) oder das Symbol für einen 5 V-Impuls.

Die Negationszeichen bedeuten hier nicht "0"-aktiv, sondern geben das Komplement, d.h. das Gegenteil, einer Information an. Das Komplement von 0 ist 1 und umgekehrt.

Die Elektronik innerhalb der CPU besteht aus folgenden Teilen:

1. Taktoszillator

Der Takt wird in einem eigenen Kapitel (Siehe: Kapitel 2.2.8.) beschrieben

2. Befehlsregister

Die Befehle werden der CPU durch die vier Befehlseingänge (B0, B1, B2 und B3) zugeführt und mit der negativen Taktflanke von X1 ins Befehlsregister übernommen.

Negative Taktflanke bedeutet Übergang von 5 V auf 0 V des Taktes. Dies geschieht z.B. in der Stellung "Hand-Takt" beim Drücken des Takt-Tasters.

3. Kontrolleinheit

Die Befehle gelangen vom Befehlsregister zur Kontrolleinheit und werden dort decodiert. Liegt ein Steuerbefehl an, so wird ein 5 V-Impuls mit der Dauer einer Taktperiode, beginnend mit der negativen Taktflanke von X1, erzeugt. Dieses Steuersignal (FLAG) erscheint, je nach Befehl, an einem der vier Steuerausgänge (Pin 9...12). Ist der Befehl ein Logikbefehl oder ein Eingabebefehl, so wird er zur Zentralen Logikeinheit weitergeleitet.

Von besonderer Charakteristik ist der Programmverzweigungsbefehl SKZ. Er hat folgende Wirkung: Steht das Ergebnisregister auf "1", so ist er bedeutungslos, d.h., das Programm läuft normal weiter.

Steht das Ergebnisregister auf "0", so wird der nächste Befehl ignoriert, und das Programm läuft bei dem übernächsten Befehl weiter. Die Entscheidung, ob der nächste Befehl ausgeführt wird oder ob er übersprungen wird, findet in der Kontrolleinheit statt. Zu diesem Zweck ist der Ausgang Q des Ergebnisregisters mit der Kontrolleinheit verbunden.

4. Zentrale Logikeinheit

Die Zentrale Logikeinheit hat als Eingänge die Datenleitung und den Ausgang des Ergebnisregisters. Die beiden Bits, die über diese Eingänge in die Zentrale Logikeinheit gelangen, werden durch eine logische Funktion, vorgegeben durch den jeweiligen logischen Befehl, verknüpft. Das Ergebnis dieser Operation wird an das Ergebnisregister weitergeleitet. Bei einem Eingabebefehl wird der Inhalt der Datenleitung bzw. dessen Komplement an das Ergebnisregister weitergeleitet.

5. Ergebnisregister

Das Ergebnisregister speichert immer das Ergebnis der letzten Logikoperation (beim Logikbefehl), bzw. die Information der Datenleitung (beim Eingabebefehl). Das Ergebnisregister ist ein D-Flipflop (Verzögerungs-Flipflop, engl. Delay-D). Die jeweils am Eingang D liegende Information (hier: das Ergebnis der Zentralen Logikeinheit) wird bei einem D-Flipflop zu einem bestimmten Zeitpunkt (hier: bei der positiven Taktflanke von X1 am Takteingang C) eingespeichert und erscheint am Ausgang Q. Der invertierte Ausgang wird mit \bar{Q} bezeichnet. Der Ausgang Q des Ergebnisregisters ist an Pin 15 geführt. Positive Taktflanke bedeutet Übergang von 0 V auf 5 V des Taktes. Dies geschieht z.B. in der Stellung "Hand-Takt" beim Loslassen des Takt-Tasters.

6. Die Eingabeschaltung und das IEN-Register

Das IEN-Register ist ein Eingangs-Freigabe-Register (engl. Input ENable-IEN). Es ist ein D-Flipflop mit dem Eingang D, verbunden mit der Datenleitung, und dem Eingang C, verbunden mit der Kontrolleinheit. Beim Befehl IEN kommt an Eingang C ein Taktimpuls. Nur dann wird die Information der Datenleitung am Eingang D eingelesen, gespeichert und zum Ausgang Q durchgeschaltet. Wie beim Ergebnisregister wird die Information bei der positiven Taktflanke von X1 eingespeichert. Der Inhalt des IEN-Registers kann nur durch einen neuen IEN-Befehl geändert werden.

Der Ausgang des IEN-Registers ist mit dem Eingang eines UND-Gliedes (Schaltzeichen: &) verbunden. Am zweiten Eingang des UND-Gliedes liegt die Datenleitung. Ist die Eingabe nicht freigegeben, d.h., liegt am Ausgang des IEN-Registers eine "0", so erscheint auch am Ausgang des UND-Gliedes eine "0", unabhängig vom Zustand der Datenleitung. Ist die Eingabe freigegeben, d.h., liegt am Ausgang des IEN-Registers eine "1", so erscheint am Ausgang des UND-Gliedes immer der Zustand der Datenleitung. Eine Wahrheitstabelle soll diese Zusammenhänge noch einmal verdeutlichen:

Eingänge des UND-Gliedes		Ausgang des UND-Gliedes
IEN-Register	Datenleitung	
0	0	0
0	1	0
1	0	0
1	1	1

Das IEN-Register gibt also den Datenweg zur Zentralen Logikeinheit frei, wenn es eine "1" speichert. Hervorzuheben ist, daß das IEN-Register, genau wie das im folgenden beschriebene OEN-Register, über die Datenleitung geladen wird.

7. Die Ausgabeschaltung und das OEN-Register

Das OEN-Register ist ein Ausgangs-Freigabe-Register (engl. Output ENable - OEN). Dieses D-Flipflop arbeitet wie das IEN-Register. Nur erscheint am Eingang C ein Taktimpuls beim Befehl OEN. Durch eine "1" in diesem Register wird die Ausgabe von Daten ermöglicht. Daten werden ausgegeben bei den beiden Ausgabebefehlen STO und STOC. Je nach Ausgabebefehl sendet die Kontrolleinheit für die Dauer einer Taktperiode, beginnend mit

der negativen Taktflanke von X1, eine "1" auf die STO - oder auf die STOC-Leitung. Diese beiden Leitungen gehen auf ein ODER-Glied (Schaltzeichen: 1). Liegt einer der beiden Befehle an, so ist der Ausgang des ODER-Gliedes "1". Der Ausgang dieses ODER-Gliedes ist auf einen Eingang eines UND-Gliedes geführt. Ein zweiter Eingang ist über einen Inverter (Schaltzeichen: 1 mit am Ausgang nachgesetztem Kreis) mit dem Taktoszillator verbunden. Am dritten Eingang liegt der Ausgang Q des OEN-Registers. Nur wenn im OEN-Register eine "1" gespeichert ist, ein Ausgabebefehl anliegt, d.h., wenn am Ausgang des ODER-Gliedes eine "1" erscheint, und der Takt "0" ist, d.h., wenn am Ausgang des Inverters eine "1" anliegt, so ist auch am Ausgang des UND-Gliedes eine "1". Dieser Ausgang steuert den Eingabe- und den Ausgabebaustein. Deshalb ist er über Pin 2 mit der Schreib-/Leseleitung verbunden. Zwei Wahrheitstabellen sollen die Zusammenhänge noch einmal verdeutlichen:

Eingänge des ODER-Gliedes		Ausgang des ODER-Gliedes
STO-Leitung	STOC-Leitung	
0	0	0
0	1	1
1	0	1
-	-	-

Eingänge des UND-Gliedes			Ausgang des UND-Gliedes
OEN-Register	Takt	ODER-Glied	
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Bei dem Befehl STO wird der Inhalt des Ergebnisregisters (Ausgang Q) und bei dem Befehl STOC wird der invertierte Inhalt des Ergebnisregisters (Ausgang Q) über die Datenleitung ausgegeben, wenn die Ausgabe freigegeben ist. Die Datenausgabe wird in Abb. 5 durch drei symbolische Schalter dargestellt. Das Schließen der Schalter wird durch die STO-Leitung, durch die STOC-Leitung und durch das OEN-Register bestimmt. Eine "1" schließt den Schalter. Die Schalter ermöglichen hinsichtlich der Datenleitung drei Ausgangszustände der Ausgangsschaltung: "1", "0" (beide niederohmig) und "Ausgang offen" (hochohmig). Solche Schaltungen werden TSL-Schaltungen (Drei-Zustands-Logik, engl. Tri State Logik - TSL) genannt. Eine Wahrheitstabelle soll den Ausgangszustand der Ausgangsschaltung bzw. den Inhalt der Datenleitung für die verschiedenen Zustände der drei Schalter aufzeigen:

STO-Leitung	STOC-Leitung	OEN-Register	Ausgangszustand bzw. Datenleitung
0	0	0	Ausgang offen
0	0	1	Ausgang offen
0	1	0	Ausgang offen
0	1	1	ER
1	0	0	Ausgang offen
1	0	1	ER
-	-	0	-
-	-	1	-

Liegt der Ausgangszustand "Ausgang offen" vor, so ist die Ausgangsschaltung von der Datenleitung entkoppelt. Das hat den Vorteil, daß dann die Dateneingabe, die auch über die Datenleitung stattfindet, nicht gestört werden kann. Was aber passiert, wenn der Ausgangszustand "0" oder "1" ist, wie es Zustand 4 und 6 der Wahrheitstabelle zeigen? Dabei müssen wir zwei Fälle unterscheiden: Ist der Takt "0", so liegt am Pin 2 der CPU und damit auch auf der Schreib-/Leseleitung eine "1". Dadurch ist der Eingabebaustein gesperrt und der Ausgabebaustein freigegeben. Ist der Takt "1", so arbeitet der Eingabebaustein auch nicht. Er wird dann über Pin 14 der CPU (Taktausgang) gesperrt. Auf diese Weise wird erreicht, daß die CPU nie gleichzeitig Daten aufnehmen und Daten ausgeben kann, was sonst zu Kurzschlüssen führen könnte.

8. Der RESET-Eingang

Durch das Betätigen des RESET-Tasters erscheint am Pin 1 der CPU ein 5 V-Signal. Dieses Signal geht auf ein Ausschalt-Verzögerungsglied. Eine Signaländerung am Eingang dieses Verzögerungsgliedes von 0 V auf 5 V bewirkt eine sofortige Signaländerung am Ausgang von 0 V auf 5 V. Eine Signaländerung am Eingang von 5 V auf 0 V bewirkt erst nach einer bestimmten Verzögerungszeit eine Signaländerung am Ausgang von 5 V auf 0 V. Die Verzögerungszeit beträgt hier ca. 0,1 μ s (millionstel Sec.). Der Ausgang des Verzögerungsgliedes ist unter anderem mit dem Taktoszillator verbunden. Ein 5 V-Impuls an Pin 1 setzt den Takt von X1 (Oszillatorausgang) auf "1" und hält diesen Zustand solange, bis der Ausgang des Verzögerungsgliedes wieder auf 0 V geht. Die eigentliche Aufgabe des RESET-Signals besteht aber darin, Informationen zu löschen. Zu diesem Zweck ist der Ausgang des Verzögerungsgliedes auch mit der Kontrolleinheit und mit den drei D-Flipflops (Ergebnisregister, IEN-Register und OEN-Register) verbunden. Über einen 5 V-Impuls an die Kontrolleinheit wird ein eventuell anliegendes Steuersignal (FLAG) wieder auf 0 V gesetzt. Genauso werden die drei D-Flipflops über einen speziellen Eingang R durch einen 5 V-Impuls zurückgesetzt. Die Inhalte der Register sind danach gelöscht, d.h., an ihren Q-Ausgängen erscheint eine "0".

2.2.2. DER EINGABEBAUSTEIN CD 4051

Der Eingabebaustein CD 4051 ist hier ein 1-aus-8-Multiplexer. Multiplexer haben die Aufgabe, aus verschiedenen angebotenen Daten die gewünschten Daten auszuwählen und über die Ausgänge weiterzuleiten. Der 4051 hat acht Eingänge (E0 ... E7) und einen Ausgang (Datenleitung). Vier dieser Eingänge (E1, E2, E3 und E4) sind mit jeweils einem Schalter verbunden. Durch Schließen eines Schalters wird der dazugehörige Eingang auf "1" gesetzt. Ist der Schalter offen, so ist der Eingang über einen Pull down-Widerstand (Widerstände, die mit 0 V verbunden sind, heißen Pull down-Widerstände) mit 0 V verbunden, d.h., der Eingang wird auf "0" gesetzt. Liegt an der betrachteten Leitung 5 V, so hat der Widerstand praktisch keinen Einfluß. Wird auf die Leitung kein Signal gegeben, so zieht der Widerstand sie auf 0 V. Dadurch wird erreicht, daß die Leitung nur zwei Zustände zuläßt: 0 V ("0") und 5 V ("1"). Auf demselben Prinzip beruhen die Pull up-Widerstände. Diese Widerstände sind mit einer Spannungsquelle (hier: 5 V) verbunden und ziehen die Leitung, wenn kein Signal anliegt, auf 5 V.

Die vier Schalter sind in einem Dual-In-Line-Gehäuse (DIL-Gehäuse) integriert. Abb. 6 zeigt die Pinbelegung des CD 4051.

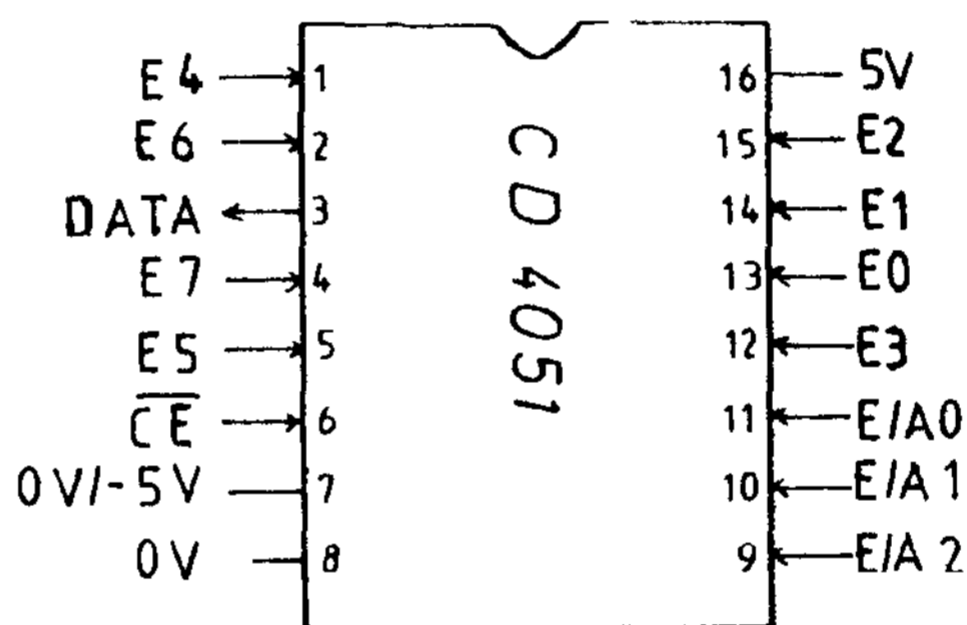


Abb. 6: Pinbelegung des CD 4051

Bedeutung der einzelnen Beine:

- Pin 1 : E4 Eingangsleitung Nr. 4
- Pin 2 : E6 Eingangsleitung Nr. 6
- Pin 3 : DATA Die Datenleitung ist der Ausgang des Bausteins. Die Information einer der acht Eingänge wird über diese Leitung an die CPU geschickt.
- Pin 4 : E7 Eingangsleitung Nr. 7
- Pin 5 : E5 Eingangsleitung Nr. 5
- Pin 6 : CE Chip Enable, Chip befähigen:
Ein 0 V-Signal an diesem Eingang bewirkt ein Arbeiten des Bausteins (Chip). Liegen hier 5 V an, so ist der 4051 nicht in Betrieb und seine Ausgabe ist gesperrt. Daten sollen nur ausgegeben werden, wenn die CPU Daten über die Datenleitung hereinholt und wenn kein Taktimpuls anliegt. In dem Augenblick ist der Takt "0" und die Schreib-/Leseleitung "0". Diesen Zustand

erkennt die Schaltung mit Hilfe eines ODER-Gliedes, Abb.7, an dessen beiden Eingängen die Schreib-/ Leseleitung und der Taktausgang von der CPU angeschlossen sind. Das ODER-Glied gibt nur dann durch eine "0" an Pin 6 die Schaltung frei, wenn beide Eingänge "0" sind. Eine Wahrheitstabelle soll die möglichen Zustände noch einmal aufzeigen:

Eingänge des ODER Gliedes		Ausgang des ODER- Gliedes
\bar{R}/W	Takt	\bar{CE}
0	0	0
0	1	1
1	0	1
1	1	1

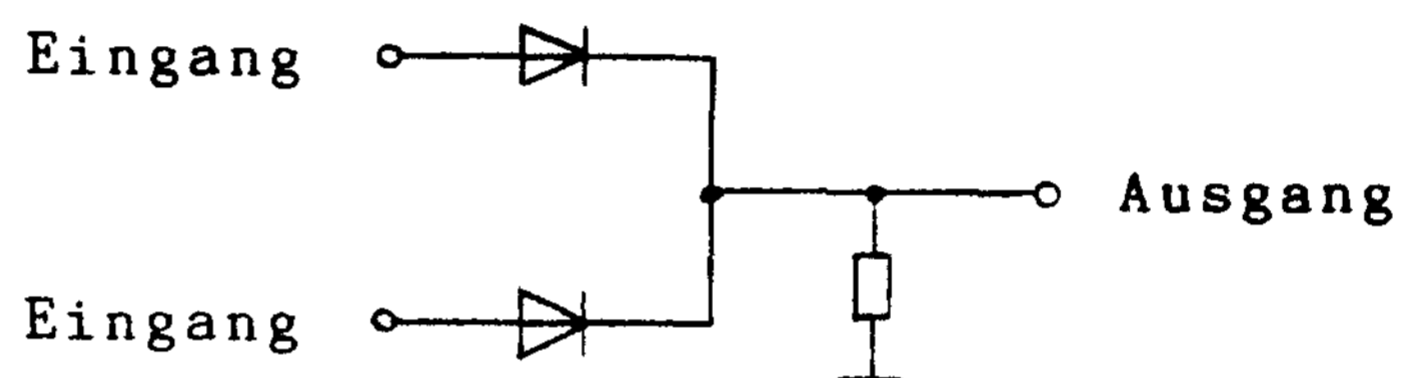


Abb. 7: Aufbau eines ODER-Gliedes aus einzelnen Bauelementen

- Pin 7 : 0V/ 5V Da sich der 4051 im Digitalbetrieb befindet, liegen hier 0 V an. Im Analogbetrieb würden hier -5 V anliegen.
- Pin 8 : 0V Masseanschluß
- Pin 9 : E/A 2,
- Pin 10: E/A 1,
- Pin 11: E/A 0 Die Entscheidung, welche der acht Eingangsleitungen die Information für die Datenleitung liefert, treffen die Ein-/ Ausgabeadressen. Mit diesen drei Leitungen kann man acht verschiedene Adressen anwählen. (Siehe: Kapitel 2.1.)
- Pin 12: E3 Eingangsleitung Nr. 3
- Pin 13: E0 Eingangsleitung Nr. 0
- Pin 14: E1 Eingangsleitung Nr. 1
- Pin 15: E2 Eingangsleitung Nr. 2
- Pin 16: 5V Betriebsspannung

2.2.3. DER AUSGABEBAUSTEIN MC 14599

Der Ausgabebaustein MC 14599 ist hier ein 1-auf-8-Demultiplexer mit Speicherwirkung. Demultiplexer haben die Aufgabe, das am Eingang erscheinende Signal zu einem bestimmten Ausgang durchzuschalten. In diesem Fall wird die Information der Datenleitung, je nach Ein-/Ausgabeadresse, auf eine der acht Ausgangsleitungen geschickt und solange gehalten, bis die Information entweder durch Drücken des Reset-Tasters gelöscht wird oder bis eine andere Information auf diese Leitung kommt. Abb. 8 zeigt die Pinbelegung des MC 14599.

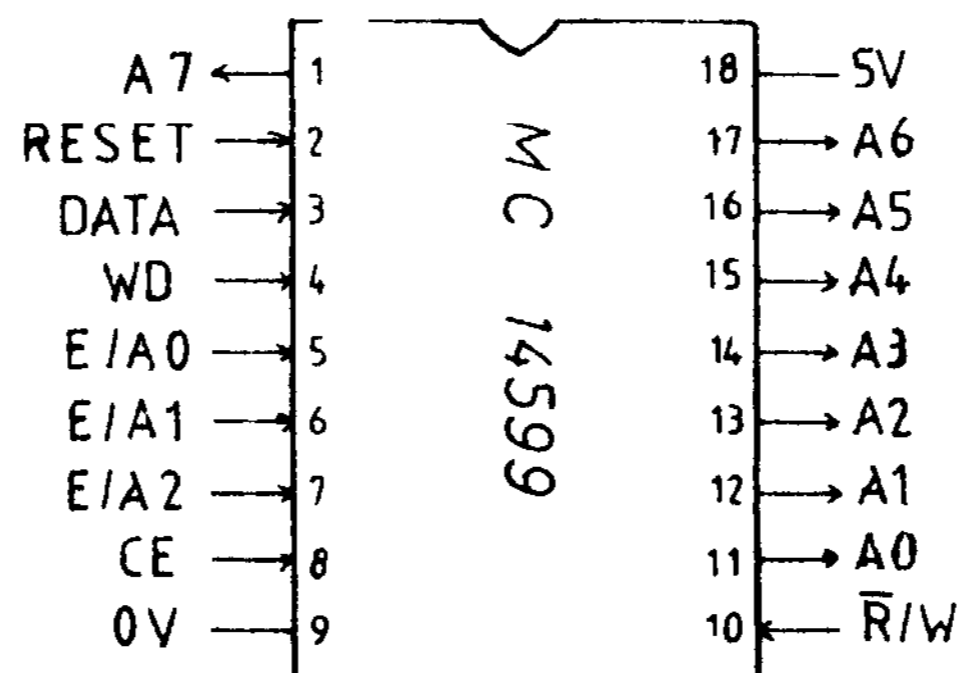


Abb. 8: Pinbelegung des MC 14599

Bedeutung der einzelnen Beine:

- Pin 1: A7 Ausgangsleitung Nr. 7
- Pin 2: RESET Normalerweise ist dieses Bein über einen Widerstand mit 0 V verbunden. Verbindet man es aber mit Hilfe eines Tasters (Reset-Taster) mit 5 V, dann werden alle Informationen an den Ausgangsleitungen gelöscht.
- Pin 3: DATA Die Datenleitung ist der Eingang des Bausteins. Die Information für einen der acht Ausgänge kommt über diese Leitung von der CPU herein.
- Pin 4: WD Write Disabled, Schreiben unfähig machen: Dieses Bein ist mit dem Takt verbunden, da der 14599 immer dann Daten ausgeben soll, wenn er über Pin 8 angesprochen wird.
- Pin 5: E/A 0,
- Pin 6: E/A 1,
- Pin 7: E/A 2 Die Entscheidung, welche der acht Ausgangsleitungen die Information der Datenleitung erhält, treffen auch hier, wie beim Eingabebaustein, die Ein-/Ausgabeadressen.
- Pin 8: CE Chip Enable, Chip befähigen: Ein 5 V-Signal an diesem Eingang bewirkt eine Freigabe des Bausteins. Da der 14599 nur arbeiten soll, wenn die CPU Daten über die Datenleitung ausgibt, ist Pin 8 mit der Schreib-/ Leseleitung verbunden. Bei der Datenausgabe ist diese Leitung nämlich auf 5 V.
- Pin 9: 0V Masseanschluß

Pin 10: \bar{R}/W Da der 14599 hier nur als Ausgabebaustein verwendet werden soll, kann dieser Anschluß direkt mit Pin 8 verbunden werden. Bei 0 V an Pin 10 würde der 14599 einen Eingabebaustein darstellen.

Pin 11: A0 Ausgangsleitung Nr. 0
 Pin 12: A1 Ausgangsleitung Nr. 1
 Pin 13: A2 Ausgangsleitung Nr. 2
 Pin 14: A3 Ausgangsleitung Nr. 3
 Pin 15: A4 Ausgangsleitung Nr. 4
 Pin 16: A5 Ausgangsleitung Nr. 5
 Pin 17: A6 Ausgangsleitung Nr. 6
 Pin 18: 5V Betriebsspannung

2.2.4. DIE EINGABETASTATUR

Die Eingabetastatur enthält vierzehn Tasten für die Befehle und acht Tasten für die Ein-/ Ausgabeadressen. Der Befehlscode und die Adressen werden durch eine Diodenmatrix codiert. Jeder Befehl und jede Adresse besteht aus Zahlen im Binärsystem, deren Stelle durch 0 V ("0") bzw. 5 V ("1") dargestellt werden. Was aber ist eine "Codierung durch eine Diodenmatrix"? Abb. 9 zeigt dafür ein Beispiel. Der Befehl IEN ist durch 1010 und der Befehl OEN durch 1011 codiert. Zur Codierung ist es nur notwendig, die Stellen, die "0" werden sollen, mit 0 V zu verbinden, denn durch Pull up-Widerstände sind alle Leitungen schon mit 5 V verbunden. Wird der Taster für den IEN-Befehl gedrückt, so liegt an Pin 4, 5, 6 und 7 der CPU die Bitfolge 1 10 an und die Kontrolleinheit der CPU erkennt, daß der Befehl IEN gemeint ist. Genauso wie der IEN-Befehl und der OEN-Befehl sind auch die anderen zwölf Befehle mit Hilfe der Diodenmatrix codiert. Für jede "0" im Befehlssatz wird also eine Diode benötigt. Somit sind 31 Dioden nötig.

An dem Beispiel in Abb. 9 wird auch deutlich, warum Dioden erforderlich sind. Wird nicht der IEN- sondern der OEN-Taster gedrückt, so wären ohne Dioden die Leitungen Nr. 6 und Nr. 4 miteinander gekoppelt und die falsche Kombination 1010 würde anliegen.

Die Dioden dienen also zur Entkopplung der einzelnen Tasten und Leitungen.

Die Ein-/ Ausgabeadressen sind ebenfalls durch eine Diodenmatrix verdrahtet. Sie sind im Dualsystem codiert (Siehe: Kapitel 2.1.).

Man erkennt, daß man für die Ein-/ Ausgabeadresse Nr. 7 keinen Taster benötigt, da sie durch 111 codiert ist und man keine Leitung auf 0 V herunterziehen braucht. Der Taster ist trotzdem vorhanden, wenn er auch nicht beschaltet ist, um einen unerfahrenen Programmierer nicht zu verwirren.

Für die Ein-/ Ausgabeadressen sind die Leitungen Nr. 0, 1 und 2 vorgesehen. Es sind 12 Dioden für die Codierung der Adressen nötig. Für eine Erweiterung der Adressen kann die Leitung Nr. 3 verwendet werden (Siehe: Kapitel 2.1.).

Zu jedem Befehl gehört eine Ein-/ Ausgabeadresse. Sie ergeben zusammen einen Programmschritt. Der Befehl gelangt zur CPU und die dazugehörige Ein-/ Ausgabeadresse gelangt zum Eingabe- und

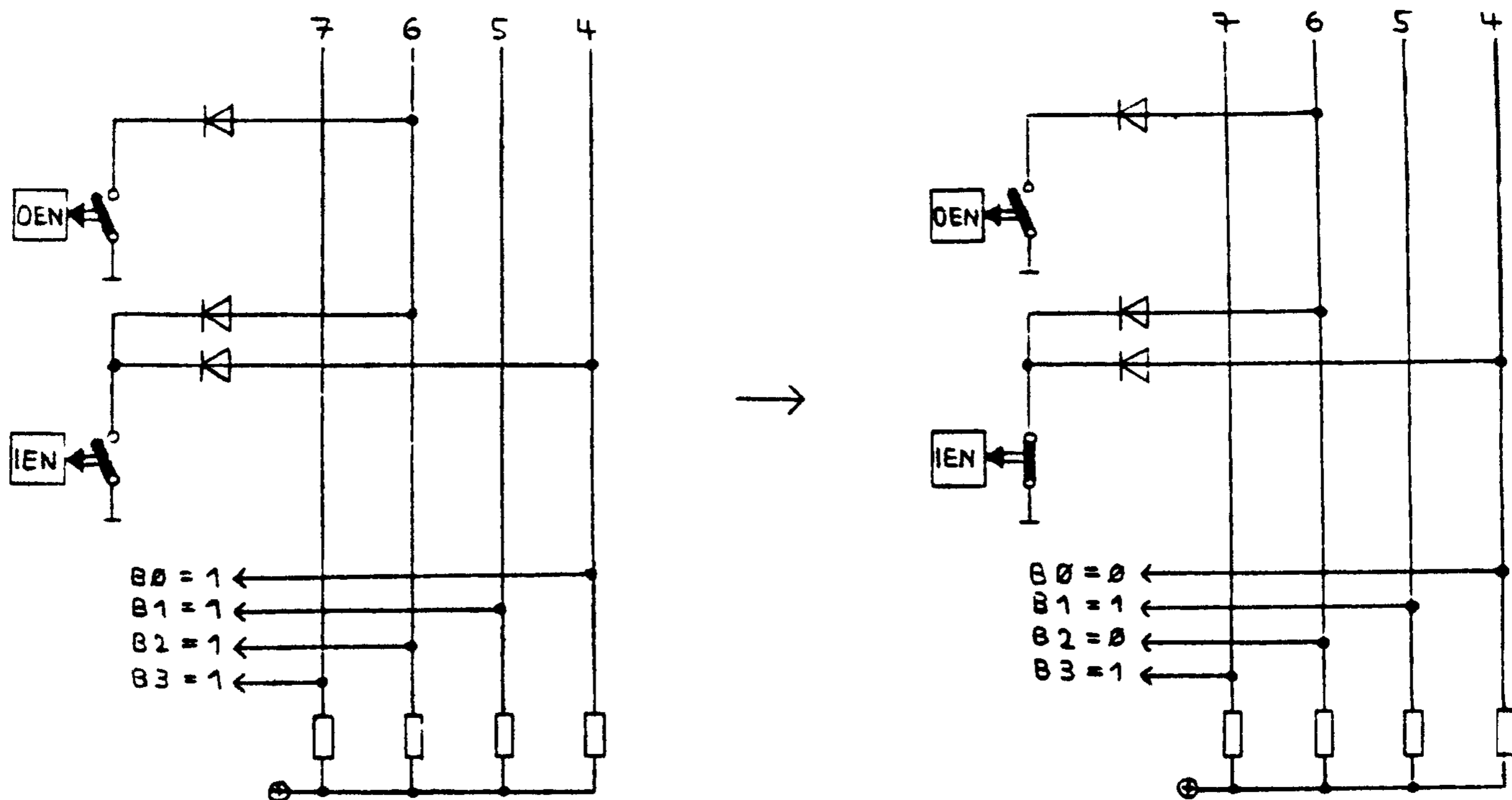


Abb. 9: Beispiel für die Codierung zweier Befehle

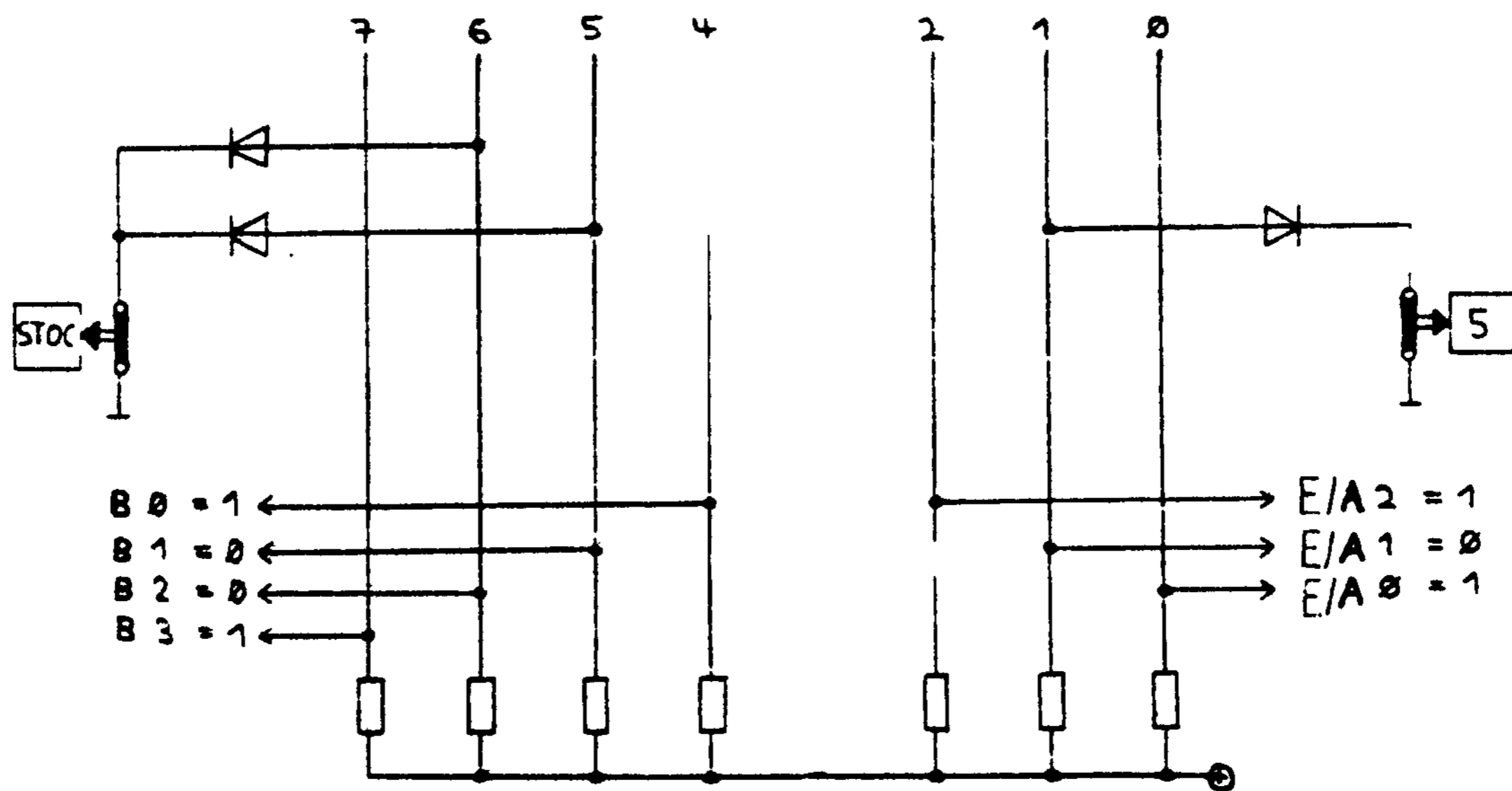


Abb. 10: Beispiel für die Codierung eines Programmschrittes

zum Ausgabebaustein. Abb. 10 zeigt die Codierung eines Befehls mit seiner Adresse. Als Beispiel wird der Programmschritt STOC 5 verwendet. Der Befehl STOC wird durch die Bitfolge 1001 codiert und die Ein-/ Ausgabeadresse Nr.5 ist im Dualsystem die Kombination 101.

Die Eingabetastatur kann auch durch sieben Schalter ersetzt werden. Man muß dann per Hand den Code für jeden Programmschritt einstellen. Durch Schließen eines Schalters wird die dazugehörige Leitung auf "0" gesetzt. Ist der Schalter offen, so ist die Leitung über einen Pull up-Widerstand mit 5V verbunden, d.h., die Leitung wird auf "1" gesetzt.

2.2.5. DER PROGRAMMSPEICHER 2112

Der Programmspeicher 2112 ist ein 256 * 4-Bit-Speicher mit wahlfreiem Zugriff (engl. Random Access Memory-RAM). Das RAM stellt einen Schreib-/ Lese-Speicher dar, d.h., der Speicherinhalt kann beliebig oft verändert werden. Neben dem RAM gibt es noch unter anderem einen Nur-Lese-Speicher (engl. Read Only Memory-ROM). Das ROM ist ein Festwertspeicher, d.h., der gespeicherte Inhalt kann nicht verändert werden. Es ist einmalig - bei seiner Herstellung - programmiert.

Die Speicherzellen des 2112 sind D-Flipflops (Siehe: Kapitel 2.2.1). Der Speicherinhalt geht leider beim Abschalten der Versorgungsspannung verloren.

Für die Speicherung eines Befehls und seiner Ein-/ Ausgabeadresse werden sieben Speicherzellen benötigt. Daher werden zwei Speicher 2112 (a) und 2112 (b) verwendet. Abb. 11 zeigt die Pinbelegung der beiden Bausteine.

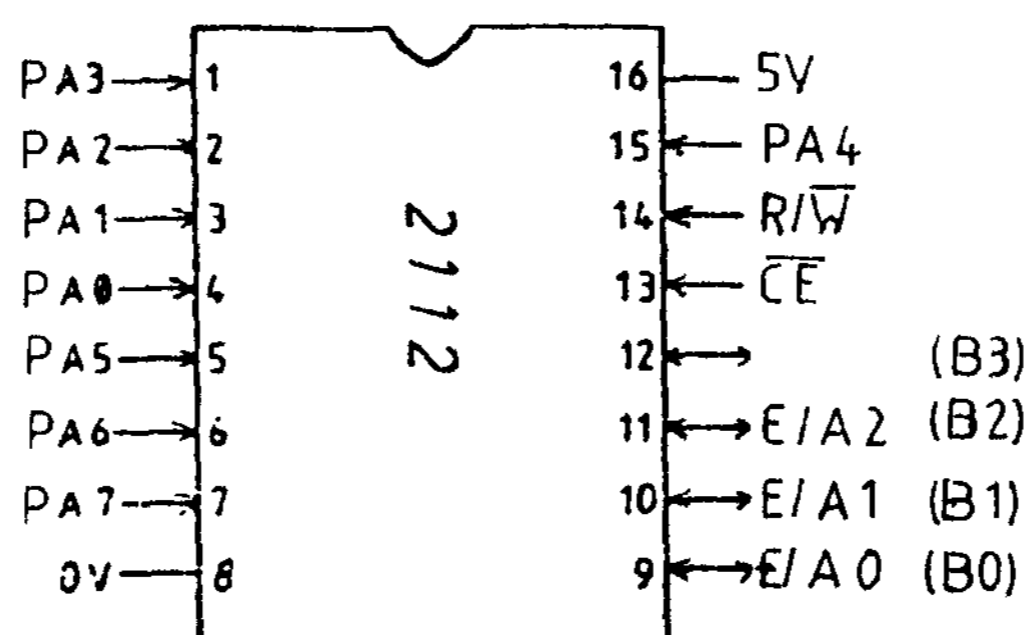


Abb.11: Pinbelegung des 2112 a (2112 b)
 "A" steht für Ausgang
 "PA" steht für Programmadresse

Bedeutung der einzelnen Beine:

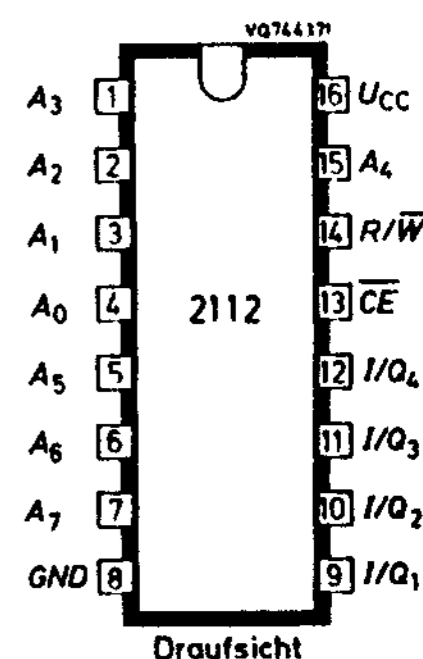
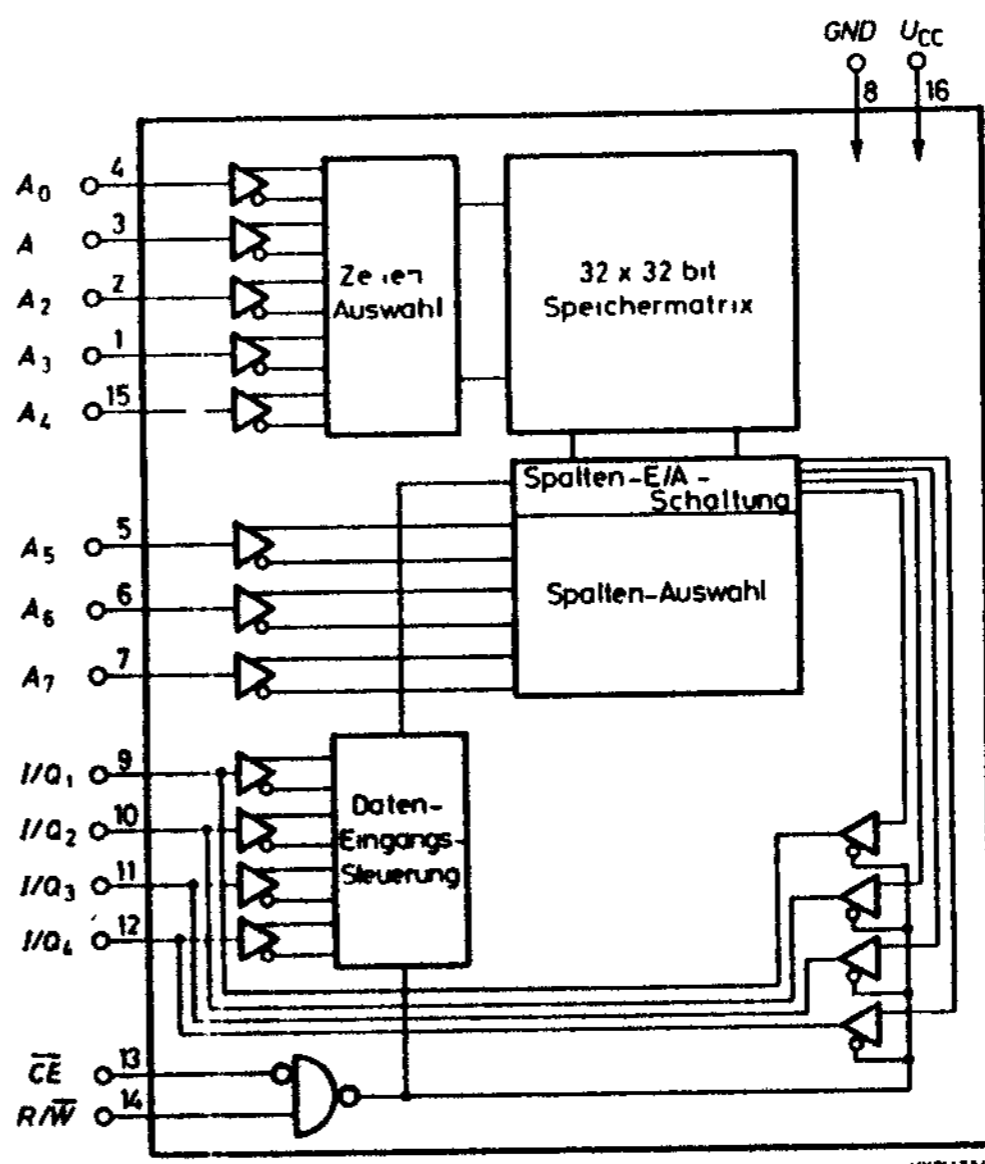
- Pin 1: PA3,
 Pin 2: PA2,
 Pin 3: PA1,
 Pin 4: PA0,
 Pin 5: PA5,
 Pin 6: PA6,
 Pin 7: PA7,
 Pin 15: PA4 Programmadressen
 Pin 8: 0V Masseanschluß
 Pin 9: E/A0 (B0),
 Pin 10: E/A1 (B1),
 Pin 11: E/A2 (B2),
 Pin 12: - (B3) Beim 2112 (a) liegen an Pin 9 ... 11 die Ein-/Ausgabeadressen an. Pin 12 kann für eine Erweiterung vorgesehen werden. Beim 2112 (b) liegen an Pin 9 ... 12 die Befehle an. Abhängig von Pin 14 wirken diese vier Anschlüsse entweder als Eingänge, d.h., die anliegenden Bits werden eingespeichert, oder als Ausgänge, d.h., die gespeicherten Werte werden ausgelesen, wobei sich der Inhalt der Speicherzellen nicht ändert. Damit bei der Ausgabe von einer "1" kein Kurzschluß entsteht, wenn von der Eingabetastatur durch Drücken eines Tasters eine "0" über dieselbe Leitung kommt, sind die acht Leitungen zur Eingabetastatur mit acht Schutzwiderständen versehen. Über diese Widerstände können dann die 5V Spannungsdifferenz zwischen der "0" und der "1" abfallen.
 Pin 13: \overline{CE} Chip Enable, Chip befähigen: Ein 0V-Signal an diesem Eingang bewirkt ein Arbeiten des Bausteins. Da der 2112 immer betriebsbereit sein soll, ist dieses Bein mit Masse verbunden.
 Pin 14: R/\overline{W} Normalerweise ist dieses Bein über einen Widerstand mit 5V verbunden. In diesem Zustand können Speicherinhalte über Pin 9 ... 12 ausgelesen werden. Verbindet man Pin 14 mit Hilfe eines Tasters (Schreib-Taster) mit 0V, dann sind Pin 9 ... 12 Eingänge, d.h., die Bits, die dort anliegen, werden eingespeichert. Alte Speicherinhalte werden dabei überschrieben.
 Pin 16: 5V Betriebsspannung

2112 2112-1 2112-2

256 x 4 bit-RAM, statisch

Technologie: N-Kanal-MOS
 Zugriffszeit: 2112: max. 1000 ns
 2112-1: max. 500 ns
 2112-2: max. 650 ns

Speisespannung: + 5 V
 Speisestrom: max. 70 mA
 2 Freigabeeingänge
 „three state“-Ausgänge
 TTL-kompatibel
 Gehäuse: F, N



2.2.6. DER PROGRAMMZÄHLER CD 4029

Die beiden Programmzähler CD 4029 (a) und CD 4029 (b) stellen hier je einen 4-Bit-Binärzähler dar, die so miteinander verbunden sind, daß sie zusammen einen 8-Bit-Binärzähler ergeben. Dieser soll die Ansteuerung der acht Eingänge für die Programmadressen der Speicher (Pin 1 ... 7 und Pin 15 des 2112) übernehmen.

Der 8-Bit-Zähler beginnt bei der Adresse 0, zählt hoch bis zur Adresse 255 und fängt wieder bei 0 an zu zählen.

Es muß beachtet werden, daß der Zähler keine besondere Rücksetzmöglichkeit hat. Das Rücksetzen auf die Programmadresse 0 geschieht mit Hilfe des Preset-Schalters.

Abbildung 12 zeigt die Pinbelegung der Bausteine.

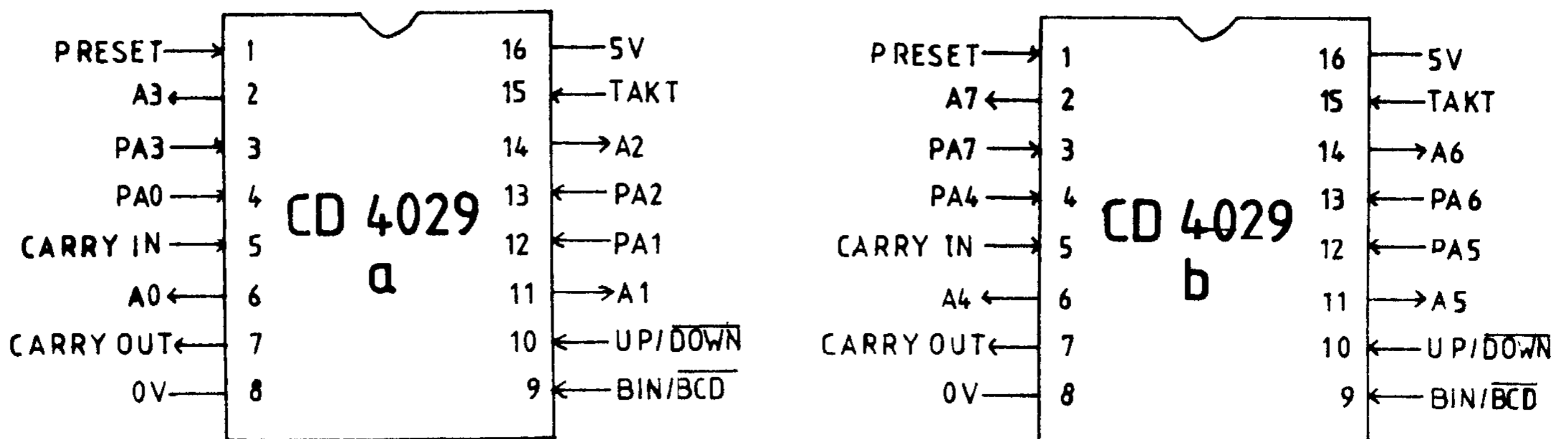


Abb. 12: Pinbelegung des CD 4029 (a) und CD 4029 (b)

Bedeutung der einzelnen Beine:

Pin 1: PRESET Normalerweise ist dieses Bein über einen Widerstand mit 0 V verbunden. Verbindet man es aber mit 5 V, so wird die Programmadresse zurückgesetzt, d.h., der Zähler zählt bei 0 weiter. Man kann Pin 1 mit Hilfe des Preset-Tasters oder durch den Befehl JUMP mit 5 V verbinden. Durch den Befehl JUMP wird in der CPU ein 5V-Impuls mit der Dauer einer Taktperiode erzeugt (siehe: Kapitel 2.2.1.). Dieses Steuersignal erscheint an Pin 9 der CPU. Von dort erhalten der 4029 (a) und der 4029 (b) über eine Leitung den 5V-Impuls.

Pin 2: PA3 (PA7)

Pin 6: PA0 (PA4)

Pin 11: PA1 (PA5)

Pin 14: PA2 (PA6) Beim 4029 (a) liegt an Pin 2, 6, 11 und 14 die untere Hälfte der Programmadresse. Beim 4029 (b) liegt hier die oberer Hälfte der Programmadresse.

Pin 3: 0V

Pin 4: 0V

Pin 12: 0V

Pin 13: 0V

- Pin 5: CARRY IN, Übertrag-Eingang
- Pin 7: CARRY OUT, Übertrag-Ausgang:
 Pin 5 des 4029 (a) ist mit 0V verbunden. Pin 7 des 4029 (a) ist mit Pin 5 des 4029 (b) verbunden. Pin 7 des 4029 (b) ist offen gelassen. Mit Hilfe dieser Anschlüsse gelingt es, aus den beiden 4-Bit-Zählern einen 8-Bit-Zähler zu bauen. Liegt Pin 5 an 0V, so zählt der Baustein bei jedem Taktimpuls. Dies ist beim 4029 der Fall. Während er von 0 (PA0=PA1=PA2=PA3=0) bis 15 (PA0=PA1=PA2=PA3=1) zählt, liegen an Pin 7 des 4029 (a) und damit an Pin 5 des 4029 (b) 5V an. Der 4029 (b) zählt also während dieser Zeit nicht (PA4=PA5=PA6=PA7=0). Ist der 4029 (a) bei 15 angekommen, so geht Pin 7 des 4029 (a) und damit Pin 5 des 4029 (b) auf 0V. Beim nächsten Taktimpuls zählen also beide Bausteine. Aus Programmadresse 15 (PA0=PA1=PA2=PA3=1, PA4=PA5=PA6=PA7=0) wird Programmadresse 16 ((PA0=PA1=PA2=PA3=0, PA4=1, PA5=PA6=PA7=0). Danach liegen an Pin 7 des 4029 (a) und an Pin 5 des 4029 (b) wieder 5V an und der 4029 (b) zählt nicht mehr. Durch Steuern der 4-Bit-Zähler mit einem gemeinsamen Takt ergibt sich ein voll synchroner 8-Bit-Zähler.
- Pin 9: Bin/BCD Binär-Zähler/BCD-Zähler:
 Da der 4029 hier ein Binär-Zähler ist, liegt dieses Bein an 5V. Binär-Zähler zählen von 0 bis 15 und fangen dann wieder bei 0 an. Würde Pin 9 an 0V liegen, so wäre der 4029 ein BCD-Zähler (Binärcode für Dezimal-Ziffern - engl. Binary Coded Decimal - BCD). BCD-Zähler zählen von 0 bis 9 und fangen wieder bei 0 an. (Ist er bei 9 angekommen, so geht Pin 7 auf 0V bis nach dem nächsten Taktimpuls.)
- Pin 10: UP/DOWN Vorwärts-Zähler/Rückwärts-Zähler:
 Da der 4029 hier ein Vorwärts-Zähler ist, liegt dieses Bein an 5V. Vorwärts-Zähler zählen von 0 bis 15 (bzw. bis 9) und fangen wieder bei 0 an. Rückwärts-Zähler zählen von 15 (bzw. von 9) bis 0 herunter und fangen dann wieder oben an.
- Pin 15: TAKT Dieses Bein ist mit Pin 14 der CPU verbunden. Der 4029 schreitet um eine Zählung fort bei jeder positiven Taktflanke von X1 (Pin 14 der CPU), wenn die Zählung freigegeben ist.
- Pin 16: Betriebsspannung

2.2.7. DIE ANZEIGEEINHEIT

Die Anzeigeeinheit besteht aus vier Bausteinen ULN 2003 und 25 Leuchtdioden. Eine Leuchtdiode (Schaltzeichen: Diode mit zwei Pfeilen) ist eine Licht aussendende Diode (engl. Light Emitting Diode-LED). 15 Leuchtdioden zeigen den Informationsgehalt der Ein- und Ausgänge, vier den Befehlscode, drei die Nummer der Ein- oder Ausgabeleitung im Dualsystem, eine die Information auf der Datenleitung, eine den Inhalt des Ergebnisregisters und eine den Takt.

Ein ULN 2003 ist im Prinzip ein 7-fach-Inverter. Er hat sieben Eingänge und sieben Ausgänge. Jeder Eingang ist über einen Inverter mit einem Ausgang verbunden. Die Inverter haben einen offenen Kollektor (engl. open collector). Ein solcher Schaltkreis bewirkt folgendes: Liegt am Eingang des Inverters eine "1", so liegt am Ausgang eine "0". Liegt aber am Eingang eine "0", so ist der Ausgangszustand normalerweise undefiniert. Eine Schaltung, die an diesen Ausgang angeschlossen werden kann, bestimmt dann den Ausgangszustand des Inverters.

Abb. 13 zeigt das Prinzip der Ansteuerung einer Leuchtdiode.



Abb. 13: Prinzip der Ansteuerung einer Leuchtdiode

Hier ist der Ausgang über einen Widerstand und eine Leuchtdiode mit 5V verbunden, so daß er auf 5V gezogen wird. Der Widerstand dient dazu, den Strom der Leuchtdiode zu begrenzen. Damit auch der Ausgangsstrom des Bausteins, z.B. des Ausgabebausteins, der mit dem Inverter verbunden ist, nicht zu groß wird, ist vor dem Inverter noch ein Widerstand geschaltet.

Mit Hilfe dieser Schaltung wird folgendes erreicht: Liegt am Eingang des Inverters eine "1", so liegt am Ausgang eine "0", d.h., die Leuchtdiode leuchtet. Leuchten bedeutet "1". Liegt am Eingang des Inverters eine "0", so liegt am Ausgang eine "1", d.h., die Leuchtdiode leuchtet nicht, was eine "0" bedeutet. Der Zustand der Leuchtdiode zeigt uns also die Information der Leitung an, die vor den Inverter geschaltet ist.

Bei 25 Schaltungen, die alle nach dem in Abb. 13 gezeigten Prinzip aufgebaut sind, können 25 Informationen angezeigt werden. Da jeder ULN 2003 sieben Inverter enthält, benötigt man vier Bausteine. Die Pinbelegung der vier Bausteine kann in Abb. 25 nachgeschaut werden.

Abb. 14 zeigt den prinzipiellen Aufbau eines Bausteins.

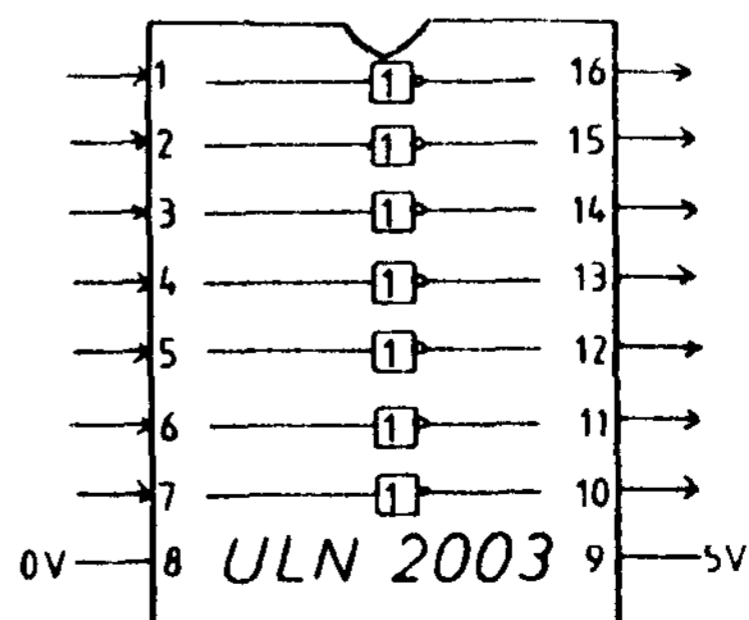


Abb. 14: Prinzipieller Aufbau des ULN 2003

2.2.8. DIE TAKTEINHEIT

Drei Baugruppen sind für die Erzeugung des Taktes zuständig: Eine bistabile Kippstufe, ein Timer und der Taktoszillator der CPU.

Welche Baugruppe gerade ausschlaggebend für den Takt ist, bestimmt die Stellung des Taktwahlschalters.

1. Stellung "Hand-Takt"

In der Stellung "Hand-Takt" wird das Taktsignal mit einer bistabilen Kippstufe erzeugt. Abb. 15 zeigt den Aufbau einer bistabilen Kippstufe mit Taster.

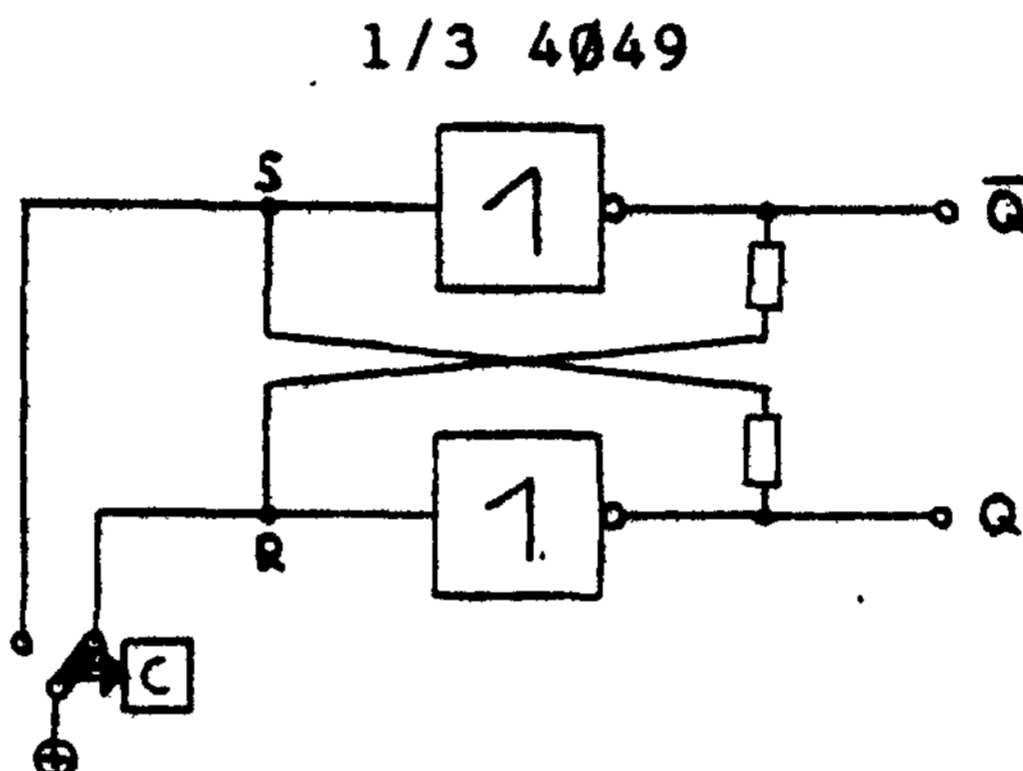


Abb. 15: Bistabile Kippstufe mit Taster

Normalerweise ist der Eingang R (Reset) der bistabilen Kippstufe über den Takt-Taster mit 5 V verbunden. Dadurch ist der Ausgang Q auf "0". Der Eingang S (Set), der mit dem Ausgang \bar{Q} verbunden ist, liegt auch auf "0" und damit der Ausgang Q auf "1".

Drückt man den Takt-Taster, so ist der Eingang S mit 5 V verbunden. Dadurch ist der Ausgang \bar{Q} auf "0", der Eingang R, der mit dem Ausgang \bar{Q} verbunden ist, auf "0" und der Ausgang Q auf "1". Nach dem Loslassen des Takt-Tasters ist der ursprüngliche Zustand wieder hergestellt. Während des Drückens beziehungsweise Loslassens gibt es eine kurze Übergangszeit, in der keiner der beiden Eingänge R oder S mit 5 V verbunden ist. Während dieser Zeit bestimmen die Ausgänge Q und \bar{Q} die Werte der Eingänge R und S über die beiden Rückkopplungen von \bar{Q} nach R und Q nach S. Der letzte Zustand der bistabilen Kippstufe bleibt also erhalten. Dadurch wird verhindert, daß Störungen wie Prellen, Kontakt-schwierigkeiten beim Betätigen des Takt-Taster auftreten.

In den beiden Rückkopplungszweigen befinden sich zwei Schutzwiderstände. Beim Drücken des Tasters springt z.B. das Signal am Eingang S von "0" auf "1". Der Ausgang \bar{Q} und damit der Eingang R reagieren nicht sofort auf diese Änderung, da das Durchlaufen des Signals durch den Inverter eine gewisse Zeit beansprucht (Laufzeit). Genauso lange braucht das Signal am Eingang R, um über den anderen Inverter auf den Ausgang Q zu gelangen. Erst nach einer Verzögerungszeit, die etwa gleich der doppelten Laufzeit ist, springt also das Signal am Ausgang Q von "0" auf "1". Während dieser Verzögerungszeit liegt am Eingang S

ein "1" und am Ausgang Q eine "0". Da aber Ausgang Q mit Eingang S verbunden ist, muß ein Widerstand dazwischen geschaltet werden, damit kein Kurzschluß entsteht. Die gleiche Problematik zeigt sich auch beim Loslassen des Tasters. Es muß also auch zwischen Eingang R und Ausgang \bar{Q} ein Schutzwiderstand geschaltet werden.

Der Ausgang Q der bistabilen Kippstufe kann durch den Taktwahlschalter mit Pin 13 der CPU verbunden werden.

2. Stellung "Langsam-Takt"

In der Stellung "Langsam-Takt" wird das Taktsignal eines Timers wirksam. Hier wird der Timer NE 555 eingesetzt, der mit den Widerständen R_1 und R_2 und dem Kondensator C beschaltet ist:

Abb. 16.

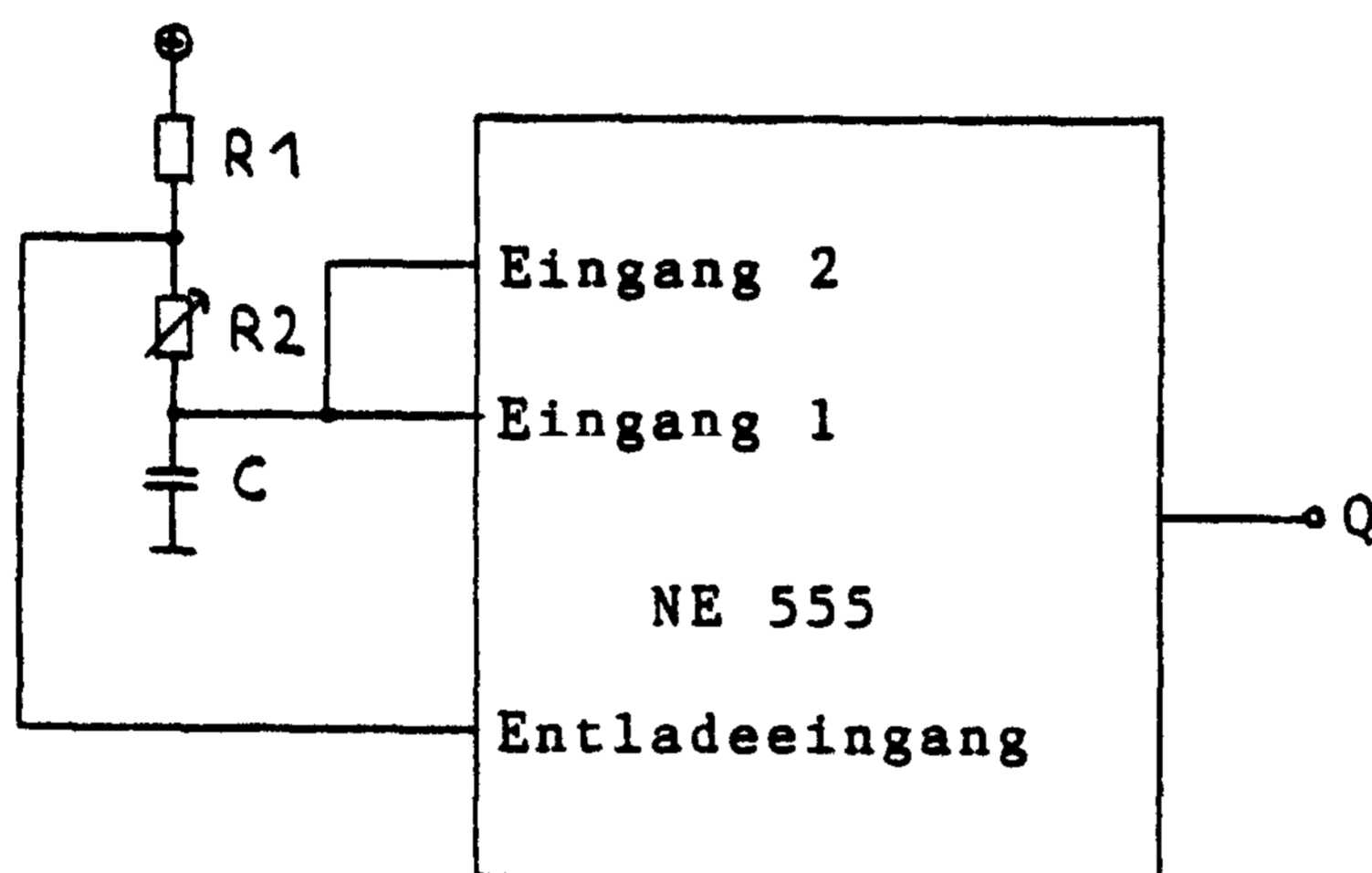


Abb. 16: Timer NE 555 mit prinzipieller Beschaltung

Dieser Kondensator wird von der Betriebsspannung über die beiden Widerstände aufgeladen, d.h., die Spannung am Kondensator steigt. Diese Spannungen liegen an zwei Eingängen des Timers. Im Timer werden sie mit zwei fest vorgegebenen Spannungen verglichen: Spannung am Eingang 1 mit $1/3$ der Betriebsspannung, Spannung am Eingang 2 mit $2/3$ der Betriebsspannung. Es sind also drei Fälle zu unterscheiden: 1. Die Kondensatorspannung ist kleiner $1/3$ der Betriebsspannung, am Ausgang des Timers liegt eine "1". 2. Die Kondensatorspannung liegt zwischen den beiden Vergleichsspannungen des Timers, der Ausgangszustand des Timer ändert sich nicht. 3. Ist die Kondensatorspannung größer als $2/3$ der Betriebsspannung, geht der Ausgang Q des Timers auf "0". Dadurch wird der Kondensator über den Widerstand R_2 durch den Entladeeingang entladen, d.h., die Kondensatorspannung sinkt. Wird die untere Vergleichsspannung erreicht, geht der Ausgang des Timers wieder auf "1". Der Entladevorgang stoppt und der Aufladevorgang setzt wieder ein. Auf diese Weise ändert sich der Ausgangszustand periodisch. Dabei werden die Dauer der beiden Ausgangszustände durch die Größen der äußeren Beschaltung bestimmt. Große Werte bewirken z.B. eine langsame Änderung. Die

Geschwindigkeit kann am besten mit Hilfe eines variablen Widerstandes (Potentiometer) eingestellt werden. Da der Widerstand R_2 am Auflade- und am Entladevorgang beteiligt ist, eignet sich dieser besonders gut dafür.

Der Zusammenhang zwischen der Ausgangsfrequenz f (Anzahl der Taktperioden pro Sekunde) und der äußeren Beschaltung (R_1 , R_2 , C), wie sie in Abb. 16 dargestellt ist, lautet als Formel

$$f = \frac{1,44}{(R_1 + 2 \cdot R_2) \cdot C}$$

Anhand der Pinbelegung des Timers (Abb. 17) wird noch kurz die Bedeutung der einzelnen Beine erläutert. Näheres zum Timer ist im Anhang zu finden.

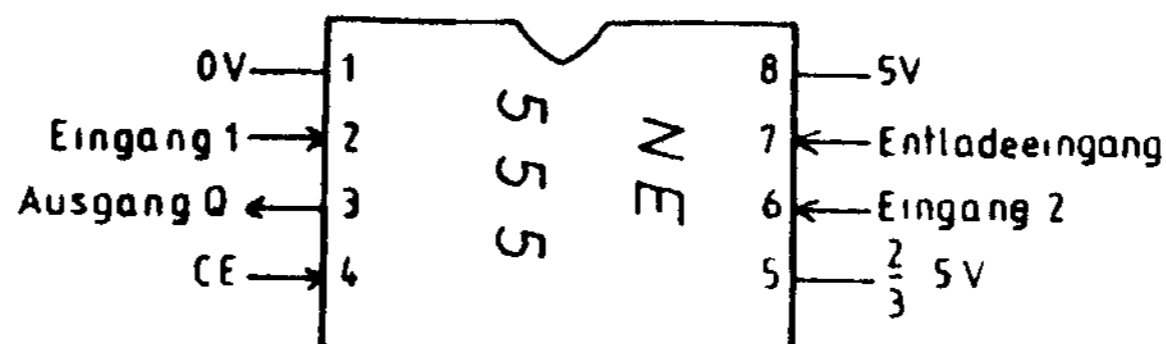


Abb. 17: Pinbelegung des Timers NE 555

Pin 1:	0 V	Masseanschluß
Pin 2:	Eingang 1	Die hier anliegenden Spannung wird mit 1/3 der Betriebsspannung verglichen.
Pin 3:	Q	Dies ist der Ausgang des Timers. Er ist über einen hochohmigen Widerstand mit Pin 13 der CPU verbunden.
Pin 4:	CE	Chip Enable, Chip befähigen: Ein 5 V Signal an diesem Eingang bewirkt ein Arbeiten des Bausteins. Damit der NE 555 betriebsbereit ist, wird dieses Signal mit 5 V verbunden.
Pin 5:	2/3 von 5 V	2/3 der Betriebsspannung im unbeschalteten Zustand dieses Bausteins.
Pin 6:	Eingang 2	Die hier anliegende Spannung wird mit 2/3 der Betriebsspannung verglichen.
Pin 7:	Entladeeingang	Beim Entladevorgang wird der Kondensator über einen Widerstand (hier R_2) und über dieses Beinchen entladen. Während dieser Zeit ist dieser Anschluß also intern mit Masse verbunden.
Pin 8:	5 V	Betriebsspannung

3. Stellung "Schnell-Takt"

In der Stellung "Schnell-Takt" kommt der Takt allein vom Taktoszillator der CPU. Mit dem Taktwahlschalter wird der Ausgang X1 des Taktoszillators (Pin 14 der CPU) über einen Widerstand mit dem Eingang X2 des Taktoszillators (Pin 13 der CPU) verbunden, d.h. rückgekoppelt. Die Werte dieses Rückkopplungswiderstandes bestimmen die Taktfrequenz: Abb. 18.

Die Unterteilung der beiden Achsen sind logarithmische, d.h., die Striche zwischen z. B. 10k und 100k bedeuten entsprechend 15k; 20k; 30k; 40k; ...90k. Die Striche zwischen z.B. 100k und 1M bedeuten entsprechend 150k; 200k; 300k; ...; 900k. Auf diese Art und Weise ist das ganze Schaubild unterteilt.

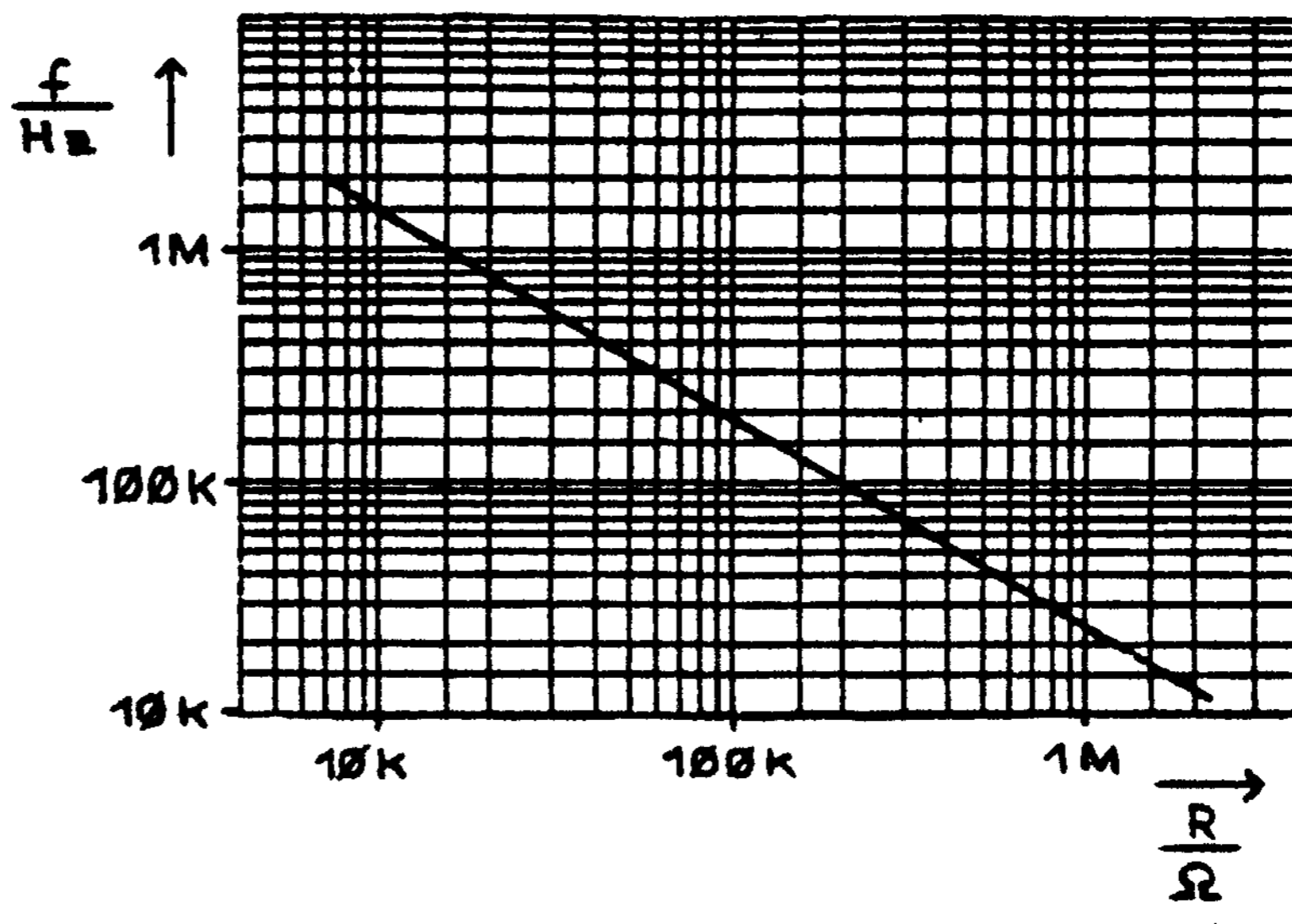


Abb. 18: Abhängigkeit der Taktfrequenz f vom Widerstand R für den Taktoszillator der CPU

Die Zusammenschaltung der gesamten Takteinheit des WDR-1-Bit-Computers zeigt Abb. 19.

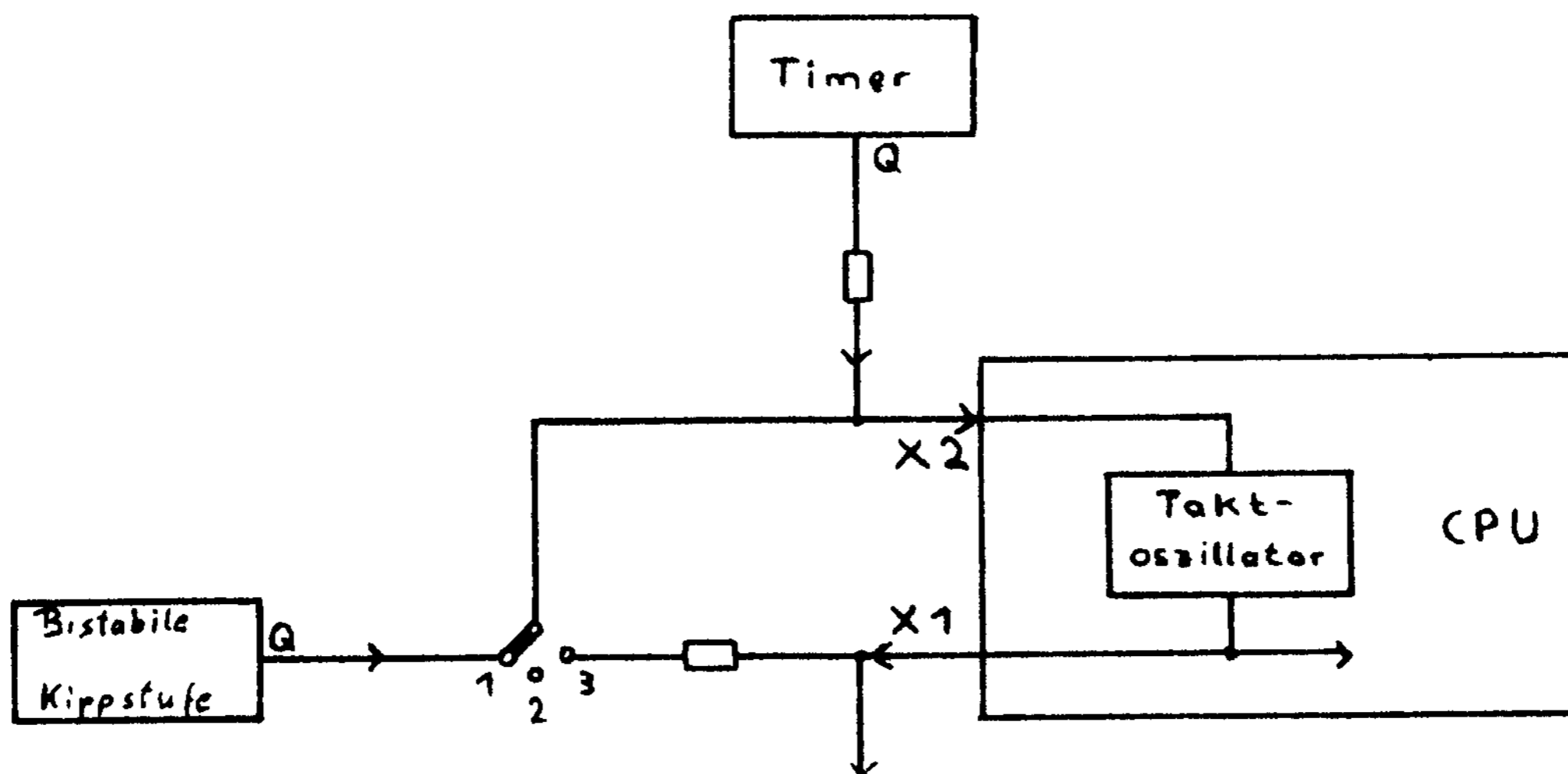


Abb. 19: Die Takteinheit

Folgende Betriebsmöglichkeiten ergeben sich:

1. In Stellung 1 (Hand-Takt) und 2 (Langsam-Takt) wird dem Taktoszillator ein fremdes Taktsignal an Eingang X2 eingeprägt. Dieses erscheint an X1 invertiert. Von dort beziehen alle Bausteine, die einen Takt benötigen, ihr Taktsignal.

In Stellung 1 sind die bistabile Kippstufe und der Timer mit X2 verbunden. Der Timer ist nur "lose" angekoppelt, da zwischen ihm und X2 ein hochohmiger Widerstand liegt. (Hochohmige Ankopplung bedeutet lose Ankopplung) Da der Ausgang Q der bistabilen Kippstufe niederohmig ist, wird dessen Taktsignal wirksam. In Ruhestellung der bistabilen Kippstufe, d.h., wenn der Takt-Taster nicht betätigt wird, liegt am Ausgang Q der bistabilen Kippstufe und damit am Eingang X2 eine "0".. Am Ausgang X1 liegt dann eine "1".

In der Stellung 2 ist nur der Timer mit X2 verbunden. Dessen Taktsignal erscheint invertiert an X1.

2. In Stellung 3 (Schnell-Takt) wird das eigene Taktsignal der CPU über einen Widerstand rückgekoppelt. Auch hier wird das Signal des Timers wegen der losen Ankopplung nicht wirksam. Der Rückkopplungswiderstand muß also wesentlich kleiner sein als der Ankopplungswiderstand des Timers.

2.2.9. DIE STROMVERSORGUNG

Die Stromversorgung wird benötigt, um aus der Netzspannung (220 V Wechselspannung) die Betriebsspannung (5 V Gleichspannung) zu erzeugen.

Die Stromversorgung besteht aus einem dem WDR-1-Bit-Computer vorzuschaltenden Transformator mit einer Ausgangsspannung von ca. 9 V. Er sollte mindestens 350 mA Strom liefern können. Die Abb.20 zeigt die Elemente der Stromversorgung ohne den Transformator.

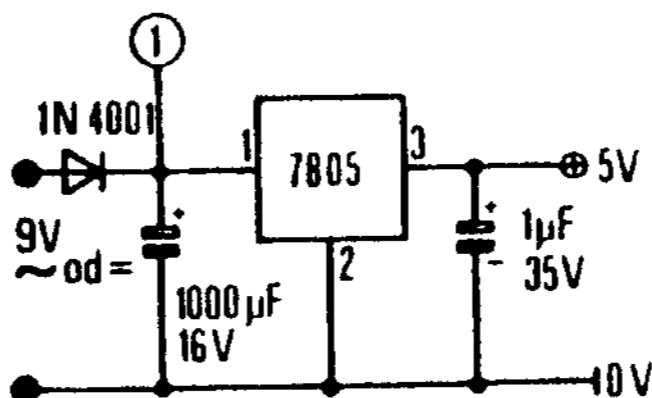


Abb. 20 Die Stromversorgung

Die Stromversorgung besteht aus einer Diode, zwei Kondensatoren und aus einem Spannungsregler.

Die Diode richtet die Wechselspannung gleich. (Wird eine Gleichspannung von ca. 9 V als Stromversorgung für den WDR-1-Bit-Computer benutzt, dient die Diode als Verpolungsschutz).

An der Diode fallen im leitenden Zustand ca. 0,6 V Spannung ab. Ist die Ausgangsspannung des Transformators um 0,6 V größer als die Kondensatorspannung, so leitet die Diode und der Kondensator wird aufgeladen, also steigt die Kondensatorspannung. Die maximale Kondensatorspannung beträgt also $12,7 \text{ V} - 0,6 \text{ V} = 12,1 \text{ V}$.

Über die nachfolgende Schaltung, dem WDR-1-Bit-Computer, würde der Kondensator aber immer wieder teilweise entladen, abgesehen davon, daß diese Spannung zu groß wäre. Um die durch den Verbraucher entstehenden Schwankungen der Kondensatorspannung aufzufangen, wird ein Spannungsregler hinter den Kondensator geschaltet. Der Spannungsregler 7805 hat drei Anschlüsse: Pin 1 ist der Eingang (hier: ca. 12,1V), Pin 2 ist mit Masse verbunden und Pin 3 ist der Ausgang (5 V). Die Ausgangsspannung des Spannungsreglers ist die stabilisierte Betriebsspannung für den WDR-1-Bit-Computer. Ein Kondensator ist zwischen Pin 2 und Pin 3 geschaltet, um Störungen, die durch Ein- und Ausschaltvorgänge verursacht werden, zu unterdrücken. Auch wenn der Spannungsregler bei Betrieb des WDR-1-Bit-Computers heiß wird, ist es nicht nötig, den Spannungsregler zum Schutz gegen Übertemperatur mit einem Kühlkörper zu versehen.

2.2.10. DER 6-FACH-INVERTER CD 4049

Die beiden Inverter des 1-Bit-Computers, die einen Teil der Takteinheit darstellen, sind Bestandteile des Bausteins CD 4049. Der 4049 ist ein 6-fach-Inverter.

Abb. 21 zeigt den Aufbau eines Bausteins.

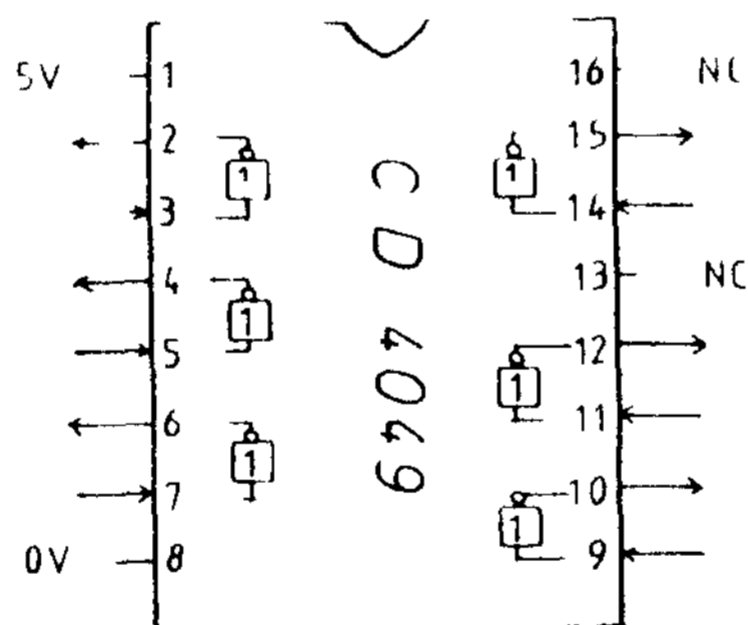


Abb. 21: Aufbau des CD 4049

Für die Takteinheit werden nur zwei Inverter benötigt. Gewählt wurden die Inverter 1 (Pin 3 und 2) und Inverter 6 (Pin 14 und 15). Pin 13 und 16 sind beim 4049 nicht angeschlossen (engl.: Not connected = NC). Die Eingänge der unbenutzten Inverter werden an Masse oder an die Betriebsspannung angeschlossen, damit diese einen definierten Eingangszustand haben.

2.2.11. DAS ODER-GLIED

Das ODER-Glied des 1-BIT-Computers, das an Pin 6 des CD 4051 angeschlossen ist, kann der Einfachheit halber aus einzelnen Bauelementen aufgebaut werden. Eine entsprechende Schaltung wurde bereits in Abb. 7 vorgestellt.

2.3. DAS ZUSAMMENWIRKEN DER FUNKTIONSEINHEITEN

Abb. 22 zeigt den Gesamtschaltplan des WDR-1-Bit-Computers. Aus ihm kann die Dimensionierung der Bauelemente entnommen werden. Die Eingabe von Programmen erfolgt durch einfache Schalter, die in einer weiteren Stufe durch eine Tastatur ersetzt werden können.

Da der Gesamtschaltplan (Abb.22) für einen Anfänger doch sehr komplex wirkt, sind in den Abb. 23 bis 25 Auszüge aus dem Gesamtschaltplan dargestellt. Dort wird das Zusammenwirken der wichtigsten Bausteine, also des Mikroprozessors, des Eingabebausteins, des Ausgabebausteins, der Programmspeicherbausteine und der Programmzählerbausteine, gezeigt, wobei jeweils andere Bausteine in den Mittelpunkt der Betrachtung rücken.

Abb. 23 zeigt die Betrachtung des Eingabebausteins, Abb. 24 zeigt die Beschaltung des Ausgabebausteins und Abb. 25 zeigt die Beschaltung der Programmspeicher- und Programmzählerbausteine. Eine gesonderte Abbildung, in der der Mikroprozessor in den Mittelpunkt rückt, erfolgt nicht, da dies schon im Gesamtschaltplan geschieht.

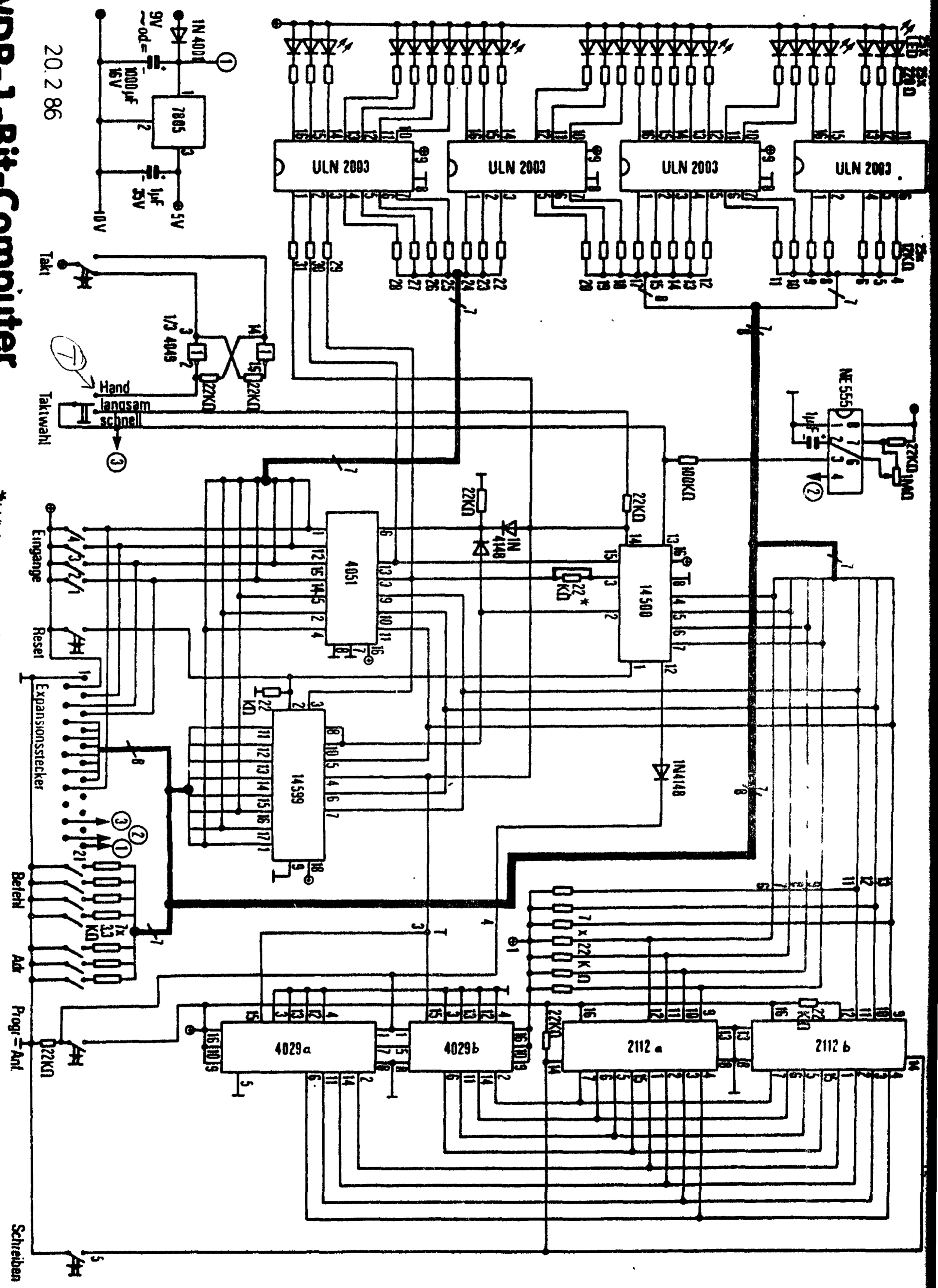
WICHTIGER HINWEIS: Der 22 Kiloohm-Widerstand der Datenleitung zwischen Pin 3 des 14500, Pin 3 des 4051 und Pin 3 des 14599 sollte fehlen, d.h. durch eine Drahtbrücke ersetzt werden, da bei einigen Eingabebausteinen 4051 der WDR-1-Bit-Computer im Schnelltakt nicht läuft. Er war ursprünglich als Schutzwiderstand gedacht.

Für den Expansionsstecker gilt folgende Anschlußbelegung:

Anschluß 1:	0 V	Masseanschluß
Anschluß 2:	5 V	Betriebsspannung
Anschluß 3:	E 3	Eingangsleitung Nr. 3
Anschluß 4:	E 2	Eingangsleitung Nr. 2
Anschluß 5:	E 1	Eingangsleitung Nr. 1
Anschluß 6:	A 7	Ausgangsleitung Nr. 7
Anschluß 7:	A 6	Ausgangsleitung Nr. 6
Anschluß 8:	A 5	Ausgangsleitung Nr. 5
Anschluß 9:	A 4	Ausgangsleitung Nr. 4
Anschluß 10:	A 3	Ausgangsleitung Nr. 3
Anschluß 11:	A 2	Ausgangsleitung Nr. 2
Anschluß 12:	A 1	Ausgangsleitung Nr. 1
Anschluß 13:	A 0	Ausgangsleitung Nr. 0
Anschluß 14:	E 4	Eingangsleitung Nr. 4
Anschluß 18:	Eingang für eine digitale Abfrage	
Anschluß 20:	Chip-Enable-Anschluß vom NE 555	
Anschluß 21	Stromversorgungsanschluß für Peripheriebausteine.	

Das Zusammenwirken der Funktionseinheiten wird durch den Takt gesteuert. Welche Baugruppen angesprochen werden, wird durch die Befehlsart bestimmt. Die Freigabebefehle und der Programmverzweigungsbefehl wurden bereits behandelt. Daher soll hier nur das Zusammenwirken der Funktionseinheiten bei den Eingabebefehlen, den Ausgabebefehlen, den Rechenbefehlen und den Steuerbefehlen erklärt werden. Wir gehen hier davon aus, daß bereits ein Programm im Programmspeicher steht und die Ein- und die Ausgabe freigegeben ist. Wie ein Programm eingegeben wird,

WDR-1-1-Bit-Computer



* Widerstand überbrücken!

Abb. 22: Schaltplan des WDR-1-1-Bit-Computers

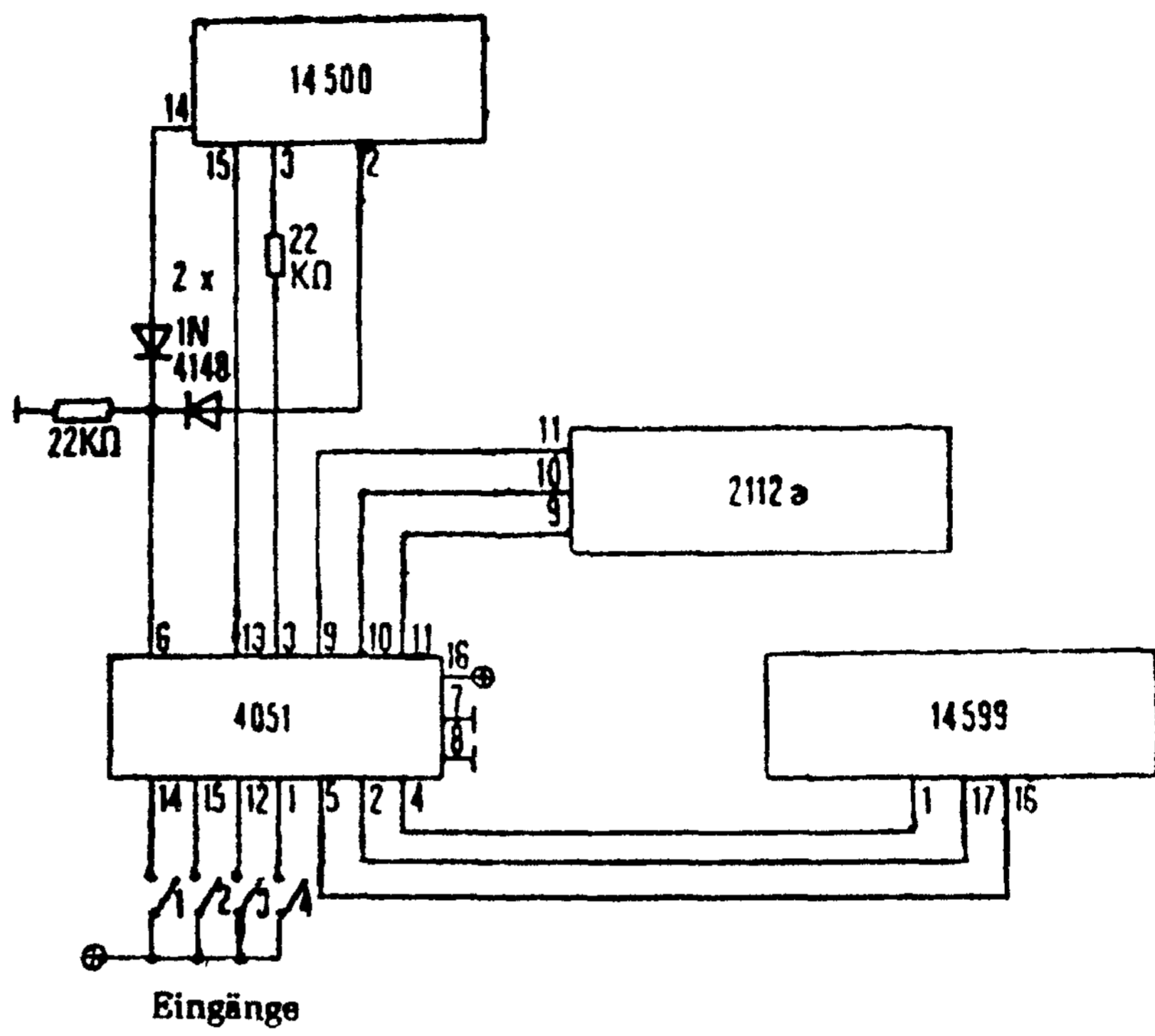


Abb. 23: Eingabeblock

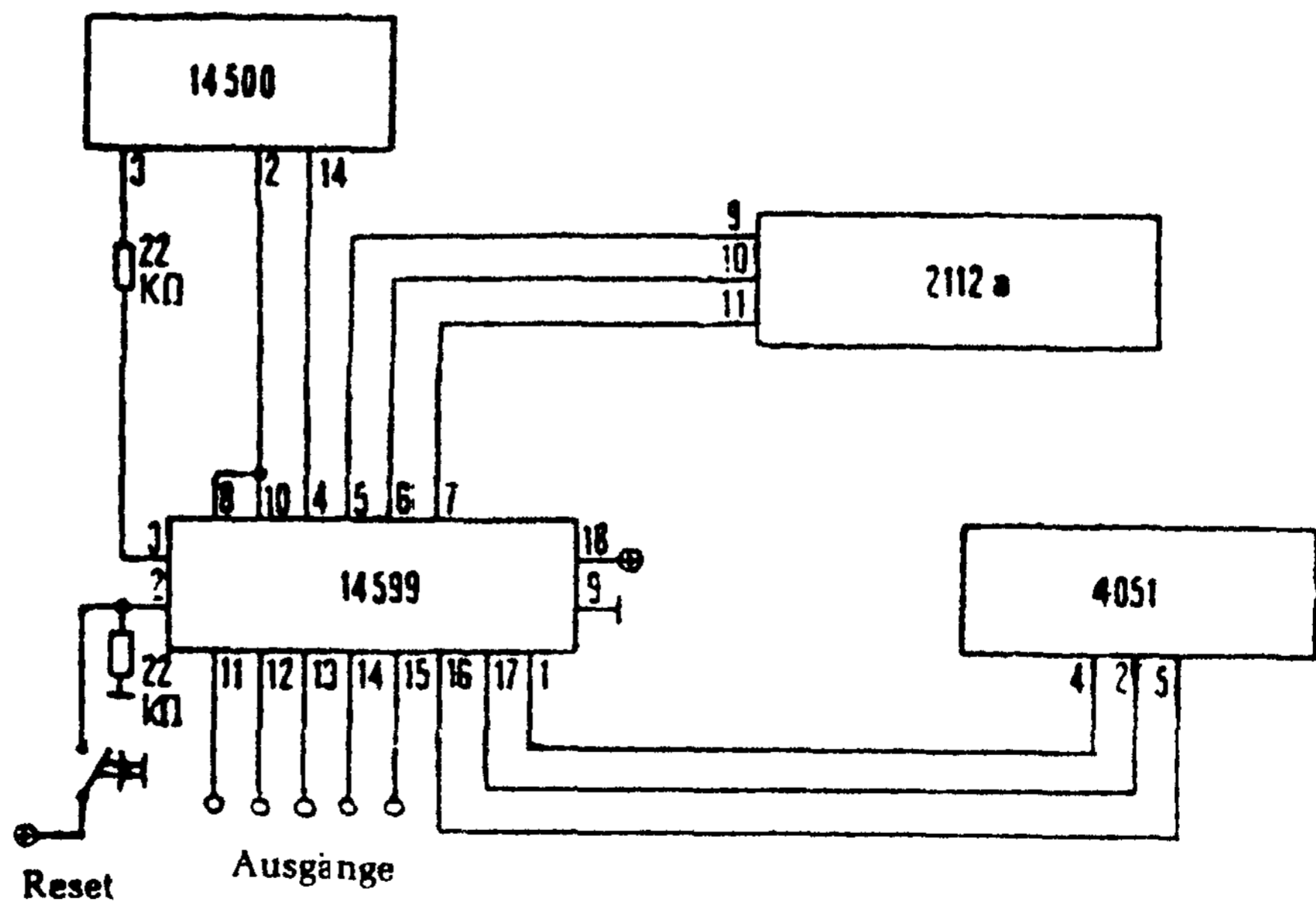


Abb. 24: Ausgabeblock

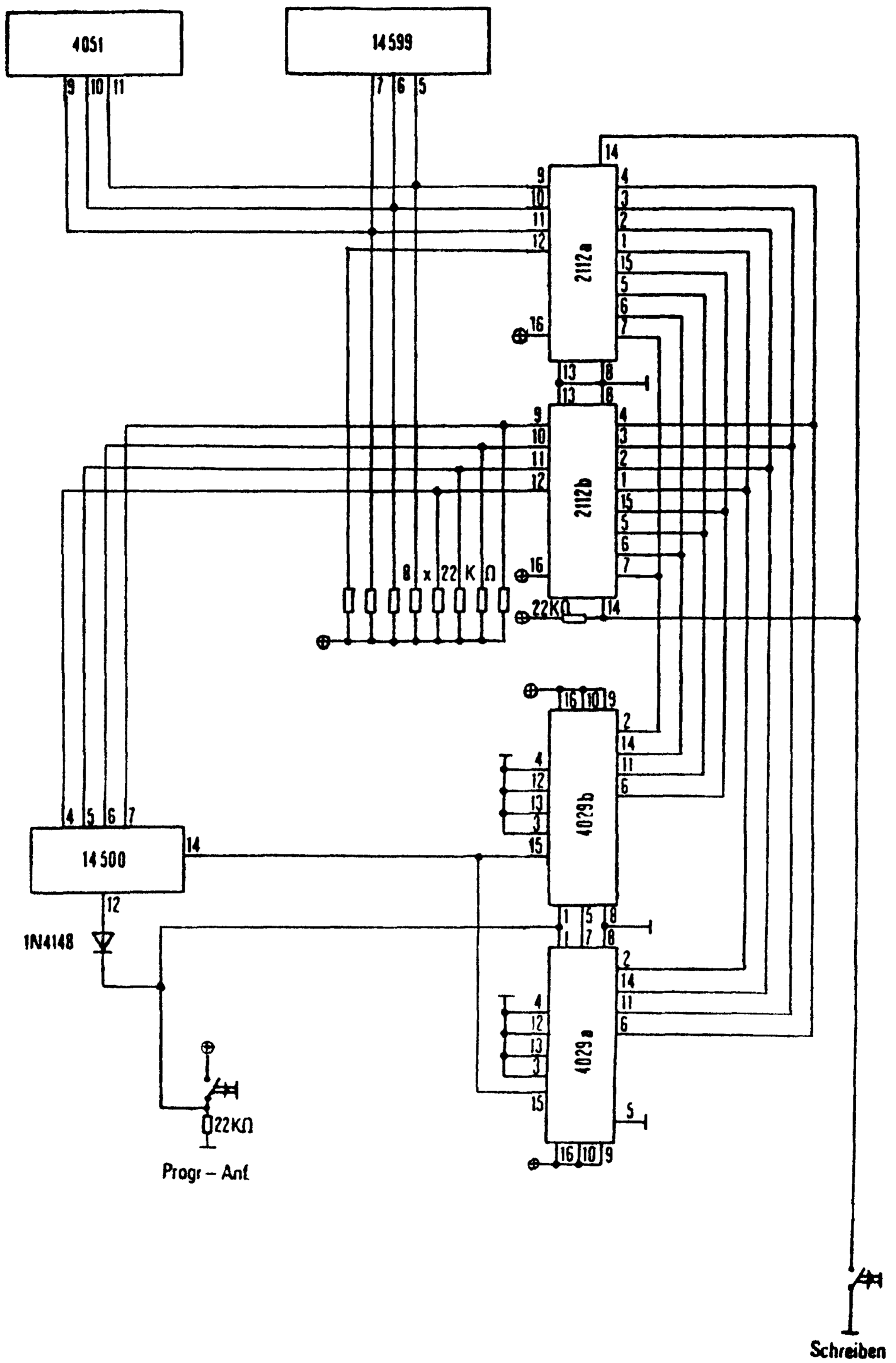


Abb. 25: Zähler- und Speicherblock