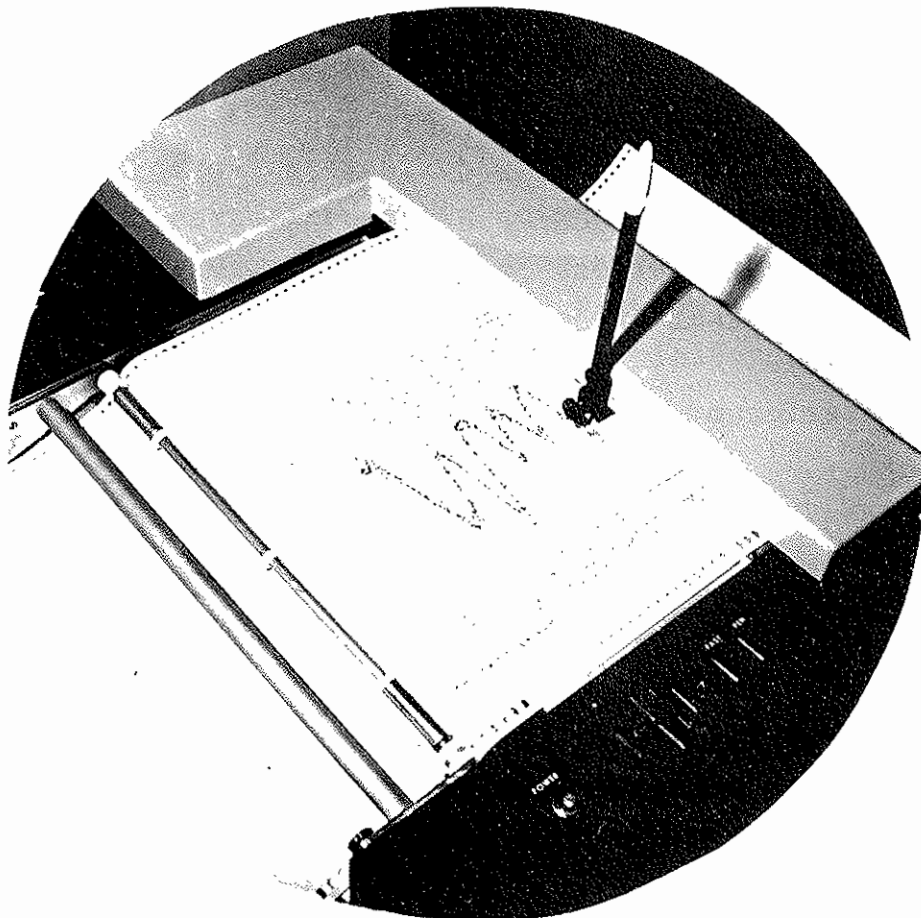

Interface

herausgegeben von
der Abt. Hybridrechenanlage
des Rechenzentrums
der Technischen Hochschule Wien

Nummer 2
Februar 1975



EAI 140 DATAPLOTTER

INHALTSVERZEICHNIS

	Seite
Einleitung	4
Technischer Betrieb	6
Kurse	7
Steuerkarten in JCS/MP 6	9
Verwendung der Massenspeicher im Betriebssystem JCS/MP 6	10
FORTRAN IV Compiler JCSEFOR	12
Exekutieren von Programmen mit sehr großem Kernspeicherbedarf Segmentieren in JCS/MP 6	14
IPP1 - Interpolations- und Plotter- Programmpaket für eindimensionale Probleme	17
Benutzerforum	20

Eigentümer, Herausgeber, Verleger: Rechenzentrum der Technischen Hochschule Wien, Abteilung Hybridrechenanlage, Vervielfältigung: Österreichische Hochschülerschaft Technik, für den Inhalt verantwortlich: Dipl.Ing. W. Kleinert, alle: Gußhausstr. 27-29
1040 Wien.

In memoriam Erich Bukovics

Der schmerzliche Verlust, den die Technische Hochschule Wien durch das Ableben von o.Prof. Dr. Erich Bukovics schlagartig erlitt, ist auch für das interfakultäre Rechenzentrum, dessen Vorstand Bukovics durch viele Jahre angehörte, von großer Tragweite. Hat sich doch Bukovics dem Aufbau, dem Betrieb und der Erweiterung der Hybridrechenanlage mit großer Hingabe gewidmet, wie er allen Aufgaben in führender Stellung im Kollegium, als mehrjähriger Rektor und Prorektor, im Rechenzentrum und Institut hohe Sorgfalt beigemessen und die Lehre der Mathematik stets wahrhaft vorbildlich gestaltet hat.

Alle, die mit Bukovics in fachlichem oder organisatorischem Kontakt standen, bewunderten neben seinen wissenschaftlichen und fachlichen Leistungen seinen klaren Verhandlungsstil und seinen Blick für das Wesentliche jeder Problematik. Wir werden Erich Bukovics als leuchtendes Vorbild in bleibender Erinnerung behalten.

Vorstand, Leitung und Mitarbeiter
des Interfakultären Rechenzentrums der
Technischen Hochschule Wien

EINLEITUNG

Der Schwerpunkt dieser Nummer des Interface liegt auf dem Aspekt der graphischen Datendarstellung speziell am Dataplotter. An der Hybridrechenanlage steht für Forschung und Lehre ein kleiner Incrementalplotter, EAI 140, zur Verfügung (siehe Titelfoto). Es handelt sich dabei um ein Gerät, das aufgrund seiner einfachen Ausführung zwar nicht als Präzisionsplotter angesprochen werden kann, aber für eine mehr qualitative Ergebnisdarstellung den meisten Ansprüchen gerecht wird.

SPEZIFIKATIONEN DES EAI 140 DATAPLOTTER

Schrittlänge: 0.005 inch = 0.127 mm
Geschwindigkeit: 450 Schritte/sec
Papier: gefaltete Endlosformulare
8 1/2 x 11 inch
Papierbreite erlaubt Zeichnung im
A3-Format

Prinzipiell müssen auf einem Incrementalplotter alle zu zeichnenden geometrischen Figuren auf Polygonzüge zurückgeführt werden. Die vom Hersteller gelieferte Basissoftware ermöglicht das Zeichnen dieser Elementarschritte, das Skalieren, sowie das Zeichnen und Beschriften von Achsen. An der Hybridrechenanlage wurde diese Basissoftware standardisiert (sie ist z.B. mit CALCOMP kompatibel) und läuft unter dem Betriebssystem JCS/MP 6. Plotterjobs werden hauptsächlich im Hintergrundbetrieb zu hybriden oder anderen Anwendungsprogrammen gerechnet und stellen i.a. keine zusätzliche Belastung des Rechenbetriebes dar.

Derzeit wird gemeinsam mit der Abteilung Digitalrechenanlage des Rechenzentrums daran gearbeitet, Daten, die am Großrechner erstellt worden sind, standardisiert in gepackter binärer Form auf Karten zu stanzen, um sie an der Hybridrechenanlage in Plotterprogrammen weiterverarbeiten zu können.

Die vom Hersteller mitgelieferte Basissoftware ermöglicht prinzipiell die Lösung aller Benutzerprobleme. Wie bei einer Rechenanlage, bei der alle Möglichkeiten mit Hilfe einer Assemblersprache ausschöpfbar sind, die Verwendung einer höheren Programmiersprache jedoch die Benützung des Rechners aber wesentlich erleichtert, so erhebt sich

auch hier die Frage, wie man die Verwendung des Plotters dem Benutzer ohne Kenntnisse der Basissoftware ermöglichen kann. Einen ersten Schritt in dieser Richtung stellt das in dieser Nummer beschriebene Programmpaket dar, das im Teamwork von Angehörigen des Institutes für Numerische Mathematik (Dipl.Ing. C.W. Überhuber), von Angehörigen des Hybridrechenzentrums und von Studenten der Mathematik erstellt wurde.

Die Technische Hochschule Wien veranstaltet vom 26.5. bis 28.5.1975 ein

SEMINAR ÜBER DEN EINSATZ VON HYBRIDRECHEN - SYSTEMEN IN WISSENSCHAFT, FORSCHUNG UND TECHNIK

Dieses Seminar soll interessierte Fachleute der Hochschulen und der Wirtschaft aus Österreich, der Schweiz, dem süddeutschen Raum und den östlichen Nachbarstaaten zusammenführen. Ziel dieses Seminars ist es, das Anwendungsspektrum von Hybridrechnern im allgemeinen und das des an der hiesigen Hochschule installierten im besonderen aufzuzeigen.

Anlässlich dieses Seminars sind eine Reihe von Gastvorträgen ausländischer Wissenschaftler geplant. Unter anderem sind Vorträge und Demonstrationen am Rechner vorgesehen, die den vorteilhaften Einsatz des Hybridrechners an Anwendungsbeispielen aus dem Gebiet der Biometrik und der Verfahrenstechnik aufzeigen sollen.

Die organisatorische Vorbereitung dieses Seminars, das von der Firma EAI unterstützt wird, haben der Arbeitsbereich Regelungstheorie und Hybridrechentechnik des Instituts für Technische Mathematik und die Abteilung Hybridrechenanlage des interfakultären Rechenzentrums der Technischen Hochschule Wien übernommen.

Interessenten werden gebeten, möglichst bald schriftlich oder telephonisch mit der Hybridrechenanlage Kontakt aufzunehmen (Frau Irmgard Husinsky, Abt. Hybridrechenanlage des Rechenzentrums der Technischen Hochschule Wien, Gußhausstr. 27-29, A-1040 Wien, Tel. (0222) 65 37 85/832 DW).

ÖFFNUNGSZEITEN DER HYBRIDRECHENANLAGE:

Montag - Freitag

8 - 20 Uhr

NEUER LOCHER:

Ab Februar 1975 wird auf dem Gang vor der Hybridrechenanlage (1040 Wien, Gußhausstr. 27-29, 4. Stock, Zimmer 1404/05) ein Locher für 26-Code aufgestellt sein. Für die Benützung dieses Lochers werden Reservierungslisten ausgehängt, in die Benützer Reservierungen unter ihrer Jobnummer eintragen können. Der Locher im Rechenraum locht 29-Code. Diese beiden Codes können an der Hybridrechenanlage nicht in einem Job gemischt werden. Bis auf weiteres werden Programme, die im 26-Code gelocht sind, vorrangig behandelt. Im Closed-Shop-Betrieb können jedoch auch Programme im 29-Code eingegeben werden. Die Eingabe ist je nach dem Code in die entsprechende Eingabelade zu legen. Die Datenkarten müssen in demselben Code wie die Programmkarten gelocht sein.

CLOSED-SHOP-BETRIEB:

Montag bis Freitag

Eingabe: 8 Uhr, 12 Uhr, 17 Uhr

Kurzjobs mit einer maximalen Rechenzeit von 5 Minuten werden nach Möglichkeit innerhalb von ein bis zwei Stunden nach der Eingabe gerechnet.

Bei Jobs mit maximaler Rechenzeit von 10 Minuten erfolgt die Ausgabe spätestens bei Beginn der nächsten Eingabe. Längere Jobs sowie umfangreiche Plotterjobs werden im Laufe des Tages, bis spätestens 20 Uhr, gerechnet.

KURSE

Im Sommersemester 1975 werden folgende Kurse abgehalten werden:

RH7 SOFTWAREUNTERSTÜTZUNG FÜR DIE BENÜTZUNG DES PACER 100
ALS PLOTTER SYSTEM
Einführung in das Programmpaket IPP1
Ein Vortrag mit anschließender Diskussion
Termin: 11. 4. 1975 14 Uhr
Ort: 1040 Wien, Gußhausstr. 27-29, 4. Stock
Seminarraum der mathematischen Institute (Zimmer 1415)
Vortragender: Dipl.Ing. C.W. Überhuber

Vom Institut für technische Mathematik werden folgende Kurse abgehalten:

AH1 PROGRAMMIEREN VON ANALOGRECHNERN
AH2 PROGRAMMIEREN VON HYBRIDEN ANALOGRECHNERN
mit praktischen Übungen
Termin: 17.2.1975 bis 21.2.1975
Beginn: 17.2.1975 9 Uhr
Ort: 1040 Wien, Gußhausstraße 27-29
4.Stock Seminarraum der mathematischen Institute
(Zimmer 1415)

Für das Sommersemester 1975 sind folgende Kurse geplant:

AH3 VERFAHREN DER HYBRIDEN PROGRAMMIERUNG
AH4 HYBRIDE PROGRAMMIERSPRACHEN
Termin: im April oder Mai
Für das Verständnis der Kurse AH3 und AH4 sind Grundkenntnisse der Programmierung von Analogrechenanlagen erforderlich.
Einladungen zu diesen Kursen werden rechtzeitig an alle jene erfolgen, die bereits einen Programmierkurs besucht haben (AH1,AH2). Interessenten, die noch keinen Kurs besucht haben, werden gebeten, ihren Namen und ihre Adresse bekanntzugeben bei:

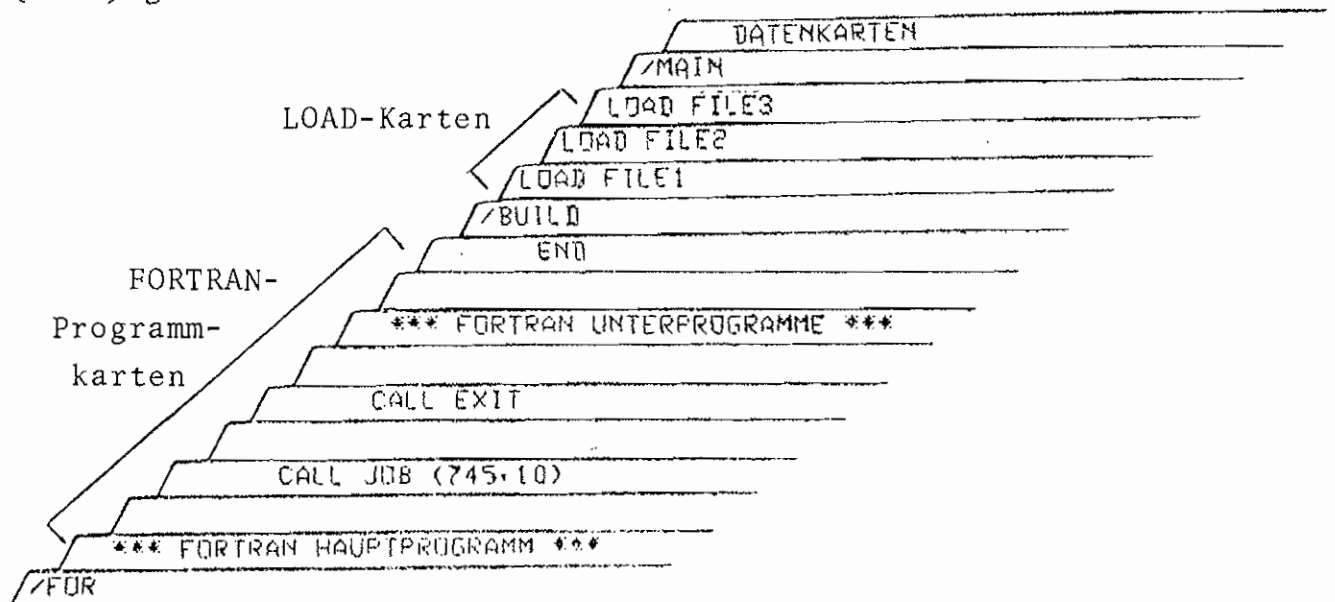
Dipl.Ing. H.P. Petrovsky (Klappe 830) oder
Dipl.Ing. F. Rattay (Klappe 899),
beide Institut für Technische Mathematik, Gußhausstraße
27-29, 1040 Wien.
Dort können auch nähere Auskünfte über die Kurse AH1, AH2,
AH3, AH4 eingeholt werden.

Im Wintersemester 1974/75 wurden an der Hybridrechenanlage folgende Kurse abgehalten:

- RH2 BENÜTZUNG DES BETRIEBSSYSTEMS JCS/MP 6
Es wurden das Operating, die Verwendung der Systemprogramme, das Troubleshooting ausführlich behandelt. Anschließend wurde auf die interne Struktur des Betriebssystems eingegangen. Informationsblätter über die Benutzung des Betriebssystems sind erhältlich.
- RH3 BESONDERHEITEN IN FORTRAN AN DER HYBRIDRECHENANLAGE
Es wurde ein Vortrag gehalten. Bei Bedarf wird dieser Vortrag jederzeit wiederholt.
Informationsblätter sind erhältlich.
- RH4 HYBRIDE FORTRAN PROGRAMMIERUNG
Die Hybrid Linkage Subroutines wurden einzeln ausführlich besprochen und ihre Anwendung wurde an mehreren Beispielen ausführlich demonstriert. Ein ganzer Vortrag diente einer Einführung in CSMP (Continuous System Modelling Program), das durch digitale Simulierung analoger Vorgänge Parameterabschätzungen für Hybridprogramme gestattet.
- RH6 EAI-ASSEMBLER
Es wurden die Befehle des EAI-Assembler vorgetragen und Übungsbeispiele durchgearbeitet.
Ausführliche Unterlagen über den EAI-Assembler werden demnächst fertiggestellt und an der Hybridrechenanlage erhältlich sein.

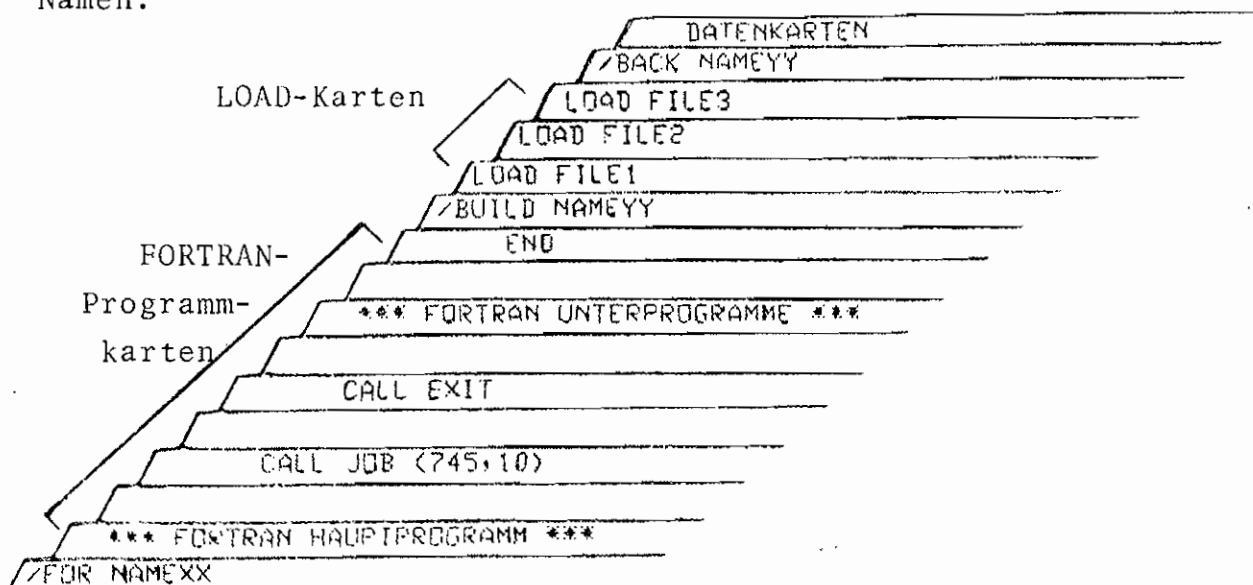
STUEKARTEN IN JCS/MP 6

Beispiel: Steuerkarten für ein FORTRAN-Programm, das auf Level 1 (Main) gerechnet werden soll.



Nach der /BUILD-Karte können maximal 5 LOAD-Karten angeführt werden, auf denen je ein Name eines externen Plattenfiles angegeben werden kann, das zum Programm dazugeladen werden soll.

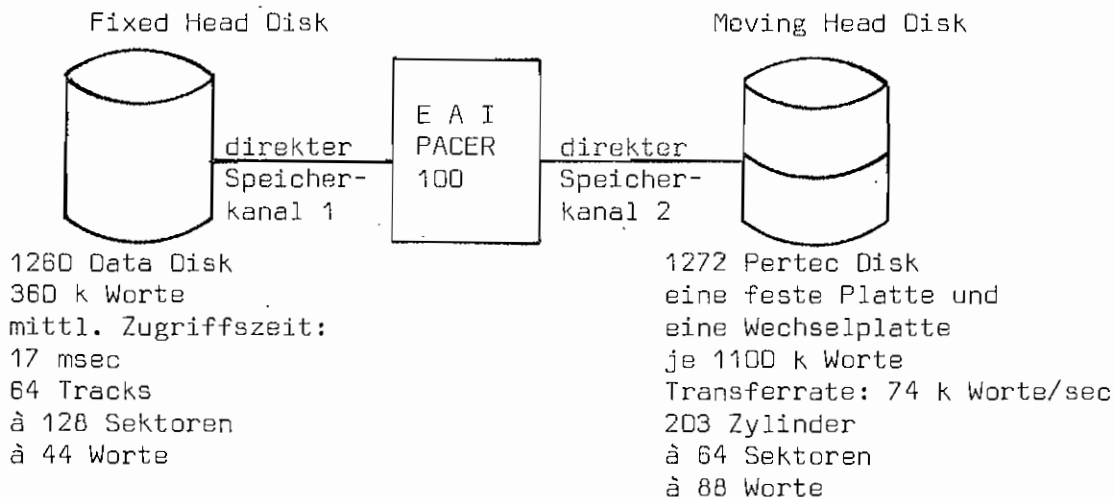
Beispiel: Steuerkarten für ein FORTRAN-Programm, das auf Level 2 (Back) gerechnet werden soll. Das Programm erhält einen speziellen Namen.



Auf den Steuerkarten /FOR, /BUILD, /MAIN und /BACK kann nach einem Leerzeichen ein Programmname angegeben werden (6alphanum. Zeichen). Für Assemblerprogramme ist statt der /FOR-Karte eine /ASM-Karte zu verwenden.

VERWENDUNG DER MASSENSPEICHER IM BETRIEBSSYSTEM JCS/MP 6

Zur Speicherung von Daten und Programmen stehen an der Hybridrechenanlage eine Fixed Head Disk (Festkopfflatte) und eine Moving Head Disk (Wechselplatte) zur Verfügung.



Die mittlere Zugriffszeit auf der Moving Head Disk ist wesentlich abhängig von der Organisation der Files. Innerhalb eines Zylinders beträgt die mittlere Zugriffszeit nur 12.5 msec. Die mittlere Zugriffszeit auf einen Zylinder beträgt jedoch 47 msec.

Am günstigsten ist daher eine konsekutive Abarbeitung der Files.

Zur Errichtung von permanenten Daten- und Programmfiles stehen dem Benutzer Wechselplatten zur Verfügung. Platten mit der Kapazität von 1.1 Mio. Worten à 16 bit können über die Hybridrechenanlage zum Preis von ca. S 2400.- (Kostensersatz) bezogen werden. Bei Verwendung großer Dateien sowie zur Speicherung von großen Programmen oder Programmsystemen empfiehlt sich die Anschaffung einer Platte. Das Abspeichern von Programmen macht das Einlesen der Programmkarten, das Übersetzen und Binden der Programme vor jeder Exekution überflüssig und ermöglicht somit ein rasches und bequemes Arbeiten an der Hybridrechenanlage. Im Betriebssystem JCS/MP 6 existiert ein Utility-Programm, mit Hilfe dessen der Operator Programmfiles von der Wechselplatte auf die Festkopfflatte (innerhalb von Sekunden) übertragen kann, wo das Betriebssystem die Programme bedient, und umgekehrt.

Für Benutzer, die keine eigene Platte anschaffen wollen, stehen an der Hybridrechenanlage Platten bereit, auf denen nach Rücksprache mit dem Operator Daten- und Programmfiles in beschränkter Größe für beschränkte Dauer errichtet werden können.

Die Festkopfflatte wird ausschließlich vom Betriebssystem verwendet, daher stehen Daten- und Programmfiles dem Benutzer nur während der Exekution seines Jobs zur Verfügung. Die Errichtung von Datenfiles auf der Festkopfflatte ist daher nur dann sinnvoll, wenn ein schneller, nichtsequentieller Zugriff zu den Daten unbedingt erforderlich ist (z.B. Hybridprogramm). In diesem Fall empfiehlt es sich, die Daten anschließend auf die Wechselplatte zu übertragen, falls sie konserviert werden sollen.

WANN VERWENDET MAN DATENFILES AUF DER PLATTE ?

1. wenn man über längere Zeit immer wieder einen Satz von Daten benötigt, der einmal von einem Programm erstellt wurde, und es nicht zweckmäßig ist, diesen immer neu zu errichten.

2. wenn der Speicherbedarf der vom Programm anfallenden Daten den zur Verfügung stehenden Kernspeicherplatz bersteigt und die Daten logisch leicht segmentierbar sind (z.B. verschiedene Abtastungszyklen vom Analogrechner).
3. Bei Auftreten eines großen Datenfeldes, das nicht logisch segmentierbar ist und zu dem ein wahlweiser, nichtsequentieller Zugriff möglich sein soll, empfiehlt sich die Verwendung eines virtuellen Feldes.

PROGRAMMIERUNG VON DATENFILES AUF DER WECHSELPLATTE DURCH FORTRAN-AUFRUFBARE UNTERPROGRAMME

Ein Datenfile besteht aus einer Anzahl von Records und diese wiederum aus einer Anzahl von 16-bit Worten. Der Recordzähler ist eine interne Kenngröße, die angibt, welcher Record als nächster behandelt wird (beim Lesen oder Schreiben).

Das Definieren eines Datenfiles geschieht durch Aufruf des Unterprogrammes PFILE, wobei als Parameter eine Filenummer und die Größe des Files (Anzahl der Records, Anzahl der Worte pro Record) angegeben werden müssen. Ein durch PFILE definierter Datenfile bleibt auf der Platte, bis er vom Operator gelöscht wird. Besteht ein Datenfile bereits auf der Wechselplatte, dann dient PFILE zur Positionierung des bereits definierten Datenfiles. Jeder File muß zuerst positioniert werden, bevor man Lese- oder Schreiboperationen durchführt. Nach jedem PFILE-Aufruf steht der Recordzähler auf dem ersten Record des Files.

Der Zugriff auf einen bereits definierten Datenfile erfolgt durch Aufruf der Unterprogramme TDFILE (Schreiben auf einen Datenfile) bzw. OFFFILE (Lesen von einem Datenfile), wobei als Parameter nur das Feld oder die Variable angegeben wird, die übertragen werden soll.

Bei jedem TDFILE- oder OFFFILE-Aufruf wird ein Record des Files beschrieben bzw. gelesen. Anschließend wird der Recordzähler um eins erhöht und steht nun auf dem nächsten Record. Jeder TDFILE- und OFFFILE-Aufruf bezieht sich auf den File, der zuletzt im Programm mit PFILE positioniert wurde.

Das Positionieren des Recordzählers erfolgt durch die Unterprogramme RESTEP (Zurücksetzen des Recordzählers um einen Record) und GFILE (Positionieren auf gewünschten Record).

DATENFILES AUF DER FESTKOPFPLATTE

Die Programmierung ist analog zu jener der Wechselplatte.

Unterprogramm für Festkopfplatte entspricht für Wechselplatte

SFILE	PFILE
TODISK	TDFILE
OFDISK	OFFFILE
RECORD	RESTEP
MFILE	GFILE

Die Parameter haben dieselbe Bedeutung wie jene der entsprechenden Unterprogramme für die Wechselplatte.

FEHLERMELDUNGEN

Bei fehlerhaften Diskoperationen wie z.B. Recordüberschreitung, Recordüberschreitung beim Lesen oder Schreiben wird das Programm vom Betriebssystem abgebrochen und eine entsprechende Meldung ausgegeben.

VIRTUELLES FELD

Falls ein Programm ein sehr großes Feld verwendet, das den vorhandenen Kernspeicherbedarf übersteigt, kann es sinnvoll sein, das Feld auf der (Festkopf-)platte zu definieren.

Durch Aufruf des Unterprogrammes DEFINE wird ein dreidimensionales Feld der angegebenen Größe auf der Platte errichtet.

Der Zugriff auf das virtuelle Feld erfolgt elementweise über die Routinen WDISK (Wertzuweisung) und RDISK (Lesen).

Die Rechenzeit bei Verwendung eines virtuellen Feldes ist grundsätzlich wesentlich länger als bei Verwendung von Datenfiles.

Ausführliche Unterlagen über die Verwendung der beschriebenen Disk-Routinen sind an der Hybridrechenanlage erhältlich.

FORTRAN IV COMPILER JCSFOR

Der im Rahmen des Betriebssystems JCS/MP 6 laufende FORTRAN-Compiler JCSFOR wurde auf der Basis des EAI FORTRAN-Compilers Version H00 erweitert und adaptiert. JCSFOR wird aktiviert durch eine /FOR-Karte, an die sich das Hauptprogramm sowie etwaige Subroutine- und Function-Unterprogramme anschließen. Bis zur nächsten /-Karte werden alle Karten als Source-Eingabe für den FORTRAN-Compiler interpretiert. Programme werden nur dann richtig compiliert, wenn im Hauptprogramm als erster exekutierbarer Befehl ein Aufruf des Systemunterprogramms JOB stattfindet. Falls nur Subroutine- und Function-Unterprogramme übersetzt werden sollen, ist zur Jobkennung ein Dummy-Hauptprogramm mit CALL JOB (NR,ITIME) und END voranzulegen. Jobs, die richtig compiliert wurden, enthalten im Exitausdruck von JCSFOR die Meldung JOB CORRECT. Nur solche Jobs können gebunden und exekutiert werden.

Beim Auftreten fast aller syntaktischen Fehler wird der Job mit ERROR IN JOB abgeschlossen. Syntaktische Fehler werden sofort nach ihrer Erkennung in Textform (nicht, wie bisher üblich, durch einen Fehlercode) mitgeteilt. Die Spalte, in der der Fehler registriert wurde, wird markiert.

```
DO I=1,10
```

```
-----0-----
FALSCHES STATEMENT NR.
```

Nicht immer wird ein Fehler, wie hier, unmittelbar in der Zeile erkannt, in der er auftritt. Fehler in EQUIVALENCE-Anweisungen etwa

werden erst nach dem ersten exekutierbaren Befehl als solche erkannt. Die entsprechende Fehlermeldung in der Programmliste mag dann etwas deplaciert wirken, weshalb auch auf diesen Umstand ausdrücklich hingewiesen sei.

Erwähnt soll auch werden, daß ein Befehl ohne Anweisungsnummer nach einem GOTO wohl eine Fehlermeldung bewirkt, und zwar

```
GOTO 20
M#5
```

-----0-----
STATEMENT WIRD NIE ERREICHT

das Programm aber, wenn es sonst richtig ist, trotzdem gebunden und exekutiert werden kann. Meistens hat der Programmierer aber in diesem Fall eine Anweisungsnummer vergessen. Nicht definierte Anweisungsnummern, auf die Bezug genommen wurde, werden erst am Ende des jeweiligen Haupt- bzw. Unterprogramms erkannt. Sie werden nach jedem Haupt- und Unterprogramm in folgender Form angeführt:

```
CALL JOB (805,2)
A = 3,14159
WRITE (6,20) A
CALL EXIT
END
```

```
U      .20
```

In diesem Beispiel ist die Anweisungsnummer 20 nicht definiert.

Die Übersetzungszeit, d.i. die Exekutionszeit des FORTRAN-Compilers, wird der Jobnummer, die im CALL JOB-Befehl angeführt ist, als Compilierungszeit verrechnet. Die Totalzeit, die im Exitausdruck angegeben wird, beinhaltet nur die Zeiten für Compilieren, Assemblieren, Laden und Binden der Programme.

Leerkarten können zur optischen Gliederung des Lineprinteroutputs verwendet werden. Eine Karte mit der Lochung 12-9-8-4 (Form Character) in der ersten Spalte bewirkt einen Vorschub auf die nächste Seite.

In der nächsten Nummer des Interface wird in einem Artikel auf effektivere Codegenerierung für den fortgeschrittenen FORTRAN-Programmierer eingegangen werden.

EXEKUTIEREN VON PROGRAMMEN MIT SEHR GROSSEM

KERNSPEICHERBEDARF

SEGMENTIEREN IM BETRIEBSSYSTEM JCS/MP 6

Programme, die mehr als 12 k 16-bit Worte Speicherbedarf benötigen, behindern den Multiprogrammbetrieb in JCS/MP 6. Bei Programmen dieser Größe sollte man daher von der Möglichkeit Gebrauch machen, das Programm in Teilen hintereinander zu exekutieren. Dafür muß das Programm in logische Teile gegliedert werden, die unabhängig voneinander als Einzelprogramme exekutiert werden können, und deren Zusammenhang über Daten gegeben ist, die von einem Programm in das andere übertragen werden.

AUFBEREITUNG DER PROGRAMME

Jedes dieser logischen Einzelprogramme muß mit einem CALL JOB (NR,ITIME) als erstes exekutierbares Statement beginnen (NR ist die Jobnummer des Benützers, ITIME die maximale Rechenzeit dieses Einzelprogrammes in Minuten). Falls am logischen Ende eines Programms

CALL LINK

steht, wird das Programm abgeschlossen (die Kontrollmeldung am Lineprinter wird unterdrückt) und das Programm, dessen Name von der nächsten /MAIN Steuerkarte (siehe auch unter Steuerkarten) eingelesen wird, wird als nächstes Programm exekutiert. Auf diese Art können beliebig viele Programme in beliebiger Reihenfolge aufeinanderfolgend exekutiert werden. Das letzte Programm wird durch ein CALL EXIT beendet. Falls ITIME im JOB-Aufruf ein negatives Vorzeichen hat, wird der Ausdruck der Titelseite am Lineprinter für dieses Programm unterdrückt. So kann man einen zusammenhängenden Lineprinterausdruck von mehreren Programmen erhalten.

DATENÜBERGABE

Die Commonblöcke eines Programms, das mit CALL LINK beendet wurde und die darin enthaltenen Werte bleiben zur Gänze erhalten und ermöglichen den Datentransfer zwischen den Einzelprogrammen. Es ist darauf zu achten, daß die Größe der Commonblöcke in allen Programmen übereinstimmt. Die Commonblöcke in den Einzelprogrammen müssen ihrer Reihenfolge nach von unten her (beginnend beim zuletzt angeführten) immer übereinstimmen. In jedem Programm können vor den bereits bestehenden Commonblöcken weitere angeführt werden.

Beispiel:	Programm	Commonblöcke, in folgender Reihenfolge angeführt
	PR1	A2 A1
	PR2	A1
	PR3	A3 A2 A1
	PR4	Q A3 A2 A1

Es ist daher sinnvoll, die am häufigsten verwendeten Commonblöcke als jeweils letzte anzuführen (z.B. A1, A2). Commonblöcke, die nur programmintern verwendet werden, (z.B. Q) müssen vor allen anderen Blöcken angefügt werden (siehe Programm PR4).

Es können bis zu 64 Commonblöcke verwendet werden.

Zur Übergabe von Daten können auch Datenfiles auf der Platte verwendet werden.

WANN IST SEGMENTIEREN NOTWENDIG ?

Großen Kernspeicherbedarf benötigt vor allem der formattisierte FORTRAN Input/Output (ca. 4 k 16-bit Worte). Es empfiehlt sich daher, bei Rechenprogrammen mit großem Kernspeicherbedarf den Input/Output von Daten in getrennten Programmen durchzuführen.

Auch bei Programmen, die mit Data Plotter Unterprogrammen arbeiten, ist es günstig, z.B. die Auswertung von Daten von dem Zeichnen der Daten zu trennen (Datenübergabe über Plattenfiles).

BEISPIEL: BERECHNUNG UND PLOTTEN EINER AUSGLEICHSFUNKTION ZU GEGEBENEN DATENPUNKTEN

1. Das komplette Programm beinhaltet:

Einlesen der Daten, Aufruf von ASP1 (Berechnung der glättenden kubischen Splinefunktion, aus Programmpaket IPP1), Aufruf von ASPLOT (Zeichnen der Ausgleichsfunktion und der Datenpunkte auf dem Plotter, IPP1), sowie Kontrollausdrucke am Lineprinter.

Speicherbedarf: 14.7 k 16-bit Worte, d.h. das Programm kann nicht auf einem Standardlevel im Multiprogrammingbetrieb gerechnet werden (max.12 k)

```

      DIMENSION F(80),X(80),FST(80),S(80),DS(80),W(720)
      CALL JOB(78,10)
      READ (5,2) N
      READ (5,1) (F(I),X(I),I=1,N)
      WRITE (6,3) N
      WRITE (6,4) (I,F(I),X(I),I=1,N)
      GLATT = 5
      DO 100 I=1,N
100  FST(I) = ,1
C
      CALL ASP1 (N,X,F,FST,GLATT,S,DS,IER,W)
C
      WRITE (6,4) (I,S(I),DS(I),I=1,N)
C
      CALL ASPLOT (N,X,F,GLATT,S,DS,4,4HFELD,14,14MMAGNETISIERUNG)
C
      CALL EXIT
1  FORMAT(16F5,0)
2  FORMAT(I3)
3  FORMAT(1X,I3)
4  FORMAT(1X,I3,3X,2E12,4)
      END

```

2. Segmentieren des Programms in drei Einzelprogramme:

INPTPR Einlesen der Datenpunkte, Kontrollausdrucke, Übergabe der Daten in Commonblock A1

MAINPR Aufruf von ASP1, Übergabe der Bestimmungsdaten der Splinefunktion in Commonblock A2

PLOTPR Plotten der Daten und der Splinefunktion, Kontrollausdrucke

Steuerkarten- und Programmreihenfolge:

/FOR

```

DIMENSION F(80),X(80)
COMMON /A1/N,F,X
CALL JOB (78,10)
READ (5,2) N
READ (5,1) (F(I),X(I),I=1,N)
WRITE (6,3) N
WRITE (6,4) (I,F(I),X(I),I=1,N)
1 FORMAT(16F5,0)
2 FORMAT(I3)
3 FORMAT(1X,I3)
4 FORMAT(1X,I3,3X,2E12,4)
CALL LINK
END

```

/BUILD INPTPR

/FOR

```

DIMENSION F(80),X(80),S(80),DS(80),FST(80),W(720)
COMMON /A2/S,DS,GLATT
COMMON /A1/N,F,X
CALL JOB (78,-10)
GLATT = 5.
DO 100 I=1,N
100 FST(I) = 1
C
CALL ASP1 (N,X,F,FST,GLATT,S,DS,IER,W)
C
CALL LINK
END

```

/BUILD MAINPR

/FOR

```

DIMENSION F(80),X(80),S(80),DS(80)
COMMON /A2/S,DS,GLATT
COMMON /A1/N,F,X
CALL JOB (78,-10)
WRITE (6,4) (I,S(I),DS(I),I=1,N)
C
CALL ASPLOT (N,X,F,GLATT,S,DS,4,4HFELD,14,14MHMAGNETISIERUNG)
C
4 FORMAT(1X,I3,3X,2E12,4)
CALL EXIT
END

```

/BUILD PLOTTPR

/MAIN INPTPR

Datenkarten

/MAIN MAINPR

/MAIN PLOTTPR

/END

Der Speicherbedarf jedes dieser drei Programme liegt jetzt unter 12 k. Sie können im Multiprogrammingbetrieb gerechnet werden.

Beim Segmentieren des Programmes besteht gegenüber dem kompletten Programm praktisch kaum ein Rechenzeitverlust.

IPP1 - INTERPOLATIONS- UND PLOTTER-PROGRAMMPAKET

FÜR EINDIMENSIONALE PROBLEME

WAS IST DIE AUFGABENSTELLUNG VON IPP1 ?

IPP1 enthält an die 60 FORTRAN-Unterprogramme, die der Darstellung von qualitativen und quantitativen Zusammenhängen von zwei Veränderlichen dienen, die durch N Wertepaare (x_i, y_i) , $i=1(1)N$ gegeben sind. Diese Darstellung erfolgt durch die Berechnung von möglichst leicht auswertbaren Funktionen, die durch diese gegebenen Punkte verlaufen (Interpolation) und außerdem durch die graphische Darstellung dieser Funktionen auf einem Plotter. Obwohl einige Ausgleichsprogramme in IPP1 enthalten sind, wird die Behandlung von Ausgleichsproblemen in einem separaten Programmpaket erfolgen.

BESTEHEN EINSCHRÄNKUNGEN FÜR DIE WERTEPAARE (x_i, y_i) ?

In IPP1 sind zwei Möglichkeiten berücksichtigt:

- a.) Die x -Werte sind monoton steigend (oder fallend) $x_1 < x_2 < \dots < x_N$.
In diesen Fällen kann eine eindeutige Funktion durch die Punkte (x_i, y_i) gelegt werden.
- b.) Die x -Werte sind nicht monoton.
Hier besteht die Möglichkeit, eine ebene Kurve durch die Punkte (x_i, y_i) zu legen. Falls der erste und der letzte Punkt übereinstimmen, kann eine geschlossene ebene Kurve durch diese Punkte gelegt werden.

Es bestehen also keine Einschränkungen für die gegebenen Punkte.

WELCHE FUNKTIONEN STEHEN FÜR DIE INTERPOLATION ZUR VERFÜGUNG ?

- a.) Polynome
- b.) rationale Funktionen (nach J. Stoer)
- c.) stückweise kubische Funktionen nach H. Akima
- d.) kubische Splinefunktionen
- e.) rationale Splinefunktionen

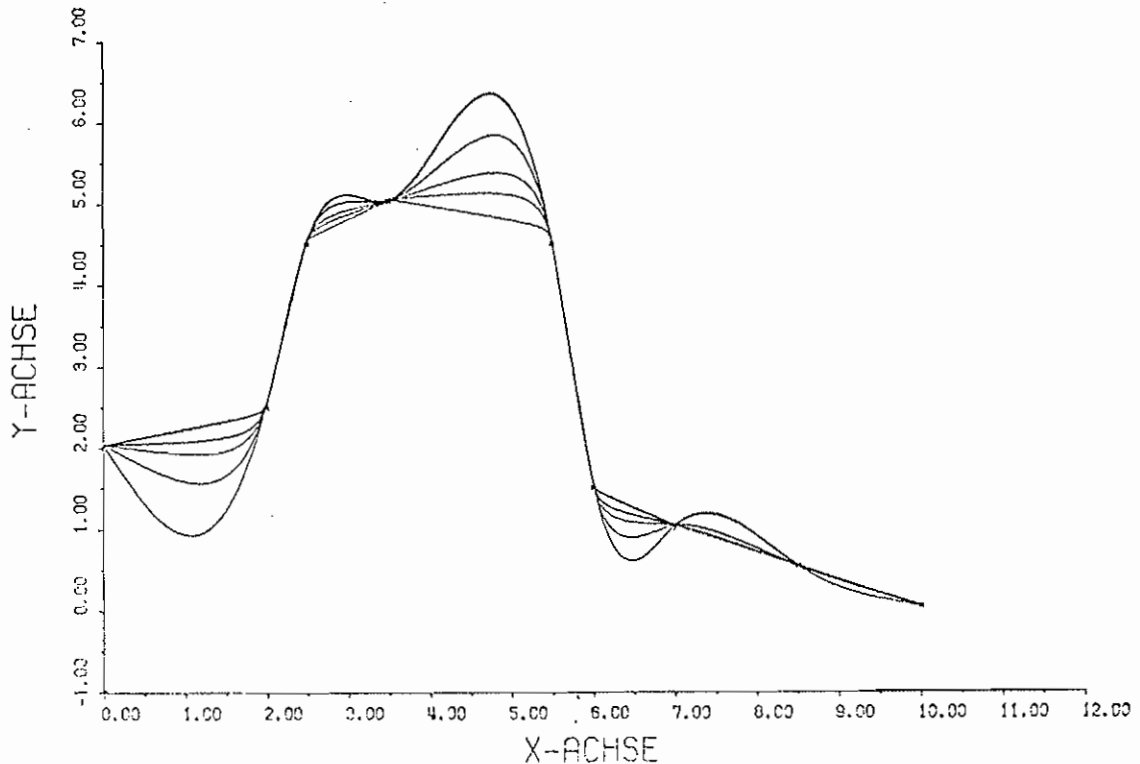
WANN VERWENDET MAN WELCHE INTERPOLATIONSFUNKTION ?

Man kann zwei Fälle unterscheiden:

- a.) Die Datenpunkte liegen "dicht" beieinander, d.h. die Anzahl der Punkte ist mindestens zehnmal so hoch wie die zu erwartende Anzahl der Wendepunkte der Funktion durch die (x_i, y_i) (z.B. in Tabellenwerken von Standardfunktionen). Hier ist die Verwendung von Polynomen oder rationalen Funktionen (nach J. Stoer) zu empfehlen.
- b.) Der häufig auftretende Fall "weit auseinander liegender" Punkte, d.h. die Anzahl der Punkte ist nicht zehnmal so hoch wie die erwartete Anzahl von Wendepunkten, oder anders ausgedrückt: an Stellen mit starken Schwankungen der zweiten Ableitung liegen nur wenige Datenpunkte vor. In diesen Fällen macht sich starkes Überschwingen des Interpolationspolynoms bemerkbar (speziell in den Randintervallen), die Interpolationsfunktion ist also nicht "glatt" genug. Geeignete "glatte" Interpolationsfunktionen für diesen Fall

sind die stückweise kubischen Funktionen nach H. Akima, die kubischen Splinefunktionen und die rationalen Splinefunktionen. Da diese Funktionen auch bei Fall a.) angewendet werden können, wurde das Hauptgewicht auf diese Funktionen gelegt. Dabei haben Funktionen nach Akima und die kubischen Splinefunktionen feste Glattheitseigenschaften, während die Glattheit der rationalen Splinefunktionen durch die Wahl eines Parameters gesteuert werden kann. Dadurch ermöglichen die rationalen Splinefunktionen eine sehr flexible Anpassung an spezielle Interpolationssituationen.

Rationale Splinefunktion mit verschieden stark gedämpftem Überschwingverhalten



WIE WERDEN EBENE KURVEN DARGESTELLT ?

Für ebene Kurven wurde generell die Sehnenlänge zwischen zwei aufeinanderfolgenden Punkten als Parameter t eingeführt. Auf diese Art werden ebene Kurven durch die Berechnung von zwei eindeutigen Interpolationsfunktionen $((t_i, x_i)$ und (t_i, y_i) , $i=1(1)N$) erhalten.

WIE ERFOLGT DAS PLOTTEN DER INTERPOLATIONSFUNKTIONEN ?

Die Plotter-Elementarsoftware enthält neben Skalierungs- und Achsenbeschriftungsprogrammen nur Unterprogramme für die geradlinige Verbindung von zwei Punkten (wobei diese Gerade in kleine Teilschritte - increments - zerlegt wird, die dann von der Zeichenfeder ausgeführt werden). Alles was über Polygonzüge hinausgeht muß also in eigenen Programmen auf Polygonzüge zurückgeführt werden. IPP1 enthält daher für alle verwendeten Interpolationsfunktionen Unterprogramme zur Approximation durch Polygonzüge, wobei durch geeignete Kompromisse zwischen Rechenzeit und Approximationsgüte hohe Effizienz erreicht wurde.

WELCHE EINARBEITUNGSZEIT BENÖTIGT MAN, UM MIT DEN PLOTTERPROGRAMMEN UMGEHEN ZU KÖNNEN ?

Gar keine. Die Programme in IPP1 ermöglichen das Plotten von Interpolationsfunktionen unter bestimmten Standardannahmen (z.B. Zeichnung erfolgt im DIN-A3-Format, Achsen werden gezeichnet und beschriftet, etc.) ohne die geringsten Plotterkenntnisse (software oder hardware) vom Benutzer vorauszusetzen.

WIE KÖNNEN DIE IPP1-PROGRAMME ZUR ANFERTIGUNG GANZ SPEZIELLER DARSTELLUNGEN VERWENDET WERDEN ?

Die Unterprogramme von IPP1 können entsprechend folgender Hierarchie eingeteilt werden:

a.) benutzerorientierte Hilfsprogramme

Die Verwendung setzt keine Spezialkenntnisse voraus. Hier werden Standardannahmen getroffen; die Skalierung festgelegt und die Achsen gezeichnet; die Funktion durch einen Polygonzug approximiert und gezeichnet. Diese Programme sind so gestaltet, daß sie leicht vom Benutzer an spezielle Probleme angepaßt werden können und werden auf Wunsch auf Lochkarten zur Verfügung gestellt. Dafür werden allerdings gewisse Grundkenntnisse der Plotter-Elementarsoftware benötigt.

b.) mathematische Unterprogramme

Diese Programme sind möglichst effizient und maschinenunabhängig programmiert und stellen Bausteine dar, die zur direkten Verwendung durch den Benutzer oder für die Verwendung in benutzerorientierten Hilfsprogrammen (z.B. Polygonzugapproximation) gedacht sind. Veränderungen dieser Programme durch den Benutzer setzen Spezialkenntnisse voraus.

c.) geräteorientierte Unterprogramme

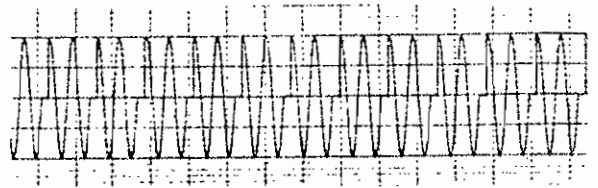
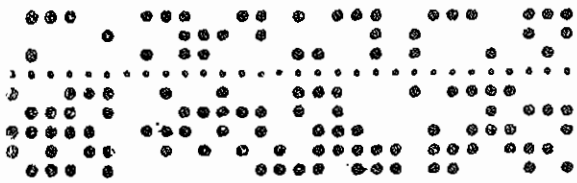
Hier erfolgt z.B. die Steuerung des Zeichenstifts. In diesen Unterprogrammen sind die maschinenabhängigen Teile der Plotterroutinen enthalten, die beim Wechsel auf andere Anlagen zu ändern sind.

WIE SIND DIE PROGRAMME DES IPP1 VERFÜGBAR ?

Sämtliche Programme sind auf Platte gespeichert und können von FORTRAN-Programmen aus direkt aufgerufen werden.

WELCHE UNTERSTÜTZUNG ERHÄLT MAN BEI DER BENÜTZUNG DES IPP1 ?

Für alle Programme stehen schriftliche Beschreibungen zur Verfügung. Falls die dort enthaltenen Informationen in Spezialfällen nicht ausreichend sein sollten, wird von der Programmberatung die notwendige Unterstützung geleistet.



BENÜTZERFORUM

PLOTTEN VON MOSSBAUERSPEKTREN

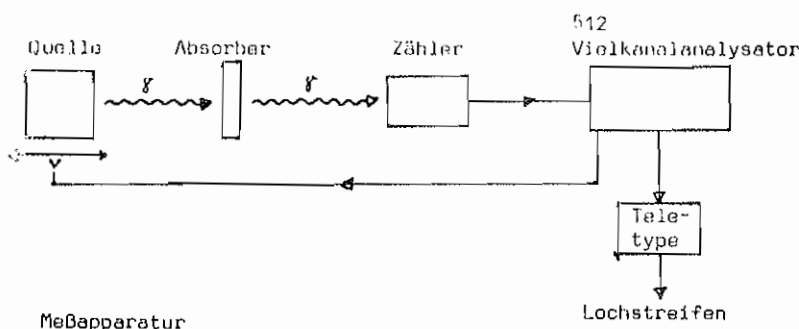
R. Haferl
W. Steiner
G. Wiesinger
Institut für Experimentalphysik
Institut für Angewandte Physik

Die Untersuchung der Verbindungen, welche die seltenen Erd-Metalle einschließlich Yttrium mit den Übergangselementen Fe und Co bilden, ist ein interessantes Teilgebiet der angewandten Physik.

Von übergeordneter Bedeutung ist dabei die Kenntnis der magnetischen Eigenschaften dieser intermetallischen Verbindungen, und im besonderen die Anordnung der Fe-Atome im Kristallgitter. Zur Ermittlung der Einflüsse dieser Fe-Atome auf die Ausbildung eines magnetischen Ordnungszustandes im Gitter ist eine Computer-Auswertung der durch Mössbauer-Messungen ermittelten Spektren vorteilhaft, wobei die tatsächlich gemessenen mit überlagerten theoretisch berechneten verglichen werden.

Das erstellte Programm läßt eine Überlagerung von bis zu zehn bezüglich Hyperfeinfeldern, Quadrupolaufspaltungen, Isomerieverschiebungen, Linienbreiten und Intensitäten unterschiedlichen Spektren zu.

Bevor auf die Auswertung und das Plotten eingegangen wird, möge eine Schemazeichnung die Durchführung der Messungen erläutern:



Unter Quelle ist ein radioaktiv strahlendes Präparat zu verstehen, das Gamma-Quanten emittiert. Diese Quelle wird mit konstanter Beschleunigung bewegt, wodurch sich (Doppler-Prinzip!) eine Bestrahlung des Absorbermaterials mit einer äußerst scharf definierten, jedoch in einem vorgegebenen Bereich variierbaren Strahlung realisieren läßt. Im Absorber findet, bestimmt durch die Energieniveaus des Fe^{57} -Atoms, Resonanzabsorption- und Emission bestimmter Energien statt. Die durch diesen Effekt gefilterte Strahlung der Quelle wird mit einem Proportionalzählrohr registriert. Die elektronischen Impulse werden dann in einem Vielkanalanalysator gespeichert (Abb. 1).

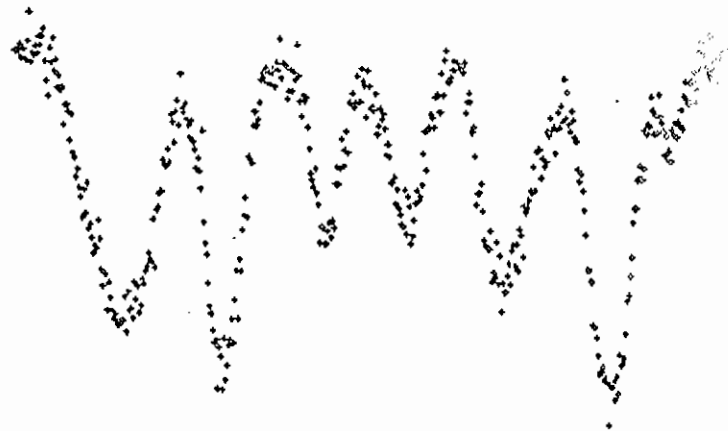


Abb. 1 Meßspektrum

Die Ausgabe dieser Messungen erfolgt auf Lochstreifen mittels eines angeschlossenen Teletypes. Diese Streifen können unmittelbar vom Lochstreifenleser der Hybridrechenanlage gelesen werden. Jeder Datensatz repräsentiert eine Meßzeit zwischen 24 und 240 Stunden.

Ziel der Auswertung ist es, aus den Meßwerten die besten Näherungen der Parameter des theoretischen Mössbauerspektrums mit einem nichtlinearen Ausgleichsverfahren zu ermitteln (Parameteridentifikation). Dieses theoretische Spektrum wird durch eine Überlagerung von Lorenzlinien gebildet und läßt sich folgendermaßen darstellen:

$$(1) F(V) = U_0 - \sum_{i=1}^N \sum_{j=1}^6 \frac{A_{ij} \cdot U_{4i} \cdot U_5^2}{(V - U_{3i} - U_{2i} \cdot B_{ij} - U_{1i} \cdot C_{ij})^2 + U_5^2}$$

Parameter:

U_{1i} .. Hyperfeinfelder

U_{2i} .. Quadrupolaufspaltungen

U_{3i} .. Geschwindigkeit der 1. Linie

U_{4i} .. Intensitäten

U_5 .. halbe Halbwertsbreiten

U_0 .. Zählratenuntergrund

Konstante:

N ... Anzahl der nichtäquivalenten Fe-Plätze

$N \in \{1, \dots, 10\}$ je nach Art der Messung

A_{ij}, B_{ij}, C_{ij} ... Konstante

Für eine Beurteilung der Approximationsgüte des Ausgleichs ist es wünschenswert, die Meßdaten samt theoretischem Spektrum auf einem Plotter graphisch darzustellen. Dies läßt sich mit dem Aufruf von zwei Unterprogrammen des Pakets IPP1 leicht bewerkstelligen.

Das Unterprogramm SPLAX liefert eine Splineapproximation der Funktion (1) mit Zeichengenauigkeit und benötigt dazu nicht mehr als 180 Funktionsauswertungen.

Das Zeichnen dieser Splineapproximierenden erfolgte mit einem Aufruf des Unterprogrammes SPLOT, das Achsenbeschriftungen und Skalierungen selbsttätig vornimmt und die Funktion sowie die Datenpunkte am Plotter darstellt (Abb. 2).

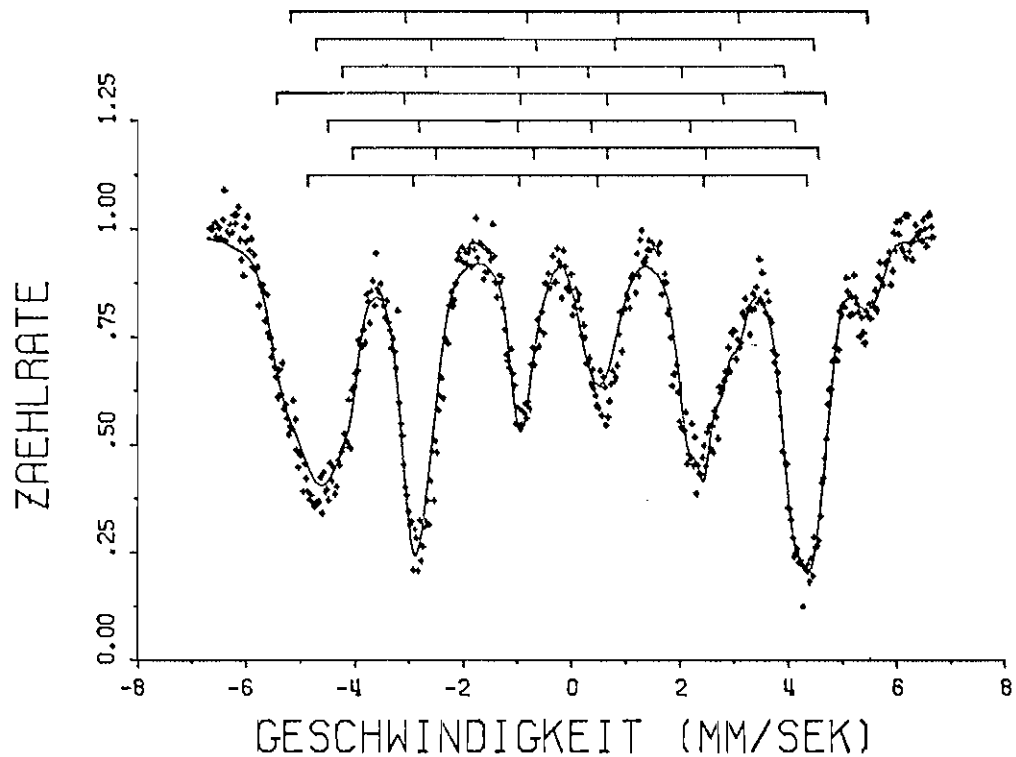


Abb. 2
(Original Din A3)

PACER MUSICALE

A. Gsandtner
Hybridrechenanlage
des Rechenzentrums der Technischen Hochschule
Wien

Das Softwarepaket "Pacer Musicale" ist ein interaktives Hybridprogramm zur Analyse und Synthese von Musik.

"Pacer Musicale" hat in der jetzigen Ausbaustufe drei Schwerpunkte:

1. Abspielen von Musikstücken, deren Noten auf Karten eingegeben werden
2. Erkennen der Noten aus gesungenen oder gespielten Melodien
3. Noten schreiben (in Arbeit)

Über diesen Modulen steht ein interaktiver Supervisor, der mittels einer bequem zu verwendenden Steuersprache die Funktionen des Programmes aktiviert. Das Programm besitzt permanente Datenfiles, die Information über bisherige Runs und die Daten der jemals eingelesenen oder analysierten Musikstücke enthält. Derzeit beschränkt sich Pacer Musicale auf einstimmige Musikstücke, da der Aufwand bei der Analyse mehrstimmiger Musikstücke eminent ansteigt (Fourier-Analyse!).

1. Synthese von Musik

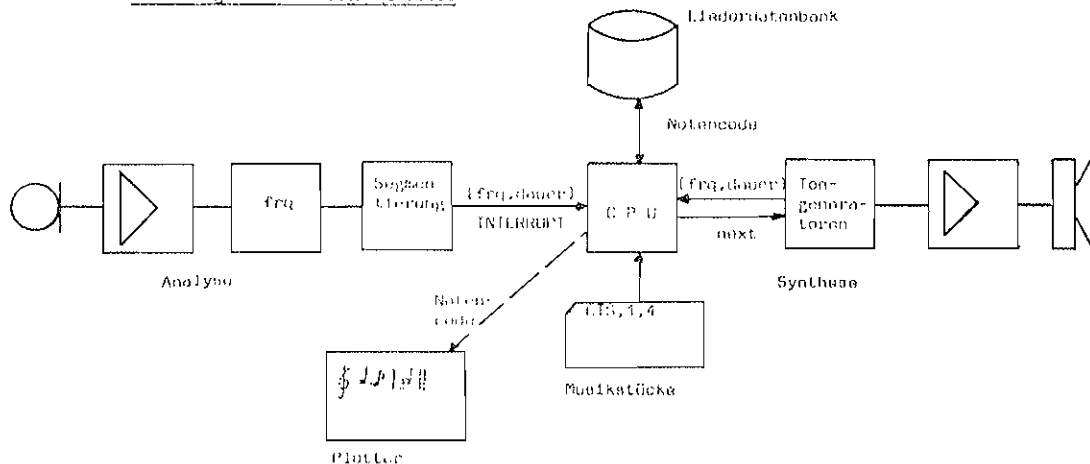
Für jede zu spielende Note wird eine Karte mit einem Code eingegeben, der der gebräuchlichen Bezeichnung der Noten entspricht, etwa für eine Viertelnote eines eingestrichenen cis: cis,1,4. Diese Karte wird mit Hilfe einer anfangs angegebenen Tempobezeichnung in ein

Paar (freq,dauer) umcodiert und auf den permanenten File gelegt. Ist ein Lied zu Ende gelesen, erfolgt ein Kontrollvorspiel. Die Generierung des Musikstücks erfolgt auf dem Analogrechner, der als Zusatz nur einen Verstärker und einen Lautsprecher benötigt. Der Digitalrechner stellt dem Analogrechner das Paar (freq,dauer) zur Verfügung, der Analogrechner generiert einen Ton mit der Frequenz freq und der Länge dauer und schaltet dann auf den nächsten Ton weiter. Das Klangspektrum der generierten Töne ist dabei sowohl analog als auch digital festlegbar. Einige Möglichkeiten sind: Sinus-, Dreiecks-, Rechtecks-, analoge und digitale Funktionsgeneratoren, sowie die Mischung bzw. Überlagerung dieser Schwingungen. In einer weiteren Stufe wird auch der Klang digital wählbar sein. Derzeit erfolgt die Wahl und Änderung des Klanges von einem Mischpult aus. Eine Erweiterung auf mehrstimmige Musikstücke ist für die Synthese geplant.

2. Frequenzanalyse von Musikstücken

Die Analyse segmentiert das Musikstück in Elementarabschnitte (=Noten), die entweder durch das Auftreten einer Pause oder durch eine Frequenzänderung begrenzt werden. Die Frequenz wird durch Messung der Zeit von einem Maximum des Signals bis zum nächsten bestimmt. Durch ein Analogprogramm wird die Dauer einer konstanten Frequenz bestimmt und Fehler bei der Frequenzbestimmung werden korrigiert. Bei jedem Notenwechsel wird das Paar (freq,dauer) an den Digitalrechner übertragen und in den Notenschriftcode umgewandelt. Nach Ende der Analyse wird das Musikstück auf den permanenten Datenfile gelegt. Natürlich kann dieses Musikstück nach Punkt 1. wieder synthetisiert werden.

Blockdiagramm von Pacer Musicale

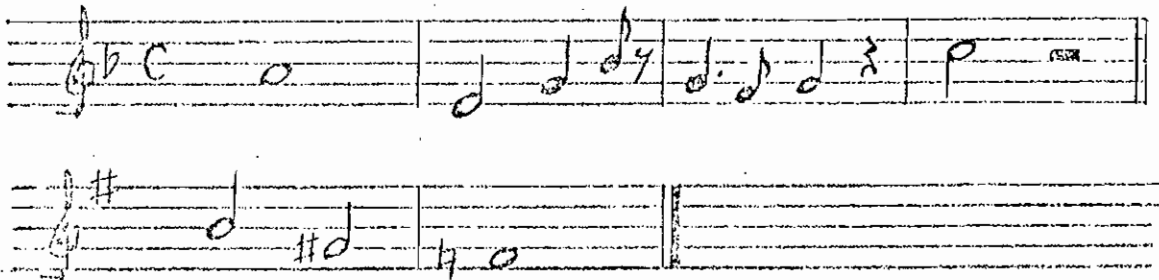


Vor der Fertigstellung steht ein Modul zum Notenschreiben aus dem erwähnten Notenschriftcode. Auf dem Digitalplotter wird konventionelles Notenmaterial gezeichnet, das sich von händisch geschriebenen oder gedruckten Noten kaum unterscheidet und von Musikern gelesen oder gespielt werden kann.

Das Programmsystem "Pacer Musicale" zeigt, wie das Gebiet der Computermusik von einem Hybrid-

rechner wesentlich ökonomischer als rein digital abgedeckt werden kann. Erst durch die analoge Vorverarbeitung und Informationsverdichtung wird eine Echtzeitverarbeitung von Musiksignalen möglich. Außerdem ist der Analogrechner eine optimale Schnittstelle zur akustischen Ausgabe.

"Pacer Musicale" baut nicht - wie viele Computermusikprojekte - auf vordergründige Effekte auf, sondern ist ein Utilityprogramm sowohl für den konventionellen Musiker als auch für Computermusiker.



Herr Anton Gsandtner ist seit Oktober 1974 Mitarbeiter an der Hybridrechenanlage. Seine Arbeitsgebiete sind: Programmberatung für Hybridbenutzer, Hardwareentwicklung und -modifikation, Data Acquisition, Datenfernverarbeitung.

INTERFACE Februar 1975