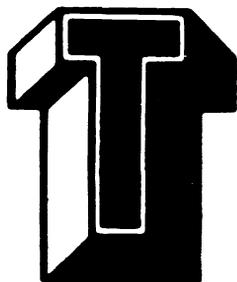


Kleinstrechner

**Tendenzen
und Theorien**



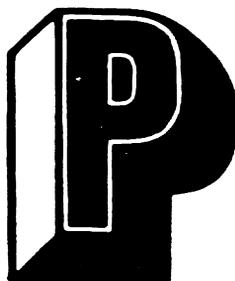
**Kleinstrechner
in der Schule**

**Informationen
und Ideen**



Heimcomputer

**Programme
und Projekte**



**Mikroprozessor
U 880**

**Spaß
und Spiel**



Kleinstrechner-TIPS

Heft 1

Mit 17 Bildern

Herausgegeben von

Prof. Dr.-Ing. Hans Kreul

Dr.-Ing. Wilhelm Leupold

Dr. sc. techn. Thomas Horn



VEB Fachbuchverlag Leipzig

Inhalt

Was will die Broschürenreihe Kleinstrechner-TIPS? 3

Leupold: Was brachte der Mikroprozessor U 880? 6

Gutzer: Kleinstrechentechnik in der Hand des Ingenieurs 17

Hamm: Erfahrungen beim Einsatz programmierbarer Kleinstrechner in der Schule 24

Schönfelder: HOMECOMPUTER – ein neues Gebiet für den Amateur (Teil 1) 33

Kreul: Quadratische Gleichungen – einmal anders gelöst 41

Gutzer; Pihale: Das MASTER-MIND-Spiel – der K 1003 als Herausforderer 44

Rechentechnische Begriffe für den Laien erklärt 40 und 58

Gutzer: Nimm-Spiel gegen den K 1003 (Teil 1) 59

Neues aus aller Welt 63

Was will die Broschürenreihe Kleinstrechner-TIPS?



Die moderne Rechentechnik hat in Form von Großcomputern, mittleren und kleinen Rechenanlagen, Prozeßrechnern u. a. m. in den letzten Jahren in alle Gebiete der Volkswirtschaft ihren Einzug gehalten. Es ist kaum noch ein Gebiet unseres Lebens denkbar, in das die elektronische Datenverarbeitung nicht in irgendeiner Weise eingegriffen hätte.

Mit der Entwicklung der hochleistungsfähigen Schaltkreise der Mikroelektronik sind nun in den letzten zehn Jahren kleine und kleinste Rechengерäte mit einem vor dieser Zeit unvorstellbaren Leistungsvermögen entstanden, die man vielfach ohne große Vorkenntnisse zur Lösung seiner beruflichen und seiner privaten Aufgaben nutzen kann. Es existiert heute bereits eine kaum noch übersehbare Vielfalt von elektronischen Taschenrechnern, programmierbar und nicht-programmierbar, von programmierbaren Tischrechnern, von Mikrorechnern und Mikrorechnersystemen, von Heimcomputern, Videocomputern usw., und es ist bei weitem noch kein Ende dieser stürmischen Entwicklung abzusehen. Die einzelnen Geräte werden immer mehr verkleinert, ihre Rechengeschwindigkeit wird vergrößert, ihr Energiebedarf wird immer mehr verringert, die Nutzerfreundlichkeit der Geräte wird ständig erhöht, indem man Kleinst-

rechner herstellt, die mit Hilfe von problemorientierten Programmiersprachen wie BASIC und PASCAL programmiert werden können.

Für den Fachmann auf dem Gebiet der elektronischen Datenverarbeitung stellt diese Entwicklung nichts Besonderes mehr dar. Er kennt die theoretischen Grundlagen, auf deren Basis die elektronischen Rechengерäte, seien sie sehr groß oder aber auch sehr klein, in der Lage sind zu arbeiten. Ihm steht ein großes Angebot an Literatur zur Verfügung, um sich über die Theorie oder über die Anwendungsmöglichkeiten der verschiedenartigen Rechenanlagentypen informieren zu können. Für ihn sind die elektronischen Rechner zum selbstverständlichen täglichen Arbeitsmittel oder Arbeitsgegenstand geworden.

Stellt man jedoch die »Profis« der elektronischen Rechentechnik den vielen, vielen Laien gegenüber, die einen Taschenrechner besitzen, die mit einem Tischrechner oder einem Bürocomputer arbeiten möchten, die sich einen Heimcomputer oder einen Videocomputer anschaffen wollen, so bilden diese Laien eine ungeheure Mehrheit gegenüber den erstgenannten Profis.

Wer hilft aber diesen vielen Laien, die die Rechentechnik nutzen wollen, bei der Bewältigung ihrer Probleme?

Durch geeignete Bedienungsanleitung-

gen und Anwenderhandbücher wird es dem Anfänger zwar relativ leicht gemacht, sich in die Arbeitsweise und die Anwendungsmöglichkeiten seines Kleinstrechners einzuarbeiten, aber schon nach kurzer Zeit wird er feststellen, daß er bei bestimmten Aufgabenstellungen nicht mehr weiterkommt, daß er Unterstützung oder wenigstens einen guten Rat brauchte, um mit der Lösung seines Problems voranzukommen. Die Spezialliteratur ist für ihn kaum verständlich, weil sie zu viel Grundwissen voraussetzt, das er nicht besitzt. Und Bücher oder Broschüren, in denen über die Lösung seiner Aufgaben berichtet wird, die gibt es kaum. So kommt es, daß sich sicher viele Besitzer von Klein- und Kleinstrechnern mit gleichartigen oder ähnlichen Problemen herumschlagen, viel Zeit für deren Lösung aufwenden, wobei sie manche Stunde des Nachdenkens einsparen könnten, wenn es ein Organ gäbe, das sich mit solchen »einfacheren« Problemen beschäftigt. Denn auch hier gilt die Devise:

Erfahrungsaustausch ist die billigste Investition.

Aus diesem Grunde wurde von den Mitarbeitern des VEB Fachbuchverlag vorgeschlagen, eine Broschürenreihe zu entwickeln, die sich mit eben jenen kleineren Problemen und Fragen beschäftigt, die dem Anfänger und dem Laien auf dem Gebiet der elektronischen Rechentechnik manches Kopfzerbrechen bereiten. So entstand unter Mitwirkung eines Herausgeberkollektivs, das aus den Herren Prof. Dr.-Ing. KREUL von der Ingenieurhochschule Zittau sowie Dr. sc. techn. HORN und Dr.-Ing. LEUPOLD von der Ingenieurhochschule Dresden besteht, die Reihe

Kleinstrechner - TIPS,

die in populärwissenschaftlicher Form die unterschiedlichsten Fragen aus dem

gesamten Gebiet der Anwendung von Klein- und Kleinstrechnern beleuchten soll.

Durch das Wort TIPS soll gleichzeitig das Spektrum angedeutet werden, mit dem sich diese Broschürenreihe beschäftigt:

- T** wie Tendenzen und Theorien,
- I** wie Informationen und Ideen,
- P** wie Programme und Projekte
sowie
- S** wie Spaß und Spiel.

Es sollen also die unterschiedlichsten Themenkomplexe behandelt werden, die für den interessierten Laien Bedeutung besitzen. Die Leser dieser Broschüre sind aufgefordert, an der weiteren Entwicklung dieser Reihe mitzuwirken. Sie können Vorschläge für zu behandelnde Themen sowie eigene kurze Beiträge, die die Länge von 10 Schreibmaschinenseiten, zweizeilig geschrieben, nicht überschreiten sollten, an eines der Mitglieder des Herausgeberkollektivs einreichen. Sie können aber auch Fragen an die Autoren der einzelnen Artikel richten und mit diesen in Erfahrungsaustausch treten. Die Autorenanschriften sind am Ende jedes einzelnen Artikels angefügt.

Im ersten Beitrag des vorliegenden Hefts Nr. 1 der Kleinstrechner-TIPS berichtet Dr.-Ing. W. LEUPOLD über den gegenwärtigen Stand und die Tendenzen der Produktion von Schaltkreisen für Mikrorechner und Mikroprozessoren in der DDR. Ein ähnlicher Übersichtsbeitrag über die Taschenrechnerproduktion unseres Landes ist für Heft 3 der Reihe vorgesehen.

Die beiden nachfolgenden Beiträge von Dr. H. GUTZER und von Dr. G. HAMM informieren über die Möglichkeiten der Anwendung von Taschenrechnern und programmierbaren Kleinstrechnern in technischen Berufen sowie in der all-

gemeinbildenden Schule. Insbesondere der Artikel von Dr. HAMM soll Anregungen dafür geben, wie man bereits unsere Schüler der polytechnischen Oberschulen für die rationelle und effektive Nutzung der zur Verfügung stehenden Kleinstrechner interessieren kann.

Der folgende Artikel von Dr. SCHÖNFELDER soll vor allem die technisch interessierten Leser des Heftes ansprechen, die sich selbst einen Heimcomputer bauen wollen, der an ein normales Fernsehgerät angeschlossen werden kann. Dieser Artikel wird in den folgenden beiden Heften der Reihe fortgesetzt werden, so daß dann eine vollständige Bauanleitung für ein derartiges Eigenbau-Rechengerät zur Verfügung stehen wird.

Danach wird von Dr. GUTZER und von W. PIHAULE ein Programm vorgestellt, mit dessen Hilfe der elektronische Tischrechner K 1003 als Herausforderer gegen einen Mitspieler für das Master-Mind-Spiel fungieren kann, das in letzter Zeit eine weite Verbreitung gefunden hat, weil es hohe Anforderungen an die logischen und kombinatorischen Fähigkeiten der Spieler stellt.

Schließlich wird auch gezeigt, wie man durch die geeignete Wahl der entsprechenden Rechenverfahren Probleme recht effektiv mit seinem einfachen Taschenrechner lösen kann:

Die Lösungen quadratischer Gleichun-

gen werden auf eine unkonventionelle Weise ermittelt, wobei der erforderliche Rechenaufwand wesentlich geringer wird als bei der normalerweise üblichen Lösung mit Hilfe der Lösungsformeln für quadratische Gleichungen. Die Rechenabläufe für die herkömmliche Vorgehensweise und für den neu vorgeschlagenen Lösungsweg werden einander gegenübergestellt, so daß der Aufwand für die beiden unterschiedlichen Lösungsverfahren unmittelbar verglichen werden kann.

GUTZER stellt abschließend ein *Programm für das sogenannte NIMM-Spiel* gegen einen elektronischen Tischrechner vor. Auch dieser Artikel ist so geschrieben, daß sichtbar wird, welche Gedankengänge und Vorarbeiten erforderlich sind, um einen Tischrechner zu befähigen, als Partner für einen Menschen bei einem einfachen Spiel zu fungieren. Das vorgestellte Programm ist bei geeigneten Modifikationen auch für andere Tisch- und Taschenrechner als den K 1003 verwendbar.

Herausgeber und Verlag hoffen auf eine gute Resonanz dieser neuen Broschürenreihe. Möge sie dazu beitragen, daß die Leser Anregungen für ihre eigene Beschäftigung mit den Geräten der neuen Rechentechnik finden, so daß sie ihre Probleme und Aufgaben besser, rationeller, schneller und damit effektiver als bisher lösen können.

Herausgeber und Verlag

Was brachte der Mikroprozessor U 880?



Der Mikroprozessor U 880 – Bestandteil eines hochintegrierten Schaltkreissystems

Als 1972 die ersten Mikroprozessoren als hochintegrierte Schaltkreise mit zunächst über 3000 Transistorfunktionen auf einem einzigen Halbleiterplättchen (Chip) bereitgestellt wurden, sprach man – nach Transistor und integrierter Schaltung – von einer *dritten Revolution der Elektronik* und vom Beginn einer *neuen Phase der technischen Revolution*. In der DDR wird nach dem Mikroprozessor U 808 seit fünf Jahren den Anwendern mit dem Mikroprozessor CPU U 880 (Central Processing Unit, Zentrale Verarbeitungseinheit) ein in seinem Leistungsvermögen wesentlich verbesserter Schaltkreis (SK) für den Einsatz in Geräten zur Informationsverarbeitung und für Steuerungsaufgaben zur Verfügung gestellt.

Dieser Mikroprozessor, der 8 Bit parallel verarbeitet (intern bei speziellen Befehlen 16 Bit), zeichnet sich gegenüber seinem Vorgänger nicht nur durch einen höheren Integrationsgrad (8200 integrierte Transistoren) und damit verbunden eine größere Zahl von Verarbeitungsfunktionen, die sich auch in der größeren Zahl und höheren Komplexität der knapp 700 ausführbaren Befehle niederschlug, sondern auch durch eine höhere Verarbeitungsge-

schwindigkeit [Taktfrequenz 2,5 MHz, typ. Befehlsarbeitungszeit (bei einem Register-Register-Transfer) 1,6 μ s] aus. Auch waren die Schnittstellen zur weiteren anzuschließenden Elektronik günstiger gewählt, und Teile davon wurden zusammen mit der CPU als Schaltkreissystem konzipiert, zu dem gehören:

- Zentrale Verarbeitungseinheit
CPU U 880 D
- SK für parallele Ein- und Ausgabe
PIO U 880 D/855 D
- SK für serielle Ein- und Ausgabe
SIO U 856 D
- SK für Zähler- und Zeitgeberfunktion
CTC U 857 D
- SK für direkten Speicherzugriff
DMA U 858 D

- Speicher-SK
statisch RAM
1 K \times 1 Bit U 202 D
dynamisch RAM
16 K \times 1 Bit U 256 D
ROM
1 K \times 8 Bit U 505 D
EPROM
1 K \times 8 Bit U 555 C

Zur Verbindung der CPU mit den Ein-/Ausgabe-SK, den Sp-SK und weiterer Zusatzelektronik dienen folgende Signalgruppen und Signale (Bild 1):

- 6 System-Steuersignale
 - MI** Maschinenzklus Eins
 - MREQ** Speicheranforderung
 - IORQ** Ein-/Ausgabeanforderung
 - RD** Lesen
 - WR** Schreiben
 - RFSH** Speicherauffrischen
 - 5 CPU-Steuersignale
 - HALT** Halt-Zustand
 - WAIT** Warten
 - RESET** Rücksetzen
 - INT** Interrupt-Anforderung
 - NMI** Anforderung des nicht maskierbaren Interrupt
 - 2 CPU-Bus-Steuersignale
 - BUSRQ** Bus-Anforderung
 - BUSAK** Bus-Anforderungsbestätigung
- 64K Speicherplätze (mit Worten zu je 8 Bit) und je 256 Ein- und Ausgabekanäle können direkt adressiert werden.

Einsatz der U 88-Schaltkreisfamilie in allen Bereichen der Volkswirtschaft

Wie kaum ein anderes Bauelement, Maschinenteil oder eine Funktionsgruppe der Vergangenheit haben die SK des U 88-Schaltkreissystems und die daraus entwickelten Baugruppen in großem Umfang in allen Bereichen von Industrie, Verwaltung, Nahrungsgüterwirtschaft, Medizin, im Umweltschutz und nunmehr beginnend auch im privaten Haushalt Anwendung gefunden. Die Zahl der Einsatzfälle ist so groß geworden, daß sie nicht mehr mit aktuellem Stand angegeben werden kann, weil diese aufgrund ihrer Vielfältigkeit kaum noch vollzählig erfaßt werden können und jeden Tag neue Anwendungen hinzukommen. Wenn im folgenden auf einige Einsatzlinien eingegangen wird, kann diese

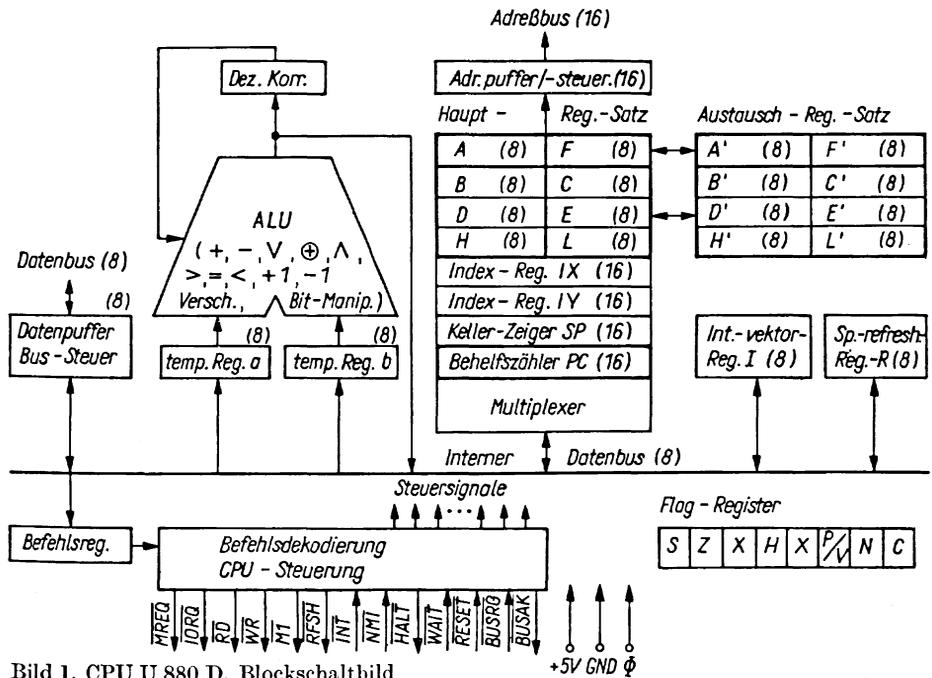


Bild 1. CPU U 880 D, Blockschaltbild

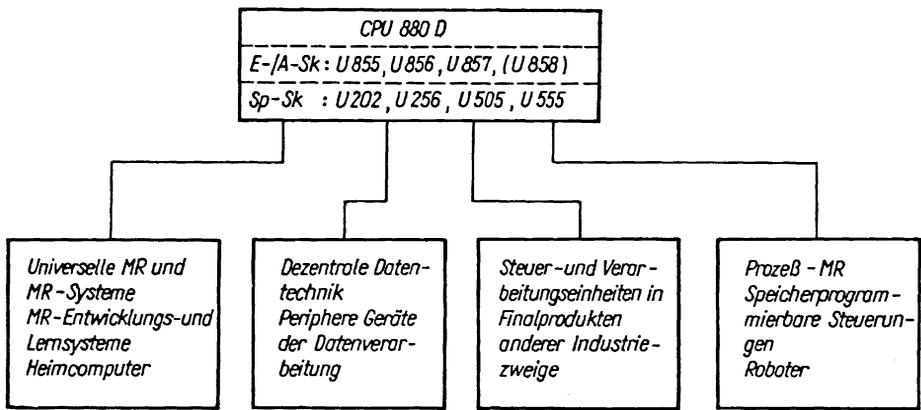


Bild 2. Einsatzbereiche der CPU U 880 D

Darstellung bei weitem keinen Anspruch auf Vollständigkeit erheben. Es wurde auch i. allg. auf die Angabe der speziellen Bezugsmöglichkeiten und der Quellen verzichtet, da dies ein sehr umfangreiches Literaturverzeichnis erfordern würde. Von einzelnen Betrieben beziehbare Baugruppen oder nachnutzbare Entwicklungen können aus dem »Zentralen Nachweisspeicher Angewandte Mikroelektronik« (ZNAM) des VEB Applikationszentrum Elektronik Berlin angefordert bzw. den Broschüren »Nachnutzbare Anwendungen Mikroelektronik« dieses Betriebes entnommen werden. Die Gruppierungen in Bild 2 stellen den Versuch einer Ordnung im Zusammenhang mit den folgenden Ausführungen dar, sind letztlich aber doch willkürlich vorgenommen worden.

Universelle Mikrorechner (MR), MR-Entwicklungs- und Lernsysteme

Vom VEB Robotron-Elektronik Zella-Mehlis wird auf der Grundlage der oben genannten hochintegrierten SK das *Mikrorechnersystem robotron K 1520* hergestellt. Dieses besteht aus OEM-Baugruppen (Original Equipment Manu-

facture, in Systemen auch anderer Hersteller einsetzbare Baugruppen) in Gestalt von Steckeinheitenmoduln (215 mm × 170 mm) mit busseitig je 2 zweireihigen 58poligen Steckverbindern. Diese und Stromversorgungsmoduln verschiedener Leistungsklassen werden in Steckeinheitseneinschüben untergebracht.

Aus dem Spektrum von Steckeinheitenmoduln können Anwenderkonfigurationen flexibel ausgewählt und zusammengestellt werden. Es umfaßt folgende typische Baugruppen (Bild 3):

- Zentrale Recheneinheiten (ZRE),
- Speichermodule: Operativspeicher (OPS), Operativ-/Festwertspeicher (OFS), programmierbare Festwertspeicher (PFS),
- Anschlußsteuereinheiten (AS₁) für den Anschluß von peripheren Geräten,
- Module zur Busverlängerung (BVE) bei Überschreitung der zulässigen Entfernungen bzw. Busbelastungen.

Die ZRE stellt die zentrale Baugruppe des MR-Systems dar, die über den Systembus (8 bit Datenbus, 16 bit Adreßbus, 20 bit Steuer- und Kontrollsignale, 14 Stromversorgungssignale)

mit den Steckeinheiten (StE) für Speicher bzw. Anschlußsteuerungen oder über den Koppelbus mit weiteren ZRE-StE Daten-, Adreß- und Steuersignale austauschen kann. Vier Varianten der ZRE können als selbständige Mikrorechner (1-Platinen-Rechner mit 1 KByte stat. RAM und 3 KByte EPROM) oder als Slave-Rechner in Mehrrechnersystemen eingesetzt werden. Die Kopplung erfolgt über den Koppelbus. Eine ZRE mit 8 KByte EPROM ist für den Anschluß einer alphanumerischen Kleinanzeige vorbereitet, eine andere mit zwei CPU U 880 und 1 KByte RAM-Speicher als 2-Prozessor-System in Geräten der dezentralen Datentechnik. Der Anschluß von Geräten der MR-Peripherie ist über je zwei 39polige Steckverbinder an der Griffseite der Anschlußsteuermodule möglich.

Vom Hersteller des MRS K 1520 werden die Anschlußsteuerungen für folgende Geräte sowie diese selbst angeboten: Bedieneinheit; Lochbandeinheit, Kassettenmagnetbandgerät,

Seriendrukker; Bediendrukker; Tastaturen; Folienspeichereinheit; Mini-folienspeichereinheit; Monitore.

Dieses Baugruppensortiment wurde in den letzten Jahren von Anwendernbetrieben dem jeweiligen Einsatzfall entsprechend erweitert. Darüber hinaus wurden von einzelnen Betrieben weitere universell einsetzbare, buskompatible Module entwickelt, die von Interessenten bezogen bzw. nachgenutzt werden können. Einige Beispiele, die von allgemeinem Interesse sein dürften, seien hier genannt. Vom VEB Robotron Büromaschinenwerk Sömmerda werden sowohl ein Thermostreifendrukker TSD 16 (als Einbaubaugruppe ohne Gehäuse) mit 16 Druckstellen je 7×5 Punkte (Zeichen 2,8 mm \times 1,8 mm) als auch eine an den K 1520-Bus anschließbare Druckerlektronik (StE 215 mm \times 170 mm) angeboten, die sich insbesondere für kleine, mobile MR-Systeme gut eignet.

Dem Bedürfnis für die Bereitstellung von E-/A-Kanälen entsprechen die im ZFT Mikroelektronik Dresden bus-

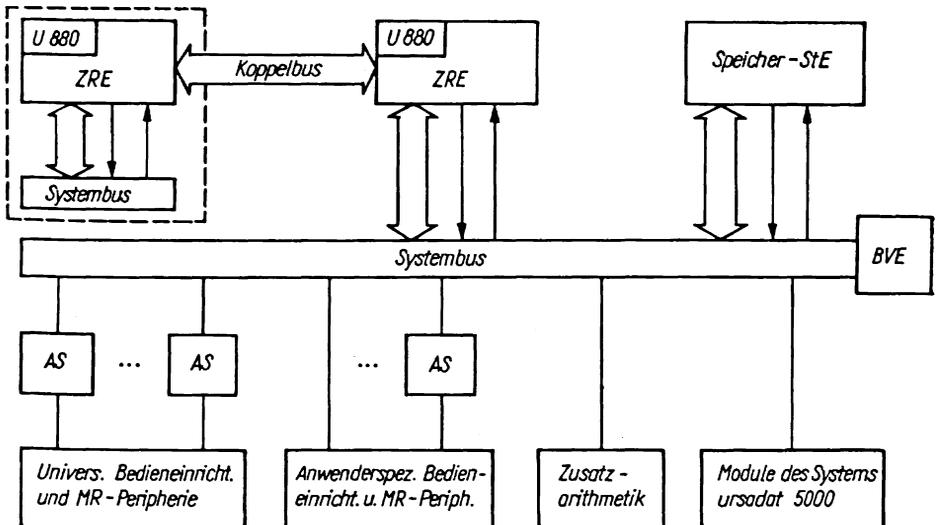


Bild 3. MR K 1520-Gerätekonfiguration

kompatibel entwickelten, nachnutzbaren, mit PIO- und CTC-SK bestückten StE mit den Bezeichnungen EAS 1 bis 3.

Ähnliche Baugruppen enthält das MR-Baukastensystem MIRAS, das von der Ingenieurhochschule Wismar (unter Einbeziehung von Baugruppen, die im VEB Numerik Karl-Marx-Stadt, dem Institut für Nachrichtentechnik Berlin und der Wilhelm-Pieck-Universität Rostock) entwickelt wurde. Es umfaßt neben Baugruppen der ersten und zweiten Peripherie Spezialbaugruppen für Tastaturen, Bildschirmsteuerung, ADU's, DAU's Anzeigebaugruppen und Baugruppen für die Prozeßanpassung, die insbesondere für den ökonomisch günstigen Aufbau kleinerer Systeme gedacht sind und zur Nachnutzung angeboten werden. Vom Zentrum für wissenschaftlichen Gerätebau Berlin der AdW der DDR wurde eine K 1520-StE zur Ausgabe grafischer Informationen mit 256×256 Bildpunkten auf ein Grafikdisplay entwickelt.

StE mit erhöhten Speicherkapazitäten brachten Entwicklungen der Sektion Informationstechnik der Technischen Hochschule Karl-Marx-Stadt: K 1520-StE für 8 KByte stat. RAM und für 64 KByte dyn. RAM.

Eine wesentliche Erhöhung der Rechengeschwindigkeiten sowohl bei Fest- und Gleitkommaadditionen und -subtraktionen sowie vor allem auch bei Multiplikationen und Divisionen (16-bit-FK-Multiplikation etwa $5 \mu\text{s}$, -Division $15 \dots 20 \mu\text{s}$) bringt der Einsatz des Arithmetik-SK U 832 auf einer vom ZFT Mikroelektronik Dresden entwickelten K 1520-StE. Mit dieser wird z.B. die Berechnung von Funktionswerten \sin und \cos in etwa $150 \mu\text{s}$ möglich.

Für den Anschluß von diskreten und analogen Prozessen bzw. Prozeßinfor-

mationen an das MRS K 1520 wurde das *System ursadat 5000* entwickelt (weitere Ausführungen dazu weiter unten).

Zu den Systemunterlagen des K 1520, die vom Hersteller angeboten werden, gehören

– Basis-MOS mit Standardmoduln (FK-, GK-, BCD-Arithmetik, Konvertierungsprogramme) und Geräuteroutinen (für die oben genannten anschließbaren peripheren Geräte und Interfaces),

– Wirtsrechner-Systemunterlagen für PRS 4000 und KRS 4200 in der Assemblersprache MAPS K 1520: Cross-Assembler, Cross-Simulations- und Testsystem, Cross-Binder und Aufbereitungssystem.

Als maschinenorientierte Programmiersprache steht SYPS K 1520 zur Verfügung.

Von vielen Anwendern wurden noch relevante Ergänzungen erarbeitet, so z. B. von der Ingenieurhochschule Dresden

– ein modulares Betriebssystem MOBS 1520 in 2 Versionen für Ein- und Zweiprozessorsysteme,
– eine Makroassemblersprache MACRO 1520.

Ein weiteres, stark ausgebautes modulares Mikrorechnersystem stellt das MPS 4944 des Zentralinstituts für Kernforschung Rossendorf der AdW der DDR dar. Neben Prozessorkarten mit dem U 808 und dem Intel 8080 enthält es solche mit dem U 880. Die Module haben die Nenngrößen $240 \text{ mm} \times 20 \text{ mm} \times 300 \text{ mm}$ (die darin verwendeten Leiterplatten $215 \text{ mm} \times 170 \text{ mm}$ mit zwei 58poligen Steckverbindern). Auf der Frontplatte sind Bedien- und Anzeigeelemente sowie Steckverbinder für Ein- und Ausgänge angebracht. Zum System gehören mehrere Speicherkar-

ten, Anschlußsteuerungen für Standardinterfaces (Sif 1000, SI 1.2, SI 2.2, FS, ESER), Bedieneinheiten, Prozeßkoppeleinheiten (mehrere ADC und DAC, Digital-Ein- und -Ausgabe), ein Hardware-Multiplizierer, DMA-Einheit Multiplexer (Meßstellenumschalter), Timer, Busverlängerungseinheiten und ein spezieller CAMAC-Anschluß.

Das System enthält softwaremäßig Basis-Assembler, Cross-Assembler für KRS 4200, Cross-Makroassembler und PL/M-Compiler für ES 1022/1040, Simulations- und Testsystem, Standardbetriebsprogramme (Lader, Standard-, CAMAC- und Testmonitor), Arithmetikprogramme für FK und GK sowie Ein-/Ausgaberroutinen.

Unter Verwendung des MPS 4944 wurde z. B. an der Ingenieurhochschule Dresden unter Beibehaltung der modularen Bauweise das Parallelprozessorversuchssystem PVS 82 mit maximal 52 Funktionsmoduln (zugeordnet vier Parallelprozessoren und einem Regieprozessor) entwickelt.

Auch vom Kombinat Robotron wird ein Multi-Mikrocomputer-System Robotron 238 in Gestalt eines Sortiments von Funktionsmoduln für bis zu 16 Prozessoren angeboten, die durch einen Universalbus auf einen Adreßraum von 1 MByte zugreifen können. Auch hier stellt der U 880 wieder das Kernstück der Module dar.

Schon sehr frühzeitig wurde es international im Zusammenhang mit dem Einsatz von Mikrorechnern für notwendig erachtet, sowohl Entwicklungssysteme für eine effektive Einsatzvorbereitung als auch Lernsysteme für die Qualifizierung zur Verfügung zu haben. Auch auf diesem Gebiet ist ein entsprechendes Angebot zu verzeichnen. Vom VEB Kombinat Robotron wurde das Mikrorechnerentwicklungssystem (MRES) A 5601 mit seinem Betriebssystem MEOS 1521 als Programmier-

arbeitsplatz für die Programmentwicklung auf der Basis der Assemblersprache SYPS K 1520 und als Prüfrechner für die Prüfung und Testung eines K 1520-Anwendersystems herausgebracht. Es umfaßt eine K 1520-Konfiguration mit ZRE, bis zu 7 Speicher-StE und 2 bis 4 Folienspeichern, peripheren Geräten (u. a. Bildschirm), EPROM-Programmierereinrichtung und Busverstärker. Für den Anschluß eines Anwendersystems und eine effektive Programmtestung besitzt es weiterhin die Baugruppen Seiten-, Funktions- und Haltepunktsteuerung sowie einen Echtzeitspeicher zur Aufzeichnung, Identifizierung und Rückverfolgung bis zu 512 CPU-Zyklen. Das MEOS 1521 enthält auf Festwertspeichern residente Betriebssystemfunktionen (u. a. für die genannten Baugruppen) und auf Folienspeichern oder Kassettenmagnetband abgelegte transiente Betriebssystemfunktionen, wie z. B. Editor, Assembler, Binder.

Das von der Technischen Hochschule Ilmenau entwickelte und im VEB Elektronik Gera gefertigte Gerätesystem *Mikrocombi* kann

- als Entwicklungssystem mit dialogorientierter Programmentwicklung auf Assembler- und Makroassemblerniveau,
- als Mikroprozessortest- und Prüfgerät,
- als Mikroprozessorablehrautomat zur rechnergestützten Ausbildung

eingesetzt werden. Hervorzuheben sind die Möglichkeiten der Analogwert-Ein- und -Ausgabe. Vom Hersteller wird zusätzlich eine umfangreiche Firmware angeboten. Im Zusammenhang mit dem Mikrocombi wurde an der TH Ilmenau eine Makroassemblersprache EPAS 80 (Elementarprozeß-Assemblersprache) für ein effektiveres Programmieren entwickelt.

Der Typ eines Lernsystems ist besonders ausgeprägt festzustellen beim *Poly-Computer 880*, der an der Sektion Informationstechnik der TH Karl-Marx-Stadt entwickelt wurde und im VEB Kombinat Polytechnik und Präzisionsgeräte Karl-Marx-Stadt hergestellt wird. Dieser dient vorzugsweise dem Kennenlernen der geräte- und programmtechnischen Funktionsweise von Mikrorechnern. Dieses Anliegen wird durch die umfangreichen Unterlagen in Gestalt eines Bedien-, eines System- und eines zweiteiligen Arbeitsbuches wirkungsvoll unterstützt.

Die – im Vergleich zu ihrem Leistungsvermögen – niedrigen Preise der hochintegrierten Schaltkreise gaben international ab Mitte der 70er Jahre dem Hobby-Markt starke Impulse. Es wurden Bausätze angeboten, aus denen sowohl kleine Lernsysteme als auch Mikrocomputer für persönliche Zwecke in Haushalt und Familie (Übung von Fremdsprachen, Mathematik und anderen Disziplinen, Haushalts-Buchführung und -Statistik, Erstellen und Verwalten von Karteien, Computerspiele, Gymnastikübungen, Steuerung von Haushaltgeräten, Überwachung von Sicherheitseinrichtungen usw.) aufgebaut werden konnten. Da die Preise für die Grundkonfiguration die »Geschenkschwelle« unterschritten, fanden sie schnell als sog. Personal-Computer, Home- (bzw. Heim-) Computer größere Verbreitung. Das Grundgerät umfaßt einen Mikrorechner, zusammen mit der Ansteuerung für ein handelsübliches Fernsehgerät und Programm- und Datenspeicher in das Gehäuse einer Bedientastatur mit einigen Signallämpchen eingebaut. An Software gehören dazu ein kleines Betriebssystem (Monitor) und Interpreter für dialoggünstige Programmiersprachen, wie z. B. BASIC oder PASCAL.

Diese Anwendung der Mikroprozessor-

technik und insbesondere des U 880 dürfte mit dem zu erwartenden Angebot von Heimcomputern aus DDR-Produktion für weite Kreise unserer Bevölkerung von Interesse sein. Da neben der industriellen Fertigung von Heimcomputer-Bausätzen auch der individuelle Aufbau solcher Mikrorechner weiterhin erfolgen wird, gibt G. SCHÖNFELDER – beginnend in diesem Heft – dafür eine Anleitung.

Eine neue Generation von Geräten der dezentralen Datentechnik mit dem U 880

In den sechziger und siebziger Jahren wurden weltweit und besonders auch von Betrieben der DDR vielfältige Formen von Geräten der »Klein-Datenverarbeitung«, wie Buchungs-, Fakturier-, Schreib- und Organisationsautomaten, Datenerfassungs- und Dateneingabegeräte entwickelt und in hohen Stückzahlen gefertigt. Alle diese Geräte besaßen Steuer-, Rechen- und Speicherwerke, die sich von Gerätetyp zu Gerätetyp z.T. sehr stark voneinander unterschieden. Die Verfügbarkeit von Mikroprozessoren, und zwar speziell der Schaltkreise des U 88-Systems, ermöglichte prinzipiell neue Lösungen, die in den letzten Jahren vom Kombinat Robotron neu entwickelten und gefertigten Einrichtungen der dezentralen Datentechnik und in peripheren Geräten der ESER-Rechner ihren Niederschlag fanden. Zu den mit dem U 880 bzw. mit ZRE-Moduln des MRS K 1520 als Kernstück hergestellten Geräten und Einrichtungen gehören:

- die Bürocomputer A 5110/20/30 mit einem Compiler für PASCAL-1520,
- die Buchungs-, Fakturier- und Abrechnungsautomaten A 5101 und 5103,
- für die Datenerfassung, -aufbereitung und -konvertierung die Datenerfas-

sungsgeräte A 5201 und 5203, der Konverter A 5202 und der Belegleser K 6703,

- das Datensammelsystem A 520 mit max. 8 Erfassungsplätzen,
- das elektrische Schreibsystem A 5301,
- für die Datenfernverarbeitung der Leitungsmultiplexor K 8522 und der Leitungskonzentratoren K 8521,
- für die Kommunikation das Bildschirm-Ein-Ausgabegerät K 8911, die Datenstation K 8913, das universelle Bildschirmterminal K 8931, das Druckerterminal K 8951 und das Platzreservierungsterminal K 8927 (als Modifikation des A 5203).

Als Software für diese Geräte existieren Betriebssysteme, wie z. B.

- SIOS 1526 als modulares Simultan-Betriebssystem für Einrichtungen zur Datenerfassung, Buchung, Fakturierung und Rechnerkommunikation,
- BROS 1525 als Betriebssystem für druckerorientierte Geräte für Buchung, Abrechnung und Fakturierung.

Der Einsatz ist prinzipiell überall möglich. So können z. B. mit dem Platzreservierungsterminal Aufgaben im Transportwesen, in der Gastronomie u. a. wesentlich effektiver und rationeller erledigt werden als bisher.

Auch periphere Geräte von Rechnern der höheren Leistungsklassen werden zunehmend unter Verwendung von Mikroprozessoren aufgebaut. Als Beispiel sei nur der Bedien- und Serviceprozessor EC 7069 (für die ESER-Zentraleinheit 2655) genannt, der Teile aus dem Programm »Dezentrale Datentechnik« wiederverwendet.

Der Mikroprozessor U 880 — eine universelle Steuer- und Verarbeitungseinheit

Die Zahl der Einsatzfälle der CPU U 880 als Steuer- und Verarbeitung-

seinheit in Finalprodukten von Industriezweigen außerhalb der Datenverarbeitungs-, Rechen- und Automatisierungstechnik überwiegt gegenüber den anderen in Bild 2 gezeigten Einsatzlinien. Dabei ist der Bogen der Anwendungsfälle sehr weit gespannt. Er reicht von der Verwendung als Kernstück von Maschinen- und Fertigungssteuerungen (DNC-Maschinen, Druckmaschinen, automatische Leiterplattenbestückung usw.) über den Einsatz für Meßwertaufnahme und -verarbeitung in der Meß- und Wägetechnik (Spektrometer, Dosier- und Abfülleinrichtungen, Flächenmassenmessungen, Meßsteuerungen für Beschichtungen und Schweißvorgänge usw.), in Steuerrechnern z. B. für die Herstellung von elektronischen Hybridschaltungen . . . bis zur modernen Biomeß- und Medizintechnik. Einige Beispiele aus dem zuletzt genannten Einsatzbereich sollen die Breite der Anwendung veranschaulichen. Es liegen Einsatzfälle vor bzw. sind durch entsprechende Entwicklungen vorbereitet:

- zur Dialogverarbeitung von Patientendaten, zur Krankenhausstatistik,
- zur Patientenüberwachung, zur Geburtsüberwachung,
- zur Biosignalanalyse und -verarbeitung, zur Funktionsdiagnostik (kardiologische, respiratorische, neurologische),
- für die klinische Radiologie, Röntgendiagnostik, Nuklearmedizin,
- für die Laborautomatisierung,
- für on-line-Auswertungen von dynamischen Belastungsuntersuchungen,
- zur automatischen Bildverarbeitung.

Dabei werden in den Geräten sowohl K 1520- und MPS 4944-Module, insbesondere die ZRE-Platinen (z. B. in den Neuentwicklungen des VEB Meßgerätewerk Zwönitz) als auch U 880-Minimalsysteme verwendet. Softwaremäßig kommen das Echtzeitprogramm-

system EIEX 1521, andere Echtzeitbetriebssysteme wie das SIOS 1520 und RTX 1520, die Programmierung in BASIC als auch (insbesondere bei Minimallösungen) eine problemangepaßte Programmierung in Assembler-sprache zur Anwendung.

Universelle Automatisierungssysteme – Prozeß-MR – Roboter

Neben der Datenverarbeitungs- und Rechentechnik stellt die Automatisierungstechnik ein weiteres Anwendungsfeld der MR-Technik dar, das dem Anwender Lösungen in Gestalt von Systemen anbietet, aus denen heraus dem jeweiligen Einsatzfall entsprechende Konfigurationen zusammengestellt werden können.

Ein solches, speziell dem MRS K 1520 angepaßtes und es für den Einsatz als Prozeßrechner erweiterndes System ist das *ursadat 5000* (innerhalb des Systems ursatron 5000) des VEB Elektro-Apparate-Werke Berlin-Treptow. Wie das MRS K 1520 selbst ist es modular aufgebaut und umfaßt außer den bereits beim K 1520 genannten Moduln

- Prozeß-Ein-/Ausgabe-Module (analog-Ein- und -Ausgabe, Digital-Eingabe statisch, dynamisch und multiplex, Universal-Impulszähler, Digitalausgabe mit Transistor, Relais, Optokoppler, mit Handshake-Signalen und impulsförmig),
- Stromversorgungsüberwachung und Netzausfallanalysator,
- schnelle serielle Interfacemodule,
- Bedien- und Anzeigemodule, Zentraler Überwachungsbaustein.

Die Grundeinheiten bestehen aus EGS-Bausteinträgern mit 23 StE-Plätzen und eigener separater Stromversorgung.

Zum System gehören weiter das Be-

diengerät ursatron 5000 zur prozeßorientierten Bedienung und die Serviceeinheit ursatron 5000 für die Prüfung, Inbetriebnahme und Wartung von Anwendersystemen.

Die Systemsoftware besteht aus den Komponenten

- Echtzeitsteuerprogrammssystem EIEX 1521 (vom VEB Kombinat Robotron),
- Ein-/Ausgabemodule für die Prozeß-StE des ursadat 5000,
- Bedien- und Kommunikationsmoduln für das Bediengerät ursatron 5000,
- Standard-Unterprogramme (Arithmetik, Zeichenkettenverarbeitung, Konvertierung).

Vom VEB Geräte- und Regler-Werk Teltow (als Generalauftragnehmer kompletter Automatisierungsanlagen) wurde unter Verwendung des Systems ursadat 5000 und Ergänzung durch Eigenentwicklungen das universelle, mikrorechnergeführte *Automatisierungssystem audatec* entwickelt. Es stellt ein funktionell und räumlich dezentrales Automatisierungssystem in zwei Hierarchie-Ebenen – dem prozeßnahen Bereich und der Leitebene – dar. Die hauptsächlichen Systembestandteile sind:

- der Rechnerkern (mit der ZRE K 2521),
- das Prozeßinterface (als Schnittstelle zwischen Meß- und Stelltechnik und dem Rechnerkern),
- das serielle Zwischenblockinterface (Übertragungsgeschwindigkeit 500 kBaud bei Entfernungen bis zu 3 km und bis zu 254 anschließbare Stationen),
- das Mensch-Anlage-Interface (mit Farbdisplay und Funktionstastatur). audatec ermöglicht den Aufbau folgender Anlagenvarianten:
- Autonome Automatisierungseinrichtung (AAE) für technologische Pro-

zesse kleinen und mittleren Umfangs bis maximal 150 analoge Prozeßsignale,
– Kleinverbundanlage (KVA) für technologische Prozesse mittleren Umfangs bis etwa 500 Prozeßsignale,
– Großverbundanlage (GVA) für technologische Prozesse mittleren und großen Umfangs bis etwa 2000 Kommunikationsstellen je Bussystem/Fahrstand.

Das Basismodulpaket ist in EPROM's gespeichert und enthält an Globalfunktionen:

- Meßwertverarbeitung analoger Größen,
- Meßwertverarbeitung binärer Größen,
- Meßwertverarbeitung für Impuls-/Zählgrößen,
- Regelung.

Die Basismodule sind über meßstellenweise aufgebaute Verknüpfungslisten, die auf RAM-Speichern abgelegt werden, miteinander verbunden.

Von den weiteren Systemen zur Automatisierung, die mit dem U 880 arbeiten, modular aufgebaut und universell einsetzbar sind, sei nur noch auf die Fertigungssteuerung CNC 600 vom VEB Numerik Karl-Marx-Stadt hingewiesen.

Für Aufgaben der Prozeßmeßwert-erfassung und -verarbeitung wurde (mit dem U 880) von der TH Ilmenau in Kooperation mit dem VEB Wägetechnik Rapido Radebeul das *Mikrodat 80* entwickelt, das u. a. ein Systeminterface mit SI 1.2, Inkrementaleingang und AD-Umsetzer besitzt.

Die Mikroprozessortechnik ermöglichte, bisher in verdrahtungsprogrammierter Technik realisierte Steuerungen von Be- und Verarbeitungsmaschinen als speicherprogrammierte Steuerungen zu gestalten. Unter Verwendung des U 880 als Steuer- und Verarbeitungseinheit wurden vom VEB Elektro-

Apparate-Werke Berlin-Treptow die speicherprogrammierbaren Steuerungen *ursalog 5010 und 5020* entwickelt. Die zur Programmierung verwendete Fachsprache PROLOG 1 bzw. 2 gestattet dem Anwender – vom Ablaufgraphen ausgehend – eine einfache Notation in Form von Makros bezüglich der Assemblersprache.

Der wohl augenfälligste Fortschritt in vielen Bereichen der materiellen Produktion war in den letzten Jahren durch den Einsatz von Robotern zu verzeichnen. Die Übernahme anspruchsvoller Aufgaben durch Industrieroboter erfordert nicht nur Steuer-, Speicher- und Verarbeitungsfunktionen, sondern auch die Fähigkeit, angelernt werden zu können (teach-in). Hierbei müssen z. T. recht komplizierte Algorithmen schnell und mit hohen Ansprüchen an die Genauigkeit abgearbeitet werden. Digitale, mikroelektronisch realisierte Steuerungen – meist mit einem oder mehreren Mikroprozessoren als Kernstück – erfüllen diese Forderungen. In der DDR wurden in wenigen Jahren Tausende von Industrierobotern entwickelt und zum Einsatz gebracht. Nur zwei von ihnen, die mit dem U 880 arbeiten und universell einsetzbar sind, sollen als Beispiel genannt werden.

Der *Industrieroboter PHM 4* bzw. *PHM 40* (als Weiterentwicklung) des VEB Kombinat Robotron stellt ein programmierbares Handhabungsmittel als Gelenkroboter mit 5 Freiheitsgraden (und Greiffunktion) dar. Die Steuerung SIR 40 kann in zwei Betriebszuständen arbeiten: Teach-in- und Automatik-Betrieb. Der Roboter besitzt eine Wiederholgenauigkeit von 0,3 mm, seine maximalen Verfahrgeschwindigkeiten für die 3 Achsen (bei 500 mm Radius) liegen zwischen 0,2 und 0,5 m/s, die Handhabemasse beträgt max. 250 g. Die zugehörige Software

besteht aus einer modular aufgebauten zentralen Ablaufsteuerung und einer Bahnsteuerung, das Anwenderprogramm aus einer definierten Anzahl anzufahrender Raumpunkte und Befehle, mit denen der Roboter und die technologische Peripherie gesteuert werden.

Die vom Forschungszentrum des Werkzeugmaschinenbaus und dem VEB Numerik Karl-Marx-Stadt gemeinsam entwickelte *Robotersteuerung IRS 600* hat den K 1520 als Rechnerkern. Sie ist für Beschickungsroboter in Zylinderkoordinatenbauweise mit bis zu 3 lagegeregelten Achsen vorgesehen, so z.B. für vier Varianten des Industrieroboters IR 2. Die Hardware umfaßt außer K 1520-StE eine Serviceeinheit, Busergänzungs- und Überwachungs-StE, Lageregelungs-StE und Anschlußsteuerungen. Die Programmierung kann durch Handeingabe, indirektes Teach-in oder über Lochstreifen erfolgen. Bis zu 1000 Raumpunkte können gespeichert werden. Zum modular aufgebauten Programmsystem gehören ein Betriebssystem, Steuerprogramm, Teach-in-Programm, Positionier- und Diagnosemodule und die Bedienung der Serviceeinheit.

Resümee

Die Mikroelektronik und insbesondere die Mikroprozessortechnik hat die Entwicklung von Industrie und Wirtschaft der DDR in den letzten Jahren

maßgeblich beeinflusst. Eine hervorragende Rolle spielte dabei der leistungsfähige MP U 880, unter den vielen Anwendungsfällen gibt es solche mit 4stelligen Produktionsstückzahlen pro Jahr. Immer stärker wird auch das Leben unserer Bürger von dieser Technik beeinflusst. Viele von ihnen haben sich deshalb auf diesem Gebiet weiter qualifiziert, und auch im Bereich der Volksbildung wurden Maßnahmen ergriffen, die dieser Entwicklung Rechnung tragen. Rückblickend darf man einschätzen, daß der U 880 wesentlich zu einer schnellen Weiterentwicklung der Technik und zu einem weiteren Ansteigen der Arbeitsproduktivität beigetragen hat. Weitere MP- und MR-Systeme sind ihm inzwischen gefolgt bzw. sind in Entwicklung, die den Wünschen nach höheren Verarbeitungsbreiten und -geschwindigkeiten, höherer Komplexität (durch Integrationsgrade bis zu 35000 Transistorfunktionen), verbesserter Programmierbarkeit usw. Rechnung tragen. Sie werden in den kommenden Jahren dazu beitragen, die Effektivität der Arbeit in Industrie, Verwaltung und im privaten Bereich weiter zu erhöhen.

Autor:

Dr.-Ing. Wilhelm Leupold

DDR 8019 Dresden

Vollsackstraße 7

Lektor an der Ingenieurhochschule Dresden
Sektion Informationselektronik

Kleinstrechentechnik in der Hand des Ingenieurs



Seit Menschen rechnen, wird zugleich ihr Bestreben sichtbar, sich diese Tätigkeit zu erleichtern. Dies kann sowohl durch die Nutzung neuer und einfacher durchführbarer *Rechenverfahren* als auch durch den Einsatz von *Rechenhilfsmitteln* oder Rechenwerkzeugen erfolgen. So läßt sich das altehrwürdige Rechenbrett bis auf etwa 1100 v.u.Z. zurückverfolgen, wo es, als Suan-Pan bezeichnet, im alten China in Benutzung war. Die Rechenmeister und -knechte um Johannes KEPLER (1571 bis 1630) hatten für die Ableitung der KEPLERSchen Gesetze alle Hände und Köpfe voll zu tun. So suchten die Rechenmeister nach Verfahren, um die aufwendigen Multiplikationen und Divisionen möglichst durch Additionen und Subtraktionen zu ersetzen. Der Durchbruch erfolgte hier durch die Einführung der Logarithmen, die der französische Mathematiker LAPLACE (1749–1827) mit folgenden Worten lobte: »Die Erfindung der Logarithmen kürzt monatelang währende Berechnungen bis auf einige Tage ab und verdoppelt dadurch sozusagen das Leben.«

Nachdem 1614 vom schottischen Mathematiker NAPIER (lat.: Neper) (1550 bis 1617) die erste Logarithmentafel eingeführt wurde, war die Möglichkeit realisierbar, das rationelle Rechenverfahren mit Logarithmen in ein ein-

faches Rechenhilfsmittel »hineinzu packen«. Beim 1625 eingeführten »Logarithmenlineal« des Engländers GUNTER (1561–1626) mußten die logarithmierten Strecken noch mit einem Zirkel abgegriffen werden. Die Aneinanderreihung (Addition) dieser Strecken auf der logarithmischen Skale entsprach dann einer Multiplikation. Rund 20 Jahre später führten dann WINGATE (1593–1656) und PARTRIDGE (1603–1686) die verschiebbare Zunge ein. Damit war der Rechenstab geboren, an dessen Prinzip sich bis heute nichts geändert hat.

Parallel zu diesem einfachen *Analogrechner* (Rückführung des Rechnens mit Zahlen auf den Vergleich der Längen von Strecken) wurden im 17. Jahrhundert auch mechanische *Digitalrechner* konstruiert. Am bekanntesten sind die Maschinen von SCHICKARDT (1592–1635), PASCAL (1623–1662) und LEIBNIZ (1646–1716). PASCALS Addier- und Subtrahiermaschine können Besucher des Mathematisch-Physikalischen Salons in Dresden auch heute noch bewundern (Bild 1). Und in der Begründung von LEIBNIZ zum Bau seiner Rechenmaschine wird die eingangs gemachte Feststellung untermauert. Er schrieb 1671: »Eines geistig hochstehenden Mannes ist es unwürdig, seine Zeit zu vertun mit sklavischer Rechenarbeit, denn mit einer Maschine

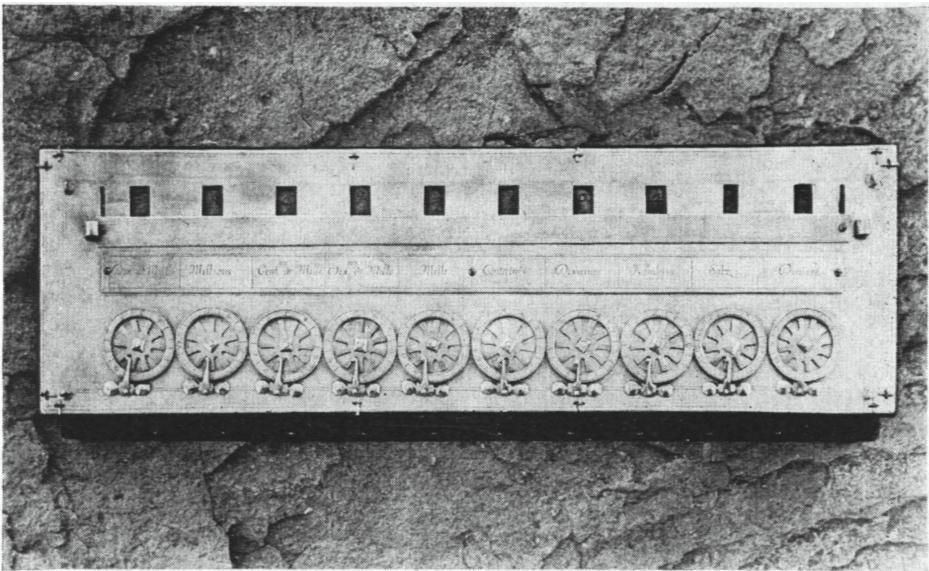


Bild 1. Rechenmaschine von PASCAL
 (Reproduktion aus GRÖTSCH, H.;
 KARPINSKI, J.: Dresden – Mathematisch-
 Physikalischer Salon. – Leipzig:
 Seemann-Verlag, 1979)

könnte auch der Allerdümmste die
 Rechnung sicher ausführen. «

So wurden Analog- und Digitalrechner
 Hand in Hand benutzt, bis um 1880
 mit dem Voranschreiten des Inge-
 nieurwesens die Massenproduktion von
 Rechenstäben einsetzte. Seit dieser Zeit
 gehörte der meist 25 cm lange Stab
 aus Holz, später Metall oder Plast zum
 »Zunftsymbol« des Ingenieurs wie der
 Äskulapstab zum Arzt.

Ein entscheidender Durchbruch bei den
 Digitalrechnern wurde 1941 erreicht.
 Der deutsche Ingenieur Konrad ZUSE
 (geb. 1910) baute die erste, mit einem
Programm steuerbare, elektrome-
 chanische Rechenmaschine Z 3. Sie arbei-
 tete mit insgesamt 2000 Relais zum
 Rechnen und Speichern und benötigte
 für eine Multiplikation rund 3 Sekun-
 den. Sie war aber die erste, die konse-
 quent die Vorzüge des *binären Zahlen-*

systems nutzte, und über die Zuver-
 lässigkeit ihres Nachfolgers, der Z 4,
 schreibt ZUSE selbst: »An der ETH
 (Eidgenössische Technische Hoch-
 schule, H.G.) in Zürich arbeitete die
 Z 4 so zuverlässig, daß sie nachts ohne
 Aufsicht lief. ... Es war ein eigen-
 artiges Gefühl, nachts um drei Uhr in
 die völlig menschenleere ETH zu gehen
 und im oberen Stock die Z 4 rattern zu
 hören. Sie war ja noch lochstreifenge-
 steuert und hatte mechanisch bewegte
 Teile, so daß man von ihr sowohl opt-
 tisch als auch akustisch unterhalten
 wurde. «

Schließlich gingen aus diesen Rechnern
 von ZUSE dann die leistungsfähigen
programmierbaren Elektronenrechner
 hervor, die zunächst mit Elektronen-
 röhren, dann mit Transistoren und
 schließlich mit integrierten Schalt-
 kreisen arbeiteten. Und so klappte bis
 zum Ende der 60er Jahre eine große
 Lücke zwischen dem leistungsfähigen,
 aber auch in jeder Hinsicht aufwendi-
 gen Rechenzentrum mit Groß- und
 Kleinrechnern und dem einfachen, in
 Geschwindigkeit und Genauigkeit sehr

bescheidenen] Rechenstab des Ingenieurs. 1970 begann sich diese Lücke zu schließen. Die japanische Firma CANON stellte den ersten Taschenrechner vor. Damit wurde ein Rechenhilfsmittel geschaffen, das klein, handlich und einfach bedienbar wie ein Rechenstab, zugleich aber fast so genau und schnell wie ein stationärer Rechner in einem Rechenzentrum war. Die ersten dieser elektronischen Taschenrechner trugen zum Teil die Bezeichnung SR, die Abkürzung für *slide-rule* (deutsch: Rechenstab). Dann ging es steil aufwärts. 1972 erschien der erste anspruchsvolle Taschenrechner für Wissenschaftler und Techniker, der auch den Bergsteiger CRIS BONINGTON im gleichen Jahr auf den Mount Everest begleitete. 1974 setzte die Autofirma MATRA-SIMCA einen Taschenrechner ein, um die 24 Stunden von Le Mans zu gewinnen, und 1975 waren Taschenrechner beim Koppelmanöver von SOJUS und APOLLO dabei.

Mit der Fertigung programmierbarer Taschenrechner wurde in dieser Zeit die Lücke zwischen den Kleinen und den Großen noch enger geschlossen, da die Kleinen nun auch die Programmierbarkeit von den stationären Rechnern im Rechenzentrum übernommen hatten. Solche *programmierbare Taschenrechner* unterstützten schon Segelyachten und Ballons als Navigationshilfe, und 1981 war ein moderner programmierbarer Taschenrechner an Bord des Space-Shuttle »COLUMBIA«.

Es gibt wohl kaum einen Bereich, bei dem seit 1970 Preisverfall mit gleichzeitigem Leistungsanstieg so rasant vonstatten gehen wie auf dem Gebiet der Tisch- und Taschenrechner, die gemeinsam als Kleinstrechner bezeichnet werden sollen. Ein Funktionselement, das 1958 noch 10,00 M und 1970 noch 10 Pfennig kostete, hatte 1980 einen Preis von nur 0,1 Pfennig. Ein heute

gebauter programmierbarer Taschenrechner mit einer Masse von 300 g übertrifft in seinem Leistungsvermögen einen 10 Jahre alten Kleinrechner (z.B. SER 2d), der den Platz von zwei Schreibtischen einnahm. Japan produzierte 1970 etwa 1,4 Millionen Taschenrechner, 1980 waren es aber 52 Millionen Stück, und 1990 sollen voraussichtlich 66,5 Millionen Taschenrechner produziert werden. Da mutet es schon eigenartig an, daß in Japan auch heute noch der Umgang mit dem Soroban (Rechenbrett) gelehrt wird und regelrechte Wettkämpfe stattfinden. Der Leiter der japanischen Computer- und Druckerfirma EPSON, TOKUJI TSUTSUMI, rechnet nach wie vor mit seinem Soroban, und japanische Firmen ziehen bei Einstellungsgesprächen auch die Leistungen der Bewerber auf dem Soroban mit hinzu.

In den sozialistischen Ländern hat sich das Angebot an Taschenrechnern in den letzten Jahren stark verbessert. Sorgen bereitet noch die mangelnde Verfügbarkeit programmierbarer Kleinstrechner. Nur in der Sowjetunion und der Volksrepublik Ungarn sind einige Typen programmierbarer Taschenrechner im Angebot. Der Ingenieur in der DDR kann den programmierbaren Tischrechner K 1003, Hersteller ist das Kombinat ROBOTRON, mit Magnetkarten- und Druckereinheit nutzen. Mit diesem einen programmierbaren Rechnertyp, der die Größe einer Schreibmaschine hat und fast 20 kg wiegt, ist das Angebot aber zu einseitig und läßt leider in der täglichen Praxis keine sinnvolle Abstufung in Abhängigkeit von den Leistungsanforderungen zu. Im Gegensatz dazu ist das Angebot an nichtprogrammierbaren Taschenrechnern (MR-Serie mit LCD-Anzeige) für Schule und Beruf sinnvoll gestaffelt.

Die Handlichkeit der modernen Kleinst-

rechner eröffnete die Möglichkeit, die Rechner zu den Daten zu bringen. Diese Dezentralisierung zeigt sich darin, daß der wissenschaftlich-technische Rechner auf dem Schreibtisch des Ingenieurs steht und der Steuerrechner direkt in die Werkzeugmaschine eingebaut ist. In früheren Jahren mußten noch die Daten zu den stationären Rechnern gebracht werden. Dabei traten oft unzumutbar große Wartezeiten für den Konstrukteur oder Technologen auf. Heute wird ihm die Möglichkeit gegeben, die Rechnungen selbst und zu jeder beliebigen Zeit am Schreibtisch auszuführen. Allerdings verlangt das vom Ingenieur ein bestimmtes Maß an rechentechnischem Wissen.

In Anbetracht dieser Tatsache stellt sich der Ingenieur in der Industrie oder im Forschungsinstitut vielleicht die Frage: Was muß ich tun, um diese neue Technik nutzen zu können, damit ich in weniger Zeit besser entwickeln, konstruieren und projektieren kann? Die Beantwortung setzt zunächst die Kenntnis über verfügbare nichtprogrammierbare und programmierbare Tisch- und Taschenrechner voraus. Steht der Rechner nach erfolgtem Kauf auf dem Tisch, dann beginnen eigentlich erst die Probleme für den Ingenieur. Während die Bedienung eines nichtprogrammierbaren Taschenrechners mit Hilfe der Bedienungsanleitung noch einfach zu erlernen ist, stellt sich bei der Einsatzvorbereitung eines programmierbaren Kleinstrechners eine gewisse *Schwellenangst* ein. Der Ingenieur möchte nun die Tatsache nutzen, daß der Zeitaufwand beim Einsatz programmierbarer Kleinstrechner nur noch rund ein Fünftel des Aufwandes beim Einsatz nichtprogrammierbarer Rechner beträgt, aber die geistige Investition erscheint ihm aus Unkenntnis der Dinge außerordentlich hoch. Zwei Gründe sind daran maßgeblich beteiligt:

1. Der Ingenieur hat es bisher nicht gelernt, seine Probleme in logische und damit in Rechenprogramme umsetzbare Schritte umzuwandeln. Der Anfänger ist einerseits enttäuscht, wie wenig der Rechner kann, andererseits aber erstaunt, wie viel man daraus, unter Einsatz des eigenen Geistes, in einem Rechenprogramm machen kann. Eine besondere Bedeutung kommt hierbei den numerischen Näherungsverfahren zu, die bisher aus Aufwandsgründen oft stiefmütterlich behandelt wurden.

2. Der Ingenieur hat das Programmieren in einer problemorientierten Sprache, in Assemblersprache oder als Tastenprogrammierung für einen bestimmten Kleinstrechner gar nicht oder nicht ausreichend gelernt. Ein methodisch schlecht aufbereitetes Bedien- und Programmierhandbuch führt dabei noch zur Erhöhung dieser Schwellenangst. Leider trifft das auch für das Programmierhandbuch des Tischrechners K 1003 zu.

Daraus ist zunächst abzuleiten, daß der Umgang mit dem Kleinstrechner, ebenso wie das Autofahren, erst *gelernt werden muß*. Die sich daraus ergebenden Schlußfolgerungen sollen im folgenden am Beispiel des programmierbaren Tischrechners K 1002/1003 des Kombi-RON erläutert werden. Sie gelten sinngemäß auch für andere programmierbare Kleinstrechner.

Natürlich kann der programmierbare Tischrechner K 1002/1003 auch als normaler Tischrechner genutzt werden. Das entspräche einer Dauerbenutzung des ersten oder zweiten Ganges beim Kraftfahrzeug. Erst die Nutzung der Programmierfähigkeiten des Rechners entspricht dem vierten oder fünften Gang. Etwa 60 % bis 80 % des Gesamtaufwandes für elektronische Rechen-technik verschlingt die *Software*, also die Herstellung von Rechenprogram-

men. Ein Programmhersteller hat das überspitzt einmal folgendermaßen ausgedrückt: Wir produzieren Software, und damit sie funktioniert, brauchen wir das Stückchen Blech, das wir Computer nennen!

Der wunde Punkt ist, wie schon erwähnt, die *geistige Investition* für den Ingenieur. Ihm bietet sich folgendes »Stufenproblem« dar:

1. Stufe:

Bedienung des Rechners lernen

2. Stufe:

Erkundungen über schon existierende und nachnutzbare Programme anstellen, die interessierenden Programme kaufen und an Hand der Bedienungsanleitung nutzen

3. Stufe:

Programmierung des Rechners lernen und selbst Programme für die eigenen Probleme schreiben.

Zum Beschreiten der 1. Stufe sollen die Rechner K 1002 und K 1003 kurz vorgestellt werden. Der Arbeitsspeicher des K 1002 umfaßt 4016 Byte, der des

K 1003 3992 Byte. Für einen Programmschritt wird 1 Byte und für eine abgespeicherte Zahl (12 Stellen, von denen 10 angezeigt werden) werden 8 Byte benötigt. Die Programmschritte (Befehle) werden im Programmspeicher und die Zahlen im Datenspeicher des Rechners gespeichert. Zur optimalen Ausnutzung von Programm- und Datenspeicher kann deren Grenze über die Tastatur vorgegeben werden. Die Bedieneinheit des Rechners (Bild 2) gliedert sich in folgende Bestandteile:

(1): Zahleneingabe und vier Grundrechenarten für manuelle Berechnungen und für die Programmierung.

(2): Tasten für die Organisation des Programmablaufes. Bei der Programmabarbeitung werden vorwiegend die Tasten STM und ST benutzt.

(3): Einstellen der Betriebsart beim Eintasten und Testen von Programmen.

(4): Lese- und Schreibtaste der Magnetkarteneinheit.

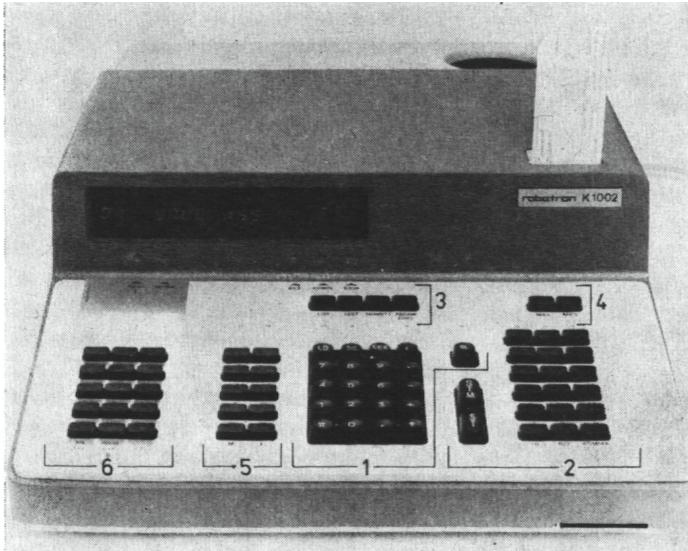


Bild 2
Programmierbarer
Tischrechner K 1002
mit Magnetkarte
im Einleseschacht

(5): Speicheroperations- und Funktionstasten $\frac{1}{x}$, x^2 , \sqrt{x} .

(6): Tastenbelegung in Abhängigkeit vom eingesetzten Funktionsblock. In der Grundausstattung ist der Funktionsblock Mathematik eingesetzt. Damit werden Winkel-, Exponential- und Logarithmusfunktionen berechnet. Dieser Block ist gegen den Funktionsblock Statistik austauschbar.

Vom Nutzer der Programme, die auf Magnetkarten gespeichert sind, werden nur die Tasten STM, ST und die Tasten der Felder (1) und (4) häufig benutzt. damit ist die Programmabarbeitung einfach und erfolgt bei Beachtung der zum Programm gehörenden Bedienungsanleitung absolut fehlerfrei. Die Ergebnisse werden mit einer programmierten Anzahl von Kommastellen angezeigt. Ganz links im Anzeigefeld befindet sich ein zweistelliger Numerateur, der zur Kennzeichnung von Ein- und Ausgabestellen bei der Abarbeitung eines Programmes dient. Die Zahl im Numerateur liefert damit über die Bedienungsanleitung die Zuordnung des Ergebnisses zur entsprechenden physikalischen Größe (z.B. Stromstärke in A, Grundzeit in min).

Das Nachfolgemodell, der K 1003, ist mit einem Thermodruckwerk ausgestattet, das den Ausdruck der Zwischen- und Endergebnisse gemeinsam mit Text gestattet. Diese Druckbogen können auf Vordrucke geklebt werden und stehen damit ohne nochmaliges Abschreiben zur weiteren Bearbeitung bereit. Ein Beispiel als Tabellendruck zur Berechnung der Aufprallgeschwindigkeit aus der eingegebenen Höhe zeigt Bild 3.

Folgende Hinweise sollen beim Beschreiten der 2. Stufe helfen. Vom Rechnerhersteller wird ein Funktionsblock Statistik geliefert, der die Berechnung von Mittelwerten, Streuun-

gen, Zufallszahlen, Regressions- und Korrelationskoeffizienten erheblich erleichtert. Für einfache Anwendungen erübrigen sich hier eigene Rechenprogramme, oder die Leistungen des Statistikblockes werden in eigene Programme eingebunden. Große Aktivitäten bei der Herstellung und dem Angebot von Programmen für den K 1002 gehen vom Arbeitskreis PKR K 1002 der Bauakademie der DDR aus. Das erarbeitete Programmverzeichnis enthält über 400 Programme, die vorwiegend das Bauwesen, aber auch Vermessungstechnik, Schweißtechnik und Ökonomie betreffen.

Allerdings ist damit das Problem »Gewußt wo« noch nicht gelöst. Leider gibt es *keine zentralen Informationen* vom Hersteller, so daß die Suche über die Literatur als Ausweichlösung bleibt. Schlüsselzeitschriften zu diesem Gebiet sind die Zeitschriften »Neue Technik im Büro« und »Rechentechnik/Datenverarbeitung«. Darüber hinaus erscheinen zunehmend Veröffentlichungen von fachspezifischen Programmen in den einzelnen Fachzeitschriften (z. B. Gießereitechnik, Radio Fernsehen Elektronik, ZIS-Mitteilungen, Schweißtechnik, Fertigungstechnik und Betrieb, Bauforschung/Baupraxis u. a.). In der Informationsstelle des Zentralinstitutes für Schweißtechnik der DDR in Halle wurde eine *Bibliographie zu Programmveröffentlichungen für den K 1002* zusammengestellt. Beim Programmkauf erhält der Nutzer im allgemeinen die Programmbeschreibung mit Bedienungsanleitung und Berechnungsbeispiel sowie die dazugehörigen Magnetkarten.

Hinweise zum Beschreiten der 3. Stufe: Zweifellos ist es je nach Kenntnisstand eine mehr oder weniger hohe Investition für den Ingenieur, das *Programmieren* des Rechners zu erlernen. Allerdings sprechen die gegenwärtige Be-

ERMITTLUNG DER AUFRALL- GESCHWINDIGKEIT

5,0M	36KM/H
6,3M	40KM/H
8,5M	46KM/H
9,0M	48KM/H
11,5M	54KM/H
15,0M	62KM/H
3,6M	30KM/H
26,0M	81KM/H
10,0M	50KM/H
45,5M	108KM/H
49,0M	112KM/H
57,6M	121KM/H

Bild 3. Rechnerausdruck des K 1003 zur Ermittlung der Aufprallgeschwindigkeit beim freien Fall

deutung der Mikroelektronik (im K 1002 steckt ja der Mikrorechner K 1510) und die langjährige Nutzung der einmal erworbenen Kenntnisse für diesen Aufwand. Programmierkenntnisse können aus dem mitgelieferten Bedienungshandbuch und in Lehrgängen des Rechnerherstellers erworben werden. Das Zentralinstitut für Schweißtechnik der DDR hat gemeinsam mit dem Bezirksverband Halle der KDT einen Lehrgang für die programmierbaren Tischrechner K 1002/1003 ins Leben gerufen. Er soll dem Anfänger die »Angst« vor dem Rechner nehmen und Grundlagen seiner Programmierung vermitteln. Über 300 Ingenieure aus dem Bezirk Halle haben diesen Lehrgang inzwischen absolviert.

Die praktische Arbeit mit dem K 1002/1003 kann in jedem Betrieb unterschiedlich organisiert sein. Es ist aber sinnvoll, daß jeder Nutzer seine eigene Programmbibliothek in Form von Magnetkarten und Bedienungsanweisungen in seinem Schreibtisch aufbewahrt. Der Rechner sollte auf einem kleinen Organisationswagen befestigt und mit diesem transportabel sein. Im allgemeinen ist der Rechner dann noch »frei verfügbar«, wenn ein Verhältnis 10 Ingenieure : 1 Rechner nicht überschritten wird.

Da der Rechner K 1002/1003 der am häufigsten vertretene programmierbare Kleinstrechner ist, sollen in dieser Broschüre auch Programme für diesen Rechner veröffentlicht werden.

Das Gebiet der Kleinstrechenteknik entwickelt sich stürmisch. Leistungsgrenzen und begriffliche Abgrenzungen sind in Anbetracht von programmierbaren Taschen-, Aktentaschen-, Tisch-, Heim- und Personalcomputern außerordentlich schwierig. Allerdings beginnt sich abzuzeichnen, daß die Tastenprogrammierung (ein Tastendruck löst den entsprechenden Befehl aus) der Kleinstrechner nur eine vorübergehende Programmierungsart war. Gegenwärtig und zukünftig werden auch die »Kleinen« mit problemorientierten Programmiersprachen wie BASIC oder PASCAL programmiert. Vorteile sind einfachere Programmierung und Unabhängigkeit des Programms vom Rechnertyp. Aber auch hier muß gelernt werden, *fangen wir rechtzeitig an!*

Autor:

Dr. Hannes Gutzer
DDR 4090 Halle-Neustadt
Am Südpark 581/6

Leiter der Informationsstelle
des Zentralinstituts für Schweißtechnik
Halle

Erfahrungen beim Einsatz programmierbarer Kleinstrechner in der Schule



Da die Mikroelektronik, die elektronische Rechentechnik und die Informationsverarbeitung immer stärker alle Bereiche des täglichen Lebens durchdringen, stellt sich für alle Stufen der Ausbildung immer deutlicher die Aufgabe der *frühzeitigen Vorbereitung* der Schüler und Studenten auf die neuen und ständig wachsenden Anforderungen der Zukunft. In den meisten Betrachtungen steht hier – m.E. völlig zu recht – die Anwendung der Rechentechnik als Mittel und als Gegenstand und Medium der Ausbildung an Hoch- und Fachschulen im Mittelpunkt. Die Vorbereitung auf die Anforderungen der Zukunft bedeutet auch die Einstimmung oder die »Gewöhnung« der Kinder und Jugendlichen an die moderne Technik und speziell die Mikroelektronik und Mikrorechentechnik.

Für ein Kind ist ein Auto in der heutigen Zeit ein normales Objekt des täglichen Lebens. Welches Verhältnis haben in der Gegenwart Kinder und Jugendliche zur modernen Elektronik und Rechentechnik? Es ist erstaunlich, welche Haltungen und Einstellungen man da antreffen kann. Leider muß man hier auch zahlreiche Erwachsene mit einbeziehen. Die Palette der Ansichten reicht vom einfachen Unverständnis des Neuen und Revolutionierenden der Mikroelektronik bis zum Nachtrauern der guten alten Zeit der

Arbeit mit den »einfachen und verständlichen« Röhren.

Viel unvoreingenommener gehen Kinder vor. Sie nehmen z. B. den Taschenrechner als ein interessantes, neues und abwechslungsreiches Spielzeug und nutzen ihn dann völlig selbstverständlich. Unsere Aufgabe als Schule muß nun darin bestehen, das anfänglich sehr oberflächliche *Interesse systematisch zu vertiefen*. Eine günstige Möglichkeit, die Kinder und Jugendlichen – also die Schüler der polytechnischen Oberschule und der erweiterten Oberschule – mit der Mikroelektronik und Rechentechnik möglichst frühzeitig zu konfrontieren, besteht im Einsatz von Kleinstrechentechnik in der Schule.

Es sollte unser Ziel sein, den Schülern die Technik zu einer ganz natürlichen Sache werden zu lassen. Die Technik muß von den Schülern als »zweite Natur des Menschen«, wie es KARL MARX formulierte, erfaßt werden. Positive Ergebnisse in diesem Entwicklungsprozeß dokumentieren sich z. B. darin, daß die an unserer Schule mit dem Kleinstrechner arbeitenden Schüler einschätzen, daß »ein Schüler, der näher und intensiver mit dem Rechner vertraut wurde, nicht wieder von ihm loskommt«. Nicht nur diese und ähnliche Aussagen der Schüler, auch die praktische Erfahrung aus der Arbeit mit den Schülern bestätigt diese Aussagen.

Schüler, die an den ersten Stunden z. B. der Arbeitsgemeinschaft regelmäßig teilnahmen, die damit Genaueres über den Rechner erfahren haben und die gewisse erste praktische Erfahrungen mit dem Rechneinsatz gesammelt haben, sind immer wieder daran interessiert, auch weiter und intensiver mit dem Gerät zu arbeiten. Dies zeigt sich auch daran, daß nach einigen Stunden keine Fluktuationserscheinungen mehr auftraten. Erfahrungen belegen, daß Schüler der Altersgruppe 12 bis 16 oder 18 Jahre in der Lage sind, *anspruchsvolle und Schöpfertum fordernde Programme* zu erstellen, die sich auch durch große Perfektion und Fehlerfreiheit auszeichnen. Ich bin deshalb der Meinung, daß wir wertvolle Zeit und auch Schöpferkraft verlieren, wenn unsere Jugendlichen frühestens mit 18 bis 21 Jahren zu Beginn des Studiums mit Fragen der Programmierung konfrontiert werden. Besonders nachdenkenswert finde ich dabei, daß gerade die Softwareentwicklung und der Systementwurf eine wichtige Möglichkeit darstellen, um weitere Fortschritte zur Erhöhung der Effektivität zu erreichen.

Einen weiteren Grund für die Einführung der modernen Rechentechnik und grundlegender Gedanken der Programmierung in den Unterricht bzw. in die außerunterrichtliche Arbeit an den Schulen sehe ich auch darin, daß bekanntlich der *Arbeitsstil eines Menschen entscheidend in der Jugend geprägt* wird. Der einmal erworbene Arbeitsstil wird dann in jedem späteren Arbeitsgebiet beibehalten. Für uns als Lehrer ergibt sich damit die Aufgabe, den Arbeitsstil der Schüler stets unter diesem Aspekt zu entwickeln bzw. zu festigen. So bietet besonders die Programmierung günstige Möglichkeiten, um z. B. das *Denken in Strukturen, das logische und planmäßige Vorgehen* bei Problem-

lösungen und eine gewisse *Strenge im logischen Denken* zu schulen.

Aus diesen Überlegungen heraus haben wir begonnen, im fakultativen Unterricht und in Vorträgen der mathematisch-naturwissenschaftlich-technischen Schülerakademie der Abiturstufe unserer Schule diese Aspekte stärker zu berücksichtigen.

Seit September 1981 baue ich gemeinsam mit dem VEB ROBOTRON-Elektronik Zella-Mehlis an unserer erweiterten Oberschule »Artur Becker« Suhl ein »Mathematisch-rechentechnisches Kabinett« auf. Vom VEB ROBOTRON-Elektronik Zella-Mehlis wurde im September 1981 ein programmierbarer Kleinstrechner vom Typ PKR ROBOTRON K 1003 zur Verfügung gestellt. An der Schule habe ich dann diesen Rechner im fakultativen Unterricht der Klassen 11 und 12 eingesetzt. Gleichzeitig nahm eine Arbeitsgemeinschaft für diese Klassenstufen ihre Arbeit auf. Insgesamt werden damit mehr als ein Viertel der Schüler eines Jahrganges der Schule im Verlaufe der zweijährigen Ausbildung mit den Grundlagen der Rechentechnik und Programmierung vertraut gemacht. Seit September 1982 wird zusätzlich eine Arbeitsgemeinschaft für die Klassen 8 und 9 des gesamten Stadtkreises durchgeführt.

Mit Beginn des Schuljahres 1982 wurde der Aufbau des mathematisch-rechentechnischen Kabinetts durch die Technische Hochschule Ilmenau durch die Übergabe zweier programmierbarer Taschenrechner vom Typ BS 21 unterstützt.

Zum Inhalt der Lehrgänge und Arbeitsgemeinschaften

Der programmierbare Kleinstrechner K 1003 und die programmierbaren Taschenrechner BS 21 werden seit 2

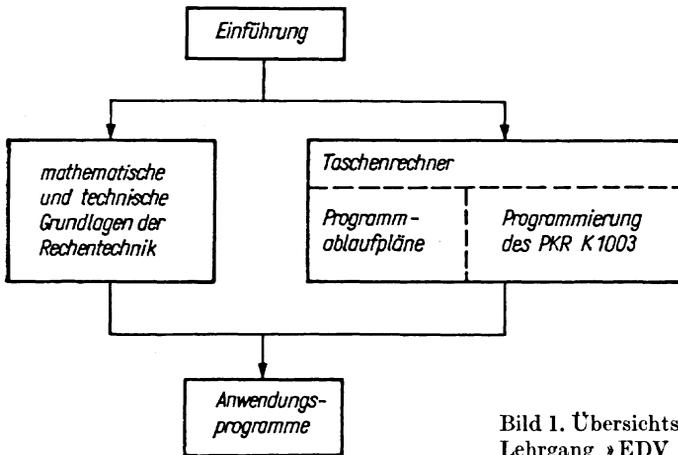


Bild 1. Übersichtsplan für den 50-Stunden-Lehrgang »EDV und Kleinstrechner«

bzw. 1 Jahr im fakultativen Unterricht und in Arbeitsgemeinschaften der Klassenstufen 8 bis 12 eingesetzt. Dabei ergab sich zwangsläufig für alle Formen ein ähnlicher Plan. Wobei die theoretischen Aspekte im 50-Stunden-Lehrgang des fakultativen Unterrichtes wesentlich stärker betont wurden als in den Zusammenkünften der Arbeitsgemeinschaften, in denen stärker dem Charakter der Freizeitbeschäftigung Rechnung getragen werden mußte. In Bild 1 ist ein zeitlicher Überblick über die Inhalte des fakultativen Lehrganges »EDV und Kleinstrechner« gegeben.

Deutlich erkennbar ist die parallele Behandlung der mathematischen und technischen Grundlagen der Rechentechnik neben der Einführung in die Programmierung. Diese Parallelbehandlung bietet den wesentlichen Vorteil, daß man unmittelbar zusammen mit der Behandlung der theoretischen Grundlagen sofort sichtbar machen kann, wie und warum dies für die praktische Arbeit mit programmierbarer Rechentechnik notwendig ist. Gleichzeitig wird durch diese Form der Stoffaufteilung auch die Phase bis zur un-

mittelbaren Arbeit mit dem Rechner verkürzt und wirkt somit positiv auf das Interesse der Schüler.

In allen Lehrgängen und Arbeitsgemeinschaften wurden die Notwendigkeit und die große Bedeutung der Einführungsstunde sichtbar. Neben allgemein üblichen einführenden Bemerkungen über die historische Entwicklung der Rechentechnik, die Möglichkeit und Notwendigkeit des Einsatzes der Rechentechnik u. ä. sollte möglichst jeder Schüler selbst einmal den Rechner bedient haben. Durch den unmittelbaren Kontakt mit dem Rechner wird bei den Schülern eine gewisse Scheu vor dem Gerät abgebaut. Gleichzeitig erhalten die Schüler dadurch auch eine erste, wenn auch sehr verschwommene Vorstellung von der Leistungsfähigkeit des Kleinstrechners, und es wird der Wunsch geweckt, selbst derartige Programme zu erarbeiten.

Als Programmbeispiele für die Einführungsstunde, in der die Schüler relativ selbständig vorhandene Programme nutzen sollen, bieten sich *Spielprogramme* an. Diese Programme können nach der Eingabe durch den Lehrer gestartet werden und sollten so organi-

siert sein, daß sie dem Schüler die weiteren Handlungen vorgeben und der Schüler fast ohne Hilfe des Lehrers den Rechner bedient. Besonders geeignet sind m.E. die recht bekannten und auch weitverbreiteten Spiele, wie Master Mind oder Mondlandung. Wichtig ist eigentlich nur, daß das Spiel einfache Spielregeln hat, einen ‚Dialog‘ Schüler – Rechner erfordert und eine – eventuell nur scheinbare – Siegeschance für den Schüler bietet. Andere Spiele, wie das bekannte Life-Spiel, sind in dieser Phase für die Schüler völlig unattraktiv, da die Maschine selbständig irgend etwas produziert. Der Schüler würde damit wieder in die Beobachterrolle – ähnlich einem Besuch im Rechenzentrum – ohne Einfluß auf die Maschine gedrängt.

Im gesamten Lehrgang werden immer wieder Spielprogramme einbezogen. Dabei ist es nicht entscheidend, ob die Programme von den Schülern erstellt werden oder ob sie nur genutzt werden. In beiden Fällen bewirkt der Einsatz von Spielen eine Auflockerung der Schülertätigkeit, vermittelt neue Impulse und liefert auch einen Beitrag zur Motivation für die weitere oft beschwerliche Beschäftigung mit den Grundlagen der Informationsverarbeitung und Rechentechnik. Es ist dabei immer wieder beeindruckend, mit welchem Eifer, Elan und Engagement sich die Schüler auf die Lösung von Programmierungsaufgaben stürzen. So haben zwei Schüler sich die Aufgabe gestellt, das bekannte Katz und Maus bzw. Hase und Jäger o.ä. benannte Spiel zu programmieren. Der Rechner sollte die Rolle der vier Fänger spielen. Innerhalb von gut einer Woche hatten diese beiden Schüler neben ihrer normalen schulischen Arbeit ohne wesentliche Hilfen dieses Programm vollständig fertig.

An dieser Stelle möchte ich nochmals

die Bedeutung und den Nutzen von Spielprogrammen für die Arbeit mit Schülern betonen. Es ist m.E. nicht entscheidend, an welchem konkreten Beispiel die Schüler den Umgang mit der Rechentechnik lernen, sondern daß sie sich mit *Interesse und Freude an der Arbeit* aktiv mit der Theorie und Praxis auseinandersetzen.

Bevor die Schüler in der Lage sind, derartige Programme selbständig zu erstellen, ist es notwendig, daß sie die fachlichen Grundlagen erlernt haben. Solange der Taschenrechner noch nicht im obligatorischen Unterricht der allgemeinbildenden Oberschule eingesetzt wird, haben wir 2 bis 3 Stunden des fakultativen Unterrichtes bzw. der Arbeitsgemeinschaft für seine genauere Behandlung verwandt. Besonderer Wert wurde dabei auf die Behandlung der *Registerstruktur* gelegt, da dies eine wichtige Voraussetzung für die Beherrschung des Taschenrechners ist. Damit wird auch eine gute Vorarbeit für die Anwendung der umgekehrten polnischen Notation beim K 1003 geleistet. Diese für alle Schüler neue logische Grundstruktur bereitete in der Anfangsphase erwartungsgemäß gewisse Schwierigkeiten. Nach recht kurzer Zeit hatten sich alle Schüler an diese Registerstruktur gewöhnt. Dabei wirkte sich wiederum günstig aus, daß die Schüler auch diese Übung bzw. Gewöhnung in der Arbeit am Rechner absolvierten.

Die Behandlung des Taschenrechners zeigte auch, daß viele Schüler den Taschenrechner nur als »Rechenmaschine für formalste und einfachste Rechnungen« einsetzen. Ein nicht unerheblicher Teil der Schüler ist nicht in der Lage, effektiv mit Speichern zu rechnen oder gar eine gestellte Aufgabe der logischen Grundstruktur des Taschenrechners entsprechend – mit oder ohne Hierarchie – umzuformulieren.

Nachdem die Schüler im Verlaufe des Lehrganges Grundlagen der Informationsverarbeitung und Rechentechnik kennengelernt haben, werden dann – teilweise auch parallel dazu – erste Programme gemeinsam aufgestellt. Dabei ist es wichtig, daß diese Beispiele einerseits einfach und überschaubar sind, andererseits aber die Wirksamkeit und den Nutzen der Rechentechnik demonstrieren. Es ist wichtig und auch richtig, wenn die Schüler sehr bald angehalten werden, selbständig oder mit geringen Hilfen Programme zu erstellen. So waren die Schüler nach spätestens 20 Stunden des Lehrganges in der Lage, völlig ohne Hilfe einfache Programme zu schreiben und auch am Rechner zu testen. Es gab auch Schüler, die wesentlich schneller das »Programmieren« erlernt hatten und auch bereits anspruchsvollere Programme erstellen konnten. Es muß auch noch erwähnt werden, daß in diesen 20 Stunden die Einführung, die Stunden zur Behandlung des Taschenrechners, der historischen Betrachtungen und mathematischer sowie technischer Grundlagen der Rechentechnik enthalten waren.

Selbstverständlich waren die Schüler auf ihre ersten eigenen Programme stolz, aber sehr bald trat der Wunsch nach »perfekten Programmen« auf. Die Programme sollten so ergänzt und überarbeitet werden, daß das Programm z.B. falsche Eingaben selbständig erkennt. Auch die Schüler der 8. und 9. Klasse machten hier keine Ausnahme. So gaben sie z.B. bei einem Programm zur Untersuchung von Zahlen auf Primzahleigenschaften keine Ruhe, bevor ihr Programm soweit perfektioniert war, daß es erkannte, daß die eingegebene Zahl negativ, Null, eins oder nicht ganzzahlig ist. Ähnliche Beispiele lassen sich in allen Schülergruppen finden. Man muß auch unter

diesem Aspekt bei der Auswahl der zu programmierenden Probleme das Niveau der entsprechenden Altersstufe beachten.

Im Verlaufe des Lehrganges verlagerten sich die Schwierigkeiten der Schüler im Umgang mit dem Rechner deutlich. Wenn es anfänglich die notwendige Strenge und Exaktheit der Gedankenführung war, so traten später mit zunehmender Programmlänge immer mehr Probleme z.B. bei der genauen Bestimmung der Sprungadressen, der indirekten Adressierung oder der Anwendung der Unterprogrammtechnik auf. An dieser Stelle wirkt sich die maschinenorientierte Programmiersprache des K 1003 nachteilig aus, da hier eine Vielzahl recht formaler Einzelbefehle notwendig ist. Auf die Schüler wirkt diese große Anzahl formaler Befehlsschritte ermüdend. Durch diesen Aufwand besteht auch die Gefahr, daß das Anliegen der Verdeutlichung des strukturellen Aspektes unter der Vielzahl der formalen Teilschritte verschüttet wird. Deshalb hätte hier eine höhere Programmiersprache – z.B. BASIC – wesentliche Vorteile. Der Vorteil der maschinenorientierten Programmiersprache besteht darin, daß sich die Schüler intensiver mit der Arbeit des Rechners befassen müssen und damit tiefgründiger in die Rechentechnik und die Technik der Programmierung eindringen.

Zum Programmieren gehört auch das Aufstellen einer entsprechenden *Programmdokumentation*. Sie enthält bei uns, neben einer Befehlsliste, Problemdarstellung, Speicherbelegung, Markenbelegung und dem Programmablaufplan auch eine relativ umfangreiche und ausführliche Anleitung zur Nutzung des Programms. Durch eine einheitliche Form der Dokumentation besteht damit die Möglichkeit, einmal aufgestellte Programme immer wieder

zu nutzen, d. h., die neuen Schüler können nach kurzer Zeit bereits die Programme älterer Schüler nutzen.

Manche Schüler haben erst im Laufe der Zeit die Notwendigkeit und den Nutzen einer Programmdokumentation verstanden. Erst als sie eigene Programme nach einer gewissen Zeit nicht mehr ohne die Hilfe der Dokumentation nutzen konnten, haben sie den Sinn dieser ihrer Meinung nach zusätzlichen und eigentlich überflüssigen Arbeit erfaßt.

Interessant war auch die Veränderung des Verhältnisses der Schüler zum Programmablaufplan. Zu Beginn der Beschäftigung mit der Programmierung wurde der Programmablaufplan als völlig selbstverständlicher Teil der Arbeit aufgestellt. Mit zunehmender Übung erschien den Schülern diese Arbeit als unnötig und überflüssig, da sie das alles im Kopf können und den Programmablaufplan nicht erst aufschreiben müssen. Mit der Zunahme des Schwierigkeitsgrades der Problemstellungen ergab sich dann ganz zwangsläufig die Einsicht, daß Programmablaufpläne *doch* erforderlich sind und eine wichtige Hilfe z. B. besonders bei zyklischen Programmteilen darstellen.

Zur gerätetechnischen Grundlage der Lehrgänge

In den Lehrgängen des fakultativen Unterrichtes und in den Arbeitsgemeinschaften werden der programmierbare Kleinstrechner K 1003 und die programmierbaren Taschenrechner BS 21 genutzt. Für die Einführung der Schüler in Grundlagen der Programmierung ist m. E. der Einsatz von programmierbaren Taschenrechnern – z. B. des sowjetischen BS 21 – nicht günstig. Der i. allg. sehr begrenzte Befehlspeicherumfang erfordert vom Nutzer

bereits eine gewisse Perfektion in der Programmierung, da man stets bestrebt sein muß, Programme so kurz wie nur möglich zu formulieren, um anspruchsvolle mathematische Probleme bearbeiten zu können. So kann man z. B. durchaus mit den 60 Befehlspeicherplätzen des BS 21 ein lineares Gleichungssystem mit 2 Unbekannten lösen, wobei man aber keine »unnötigen Befehle« einbauen darf. Zu einfache Probleme, die leicht mit dem programmierbaren Taschenrechner zu bearbeiten wären, würden ihrerseits die Einsicht der Schüler in den Nutzen der programmierbaren Rechentechnik nicht möglich machen.

Für die methodische Gestaltung des Lehrganges bzw. der Arbeitsgemeinschaft ergeben sich Vor- und Nachteile aus der Größe des Gerätes und der damit verbundenen Notwendigkeit, daß jeder Schüler oder zumindest je zwei Schüler einen programmierbaren Taschenrechner zur Verfügung haben müssen. Ein Vorteil besteht darin, daß jeder Schüler ständig am Rechner tätig ist und alle Bedienhandlungen in gleicher Front von allen gleichzeitig geübt werden können. Nachteilig wirkt sich aus, daß nur individuell gearbeitet werden kann. Eine Demonstration ist damit nicht möglich und beeinträchtigt die Möglichkeiten des Lehrers. Nachteilig ist für die Arbeit mit programmierbaren Taschenrechnern auch, daß infolge der mehrfachen Belegung der Tasten die in der Anzeige sichtbare zahlenmäßige Verschlüsselung der Befehle in keinem sichtbaren Zusammenhang zur Tastenanordnung steht und damit eine Liste der Befehlscode notwendig ist.

Durch den parallelen Einsatz des Kleinstrechners K 1003 und des Taschenrechners BS 21 wurde ein anderes Problem deutlich. Da die Schüler zuerst mit dem Kleinstrechner vertraut

wurden, erschien ihnen der programmierbare Taschenrechner nicht leistungsfähig genug und, da er neu war, auch recht kompliziert. Insgesamt zeigte sich bei der Mehrzahl der Schüler der Wunsch, weiterhin mit dem Kleinstrechner zu arbeiten. Andererseits konnte man feststellen, daß die Schüler, die sich mit dem Taschenrechner intensiver beschäftigt hatten, ihn als Hilfsmittel zur effektiven Programmierung des Kleinstrechners nutzten. Vorteilhaft ist dabei die im wesentlichen gleiche logische Grundstruktur beider Rechner. Schüler, die auch weiterhin den programmierbaren Taschenrechner nutzten, um Programme für den Kleinstrechner effektiver aufstellen zu können, machten von der Möglichkeit der Erprobung und der ständigen Kontrolle während der Phase der Programmierung Gebrauch. Es sind an dieser Stelle Ansätze eines unmittelbaren »Dialogbetriebes« zu erkennen. An dieser Stelle sieht man deutlich, daß eine optimale Arbeit mit den Schülern durch ein Mikrorechnersystem mit mehreren Terminalplätzen gegeben wäre. Man hätte dadurch die oben genannten Vorteile – die ständige und individuelle Arbeit am Rechner, sowie die Leistungsfähigkeit und Demonstrationsmöglichkeit – miteinander verbunden.

Besonders die Schüler der 8. und 9. Klasse vermißten beim Taschenrechner den alphanumerischen Drucker. Auch für alle anderen Schüler bedeutete es eine Umstellung, alle Ausgaben des Rechners nur numerisch realisieren zu können. Die alphanumerische Ausgabe des K 1003 – über den Thermodrucker – wirkt sich bei allen Schülern positiv auf das Interesse an der Rechentechnik aus. Für eine wirkungsvolle Beschäftigung der Schüler mit der Rechentechnik erscheint deshalb ein »alphanumerisches Dialogsystem« wesentlich.

Zum Verhältnis der Schüler zur Rechentechnik

In der Arbeit mit der programmierbaren Rechentechnik zeigten sich gewisse Entwicklungsetappen bei den Schülern.

In der ersten Etappe bestaunten die Schüler den Rechner als »Wunderding«, und das noch recht unspezifische Interesse wurde durch den Effekt des Staunens stimuliert. In der weiteren Beschäftigung erlangten die Schüler in einer zweiten Etappe mehr Verständnis für den Rechner und seine Wirkungsweise. Der Rechner wird zum konkreten Objekt für die Schüler, und ihr Interesse richtet sich immer stärker auf eine möglichst perfekte Handhabung des Rechners. In der eng damit verbundenen dritten Etappe versuchen die Schüler immer anspruchsvollere Programme zu erarbeiten. Sie versuchen auch die Grenzen des Rechners zu erforschen bzw. den Rechner »auszutricksen« und Fehler nachzuweisen. Dabei kommt es darauf an, daß man den Schülern bewußt macht, daß nicht der Rechner, sondern der Programmierer unexakt gearbeitet hat oder daß entsprechende Probleme durch Rundungsfehler des Rechners entstanden sind. In dieser Etappe besteht die Gefahr, daß die Beschäftigung mit dem Rechner zum reinen Handwerk ausartet. In der vierten Etappe tritt dann der Effekt ein, daß die Schüler durch die Arbeit am bzw. mit dem Rechner stimuliert werden, sich mit mathematischer Theorie zu beschäftigen. In dieser Etappe wird der Rechner für die Schüler zum echten Hilfsmittel. Ein Beispiel soll diese Etappe veranschaulichen. So kamen die Schüler auf die Idee, die Gamma-Funktion zu programmieren. In der Literatur hatten sie eine Formel gefunden, in der $\Gamma(z)$ mit Hilfe eines unendlichen Produktes

berechnet werden kann. Versuche mit dem K 1003 lieferten Rechenzeiten in der Größenordnung von Tagen für einen Funktionswert. Auch Versuche auf schnelleren Rechnern blieben erfolglos. Also ergab sich die Notwendigkeit, nach einer günstigeren Formel zu suchen. Nachdem auch die EULERSche Produktformel für die Gamma-Funktion nicht den gewünschten Erfolg brachte, da auch hier das Konvergenzverhalten zu schlecht ist, gelangten wir dann zur STIRLING'schen Reihe, und das entsprechende Programm liefert nun in weniger als einer Minute die Funktionswerte.

Es muß unser Ziel sein, alle Schüler bis zu dieser 4. Etappe zu führen. Im Verlaufe der bisherigen Tätigkeit des mathematisch-rechentechischen Kabinetts hat sich gezeigt, daß es nicht möglich sein wird, dieses Ziel mit allen Schülern in einem Jahr und erst recht nicht in allen Klassenstufen zu erreichen. Recht unproblematisch ist dies in der Arbeitsgemeinschaft der Klassen 11 und 12, da hier mathematisch interessierte Schüler das fachliche Interesse und auch die notwendige Vorbildung mitbringen. Bereits im fakultativen Unterricht der Klassen 11 und 12 bedarf es größerer Anstrengungen, um alle Schüler über die Etappe des »Handwerkeln« hinauszuführen. Für die Schüler der Arbeitsgemeinschaft der Klassen 8 und 9 wird man sich kaum das Ziel stellen, die Schüler bis zur 4. Etappe führen zu wollen. Es kommt hier darauf an, Interesse für die Rechentechnik und auch für die Mathematik selbst zu festigen und teilweise zu wecken.

Einige Ergebnisse der bisherigen Arbeit des mathematisch-rechentechischen Kabinetts

Von den Schülern des fakultativen Unterrichtes und der Arbeitsgemeinschaften wurden im Verlaufe der 2 Jahre bisher rund 50 Programme aus den Bereichen Algebra, Funktionen, einschließlich numerische Integration, Trigonometrie, Spiele, Zahlentheorie und Anwenderprogramme aufgestellt. Im vergangenen Schuljahr wurden dazu noch mehr als 40 Programme für den programmierbaren Taschenrechner BS 21 aufgestellt. Zu allen Programmen wurden Dokumentationen angefertigt, und die Programme liegen auch in Form von Magnetkarten zur sofortigen Nutzung bereit. Möglich war eine solche Breite und Anzahl von Programmen nur durch das Engagement, Interesse und die Einsatzbereitschaft der Schüler. So nehmen die Schüler am fakultativen Unterricht und teilweise noch einen Nachmittag an der Arbeitsgemeinschaft teil. Zusätzlich arbeiten die Schüler i. allg. noch einen Nachmittag pro Woche selbständig am Rechner und erproben ihre in häuslicher Arbeit entstandenen Programme. Darüber hinaus arbeiten die Schüler teilweise mehrere Tage in den Ferien wiederum selbständig am Rechner.

Für den Leiter des Lehrganges des fakultativen Unterrichtes bzw. der Arbeitsgemeinschaft ergibt sich die Aufgabe, immer wieder neue, für die Schüler interessante und ihrer Leistungsfähigkeit angepaßte Programmierungsaufgaben zu finden. Es ergibt sich auch die Notwendigkeit, solche Aufgaben zu finden, die den Schülern ein Erfolgserlebnis vermitteln. So haben wir z. B. ein Programm zur Auswertung der XXI. Bezirksolympiade Junger Mathematiker vorbereitet und eingesetzt.

Ein weiterer wichtiger Aspekt ist die notwendige Verbindung zur Praxis. Neben einer Anzahl von fertigen Programmen, die wir zu Anschauungszwecken kostenlos von verschiedensten Institutionen zur Verfügung gestellt bekamen, haben wir auch bereits Programme für Praxispartner angefertigt. So wird ein Programm für die Berechnung von Transformatoren erarbeitet, das die insgesamt knapp 4000 Speicherplätze des K 1003 voll ausnutzt.

Obwohl eine Einführung des Mikrocomputers im Unterricht der allgemeinbildenden Schule zur Zeit noch nicht aktuell ist, ist doch der Einsatz im fakultativen Unterricht eine aktuelle Frage. Am Beispiel der numerischen Integration läßt sich aber sehr gut erkennen, welche Möglichkeiten sich auch im obligatorischen Unterricht, z.B. durch numerische Simulationen, durch den Einsatz der Mikrocomputer ergeben werden.

Abschließende Bemerkungen

Die Einführung der modernen Rechentechnik in die Schule ist ein wesentlicher Beitrag zur Erhöhung der Qualität und Wirksamkeit der mathematischen Bildung und Erziehung der Schüler. So werden durch die Arbeit mit Rechnern die Exaktheit, die Beharrlichkeit, die Ausdauer, das logische Denken u.a. Eigenschaften weiter ausgeprägt.

Die bisherige Erprobung des Rechner-einsatzes hat gezeigt, daß der Einsatz im fakultativen Unterricht der Klassen 11 und 12 möglich und sinnvoll ist. Insgesamt kann man einschätzen, daß Schüler ab Klasse 8 und ganz sicher ab Klasse 10 im Verlaufe eines 50 Stunden umfassenden Lehrganges »EDV und Kleinstrechner« befähigt werden können, den programmierbaren Kleinst-

rechner K 1003 bzw. den programmierbaren Taschenrechner BS 21 selbständig zu nutzen. Dabei entwickelte sich ein großes Engagement bei den Schülern. Es ist notwendig, daß die Schüler nicht nur die Theorie und möglichst einfache Programmbeispiele vorgestellt bekommen, man muß die Schüler soweit befähigen, daß sie relativ selbständig anspruchsvolle Programme mit Hilfe des Rechners bewältigen können. Damit soll nochmals betont werden, daß es notwendig ist, einen jeden Schüler mehrfach selbständig ein Programm oder größere Programmteile anfertigen zu lassen.

Eine unabdingbare Voraussetzung für die breite Einführung der Rechentechnik in die Schulpraxis ist es, die Aus- und Weiterbildung an diesen Erfordernissen zu orientieren.

Um die erreichten Ergebnisse des fakultativen Lehrganges »EDV und Kleinstrechner« zu sicherem und anwendungsbereitem Wissen, zu Fähigkeiten und auch Fertigkeiten werden zu lassen, ist es günstig, nach der Behandlung der Rechentechnik und Programmierung im fakultativen Unterricht unbedingt die Möglichkeit für eine weitere Anwendung der Kenntnisse der Schüler zu schaffen. So bietet es sich an, einen anderen fakultativen Lehrgang – z.B. numerische Mathematik – mit Rechnerstützung durchzuführen. Dabei könnten die Schüler nach der theoretischen Behandlung jeweils das entsprechende Programm aufstellen. Damit würde die Wirksamkeit der Rechentechnik, aber auch die Anwendbarkeit numerischer Verfahren demonstriert, die ohne Rechenhilfsmittel nicht oder zumindest nicht sinnvoll anwendbar sind.

Autor:
Dr. Gerhard Hamm
DDR 6000 Suhl
Schmiedfelder Straße 55
PF 98-09

HOME COMPUTER – ein neues Gebiet für den Amateur (Teil 1)



Die stürmische Entwicklung der elektronischen Industrie und das umfangreiche Angebot an Schaltkreisen ergeben auch für den Elektronikamateur ein neues Bewährungsfeld – den *Home computer*.

Der Mikrorechner bringt auch für den Amateur umfangreiche Anwendungsmöglichkeiten mit sich. Das Spektrum reicht dabei von der Steuerung von Spielzeug, wie Modelleisenbahn oder Autorennbahn, über Fernsehspiele, Lösung mathematischer Aufgaben bis zu Datenverwaltungssystemen. Bei letzterem sind beispielsweise Dateisysteme denkbar, die einem die Suche auf Magnetbändern, in Diakästen und anderem erleichtern. Genauso breit wie die Aufgabengebiete ist die Palette der Lösungsvarianten. Ein Homecomputer hat dabei gegenüber dem speziell aufgebauten Mikrorechner den Vorteil, daß er bei entsprechender Konzipierung für das gesamte oben aufgeführte Spektrum einsetzbar ist.

Das in diesem und zwei weiteren Folgen vorgestellte *Konzept für einen Homecomputer* soll dabei als *Anregung zum Selbstbau* gedacht sein. Mit den veröffentlichten Schaltungen wird eine komplette Grundversion vorgestellt. Es wird dabei bewußt auf Leiterplattenzeichnungen verzichtet, wodurch Leiterplattenformate, Steckverbinder und teilweise auch Bauelemente

frei wählbar sind. Es muß noch darauf verwiesen werden, daß die hier vorgestellten Schaltungen nicht für kommerzielle Zwecke verwendet werden dürfen, da keine Prüfung der Schutzrechtslage erfolgte.

Systemkonzept

Das hier beschriebene Rechnersystem ist ein Teil eines größeren Homecomputers, dessen Entwicklung etwa 1980 begonnen hat. Das Grundsystem enthält den Mikroprozessor Ü 880, eine Taktzentrale, den Speicher und drei E/A-Anschlüsse. Damit beschränkt sich die Peripherie auf eine Tastatur, den Bildschirm und einen Magnetbandgeräteanschluß (Bild 1).

Dazu kämen dann Ein-/Ausgabeeinheiten zur Steuerung der speziellen Prozesse. Im erweiterten System könnte dies durch Magnetkartenleser, Lochbandleser (Handleser), Uhr, Lichtstift und Analogeingänge ergänzt werden. Dabei stehen im Vordergrund immer eine Erweiterung des Speichers und die dazu nötigen Verwaltungseinheiten.

Ausgehend von dem seit 1980 existierenden und dem zu erwartenden Bauelementeangebot enthält der Rechner ein Minimum an Spezialschaltkreisen, läßt aber den Einsatz moderner Bauelemente, wie PIO, CTC usw., zu. Dabei wurden

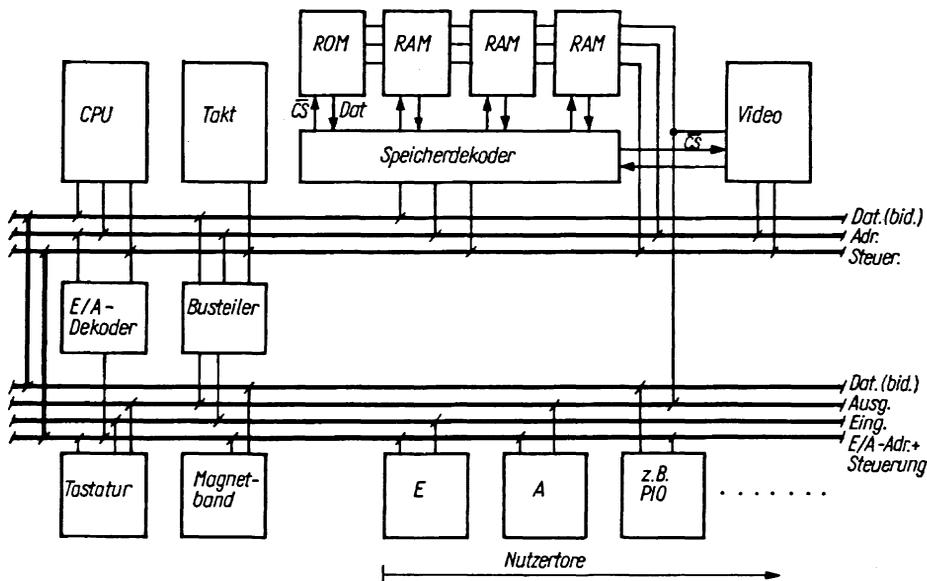


Bild 1. Systemstruktur

in dem System für den Amateur vertretbare Kompromisse eingegangen, wie der Verzicht auf DMA-Betrieb und die individuelle Rückverdrahtung einer jeden Leiterkarte. Vor allem durch letzteres wird in Kauf genommen, daß eine einmal aufgebaute Leiterplatte ihren Steckplatz im System nicht mehr ohne weiteres verändern kann. Dieses Prinzip wurde lediglich bei den Speicherplatten durchbrochen, wodurch jeder Speicherplatz wahlweise durch eine RAM- oder ROM-Karte belegt werden kann. Das geht allerdings zu Lasten einer mangelnden Leiterkartenauslastung bei den ROM's, da diese einen höheren Integrationsgrad besitzen.

Bussystem

Rechner mit dem U 880 verfügen gewöhnlich über einen bidirektionalen Datenbus, das heißt, auf einem Leitungssystem werden sowohl Daten ein- als auch ausgegeben. Sollen Ein-/Ausgabebauplätze ohne die Schalt-

kreise U 855 bis U 857 realisiert werden, so steht dem entgegen, daß der Bus nur begrenzt belastbar ist bei der Ausgabe und für die Eingabe Schaltkreise erfordert, die einen Tristateausgang besitzen, welche zur Zeit nur als Import zur Verfügung stehen. Als Abhilfe wurde im Rechnersystem ein Busteiler eingefügt, der zusätzlich zum bidirektionalen Datenbus zwei getrennte unidirektionale Busse erzeugt (Bild 2). Der damit entstehende Ausgabebus führt ständig alle Daten, die auf dem bidirektionalen Bus liegen. Er kann mit 30 TTL-Lasten beschaltet werden, wodurch auch normale TTL-IC's wie D 195, D 172 oder D 174 zur Datenausgabe verwendet werden können. Zur Eingabe wird ein Offenkollektor-Bus genutzt, der im Fall einer Eingabeanforderung des Rechners auf den bidirektionalen Datenbus geschaltet wird. Damit können Eingabebauplätze mit dem Schaltkreis D 103 aufgebaut werden.

Durch diese Konzeption ist es möglich, den Rechner großteils auch aus den eigenen Beständen aufzubauen und sehr einfache und preiswerte Prozeßanschlüsse zu realisieren. Durch die einfache Form entfallen dann auch die für den Anfänger großen Probleme der Initialisierung der E/A-Tore im Programm.

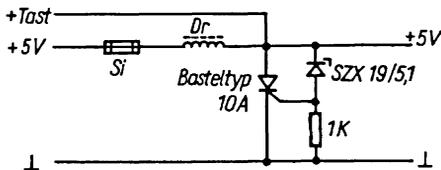


Bild 3. Schutzthyristor für +5 V

Stromversorgung

Der Rechner benötigt ein leistungsstarkes Netzteil. Den größten Stromverbrauch ergeben dabei die Speicherschaltkreise und die TTL-Bausteine. Für den EPROM werden außer den +5 V noch +12 V und -5 V benötigt. Es ergeben sich für das Gerät folgende Spannungen und Stromstärken:

Trafo 1:

- + 5 V/7 A Rechner/Speicher
- + 5 V/7 A E/A-Kanäle

Trafo 2:

- + 12 V/1 A Speicher
- + 12 V/1 A für Prozeß (Lampen, Relais, . . .)
- 5 V/1 A Speicher
- 22 V/0,1 A Tastatur

Die Netzteile sollten alle elektronisch geregelt sein und über eine Strombegrenzung verfügen. Für die Spannung +5 V hat sich die Schaltung nach [1], [2] bewährt. Die anderen Spannungen können über eine einfache Schaltung mit Transistoren stabilisiert werden. Obwohl es umstritten ist, sollte eine Ein- und Ausschaltreihenfolge beachtet werden. Das ist vor allem für die Speicher notwendig, da besonders dynamische Speicherschaltkreise sehr empfindlich darauf reagieren.

Es ist zu sichern, daß die -5 V auf jeden Fall als erste anliegen und als letzte zusammenbrechen. Zeitlich in der Mitte kommen in beiden Richtungen die +12 V. Für das Einschalten wurden 2 Schalter verwendet. Der erste schaltet Trafo 2 zu, wobei die -5 V über ein Relais die +12 V einschalten. Trafo 1 wird über einen Vorwiderstand bei abgeschalteter Last mit angeschaltet. Durch den 2. Schalter, einen Taster, werden Trafo 1 voll zugeschaltet und die +5-Regler freigegeben. Diese Maßnahme ist nötig, um keine Einschaltüberlastung der Sicherung zu verursachen, da der Trafo selbst und die etwa 40 mF Ladeelko einen hohen Einschaltstrom ziehen. Die Ausschaltreihenfolge ergibt sich fast selbständig durch die Lastverhältnisse. Die Spannung +5 V sollte über einen Parallelthyristor geschützt werden (Bild 3). Ein Ausfall des Reglers würde sonst zur Zerstörung des gesamten Rechners führen. Bei den anderen

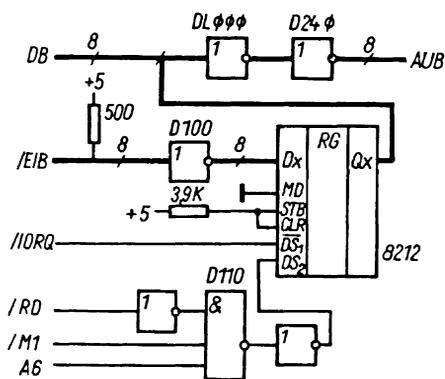


Bild 2. Busteiler

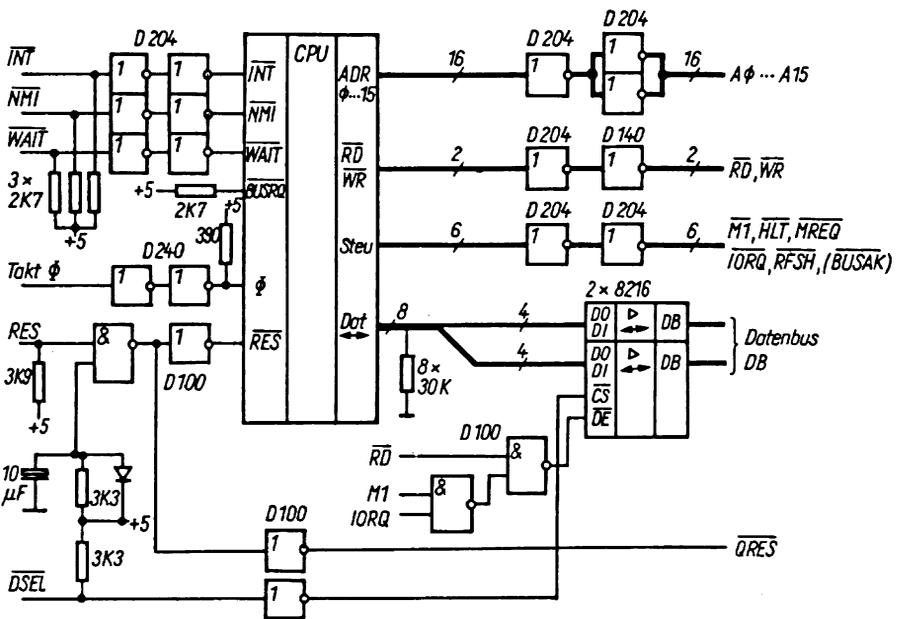


Bild 4. Prozessorkarte

Spannungen sind parallele Z-Dioden mit 1 bis 2 V über dem Spannungswert ausreichend.

Prozessorkarte

Auf der Prozessorkarte (Bild 4) befindet sich der Schaltkreis U 880 mit der nötigen Randlelektronik. Kein Prozessorsignal verläßt die Karte ungepuffert, wodurch Fehler im Gerät nicht auf den Schaltkreis direkt einwirken können. Die Adressen der CPU werden zweimal negiert und bei der zweiten Negation auf 20 Lasteinheiten verstärkt. Die Ausgänge werden mit 1 k Ω -Widerständen nach +5 V abgeschlossen. Das ist nötig, da an den Adrebleitungen kaum statische Lasten (TTL-Gatter) angeschlossen sind, aber durch die Speicherschaltkreise eine große kapazitive Belastung auftritt. Die Steuersignalausgänge sind ebenfalls doppelt negiert. Es müssen aller-

dings nur die Signale RD und WR eine höhere Belastbarkeit aufweisen, da diese für E/A-Anschlüsse oft gebraucht werden. Sie sollten möglichst in negierter und nichtnegierter Form abgreifbar sein. Für alle anderen Signale sind 10 Lasteinheiten ausreichend. Die Steuersignaleingänge sind mit 5 k Ω an +5 V zu legen, wodurch sie nur bei Bedarf beschaltet werden müssen.

Der Reseteingang geht über die Einschaltresetschaltung zum Prozessorbaustein. Die Systembausteine werden dann an den Resetausgang QRES angeschlossen, wodurch sie sowohl bei Handbedienung als auch bei Spannungszuschaltung zurückgesetzt werden. Bei der Dimensionierung ist zu beachten, daß die Zeitkonstante größer ist als die Zeit zum vollständigen Aufbau der Betriebsspannung.

Der Datenbustreiber steht gewöhnlich auf »Schreiben«, hat also zur CPU seine Eingänge. Er wird nur umgekehrt, wenn das Signal »Lesen« aktiv wird oder ein Interruptvektorlesen erfolgt.

Durch das /DSEL-Signal und die Widerstände nach Masse ist es möglich, einen »Einschaltssprung« zu realisieren und damit den Speicherbereich mit RAM zu beginnen. Solange /DSEL aktiv ist, liest der Rechner Leerbefehle und erhöht die Adresse. Er »übersieht« damit den Schreib-Lese-Speicher, der noch keine Befehle enthält. Bei Erreichen einer ausgewählten Adresse wird ein Flipflop gekippt und der Treiber freigegeben. Das FF wird durch Reset wieder rückgekippt. Später ist dann auf die Adresse 0 ein Sprung an den Programmanfang zu setzen.

Taktzentrale

Es hat sich als günstig erwiesen, alle benötigten Takte auf einer zentralen Karte zu erzeugen. Man benötigt dann nur einen Quarz, von dem alle Frequenzen abgeleitet werden. Für das System sind folgende Frequenzen erforderlich:

- 10 MHz Punktfrequenz Bildschirm (64 Zeichen),
- 2,5 MHz Rechner und E/A-Bausteine,
- 1 MHz Zeitbasis Bildschirm,
- 5 MHz Punktfrequenz Bildschirm (32 Zeichen),

Zusätzlich können noch 100 kHz als Normalfrequenz und 1 Hz bzw. 10 Hz für eine Softwareuhr bereitgestellt werden. Auf eine Schaltungswiedergabe soll hier verzichtet werden.

Es muß jedoch darauf verwiesen werden, daß der 2,5-MHz-Takt möglichst leistungsstark sein sollte (D 240), da er für alle Systembausteine verwendet wird. Ein Abschluß mit Widerständen ist zu empfehlen. Dabei sind die Beschriftungsvorschriften für die Systembausteine und den Prozessor zu beachten (390 Ω nach + 5 V).

Busteiler

Wie schon bei der Bussystembeschreibung erläutert, dient die Busteilerbaugruppe der Anpassung einfacher E/A-Baugruppen an das System.

Da der bidirektionale Datenbus durch die Ausgabelinie möglichst wenig belastet werden darf, sollte er mit LS-Gattern beschaltet werden. Durch diese geringe Last können die angeschlossenen Systembausteine ohne Treiber eingesetzt werden. Mit der Beschaltung der Eingabelinie wird die Verteilung der E/A-Kanäle auf Systemkomponenten und einfache E/A festgelegt. Der nur als TS-Treiber genutzte 8212 ist durchgeschaltet, wenn eine Eingabeanforderung im gewünschten Adreßraum (der einfachen E/A), hier 00H bis 07FH, vorliegt. Der verbleibende Adreßraum steht dann dem Anschluß von Systembausteinen an den bidirektionalen Bus zur Verfügung. Der Eingabebus wird als wired-NOR geschaltet. Um die Daten unnegiert zu lesen, erfolgt die Rücknegation für den gesamten E-Bus gemeinsam. Bei einem Abschlußwiderstand von etwa 500 Ω kann der Bus mit 20 OC-Gattern beschaltet werden [3].

Sollten in dem System Busverstärker im bidirektionalen Bus aufgenommen werden, so ist deren Steuerung mit von der Busteilerplatte abzuleiten.

Der Ausgabebus läßt sich durch Gatter mehrfach verstärken. Das gleiche ist beim E-Bus möglich, jedoch muß das dem Rechner nähere Gatter einen OC-Ausgang besitzen.

E/A-Dekoder

Eine weitere zentrale Einheit, vor allem zur Ansteuerung der einfachen E/A-Baugruppen, ist der E/A-Dekoder. Ihm kommt die Aufgabe zu, eine Vordekodierung der E/A-Adressen vorzu-

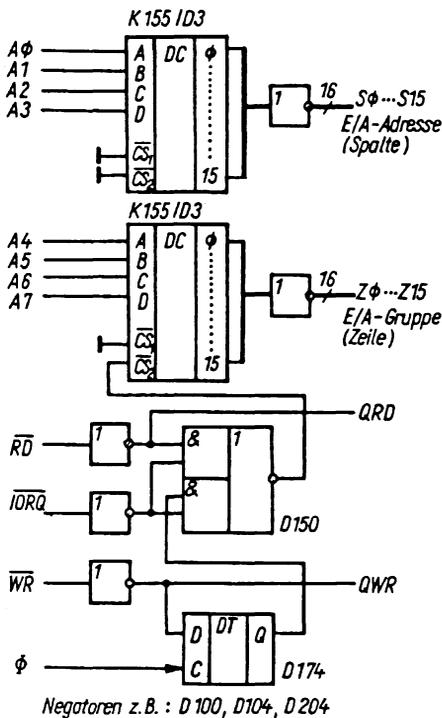
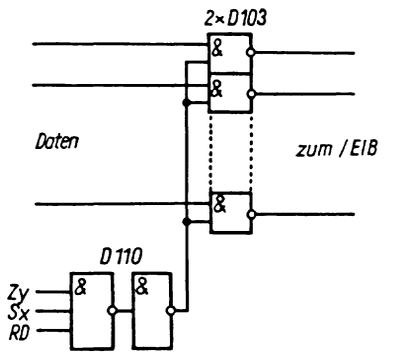


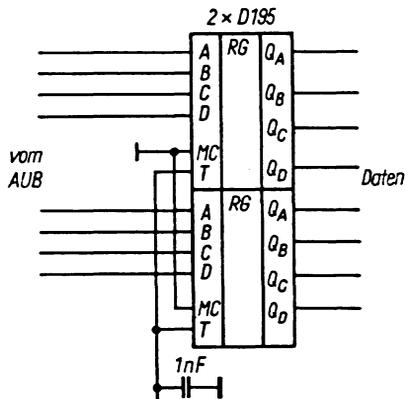
Bild 5. E/A-Dekoder

nehmen und das WR-Signal einen Takt zu verzögern. Dadurch stehen den Datensignalen eineinhalb CPU-Takte zum Einschwingen zur Verfügung, was vor allem bei flankengesteuerten Bausteinen wie D 174 notwendig ist. Der Dekoder stellt zum einen eine »Spaltenadresse« zur Verfügung, die aus den unteren vier Adreßbit gebildet wird, und eine »Zeilenadresse« aus den oberen 4 Bit. Dabei wird die Zeile mit dem Steuersignal IORQ verknüpft. Es ergeben sich damit folgende Ansteuerbeziehungen für die E/A-Baugruppen, wenn im Busteiler nur A7 eingeht (Bild 5):

Ausgabeter $S_x \times Z_y \times WR$
 Eingabeter $S_x \times Z_y \times RD$
 (mit $x = 0 \dots F / y = 0 \dots 7$)



a)



b)

Bild 6. Aufbau einfacher E/A-Anschlüsse
 a) Eingabe, b) Ausgabe

Systembausteine Z_y an /CS
 (mit $y = 8 \dots F$ /Die Adressen 0 und 1 der CPU gehen direkt an den E/A-Baustein)

Damit lassen sich je 128 einfache E/A-Tore und 8 Systembausteine ansteuern. Bei Einbeziehung von A6 in den Busteiler verändert sich das Verhältnis auf 64 zu 12 (Bild 6).

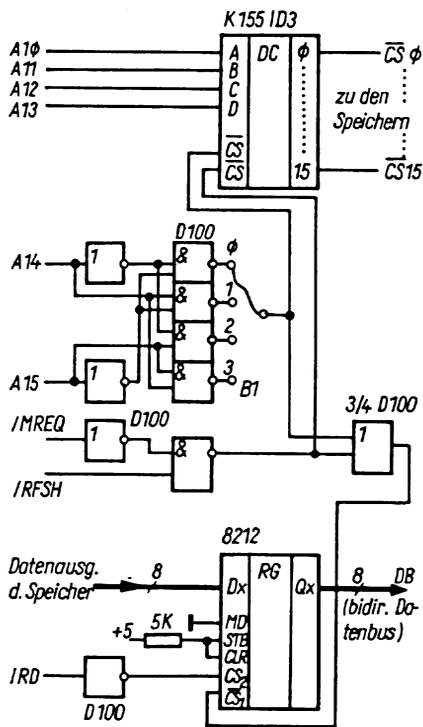


Bild 7. Speicherdekode

Speicherdekode

Um den Aufwand an Dekoderschaltkreisen und Treibern gering zu halten, wurden in dem System Speicherblockdekode eingeführt (Bild 7). Sie enthalten alle für den Speicherblock notwendigen Schaltkreise außer den eigentlichen Speicherchips.

Der Speicherdekode realisiert zentral die Dekodierung für einen Speicher- raum von 16 KByte. Welcher Block der vier möglichen das ist, wird durch die Wickelbrücke B1 festgelegt. Sollten dynamische 4K-Speicher verwendet werden, so ist die Refreshsteuerung auf der Speicherkarte zu realisieren, da der Dekode gesperrt bleibt.

Die gewonnenen Steuersignale werden gleichzeitig zur Freigabe des Puffers

genutzt, der die gelesenen Daten auf den *bidirektionalen* Datenbus des Rechners aufschaltet. Ein CS-Signal wird von einem Speicherblock abgezogen und dient dem Zugriff auf den Bildwiederholungspeicher, der damit Bestandteil des Systemspeichers ist.

Speicher

Als Speicherchips lassen sich prinzipiell alle Typen mit einer Tiefe von 1 KBit verwenden. Auf den Leiterkarten befinden sich nur Speicherschaltkreise. Die Rückverdrahtung ist dabei busartig durchgezogen und wird nur durch die Speicherdekode unterbrochen. Der Bus muß dabei folgende Signale enthalten:

Betriebsspannungen

(Masse, -5 V, +5 V, +12 V)

Adressen A0 ... A11

/WR

Ausgabebus vom Busteiler

Eingabebus zum Speicherdekode

/CS-Signale als Bus oder direkt an die Steckplätze verdrahtet.

Das Speicherschreiben erfolgt vom Ausgabebus (siehe Busteiler), da dieser kapazitiv stärker belastbar ist und die Belastung des bidirektionalen Busses dadurch gesenkt wird. Das /WR-Signal wird direkt an die Schaltkreise geführt. Werden in dem System dynamische 4-K-RAM eingesetzt, so sind die 4 CS-Signale zusammenzuführen. Es werden aber dann aus Laufzeitgründen die Adressen A10 und A11 direkt verwendet. Es ist bei diesen Schaltkreisen zu beachten, daß die CS-Eingänge über Widerstände im inaktiven Zustand gehalten werden, wenn die Spannung +5 V fehlt.

Dies ist erforderlich, da beim Einschalten nicht unter Spannung stehende TTL-Schaltkreise wie TS-Ausgänge reagieren. Bleiben die Steuereingänge

dann im verbotenen Bereich hängen, so kann das zur Zerstörung des Schaltkreises führen.

Bei Einhaltung des Busprinzips im Speicherbereich ist der Austausch von Speicherkarten verschiedener Typen untereinander möglich.

Im aufgebauten System existieren steckerkompatible 4-K-Speicherplatten mit den Schaltkreisen U202, U555, K565RU1 und I2114 bzw. HM6514.

(Fortsetzung in Heft 2)

Literatur

- [1] JUNGnickel, M.: Anwendung integrierter Spannungsregler. – In: radio fernsehen elektronik 27 (1978) H. 2. – S. 85–87, Bild 15
- [2] JUNGnickel, M.: Moderne Stromversorgungstechnik Teil 6. – In: radio fernsehen elektronik 29 (1980) H. 6. – S. 367–370, Bild 60
- [3] SCHMIED, H.; KÜHN, E.: Handbuch Integrierte Schaltkreise. – Berlin: Verlag Technik, 1978. – S. 62

Autor:

Dr.-Ing. Gert Schönfelder

DDR 8029 Dresden

Auf dem Eigen 29

Problemanalytiker im Rechenzentrum
der Sektion Informationsverarbeitung
der Ingenieurhochschule Dresden

Rechentechnische Begriffe für den Laien erklärt

CPU	Central processing unit Zentrale Verarbeitungseinheit eines Rechners
Interface	Anschlußsystem für die Verbindung der Zentraleinheit mit den an den Rechner angeschlossenen peripheren Geräten
BUS	Übertragungsleitung mit mehreren Anschlußstellen
RAM	Random access memory Lese-Schreib-Speicher mit wahlfreiem Zugriff
ROM	read-only memory Nur-Lesespeicher, Festwertspeicher

Quadratische Gleichungen – einmal anders gelöst



In der Schule sind uns zahlreiche Rechenverfahren für die Lösung der unterschiedlichsten Aufgabenstellungen beigebracht worden. Es stellt sich jedoch heraus, daß nicht alle diese Rechenverfahren besonders vorteilhaft anzuwenden sind, wenn man einen elektronischen Taschenrechner benutzt. In vielen Fällen lohnt es sich, darüber nachzudenken, ob man die gestellte Aufgabe nicht auch auf anderem Wege, als man es bisher gewohnt war, lösen kann. Häufig ergeben sich dabei Lösungswege, die mit wesentlich weniger Rechenaufwand verbunden sind, als dies mit den herkömmlichen Rechenverfahren der Fall war.

An einem einfachen *Beispiel* soll dies demonstriert werden.

Jeder weiß, daß man die beiden Lösungen x_1 und x_2 einer quadratischen Gleichung

$$x^2 + px + q = 0 \quad (1)$$

mit Hilfe der beiden Lösungsformeln

$$x_1 = -\frac{p}{2} + \sqrt{\frac{p^2}{4} - q} \quad (2^a)$$

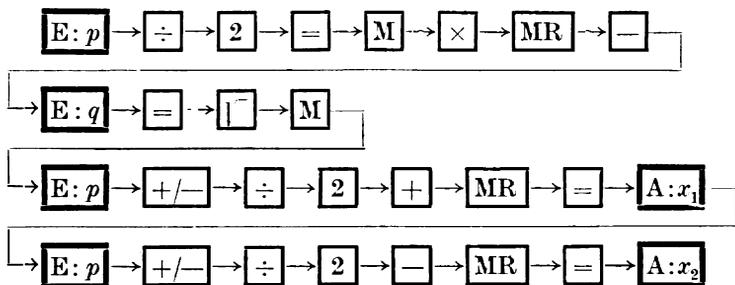
und

$$x_2 = -\frac{p}{2} - \sqrt{\frac{p^2}{4} - q} \quad (2^b)$$

ermitteln kann.

(Es soll vorausgesetzt werden, daß die zu lösende quadratische Gleichung reellwertige Lösungen besitzt.)

Hat man einen Taschenrechner zur Verfügung, der wenigstens einen Speicher, eine Taste $\boxed{+/-}$ für den Vorzeichenwechsel sowie eine Taste $\boxed{X \leftrightarrow Y}$ für den Registeraustausch besitzt, so könnte man die beiden Lösungen der quadratischen Gleichung an Hand des folgenden *Rechenplanes* berechnen:



Dabei wurde in den ersten beiden Zeilen dieses Rechenplanes der Wert der Wurzel aus Formel (2^a) bzw. (2^b) berechnet und gespeichert. In der dritten bzw. vierten Zeile des Rechenplanes werden dann die beiden Lösungen x_1 und x_2 ermittelt.

Die verwendeten Symbole sind hierbei wie folgt zu deuten: Die dünn umrandeten Felder des Rechenplanes besagen, daß die entsprechende Taste des Tastenfeldes zu drücken ist. Hingegen sollen die fett umrandeten Felder des Rechenplanes andeuten, daß eine Zahl über die Zahlentastatur eingegeben bzw. vom Display abgelesen werden soll. So ist bei **E: p** der Zahlenwert der Größe p über die Zahlentastatur einzutippen und bei **A: x₁** der Lösungswert x_1 aus dem Display abzulesen.

Erinnert man sich nun an seine Schulzeit, so kommt einem vielleicht im Zusammenhang mit den quadratischen Gleichungen der Wurzelsatz von VIETA in den Sinn, von dem die meisten Schüler glauben, er wäre nur deshalb in den Schulstoff aufgenommen worden, daß in den Prüfungen danach gefragt werden kann.

Dieser Wurzelsatz von VIETA besagt, daß zwischen den Koeffizienten p und q einer quadratischen Gleichung und deren Lösungen x_1 und x_2 die Zusammenhänge

$$x_1 + x_2 = -p \quad (3)$$

$$x_1 \cdot x_2 = q \quad (4)$$

bestehen.

Beachtet man dies, so lassen sich beispielsweise mit Hilfe der Formel (3) die acht Taschenrechneroperationen aus der vierten Zeile des obigen Rechenplanes auf die folgenden fünf Operationen reduzieren:

$$\rightarrow \boxed{+} \rightarrow \boxed{\text{E: } p} \rightarrow \boxed{=} \rightarrow \boxed{+/-} \rightarrow \boxed{\text{A: } x_2} .$$

Man kann aber auch ganz anders an die Lösung einer quadratischen Gleichung herangehen. Löst man nämlich die Gleichung (1) formal in der folgenden Weise nach x auf:

$$x = -p - \frac{q}{x}$$

und setzt in die rechte Seite für x einen Anfangswert x_1 ein, so erhält man links einen Wert für x , den wir mit x_2 bezeichnen wollen. Dieser Wert x_2 ist unter bestimmten Voraussetzungen, auf die hier nicht näher eingegangen werden soll, über die jedoch beispielsweise in »Mathematik für Ingenieur- und Fachschulen – Band I«, S. 227 ff. des VEB Fachbuchverlag Leipzig nachgelesen werden kann, ein Zahlenwert, der näher an der tatsächlichen Lösung der quadratischen Gleichung liegt, als dies beim Anfangswert x_1 der Fall war. – Wiederholt man nun die entsprechende Rechnung mit dem soeben gefundenen Wert x_2 , so ergibt sich ein noch besserer Näherungswert für die Lösung, der mit x_3 bezeichnet werden soll. Mit der weiteren Fortsetzung dieses Verfahrens kann man die Lösung einer quadratischen Gleichung mit jeder gewünschten Genauigkeit ermitteln. (Die fehlende zweite Lösung könnte, wenn die erste bekannt geworden ist, beispielsweise mit Hilfe des Wurzelsatzes von VIETA leicht berechnet werden.)

Fassen wir diese *Vorgehensweise* zusammen:

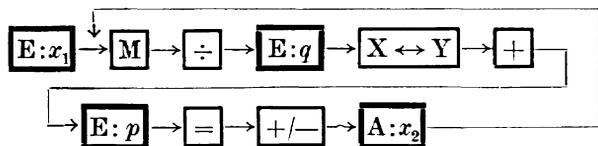
Man geht von einem Näherungswert x_1 für die Lösung der quadratischen Gleichung aus und berechnet mit seiner Hilfe einen Wert x_2 aus

$$x_2 = -\left(\frac{q}{x_1} + p\right). \quad (5)$$

Der gefundene Wert x_2 wird dann erneut als Ausgangswert für eine neue Näherungsrechnung benutzt, und das Verfahren wird solange fortgesetzt, bis

die Lösung der quadratischen Gleichung mit einer gewünschten Genauigkeit ermittelt worden ist (d.h. bis sich die letzten Stellen der Näherungswerte von einer bestimmten Stelle ab nicht mehr ändern.)

Ein entsprechender *Rechenplan* für diese Vorgehensweise könnte wie folgt aussehen:



Beenden des Verfahrens, wenn die gewünschte Genauigkeit des Ergebnisses erreicht ist. Dann Fortsetzung mit



Vergleicht man diesen letzten Rechenplan mit dem, der anfangs für die Berechnung mit Hilfe der Lösungsformel aufgestellt worden ist, so erkennt man, daß die hier erforderlichen Rechnungen wesentlich einfacher zu überblicken sind, als dies vorn der Fall war.

Derartige Rechenverfahren, die – von einem Anfangswert ausgehend – immer besser werdende Näherungswerte für die gesuchte Lösung ermitteln, werden in der Mathematik *Iterationsverfahren* genannt. Sie sind für die Anwendung von Taschenrechnern meist sehr geeignet. Da ein und derselbe Rechengang mehrfach wiederholt werden muß, sind natürlich programmierbare Taschenrechner besonders gut für derartige Rechnungen geeignet. Bei der Anwendung programmierbarer Taschenrechner müßte der Rechenplan dann nur noch durch eine Bedingung erweitert werden, die angibt, wann die erforderlichen Rechnungen abgebrochen werden können, weil die zu erreichende Genauigkeit des Ergebnisses erreicht worden ist.

Anmerkung 1

Das hier vorgestellte Iterationsverfahren ist nicht uneingeschränkt anwendbar. Es darf nur dann verwendet werden, wenn für die zuerst zu berechnende Lösung x_1 der quadratischen Gleichung (1) gilt

$$|x_1| > \sqrt{|q|}.$$

Anmerkung 2

Ist die in Anmerkung 1 genannte Bedingung nicht erfüllt, so kann die Lösung der quadratischen Gleichung (1) mit einer der beiden folgenden Iterationsformeln in Angriff genommen werden:

$$x_2 = -\frac{x_1^2 + q}{p}, \text{ gültig für } |x_1| < \frac{|p|}{2}$$

oder mit

$$x_2 = \sqrt{-(p x_1 + q)}, \text{ gültig für } x_1 < -\frac{p}{q}.$$

Der Leser versuche, die zugehörigen Rechenpläne für die Anwendung dieser Iterationsformeln selbständig aufzustellen.

Anmerkung 3

Das hier für quadratische Gleichungen vorgestellte Iterationsverfahren läßt sich verallgemeinern für beliebige Gleichungen mit einer Unbekannten.

Autor:

Prof. Dr.-Ing. Hans Kreul

DDR 8800 Zittau

Bruno-Schröter-Straße 1

a. o. Professor an der Ingenieurhochschule Zittau

Abteilung EDV und Rechentechnik

Das MASTER-MIND-SPIEL – der K 1003 als Herausforderer



1. Zum Spielverlauf

Noch vor der von Prof. ERNÖ RUBIK aus Ungarn ausgelösten »Zauberwürfelepidemie« machte ein Logikspiel von sich reden, das unter den Begriffen *Superhirn* oder *Master Mind* (deutsch: Meister des Verstandes oder Meister des Gedächtnisses) bekannt wurde. In diesem Spiel kämpft ein Entschlüsseler gegen einen Herausforderer. Der Herausforderer ordnet, für den Entschlüsseler nicht sichtbar, eine Anzahl farbiger Stecker an bestimmten Positionen an. Der Entschlüsseler hat die Aufgabe, schrittweise die richtigen Farben **und** die richtigen Positionen zu ermitteln. Nach jedem Schritt (Lösungsversuch) des Entschlüsselers gibt der Herausforderer folgende zwei Antworten:

1. Anzahl der richtigen Farben an richtiger Position; 2. Anzahl der richtigen Farben an falscher Position.

Aus diesen Informationen stellt der Entschlüsseler den nächsten Schritt zusammen. Der große Reiz des Spiels besteht darin, *aus den zu jedem Schritt (Lösungsversuch) gegebenen Informationen mit einer möglichst geringen Anzahl von Schritten die richtigen Farben an den richtigen Positionen herauszufinden.*

Ein solches Master-Mind-Spiel gibt es z.B. in der VR Polen. Es ist ein 4-

aus-6-Spiel, d.h., es müssen 4 aus 6 möglichen Farben und deren richtige Position herausgefunden werden. Man kann das Spiel nach vorheriger Absprache komplizierter machen, indem man es zuläßt, daß Farben mehrfach auftreten können, und man durch Freilassen eines Loches eine 7. Kodefarbe zuläßt. Diese Sonderfälle sollen im folgenden nicht betrachtet werden. Bild 1 zeigt auf dem Spielbrett vorn die für den Entschlüsseler nicht sichtbare Kombination, die erraten werden soll. Nach dem ersten Lösungsversuch des Entschlüsselers (1. Reihe mit vier farbigen Steckern) antwortet der Herausforderer auf dem mit 4 Löchern im Quadrat angeordneten Antwortfeld

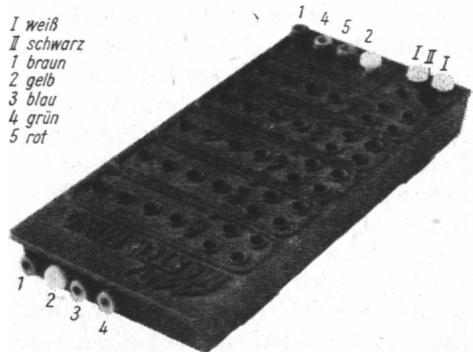


Bild 1. Master Mind

folgendermaßen: Schwarzer Stecker = Farbe und Position sind richtig; weißer Stecker = Farbe ist richtig, aber Position ist falsch; kein Stecker = Farbe ist falsch.

Mit dem Aufkommen programmierbarer Tisch- und Taschenrechner lag der Gedanke nahe, die Funktion des Herausforderers dem Rechner zu übertragen. In Ermangelung von Bildschirm mit Farbgrafik bietet sich die einfache Möglichkeit, die Farben durch Ziffern zu ersetzen. Diese zu erhaltende Ziffernkombination kann mit einem Zufallszahlengenerator relativ leicht erzeugt werden. Im Rechenprogramm müssen dann noch die Antwortmöglichkeiten auf die Eingaben des Entschlüssellers realisiert werden, und fertig ist der »elektronische Herausforderer«. In der Literatur gibt es nach diesem Prinzip eine Reihe von Programmen für unterschiedliche Rechner und mit unterschiedlichen Schwierigkeitsgraden. Meistens begnügt man sich auch nicht mit einem 4-aus-6-Spiel, sondern programmiert kompliziertere Varianten wie 4 aus 9, 5 aus 9, 4 aus 10 oder 5 aus 10 mit und ohne Doppelungen von Ziffern (Farben).

2. Zur Realisierung des K-1003-Programms

Für unser K-1003-Rechenprogramm haben wir die anspruchsvolle 5-aus-10-Variante gewählt. Allerdings kommt im vorliegenden Programm *jede Ziffer (Farbe) nur genau einmal* vor. Der Vorzug des im folgenden beschriebenen Programms besteht in der konsequenten Nutzung des Druckers des K 1003 zu Protokollzwecken. Damit wird der gesamte Ablauf übersichtlich, und beim Entschlüsseler erübrigt sich ein Mitschreiben der einzelnen Schritte (Lösungsversuche). Natürlich kostet dieser

Protokollservice Speicherplatz, so daß für das Programm fast 1 KByte benötigt werden.

Der Programmablaufplan in Bild 2 soll den Programmaufbau verdeutlichen. Er untergliedert sich in zwei Teile. Im ersten Teil (Bild 2a) wird das Spiel vom Rechner vorbereitet, indem ein Zufallszahlengenerator 5 Zahlen (Farben) aus 10 möglichen auswählt und in die Datenregister 001 bis 005 bringt. Dieser Zufallszahlengenerator wurde als Unterprogramm realisiert und aus Rechenzeitgründen an den Anfang des Programmspeicherbereiches plaziert. Nach der Erzeugung einer jeden Zahl wird verglichen, ob die gleiche Zahl bereits ausgewählt wurde. Wenn ja, dann muß der Vorgang wiederholt werden, da nach Vereinbarung keine Zahlen (Farben) mehrfach auftreten dürfen. Möglich sind die Zahlen von 1 bis 10. Ist dieser Vorgang beendet, dann erfolgt der Ausdruck:

EIN: SUCHFARBEN
AUS: GOLD/SILBER

1. 2. 3. 4. 5.

Damit wird zur Eingabe des ersten Schrittes (Lösungsvorschlages) für die Positionen 1 bis 5 aufgefordert. Die Antworten GOLD/SILBER haben folgende Bedeutung: GOLD = Farbe und Position richtig; SILBER = Farbe richtig, aber Position falsch.

Im zweiten Programmteil (Bild 2b) versucht der Entschlüsseler, den »Ziffernkodex (Farbenkodex) zu knacken«. Zunächst wird aber im Programm der Schrittzähler (Datenregister 011) inkrementiert, und die Datenregister 012, 013 und 014, die die Anzahl für GOLD und SILBER für den Druck und für die Anzeige bereithalten, werden gelöscht. Der Entschlüsseler gibt nun seinen Lösungsversuch, also 5 Zahlen von 1 bis 10, in den Rechner ein, wobei nach jeder Zahl ST zu drücken

ist. Diese 5 Zahlen werden in die Datenregister 006 bis 010 transportiert und über den Drucker ausgedruckt. Danach findet der Vergleich der Inhalte der Datenregister 001 bis 005 (vom Rechner erzeugter »Geheimkode«) mit dem ersten Lösungsversuch (Datenregister 006 bis 010) statt. Der Protokollausdruck kann dann z.B. so aussehen:

2 7 4 6 1
1 GOLD 3 SILBER.

Der Entschlüsseler hat also im 1. Schritt (Lösungsversuch) die Zahlen 2 7 4 6 1 eingegeben. Davon sind 3 Zahlen (Farben) richtig, aber an falscher Position, während 1 Zahl (Farbe) richtig und zugleich an richtiger Position ist. Im Programm wird dazu folgendermaßen vorgegangen: Stimmen z.B. die erste eingegebene Zahl (in 006) und die erste, zufällig vom Rechner erzeugte Zahl (in 001) überein, so wird das Unterprogramm »GOLD« aufgerufen. Es beinhaltet lediglich den Befehl, die Datenregister 012 und 014 zu inkrementieren. Befindet sich aber die erste eingegebene Zahl des Entschlüssellers in einem der Datenregister 002 bis 005, so stimmt zwar die Farbe, aber nicht die Position. In diesem Fall wird das Unterprogramm »Silber« aufgerufen. Hierzu wird das Datenregister 013 inkrementiert und zum Inhalt des Datenregisters 014 wird 0,1 addiert. Die Datenregister 012 und 013 bedienen den Drucker, während das Datenregister 014 zusätzlich das Anzeigefeld des Rechners bedient. So steht z.B. die Zahl 1, 3 im Anzeigefeld für 1mal Gold und 3mal Silber.

Wenn sich die erste vom Entschlüsseler eingegebene Zahl (Datenregister 006) in keinem der Datenregister 001 bis 005 befindet, dann war diese Zahl falsch.

Zunächst werden im Programm alle

fünf eingegebenen Zahlen auf »Gold« abgefragt. Wird das Ergebnis aller 5 Vergleiche mit »ja« beantwortet, dann wurde das Unterprogramm »Gold« fünfmal aufgerufen, und das Spiel endet mit folgendem Druckbefehl:

GRATULIERE
SIE SIND EIN PFIFFIKUS!

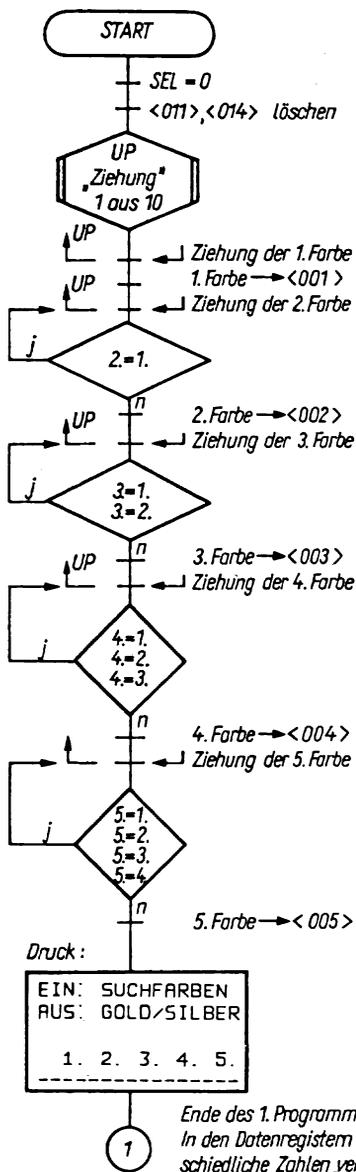
Der Rechner springt dann zum Programmumfang und erzeugt einen neuen »Geheimkode«.

Die zunächst vollständige »Goldabfrage« hat folgenden Vorteil: Tritt wirklich fünfmal »Gold« auf, dann können die Abfragen nach »Silber« unterbleiben. Das spart Rechenzeit. Immerhin finden im Programm 25 Vergleiche mit je 10 Befehlen statt, nämlich fünfmal die Frage nach »Gold« und zwanzigmal die Frage nach »Silber«. Verläuft die »Goldabfrage« nicht erfolgreich, so sind noch nicht alle Zahlen auf der richtigen Position gefunden worden, und die zwanzig »Silberabfragen« müssen durchgeführt werden. Nachdem alle Abfragen stattgefunden haben, ist der 1. Schritt (Lösungsversuch) beendet, und der schon erwähnte Ausdruck erfolgt.

Vor dem folgenden Schritt wird der Schrittzähler (Datenregister 011) abgefragt. Gilt $\langle 011 \rangle = 6$, es wurden also schon 6 Schritte durchgeführt, ohne die richtige Lösung zu finden, so erfolgt der Ausdruck:

NOCH EIN SCHRITT ZUM SIEG,
SONST SPIELVERLUST

Diese Festlegung erscheint dem Anfänger zunächst vermessen und schwierig realisierbar. Die Erfahrungen einer Vielzahl durchgeführter Spiele zeigen aber, daß im 7. Schritt alle Kombinationen gefunden werden können (Hinweise zur Spielstrategie erfolgen im nächsten Abschnitt).



<000> = Zufallszahl 0,xxxxxx
(Eingabe nur vor 1. Spiel)

<001> = 1. Farbe (1 aus 10)

<002> = 2. Farbe (1 aus 10)

<003> = 3. Farbe (1 aus 10)

<004> = 4. Farbe (1 aus 10)

<005> = 5. Farbe (1 aus 10)

<006> = im 1. Teil = Hilfsregister
im 2. Teil = 1. Suchfarbe (I.)

<007> = 2. Suchfarbe (II.)

<008> = 3. Suchfarbe (III.)

<009> = 4. Suchfarbe (IV.)

<010> = 5. Suchfarbe (V.)

<011> = Schrittzähler

<012> = n Gold

<013> = n Silber

<014> = n Gold, n Silber

Ende des 1. Programmteils, das Spiel ist vorbereitet.
In den Datenregistern <001>...<005> sind 5 unterschiedliche Zahlen versteckt*, die im 2. Programmteil vom Spieler in der richtigen Reihenfolge zu suchen sind.

Bild 2a. Programmablaufplan

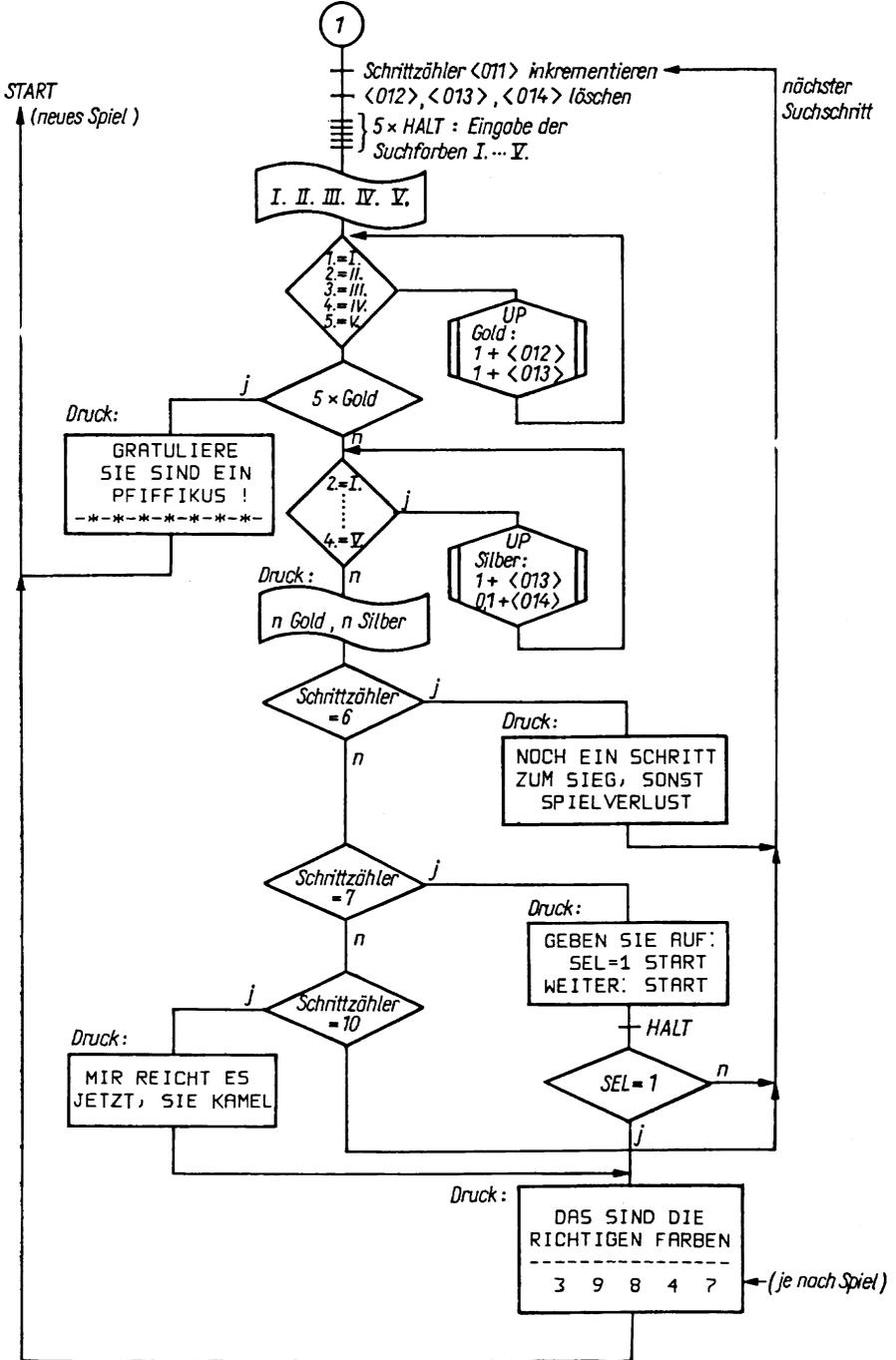


Bild 2b

Wurde nach dem 7. Schritt die Lösung immer noch nicht gefunden, dann gilt das Spiel als verloren (siehe PAP in Bild 2b). Nach Drücken der Tasten **SEL = 1** **ST** wird die vom Rechner erzeugte Zahlenkombination ausgedruckt und damit das »Geheimnis gelüftet«.

Wird das Programm aber mit **ST** fortgesetzt, so kann noch bis zum 10. Schritt versucht werden, die Zahlenkombination zu finden (siehe PAP in Bild 2b).

Das Programm umfaßt 987 Befehle und ist auf S. 51 ff. angegeben. Der Programmstart geschieht folgendermaßen:

1. **D/P** 015 MKL
2. Beim Start des ersten Spiels ist eine Zufallszahl 0,XXXXX (X = Ziffern 0...9) in das Datenregister 000 zu transportieren. Für jedes weitere Spiel ist dies nicht mehr erforderlich.
3. Programmstart mit **STM** **NEUGR**

Das Programm arbeitet mit dem Funktionsblock Mathematik.

3. Zur erfolgreichen Spielstrategie

Der erste Schritt des Entschlüssellers ist völlig beliebig. Beim zweiten Schritt ist dies aber schon nicht mehr der Fall, da die Antworten des Herausforderers (K-1003-Programm) beachtet werden müssen. Beim vorliegenden 5-aus-10-Spiel ergeben sich nach dem ersten Schritt meist 2 oder 3 richtige Zahlen. Ziel ist es nun, mit jedem Suchschritt die Zahl der Kombination so einzunengen, daß im 5., 6. oder 7. Schritt nur noch die richtige Kombination übrig bleibt.

Dazu folgende Grundregel: In jedem Schritt muß die neue Kombination so betrachtet werden, als wäre sie bereits die richtige Lösung, also 5 richtige Zahlen an der richtigen Position. Dann

müssen alle vorherigen Schritte auf die richtige Antwort des Herausforderers gegenüber der neuen Kombination überprüft werden. Nur wenn in **jedem** Schritt die Antwort mit der neuen Kombination übereinstimmt (in bezug auf Anzahl der richtigen Zahlen **und** richtigen Positionen), ist die neue Kombination eine mögliche Lösung. Wenn auch nur in einem einzigen Schritt in nur einer Position ein Widerspruch zwischen der Antwort und der neuen Kombination auftritt, dann ist die neue Kombination keine mögliche Lösung. Dieser Schritt darf nicht eingegeben werden, sondern es muß nach anderen Kombinationen gesucht werden, die nicht zu Widersprüchen führen. Ein Beispiel soll die Interpretation der Antworten des Herausforderers (K-1003-Programm) und die daraus abzuleitenden logischen Schlüsse zeigen.

1. Schritt

Eingabe: 2 5 8 9 4
Antwort: 0 GOLD 3 SILBER

Überlegungen zum nächsten Schritt:

- Von den Zahlen aus dem 1. Schritt müssen drei im nächsten Schritt weiter verwendet werden. Versuch mit 4, 5, 9.
- Diese Zahlen dürfen nicht wieder auf die gleiche Position gesetzt werden.
- Von den noch nicht benutzten Zahlen 1, 3, 6, 7, 10 müssen zwei beliebige ins Spiel gebracht werden. Versuch mit 6 und 7.

2. Schritt

Eingabe: 5 7 9 4 6
Antwort: 1 GOLD 1 SILBER

- Anzahl der richtigen Zahlen ist geringer als im 1. Schritt, also wurden nicht die drei richtigen Zahlen aus dem 1. Schritt erfaßt.
- Von den übernommenen Zahlen 4, 5, 9 aus dem 1. Schritt sind maximal

- zwei richtig, also **muß** von den Zahlen 2 und 8 mindestens eine richtig sein und wieder ins Spiel gebracht werden.
- Von den Zahlen 4, 5, 9 aus dem 1. Schritt **muß** mindestens eine richtig sein, deshalb kann von den Zahlen 6, 7 höchstens eine richtig sein.
 - Wenn 6 oder 7 richtig ist, **müssen** sogar beide nicht gespielte Zahlen 2 und 8 aus dem 1. Schritt richtig sein.
 - Aus dem 1. Schritt müssen drei Zahlen wiederverwendet werden, eine davon muß die Zahl 2 oder 8 sein (eventuell auch beide).
 - Aus dem 2. Schritt dürfen nur zwei Zahlen wiederverwendet werden.
 - Wenn von den Zahlen 2 und 8 nur eine benutzt wird, dann dürfen die Zahlen 6 und 7 nicht verwendet werden, damit also zwei Zahlen von 1, 3 oder 10.
 - Von den zwei Zahlen aus dem 2. Schritt muß eine auf die gleiche Position gesetzt werden (1 GOLD), und alle anderen müssen die Position wechseln.

3. Schritt

Eingabe: 8 3 5 4 1
 Antwort: 0 GOLD 1 SILBER

- Die Zahl 4 hat nicht das »Gold« im 2. Schritt bewirkt.
- Aus dem 2. Schritt ist bekannt, daß 2 oder 8 oder beide richtig sind.
- Wenn 8 richtig wäre und 2 falsch, müßten die Zahlen 3, 4, 5, 1 falsch sein. Dann würde man aber für den 4. Schritt mit den Zahlen 8 und 9 nur zwei Zahlen aus dem 1. Schritt finden. Aber diese Annahme ist falsch.
- Wenn 2 richtig und 8 falsch wäre, müßten 1, 3, (4 oder 5), 6, 7, 8 falsch sein. Da es aber nur fünf falsche Zahlen geben kann, **muß 2 und 8** richtig sein.
- Wenn 8 richtig ist, sind 3, 5, 4, 1 falsch.

- Wenn 2 **und** 8 richtig sind und 4 und 5 falsch, dann sind die drei richtigen Zahlen aus dem 1. Schritt die Zahlen 2, 8, 9.
- Wenn 9 richtig ist und 4, 5 falsch, muß 6 oder 7 (siehe 2. Schritt) richtig sein.
- Richtig sind demnach 2, 8, 9 und (6 oder 7). Da 1, 3, 4, 5, (6 oder 7) falsch sind, ist die noch nicht benutzte Zahl 10 die fehlende richtige Zahl.
- Klärung, ob 6 oder 7 richtig ist. Versuch: 6 ist richtig.
- Da im 2. Schritt 1 GOLD auftrat, muß die 9 oder die 6 auf die gleiche Position wie im 2. Schritt gebracht werden. Versuch: 9 auf gleiche Position wie im 2. Schritt, da noch nicht sicher ist, ob 6 oder 7 richtig ist.
- Alle anderen Zahlen dürfen nicht mehr auf die gleiche Position, also 2 nicht mehr an erste Position, 6 nicht mehr an fünfte Position, 8 nicht mehr an erste und dritte Position. 10 darf jede Position annehmen und ist mit Sicherheit eine richtige Zahl.

4. Schritt

Eingabe: 10 8 9 6 2
 Antwort: 0 GOLD 4 SILBER

- 6 ist falsch, und 7 ist richtig.
- Da die Zahl 9 auf der dritten Position nicht das »Gold« erbracht hat, **muß** es im 2. Schritt die Zahl 7 gewesen sein.
- Alle 5 Farben sind nun bekannt, und die Zahl 7 muß auf die zweite Position.
- Die Zahl 2 kann nur noch auf die dritte oder vierte Position. Versuch: 2 auf vierte Position.
- Die Zahl 8 kann somit nur noch auf die fünfte Position.
- Die Zahl 10 kann nur noch auf die dritte Position, und für 9 bleibt nur noch die erste Position übrig.

0000	MRK	0040	KUL	0080	MRK	0120	UP
0001	KUL	0041	TRX	0081	3	0121	KUL
0002	TRX	0042	1	0082	STM	0122	TRX
0003	MOD	0043	SUB	0083	UP	0123	1
0004	RI	0044	=0	0084	KUL	0124	SUB
0005	ADD	0045	STM	0085	TRX	0125	=0
0006	ETX	0046	1	0086	1	0126	STM
0007	KNO	0047	ST	0087	SUB	0127	4
0008	INT	0048	ST	0088	=0	0128	ST
0009	SUB	0049	TRX	0089	STM	0129	ST
0010	TXR	0050	6	0090	3	0130	TRX
0011	MOD	0051	TXR	0091	ST	0131	6
0012	1	0052	2	0092	ST	0132	TRX
0013	0	0053	MRK	0093	TRX	0133	2
0014	MUL	0054	2	0094	6	0134	SUB
0015	INT	0055	STM	0095	TRX	0135	=0
0016	1	0056	UP	0096	2	0136	STM
0017	ADD	0057	KUL	0097	SUB	0137	4
0018	TXR	0058	TRX	0098	=0	0138	ST
0019	6	0059	1	0099	STM	0139	ST
0020	UP	0060	SUB	0100	3	0140	TRX
0021	MRK	0061	=0	0101	ST	0141	6
0022	NGR	0062	STM	0102	ST	0142	TRX
0023	S=0	0063	2	0103	TRX	0143	3
0024	GL	0064	ST	0104	6	0144	SUB
0025	TXR	0065	ST	0105	TRX	0145	=0
0026	1	0066	TRX	0106	3	0146	STM
0027	1	0067	6	0107	SUB	0147	4
0028	TXR	0068	TRX	0108	=0	0148	ST
0029	1	0069	2	0109	STM	0149	ST
0030	4	0070	SUB	0110	3	0150	TRX
0031	STM	0071	=0	0111	ST	0151	6
0032	UP	0072	STM	0112	ST	0152	TRX
0033	KUL	0073	2	0113	TRX	0153	4
0034	TXR	0074	ST	0114	6	0154	SUB
0035	1	0075	ST	0115	TXR	0155	=0
0036	MRK	0076	TRX	0116	4	0156	STM
0037	1	0077	6	0117	MRK	0157	4
0038	STM	0078	TRX	0118	4	0158	ST
0039	UP	0079	3	0119	STM	0159	ST

0160	TRX	0200		0240	1	0280	2
0161	6	0201		0241	TRX	0281	DRU
0162	TRX	0202	1	0242	1	0282	U
0163	5	0203	.	0243	4	0283	7
0164	MOD	0204		0244	KNO	0284	DRU
0165	LOE	0205	2	0245	LOE	0285	2
0166	TEX	0206	.	0246	TRX	0286	DRU
0167	E	0207	.	0247	1	0287	U
0168	I	0208	3	0248	4	0288	8
0169	N	0209	.	0249	TRX	0289	DRU
0170	:	0210		0250	1	0290	2
0171		0211	4	0251	2	0291	DRU
0172	S	0212	.	0252	TRX	0292	U
0173	U	0213	.	0253	1	0293	9
0174	C	0214	5	0254	3	0294	DRU
0175	H	0215	.	0255	KNU	0295	2
0176	F	0216	-	0256	NUM	0296	DRU
0177	A	0217	-	0257	STP	0297	U
0178	R	0218	-	0258	TRX	0298	1
0179	B	0219	-	0259	6	0299	0
0180	E	0220	-	0260	KOM	0300	DRU
0181	N	0221	-	0261	0	0301	2
0182	ZS	0222	-	0262	STP	0302	TEX
0183	A	0223	-	0263	TRX	0303	TRX
0184	U	0224	-	0264	7	0304	6
0185	S	0225	-	0265	STP	0305	TRX
0186	:	0226	-	0266	TRX	0306	1
0187		0227	-	0267	8	0307	SUB
0188	G	0228	-	0268	STP	0308	=0
0189	O	0229	-	0269	TRX	0309	UP
0190	L	0230	-	0270	9	0310	0
0191	D	0231	-	0271	STP	0311	6
0192	/	0232	ZS	0272	TRX	0312	5
0193	S	0233	TEX	0273	1	0313	8
0194	I	0234	1	0274	0	0314	TRX
0195	L	0235	TRX	0275	PAU	0315	7
0196	B	0236	ADD	0276	TRX	0316	TRX
0197	E	0237	1	0277	6	0317	2
0198	R	0238	1	0278	TEX	0318	SUB
0199	ZS	0239	KOM	0279	DRU	0319	=0

0320	UP	0360	TRX	0400	6	0440	=0
0321	0	0361	1	0401	8	0441	UP
0322	6	0362	2	0402	TRX	0442	0
0323	5	0363	SUB	0403	6	0443	6
0324	8	0364	=0	0404	TRX	0444	6
0325	TRX	0365	0	0405	5	0445	8
0326	8	0366	6	0406	SUB	0446	TRX
0327	TRX	0367	8	0407	=0	0447	7
0328	3	0368	1	0408	UP	0448	TRX
0329	SUB	0369	TRX	0409	0	0449	5
0330	=0	0370	6	0410	6	0450	SUB
0331	UP	0371	TRX	0411	6	0451	=0
0332	0	0372	2	0412	8	0452	UP
0333	6	0373	SUB	0413	TRX	0453	0
0334	5	0374	=0	0414	7	0454	6
0335	8	0375	UP	0415	TRX	0455	6
0336	TRX	0376	0	0416	1	0456	8
0337	9	0377	6	0417	SUB	0457	TRX
0338	TRX	0378	6	0418	=0	0458	8
0339	4	0379	8	0419	UP	0459	TRX
0340	SUB	0380	TRX	0420	0	0460	1
0341	=0	0381	6	0421	6	0461	SUB
0342	UP	0382	TRX	0422	6	0462	=0
0343	0	0383	3	0423	8	0463	UP
0344	6	0384	SUB	0424	TRX	0464	0
0345	5	0385	=0	0425	7	0465	6
0346	8	0386	UP	0426	TRX	0466	6
0347	TRX	0387	0	0427	3	0467	8
0348	1	0388	6	0428	SUB	0468	TRX
0349	0	0389	6	0429	=0	0469	0
0350	TRX	0390	8	0430	UP	0470	TRX
0351	5	0391	TRX	0431	0	0471	2
0352	SUB	0392	6	0432	6	0472	SUB
0353	=0	0393	TRX	0433	6	0473	=0
0354	UP	0394	4	0434	8	0474	UP
0355	0	0395	SUB	0435	TRX	0475	0
0356	6	0396	=0	0436	7	0476	6
0357	5	0397	UP	0437	TRX	0477	6
0358	8	0398	0	0438	4	0478	8
0359	5	0399	6	0439	SUB	0479	TRX

0480	8	0520	6	0560	TRX	0600	DRU
0481	TRX	0521	6	0561	2	0601	1
0482	4	0522	8	0562	SUB	0602	
0483	SUB	0523	TRX	0563	=0	0603	G
0484	=0	0524	9	0564	UP	0604	0
0485	UP	0525	TRX	0565	0	0605	L
0486	0	0526	3	0566	6	0606	D
0487	6	0527	SUB	0567	6	0607	
0488	6	0528	=0	0568	8	0608	DRU
0489	8	0529	UP	0569	TRX	0609	X
0490	TRX	0530	0	0570	1	0610	1
0491	8	0531	6	0571	0	0611	
0492	TRX	0532	6	0572	TRX	0612	5
0493	5	0533	8	0573	3	0613	I
0494	SUB	0534	TRX	0574	SUB	0614	L
0495	=0	0535	9	0575	=0	0615	B
0496	UP	0536	TRX	0576	UP	0616	
0497	0	0537	5	0577	0	0617	TEX
0498	6	0538	SUB	0578	6	0618	ZS
0499	6	0539	=0	0579	6	0619	TRY
0500	8	0540	UP	0580	8	0620	1
0501	TRX	0541	0	0581	TRX	0621	1
0502	9	0542	6	0582	1	0622	MOD
0503	TRX	0543	6	0583	0	0623	6
0504	1	0544	8	0584	TRX	0624	SUB
0505	SUB	0545	TRX	0585	4	0625	=0
0506	=0	0546	1	0586	SUB	0626	0
0507	UP	0547	0	0587	=0	0627	7
0508	0	0548	TRX	0588	UP	0628	4
0509	6	0549	1	0589	0	0629	8
0510	6	0550	SUB	0590	6	0630	TRX
0511	8	0551	=0	0591	6	0631	1
0512	TRX	0552	UP	0592	8	0632	1
0513	9	0553	0	0593	TRX	0633	MOD
0514	TRX	0554	6	0594	1	0634	7
0515	2	0555	6	0595	3	0635	SUB
0516	SUB	0556	8	0596	TRX	0636	=0
0517	=0	0557	TRX	0597	1	0637	0
0518	UP	0558	1	0598	2	0638	8
0519	0	0559	0	0599	TEX	0639	0

0640	2	0680	UP	0720	I	0760	C
0641	TRX	0681	ZS	0721	K	0761	H
0642	1	0682	TEX	0722	U	0762	R
0643	1	0683		0723	S	0763	I
0644	MOD	0684		0724		0764	T
0645	1	0685		0725	!	0765	T
0646	0	0686	G	0726	ZS	0766	Z
0647	SUB	0687	R	0727	-	0767	U
0648	=0	0688	A	0728	*	0768	M
0649	0	0689	T	0729	-	0769	
0650	9	0690	U	0730	*	0770	S
0651	4	0691	L	0731	-	0771	I
0652	6	0692	I	0732	*	0772	E
0653	SPR	0693	E	0733	-	0773	G
0654	0	0694	R	0734	*	0774	,
0655	2	0695	E	0735	-	0775	
0656	3	0696	ZS	0736	*	0776	S
0657	4	0697		0737	-	0777	Q
0658	1	0698		0738	*	0778	N
0659	TXR	0699	S	0739	-	0779	S
0660	ADD	0700	I	0740	*	0780	T
0661	1	0701	E	0741	-	0781	ZS
0662	2	0702		0742	TEX	0782	
0663	TXR	0703	S	0743	ZS	0783	
0664	ADD	0704	I	0744	ZS	0784	S
0665	1	0705	N	0745	ZS	0785	P
0666	4	0706	D	0746	STM	0786	I
0667	UP	0707		0747	NGR	0787	E
0668	1	0708	E	0748	ZS	0788	L
0669	TXR	0709	I	0749	TEX	0789	V
0670	ADD	0710	N	0750	N	0790	E
0671	1	0711	ZS	0751	O	0791	R
0672	3	0712		0752	C	0792	L
0673	MOD	0713		0753	H	0793	U
0674	DP	0714		0754		0794	S
0675	1	0715	P	0755	E	0795	T
0676	TXR	0716	F	0756	I	0796	TEX
0677	ADD	0717	I	0757	N	0797	ZS
0678	1	0718	F	0758		0798	SPR
0679	4	0719	F	0759	S	0799	2

0800	3	0840	E	0880	D	0920	DRU
0801	4	0841	R	0881	I	0921	2
0802	ZS	0842	:	0882	E	0922	DRU
0803	TEX	0843		0883	ZS	0923	U
0804		0844	S	0884	R	0924	2
0805	G	0845	T	0885	I	0925	DRU
0806	E	0846	A	0886	C	0926	2
0807	B	0847	R	0887	H	0927	DRU
0808	E	0848	T	0888	T	0928	U
0809	N	0849	TEX	0889	I	0929	3
0810		0850	ZS	0890	G	0930	DRU
0811	S	0851	KOM	0891	E	0931	2
0812	I	0852	1	0892	N	0932	DRU
0813	E	0853	TRX	0893	-	0933	U
0814		0854	1	0894	F	0934	4
0815	A	0855	4	0895	A	0935	DRU
0816	U	0856	STP	0896	R	0936	2
0817	F	0857	SEL	0897	B	0937	DRU
0818	:	0858	0	0898	E	0938	U
0819	ZS	0859	8	0899	N	0939	5
0820		0860	6	0900	-	0940	DRU
0821		0861	6	0901	-	0941	2
0822		0862	SPR	0902	-	0942	TEX
0823	S	0863	2	0903	-	0943	ZS
0824	E	0864	3	0904	-	0944	ZS
0825	L	0865	4	0905	-	0945	ZS
0826	=	0866	KOM	0906	-	0946	STM
0827	1	0867	0	0907	-	0947	NGR
0828		0868	TEX	0908	-	0948	TEX
0829	S	0869		0909	-	0949	
0830	T	0870		0910	-	0950	M
0831	A	0871	D	0911	-	0951	I
0832	R	0872	A	0912	-	0952	R
0833	T	0873	S	0913	-	0953	
0834	ZS	0874		0914	-	0954	R
0835		0875	S	0915	-	0955	E
0836	W	0876	I	0916	TEX	0956	I
0837	E	0877	N	0917	TRX	0957	C
0838	I	0878	D	0918	1	0958	H
0839	T	0879		0919	TEX	0959	T

5. Schritt

Eingabe: 9 7 10 2 8

Antwort: 3 GOLD 2 SILBER

- Zur Positionsüberprüfung wird mit der Zahl begonnen, von der ganz sicher ist, daß sie sich auf der richtigen Position befindet, also die Zahl 7 auf der zweiten Position.

- Überprüfung der nächsten Positionen; Beginn mit der Zahl 9 auf der ersten Position:

- 9 kann nicht mit dritter Position wechseln, denn 10 lag schon im 4. Schritt auf erster Position,
- 9 kann nicht mit vierter Position wechseln, denn 2 lag schon im 1. Schritt auf erster Position,
- 9 kann nicht mit fünfter Position wechseln, denn 8 lag schon im 3. Schritt auf erster Position.

Damit verbleibt die Zahl 9 auf ihrer Position.

- Überprüfung der Zahl 10 auf der dritten Position:

- 10 kann mit vierter Position wechseln, denn 2 befand sich noch nicht auf dritter Position,
- 10 kann nicht mit fünfter Position wechseln, denn 8 lag schon im 1. Schritt auf dritter Position.

Damit können nur noch die Zahlen 2 und 10 ihre Positionen tauschen.

6. Schritt

Eingabe: 9 7 2 10 8

5 GOLD 0 SILBER

- Rechnerausdruck:
GRATULIERE
SIE SIND EIN
PFIFFIKUS!

0960
0961 E
0962 S
0963 ZS
0964 J
0965 E
0966 T
0967 Z
0968 T
0969
0970
0971 S
0972 I
0973 E
0974
0975 K
0976 A
0977 M
0978 E
0979 L
0980 TEX
0981 ZS
0982 SPR
0983 8
0984 6
0985 6
0986 END
0987 NUL
0988 NUL

4. Zum Schluß

Am Anfang erwähnten wir den ungarischen Zauberwürfel. Er fordert mit seinen 43 Trillionen verschiedenen Farbkombinationen zweifellos unseren Geist heraus, und der amtierende Weltmeister ordnet ihn in weniger als 20 Sekunden. Bei unserem Spielprogramm kam es nicht auf Tempo an, obwohl auch diese Variante, dem Blitzschach ähnlich, denkbar wäre. Die folgende Betrachtung liefert die Anzahl der Kombinationsmöglichkeiten: Da 5 aus 10 Zahlen auszuwählen sind, ergeben sich $\binom{10}{5} = 252$ Möglichkeiten. Da aber nicht nur die Zahlen auszuwählen sind (wie bei 5 aus 35 und 6

aus 49), sondern auch noch die richtigen Positionen gesucht werden müssen, muß diese Zahl der Möglichkeiten noch mit $5! = 120$ Möglichkeiten multipliziert werden. Damit ist nur eine von 30240 Möglichkeiten die richtige. Und das in sieben Schritten zu schaffen erfordert schon einen meisterhaften Verstand.

Autoren:

Dr. Hannes Gutzer
DDR 4090 Halle-Neustadt
Am Südpark 581/6

Werner Pihaule
DDR 7025 Leipzig
Ostrowskistr. 25

Rechentechnische Begriffe für den Laien erklärt

- Assembler** Maschinorientierte Programmiersprache unter Verwendung symbolischer Begriffe für die ausführbaren Maschinenbefehle
- BASIC** Leicht erlernbare problemorientierte Programmiersprache, die insbesondere bei Heimcomputern verwendet wird
- PASCAL** Anspruchsvollere und sehr leistungsfähige problemorientierte Programmiersprache. Sie wird u. a. als PASCAL-1520 bei den Bürocomputern eingesetzt.
- Compiler** Programm, das ein in einer problemorientierten Programmiersprache geschriebenes Programm in die Maschinensprache des jeweiligen Rechners übersetzt
-

Nimm-Spiel gegen den K 1003



Schon vor dem Anbruch der Computer-ära wurde das *Nimm-Spiel* gespielt. Ohne die leiseste Ahnung von einem Taschen- oder Tischrechner spielten zwei Personen gegeneinander. Das war und ist auch heute noch mit ganz einfachen Spielregeln möglich: Auf den Tisch wird eine bestimmte Anzahl von Gegenständen gelegt. Hier bieten sich Streichhölzer an, weshalb das Spiel auch als *Streichholzspiel* bekannt ist. Weiterhin wird festgelegt, welche maximale Anzahl von Streichhölzern jeder Spieler bei jedem Zug wegnehmen darf, wobei mindestens ein Holz wegzunehmen ist. Verloren hat diejenige Person, die das letzte Streichholz wegnehmen muß.

Im praktischen Spielverlauf wurde oft deutlich, daß, von wenigen Ausnahmen abgesehen, stets die gleiche Person gewann. Sie war es meistens auch, die das Streichholzspiel vorschlug, weil sie offenbar wußte, »wie es geht«, und auf die Unwissenheit des Gegners hoffte.

Wir wollen bei diesem Spiel die eine der beiden Personen durch den programmierbaren Tischrechner K 1003 ersetzen, also gegen den Rechner spielen. Es eignen sich dafür natürlich auch andere programmierbare Rechner. Ein tastenprogrammierbarer Taschenrechner sollte aber mindestens 150 Befehlsplätze und 5 Datenplätze haben. Besonders geeignet sind Rechner mit

Drucker (K 1003, TI 58/59, HP 41, PC-Serie von Sharp, PB-Serie und FX-Serie von Casio u. a.). Sie drucken den Spielverlauf aus und ermöglichen über Textausdrucke einen Minidialog mit dem Rechner. Das wiederum ist bedienerfreundlich, weil umfangreiche Beschreibungen des Programmablaufes entfallen. Die Textprogrammierung »frißt« allerdings Befehlsschritte, wie wir noch sehen werden.

Wir haben aus einer Vielzahl von Spielen hier bewußt das Streichholzspiel gewählt, da es, vor allem für den Anfänger, einen zweifachen Erkenntnisgewinn bringt (vom Spaß einmal ganz abgesehen).

1. Das Streichholzspiel ist gut geeignet, um die Schritte von P (wie Problem) nach P (wie Programm) zu zeigen, nämlich: *Problemanalyse, Lösungsalgorithmus, Programmablaufplan, Rechenprogramm*.

2. Das Streichholzspiel macht deutlich, daß der Rechner mit seinem Programm nie »schlauer« sein kann als der Mensch. Damit ist natürlich nur der Mensch gemeint, der um die Dinge weiß.

Dieses Spiel bietet damit auch für Arbeitsgemeinschaften, die u. U. noch keinen programmierbaren Rechner zur Verfügung haben, Denkvergnügen, Diskussionsstoff und Spaß.

Beginnen wir nun mit der **Problem-analyse**. Der Rechner und wir, also die beiden Gegner, versuchen, die *Gewinnstrategie* anzuwenden. Kennen wir sie nicht, so haben wir keine Siegeschancen. Um diese Gewinnstrategie zu erfahren, werden wir »das Pferd am Schwanz aufzäumen«. Dazu betrachten wir folgendes Beispiel: Die Gesamtzahl G der Streichhölzer sei $G = 15$. Die maximale Anzahl pro Zug, kurz als Maximalzug M bezeichnet, sei $M = 3$. Damit liegen 15 Streichhölzer auf dem Tisch. Der Rechner und wir dürfen also pro Zug 1 oder 2 oder 3 Hölzer wegnehmen. Damit der Rechner verliert, muß er in der letzten Runde genau 1 Streichholz vorfinden, das er notgedrungen ziehen muß. In der vorletzten Runde aber muß der Rechner 5 Hölzer vorfinden. Wenn er nämlich 3 Hölzer wegnimmt (Rest 2), dann nehmen wir in der letzten Runde nur 1 Holz weg. Auch das Wegnehmen von einem Holz oder zwei Hölzern führt in der letzten Runde stets zu seiner Niederlage. Führen wir diesen Gedanken fort, dann muß der Rechner in der drittletzten Runde 9 und in der viertletzten Runde 13 Hölzer vorfinden, damit wir zum Sieg gelangen.

Damit lautet für $M = 3$ die Gewinnreihe: 1, 5, 9, 13 . . .

Da die Gesamtzahl $G = 15$ ist, interessieren uns die weiteren Glieder der Gewinnreihe nicht mehr. Da wir aber das Problem für beliebige G , M (wobei $G \gg M$) analysieren wollen, stellen wir folgende allgemeine Gewinnreihe auf:

$$1, M + 2, 2M + 3, 3M + 4, \\ 4M + 5, \dots$$

Anders ausgedrückt ergibt sich ein beliebiges Glied der Gewinnreihe aus der Summe der vorherigen Glieder plus $(M + 1)$.

Wir erkennen daraus, daß der Gewinner schon nach dem ersten Zug feststeht.

Wollen wir gewinnen, so müssen wir sofort nach Spielbeginn auf die Gewinnreihe »springen«. NIMM-Spiel-Programme in Rechenzentren gaukeln dem ahnungslosen Nutzer oft Großzügigkeit vor, indem sie die Wahl lassen, ob Rechner oder Person beginnen soll. Auch ich erinnere mich an die Zeit, in der den Besuchern des Zentralinstitutes für Schweißtechnik in Halle nicht nur schweißtechnische Forschungsergebnisse, sondern auch so ein »großzügiges« NIMM-Spiel vorgeführt wurde. Lassen wir aber dem Rechner den ersten Zug, dann »springt« er sofort auf die Gewinnreihe und hat den Sieg für sich schon in der Tasche (vielleicht besser: im Speicher).

Es gibt allerdings für das eben Gesagte eine einzige *Ausnahme*. Gehört nämlich die Gesamtmenge G selbst zur Gewinnreihe, dann gelingt es dem Beginner im 1. Zug nicht, auf die nächstkleinere Zahl der Gewinnreihe zu springen. Er darf aber auch nicht 0 Streichhölzer ziehen, um auf der Gewinnreihe zu bleiben. Sollte Ihnen also bei Ihrem nächsten Besuch in einem Rechenzentrum beim NIMM-Spiel der Rechner die Wahl des Beginnens lassen, dann wählen sie M und G so, daß G auf der Gewinnreihe liegt, und lassen »großzügig« den Rechner beginnen. Anschließend »springen« Sie dann auf die Gewinnreihe und besiegen ihn trotzdem.

Da der eben geschilderte Fall eine Ausnahme ist, wollen wir in unserem Spielprogramm von vornherein festlegen, daß wir beginnen. Nach dieser Problem-analyse können Sie nun schon gegen einen unwissenden Partner spielen und, sofern Sie sich im Spiel nicht verrechnen, gewinnen. Hier noch zum Abschluß gebräuchliche Gewinnreihen:

Für $M = 3$ gilt:

$$1, 5, 9, 13, 17, 21, 25, 29, \dots$$

Für $M = 4$ gilt:

1, 6, 11, 16, 21, 26, 31, 36, ...

Für $M = 5$ gilt:

1, 7, 13, 19, 25, 31, 37, 43, ...

Der **Lösungsalgorithmus**, wir bezeichnen den Kern der Sache als Gewinnstrategie, trat ja schon bei der eben angestellten Problemanalyse zutage. Für das angestrebte Programm werden wir folgende Schritte zu bearbeiten haben:

1. Eingabe der Gesamtzahl G und des Maximalzuges M ,
2. Eingabe der Anzahl P der Hölzer, die wir zu Beginn ziehen,
3. Prüfung, ob $M \geq P > 0$ ist. Falls dies nicht der Fall ist, wird die falsche Eingabe als Betrugsversuch gewertet und der Ablauf wird abgebrochen.
4. Aufbau der Gewinnreihe, deren aktuellen Wert wir mit W bezeichnen wollen.
5. Sollten wir uns auf der Gewinnreihe befinden, dann tritt für die »Verlegenheitszüge« des Rechners ein Pseudozufallszahlengenerator in Aktion, der aus einer Zufallszahl Z einen zufälligen Rechnerzug P ($M \geq P > 0$) erzeugt und damit für Abwechslung sorgt. Aus diesem Grunde wird auch beim Programmstart die Anfangszahl $Z = 0,5$ vorgegeben.
6. Im Programmablauf muß vor jedem Zug (von uns und vom Rechner) geprüft werden, ob nur noch genau 1 Streichholz übrig ist. Sobald diese Situation eintritt, wird das Programm beendet, und der Drucker weist den Sieg des Rechners oder unseren Sieg (das ist natürlich erfreulicher) aus.

Diese groben Darstellungen des Lösungsalgorithmus reichen für uns als Programmierer natürlich noch nicht aus, um daraus ein Rechenprogramm zu erstellen. Wir werden deshalb in einem **Programmablaufplan**, kurz als PAP bezeichnet, den gesamten Ablauf

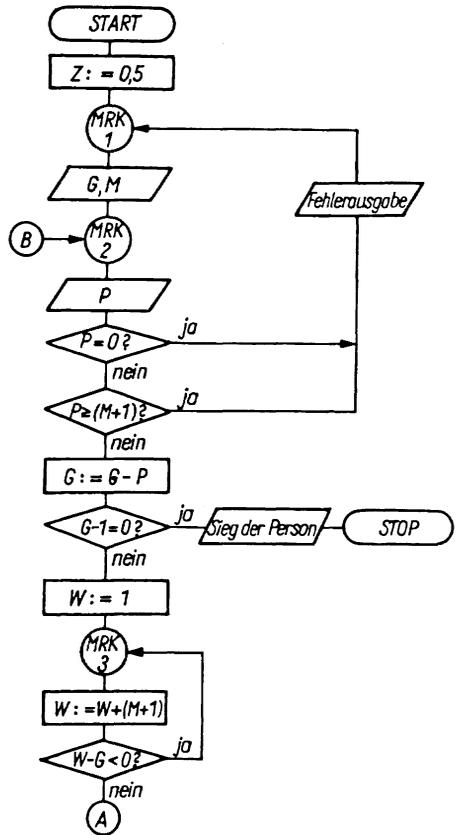


Bild 1. Programmablaufplan

mit den entsprechenden Formelzeichen genau darstellen. Dazu noch einmal die verwendeten Formelzeichen:

- G Gesamtzahl der Streichhölzer
- M Maximalzug (maximale Anzahl pro Zug)
- P Anzahl eines Zuges von Mensch und Rechner
- W Aktueller Wert der Gewinnreihe
- Z Zufallszahl für die Erzeugung eines »Verlegenheitszuges« P durch den Rechner

Den Programmablaufplan in der gebräuchlichen Symbolik zeigt Bild 1.

Nach dem Programmstart wird die Anfangszahl $Z = 0,5$ für den Pseudozufallsgenerator bereitgestellt. Die Marken 1 bis 3 (MRK 1, MRK 2, MRK 3) werden uns die Programmierarbeit erleichtern, da sie *Einsprungstellen in das Programm* festlegen. Zunächst werden G und M eingegeben. Nach Eingabe von P wird dessen Richtigkeit geprüft. Im weiteren wird mit $G - 1 = 0$? die *Endebedingung* geprüft und dann, bei Verneinung dieser Frage, der aktuelle Wert W der Gewinnreihe gebildet. Je nachdem, ob wir auf der Gewinnreihe sind oder ob sich der Rechner darauf befindet, wird das Programm verzweigt. Sind wir auf der Gewinnreihe, dann bildet der Rechner mit Hilfe des Pseudozufallszahlengenerators einen »Verlegenheitszug« P (rechter Teil des PAP). Es gibt viele Möglichkeiten, solche Pseudozufallszahlen zu erzeugen. Im PAP ist eine Variante angegeben, die $\pi + Z$ im Exponenten einer e-Funktion benutzt. Manche Rechner haben in austauschbaren Moduln (z. B. TI 58/59) oder Blöcken (z. B. K 1002/1003) solche Pseudozufallszahlengeneratoren fest programmiert. Allerdings ist der Generator im Statistik-Block des K 1002/1003 sehr bescheiden, was die Güte der Zufälligkeit anbelangt. Die als zehnstelliger Dezimalbruch einzugebende Anfangszahl wird mit der Zahl 29 multipliziert. Der entstehende Nachkommateil liefert dann eine Zufallszahl zwischen 0 und 1. Die Normierung für P mit $M \geq P > 0$ kann dann wie in unserem PAP über $P = \text{int}(M \cdot Z) + 1$ erfolgen [int ($M \cdot Z$) heißt ganzzahliger Anteil von ($M \cdot Z$)]. Ist der Rechner aber selbst siegesgewiß auf der Gewinnreihe, dann wird der linke Programmzweig des PAP durchlaufen, bei dem auch die Endebedingung für den Sieg des Rechners geprüft wird.

Der Anfänger kann mit Papier, Blei-

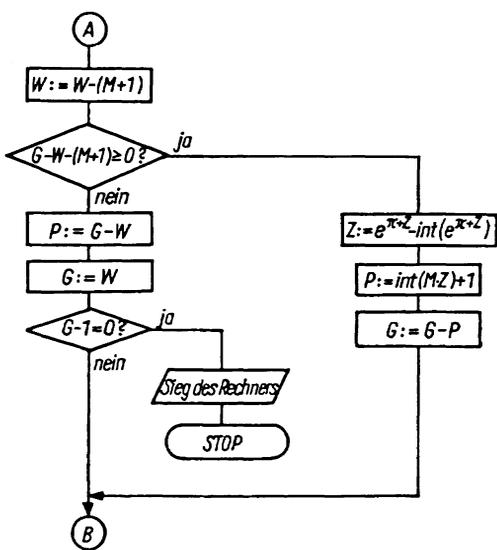


Bild 1 (Fortsetzung)

stift und einem Beispiel versuchen, diesen PAP im »Trockenkurs« abzuarbeiten. Dabei ist besonders auf solche Ergibt-Anweisungen wie $G := G - P$ oder $W := W + (M + 1)$ zu achten. Die Krönung unserer Bemühungen soll aber nun das **Rechenprogramm** sein. Allerdings wird an dieser Stelle deutlich, daß der *entscheidende geistige Aufwand in den Stufen Problemanalyse und Lösungsalgorithmus* steckt. Mit einem guten Programmablaufplan und soliden Programmierkenntnissen ist dieser letzte Schritt bei weitem nicht so »geistintensiv« wie die vorangegangenen. Wir wollen, wie schon eingangs erwähnt, das Programm für den tastenprogrammierbaren Tischrechner K 1003 unter effektiver Nutzung des Thermodruckwerkes schreiben. Besitzer anderer programmierbarer Rechner können den PAP aus Bild 1 natürlich auch für ihren Rechner in ein Rechenprogramm umsetzen. Ebenso gut wäre die

Programmierung in einer problemorientierten Sprache wie FORTRAN, BASIC oder PASCAL möglich. Für den K 1003 werden wir die Datenspeicher folgendermaßen belegen:

Datenspeicherplatz

000	enthält	$M + 1$
001	enthält	G
002	enthält	W
003	enthält	P
004	enthält	Z

Es werden also insgesamt 5 Datenspeicherplätze benötigt. Damit kann die beim K 1003 beim Einschalten vorhandene Speicherplatzverteilung mit 7 Datenspeicherplätzen beibehalten werden.

Der *Numerateur* beim K 1002/1003 hilft beim Betrieb ohne Drucker bei der *Kennzeichnung der Ein- und Ausgaben*. Obwohl er im Druckerbetrieb mit Dialog (so wollen wir unser Programm erstellen) nicht benötigt wird, wollen

wir ihn doch für die K-1002-Besitzer ins Programm einbauen. Allerdings müßten für den nichtdruckenden Betrieb noch einige Programmänderungen, insbesondere für die Ausgabe der Niederlage- oder Siegesnachricht, vorgenommen werden. Für den Numerateur (NUM) gilt folgendes:

bei NUM = 01	Eingabe von G
bei NUM = 02	Eingabe von M
bei NUM = 03	Eingabe von P

Das vollständige Rechenprogramm für den K 1003 und die Fortsetzung des Artikels erscheinen in Heft 2.

Autor:

Dr. Hannes Gutzer
DDR 4090 Halle-Neustadt
Am Südpark 581/6

Leiter der Informationsstelle
des Zentralinstituts für Schweißtechnik
Halle

Neues aus aller Welt

In der Presse der vergangenen Zeit wurde folgende Meldung veröffentlicht:

LEICHTESTER RECHNER

Tokio. Den leichtesten und dünnsten Taschenrechner der Welt will die japanische Firma CASIO COMPUTER Co. nach eigenen Angaben herausgebracht haben. Sein Gewicht soll 12 Gramm, die Dicke 0,8 Millimeter betragen.

Zum Vergleich sollen die entsprechenden Werte des Rechners MR 610 gegenübergestellt werden: er wiegt nach Werksangaben 90 Gramm und seine Dicke beträgt 8,5 Millimeter.

Wenn Sie mich fragen sollten: Mir ist der MR 610 leicht und dünn genug. Wichtig ist für mich, daß er zuverlässig und genau rechnet. Und das tut er!

HK

© VEB Fachbuchverlag Leipzig 1984

1. Auflage

Lizenznummer 114-210/49/84

LSV 1083

Verlagslektor: Helga Fago

Gestaltung: Lothar Gabler

Printed in GDR

Satz und Druck: Messedruck Leipzig,

Bereich Borsdorf III-18-328

Redaktionsschluß: 15. 3. 1984

Bestellnummer: 546 920 8

00780

Vorschau

Heft 2

Horn: Zur Programmierung einer einfachen Aufzugssteuerung mit dem Mikroprozessor U 880

Schönfelder: HOMECOMPUTER – ein neues Gebiet für den Amateur (Teil 2)

Gutzer: Nimm-Spiel gegen den K 1003

Kreul: Wie rechnet ein elektronischer Taschenrechner?

Keller: Der Heimcomputer Z 9001

Heft 3

Ehrsam: Die Taschenrechnerproduktion der DDR

Schönfelder: HOMECOMPUTER – ein neues Gebiet für den Amateur (Teil 3)

Hübner: Polycomputer 880

Schönfelder: Computerspiele – mehr als eine Spielerei

Horn: Wie kann ein Programm systematisch entworfen werden?

Die Broschürenreihe
KLEINSTRECHNER-TIPS

behandelt

- Tendenzen und Theorien
- Informationen und Ideen
- Programme und Projekte
- Spaß und Spiel

und will dem Laien auf dem Gebiet der Informationsverarbeitung Anregungen geben für seine Arbeit mit

- einfachen oder programmierbaren Tisch- und Taschenrechnern,
- Heim- und Videocomputern,
- Mikrorechnern

und anderen modernen Rechenhilfsmitteln.