

**Anleitungsbuch**  
**für das**  
**Microprozessor Lehrsystem Microtronic**

***Diplomarbeit***  
***von***  
***Jörg Vallen***

**betreuender Dozent: Prof. Dr. Lüdecke**

**Fachhochschule des Saarlandes**

## Inhaltsverzeichnis

Vorwort	Seite 3
1. Einleitung	Seite 4
2. Test des Prototyps und dessen Weiterentwicklung	Seite 6
3. Die Erarbeitung eines verständlichen Anleitungsbuches	Seite 8
4. Schwierigkeiten	Seite 10
5. Überprüfung des Computers auf Erweiterungsmöglichkeiten	Seite 11

Versicherung des Diplomanden

Anleitungsbuch 1. Teil

Einführung in die Mikroprozessor-Technik

Anleitungsbuch 2. Teil

Programme zum Experimentieren mit dem Computer

## Vorwort

Im Herbst 1979 entschloß sich die Firma BUSCH GmbH & Co. KG in 6806 Viernheim zur Entwicklung eines Mikro-Computer-Lehrsystems.

Die Aufgabenstellung war vielfältig:

Das neue Mikro-Computersystem sollte es jedem Laien ermöglichen, sich durch ein Selbststudium in die Mikroprozessor-Technik einzuarbeiten.

Für praktische Übungen war die Entwicklung eines leicht bedienbaren und übersichtlichen Experimentier-Computers erforderlich.

Besondere Bedeutung wurde der Verfassung eines Anleitungsbuches beigemessen, um auch einem Laien ohne Vorkenntnisse den Einstieg in diese moderne Technik zu ermöglichen. Durch leicht verständliche Texte und praxisbezogene Versuchsprogramme mußten die Grundbegriffe, die Arbeitsweise und Anwendungsmöglichkeiten des Mikroprozessors dargestellt werden.

An dieser Stelle möchte ich mich besonders bei der Firma BUSCH GmbH bedanken, die es mir ermöglicht hat, bereits in der Planungsphase aktiv an diesem interessanten Projekt mitzuarbeiten. Hierdurch ergab sich die Möglichkeit, daß ich mich selbst Schritt für Schritt in die Materie einarbeiten und viele praktische Erfahrungen sammeln konnte.

Mein Dank gilt auch der Firma MRT (Meß- und Regeltechnik) in Kaisersbach, welche die Hardware und Software (Betriebssystem) des Experimentier-Computers den Vorstellungen der Firma BUSCH entsprechend entwickelt hat. Durch mehrtägige Besuche und Mitarbeit bei der Firma MRT konnte ich mich in die Mikroprozessor-Technik weitgehend einarbeiten, wobei ich durch eine ganze Reihe von Änderungs- bzw. Verbesserungsvorschlägen mithelfen konnte, ein optimales Experimentier-System zu entwickeln.

November 1981

## 1. Einleitung

Zum Zeitpunkt des Entwicklungsbeginns für einen Experimentier- und Lern-Computer im Herbst 1979 war das Angebot von Mikro-Computern dieser Art auf dem Markt kaum anzutreffen. Es gab noch keine Geräte, welche es Nichtfachleuten ermöglichte, sich in die Mikroprozessor-Technik<sup>+</sup> einzuarbeiten.

Es wurden lediglich sogenannte Entwicklungs-Systeme von großen Halbleiter-Herstellern angeboten, um Technikern mit entsprechender Vorbildung die Möglichkeit zu geben, die Arbeitsweise eines bestimmten Mikroprozessor-Typs kennenzulernen. Dementsprechend waren diese Systeme von der elektronischen Seite ausgelegt. Die von Spezialisten verfaßten Anleitungen und Beschreibungen waren (und sind es größtenteils auch heute noch) für Nichtfachleute ohne Grundkenntnisse unverständlich.

Es war vorauszusehen, daß die vielseitigen Verwendungs- und Einsatzmöglichkeiten eines Mikro-Computers, unter der Voraussetzung, daß die Arbeitsweise eines Mikroprozessors allgemein verständlich demonstriert werden kann, großes Interesse finden werden.

Als Käuferkreis sollten zunächst technisch interessierte Laien angesprochen werden, die ganz einfach wissen möchten, wie ein Computer funktioniert, wie man ihn programmiert und wie man ihn dazu bringt, bestimmte Befehle auszuführen.

Bei den Vorarbeiten war bald zu erkennen, daß die Thematik auch für Studenten, für den Informatiker-Nachwuchs und nicht zuletzt für Techniker und Ingenieure interessant sein könnte, weil ein solcher Lern-Computer nicht nur für "Spiele und Spielereien", sondern für echte Programmierungen und Steuerungsaufgaben eingesetzt werden kann.

Mittlerweile hat es sich herumgesprochen, daß der Micro-Computer ein sehr universelles "Beschäftigungsmittel" unserer Zeit darstellt. Als Nebeneffekt stellt sich eine Verbindung für viele praktische Anwendungen ein, die vor wenigen Jahren nahezu undenkbar waren.

<sup>+</sup> Technische Fachausdrücke siehe Fachwortverzeichnis im Anleitungsbuch 1. Teil Seite 78/79

Unter diesen, bei Entwicklungsbeginn teilweise noch nicht absehbaren Kriterien, wurde in Zusammenarbeit mit der Firma Texas Instruments Deutschland (in Freising) zunächst ein Pflichtenheft ausgearbeitet, in welchem die Daten und Anforderungen festgelegt wurden, welche der Computer erfüllen sollte.

Als geeignete CPU<sup>+</sup> wurde der Texas Mikroprozessor TMS 1600 ausgewählt. Dieser Prozessor konnte durch einen maskenprogrammierten ROM<sup>+</sup>-Speicher mit verhältnismäßig großer Kapazität durch wenige zusätzliche Bauelemente zu einem Mikro-Computer ausgebaut werden. Dieser ROM-Speicher enthält das Betriebs-System des Mikro-Computers. Durch ein Festprogramm sollte das ROM so programmiert werden, daß die Ansteuerung einer Leuchtanzeige und von Peripherie-Bauelementen möglich wurde, ebenso wie die Abfrage der Tastatur, leichte Abrufmöglichkeit fest eingegebener Vorführprogramme und vor allem sollte das ROM die Auswertung eines gehobenen Maschinenbefehlssatzes übernehmen, um die Arbeit und Programmierung des Mikroprozessors für Nichtfachleute zu erleichtern.

Im Pflichtenheft wurde zunächst ein Befehlssatz von 20 Einzelbefehlen mit den wichtigsten Befehlen der verschiedenen Mikroprozessoren festgelegt. Dieser Befehlssatz war von Anfang an leicht programmierbar und übersichtlich aufgebaut.

Nachdem die Möglichkeiten des Mikroprozessors mit den Anforderungen des Pflichtenheftes abgedeckt waren, konnte im Sommer 1980 die Planungsphase beendet werden. Nun übernahm die Firma MRT die Programmierung des Betriebs-Systems und die Erstellung eines Prototyps, welcher zunächst einen Entwicklungsprozessor enthielt, als Ersatz für den später vorgesehenen maskenprogrammierten Mikroprozessor. Das Betriebs-System wurde zunächst in sogenannten EPROMs (siehe Seite 70) gespeichert, um die mehrfach zu erwartenden Änderungen des Betriebs-Systems durchführen zu können.

## 2. Test des Prototyps und dessen Weiterentwicklung

Gegen Ende des Jahres 1980 wurde der erste Prototyp mit einem funktionierenden Betriebs-System in Übereinstimmung mit dem Pflichtenheft fertiggestellt.

Meine Aufgabe war es, diesen Prototyp zu testen, um Fehler des Betriebs-Systems festzustellen. Außerdem mußte überprüft werden, ob das System in dieser Konfiguration den praktischen Anwendungsmöglichkeiten genügt, bzw. ob durch Änderungen und Erweiterungen des Befehlssatzes die Anwendbarkeit weiter verbessert werden kann.

Die ersten Test ergaben, daß alle im Pflichtenheft aufgeführten Anforderungen erfüllt wurden. Bei den folgenden Tests zeigte sich aber, daß das Betriebs-System des Mikro-Computers bei bestimmten Programmierungen, z.B. nach Durchführung der vorgesehenen "Sprungbefehle" in eine "Endlos-Schleife" kam. Das Betriebs-System überprüfte nicht mehr die Tastatur, ob eine Taste betätigt wurde, wodurch der gesamte Computer nicht mehr bedienbar war.

Es zeigte sich ferner, daß einige wichtige Befehle für die leichte und kapazitätssparende Programmierung fehlten, wie z.B. der Sprung in ein Unterprogramm, Addition, Subtraktion, logische Verknüpfungen mit einer Konstanten etc. Das Betriebs-System wurde geändert und erweitert. Die nun folgenden Testarbeiten brachten mir wesentliche Erkenntnisse der Computer-Logik, welche letztlich zur nochmaligen Erweiterung des Befehlssatzes führten, wie z.B. komplexe Befehle für die Multiplikation einer 6-stelligen Zahl, Erzeugung von Zufallszahlen, Übernahme der im Betriebs-System ständig mitlaufenden Uhrzeit etc. Hierdurch konnte die Programmierung erneut verbessert und erleichtert werden. Anstelle des ursprünglich vorgesehenen Befehlssatzes mit 20 Einzelbefehlen stand inzwischen ein Befehlssatz mit über 40 Einzelbefehlen zur Verfügung.

Nach jeder Änderung und Erweiterung des Betriebs-Systems wurde der Prototyp durch Austausch der EPROMs den neuen Gegebenheiten angepaßt.

Jede Änderung des Betriebs-Systems mußte durch viele Befehlskombinationen neu getestet werden, wobei oftmals soeben integrierte Vorteile auch wieder Änderungen beim bereits bestehenden Betriebs-System mit sich brachten. Nun mußten spezielle Prüfprogramme erstellt werden, die möglichst viele Befehlskombinationen enthielten, um evtl. doch noch vorhandene Fehler zu erkennen.

Im Frühsommer 1981 konnte die Test- und Entwicklungsphase des Betriebsprogramms abgeschlossen werden. Die Firma Texas Instruments übernahm das Betriebs-System, das auf einer Diskette gespeichert war, um die Maskenprogrammierung (siehe Seite 44) des Mikroprozessors durchzuführen.

Erst im September 1981 konnte Texas Instruments die ersten 10 Mikroprozessoren fertigstellen. Alle bis dahin erstellten Programme mußten mit diesen Prototypen erneut getestet werden, um dann eine Freigabe für die Produktion der Mikroprozessoren in größeren Stückzahlen zu erreichen. Inzwischen war auch die gesamte Hardware des Experimentier-Computers fertiggestellt, wodurch die Testläufe im endgültigen Lieferzustand möglich waren. Neben dem Betriebsprogramm und den vorgesehenen Programmierungen mußte auch überprüft werden, ob z.B. Störungen durch elektromagnetische Strahlungen (z.B. von Fernhergeräten, Einschalten von Haushaltsgeräten usw.) oder statische Aufladungen des Kunststoffgehäuses auftreten können. So wurde u.a. festgestellt, daß das System nicht einwandfrei arbeitet, wenn an den Computer-Eingängen Spannungen von mehr als 5 Volt angelegt wurden, was z.B. durch Änderung der Hardware weitgehend neutralisiert werden konnte. Die praxisbezogenen Tests ergaben u.a. auch, daß es vor allem bei großen Programmeingaben wünschenswert war, solche Programme unabhängig von der Netzstromversorgung über einen längeren Zeitraum sichern zu können. Eine solche Datensicherung wurde durch nachträglichen Austausch des RAMs gegen ein CMOS-RAM (siehe Seite 70) erreicht. Durch die sehr geringe Stromaufnahme des CMOS-RAMs konnte eine Datensicherung durch eine kleine Batterie über mehrere Wochen hinweg erzielt werden. Diese und andere Verbesserungen mußten wiederum durch Änderung der Hardware durchgeführt werden.

### 3. Die Erarbeitung eines Anleitungsbuches

---

Ab Juni 1981 konnte ich mit den Vorarbeiten für das Anleitungsbuch beginnen. Sehr bald sollte sich herausstellen, daß die Weitergabe einer möglichst verständlichen Mikroprozessor-Logik erheblich schwieriger war, als dies von mir eigentlich angenommen wurde. Durch meine intensive Beschäftigung hatte ich in der Zwischenzeit einen Wissensstand erreicht, der meine anfänglichen Schwierigkeiten vergessen ließ. Daher war es zweckmäßig, daß meine ersten Manuskripte mit einigen lernwilligen Laien bei gleichzeitiger zur Verfügungstellung eines Prototyps des Lern-Computers durchgearbeitet wurden. Auf diese Weise konnte ich erkennen, welche Mikroprozessor-Funktionen und Beschreibungen verständlich oder in der Aussage verbesserungsbedürftig waren. Einerseits ergab sich eine mühsame und zeitaufwendige Arbeitsweise, andererseits ergab sich aber die befriedigende Feststellung, daß es tatsächlich möglich war, absolute Nichtfachleute Schritt für Schritt in die Mikroprozessor-Technik einzuführen.

Auf diese Weise entstanden die zweiteiligen Anleitungsbücher, die den Mittelpunkt meiner Diplomarbeit darstellen.

Der erste Teil des Anleitungsbuches enthält eine Einführung in die Mikroprozessor-Technik, die Beschreibung der Einzelbefehle und Erklärungen der wichtigsten, immer wiederkehrenden Programm-Routinen, wie solche für mehrstellige dezimale Additionen, Subtraktionen etc. notwendig sind. Die Zusammenarbeit mit dem erwähnten Lienteam zeigte mir auch, daß es z.B. sehr wesentlich war, für wiederkehrende Begriffe und Bezeichnungen möglichst die gleichen Fachworte zu verwenden, um nicht durch eine Wortvielfalt unnötige Verwirrungen herbeizuführen. Großen Wert legte ich auch darauf, daß das unvermeidliche Lesen und Lernen durch praktische Computer-Arbeit und die hieraus resultierenden "Aha"-Erlebnisse aufgelockert wird. Durch Zusammenarbeit mit einem Graphiker (Atelier Wuthe in Weinheim) konnte ich erreichen, daß die an und für sich trockene Materie durch Einbeziehung des "Buschi"-Firmenmaskottchens teilweise zur "schmunzelnden Lektüre" geworden ist.



Der zweite Teil des Anleitungsbuches enthält überwiegend Programm-vorschläge (mit teilweise kurzgefaßten Erklärungen), um die verschiedenen Einsatzmöglichkeiten eines Mikroprozessors aufzuzeigen. Der jeweilige Programmaufbau kann durch vollständig wiedergegebene Programm-Ablaufpläne verfolgt werden. Die Programmbeschreibung beschränkt sich auf die wesentlichsten Hinweise und Feststellungen. Am Ende viel es mir schwer, die Arbeiten an diesen Anleitungsbüchern abubrechen, denn auf Grund vieler weiterer Programmideen wäre es ohne weiteres möglich gewesen, den Umfang der Bücher noch beträchtlich auszuweiten.

Alle in den Anleitungsbüchern enthaltenen Programme wurden von mir erstellt und sorgfältig getestet. Eine gewisse Problematik ergab sich vor allem bei Programmen mit vielen Berechnungen (z.B. Taschenrechner-Programm, Berechnung der Tage zwischen zwei Daten etc.). Bei solchen Programmen hätte die Anwendungsmöglichkeit einer höheren Programmiersprache eindeutige Vorteile gebracht. Durch die Programmierung mit dem zwar gehobenen, aber trotzdem maschinenähnlichen Microtronic-Befehlssatz müssen bereits bei einfachen Additionen relativ komplizierte Programme geschrieben werden. Es war zu berücksichtigen, daß die Verwendung einer höheren Programmiersprache ein wesentlich umfassenderes Betriebs-System, eine aufwendigere Hardware und eine höhere Speicherkapazität erforderlich gemacht hätte. Dies wiederum hätte den vom Hersteller vorgegebenen Verkaufspreis für ein Lern-Computersystem in erheblichem Maße beeinflußt.

Andererseits sollte die Anwendung eines Mikroprozessors gezeigt und demonstriert werden; denn entsprechend programmierte Mikroprozessoren finden auch heute noch das größte Anwendungsgebiet in Taschenrechnern. So enthalten die bekannten Taschenrechner von Texas Instruments TI 58, bzw. TI 59 ähnliche Prozessoren wie solche im Microtronic-Lern-Computer verwendet werden. Auch der Sharp-Pocket-Computer 1211 enthält ebenfalls einen 4-Bit-Mikroprozessor eines japanischen Herstellers.

Der Nachteil des maschinenähnlichen Microtronic-Befehlssatzes hat auch einen Vorteil: Durch die möglichen Bit-Manipulationen sind z.B. Steuerungsaufgaben durchführbar, die mit den meisten höheren Programmiersprachen nicht möglich sind.

#### 4. Schwierigkeiten

Gewisse Schwierigkeiten ergaben sich bei einem Programm, welches die Funktion eines Frequenz-Zählers verdeutlichen sollte (s. Anleitungsbuch 2. Teil, Seite 62).

Erste Versuchsprogramme, die einen, an einem Eingang anliegenden Takt mitzählen sollten (um die Frequenzen zu berechnen), versagten schon bei der geringen Frequenz von mehr als 4 Hz. Der Mikro-Computer arbeitet relativ langsam, die Takt-Frequenz der CPU beträgt lediglich ca. 500 kHz und das Betriebs-System nimmt einen beträchtlichen Teil der Arbeitszeit des Mikroprozessors in Anspruch. Durch Versuche konnte ich jedoch feststellen, daß die ständig mitlaufende Uhrzeit (welche ihre Takt-Impulse ebenfalls durch einen Eingang erhält) bis zu einer Frequenz von ca. 60 Hz einwandfrei mitzählen kann. Als Zählprogramm für den Frequenz-Zähler wurde daher die Uhrzeit des Betriebs-Systems verwendet und durch ein zusätzliches Programm ausgewertet. Die zu messende Versuchsfrequenz wird durch eine astabile Kippstufe erzeugt, welche durch Bauelemente aus einem Experimentier-Kasten des BUSCH-Electronic-Studios aufgebaut werden kann. Gewisse Probleme ergaben sich auch durch die begrenzt zur Verfügung stehende Anzahl von Widerständen und Kondensatoren. Die Kippstufe mußte einerseits so ausgelegt werden, daß die erzeugte Frequenz eine einwandfreie Rechteck-Frequenz ergab und nicht durch zu lange Ladezeiten eines Kondensators zu einer Sägezahn ähnlichen Schwingung verzerrt wird. Andererseits sollte das Tastverhältnis bei der größten zu messenden Frequenz (50 Hz) ungefähr 1 : 1 sein.

Eine gewisse Problematik stellt auch die Austestung von Programmen dar, die eine Echtzeit-Verarbeitung durchführen müssen, weil nach einem Programmdurchlauf ein evtl. vorhandener Fehler schwer zu lokalisieren ist. Ein Oszillograph war ein unentbehrliches Hilfsmittel, um zu kontrollieren, ob die an den Eingängen ankommenden Impulse einwandfrei sind und das Programm auf solche Signale wie gewünscht reagiert.

## 5. Überprüfung des Computer-System auf Erweiterungsmöglichkeiten

Außer den in den Anleitungsbüchern aufgeführten Programm-Möglichkeiten wurde noch eine Vielzahl weiterer Programme ausgearbeitet, um die späteren Erweiterungsmöglichkeiten des Computer-Systems zu testen. Beispiele:

Mit Hilfe des preiswerten 7-Bit-Analog-Digitalwandlers<sup>x</sup> TL 507 können relativ einfach sehr genaue Spannungs-, Strom-, Licht- und Temperatur-Messungen durchgeführt werden. Dieser A/D-Wandler arbeitet nach dem sogenannten "dual-slope" Verfahren. Eine Ansteuerung durch den Mikro-Computer ist verhältnismäßig einfach, weil der A/D-Wandler mit jedem vom Computer kommenden Impuls eine Vergleichsspannung erhöht, bis die gleiche Größe wie die zu messende Spannung erreicht ist. Sobald beide Spannungen gleich sind, wird ein Ausgang des A/D-Wandlers<sup>x</sup> high und die vom Computer intern mitgezählten Impulse können durch ein Programm ausgewertet werden. Die zu messende Spannung kann z.B. durch einen NTC-Widerstand erzeugt und durch ein entsprechendes Programm in eine Temperatur umgewandelt werden. Die Genauigkeit beträgt hierbei ca.  $0,5^{\circ}\text{C}$  und ist ausreichend um z.B. eine Raumtemperatur-Steuerung (mit einem elektrischen Heizgerät) zu programmieren.

Eine andere Erweiterungsmöglichkeit befaßt sich mit der Vervielfachung der je 4 Ein- und Ausgänge des Mikro-Computers. Durch den Aufbau einer zusätzlichen elektronischen Speicherschaltung (welche durch ein entsprechendes Programm angesprochen wird) ist es möglich, die Zahl der Ein- und Ausgänge auf 16 zu erhöhen. Dies ist wichtig,

<sup>x</sup> Analog-Digitalwandler (A/D-Wandler) dienen der Umwandlung von analogen Spannungswerten in entsprechende digitale Größen, die dann von einem Computer weiterverarbeitet werden können.

wenn größere, bzw. mehrfache Steuer- und Regelaufgaben mit dem Computer überwacht und durchgeführt werden sollen.

Die Aufspaltung der Ausgänge übernimmt eine Schaltung, welche aus vier 4-Bit-Registern besteht. Durch Programmierung ist es möglich, die Register abwechselnd anzusprechen und entsprechende Daten an die Register weiterzugeben.

Der TTL IC 7442 (BCD zu Dezimal-Dekoder) bietet eine einfachere Möglichkeit der Aufspaltung der vier Computer-Ausgänge. Je nach dem, welcher binäre Wert an den vier Eingängen des IC's vorhanden ist, wird einer der zehn Ausgänge aktiviert. Der IC 7442 hat allerdings den Nachteil, daß jeweils immer nur ein Ausgang gleichzeitig aktiviert werden kann.

Für die Aufspaltung der Eingänge bietet sich eine ähnliche Möglichkeit an. Der IC 74147 (Dezimal zu BCD Prioritäts-Encoder) hat vier Eingänge, welche einen entsprechenden binären 4-Bit-Ausgang ansprechen. Wird beispielsweise der sechste Eingang des IC's aktiviert, bildet sich an seinem Ausgang der duale Wert 0110. Werden mehrere Eingänge des IC's angesprochen, entspricht der Wert der Ausgänge dem Wert des höchstwertigen angesprochenen Einganges.

Ebenfalls eine gute Möglichkeit bietet der 16 zu 1 Datenselektor IC 74150. Er hat 16 Daten-Eingänge und einen 4-Bit-Adress-Eingang. Über den Adress-Eingang kann einer der 16 Daten-Eingänge angesprochen und das dort anstehende Signal zum Computer weitergegeben werden.

Im Versuchsstadium befindet sich u.a. noch ein Cassetten-Interface, um mit Hilfe eines handelsüblichen Cassetten-Recorders Programme auf Cassetten aus- und wieder einzuladen. Vortests führen auch zur Annahme, daß der Microtronic-Mikro-Computer durch ein entsprechendes Zusatzmodul zum sprechenden Computer erweitert werden kann.

## ERKLÄRUNG

Ich versichere, daß ich die Diplomarbeit selbständig verfaßt habe und keine anderen als die angegebenen Hilfsmittel benutzt wurden. Das Thema bzw. die Arbeit wurde in gleicher oder ähnlicher Form nicht bereits an anderer Stelle vorgelegt.

Saarbrücken, den \_ \_ \_ \_ \_

---

Unterschrift