

informum

INfOrMationsforum des Rechenzentrums der Universität Münster

Jahrgang 6, Nr. 3

Juli 1982

Inhalt	Seite
Lehrveranstaltungen im 2. Halbjahr 1982	2
Programmkatalog des Rechenzentrums	3
NAG-Bibliothek, Mark 2	9
Personalia	10
DIGITAL	11
Überblick Mikrocomputer-Hardware	13
Zum PI/I-Datentyp DECIMAL FLOAR(33)	26
Störungen im Rechenbetrieb	27
informum-Quiz	27
Sprechzeiten	29

Lehrveranstaltungen im 2. Halbjahr 1982

1) Veranstaltungen in der vorlesungsfreien Zeit

In der vorlesungsfreien Zeit vor Beginn des WS 1982/83 werden vom Rechenzentrum einige ganztägige Intensivkurse durchgeführt, in denen Stoffvermittlung und Übungen integriert sind. Diese Veranstaltungen sollen durch entsprechende Betreuung der Teilnehmer eigene Programmierübungen fördern. Das bedingt eine Begrenzung der Teilnehmerzahl der einzelnen Veranstaltungen. Interessenten werden deshalb gebeten, sich möglichst bald, spätestens jedoch eine Woche vor Beginn der entsprechenden Veranstaltung im Dispatch des Rechenzentrums (Raum 02) in die Anmeldelisten einzutragen.

Die Teilnehmer dieser Kurse werden gebeten, diese im WS 1982/82 zu belegen. Dies ist neben der bestandenen Abschlußklausur Voraussetzung für die Aushändigung eines Scheines über die erfolgreiche Teilnahme.

Vorkenntnisse in der Programmierung sind für diese Lehrveranstaltungen nicht erforderlich.

30.8. - 10.9. 1982: Programmieren in FORTRAN

FORTRAN ist eine einfache, aus verhältnismäßig wenigen Sprachelementen bestehende und schnell zu erlernende Programmiersprache, die sich hauptsächlich zur Bearbeitung numerischer Probleme, insbesondere aus dem naturwissenschaftlichen Bereich eignet.

Dozenten: Nienhaus/Steinlausen
Verl.-Nr.: 320017/320024
Hörsäle: M2 (vormittags), M1, M4 (nachmittags)
Beginn: 30.8.1982 um 9 Uhr c.t. in M2

13.9. - 1.10. 1982: Programmieren in PL/I

Ziel der Vorlesung ist eine allgemeine Einführung in die Programmierung. Es soll die systematische Konstruktion von Algorithmen für einfache Probleme sowie die Formulierung dieser Algorithmen in der Sprache PL/I intensiv geübt werden.

Vorkenntnisse werden nicht vorausgesetzt.

Dozent: Neukäter/Pudlatz
Verl.-Nr.: 320036/320040
Hörsaal: großer Hörsaal im Institut für Pharmazeutische Chemie
Beginn: 13.9.1982 um 9 Uhr c.t.

4.10. - 15.10. 1982: Programmieren in ALGOL W

ALGOL W ist eine einfache, aus wenigen Grundelementen bestehende Programmiersprache, die aufgrund ihres klaren Aufbaus besonders

dem Anfänger das Erlernen der Methoden und Techniken der strukturierten Programmierung erleichtert. Sie eignet sich außer zur Bearbeitung numerischer Probleme auch für Fragestellungen der Informatik.

Dozent: Wickenscheidt
Vorl.-Nr: 320055
Hörsaal: M5
Beginn: 4. 10. 1982 um 9 Uhr c.t.

27.9. - 8. 10. 1982: Einführung in die Programmierung mit dem statistischen Programmpaket SPSS

Vorgestellt wird das Programmsystem STSS ("Statistical Package for the Social Sciences") in der am hiesigen Rechenzentrum implementierten Version 9. Mit diesem System stehen bequem aufrufende Programme zu den gebräuchlichen univariaten und multivariaten statistischen Verfahren bereit. Als typisch für die Anwendungsfreundlichkeit des Systems sei angeführt, daß viele Benutzer unseres Hauses die Auswertung ihrer Probleme - in der Regel die Auswertung von Fragebogenaktionen - ausschließlich mit den Prozeduren des SPSS durchführen.

In dieser Veranstaltung wird den Teilnehmern das programmtechnische Rüstzeug zur Durchführung derartiger Auswertungen vermittelt. Solide Grundkenntnisse bezüglich der anzusprechenden statistischen Verfahren wie Kenntnisse der Anwendungsmöglichkeiten dieser Verfahren im jeweiligen Fachgebiet sind erwünscht und bei den praktischen Übungen von großem Nutzen.

Dozenten: Steinhausen, Zirkendörfer
Vorl.-Nr.: 320060
Hörsaal: M6
Beginn: 27.9. 1982 um 9 Uhr c.t.

2) Veranstaltungen in WS 1982/83

Einführung in die EDV

Die Vorlesung vermittelt grundlegende Kenntnisse über die Arbeitsweise von elektronischen Rechanlagen.

Zum Verständnis des Aufbaus eines Rechners führt sie in die Grundlagen der technischen Informatik ein. Hier sind folgende Themen vorgesehen:

- Schaltwerkentwurf
- Steuerwerke (mikroprogrammierbare Schaltwerke, Mikroprozessoren)
- Rechenwerkstrukturen
- Speicher (Arbeitsspeicher, periphere Speichermedien)
- Ein/Ausgabe-Geräte

Im zweiten Teil beschäftigt sich die Vorlesung mit einigen Themen aus der praktischen Informatik:

- Algorithmen und Programmierung
- Datenstrukturen und ihre Verarbeitung
- Betriebssysteme

Vorkenntnisse werden nicht vorausgesetzt.

Dozent: Kisker

Vorl.-Nr.: 320074
Hörsaal: M6
Zeit: Mo 13-15, Beginn: 25.10.1982

Programmieren in FORTRAN

(Inhalt der Lehrveranstaltung siehe oben, Vorlesungsnummer 320017/320024)

Dozent: Ost
Vorl.-Nr.: 320089
Hörsaal: M4
Zeit: Mi 13-15, Beginn: 20.10.1982

Programmieren in FORTRAN

(Inhalt der Lehrveranstaltung siehe oben, Vorlesungsnummer 320017/320024)

Dozent: Reichel
Vorl.-Nr.: 320093
Hörsaal: M3
Zeit: Di 13-15, Beginn: 19.10.1982

Programmieren in PL/I (numerische Anwendungen)

PL/I ist eine universell verwendbare Programmiersprache. Die Vorlesung richtet sich an Hörer, die das Programmieren erlernen und dabei Kenntnisse über PL/I erwerben möchten. Die vorgeführten Beispiele und die Aufgaben werden in erster Linie den numerischen Anwendungsbereichen entnommen.

Dozent: Mertz
Vorl.-Nr.: 320108
Hörsaal: M6
Zeit: Di 13.30-15, Beginn: 19.10.1982
+ 2 Std. Übungen

Programmieren in PL/I (nichtnumerische Anwendungen)

Ziel der Vorlesung ist eine allgemeine Einführung in die Programmierung. Es soll die systematische Konstruktion von Algorithmen in der Sprache PL/I intensiv geübt werden. Die Beispiele werden hauptsächlich aus dem nichtnumerischen Bereich gewählt. Vorkenntnisse werden nicht vorausgesetzt.

Dozent: Benduhn-Mertz
Vorl.-Nr.: 320112
Hörsaal: M4
Zeit: Di 11.30-15, Beginn: 19.10.1982
+ 2 Std. Übungen

Programmieren in PASCAL

PASCAL ist eine moderne Programmiersprache aus der "ALGOL-Familie", die sich sowohl für Fragestellungen der Informatik als auch für Anwendungen aus dem numerischen und nichtnumerischen Bereich eignet. Die Sprache ist auch dadurch von zunehmendem

Interesse, daß sie auf vielen in den letzten Jahren auf den Markt gekommenen Mikrorechnern eingesetzt werden kann. In dieser Lehrveranstaltung wird eine Einführung in das Programmieren anhand von PASCAL gegeben, wobei die besonderen Struktur- und Ausdrucksmöglichkeiten beim Programmwurf und bei der Datenbeschreibung betont werden sollen. Besondere Vorkenntnisse sind nicht erforderlich.

Dozent: Kaspar
Vorl.-Nr.: 320127
Hörsaal: M5
Zeit: Mi 9-11, Beginn: 27.10.1982

Programmieren in PL/I für Fortgeschrittene

Die Vorlesung wendet sich an Hörer, die bereits an einem PL/I-Grundkurs teilgenommen haben. Anhand von Ausschnitten aus am Rechenzentrum benutzten PL/I-Programmsystemen, wie z.B. Plot-Preview, Plottersteuerung, SCRIPT und anderen sollen höhere Programmieretechniken vermittelt werden. Hierzu zählen Themen wie Ein- und Ausgabe auf Platten und Bändern, Datenorganisation, Unterprogrammetechniken, Listenverarbeitung, Multitasking und die Benutzung des PL/I-Preprocessors. Der Hörer soll einen Einblick in die Möglichkeiten einer universellen Programmiersprache erhalten und in die Lage versetzt werden, effiziente und gut strukturierte Programme zu schreiben.

Dozent: Sturm
Vorl.-Nr.: 320131
Hörsaal: M4
Zeit: Mi 15-17, Beginn: 27.10.1982

Methoden der Anwendungsprogrammierung

Im Rahmen dieser Vorlesung werden verschiedene, heute verwendete Techniken und Werkzeuge der Programmierung vorgestellt. Mit wachsender Komplexität der Anwendungen hat sich in den letzten Jahren immer mehr die Ansicht durchgesetzt, daß zur erfolgreichen Programmierung mehr gehört, als nur die Kenntnis einer oder mehrerer Programmiersprachen. Dies führte zur Entwicklung der Software-Technologie, die heute als eigenständiges Gebiet gilt, für das Stichworte wie strukturierte Programmierung, Top-Down-Entwurf, formale Spezifikationen und Modularisierung charakteristisch sind. Diese und weitere Aspekte werden in der Vorlesung behandelt.

Hörer dieser Vorlesung sollten eine Programmiersprache kennen und etwas Erfahrung in der Programmierung haben.

Dozent: Elix
Vorl.-Nr.: 320146
Hörsaal: Seminarraum Rechenzentrum, Einsteinstr. 60
Zeit: Mi 10-12, Beginn: 27.10.1982

Höhere Programmiersprachen: SNOBOL4

SNOBOL4 eignet sich besonders zur Bearbeitung nichtnumerischer Probleme (Manipulation von Zeichenketten, Listenverarbeitung)

etc.) und unterscheidet sich in seinem Aufbau wesentlich von anderen Programmiersprachen wie FORTRAN, PASCAL und PL/I. Dadurch ergeben sich andersartige und angenehme Ausdrucksmöglichkeiten für den Benutzer.

Außerdem wird über neuere Sprachentwicklungen auf der Grundlage der Konzeption von SNOBOL4 berichtet.

Programmierkenntnisse sind für die Teilnahme an dieser Lehrveranstaltung erwünscht.

Dozent: Fosse

Vorl.-Nr.: 320150

Hörsaal: Seminarraum Bibliothek

Zeit: Di 15.30-17, Beginn: 26.10.1982

Datenanalyse und statistische Auswertung mit Hilfe eines statistischen Analysesystems

Vorgestellt wird das Programmsystem SAS zur Aufbereitung, Abspeicherung und Pflege von Dateien, zur Erstellung von Listen und Tabellen und zur Durchführung zahlreicher gebräuchlicher statistischer Verfahren. Von programmtechnischer Seite her wird ein reichliches Angebot von Formaten und Einlesevariationen bereitgestellt, ausführlich werden Methoden zum Mischen und Pflegen von Dateien behandelt. Die Auswahl der zu besprechenden problemorientierten Prozeduren wird erst im Laufe der Veranstaltung in Abstimmung mit den Teilnehmern festgelegt; Benutzerprobleme können dabei berücksichtigt werden.

Die Teilnehmer sollten gute Kenntnisse in einer Programmiersprache aufweisen können oder aber Programmiererfahrung im SPSS mit Dateien, die im SESS schwierig auszuwerten sind.

Dozent: Zörkendörfer

Vorl.-Nr.: 320165

Hörsaal: M6

Zeit: Mi 9-11, Beginn (Vorbesprechung): 20.10.1982

Programmsysteme zur Textverarbeitung: Oxford Concordance Package

Ziel dieser Vorlesung ist es, eine Einführung in das Programmsystem OCP (Oxford Concordance Package) zu geben. Dieses System, welches am Rechenzentrum der Universität Oxford entwickelt und zur Verbesserung der Portabilität in FORTRAN implementiert wurde, ermöglicht es, ohne besonderen Programmieraufwand durch eine besondere Kontrollsprache Konkordanzen oder Wortlisten zu Texten einer Vielzahl von Sprachen und Alphabeten zu erzeugen. Allen Geisteswissenschaftlern, insbesondere jenen, denen bisher die Schwelle des Erlernens einer Programmiersprache als Einzug in die sprachwissenschaftliche Datenverarbeitung zu hoch erschien, wird diese Einführung in das Programmsystem OCP besonders empfohlen.

Dozent: Slaby

Vorl.-Nr.: 320170

Hörsaal: Seminarraum Rechenzentrum, Einsteinstr. 60

Zeit: Mi 13.30-15, Beginn: 20.10.82

Rechnernetze

Seit der Einführung der Mikrocomputer werden immer häufiger auch einzelne Arbeitsplätze mit DV-Geräten ausgestattet. Auf Grund der Informationsflüsse in Betrieben dürfen diese Geräte nicht als unabhängige DV-Inseln angesehen werden, vielmehr müssen sie miteinander kommunizieren können und einen Datenaustausch ermöglichen. Daher kommt der Entwicklung von lokalen und globalen Rechnernetzen eine zentrale Bedeutung zu. Während erstere über kürzere Entfernungen sehr hohe Übertragungsgeschwindigkeiten gestatten, erlauben letztere bei kleineren Geschwindigkeiten die Überwindung großer Entfernungen.

In der Veranstaltung sollen technische und logische Grundlagen sowie wirtschaftliche und organisatorische Aspekte der Rechnernetze besprochen werden.

Dozent: Held/Schulze

Vorl.-Nr.: 320184

Hörsaal: M5

Zeit: Mo 13-15, Beginn: 25.10.1982

Vortragsreihe über das Betriebssystem OS/VS2-MVS

In einer Folge von Einzelvorträgen soll zunächst ein Überblick über die Funktionen und die Arbeitsweise des Betriebssystems OS/VS2 MVS für IBM-Großrechenanlagen gegeben werden, um danach einzelne Komponenten und wichtige ergänzende Systemprogramme detailliert darzustellen. Die vorgesehenen Themen umfassen:

• Konzept des virtuellen Speichers im MVS

• MVS Supervisor

• MVS Job Management

• Job Entry Subsystem 2

• Betriebsmittelverwaltung

• Zugriffskontrolle

• Dateiverwaltung

• Fehlerüberwachung

Die Themen der Vorträge werden zu Beginn des Wintersemesters bekannt gegeben.

Dozenten: Elix/Mertz/Meyer-Einsche/Ost/Schulze/Sturm

Vorl.-Nr.: 320199

Hörsaal: M6

Zeit: Mi 15-17, Beginn: 27.10.82

Seminar über Computermusik

In dem Seminar werden Arbeiten aus wissenschaftlichen Zeitschriften besprochen, die sich auf die Erzeugung von Musik mit Hilfe von Digitalrechnern beziehen. Dabei werden die Grundprobleme der Klang-Synthese und des Einsatzes von Computern in Echtzeit im Vordergrund stehen. Außerdem werden Fragen der Kommunikation zwischen Mensch und Musik-Computer behandelt.

Dozent: Brockhoff/Hornung/Slaby

Vorl.-Nr.: (071436)

Hörsaal: Seminarraum Musikwissenschaften

Zeit: Mo 14-15.30, Beginn: 25.10.82

Kolloquium über Themen der Informatik

Im Rahmen des Kolloquiums werden Vorträge über neuere Entwicklungen der Informatik gehalten.

Vorl.-Nr.: 320203

Hörsaal: M6

Zeit: Fr 15-17 bzw. 17-19

Vortragstermine werden durch besonderen Aushang im Rechenzentrum bekanntgegeben

Programmkatalog des Rechenzentrums

F.-P. Spellmann

Der Programmkatalog des Rechenzentrums gilt in zwei Bänden einen Überblick über die am Rechenzentrum implementierte Software (Unterprogrammbibliotheken, Programmsysteme usw.).

Der erste Band enthält im wesentlichen die nach thematischen Gesichtspunkten untergliederte Sammlung der dokumentierten Programme, wobei zu den einzelnen Programmen Kurzbeschreibungen und Hinweise auf die zugehörigen ausführlichen Dokumentationen enthalten sind.

Der zweite Band ist als Ergänzung zum ersten Band gedacht und besteht aus ausführlichen Dokumentationen zu Programmen, die vom Rechenzentrum erstellt bzw. gesammelt wurden und nicht in anderer Form vorliegen.

Nachdem der Programmkatalog in der Vergangenheit (mit dem Schnelldrucker erstellt) nur in wenigen Exemplaren zur Einsicht auslag (z.B. in der Programmierberatung und der Bibliothek des Rechenzentrums), wird das Rechenzentrum den Band 1 des Programmkatalogs in Kürze als Software Information Nr.6 in vervielfältigter Form anbieten, um die Benutzer des Rechenzentrums in weiterem Rahmen über die bereitstehende Software zu informieren. Es ist außerdem vorgesehen, später bei Bedarf aktualisierte Neuauflagen zu erstellen. Sobald die vervielfältigten Exemplare verfügbar sind und zum Kauf angeboten werden können, wird eine entsprechende Bekanntmachung erfolgen.

Der zweite Band des Programmkatalogs, dessen Gegenstand die Dokumentation spezieller, im ersten Band gesondert gekennzeichnete Programme ist, ist mit seinen detaillierteren Ausführungen bei der Auswahl von Programmen weniger nützlich. Er wird deshalb als Software Information Nr.7 des Rechenzentrums wie bisher nur in geringer Stückzahl (wegen des Aufwands der Herstellung) mit dem Schnelldrucker erstellt und liegt in der Programmierberatung und der Bibliothek des Rechenzentrums zur Einsicht aus. Einrichtungen der Universität, die zahlreiche Benutzer des Rechenzentrums betreuen, können auf Antrag ein Exemplar erhalten.

NAG-Bibliothek, Mark 9

L.-P. Spellmann

Seit dem 1.7.82 ist die NAG-Bibliothek in der neuen Version Mark 9 im Einsatz. Sie enthält 17 neue primäre, d.h. zum Aufruf von Benutzerprogrammen vorgesehene Routinen. 20 alte Routinen aus Mark 8, deren Ersetzung u.a. in der letzten Ausgabe des *inform* angekündigt wurde, sind nicht mehr verfügbar. Die neu eingeführten Programme sind drei Bereichen zuzurechnen:

C05NEF, -NCE, -PEF, -PCF, -ZAF

Bestimmung von Nullstellen für Systeme von N nichtlinearen Funktionen mit N Variablen (mit einer Modifikation der Powell Hybrid Methode). Durch Auswahl unter den angebotenen Routinen kann der Benutzer entscheiden, ob er die partiellen Ableitungen der Funktionen angeben will oder nicht und ob er eine "easy-to-use" Routine mit wenigen Parametern oder eine flexiblere Routine mit umfangreicheren Möglichkeiten zur Steuerung wünscht. C05ZAF dient als Hilfsroutine zur Überprüfung der Korrektheit der benutzerspezifizierten partiellen Ableitungen und kann bei Bedarf eingesetzt werden.

Hinweis: C05NEF und C05NCE sind als Ersatz für das alte Programm C05NAF vorgesehen, das (als einziges Programm aus Mark 9) in der nächsten Version Mark 10 der NAG-Bibliothek nicht mehr enthalten sein wird.

E01RAF, -REF

Bestimmung und (anschließende) Auswertung einer interpolierenden rationalen Funktion zu einer Reihe von Datenpunkten und zugehörigen Funktionswerten

G13AAF bis G13AHF (8 Routinen)

Der neue Abschnitt G13 der NAG-Bibliothek ist der Zeitreihenanalyse gewidmet. Die enthaltenen Routinen erfüllen u.a. folgende Aufgaben: saisonale und nicht-saisonale Differenzierung einer Zeitreihe; Bestimmung von (partiellen) Autokorrelationen; vorbereitende Schätzung der Parameter eines ARMA-Modells auf Basis der vorher ermittelten Autokorrelationsfunktion; Anpassung eines ARIMA-Modells an eine beobachtete Zeitreihe und darauf aufbauend Vorhersage der Weiterentwicklung der Zeitreihe; Einbringung neuer Beobachtungen einer Zeitreihe in ein schon vorher bestimmtes ARIMA-Modell des zugrundeliegenden Prozesses.

X02AFF, -AFF

Bei Aufruf von X02AFF bzw. X02AHF werden die größten zulässigen positiven Argumente für die FORTRAN-Funktionen DEXI (EXP) bzw. DSIN (SIN) und DCOS (COS) zurückgegeben. Diese maschinenabhängigen Werte werden speziell von anderen NAG-Programmen, die eine der genannten FORTRAN-Funktionen aufrufen, zur Vermeidung vorhersehbarer Fehler benötigt.

(Neben den oben genannten Programmversionen in doppelter Genauigkeit sind jeweils auch deren Versionen in einfacher Genauigkeit angesprochen, so etwa mit C05NEF auch C05NBE.)

Personalia

Zum 31.3.1982 hat Herr Krause seine Tätigkeit als studentischer Mitarbeiter am Rechenzentrum beendet.

Zum 1.4.1982 wurden die Herren H. Aschmann, G. Hillebrand und E. Knust als studentische Mitarbeiter am Rechenzentrum eingestellt.

Herr K. Lange ist seit dem 3.5.1982 als Operateur bei uns tätig.

Zum 1.6.1982 ist Herr L. Lasthaus-Freund nach bestandener Diplomprüfung als wissenschaftlicher Mitarbeiter in der Abteilung linguistische Datenverarbeitung am Rechenzentrum eingestellt worden. Herr Lasthaus-Freund war bisher schon in diesem Bereich als studentischer Mitarbeiter tätig.

Herr M. Lauterbach hat uns nach nur kurzer Mitarbeit in der Abteilung Numerische Datenverarbeitung zum 1.7.82 wieder verlassen und eine Stelle als wissenschaftlicher Mitarbeiter am Institut für Numerische und instrumentelle Mathematik angetreten.

Nach Erlangung seines Diploms in Mathematik hat Herr M. Schücke seine Tätigkeit als studentischer Mitarbeiter am Rechenzentrum beendet.

Impressum

Redaktion inforum

W. Bosse	(Tel. 83-2461)
E. Pudlatz	(Tel. 83-2472)
R. Schmitt	(Tel. 83-2475)
W. A. Slaty	(Tel. 83-2681)
E. Sturm	(Tel. 83-2609)

Satz: T. Engelbert, C. Icking
Druck: E. Mecke

Rechenzentrum der Universität
Einsteinstr. 60
4400 Münster

Auflage dieser Ausgabe: 700
Redaktionsschluß der nächsten Ausgabe: 30.09.1982

DIGITAB

E. Sturm

Dank der Mithilfe von Benutzern konnte das Programm zur Unterstützung des graphischen Tablettts TEKTRONIX 4953 seine endgültige Gestalt annehmen. Es steht jetzt durch die JCI-Anweisung

// IXIC DIGITAB

zur Verfügung. Es kann, wie schon in inforum Jg. 6, Nr. 1 beschrieben, ab 16 Uhr alternativ zu PLOT-PREVIEW laufen. In der JOB-Anweisung ist als Jobtyp GFAF anzugeben.

DIGITAB wird durch ein "Menü" gesteuert, wie es auch bei PLOT-PREVIEW üblich ist. Man sollte jedesmal als erstes die Wahl "P" treffen, um sich einige wesentliche Punkte bei der Benutzung des graphischen Tablettts ins Gedächtnis zurückzurufen. Diese Punkte sollen auch hier schon einmal besprochen werden.

Vor dem Digitalisieren muß das Tablett sorgfältig mit Hilfe des beiliegenden Stabmagneten magnetisiert werden, sonst treten später an einzelnen Stellen wilde Koordinatensprünge auf. Sodann sollte man die zu digitalisierende Vorlage mit Klebstreifen auf dem Tablett befestigen. Hierzu ist die Petrietsart "AUTOM." hilfreich, da man, wenn man mit dem Stift auf dem Tablett hin- und herfährt (Stiftschalter nicht drücken!), auf dem Bildschirm den sogenannten Cursor verfolgen kann. So kann man feststellen, welcher Bereich des Tablettts zum Digitalisieren benutzt werden kann. Verläßt man mit dem Stift diesen sogenannten "Präsenzbereich" seitlich oder durch Abheben (etwa 2 mm), so bleibt der Cursor auf dem Bildschirm stehen.

Sollte man Schwierigkeiten mit dem Befestigen der Vorlage haben, wähle man die Betriebsart "PAUSE". Der Bildschirm wird dann zur Schonung der Bildröhre gelöscht, und das Programm wartet beliebig lange (nicht nur 5 Minuten, wie in anderen Betriebsarten) auf eine Eingabe. Nach Drücken der RETURN-Taste erscheint das Menü wieder.

Zum Digitalisieren kann man in Abhängigkeit von dem gewünschten Ergebnis zwischen den folgenden drei Betriebsarten auswählen:

1) TEXT:

An jeder Stelle kann erklärender Text eingefügt werden. Dieser Text erscheint sowohl auf dem Bildschirm an der letzten Cursor-Position als auch in der noch zu besprechenden Ausgabe-Datei.

2) EINZELN:

In diesem Modus können Polygonzüge aufgenommen werden. Hat man mit dem Stift den gewünschten Punkt fixiert, drückt man den Stift fester auf das Tablett. Der eingebaute Schalter bewirkt dann die Übertragung eines Punktes. Verläßt man mit dem Stift nicht den Präsenzbereich, werden alle so erzeugten

Punkte auf dem Bildschirm durch einen Strich verbunden. Durch ein zwischenzeitliches Verlassen des Präsenzbereiches wird jeweils ein neuer Polygonzug begonnen. Es erfordert allerdings eine gewisse Übung, den Stift nicht schon aus Unachtsamkeit abzuheben.

3) AUTOM.:

In diesem Modus wird bei gedrücktem Stiftschalter automatisch ein Punkt nach dem anderen übertragen. Alle Punkte werden ebenfalls auf dem Bildschirm als Polygonzug gezeichnet. Bei nicht gedrücktem Stiftschalter kann man auf dem Bildschirm verfolgen, welche Koordinaten beim Drücken erzeugt wurden. Bei Verlassen des Präsenzbereiches wird der Polygonzug unterbrochen.

Man kann durch Drücken der RETURN-Taste eine Betriebsart verlassen und nach Erscheinen des Menüs eine andere wählen (einschließlich PAUSE). Das Tablett ist nur dann bereit zur Übertragung eines Punktes, wenn ein Piep ertönt und am Steuergerät die READY-Lampe leuchtet.

Nun zu der Frage, wo man die beim Digitalisieren erzeugten Informationen abholen kann. Normalerweise stehen die Texte und Koordinaten in einem PL/I-gemäßen Format unter dem DD-Namen PICOUT als SYSOUT-Dataset zur Verfügung. Jeweils durch ein Blank getrennt, könnte man z.B. finden (als Klartext!)

T 'BEISPIEL' M O O D 1024 1024 usw.

Diese Daten lassen sich in PL/I durch eine abwechselnde Verwendung von

```
GET IDIT (TYP) (A(1)); und
GET LIST (TEXT);      bzw.
GET LIST (X,Y);        weiter verarbeiten.
```

Als TYP bedeutet T Text, M Move (vorher Präsenzbereich verlassen) und D Draw (also Strich auf dem Bildschirm), die Koordinaten bewegen sich im Bereich 0 bis 1024. Da dieses freie Format mit FORTRAN-Mitteln kaum zu bearbeiten ist, besteht die Möglichkeit, durch den alternativen Aufruf

```
// EXEC DIGITAB,PAME=FORTRAN
```

nach dem Format (6(A1,2F5.0,1X)) die Daten in festen Positionen auszugeben. Taucht ein T auf, so erscheint anstatt einer X-Koordinate die Länge des Textes. Der Rest des Ausgabe-Records (Länge 80) bleibt frei und auf dem nächsten Record steht ab Spalte 1 der entsprechende Text. Danach geht es wiederum erst am Anfang des nächsten Records weiter.

Die Geräte werden im Dispatch ausgegeben. Zur Einweisung in das graphische Tablett steht Herr Reichel zur Verfügung (Tel. 2481).

Überblick Mikrocomputer-Hardware

Mikrocomputer sind seit einigen Jahren "im Kommen". An einigen Universitätsinstituten sind bereits etliche als dezentrale Rechner im Einsatz. Das Rechenzentrum verfolgt die Entwicklung dieser kleinen preiswerten Rechner mit Interesse, ist selbst mit der Konfigurierung solcher Anlagen und deren Programmierung befaßt und ist bereit, bei der vom jeweiligen Problem abhängenden Auswahl eines Rechners Hilfestellung zu leisten. Für die Überlassung des folgenden Artikels sind wir Herrn Klaus Peschlow vom Rechenzentrum der Universität Köln zu Dank verpflichtet.

1. Geschichte und Grundbegriffe

Aufgrund der rasanten technischen Entwicklung auf diesem Gebiet sind Begriffsdefinitionen und Einteilungskriterien schnell überholt und damit mißverständlich. Deshalb ist zum Verständnis ein Abriß der "Entwicklungsgeschichte" der Mikrocomputer erforderlich:

Die klassischen Datenverarbeitungsanlagen teilt man in mehrere Entwicklungsgenerationen ein: Die "erste Generation" realisierte die logischen Grundfunktionen mit Elektronenröhren, die "zweite" mit Transistoren und die "dritte" mit integrierten Schaltkreisen (ICs=Integrated Circuits), in denen die Funktion mehrerer Transistoren zusammengefaßt wurde. Die "dritte Generation" kam Anfang der sechziger Jahre aus den Labors und begründete den Siegeszug der Computer. Die klassischen Datenverarbeitungsanlagen dieser Art, z.B. Rechner der Computerfamilie IBM 360, benötigten zur Unterbringung einen großen Raum ("Maschinen-saal"). Die Zentraleinheiten (CPUs=Central Processing Units) dieser Anlagen konnten mindestens 32 Bit breite Datenwörter verarbeiten und verfügten neben einer Festkomma-Arithmetik auch über Gleitkomma-Arithmetik. Eine wesentliche Differenzierung innerhalb dieses CPU-Typs ergab sich aus Unterschieden im Befehlsvorrat, in der maximal verwendbaren Zentralspeicherkapazität und vor allem in der Prozessorgeschwindigkeit.

In den sechziger Jahren wurden für spezielle Anwendungen, insbesondere im Bereich der Prozeß-Steuerung, kleinere CPU-Typen mit geringeren Datenwortbreiten entwickelt. Diese CPUs verfügten neben den logischen Operationen nur über eine Festkomma-Arithmetik, Gleitkomma-Operationen mußten durch Programme emuliert, d.h. nachgebildet werden. Anlagen dieser Art konnte man in Schrankgröße bauen, man nannte sie "Minicomputer". Neben einer Datenwortbreite von 16 Bit war als ein weiteres technisches Merkmal die Verwendung eines "Bussystems" für sie typisch: Alle Baugruppen eines solchen Computers sind in einheitlicher Weise an eine gemeinsame "Signalsammelschiene", den "Bus", angeschlossen. Die einzelnen Baugruppen des Computers werden als Steck-Karten (Steck-Platinen, Plug-in-Boards) ausgeführt und je nach Bedarf in die freien Steckplätze (Slots) des Bussystems (auf der Mutter-Platine oder dem Backplane) zugesteckt. Dies

bedeutet einen vereinfachten, flexibleren Aufbau und leichte Umrüstbarkeit des Computers und damit Fertigungs- und Kostenvorteile ("klassischer" Minicomputer: DEC PDP-11 mit UNIBUS). Die Einsatzbereiche für die Minicomputer weiteten sich mit der Verfügbarkeit von allgemeinen Betriebssystemen und Programmiersprachen aus: Schließlich wurden Minicomputer auch als interaktive Mehrplatz-Systeme zur Softwareentwicklung oder für Ausbildungszwecke sowie als kommerzielle Systeme für die Datenverarbeitungsaufgaben kleiner Unternehmen angeboten.

Die Entwicklung kleiner Rechenanlagen derartiger Leistungsfähigkeit war nur durch die Fortschritte bei der Herstellung integrierter Schaltkreise möglich, die es ermöglichten, immer mehr Halbleiterfunktionen auf einem winzigen Halbleiterplättchen, dem "Chip", zu konzentrieren. Während bis Ende der sechziger Jahre nur Teilfunktionen einer Computerzentraleinheit in einzelnen ICs untergebracht werden konnten, wurde es Anfang der siebziger Jahre durch die vergrößerte Packungsdichte (LSI= Large Scale Integration) der ICs möglich, die kompletten Funktionen einer Zentraleinheit (Steuerwerk und Rechenwerk) auf einem einzigen Chip unterzubringen! Diese Single-Chip-CPUs nannte man "Mikroprozessoren". Der erste Mikroprozessor wurde von der Firma INTEL Ende 1971 unter dem Namen INTEL 4004 auf den Markt gebracht. Er hatte eine Datenwortbreite von 4 Bit und beinhaltete 2300 Transistorfunktionen. (Zum Vergleich: Die neuesten zur Zeit angekündigten Mikroprozessoren haben 32 Bit Datenbreite und beinhalten etwa 500.000 Transistorfunktionen.) Das Entscheidende an dieser Leistung ist nicht so sehr der erreichte Miniaturisierungsgrad, sondern die Tatsache, daß derartige ICs in Massenstückzahlen billig produziert werden können. Beim Zentralspeicher (central memory), neben der CPU die zweite entscheidende Baugruppe von Computern, vollzog sich eine ähnliche Entwicklung: Der bisher verwendete teure Magnetkernspeicher konnte zunehmend durch Halbleiterspeicher in Form integrierter Schaltkreise ersetzt werden.

So kamen bis Mitte der siebziger Jahre komplette Computer auf der Basis von Mikroprozessoren und IC-Halbleiterspeichern auf den Markt, die im Vergleich zu konventionellen Computern ein erstaunlich verbessertes Preis-Leistungsverhältnis aufwiesen. Die darin verwendeten Standard-Mikroprozessoren haben eine Datenwortbreite von 8 Bit (=1 Byte) und können mit einer Adresswortbreite von 16 Bit einen maximal 64 KByte großen Zentralspeicherraum ansprechen (adressieren). Diese CPUs, auch MPUs=Micro Processing Units genannt, verfügen nur über logische Grundoperationen und eine Festkomma-Arithmetik ohne Multiplikation und Division. Nach ihrer Datenwortbreite wurden sie 8-Bit-Mikroprozessoren genannt.

Die Verarbeitungsgeschwindigkeit dieser Prozessoren ergibt sich aus der minimal möglichen Taktzeit und der Zahl der Maschinentakts, die zur Ausführung eines Befehls benötigt werden. Typisch sind Taktzeiten von 1 Mikrosekunde (= 1 millionstel Sekunde entsprechend einer Taktfrequenz von 1 MHz) bis hinunter zu 200 Nanosekunden (= 200 milliardstel Sekunden entsprechend

einer Taktfrequenz 5 MHz). Die Ausführungszeit für einfache Befehle, etwa einer 8-Bit-Addition liegt damit je nach Prozessor bei etwa 1 bis 4 Mikrosekunden.

Die wichtigsten 8-Bit-Mikroprozessoren sind der 8080 der Firma INTEL, der 280 von Zilog, der 6800 der Firma Motorola und der 6502 der Firma MOS Technology. Aufbauend auf diesen Prozessoren wurden verbesserte und schnellere Versionen sowie Nachfolger mit 16-Bit-Datenwortbreite ("16-Bit-Mikroprozessoren") entwickelt, die in ihrer Leistung den bisherigen Minicomputern (z.B. der ILC PDP-11-Serie) vergleichbar sind.

In der Labor- und Testphase befinden sich zur Zeit die ersten 32-Bit-Mikroprozessoren, die die Leistungsdaten bisheriger "großer" CPUs erreichen sollen. Dabei ist es nicht nur wichtig, daß die Arbeiten bisheriger Großrechner teilweise von solchen Systemen übernommen werden können, sondern daß Aufgaben in Angriff genommen werden können, die bisher nicht oder nur mit nicht vertretbarem Aufwand gelöst werden konnten, z.B. bei der zeitkritischen Verarbeitung von Signalen in der Meß- und Kommunikationstechnik.

Einen Überblick über die bekanntesten 8-Bit-Mikroprozessor-Familien gibt folgende Skizze:

	Hersteller			
	INTEL	ZILOG	MOTOROLA	MOS-Technology
Grundversion:	8080	280	6800	6502
Folgeversionen:	8085	2800	6809	
16-Bit-	8086	28000	68000	
Nachfolger:	8088			
	80186			
	80286			
32-Bit-				
Nachfolger:	80386			

In der Regel gibt es die aufgeführten Mikroprozessoren in verschiedenen Geschwindigkeitsversionen: z.B. 280 mit 2,5 MHz-, 280A mit 4 MHz- und 280B mit 6 MHz-Systemtakt.

Die 280-Linie ist der 8080-Familie eng verwandt, da der 280 praktisch eine Weiterentwicklung des 8080 ist und alle 8080-Maschinenprogramme auch auf dem 280 ablaufen können.

Ähnlich wie bei den Minicomputern weitete sich bei den Mikrocomputern der Einsatzbereich ständig aus: Zu den Spezialanwendungen in der Prozeß-, Meß- und Regeltechnik, dem primären Einsatzbereich, kam ein ständig wachsender Markt von zusätzlichen

Spezialanwendungen ("dedizierter" Einsatz) sowie die Verwendung als Miniatur-Universal-Datenverarbeitungsanlage. Aus diesen Anwendungen bildeten sich die folgenden Begriffe heraus: Fobby-, Heim-, Personal-, Kompakt-, Tisch-, Text-, Geschäfts-(Small-Business)-Computer!

Die genannten Prozessorfamilien setzten sich in den verschiedenen Bereichen unterschiedlich durch: An erster Stelle hat sich die 8080-/Z80-Linie behauptet. Für diese Prozessoren steht eine Vielzahl von Betriebs- und Anwendungssoftware zur Verfügung. Der 6802 hat eine hervorragende Stellung auf dem Markt der Heim- und kleineren Geschäftscomputer (durch die Verwendung in den APPLE-II/III- und COMMODORE-PET-/CBM-40xx/CBM80xx-Rechnern). Der 6800 und sein Nachfolger 6809 sind bisher weniger als die beiden vorgenannten auf "artfremde" Gebiete vorgeedrungen.

Die Weiterentwicklung vollzieht sich z.Zt. zweigleisig: Einerseits werden verbesserte 8-Bit-Mikroprozessoren (insbesondere mit gesteigerter Verarbeitungsgeschwindigkeit und größerem Adreßraum) auf den Markt gebracht, andererseits wird die 16-Bit-Prozessorlinie weiterverfolgt. Hier zeichnet sich ab, daß der INTEL 8086 und der MOTOROLA 68000 sich vor dem Z8000 am Markt durchsetzen werden. Während der 68000 von vornherein den Vorteil der größeren Leistungsfähigkeit aufgrund einer internen 32-Bit-Architektur bietet, hat INTEL den 8086 mittlerweile durch die Ankündigung des 80186 mit doppelter Leistung des 8086, des 80286 mit dreifacher Leistung des 8086 und interner 32-Bit-Struktur (und seit neuestem des 80386 mit voller 32-Bit Architektur) gut in ein aufwärtskompatibles Spektrum von Mikroprozessoren eingefügt. Die Firma ZILOG war mit ihren Z8000 offenbar vom Pech verfolgt: Trotz ihres objektiv gesehen sehr leistungsfähigen Prozessors stieg anscheinend ein Teil der Anwender nach Liefer-schwierigkeiten und technischen Problemen beim Z8000 auf die Konkurrenz um, und schließlich entschieden sich Hewlett-Packard und TANDY für die Verwendung des 68000 in ihren neuesten Geräten. IBM wählte den INTEL 8088, eine "abgemagerte" Version des 8086 mit 8-Bit-Datenbus statt 16-Bit-Datenbus für seinen "Personal-Computer", und im Sog der IBM-Entscheidung ("IBM-kompatibel") folgen viele Anbieter mit 8088- oder 8086-basierten Modellen nach.

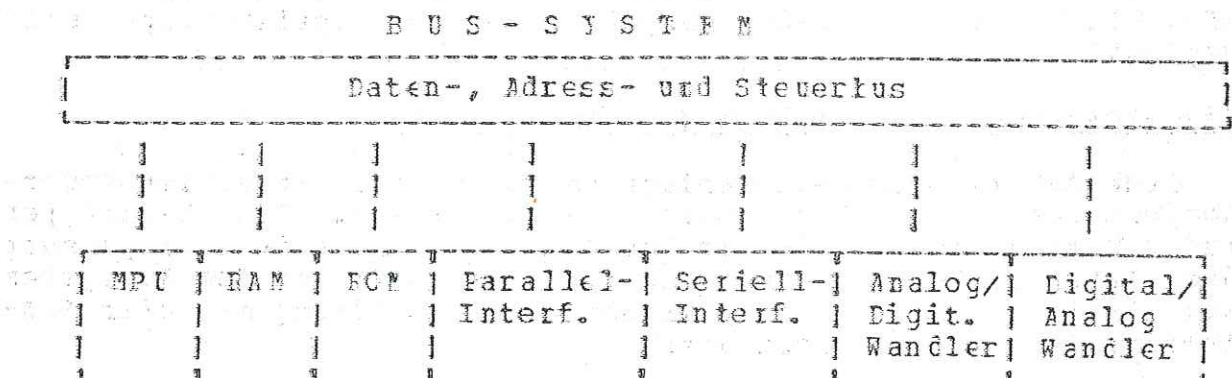
Einige Hersteller haben bereits 32-Bit-Mikroprozessoren angekündigt (INTEL iAPX-432 und 80386, Hewlett-Packard). Hewlett-Packard will bis Ende 1982 einen Mikrocomputer auf der Basis des neu entwickelten 32-Bit-Prozessors vorstellen! Ferner werden für Spezialanwendungen immer leistungsfähigere Single-Chip-Mikrocomputer entwickelt, bei denen nicht nur der Prozessor, sondern auch der Hauptspeicher und Ein-/Ausgabe-Funktionen in einem einzigen Chip enthalten sind.

2. Komponenten und Aufbau eines Mikrocomputers

Mikroprozessor (MPU) und Bussystem

Kernstück eines Mikrocomputers ist der Mikroprozessor (MPU), der funktionell weitgehend der klassischen CPU mit Steuerwerk und Rechenwerk entspricht. Er ist mit den weiteren Baugruppen durch ein Bussystem (vgl. oben) verbunden. Durch das Bussystem werden sämtliche Daten (Daten-Bus), Adressen (Adress-Bus) und Steuerinformationen zur Koordinierung der Busbenutzung (Steuer-Bus) übertragen. Man kann sich das Bussystem als eine Kabelverbindung mit vielen parallelen Leitungen vorstellen. Bei den üblichen Prozessoren werden 8 Leitungen für den Datenbus, 16 Leitungen für den Adressbus entsprechend der Datenwort- bzw. Adresswortbreite und je nach Prozessor unterschiedlich viele Leitungen für den Steuerbus benötigt.

Bild 1: Grundkomponenten eines Mikrocomputersystems



Zentralspeicher (RAM und FCM)

Zum Arbeiten benötigt der Prozessor einen Zentralspeicher für Programme und Daten. Man unterscheidet zwischen Schreib-Lese-Speicher (RAM, Random Access Memory) dessen Inhalt sowohl gelesen, als auch verändert werden kann, und Festwert-Speicher (FCM, Read Only Memory), dessen Inhalt nur gelesen, jedoch nicht verändert werden kann. FCMs werden zur Speicherung von Programmen verwendet, die während der Benutzungszeit des Mikrocomputers unverändert bleiben, etwa Programme, die nach dem Einschalten des Gerätes gleichbleibende Initialisierungsprozesse ausführen, oder Teile des Betriebssystemes, die ständig verfügbar sein sollen. Beim Schreib-/Lese-Speicher unterscheidet man zwischen dynamischem und statischem RAM. Der dynamische RAM verliert die gespeicherte Information innerhalb kürzester Zeit, wenn er nicht elektrisch "aufgefrischt" (Refresh) wird, während der statische RAM die gespeicherte Information erst beim Abschalten des Geräts verliert. Außerdem gibt es bestimmte statische RAMs (CMOS-RAMs), die aufgrund ihres geringen Stromverbrauchs mit Hilfe einer kleinen Batterie "nicht-flüchtig" (non-volatile) gemacht werden können, also die Information wie ein ROM auch nach dem Abschalten behält. Derartiger RAM hat auch besonders kurze Zugriffs-

zeiten (70 Nanosekunden), ist aber leider wesentlich teurer als dynamischer RAM.

Die zur Zeit marktüblichen dynamischen RAMs sind aus 16-KBit-Chips aufgebaut. Man benötigt also 8 ICs dieser Art für 16 KB RAM (1 BYTE = 8 Bit) bzw. 32 ICs für 64 KB. Die Zugriffszeit dieser Speicher liegt bei 200 Nanosekunden. Seit 1981 sind auch dynamische 64-KBit-Chips mit 150 Nanosekunden Zugriffszeit verfügbar, die zunehmend die 16-KBit-Chips ablösen. Damit kann man bei gleichbleibender IC-Zahl, also mit gleichem Platzbedarf und Schaltungsaufwand nun die vierfache RAM-Kapazität unterbringen bzw. den freigewordenen Platz bei gleichbleibender Kapazität anders nutzen. Kleine Mikrocomputer mit 1 Megabyte Zentralspeicher sind bereits auf dem Markt. Da die Adressierungsfähigkeit der meisten (8-Bit-) Prozessoren nur bis 64 KB reicht, nutzt man den zusätzlich verfügbaren Speicher als Sekundärspeicher ('virtuellen' Massenspeicher) für Daten und Programme und erhält dadurch große Leistungsverbesserungen bei Ein-/Ausgabe-intensiven Anwendungen. Diese Entwicklung setzt sich schnell fort: 128-KBit-Chips wurden bereits angekündigt, 1 Megabit-Chips sind geplant!

Ein-/Ausgabe-Schnittstellen (I/O-Interfaces)

Jede Datenverarbeitungsanlage benötigt (bei sinnvollen Anwendungen) eine Daten-Verbindung zur Außenwelt. Die Baugruppen eines Computers, die die in der Regel erforderliche Anpassung der Datendarstellung zwischen der Außenwelt und dem Bussystem vornehmen, nennt man Ein-/Ausgabe-Anpass-Schaltungen- oder E/A-Schnittstellen (I/O-Interfaces).

Parallele Ein-/Ausgabe

Da auf dem Datenbus die Daten als parallele digitale Signale auftreten, ist die einfachste Form der Ein-/Ausgabe die bit-parallele Ein-/Ausgabe. Eine Baugruppe, die diese Aufgabe wahrnimmt, nennt man "Paralleles Interface". Diese Form der Ein-/Ausgabe ist sehr gut brauchbar für den schnellen Signalaustausch mit Geschwindigkeiten bis zu 1 Million Bit/Sekunde (=1 Megabaud) und für Steuerungszwecke, etwa die Auswertung von Sensor-Impulsen und Betätigung von Relais.

Serielle Ein-/Ausgabe

Für bit-parallele Datenübertragung benötigt man der Datenwortbreite entsprechend viele parallele Leitungen. Bei Anwendungen, bei denen Daten über größere Entfernungen übertragen werden müssen, ist die parallele Übertragung vieler Signale teuer. Bei vielen vorhandenen Datenübertragungseinrichtungen ist von vornherein nur ein "Kanal" (entsprechend einer Leitung) in Send- und Empfangsrichtung vorgesehen. Deshalb benötigt man Schaltungen, die die bit-parallele Datendarstellung auf dem Bus in eine bit-serielle umwandeln und umgekehrt. Aus den 8 parallel anfallenden Bits eines 8-Bit-Datenworts werden dann 8 aufeinanderfol-

gende Bits auf einer Leitung und umgekehrt. Zusätzlich müssen Steuerimpulse erzeugt und erkannt werden, um z.B. den Anfang eines Datenworts zu markieren. Baugruppen, die diese Aufgabe durchführen, nennt man "serielle Interfaces". Die erzielbaren Übertragungsgeschwindigkeiten hängen von der Betriebsart ab: Bei sogenanntem Synchronbetrieb können etwa 500000 Bits/Sekunde (=500 kBaud) erreicht werden, bei asynchronem Betrieb etwa 40 kBaud. Man benutzt serielle Interfaces z.B. zum Anschluß an Datenfernübertragungseinrichtungen oder auch zum Anschluß von lokalen Peripheriegeräten mit relativ langsamer Übertragungsgeschwindigkeit (Drucker, Sichtgeräte).

Um die Kopplung von Geräten mit serieller Schnittstelle an Datenfernübertragungseinrichtungen zu erleichtern, gibt es einen international anerkannten Standard, die sogenannte V.24- oder RS-232-Schnittstelle (das erste ist die europäische, das zweite die amerikanische Normbezeichnung). Darin sind neben der Daten-Sende- und Empfangsleitung noch einige Steuersignale zum Betrieb in Verbindung mit Datenfernübertragungseinrichtungen vorgesehen (z.B. "Gerät sendebereit" u.a.). Außerdem sind die zu benutzenden Stecker und Buchsen sowie die Belegung der einzelnen Kontaktstifte (Pins) festgelegt. Durch die große Verbreitung dieser Norm haben sich die Möglichkeiten für den Anwender, Geräte verschiedener Hersteller miteinander zu koppeln, wesentlich verbessert.

Analog-Digital-Wandler (ADC und DAC)

Mit dem Begriff "Computer" meint man heute normalerweise Datenverarbeitungsanlagen, die die Information intern digital (ziffernmäßig) darstellen (Digitalrechner). Oft trifft man jedoch in der Realität auf analog (als physikalische Größe, meist als elektrische Spannung) dargestellte Information. Aus Kostengründen ist es u.U. sinnvoll, die Wandlung eines Analogsignals in digitale Darstellung (oder umgekehrt) nicht mit einem getrennten Gerät vorzunehmen, sondern in den Mikrocomputer zu integrieren. Baugruppen, die diese Aufgabe wahrnehmen, heißen "Analog-Digital-Wandler", kurz "ADC" (Analog-to-Digital-Converter) bzw. "Digital-Analog-Wandler", kurz "DAC" (Digital-to-Analog-Converter). Gängige ADC haben eine "Auflösung" von 12 Bit (d.h. 1/4096 des Messbereichs ist digital unterscheidbar) und führen eine Wandlung in 2⁵ Mikrosekunden durch. Mit einem derartigen Wandler kann man max. etwa 30000 Werte/Sekunde im Computer erfassen. Es gibt Hochgeschwindigkeitswandler mit weit höheren Durchsatz (mehr als eine Million Werte/Sekunde).

3. Allgemeine Peripheriegeräte: Tastatur, Bildschirm, Drucker

Tastatur und Bildschirm (Datensichtgerät)

Abgesehen von Anwendungen, bei denen der Mikrocomputer eine vollautomatische Steuerungsfunktion ausübt, ist normalerweise vorgesehen, daß der Benutzer über eine Tastatur Befehle geben

und auf einem Sichtgerät die Antworten des Computers verfolgen kann. Für den gleichen Zweck gibt es an großen Rechenanlagen Datensichtgeräte (Bildschirm-Terminals), die in vielen Fällen ohne Schwierigkeiten über ein serielles Interface auch an einem Mikrocomputer benutzt werden können. Aus Kostengründen und zur Erzielung eines möglichst kompakten Computersystems (man denke an Platzprobleme in Labors und Büros) kann man wesentliche Funktionen eines Bildschirm-Terminals in den Mikrocomputer verlegen: Die Tastatur wird direkt an eine Parallelschnittstelle angeschlossen. Mit einer speziellen Baugruppe, dem sogenannten Video-Controller, werden die auszugehenden Daten in Steuerungssignale für einen TV-Monitor umgewandelt. Bei Geräten für den Hobby-Einsatz kann man nach Umsetzung des TV-Signals im Notfall statt des benötigten TV-Monitors auch den heimischen Fernseher benutzen.

Besonders wichtig ist bei intensiver Benutzung die Beachtung ergonomischer Gesichtspunkte: Die Tastatur sollte ein angenehmes, unverkrampftes Schreiben ermöglichen und Dauerbelastungen gewachsen sein. Wiederholfunktion für die einzelnen Tasten ist bei vielen Anwendungen sehr hilfreich. Der Bildschirm muß möglichst flimmerfrei und die Zeichendarstellung muß ausreichend scharf und groß sein. Die Zeichen werden auf dem Bildschirm in der Regel als ein Punktraster dargestellt (Matrixdarstellung). Dieses Punktraster muß fein genug sein: Bei einer Auflösung von weniger als 9 mal 9 Punkten sollte die Verwendbarkeit kritisch geprüft werden. Die normale Abbildungskapazität von Sichtgeräten beträgt 24 Zeilen zu je 80 Zeichen, geringere Kapazitäten engen u.U. die Anwendungsmöglichkeiten ein (bei Textverarbeitung wünscht man sich sogar 40 mal 80 Zeichen). Die Verwendung eines Fernsehers als Bildschirm ist wegen mangelnder Bildqualität nicht zu empfehlen.

Drucker

Für die dauerhafte Ausgabe von Ergebnissen braucht man eine Druckmöglichkeit. Als Drucker kommen "Matrixdrucker" in Betracht, die die Zeichen als Punktraster darstellen. Wie bei den Sichtgeräten sollte man ein Auflösungsvermögen von weniger als 9 mal 9 skeptisch betrachten. Wenn auf sehr gute Schriftqualität Wert gelegt wird (etwa bei Anwendungen in der Textverarbeitung), wird meist ein Typenradldrucker eingesetzt. Beim Typenradschreibwerk wird ähnlich wie bei der klassischen Schreibmaschine das ganze Zeichen angeschlagen, jedoch werden keine Typenhebel und kein Kugelkopf benutzt, sondern die Typen sind sternförmig auf einer Scheibe angebracht. (Wegen der Ähnlichkeit zur Form eines Gänseblümchens heißt das Typenrad im Englischen "Daisy Wheel".) Typenradldrucker sind wesentlich langsamer (max. 50 Zeichen/Sekunde) als Matrixdrucker (max. etwa 800 Zeichen/Sekunde) und im Materialverbrauch (Typenräder, Kohle-Farbbandkassetten) und Anschaffungspreis wesentlich teurer als vergleichbare Matrixdrucker. Die Typenradldrucker haben allerdings zur Zeit noch den Vorteil, daß sich ihr Zeichensatz durch Typenradwechsel einfach austauschen läßt. Die Möglichkeit,

Matrixdrucker mit mehreren austauschbaren Zeichensätzen zu versehen oder den Zeichensatz gar frei im Programm definieren zu können, existiert zwar, jedoch sind derartige Geräte auch noch recht teuer.

Massenspeicher: Kassette, Diskette, Magnetplatte

Die auf dem Computer erstellten Programme und Daten möchte man sicher dauerhaft maschinenlesbar aufbewahren, um zu einem späteren Zeitpunkt Programme und Daten weiterbenutzen zu können. Die Benutzung nichtflüchtiger RAMs (vgl. oben) für diesen Zweck ist zwar möglich, jedoch im Normalfall viel zu teuer. Bei den großen Datenverarbeitungsanlagen, bei denen sich natürlich prinzipiell das gleiche Problem stellt, wurden schon vor langer Zeit als Massenspeicher zur Aufbewahrung von (im Vergleich zur Zentralspeicherkapazität) großen Datenmengen Magnetband und Magnetplatten eingeführt. Zum Einsatz in Verbindung mit Mikrocomputersystemen wurden billige Miniaturvarianten dieser Vorbilder entwickelt:

Audio- und Digital-Kassetten

Eine ausgesprochen billige Lösung ist die Benutzung handelsüblicher Kassettenrekorder für die Aufzeichnung: Man benutzt einfach eine serielle Schnittstelle, um einen Pieps-Ton zu steuern und erreicht so Übertragungsgeschwindigkeiten bis zu 600 Bit/Sekunde. In der Praxis bedeutet dies sehr lange Wartezeiten beim Speichern und Lesen von Programmen und Daten. Die Gesamtkapazität einer 30-min-Audio-Kassette liegt bei dieser Anwendung bei etwa 100 KB. Daneben wurden Digital-Kassetten-Laufwerke entwickelt, die eine wesentlich bessere Leistung aufwiesen (etwa 5000 Bit/Sekunde, 250 KB Gesamtkapazität), aber mit Preisen um DM 6000,-- auch unverhältnismäßig teuer waren. Inzwischen gibt es leistungsfähigere Kassettensysteme, die (ohne wesentlich teurer zu sein) bis zu 14 Megabyte Kapazität haben und meist zur Sicherung von Magnetplattendateien (vgl. unten) eingesetzt werden.

Disketten (Floppy Disks)

Die Firma IBM hat schon Anfang der siebziger Jahre zur Ablösung der Lochkarte als Datenträger bei der kommerziellen Datenerfassung die "IBM Diskette" kreiert. Die "Diskette" (= kleine Platte) ist eine kreisförmige, biegsame Plastikscheibe mit einer magnetisierbaren Beschichtung. Sie steckt in einer quadratischen Kartonhülle mit einer Kantenlänge von 8 Zoll (Standard-Diskette) oder 5 1/4 Zoll (Mini-Diskette). Während der Benutzung in einem Disketten-Laufwerk dreht sich die Diskette mit etwa 300 (Mini) bis 360 (Standard) Umdrehungen pro Minute und die Daten werden digital in konzentrischen Ringen, den Spuren, aufgezeichnet bzw. gelesen. Eine Standard-Diskette (IBM 3740) hat eine Kapazität von 256 KB und eine Datenübertragungsrate von 250000 Bit/Sekunde. Durch Verfeinerung der Aufzeichnungsverfahren (Double Density) und Nutzung beider Seiten einer Diskette (Double Side)

kommt man heute auf eine Kapazität von 1,2 Megabyte. Dies entspricht etwa 350 Schreibmaschinenseiten. Aufgrund der Biegsamkeit der Diskette hat sich in den USA der Ausdruck "Floppy Disk" durchgesetzt, auch "Flexible Disk" wird benutzt, während in Deutschland die (IBM-)Bezeichnung "Diskette" üblich ist.

Die Diskette ist nicht nur aufgrund ihrer Leistungsdaten eine ideale Ergänzung für Mikrocomputer, sondern auch ein ausgezeichnetes Medium für Datenaustausch. Wer seine Daten auf einer Standard-Diskette im IBM 3740-Format aufzeichnet, kann damit rechnen, daß diese Daten weltweit auf großen und auf kleinen DV-Anlagen gelesen werden können. Der Datenaustausch zwischen unterschiedlichen Mikrocomputersystemen über Minidisketten ist jedoch praktisch nicht möglich, da fast alle Hersteller ein anderes Aufzeichnungsverfahren verwenden.

Als Mitte der siebziger Jahre kostengünstige Diskettenlaufwerke und 8-Bit-Mikrocomputer gleichzeitig verfügbar wurden, war die Voraussetzung geschaffen, komplette kleine Datenverarbeitungsanlagen in großen Stückzahlen zu verkaufen. Dadurch sanken die Preise für Diskettenlaufwerke noch erheblich. Man kann ohne weiteres sagen, daß der Erfolg der Mikrocomputer als Miniatur-Universal-Datenverarbeitungsanlagen erst durch die Diskette möglich war.

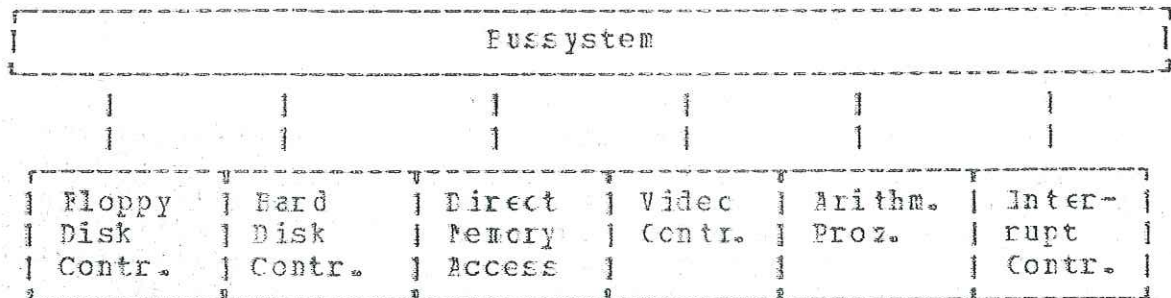
Magnetplatten (Hard Disks)

In den letzten Jahren gelang es, auf der Basis der für Magnetplattenspeicher an Großrechnern entwickelten Techniken (Winchester Technology), Miniatur-Festplattenspeicher zu bauen, die in den äußeren Abmessungen der 8- bzw. 5.25-Zoll-Diskettenlaufwerken entsprechen. Diese Platten besitzen jedoch die vielfache Kapazität (ggw. bis zu 50 MB) und etwa die 10-fache Übertragungsgeschwindigkeit wie vergleichbare Disketten. Im Gegensatz zu den "Floppy Disks" werden sie als "Hard Disks" bezeichnet. Abgesehen von ihren hervorragenden Leistungsdaten hatten sie bisher den Nachteil, daß sie fest montiert, also nicht auswechselbar sind. Daher sollte bei Benutzung einer solchen Hard Disk zusätzlich ein leistungsfähiges Kassetten- oder Diskettenlaufwerk zur Datensicherung ("Back-Up") vorgesehen werden. Seit neuestem (National Computer Conference, Juni 1982) gibt es jedoch auch Wechselplatten im 5.25-Zoll-Format.

4. Steuereinheiten und Hilfsprozessoren im Mikrocomputer

Der zentrale Mikroprozessor innerhalb eines Mikrocomputers ist oft nicht in der Lage, alle anfallenden Aufgaben allein adäquat zu lösen. Man hat deshalb versucht, die Gesamtleistung eines Mikrocomputersystems durch den Einsatz spezialisierter Steuereinheiten und Hilfsprozessoren zu verbessern. Im folgenden werden einige wichtige Beispiele genannt.

Bild 2: Steuereinheiten (Controller) und Hilfsprozessoren

Steuereinheiten für Plattenlaufwerke (Disk-Controller)

Diskettenlaufwerke und Festplatten benötigen eine komplizierte und zeitkritische Steuerung, um Daten korrekt aufzuzeichnen. Es hat sich als günstig erwiesen, einen Teil der erforderlichen Steuerlogik nicht in die Laufwerke einzubauen, sondern im Mikrocomputer selbst als programmierbares "intelligentes" Interface zwischen Bussystem und Laufwerk unterzubringen (Disk-Controller). Hier hat sich eine wesentliche Kostenreduktion dadurch ergeben, daß die Controllerfunktion für Diskettenlaufwerke in einem IC konzentriert werden konnte. Die Controller für Hard Disks müssen zur Zeit noch aus vielen diskreten Komponenten aufgebaut werden und sind entsprechend teuer (etwa DM 5000,--). In Kürze sollen die ersten Single-Chip-Controller für Hard Disks verfügbar sein. Damit wird sich der Einsatz dieser Geräte verbilligen und die Massenspeicherkapazität bei Mikrocomputern weiter verbessern.

Die Eigenschaften eines Disk-Controllers sind entscheidend für die Ausnutzung der angeschlossenen Laufwerke. So kann u.U. nur durch Verwendung eines verbesserten Controllers (z.B. Möglichkeit zur Verarbeitung doppelter Aufzeichnungsdichte, Spur-Pufferung) die nutzbare Kapazität und Übertragungsgeschwindigkeit eines Floppy- oder Hard-Disk-Laufwerkes erheblich gesteigert werden.

Videoc-Controller

Video-Controller haben die Aufgabe, digitale Daten so in Steuersignale für Video-Monitoren umzusetzen, daß die Darstellung von Zeichen oder Graphiken ermöglicht wird. Für diese Aufgabe sind programmierbare integrierte Schaltkreise verfügbar.

Direkter Speicherzugriff (DMA-Controller)

Bei sehr hohen Datentransferraten (etwa bei Benutzung von Hard Disks) reicht die Verarbeitungsgeschwindigkeit der CPU nicht aus, um die anfallenden Datenmengen zu verarbeiten. Es gibt Steuerbausteine, die derartige Daten mit höchster Bus-Geschwindigkeit direkt in den Zentralspeicher schreiben bzw. aus ihm lesen (DMA=Direct Memory Access). Während dieses zeitkriti-

schen Vorgangs übernimmt der "DMA-Controller" die Bussteuerung von der CPU und sperrt alle anderen Bus-Teilnehmer von der Benutzung aus.

Arithmetikprozessoren

Ein 8-Bit-Mikroprozessor kann arithmetische Berechnungen nur durch aufwendiges "Emulieren" von Operationen der erforderlicher Genauigkeit durchführen. Die erreichte Rechenleistung reicht für viele Anwendungen nicht aus. Mit spezialisierten Arithmetikprozessoren (z.B. AM 9512 der Fa. AMD mit 64-Bit-Gleitkomma-Arithmetik) erreicht man bei numerischen Aufgabenstellungen Rechenzeitverkürzungen.

FFT-Prozessor für Fast-Fourier-Transformationen

Ein besonders häufiger Anwendungsfall bei der Analyse von Meß-Signalen ist die Fast-Fourier-Transformation. Daher wurden spezialisierte Baugruppen für diese Aufgabe entwickelt.

Interrupt-Controller

Um zeitkritische Anforderungen (etwa bei der Erfassung von Meßwerten) erfüllen zu können, aber auf der anderen Seite den Computer nicht auf den Eintritt derartiger Ereignisse warten lassen zu müssen, sondern ihn für andere Aufgaben benutzen zu können, wurde die "Unterbrechung" (Interrupt) erfunden: Aufgrund eines Hardware-Signal bricht die CPU die Bearbeitung des gerade laufenden Programms ab und fährt mit einem für diesen Fall vorgesehenen Spezialprogramm, der "Interrupt Service Routine", fort. Ist dieses Programm beendet, wird wieder das ursprünglich bearbeitete Programm fortgesetzt. Die Möglichkeiten der Interrupt-Behandlung sind bei den Mikroprozessoren unterschiedlich gut. Zur Implementierung von flexiblen und leistungsfähigen Interruptsystemen sind programmierbare "Interrupt Controller" entwickelt worden.

5. Standard-Bussysteme für Mikrocomputer

Für Anwender, die eine bestimmte Aufgabe mit dem Mikrocomputer lösen wollen, und die davon ausgehen können, daß keine Erweiterungen der Aufgabenstellung erfolgen wird, reicht ein Mikrocomputersystem aus, das komplett auf einer Platine untergebracht (Single Board Computer) und dessen Bussystem von außen nicht zugänglich ist. Dagegen werden Anwender, die Erweiterbarkeit ihres Systems anstreben, darauf achten, daß das Bussystem eine ausreichende Reserve an freien Steckplätzen (Slots) besitzt. Besonders günstig ist es, wenn nicht nur die Steckkarte eines Herstellers verwendet werden können. Deshalb ist es interessant, inwieweit auf diesem Gebiet herstellerunabhängige Standards existieren: In Deutschland hat sich als de-facto-Standard das Bussystem "ECB" (auf der Basis des Europa-

Karten-Formats für Standard-Platinen) der Firma KONTECON etabliert. Dieser Bus ist auf der Z80-Mikroprozessor zugeschnitten. Mehrere Firmen bieten in Deutschland preisgünstige Baugruppen für dieses Bussystem an.

Ein verbreiteter herstellerunabhängiger Standardbus ist der amerikanische IEEE-696-Bus, der auf der Basis des bekannten S-100-Bus entstand. Der IEEE-696 ist sowohl für 8- als auch 16-Bit-Mikroprozessorsysteme brauchbar, da er alternativ 8-Bit- und 16-Bit-Datentransfers zuläßt, außerdem besitzt er einen auf 24 Bit erweiterten Adressbus, so daß 16 Megabyte Zentralspeicherplatz direkt adressierbar sind. Für den IEEE-696 gibt es sehr viele fertige Baugruppen und Prozessorkarten mit verschiedenen 8-Bit und 16-Bit-Mikroprozessoren. Auf der anderen Seite muß kritisch bemerkt werden, daß diese Flexibilität mit einem hohen technischen Anpass-Aufwand (und damit Kosten) erkaufte werden muß. In vielen Fällen (z.B. bei Beschränkung auf den Z80, 64 KB RAM und Standard-Peripherie) ist wahrscheinlich der ECB-Bus die billigere und technisch effizientere Lösung.

Ein weiterer Busstandard für Mikrocomputersysteme ist der IEEE-796 auf der Basis des MULTIBUS der Firma INTEL.

Als IEEE-896 ("Futurebus") ist die erste wirklich herstellerunabhängige und zukunftsorientierte Bus-Norm für Multiprozessor-Systeme in Vorbereitung. Physisch baut diese Norm auf dem Europakarten-Format (Doppelpack und Vielfache von Doppelpack) auf, das damit eine weltweite Anerkennung und Verbreitung erfahren dürfte.

Schließlich ist der IEEE-488-Bus (IEC-Bus) erwähnenswert. Dieses Bussystem ist nicht als internes Bussystem innerhalb eines Mikrocomputers gedacht, sondern als Bus zwischen Mikrocomputer und mehreren Peripheriegeräten, insbesondere Meßgeräten. Er stammt ursprünglich von der Firma Hewlett-Packard (HP-Interface Bus: HP-IB) und wurde dann von anderen bekannten Herstellern übernommen z.B. WANG und TEKTRONIX. Weitestе Verbreitung fand er schließlich durch die Firma COMMODORE, die ihn bei ihren PET- und CBM-Mikrocomputern als Standardschnittstelle vorsah. Ein Nebeneffekt dieser Entwicklung war, daß in exklusiven Meßgeräteparks auf einmal ein Billiggerät, nämlich der COMMODORE, als Steuereinheit auftauchte - zu einem Bruchteil des Preises, der bis dahin für Tischcomputer mit "IEC-Bus-Controller" gezahlt werden mußte.

Zum PL/I-Datentyp DECIMAL FLOAT(33)

S. Zerkendörfer

Es ist bekannt, daß Gleitkommazahlen von unserem Rechner intern in normierter hexadezimaler Darstellung codiert und bearbeitet werden; die Sprache PL/I sieht binäre und dezimale Gleitkommazahlen vor. Aus dieser Diskrepanz erklären sich wohl die völlig unzulängliche Realisation bzw. die geradezu irreführenden Bezeichnungen der Gleitkomma-Arithmetik in dieser Sprache - zumindest bei derzeitigen Compilern. Hierzu wird im folgenden ein Beispiel aufgeführt.

Laut Handbuch ist bei DECIMAL FLOAT eine maximale Genauigkeitsangabe von 33 Dezimalen zugelassen. (Zitat zu "precision": It specifies the minimum number of significant digits to be maintained.) Mit dem Programmausschnitt

```
PUT SKIP LIST(.995682444457782673143050248604820E-59);
PUT SKIP LIST(.995682444457782673143050248604829E-59);
```

lasse ich zwei 33-stellige dezimale Gleitkommazahlen (als Grenzer einer Folge von 10 "aufeinanderfolgenden" Zahlen aus dem nicht gerade zufällig gewählten Intervall von 16**-49 bis 10**-59) ausdrucken; der Optimizing Compiler liefert als Druckausgabe

```
9.95682444457782673143050248604817E-60
9.95682444457782673143050248604817E-60
```

Aufgrund dieses Beispiels läßt sich schnell zeigen: es gibt 10 (in Worten: zehn) paarweise verschiedene 33-stellige Gleitkommazahlen, die in der Druckausgabe mit 33-stelliger Ziffernfolge alle dasselbe Druckbild haben, dessen Wert zudem nicht einmal in dem von diesen Zahlen aufgespannten Intervall liegt! Die derzeitigen Compiler erfüllen die für den Datentyp DECIMAL FLOAT(33) geforderte Genauigkeit nicht einmal bezüglich der Operationen Eingabe/Speicherung/Ausgabe, weitergehende Untersuchungen zur Genauigkeit der eigentlichen Arithmetik sind sinnlos.

Allgemein bekannt ist, daß Gleitkommazahlen in hexadezimaler Darstellung realisiert sind, und zwar mit 6, 14 oder 28 Hexadezimalen. Hieraus erklärt sich nicht (zumindest nicht vollständig) die aufgezeigte Unzulänglichkeit, bei kritischen Betrachtungen muß auch die Genauigkeit (besser: die Ungenauigkeit) der von jeweiligen Sprachübersetzter benutzten Ein/Ausgabe-Umwandlungsroutinen Berücksichtigung finden.

Störungen im Rechenbetrieb

K. Feichel

Die folgende Tabelle gibt Auskunft über Ausfälle der Rechenanlage und Störungen im Rechenbetrieb in den Monaten Januar bis Juni 1982. Im folgenden bedeuten:

- Betriebszeit - vorgesehener planmäßiger Einsatz
 der Rechenanlage
 Störzeit - Ausfallzeit der Rechenanlage
 (Hardware, Software, Klima, Strom),
 prozentual zur Betriebszeit
 IPL - System-Neustart (Initial Program Load)
 mittlerer
 Fehlerabstand - (Betriebszeit - Störzeit) / IPL

	Jan.	Febr.	März	April	Mai	Juni
Störzeit in %	1.0	0.2	3.1	1.2	1.2	5.4
außerplanmäßige IPLs	10	8	60	15	14	38
mittlerer Fehlerabstand in Stunden	53.6	30.0	10.9	33.6	47.4	16.7

* Plattenfehler Memorex

** IBM-Systemfehler

inform-Quiz

E. Sturm

Das letzte inform-Quiz war anscheinend so schwierig, daß manche zwar nach dem Abtippen des Programms vom Rechner erfuhren, daß als Lösung die Zahl 25 herauskam, aber immer noch nicht wußten, warum! Mit Hilfe des Bedingungspräfix (CHECK): und der FLOW-Option des CHECKOUT-Compilers konnte man sich aber dann Klarheit verschaffen.

Verwirrend ist die Verwendung jeder Menge von gleichen Variablennamen, angefangen bei WHAT_IS_WHAT (der Name einer Prozedur ist nur dann innerhalb bekannt, wenn RECURSIVE angegeben wurde). Benutzt man Namen von Unterstrukturen bzw. Namen von Elementen einer Struktur mehrmals auf verschiedenen Stufen oder in mehreren Strukturen, so nimmt der II/I-Compiler an, daß man im weiteren Verlauf des Programms die Variablennamen möglichst genau

qualifiziert hat: Im Beispiel sind also A und A.A Strukturen und A.A.A ein Elementarname. F ist die einzeln deklarierte Variable, sonst hätte man A.A.F oder A.F schreiben müssen. Damit ist also klar, was an das Unterprogramm WHAT_IS_WHAT übergeben wird. Im Unterprogramm selbst sind dann wiederum As und Bs deklariert, die nun also Vorrang in der Gültigkeit haben vor den außerhalb deklarierten.

Nun zur ersten Zuweisung: Mit A kann nur die innerhalb deklarierte Struktur gemeint sein, B ist auch innerhalb deklariert, kann also nicht das außerhalb deklarierte B sein, die so gedeutete Strukturzuweisung wird also von PL/I aufgegliedert in die folgenden beiden Elementarzuweisungen:

```
A.F = A.F + A.B; ==> 6
A.A = A.A + A.B; ==> 13.
```

Z ist ein von B erzeugtes "dummy argument" (wegen der Klammern um B), X ist auf Grund der Parameterübergabe derselbe Speicherplatz wie B, also wird Z gleich 16 und in der nächsten Zuweisung X (also B) gleich 24. Y ist aus den gleichen Gründen dasselbe wie B, also $24+1=25$!

An diesem Beispiel sollte hinreichend deutlich geworden sein, was eine schlechte Wahl der Variablenamen für die Übersichtlichkeit (und wahrscheinlich auch Korrektheit) eines Programms bewirken kann. Zum Thema "gute Variablenamen" noch ein Beispiel, bei dem ich lieber gleich verrate, daß i herauskommt, ehe es jemand ablocht. Es eignet sich vor allem gut zum lauten Lesen.

STCTTFF:

```
PROCEDURE OPTIONS (MAIN);
```

```
DECL (TC, II, BY, THEN) BIN FIXED (31);
```

```
TO = 2;
```

```
IF = TO / TO;
```

```
BY = TO * TO;
```

```
DO TC = TO TO TO BY BY, TO BY TO TO BY, BY BY BY TO TO,
    BY TO BY BY TO;
```

```
IF IF THEN THEN = IF; ELSE THEN = TC;
```

```
IF = -11;
```

```
END;
```

```
PUT LIST (THEN);
```

```
END STCTTFF;
```


Sprechzeiten

Die Mitarbeiter des Rechenzentrums haben über ihre Beratungstätigkeit hinaus auch verschiedene andere Aufgaben im Rechenzentrum wahrzunehmen, die es, insbesondere bei der Verteilung des Rechenzentrums auf zwei verschiedene Gebäude, mit sich bringen, daß sie nicht immer in ihren Büros unter der angegebenen Telefonnummer erreichbar sind. Darum seien hier noch einmal - aber auch wegen der personellen Veränderungen der letzten Monate - Zeiter für eine erste Kontaktaufnahme genannt:

Name	Zimmer	Tel.	Zeit
Penduhn-Mertz	H 13	2682	di 15-16
Bosse	E 102	2461	n. V.
Fickenscheidt	E 31	2673	fr 14-15
Coorkotte	E 01	2672	mo-do 8.30-11.30 13.30-15.30
Dr. Exner	E 34	2686	mo 10-11
Dr. Held	E 104	3791	n. V.
Dr. Kamp	E 11	2474	mo, do 11-12
Kaspar	E 14	2468	mi 11-12
Kisker	H 13	2682	do 14-15
Mecke	E 02	2466	mo-fr 9-12, 14-16
Dr. Mertz	E 16	2683	mi 11-12
Iasthaus-Freund	E 26	2476	mi 15-16
Nabrotzki	E 14	2468	mi 8.30-11.30
Mienhaus	E 02	2483	di 13-14
Dr. Pudlatz	E 15	2472	di 11-12
Reichel	E 111	2481	mi 9.30-11.30
Dr. Slaby	E 110	2681	n. V.
Spellmann	E 12	2473	mi 11-12
Dr. Steinhausen	E 36	2464	mo 10-11
Dr. Zörkendörfer	E 01	2471	do 10-11

Mitarbeiter der Abteilung Systemsoftware erreichen Sie mo-fr, 9-12, 14-17 mit Ausnahme von mo 9-12, do 14-17 unter der Telefonnummer 2465.

Zeiter für die Kontaktaufnahme mit der Problemberatung, insbesondere für die Anmeldung von Rechenvorhaben sind mo, mi, fr 10-11 im Gebäude Hittorfstr. 27.

Außer den hier genannten gesicherten Zeiter für eine erste Kontaktaufnahme besteht selbstverständlich die Möglichkeit, weitere Gesprächstermine mit den einzelnen Mitarbeitern zu vereinbaren.

Gebäude: H = Hittorfstraße 27
E = Eirsteistraße 60