
Interface

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

Nummer 1
Oktober 1974



HYBRIDRECHENANLAGE EAI PACER 600

INHALTSVERZEICHNIS

	Seite
Einleitung	3
Technischer Betrieb	8
Systemkonfiguration	9
Arbeitsbereiche an der Hybridrechenanlage	10
Kurse	12
Benützung des Betriebssystems JCS/MP 6	14
Steuerkarten in JCS/MP 6	18
Sprachprozessoren	19
Besonderheiten des EAI-FORTRAN	22
Programmbibliothek	26
Benützerforum	29

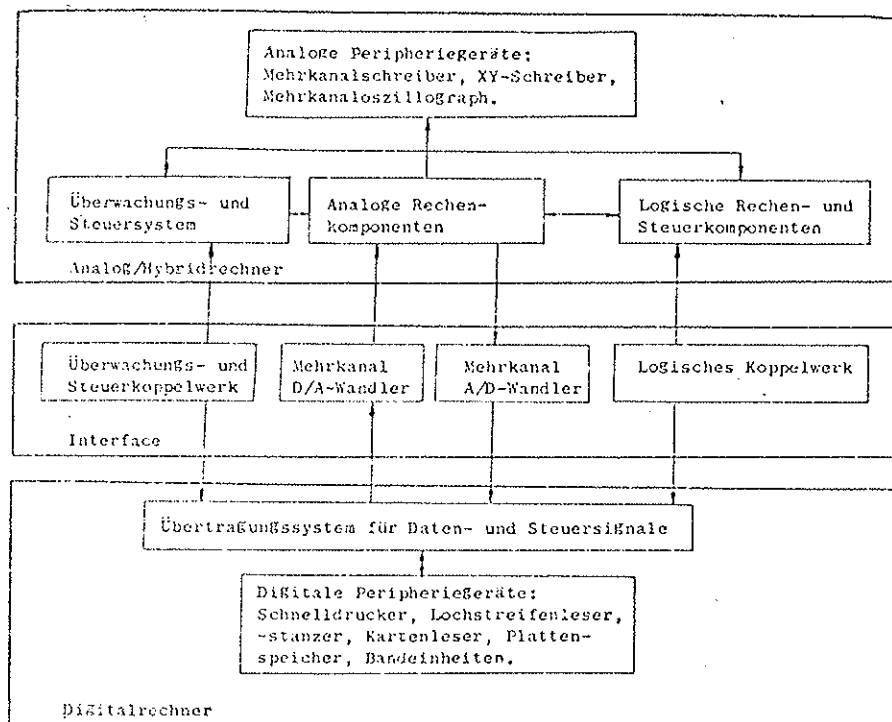
Titelfoto von INTERFOTO Wien

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.

Mit der Herausgabe der mehrmals im Jahr erscheinenden Informationen der Hybridrechenanlage 'Interface' wollen wir eine neue Verbindung (interface) zu dem Kreis der Benutzer herstellen. 'Interface' soll den Informationsfluß zu den bisherigen Benützern vergrößern und ihnen die Möglichkeit zu einem gegenseitigen Erfahrungsaustausch geben. Darüber hinaus wollen wir die Möglichkeiten und Einsatzgebiete des derzeit einzigen in Österreich installierten Hybridrechensystems einem größeren Kreis zugänglich machen. Im folgenden sollen einige häufig gestellte Fragen beantwortet werden.

WAS IST EIN HYBRIDRECHNER ?

Ein Hybridrechner besteht aus einem Digitalrechner, der über eine spezielle Verbindungseinheit (interface) mit einem oder mehreren Analogrechnern (parallel processors) gekoppelt ist. Bei dem Digitalteil handelt es sich meist um einen Rechner mittlerer Größe, der über die übliche Peripherie (Kartenleser, Schnelldrucker, Platten, usw.) verfügt und im allgemeinen in FORTRAN programmiert wird. Ferner muß der Digitalrechner mit einer flexiblen Input/Output- und Interruptstruktur ausgestattet sein. Der Analogteil gestattet die parallele Durchführung von Integrationen, arithmetischen Operationen und logischen Verknüpfungen, sowie die direkte Einbeziehung von Labor- und Meßgeräten. Das Interface dient zur gegenseitigen Steuerung der beiden Rechner, sowie zum Datenaustausch (Analog-Digital-, Digital-Analog-Wandler).



WAS IST DER VORTEIL DER HYBRIDEN GEGENÜBER DER REIN ANALOGEN RECHNUNG ?

Analogrechner sind nur bedingt zur Speicherung von Daten oder ganzer Folgen logischer Instruktionen geeignet. Die Genauigkeit eines Analogrechners ist begrenzt durch die Genauigkeit der Rechenkomponenten, die in der Regel bei 10^{-4} liegt.

Die Verwendung eines Digitalrechners in einem Hybridsystem erschließt daher ganz neue Möglichkeiten, die in folgenden Punkten zusammengefaßt werden können:

1. Die Lösung partieller Differentialgleichungen unter Verwendung von Speicherung und Playback von Funktionen im digitalen Teil und kontinuierlicher Integration im analogen Teil.
2. Die Lösung dynamischer Optimierungsprobleme mittels iterativer Methoden.
3. Die digitale Speicherung von Daten und Instruktionen, die nach Durchführung eines Programmes im Analogteil verarbeitet werden können.
4. Die hochgenaue Lösung von Problemen mit hochfrequenten Untersystemen (z.B. Raketensimulation).
5. Die Programmierung einer Folge von Lösungen, die sehr schnell auf dem Analogrechner gerechnet werden und dann vom Digitalrechner aufgenommen oder verarbeitet werden.
6. Die Programmierung des Analogrechners, d.h. automatisches Einstellen der Servopotentiometer und Testen vom Digitalrechner aus.

WO WENDET MAN HYBRIDRECHNER AN ?

- LUFT- UND RAUMFAHRT

Die Luft- und Raumfahrtindustrie ist der größte Abnehmer hybrider Rechensysteme. Darüber hinaus kommen Hybridrechner auf den verschiedensten Gebieten zur Anwendung.

- VERFAHRENSTECHNIK

Die chemische und petrochemische Industrie verwendet in zunehmendem Maße Simulationsmodelle, um Zuverlässigkeit und Wirtschaftlichkeit technischer Anlagen auszuwerten. Es wurden mathematische Modelle für Einheiten wie Destillationskolonnen, chemische Reaktoren, Wärmetauscher etc. geschaffen.

-ENERGIETECHNIK

Simulation ist seit je eine Domäne beim Entwurf von Kernenergie-

anlagen. Noch während ein neues Kernkraftwerk auf dem Zeichenbrett entsteht, muß das Gesamtverhalten der Anlage untersucht werden und kontrollierbar sein. Eine Untersuchung von kritischen Grenzfällen kann aus verständlichen Sicherheitsgründen am fertigen Objekt nicht mehr durchgeführt werden. Es sind mathematische Modelle zur Darstellung von Kernreaktoren, Wärmetauschern, Regelmechanismen und Turbinen erarbeitet worden, die eine Simulation dieser Anlagen ermöglichen.

- BIOMEDIZIN

Biomedizinische Applikationen haben sich ebenfalls zu einem bedeutenden Anwendungsfeld hybrider Rechenanlagen entwickelt. Simulation des Blutkreislaufes, des menschlichen Wärmehaushaltes und die Verarbeitung von EEG- und EKG-Daten sind nur einige Beispiele aus einer Vielzahl von Anwendungsmöglichkeiten.

- UMWELTSCHUTZ

In steigendem Maß werden verschiedene ökologische Faktoren, wie z.B. der Sauerstoffgehalt von Flüssen auf Hybridrechnern modelliert. Dadurch wird die Auswahl von kostenoptimalen Umweltschutzrichtungen ermöglicht.

- HOCHSCHULBEREICH

Hybriden Rechensystemen kommt in aller Welt eine tragende Rolle im Ausbildungsprogramm an Universitäten und Technischen Hochschulen zu. Durch die Simulation mathematischer Modelle kann das Verhalten komplexer technischer Anlagen anschaulich demonstriert werden und der kostspielige, unübersichtliche und evt. gefährliche Aufbau umfangreicher Laborexperimente wird erspart.

Neben den typischen industriellen Applikationen ergeben sich für Universitäten und Hochschulen Einsatzmöglichkeiten auf theoretisch wissenschaftlichem Gebiet. Letztere Applikationen lassen sich in drei Kategorien gliedern.

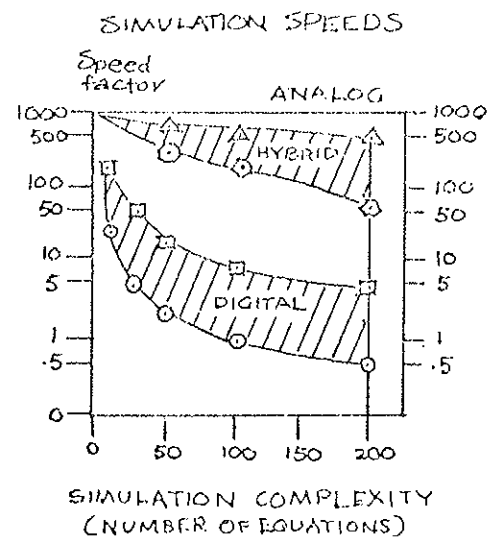
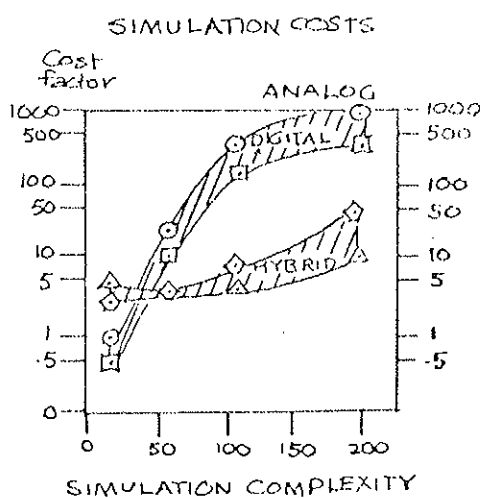
1. Untersuchungen über Sampling-Theorien, da in Hybridrechnern sowohl kontinuierliche als auch diskrete Signale verarbeitet werden.
2. Untersuchungen über numerische Rechenverfahren und Computertechnik, bei denen die exakte Lösung von Differentialgleichungen kontinuierlich mit numerischen Approximationen verglichen werden muß.

3. Theoretische Untersuchungen über Leistungsfähigkeit und Grenzen digitaler Datenverarbeitungssysteme und ihre Verwirklichung in Steuersystemen und Systemtheorien.

KÖNNEN HYBRIDRECHNER DURCH DIGITALRECHNER ERSETZT WERDEN ?

Bei der Echtzeitsimulation großer Systeme kann der Hybridrechner auf keinen Fall durch einen Digitalrechner ersetzt werden. Bei reinen Simulationsaufgaben wurden immer wieder Stimmen laut, die mit der Entwicklung großer Digitalrechner und komfortabler Simulationsprogramme dem Hybridrechner die Existenznotwendigkeit absprachen. Wenn man sich jedoch die oft gravierenden Unterschiede der Rechengeschwindigkeiten zwischen Digital- und Hybridsystemen vor Augen führt, kann man sich leicht vom Gegenteil dieser Behauptung überzeugen. Überzeugend kommt dieser Unterschied in folgendem praktischen Beispiel zum Ausdruck:

Die Simulation eines Kernkraftwerkes (dargestellt in 70 Differentialgleichungen) bei der British Central Electricity Generation Board wurde auf einem sehr schnellen Digitalrechner in London und auf auf einem großen EAI 8900 Hybridsystem in Princeton, N.J. programmiert. Der Vergleich ergab einen Rechenzeitunterschied von 200 zu 1 und einen noch größeren Unterschied bezüglich der aufgetretenen Kosten.



$$\text{Cost} = (\text{programming}) + \frac{\text{PRICE}}{\text{SPEED FACTOR}}$$

- MEDIUM SCALE DIGITAL COMPUTER
- ⊞ LARGEST DIGITAL COMPUTERS
- ⊕ MEDIUM SCALE HYBRID COMPUTER
- Δ LARGE SCALE HYBRID COMPUTER

Hybridrechner werden auch deshalb verwendet, weil sie dem Ingenieur eine praxisbezogene Bearbeitung seiner Probleme ermöglichen und vor allem den Einfluß verschiedener Parameter auf das von ihm zu konstruierende System unmittelbar erkennen lassen.

Der 'Kontakt zum Problem' geht hierbei nicht durch unübersichtliche Programmieralgorithmen und verfahrensbedingte Transformationen verloren; vielmehr ist der Zusammenhang aller Variablen jederzeit überschaubar und direkt kontrollierbar.

Die Möglichkeit, durch manuelle oder automatische Parameteränderung und gleichzeitiges Beobachten deren Auswirkungen unmittelbar in ein Programm einzugreifen, schafft eine engere Beziehung des Wissenschaftlers zu seinem Problem.

Diese unmittelbare, direkte Wechselwirkung zwischen Mensch und Rechenmaschine, auch man-machine-interface genannt, gilt nicht zuletzt als ein entscheidendes Argument für den Einsatz hybrider Rechensysteme in Gegenwart und Zukunft.

WELCHE ZUSAMMENHÄNGE BESTEHEN ZWISCHEN PROZESS- UND HYBRID-RECHENANLAGEN ?

Bei einem Prozeßrechner handelt es sich um einen speziellen Typ eines Digitalrechners, der zur Steuerung oder Regelung von technischen, biologischen u.ä. Abläufen eingesetzt werden kann. Um die Steuerung oder Regelung derartiger Abläufe zu ermöglichen, müssen entsprechende Programmsysteme mit Hilfe des Prozeßrechners geschaffen werden. Hierbei unterscheidet man einerseits die Phase der Programmerstellung samt formalem Testen dieser Programme und andererseits die Phase der Prüfung der erstellten Programme in Zusammenarbeit von Prozeßrechner und dem zu steuernden Ablauf. In bestimmten Fällen wird nun der Aufbau des Ablaufes (Experiments), der in der letzten Phase der Erstellung eines Programmsystems ja unbedingt benötigt wird, aus finanziellen, sicherheitstechnischen o.ä. Gründen nicht möglich sein. Hier bietet sich die Simulation des technischen, biologischen usw. Ablaufes auf einem Hybridrechner äußerst vorteilhaft an, wodurch Zeit und Kosten der Vorbereitungs- und Entwicklungsphase einer Prozeßrechnersteuerung oder -regelung wesentlich gesenkt werden können.

TECHNISCHER BETRIEB

Die Abteilung Hybridrechenanlage des Rechenzentrums der Technischen Hochschule Wien befindet sich in der Gußhausstraße 27-29, 1040 Wien
4. Stock, Zimmer 1401 - 1405, 1418

Öffnungszeiten:	Montag bis Freitag
(Betriebszeiten)	8 - 20 Uhr

Zur Benützung des Hybridrechners ist eine JOBNUMMER erforderlich. Antragsformulare sind beim Operator erhältlich.

Rechenzeit zur Benützung des Analogrechners oder des gesamten hybriden Systems kann im Ausmaß von maximal 2 Stunden zweimal pro Woche beim Operator (Zimmer 1404/05 oder telephonisch unter 65-37-85/803 DW) reserviert werden.

Für Kurzjobs (maximale Rechenzeit 10 Minuten) besteht ein

CLOSED-SHOP-BETRIEB

Die Ein- und Ausgabe erfolgt am Gang vor dem Rechenraum in den dafür vorgesehenen Laden.

Eingabe:

9 Uhr

12 Uhr

17 Uhr

Montag bis Freitag

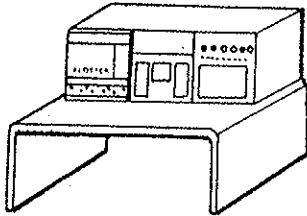
Die Ausgabe erfolgt spätestens bis zu Beginn der nächsten Closed-Shop-Zeit, im allgemeinen jedoch eine Stunde nach der Eingabe.

SYSTEMKONFIGURATION HYBRIDRECHENANLAGE

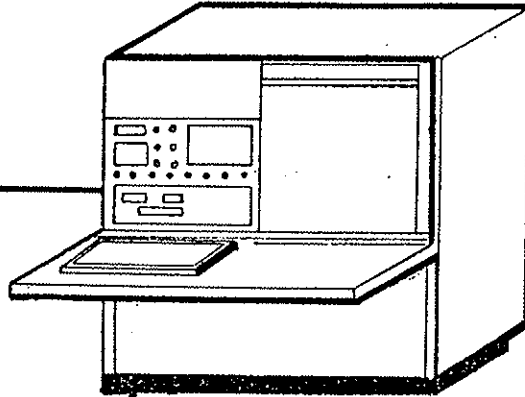
EAI PACER 600

Analog Konsole:

X-Y Schreiber
4-Strahloszillograph
6-Kanalrecorder



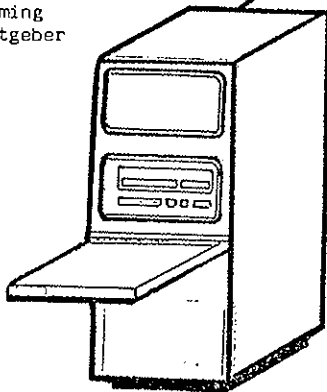
EAI 600 Analogrechner



Parallele Logik
30 Integrierer
204 Verstärker
24 Multiplizierer
Exp-, Log-, Sin-, Cos-
Funktionsgeber
Rauschgenerator
100 Servopotentiometer

EAI PACER 100 Digitalrechner

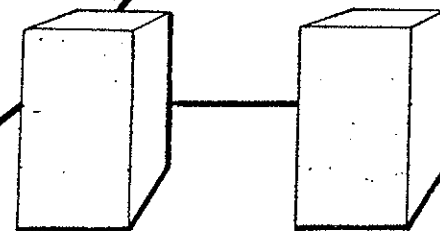
32 k Worte à 16 bit
1 µsec Zykluszeit
64 Interruptebenen
32-bit Floating Point
Processor
1 Standard I/O Kanal
2 Direkte Speicherkanäle
(max. 1Mhz Transferrate)
Multiprogramm
Intervallzeitgeber



direkter
Speicherkanal

Interface-Erweiterung

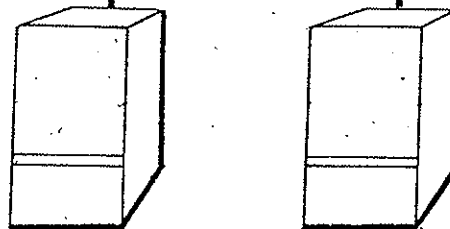
20 Digitale Potentiometer
2 Digitale Funktionsgeneratoren



693 Interface

24 Analog-Digital-Wandler
16 Digital-Analog-Wandler
4 Digital-Analog-Multiplizierer
8 Interruptebenen
8 Senselines
16 Controllines

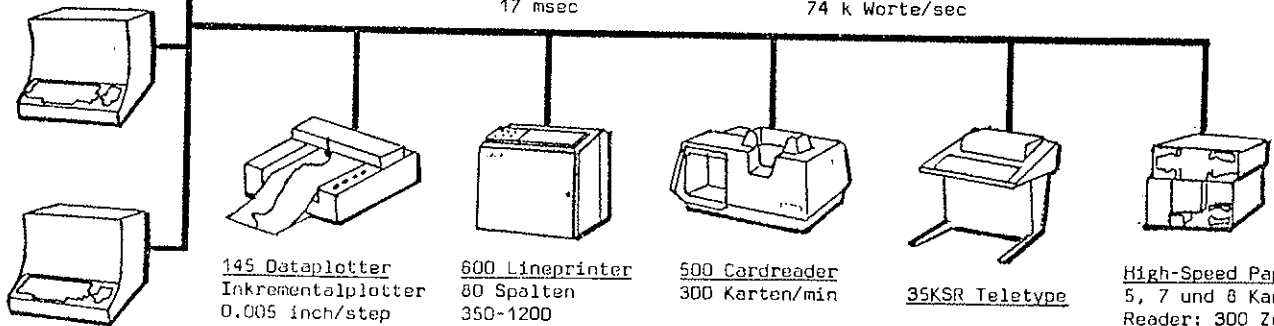
direkter Speicherkanal



Standard-
I/O Kanal

1260 Data Disc
Festkopfplatte
360 k Worte
mittlere Zugriffszeit:
17 msec

1272 Pertec Disc
Wechselplatte
2,2 mill Worte
Transferrate:
74 k Worte/sec



145 Dataplotter
Inkrementalplotter
0,005 inch/step
450 steps/sec

600 Lineprinter
80 Spalten
350-1200
Zeilen/min

500 Cardreader
300 Karten/min

35KSR Teletype

High-Speed Paper Tape
5, 7 und 8 Kanäle
Reader: 300 Zeichen/sec
Punch: 120 Zeichen/sec

EAI C109 Datensichtgeräte

64 Zeichen
21 Zeilen à 80 Zeichen
4800 baud asynchron

ARBEITSBEREICHE AN DER HYBRIDRECHENANLAGE

Auf dem Gebiet der SYSTEMPROGRAMMIERUNG wurde im letzten Jahr von Dipl.Ing. A. Blauensteiner das völlig neue Echtzeit- und multiprogrammings Betriebssystem JCS/MP entwickelt. An der Wartung, Dokumentation und Weiterentwicklung dieses Systems wird ständig gearbeitet.

Darüber hinaus wird laufend an der Verbesserung der Sprachprozessoren gearbeitet (Herr F. Blöser).

Der TECHNISCHE BETRIEB, der die Aufrechterhaltung des Rechenbetriebes besorgt und für die Unterstützung des Benützers am Rechner verantwortlich ist, wird von Herrn M. Schandl aufrecht erhalten.

An der Wartung und ständigen Verbesserung der HARDWARE, sowie an Neuentwicklungen spezieller Geräte (z.B. opt. Abtaster) wird laufend gearbeitet.

Dem Rechenzentrum steht der VERWALTUNGs-fachmann Herr W. Hassa zur Verfügung.

Die Aufgaben der ORGANISATION werden von Herrn H. Hummer wahrgenommen.

Auf dem Gebiet der PROGRAMMIERUNG wird an der Erstellung einer effektiven Programmbibliothek gearbeitet (Frau I. Husinsky). In dieser Hinsicht kann auf die Zusammenarbeit mit dem Institut für Numerische Mathematik verwiesen werden (Dipl.Ing. C. W. Überhuber). Herr W. Theischinger ist mit der Erstellung und der Auswertung von hybriden Testprogrammen betraut.

Die Hybridrechenanlage wird als wissenschaftlicher Dienstleistungsbetrieb geführt. Unser Ziel ist es, den Benützern nicht nur eine direkte Programmunterstützung und -beratung zukommen zu lassen, sondern auch soweit als möglich bei der Analyse und Lösung ihrer Probleme zu unterstützen.

Nach telefonischer Terminvereinbarung bei Herrn Hummer (tel. 65-37-85 901 oder 803/DW), der den Benutzer an den kompetenten Mitarbeiter weiterleitet, kann diese Beratung in Anspruch genommen werden.

Da aufgrund des bestehenden Personalmangels an der Hybridrechenanlage vielleicht nicht immer jemand zur Beratung zur Verfügung stehen kann, soll auch auf die KURSE hingewiesen werden, die an der Hybridrechenanlage abgehalten werden, deren Besuch das Programmieren am Hybridrechner erleichtern soll.

In diesem Zusammenhang weisen wir auf eine Folge von Programmierkursen hin, die vom Arbeitsbereich "Regelungstheorie und Hybridrechentechnik" des I. Instituts für Mathematik abgehalten werden. Diese Kurse führen in die Grundlagen der Programmier- und Anwendungsmöglichkeiten des Analog- und Hybridrechners ein.

PROGRAMMIERBERATUNG UND PROGRAMMBIBLIOTHEKSBERATUNG:

Montag - Freitag 11 - 12 Uhr
Zimmer 1403

Personal der Abt. Hybridrechenanlage

		Zimmer	Klappe
Leitung:	KLEINERT Wolfgang, Dipl.Ing.	1418	831
Mitarbeiter:	BLAUENSTEINER Albert, Dipl.Ing.	1403	832
	BLÖSER Friedrich	1401	803
	HUMMER Helmut	1402	901
	HUSINSKY Irmgard	1403	832
	SCHANDL Manfred	1404/05	803
	THEISCHINGER Walter	1402	901
Rechenraum	EAI PACER 600	1404/05	803

KURSE

Die Kurse werden aufbauend gehalten, d.h. es wird von Fragen von allgemeinem Interesse zu speziellen Problemen fortgeschritten, sodaß ein Kursbesucher, der nur einige allgemeine Informationen benötigt, auch bei vorzeitiger Beendigung des Kurses die für ihn notwendigen Informationen erhält.

Folgende Kurse werden im kommenden Wintersemester 1974/75 an der Hybridrechenanlage abgehalten:

- RH1 GERÄTETECHNIK EAI PACER 600
Zentraleinheit, Interruptsystem, I/O System (inclusive Interface und Analogrechner)
Dauer: 1 Woche
Termin: voraussichtlich im Dezember, der genaue Termin wird noch rechtzeitig bekanntgegeben.
- RH2 BENÜTZUNG DES BETRIEBSSYSTEMS JCS/MP
FORTRAN-Kenntnisse Voraussetzung
Termin: 21.10.1974 bis 25.10.1974, täglich 2 Stunden
21.10.1974 9 Uhr Besprechung, Festlegung der Vortragszeiten
Ort: 1040 Wien, Gußhausstraße 27-29
4.Stock Seminarraum (Zimmer 1415) linke Tür
Vortragender: Dipl.Ing.A. Blauensteiner
- RH3 BESONDERHEITEN IN FORTRAN AN DER HYBRIDRECHENANLAGE
1 Vortrag
wird nach Bedarf gehalten,
die Ankündigung erfolgt im Schaukasten der Hybridrechenanlage
- RH4 HYBRIDE FORTRAN-PROGRAMMIERUNG
mit Übungen
Einführung in das Hybrid Linkage Softwarepaket
Dauer: 1 Woche
Termin: voraussichtlich November, der genaue Termin wird noch rechtzeitig bekanntgegeben

- RH5 EAI-TERMINALBEDIENUNG
Dauer: 1 Woche
Termin: voraussichtlich Jänner, der genaue Termin
wird noch rechtzeitig bekanntgegeben
- RH6 EAI-ASSEMBLER
mit Übungen
Termin: 7.10.1974 bis 11.10.1974, täglich 9-12 und 13-16 Uhr
Ort: 1040 Wien, Gußhausstraße 27-29
4. Stock Seminarraum (Zimmer 1415) linke Tür
Vortragender: Dipl.Ing.A. Blauensteiner

Anmeldung telephonisch oder persönlich bei Herrn Schandl,
Rechenraum Zimmer 1404/05, tel. 65-37-85/803 DW.

Vom I. Institut für Mathematik werden folgende Kurse abgehalten:

- AH1 PROGRAMMIEREN VON ANALOGRECHNERN
- AH2 PROGRAMMIEREN VON HYBRIDEN ANALOGRECHNERN
Dauer: 1 Woche
Termin: in den Semesterferien: 17.2.1975 bis 21.2.1975
Genaueres wird noch bekanntgegeben
- AH3 VERFAHREN DER HYBRIDEN PROGRAMMIERUNG
- AH4 HYBRIDE PROGRAMMIERSPRACHEN
Termine im Sommersemester
Für das Verständnis der Kurse AH3 und AH4 sind Grund-
kenntnisse in der Programmierung von Analogrechenanlagen
erforderlich.

Nähere Auskünfte können bei Frau Prof. Dr. I. Troch (Klappe 836)
bzw. Herrn Dipl.Ing. F. Grabner (Klappe 829) eingeholt werden.

BENÜTZUNG DES BETRIEBSSYSTEMS JCS/MP 6

BESCHREIBUNG

JCS/MP 6 ist ein multiprogramming Betriebssystem für Stapelverarbeitung. Ein Supervisor ermöglicht die Kommunikation mit dem System. Für Benutzerprogramme stehen zwei Level zur Verfügung (Level 1, Level 2), von denen einer höhere Priorität besitzt (Level 1). Auf dem Level 3 werden Programme compiliert und gebunden. Jedes Benutzerprogramm kann prinzipiell auf jedem der beiden Level rechnen. Jobs auf Level 1 (Main-Jobs) können die Exekution des Hintergrundjobs (Back-Jobs) stoppen. Dies kann softwaremäßig oder durch den Operator geschehen. Hybridprogramme werden daher sinnvollerweise auf Level 1 gerechnet, wobei in zeitkritischen Phasen der Hintergrundbetrieb (Level 2, Level 3) gestoppt wird.

Im Multiprogrammingbetrieb stehen dem Hybridbenutzer auch jederzeit das gesamte hybride Interruptsystem sowie der DMA zur Verfügung. Es besteht ferner die Möglichkeit, ein Programm jederzeit zu unterbrechen und zu einem beliebigen Zeitpunkt wieder fortzusetzen. Ein Programm, das im Multiprogramming läuft, sollte nicht mehr als 12 k 16-bit Worte Speicherplatz benötigen (größere Programme siehe unten).

Es können nur Programme, die in FDRTRAN oder Assembler oder HOI geschrieben sind, gerechnet werden.

BENÜTZERPROGRAMM (FORTRAN)

1. Jedes Programm muß als erstes ausführbares Statement ein

```
CALL JDB (NR,ITIME)
```

besitzen. Auf dem ersten Parameter NR muß die Jobnummer des Benützers angegeben werden. Auf dem zweiten Parameter ITIME muß die maximale Laufzeit des Programmes in Minuten definiert werden. Diese Schranke soll im Sinne des Rechenbetriebs möglichst klein gehalten werden. Für den Closed-Shop-Betrieb sollte ITIME nicht größer als 10 sein. Nach der definierten Anzahl von Minuten wird das Programm automatisch abgebrochen.

```

S55555555 777 777          00000000 00000000      1111
S555555555 7777 7777      0000000000 0000000000      111111
S55 7777 7777 7777 7777 000 000 000 000 11111111
S55 77777777 77777777 000 000 000 000 111111 111
S555555555 777777 777777 0000000000 0000000000      111
S555555555 777 777 777 777 000 000 000 000 111
S5555555555 777 777 777 777 000000 00000000 0000000000      111
S555555555 777 777 777 777 0000000000 0000000000      111

777 00000000 555555555 777 777 777 777 777 777 777 777
777 0000000000 5555555555 777 777 777 777 777 777 777 777
777 000 555 777 777 777 777 777 777 777 777 777 777 777
777 000 555555555 777 777 777 777 777 777 777 777 777 777
777 000 555555555 777 777 777 777 777 777 777 777 777 777
777 777 777 777 777 777 777 777 777 777 777 777 777 777
7777777777 00000000 555555555 777 777 777 777 777 777 777
7777777777 00000000 555555555 777 777 777 777 777 777 777

7777777777 444 444 000000 0000000 22222222 000000
7777777777 444 444 0000 0000 0000 0000 222 222 222 222 222
7777 444 444 000 000 000 000 222 222 222 222 222 222 222
7777 4444444444 000 000 00000000 22222 00000000
7777 4444444444 000 000 00000000 22222 00000000
7777 444 000 000 000 22222 0000 0000
7777 444 000 000 000 2222 0000 0000
7777 444 000 000 000 2222 0000 0000
7777 444 00000000 00000000 2222222222 00000000
7777 444 00000000 00000000 2222222222 00000000

1111 5555555555 22222222 000000 77777777 77777777
111111 5555555555 2222222222 00000000 77777777 77777777
11111111 555 222 222 222 222 222 222 222 222 222 222 222
111111111 555 222 222 222 222 222 222 222 222 222 222 222
111 111 5555555555 22222 00000000 7777777777 7777777777
111 111 5555555555 22222 00000000 7777777777 7777777777
111 111 555 2222 0000 0000 777777 777777 777777 777777
111 111 555 2222 0000 0000 777777 777777 777777 777777
111 111 5555555555 44444 2222222222 7777777777 7777777777
111 111 5555555555 444 2222222222 00000000 7777777777

```

Bei Aufruf des Unterprogramms JOB wird die Verrechnung der Exekutionszeit des laufenden Programms gestartet. Außerdem wird am Lineprinter eine Titelseite ausgedruckt, die zur Identifikation des Outputs dient.

2. Jedes Programm muß durch die Exekution des Statements

CALL EXIT

beendet werden. Dadurch wird die Kontrolle wieder an den Supervisor zurückgegeben und der Stapelbetrieb fortgesetzt. Ferner wird die benötigte Exekutionszeit ermittelt und abgespeichert.

Der Aufruf von EXIT bewirkt auch einen Ausdruck am Lineprinter, wo Jobnummer, Exekutionszeit, Gesamtjobanzahl und Gesamtrechnenzeitverbrauch dieser Jobnummer, sowie das Datum, die Uhrzeit und der Name des Operators, der für die Durchführung dieses Programms verantwortlich war, ausgegeben werden.

Zusätzlich zum programmierten Ausstieg hat der Operator die Möglichkeit, ein Programm jederzeit zu beenden.

PROGRAMME MIT GROSSEM SPEICHERBEDARF

Programme, die mehr als 12 k 16-bit Worte Speicherplatz benötigen (maximal 24 k), können eventuell außerhalb des Multiprogrammingbetriebs gerechnet werden. Günstiger für den Ablauf des Stapelbetriebs ist es jedoch, von der Möglichkeit Gebrauch zu machen, größere Programme in Teilen hintereinander zu exekutieren.

Falls am logischen Ende eines Programms statt dem CALL EXIT ein

CALL LINK

steht, wird das laufende Programm abgeschlossen und das Programm, dessen Name von der nächsten Steuerkarte für denselben Level (MAIN oder BACK) eingelesen wird, wird als nächstes Programm auf diesem Level exekutiert.

Daten können über COMMON-Blöcke übergeben werden.

DATENFILES AUF PLATTE

JCS/MP 6 ermöglicht das Benützen von Datenfiles auf einer fixed-head Platte und einer Wechselplatte. Spezielle, von FORTRAN aufrufbare Unterprogramme dienen zum Definieren von Datenfiles, zum Lesen und zum Schreiben.

Zur genaueren Information über das Programmieren von Plattenfiles sind an der Hybridrechenanlage Unterlagen erhältlich.

VIRTUELLE FELDER

Virtuelle Felder können ebenfalls durch FORTRAN-aufrufbare Unterprogramme auf der Platte errichtet werden.

Zur genaueren Information über das Programmieren von virtuellen Feldern sind an der Hybridrechenanlage Unterlagen erhältlich.

KOMMUNIKATION MIT DEM ANALOGRECHNER

Es steht ein Softwarepaket zur Kommunikation mit dem Analogrechner in einem hybriden Programm zur Verfügung.

Beschreibungen des Softwarepaketes (Hybrid Linkage) sind an der Hybridrechenanlage erhältlich.

INPUT/OUTPUT, EINHEITENNUMMERN (FORTRAN)

JCS/MP 6 arbeitet mit einem eigenen, auf Interruptbasis gestellten I/O Driver, der vom laufenden Programm unabhängig die Ein/Ausgabe von Daten bewerkstelligt.

Folgende Peripherie ist an JCS/MP 6 angeschlossen:

	FORTRAN Einheitennummern	
	Input (READ)	Output (WRITE)
Video	26	26
Teletype	2	1
Lineprinter	-	6
Card Reader	5	-
Paper Tape	3	7

ZEITKRITISCHE HYBRIDPROGRAMME

In zeitkritischen Programmen kann man durch den Aufruf

CALL LEVEL (I)

die niedrigst erlaubte Priorität (I) verändern. Läuft ein Programm auf Level 1 und wird CALL LEVEL (1) exekutiert, so wird die Exekution auf Level 2 und 3 solange unterbrochen, bis z.B. durch CALL LEVEL (3) wieder alle 3 Level arbeiten dürfen.

		Programm läuft auf					
		Level 1 (Main)			Level 2 (Back)		
I		Level			Level		
		1	2	3	1	2	3
1		E	S	S	E	E	S
2		E	E	S	E	E	S
3		E	E	E	E	E	E

E ... Level ist erlaubt

S ... Level ist gesperrt nach dem Aufruf CALL LEVEL (I)

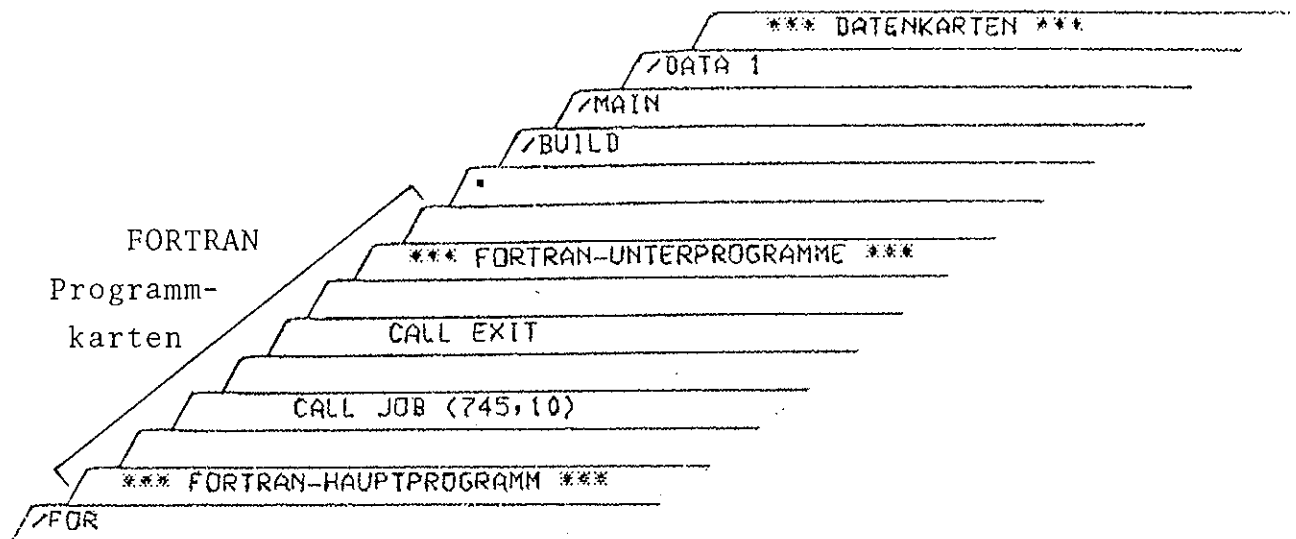
SYSTEM UNTERSTÜTZUNGSPROGRAMME

In JCS/MP 6 stehen folgende, von FORTRAN aufrufbare Unterprogramme zur Verfügung:

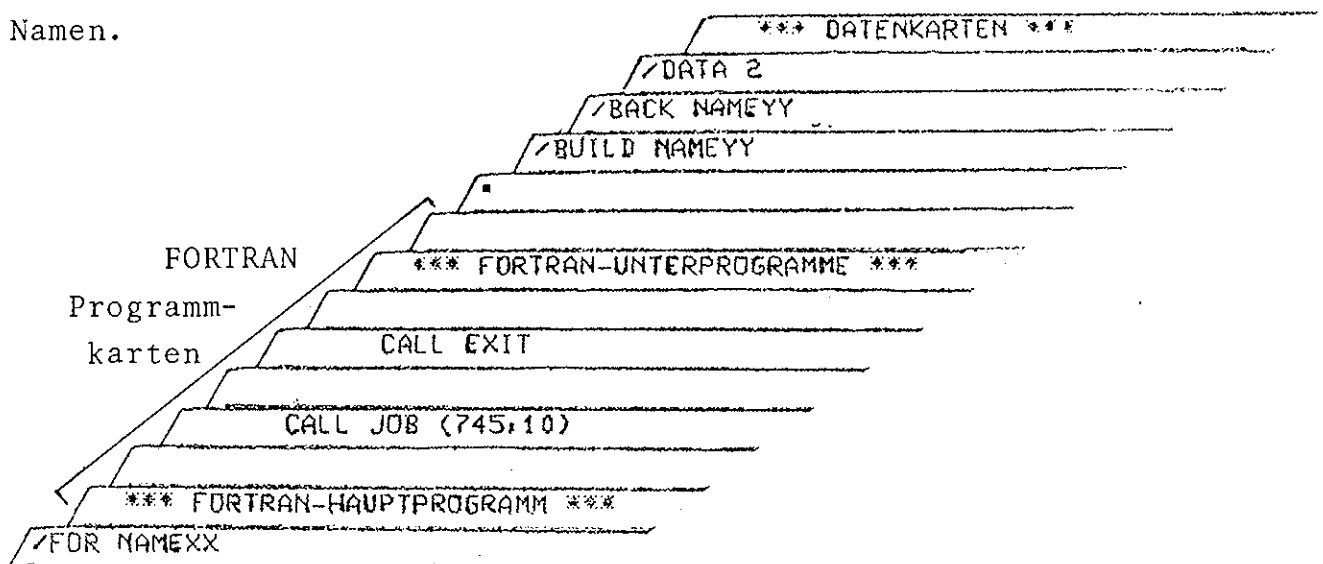
Name	Aufruf, Bedeutung	Eingangsparameter	Ausgangsparameter
HOURL	CALL HOURL (I,J) Angabe der Tageszeit	-	I - INTEGER, Stunde J - INTEGER, Minute
TIME	TIME (I) Angabe der verbleibenden Rechenzeit	I - INTEGER, keine Bedeutung	TIME - REAL Function, verbleibende Rechenzeit in Sekunden
DATUM	CALL DATUM (I,J,K) Angabe des Datums	-	I - INTEGER, Jahr J - INTEGER, Monat K - INTEGER, Tag
PAGE	CALL PAGE Seitenvorschub am Lineprinter	-	-
LINES	CALL LINES (I) Zeilenvorschub am Lineprinter	I - INTEGER, Anzahl der Zeilen	-

STEUERKARTEN IN JCS/MP 6

Beispiel: Steuerkarten für ein FORTRAN-Programm, das auf Level 1 (Main) gerechnet 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. Er muß aus 6 alphanumerischen Zeichen bestehen.

Für Assemblerprogramme ist statt der /FOR Karte eine /ASM Karte zu verwenden. Nach jeder /MAIN oder /BACK Karte muß unbedingt eine /DATA w Karte folgen, wobei w den Level angibt, auf dem das Programm läuft (1 Main, 2 Back). Die /DATA w Karte muß auch dazugelegt werden, wenn keine Datenkarten vorhanden sind.

SPRACHPROZESSOREN

FORTRAN

Der FORTRAN-Compiler ist ein äußerst schneller Übersetzer von FORTRAN-Programmen. Er erstellt das entsprechende Objekt-Programm durch einen Durchlauf (One-pass Compiler). Der erzeugte Maschinencode ist daher nicht optimiert. Der Programmierer sollte selbst darauf achten, daß z.B. in Schleifen keine unnötigen Zuweisungen stehen und ähnliches mehr. Der Compiler benötigt ungefähr 10 k Worte Speicherplatz, er wurde in Assembler von EAI geschrieben und an der Hybridrechenanlage erweitert und adaptiert. Segmentation, Overlaying und virtuelles Paging sind geplant.

Es besteht die Möglichkeit, auf Kosten zusätzlichen Speicherbedarfs Integer und Floating Point Operationen direkt im Maschinencode zu compilieren (statt über Standard Unterprogrammaufrufe). Dies erhöht wesentlich die Exekutionsgeschwindigkeit bzw. ermöglicht ein Reentrant-Programmieren mit Interruptprogrammen.

Dabei gelten folgende Richtwerte für die Exekutionszeiten (in μsec):

	INTEGER oder SCALED FRACTION	REAL
Load/Store	2	15
Addition	2	15
Subtraktion		
Multiplikation	6	20
Division	6.6	22

DOUBLE PRECISION-Operationen werden softwaremäßig realisiert, es ist daher mit etwa den 20-fachen Rechenzeiten gegenüber REAL-Operationen zu rechnen.

Der FORTRAN-Compiler verarbeitet in-line die wichtigsten Assemblerbefehle. Das ermöglicht dem fortgeschrittenen FORTRAN-Programmierer in zeitkritischen Programmabschnitten Assemblerstatements direkt in sein FORTRAN-Programm einzufügen.

HOI

HOI - Hytran Operations Interpreter - ist ein Compiler, der sowohl interaktive Befehle einzeln übersetzt und exekutiert, als auch Programmblöcke als gesamtes verarbeiten kann. Der Benutzer steht mit HOI über ein Terminal in dialogem Kontakt. Der Benutzer kann beliebige Befehle hintereinander durchführen wie bei einem Tischrechner, bzw. Programme beliebig exekutieren, stoppen, verbessern und neu starten.

Beispiele:

Tischrechner:

```
User:  2 + 2:
HOI    2 + 2 = 4.0000
```

arithmetischer Ausdruck

Hybridprogramm testen:

```
User  VAL(@A014):
HOI   VAL(@A014) = .6237
```

Auslesen eines Verstärkers

Programm:

```
User  1.01) I = 1, 1, 10! J = J + I
      1.02) J:
      J = 0
      1;
HOI   J = 55.0000
```

$$\sum_{i=1}^{10} i$$

HOI speichert Programmschritte Schritt für Schritt in "Source" im Kernspeicher ab, d.h. erst bei der Exekution wird ein entsprechender Befehl compiliert und durchgeführt. Dadurch ist ein problemloses on-line Austauschen von Befehlen und Programmteilen möglich und der Benutzer hat jederzeit Zugriff und Kontrolle der Variablen.

Da HOI in erster Linie zum Testen analoger Schaltungen und zum Auswerten der Algorithmen eingesetzt wird, ist der Nachteil der langsamen Exekutionszeit nicht schwerwiegend, insbesondere da HOI im Hintergrund zu anderen Programmen laufen kann. Zur Durchführung eines automatischen Statischen Tests für Analogschaltungen stehen in der Programmbibliothek leicht adaptierbare HOI-Programme zur Verfügung. HOI-Programme können auch von Karten eingelesen werden, Daten auch am Lineprinter ausgegeben werden. Ein umfangreiches Befehlsrepertoire ermöglicht ein komfortables Arbeiten.

HOI wurde von EAI entwickelt und benötigt ohne Buffer für Variablen und Programmspeicherung 7 k Worte.

EAI-ASSEMBLER

Der maschinenorientierte EAI-Assembler besitzt Befehle, die das mnemotechnische Kodieren der verdrahteten Instruktionen des PACER 100 Digitalrechner ermöglichen. Von diesen Befehlen sind 27 echte Verarbeitungsbefehle, 36 ermöglichen Skips und Jumps und 29 dienen der allgemeinen Kontrolle.

Dem Programmierer stehen 2 arithmetische Register (Akkumulator und dessen Erweiterung) und ein Indexregister zur Verfügung. Jeder Befehl kann maximal auf eine Kernspeicherzelle zugreifen (Einadressmaschine) und unterscheidet 8 verschiedene Adressierungsarten, wobei ein Indirekt-Level definiert ist, der über beliebig viele Stufen programmierbar ist. Zur leichteren Programmgestaltung stehen eine Reihe von Pseudobefehlen zur Verfügung.

Zur Erstellung des Objekt-Programms wird der Assembler zweimal durchlaufen. Der EAI-Assembler ist selbst in seinem Assembler geschrieben wurde von EAI entwickelt und an der Hybridrechenanlage erweitert und adaptiert.

BESONDERHEITEN DES EAI-FORTRAN

An der Hybridrechenanlage ist Standard FORTRAN ASA IV mit logischem IF, labelled COMMON-Blöcken, BLOCK DATA usw. implementiert. Folgende Einschränkungen und Besonderheiten sind zu beachten:

INTEGER-Konstante müssen aus dem Bereich $(-32768, 32767)$ sein, bedingt durch die Darstellung in einem 16-bit Wort.

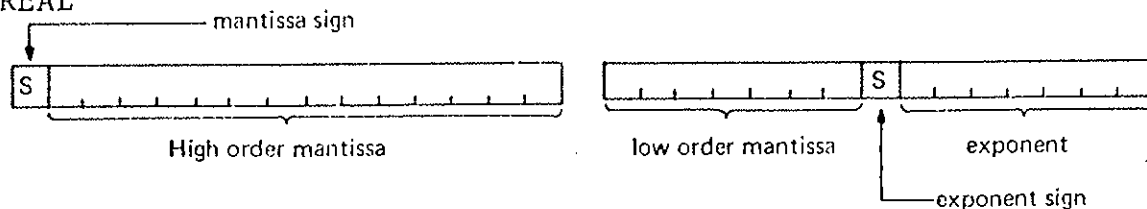
INTEGER



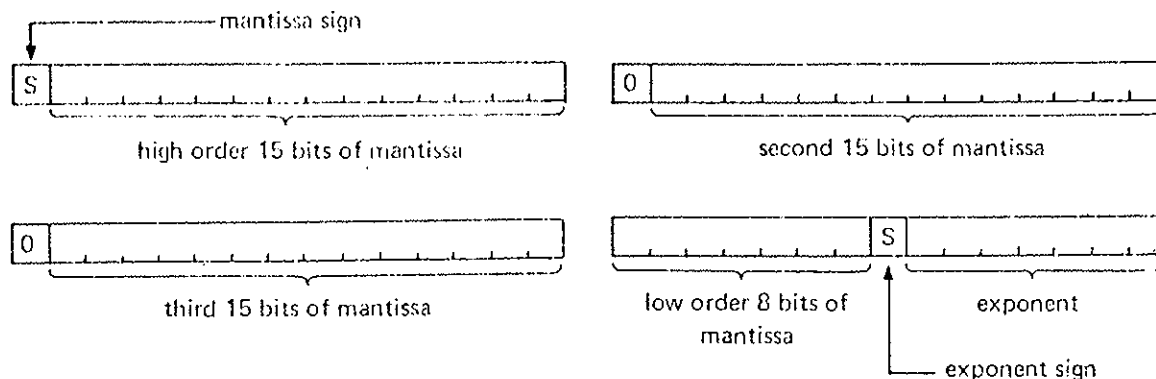
INTEGER-Konstante können auch als Oktal- oder Hexadezimalzahlen definiert werden. Sie werden gekennzeichnet durch ein vorangestelltes Hochkomma bzw. zwei Hochkomma.

REAL-Größen werden auf zwei, DOUBLE PRECISION- und COMPLEX-Größen auf vier 16-bit Worten dargestellt und haben einen absoluten Zahlenbereich von 10^{-38} bis 10^{38} .

REAL

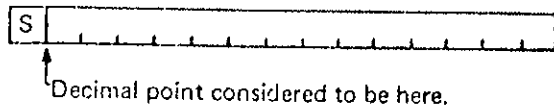


DOUBLE PRECISION



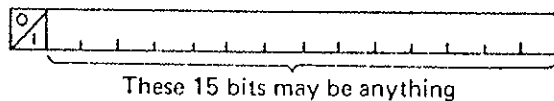
SCALED FRACTION-Konstante (Festkommakonstante), die speziell in der hybriden Programmierung Verwendung finden, müssen betragsmäßig kleiner als 1 sein. Sie werden wie Real-Zahlen mit einem angehängten S geschrieben. (z.B. .55S)

SCALED FRACTION



LOGICAL Variable und Konstante verwenden nur das Vorzeichenbit. Die logischen Operationen werden aber hardwarebedingt für alle 16 bit durchgeführt, d.h. daß auch diese bits verändert werden. Damit können z.B. Maskierungen direkt in FORTRAN vorgenommen werden.

LOGICAL



Hollerithkonstante können durch ein Apostroph am Ende der Zeichenkette vorzeitig abgeschlossen werden (z.B. 20HABC' entspricht 3HABC).

Es besteht die Möglichkeit, entsprechend den SCALED FRACTION-Konstanten, Variable zu deklarieren, und zwar in der Form

$$\text{SCALED FRACTION } v_1, v_2, \dots, v_n$$

Die zugehörige Formatangabe für Ein- oder Ausgabe hat die Form Sw, wobei w die Feldlänge inclusive Vorzeichen und Dezimalpunkt angibt.

In einem DATA-Befehl darf die Variablenliste nur aus einfachen Variablen oder speziellen Feldelementen bestehen. Der Name eines Feldes ohne Indexangabe ist unzulässig. Soll also ein Feld durch einen DATA-Befehl initialisiert werden, so sind alle Elemente des Feldes separat anzuführen:

```
DIMENSION Y(3)
DATA Y(1),Y(2),Y(3)/1.,2.,3./
```

Alphanumerische Daten können auf Variable nur nach Maßgabe des zugehörigen Speicherplatzes gespeichert werden. Es können auf INTEGER-, LOGICAL- und SCALED FRACTION-Größen maximal zwei, auf REAL-Größen maximal vier und auf COMPLEX- bzw. DOUBLE PRECISION-Größen maximal acht Zeichen gespeichert werden. Es ist unzulässig, mehr Zeichen in einem DATA-Befehl anzugeben. Werden weniger Zeichen angegeben, so wird rechtsbündig mit Leerzeichen aufgefüllt.

Als Indices bei Feldelementen sind folgende arithmetische Ausdrücke möglich, wobei c , k für INTEGER-Konstante, v für eine INTEGER-Variable steht:

c , v , $c*v$, $v\pm k$, $c*v\pm k$

Der Befehl PAUSE hat außer dem Ausdruck PAUSE am Lineprinter für die Programmdurchführung keine Bedeutung.

Jedes Programm muß als erstes exekutierbares Statement den Aufruf des Unterprogramms JOB besitzen (siehe auch Benützung des Betriebssystems JCS/MP 6).

Der Befehl STOP darf nicht zur Beendigung eines Programms verwendet werden. Programme müssen mit einem CALL EXIT oder CALL LINK enden (siehe auch Benützung des Betriebssystems JCS/MP 6).

Als letztes Statement einer DO-Schleife ist ein IF-Statement nicht zulässig. Ein Unterprogrammaufruf ist hingegen erlaubt.

Folgende Formate können zur Ein- und Ausgabe verwendet werden:

A, D, E, F, G, H, I, L, O, P, S, X

Es besteht die Möglichkeit, in zeitkritischen Programmen Teile des Programms direkt im Assembler zu codieren.

Mixed Modes:

erlaubt		nicht erlaubt					
I ^I	R ^R	L ^L	D ^D	C ^C	C ^C	D ^D	
R ^I	D ^R	I ^L	C ^L	I ^D	I ^C		
D ^I	R ^D	L ^L	I ^R	L ^D	L ^C		I INTEGER
C ^I	D ^D	R ^L	L ^R	C ^D	R ^C		R REAL
S ^I	S ^R	S ^D	S ^C	S ^L			D DOUBLE PRECISION
S ^I	S ^S	D ^S	R ^S	C ^S	L ^S	I ^S	S SCALED FRACTION
I = I	I = R	I = C	R = L	C = D			C COMPLEX
R = R	R = I	I = S	C = S	L = S	D = S		L LOGICAL
C = C	R = D	L = I	D = L				
L = L	D = R	S = I	S = C	S = L	S = D		
D = D	I = D	L = R	C = L				
C = R	D = I	L = D	R = C				
S = S	S = R	L = C	D = C				
L + L	R = S	I = L	C = I				
R + R	R + D	S + I	S + C	S + L	S + D		
I + I	D + R	L + I	R + L	I + E	C + D		
C + C	C + R	L + R	D + L	R + I			
D + D		L + D	C + L	D + I	R + S		
D + S	S + R	L + C	I + R	C + I			
		I + L	I + D	D + C			
		I + S	C + S	L + S	D + S		

ebenso für -, *, /

PROGRAMMBIBLIOTHEK

Ein wichtiger Aspekt der Kundenbetreuung besteht in der Bereitstellung getesteter Software. Neben der ständigen Entwicklung besserer Betriebssysteme sowie der Compiler-Wartung wird dem Aufbau einer Bibliothek von technisch-wissenschaftlichen Anwendungsprogrammen an der Hybridrechenanlage besondere Aufmerksamkeit gewidmet.

WAS SIND DIE AUFGABEN EINER PROGRAMMBIBLIOTHEK ?

a.) Aufwand reduzieren

Dem Benutzer soll geholfen werden, mit (oft wesentlich) weniger Aufwand zu denselben (oder besseren) Resultaten zu gelangen, die er sonst mit selbstgeschriebenen Programmen berechnen müßte. Die Erleichterung für den Benutzer besteht bezüglich des Schreibens, Testens und der Dokumentation der Programme.

b.) Qualität erhöhen

Es wird versucht, solche Programme aufzunehmen, die bezüglich Zuverlässigkeit, Schnelligkeit, Speicherbedarf usw. besser sind als jene Programme, wie sie von einem Benutzer einer technisch-wissenschaftlichen Rechenanlage (der kein hauptberuflicher Programmierer oder Mathematiker ist) geschrieben werden.

WIE VERSUCHEN WIR, DIESE AUFGABEN ZU ERFÜLLEN ?

Das Grundprinzip beim Aufbau unserer Programmbibliothek besteht darin, Fremdprogramme nicht kritiklos zu übernehmen, sondern nach entsprechenden Test- und Bewertungskriterien eine Auswahl der besten Programme zu treffen, die nach entsprechender Aufbereitung (Vereinheitlichung, Dokumentation usw.) den Benutzern zur Verfügung gestellt werden.

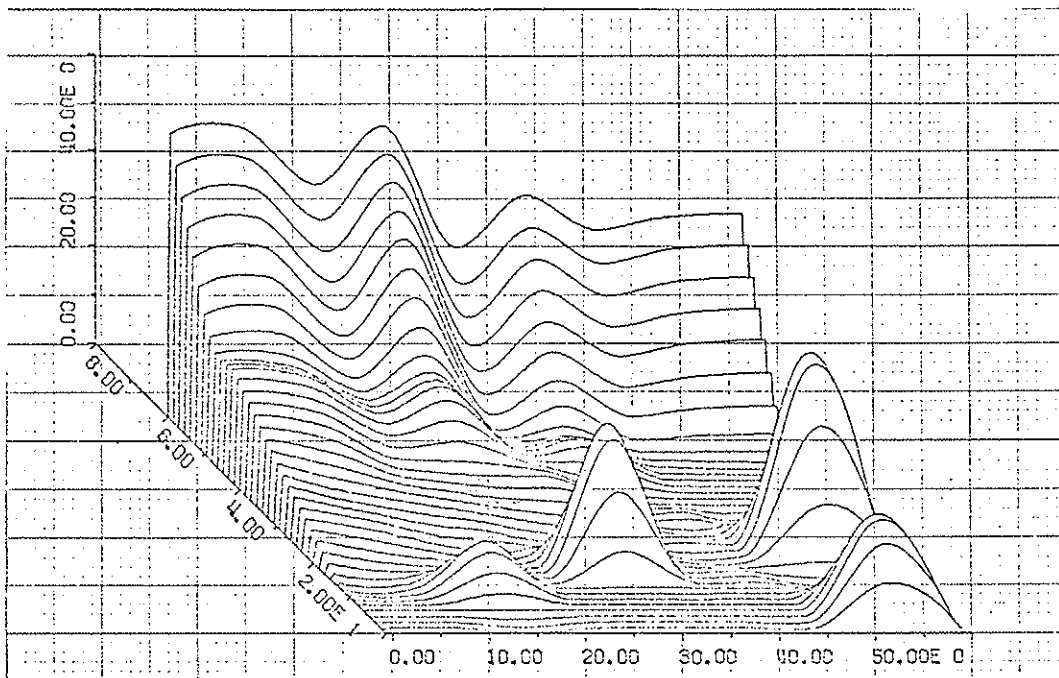
Folgendes Beispiel soll die positiven Auswirkungen, die dieser Auswahlvorgang für den Benutzer hat, verdeutlichen.

So konnte das Programm AGINT, das an der Hybridrechenanlage zur digitalen Auswertung bestimmter Integrale verwendet wird, einen aus der einschlägigen Literatur bekannten Satz von 21 Testfunktionen mit der geforderten Genauigkeit (10^{-6}) integrieren, während das Programm FROMBG auf der

CYBER 74 in 10 Fällen versagte, obwohl die Berechnung dort mit der doppelten Anzahl von Dezimalstellen erfolgt.

Die bisherigen Schwerpunkte unserer Arbeit lagen auf der Entwicklung eines Softwarepaketes zur komfortablen Bedienung des Analogrechners und des Interfaces, sowie auf der Implementierung von Integrationsverfahren für gewöhnliche Differentialgleichungen.

Einer der derzeitigen Schwerpunkte liegt auf der Entwicklung der Software für den Increment Plotter (siehe z.B. untenstehende Abbildung) und den dazugehörigen Approximations- und Interpolationsprogrammen.



WIE KANN DER BENÜTZER EINFLUSS DARAUF NEHMEN, DASS DIE PROGRAMMBIBLIOTHEK SEINEN ANFORDERUNGEN ENTSPRICHT ?

Unseren Vorstellungen entsprechend erfolgt die Weiterentwicklung der Programmbibliothek auf Grund eines Rückkopplungseffektes mit dem Kreis der Benutzer. Wir laden die Benutzer herzlich dazu ein, auf diesen Entwicklungsprozeß Einfluß zu nehmen, indem sie ihre Wünsche und Vorstellungen im direkten Gespräch mit Angehörigen der Hybridrechenanlage oder in der Interessentenversammlung bekanntgeben.

Der unmittelbare Bedarf an Programmen, die derzeit nicht verfügbar sind, wird soweit als möglich aus bestehenden fremden Bibliotheken (wie z.B. SSP, BMD, C.ACM-Algorithmen, EAI Users Library usw.) gedeckt.

Falls die Probleme von allgemeinem Interesse sind, wird eine Erweiterung der Bibliothek in der vorgeschlagenen Richtung sofort in Angriff genommen.

WIE WIRD DER BENÜTZER ÜBER DIE LEISTUNGEN INFORMIERT ?

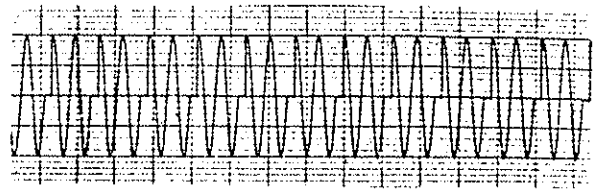
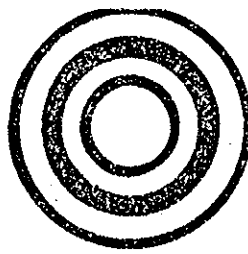
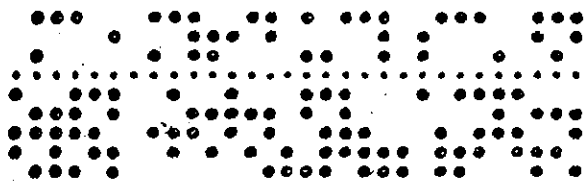
- Bei der Hybridrechenanlage werden ständig Anschläge über neu aufgenommene Programme, sowie Listen aller verfügbaren Programme ausgehängt.
- In der individuellen Programmberatung können genauere Informationen über alle verfügbaren und in Entwicklung befindlichen Programme eingeholt werden.
- Über ausgewählte Neuzugänge der Bibliothek wird laufend im 'Interface' berichtet werden.

Das FORTRAN-Unterprogrammpaket HYBRID LINKAGE

beinhaltet eine Reihe von FORTRAN-aufrufbaren Unterprogrammen, die zur komfortablen Bedienung des Analogrechners und des Interfaces für den Hybridbenützer dienen.

So ist z.B. eine einfache Handhabung der Übertragung von Zahlenwerten vom Analogrechner zum Digitalrechner und umgekehrt sowohl über den Standard I/O Kanal als auch über einen direkten Speicherkanal möglich. Es gibt Unterprogramme zur Bedienung von Interrupts und zum Übertragen von logischen Signalen. Ferner ist auch die Bedienung des Analogrechners vom Digitalrechner möglich (z.B. Potentiometersetzen, Auslesen eines Verstärkers etc.).

Kurzbeschreibungen der HYBRID LINKAGE Unterprogramme sind in der Programmberatung erhältlich.



BENÜTZERFORUM

An dieser Stelle sollen jene Jobnummernbesitzer, welche an der Hybridrechenanlage wissenschaftliche Arbeiten (Diplomarbeiten, Dissertationen etc.) ausführen, zu Wort kommen.

Es wird versucht werden, die Artikel in einer Form zu bringen, welche es auch den auf dem speziellen Gebiet nicht so versierten Kollegen ermöglicht, davon zu profitieren und Neues über Einsatzgebiete und Arbeitsweise von hybriden Rechenanlagen zu erfahren. Damit ist insbesondere die weitgehende Verwendung von Fluß- und Ablaufdiagrammen an Stelle detaillierter Lösungsbeschreibungen gemeint, wodurch die Arbeitsteilung zwischen Analogrechner und Digitalrechner sowie die Vorteile, die durch den Einsatz eines Hybridsystems entstehen, besonders deutlich hervorgehoben werden können.

Jene Jobnummernbesitzer, deren Arbeiten ein "druckreifes" Stadium erreicht haben und zur Mitarbeit bereit sind, werden freundlich eingeladen, sich bei der Abt. Hybridrechenanlage zu melden (Herr Hummer, Klappe 901).

INTERESSENTENVERSAMMLUNG

Um einen nutzbringenden Kontakt zwischen den Benützern und den Angehörigen der Hybridrechenanlage zu aktivieren, werden von Zeit zu Zeit Versammlungen abgehalten werden, bei denen die Kunden ihre Wünsche, Anregungen und Beschwerden vorbringen sollen. Außerdem können alle, die ein spezielles Problem oder allgemeines Interesse haben, bei diesen Versammlungen über die Einsatzmöglichkeiten des Hybridrechners informiert werden.

Nächster Termin:

8. 11. 1974 14 UHR

Ort: 1040 Wien
Gußhausstraße 27-29
Seminarraum (Zimmer 1415)
4. Stock

SIMULATION VON EINSCHWINGVORGÄNGEN IN HALBLEITER-LASERN

Dr. S. Gründorfer

Einleitung

Das große Interesse für Halbleiter-Laserdioden liegt u.a. in ihrer möglichen Anwendung in der optischen Nachrichtenübertragung. Die Laserdioden vereinen hohen Wirkungsgrad, Kompaktheit und robuste Konstruktion mit großer Einfachheit und Vielseitigkeit hochfrequenter direkter Modulation. Entscheidend für die Einschätzung der Anwendungsmöglichkeiten auf diesem Gebiet ist genaueste Untersuchung und Verstehen der zeitabhängigen Prozesse in diesen Lasern, wie Einschwingvorgängen am Beginn des Lichtimpulses, langen Zeitverzögerungen zwischen dem Einschalten des Stromimpulses und dem Einsetzen der stimulierten Emission und Q-Switching-Pulsen nach Abschalten des Stromimpulses.

Alle diese Effekte sind in ihrer unkontrollierten Form offenkundig unerwünscht. Um sie verhindern oder kontrolliert anwenden zu können, ist es daher außerordentlich wichtig, eine einheitliche Theorie dafür zu entwickeln.

Die hohe Konzentration flacher Donatoren in Halbleiterlasern erzeugt ein Störstellenband knapp unterhalb des Leitungsbandes. Die so erzeugten "Bandschwänze" sind von spezieller Wichtigkeit für die Berechnung aller Lasereigenschaften. In früheren Arbeiten über lange Zeitverzögerung und Q-Switching wurde daher ein Modell zugrunde gelegt, das die Formierung solcher Bandschwänze berücksichtigt.

Zeitbilanzgleichungen

Im folgenden wird ein System von 2 Zeitbilanzgleichungen, wie es zur Beschreibung einfacher Einschwingvorgänge benötigt wird, auf der Basis des Lasermodells unter Berücksichtigung der Existenz von Bandschwänzen näher untersucht. Es werden Bandschwänze mit einer Gauß'schen Zustandsdichteverteilung

$$\rho_s(E) = \rho_0 \exp\left[-\frac{(E-E_D)^2}{\eta_D}\right] \quad (1)$$

angenommen. Dabei ist η_D die Halbwertsbreite, ρ_0 die Amplitude und E_D die Energie des Zustandsmaximums des Gauß'schen Bandschwanzes, die etwa der Ionisierungsenergie der flachen Donatoren entspricht.

Eine bestimmende Funktion nicht nur für die zeitabhängigen Erscheinungen sondern auch für den Zusammenhang zwischen der Verstärkung und dem Strom beim Schwellwert, für die Abhängigkeit der Schwellstromdichte von der Temperatur, der Dotierung usw. ist die Verstärkungsfunktion $f(n)$, die proportional dem Maximum der wellenlängenabhängigen Verstärkung ist.

Beim untersuchten Modell wurde angenommen, daß an der stimulierten Emission nur die Elektronen im Bandschwanz, n_s , beteiligt sind. Die Verstärkungsfunktion wird daher proportional

n_s angenommen. Man erhält dabei für den Ein-Möden-Betrieb folgende Zeitbilanzgleichungen:

$$\frac{dn}{dt} = P - \frac{n}{\tau_e} - An_s N \quad (2)$$

$$\frac{dN}{dt} = An_s N - \frac{N}{\tau_p} + r_{sp} \quad (3)$$

Dabei ist n die El.konz. im Leitungsband der aktiven Zone des Lasers, N die Photonendichte in der oszillierenden Mode, P die Pumprate (die im Falle von Injektionslasern durch den Injektionsstrom bestimmt wird), τ_e die Lebensdauer der Elektronen im Leitungsband und Bandschwanz (für nichtstrahlende wie für strahlende spontane Rekombination), τ_p die Lebensdauer der Photonen in der oszillierenden Mode (die durch optische Resonatorverluste bestimmt wird). r_{sp} ist die spontane Emission bei der Wellenlänge der oszillierenden Mode, die aber üblicherweise gegenüber der stimulierten Emission vernachlässigt werden kann, allerdings bei der numerischen Berechnung der Bilanzgleichungen durch geeignete Randbedingungen berücksichtigt werden muß.

Der Zusammenhang zwischen der Gesamtelektronen-Konzentration n und der Konzentration der Elektronen im Bandschwanz n_s ist über eine Integralbeziehung gegeben, die auch die Fermi-Funktion $f(E, E_F)$ mit der Temperatur T und dem Fermi-Niveau E_F als Parameter enthält. Es ist

$$n = \int \rho(E) f(E, E_F) dE \equiv I_2(E_F) \quad (4)$$

$$n_s = \int \rho_s(E) f(E, E_F) dE \equiv I_3(E_F) \quad (5)$$

dabei ist $f(E, E_F) = [1 + \exp(\frac{E-E_F}{kT})]^{-1}$ die Fermi-Funktion

und $\rho(E) = \text{Max}\{\rho_s(E), \rho_B(E)\}$

mit $\rho_B(E) = \rho_L E^{1/2} \quad (6)$

als Zustandsdichte des ungestörten Leitungsbandes.

Mit $\frac{dn}{dt} = \frac{dn}{dE_F} \cdot \frac{dE_F}{dt} \equiv I_1(E_F) \cdot \frac{dE_F}{dt} \quad (7)$

lassen sich dann für die Annahme Gauß'scher Bandschwänze Gl.(2),(3) als

$$\frac{dE_F}{dt} = \frac{1}{I_1(E_F)} \left[P - \frac{I_2(E_F)}{\tau_e} - AI_3(E_F)N \right] \quad (8)$$

$$\frac{dN}{dt} = AI_3(E_F)N - \frac{N}{\tau_p} + r_{sp} \quad (9)$$

schreiben.

Numerische Methode

Die Lösung der gekoppelten Integrodifferentialgleichungen (8) und (9) wurde am Hybridrechner durchgeführt. Dabei wurden die Differentialgleichungen mit dem Unterprogramm GBS, einem rationalen Extrapolationsalgorithmus, digital gelöst, wobei bei jedem Funktionsaufruf die bestimmten Integrale I_1 , I_2 , I_3 am Analogrechner ausgewertet wurden.

Resultate

Man erhält auf diese Weise ein Bild der Einschwingvorgänge, wie für einen typischen Fall in Fig.1 dargestellt ist. Durch Veränderung der einzelnen interessierenden Parameter kann man leicht deren Einfluß auf die Zeitverzögerung bis zum Einsetzen der stimulierten Emission, auf die Dämpfungszeit und Frequenz des Einschwingvorganges und auf die Höhe des stationären Lichtimpulses feststellen.

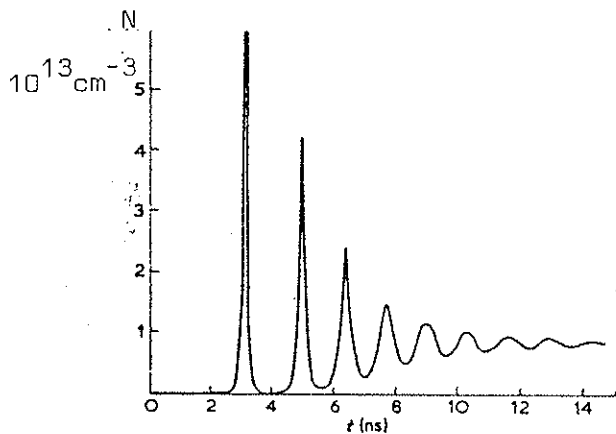


Fig. 1

Im Gegensatz zu anderen Modellen, in denen nur der Einfluß der beiden Lebensdauern τ_e und τ_h , sowie der Pumprate P auf die Einschwingvorgänge untersucht werden kann, ist beim untersuchten Modell auch der Einfluß der Donatorkonzentration N_D , die mit $\mu_D \propto N_D^{5/12}$ und $\rho_D \propto N_D^{7/12}$ in die Parameter des Gauß'schen Bandschwanzes eingeht, deutlich zu ersehen.

Die Zeit t_D , die notwendig ist, um die Ladungsträgerinversion herzustellen, nimmt bei konstanter Pumprate mit zunehmender Konzentration ab (Fig.2), ebenso die Dämpfungszeit des Einschwingvorganges, während die Frequenz dieser anfänglichen Oszillationen mit höherer Dotierung zunimmt. Die Intensität des Lichts nach Erreichen des stationären Zustands wird jedoch dadurch nicht wesentlich beeinflusst.

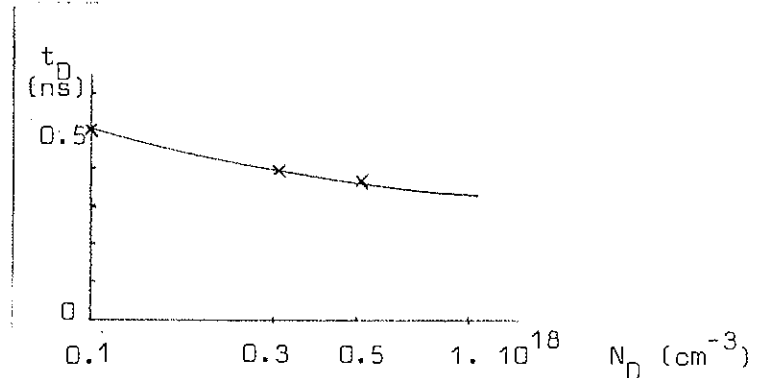


Fig. 2

Zusammenfassung

Ein System von 2 gekoppelten Zeitbilanzgleichungen zur Untersuchung zeitabhängiger Effekte in Halbleiter-Lasern wurde aufgestellt und am Hybridrechner gelöst, unter Berücksichtigung der Verschmierung der Donatorkonzentration im stark dotierten Material und der Formierung von Gauß'schen Bandschwänzen. Das Modell wurde aufgestellt unter der Annahme, daß nur die Elektronen im Bandschwanz zur stimulierten Emission beitragen und verwendet sonst keine speziellen Annahmen über die Verstärkungsfunktion. Der Vorteil dieses Modells liegt darin, daß durch die angegebene hybride Lösung der Zeitbilanzgleichungen der Einfluß der Dotierung und anderer Materialeigenschaften auf die Einschwingvorgänge leicht untersucht werden kann. Außerdem ist das Modell durch Hinzunahme einer dritten Zeitbilanzgleichung leicht erweiterbar für die Untersuchung von langen Zeitverzögerungen oberhalb der Übergangstemperatur und für Q-Switching-Effekte.

Dr. S. Gründorfer
Institut für Angewandte Physik, Universität Wien
und Ludwig Boltzmann Institut für Festkörperphysik
Wien

UNTERSUCHUNGEN AN EINEM REGLERMODELL

F. Dozler
 Institut für Regelungstechnik
 Kl. 556

In der Regelungstechnik treten häufig Probleme folgender Art auf:
 Aus einer Problemstellung der Industrie, Reaktortechnik oder z.B. auch der Kraftwerksregelung ergibt sich die Forderung, die von n Parametern abhängige Zustandsgröße des Regelsystems über ihre Parameter so zu beeinflussen, daß sie bestimmte Bedingungen erfüllt.

In der Praxis ist das z.B. dann möglich, wenn man ein Funktional f, eine sogenannte "Gütefunktion" oder "Kriterium" definieren kann, welche ihr Minimum erreicht, wenn die geforderte Bedingung erreicht ist. Das Problem reduziert sich dann durch diese stetige lineare Abbildung in den R^n auf die Suche des Minimums in einem n-dimensionalen Gütegebirge. Meist ist aber nicht nur das Minimum interessant, sondern auch besonders die "Linien gleicher Güte" in dessen Nähe, die sogenannten "Isokriterialen".

Im Rahmen eines Seminars sowie wissenschaftlicher Arbeiten an der Hybridrechenanlage wurde nun ein Satz von Programmen erstellt, die es ermöglichen, dieses Problem hybrid zu lösen.

Als Eingabe muß die diskretisierte Gütefunktion an jeder Stelle des Gütegebirges (a_1, a_2, a_3, \dots) verfügbar sein, wobei a_1, a_2, a_3, \dots die Parameter des entsprechenden Systems sind.

Als Gütefunktion verwendet man hier das Funktional
$$\int_0^T e^2 dt, \quad 0 < T \leq \infty$$

wobei $e(a_1, a_2, a_3, \dots)$ eine Fehlerfunktion ist.

Die zu optimierende Zustandsgröße ist im allgemeinen die Lösung einer Differentialgleichung, die Fehlerfunktion e und insbesondere das Integral sind numerisch schwer auszuwerten, weil die Größe des Kernspeichers bei Diskretisierung kritisch wird, wenn eine gute Genauigkeit verlangt wird.

Die Differentialgleichung samt der Auswertung der Gütefunktion wird also am besten am Analogrechner gesteckt. Andererseits wäre es sehr zeitintensiv und vor allem von einem unvertretbaren Elementararbeitsaufwand begleitet, würde man versuchen, das Minimum rein analog zu finden (ab etwa 3 Parametern sicher unvorteilhaft), besonders auch, was das Aufsuchen der Isokriterialen betrifft. Für die automatische Untersuchung einer Klasse von Systemen bietet sich also ein Hybridrechner an.

Die Verbindung zwischen dem digitalen Programm und dem Analogsystem wird durch eine "FUNCTION" im Sinne von FORTRAN dargestellt, welche als Parameter die

Parameter der zu optimierenden Zustandsgröße hat. Sie bewerkstelligt das Einstellen der Parameter am Analogrechner über DAC's, startet den Analogrechner und damit die Lösung der Differentialgleichung sowie die Auswertung der Gütefunktion (Integral) und übernimmt nach der Zeit T den Wert der Gütefunktion über einen ADC. Je nach der Strategie des Minimum-Suchprogrammes wird die FUNCTION nun mehr oder weniger oft mit verschiedenen Parameterwerten aufgerufen, bis das Minimum mit einer sinnvoll vorgegebenen Genauigkeit erreicht ist. Es werden nun die Werte der gefundenen Parameter und der Wert der Gütefunktion am Lineprinter sowie eventuell relevante Kennwerte oder Zustandsgrößen am Plotter ausgegeben. Nun werden mit einem zweiten Programm durch vom Minimum ausgehendes Abtasten Punkte im Parameterraum gesucht, für die $f/f_{\min} = \text{const.}$ gilt.

Nach eventuellem Smoothing und Interpolation können die Isokriterialen als geschlossene Kurven am Digitalplotter ausgegeben werden.

Im jetzigen Stadium der Arbeit werden diese Kurven punktweise am Analogplotter ausgegeben. Die Werte für "const" können dabei beliebig in Prozent eingegeben werden.

Diese Methode wurde nun in unserem Fall auf ein Identifikationsproblem und ein Reglersyntheseproblem angewandt, welche in den folgenden Abbildungen dargestellt sind.

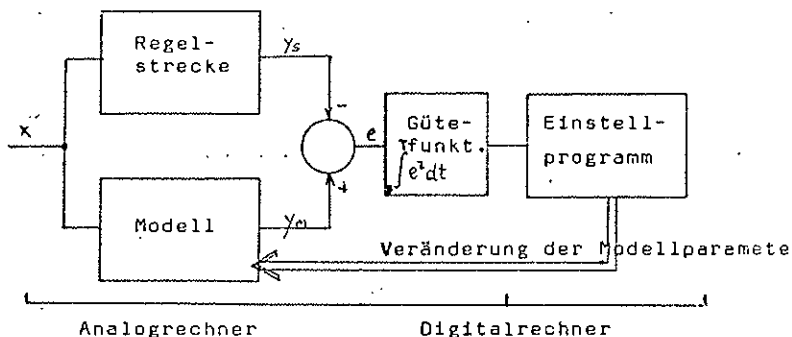


Abb.1 Identifikation mittels Parallelmodell

Abb. 1 zeigt die schematische Darstellung für eine Identifikation mittels Parallelmodell, wobei die Struktur des Modells durch eine allgemeine Differentialgleichung 2. Ordnung

$$\frac{V}{(1+T_1s)(1+T_2s)} = \frac{d\{Y_m\}}{d\{X\}} = \frac{Y_m}{X}$$

mit reellen Polen T_1, T_2 vorgegeben ist. Die Parameter V, T_1 und T_2 sind so zu finden, daß der Streckenausgang y_s dem Modellausgang y_m bei gleichem Eingang x möglichst ähnlich wird, ähnlich in dem

Sinne, daß das Funktional T

$$\int_0^T (y_m - y_s)^2 dt$$

minimal wird.

Ist die Stationärverstärkung der Strecke (Ausgangswert bei statischem Eingang) bekannt bzw. direkt meßbar, so kann V fest gewählt werden und es verbleiben T_1 und T_2 als Parameter.

Ein zweites Anwendungsbeispiel der anfangs besprochenen Hybrid-Methode war die Erstellung eines PI-Reglers für eine Strecke 4. Ordnung, sodaß das Integral über das Quadrat der Regelabweichung (ISE-

Kriterium): T

$$\int_0^T e^2 dt \text{ minimal wird.}$$

Das zugehörige Blockschaltbild ist in Abb. 2 angegeben. Zunächst wurden hier die optimalen Reglerparameter für verschiedene Objektzeitkonstanten T_0 und dann die Brauchbarkeit der Gütefunktion T

$$\int_0^T (e^2 + \tau^2 e^2) dt$$

für verschiedene τ untersucht.

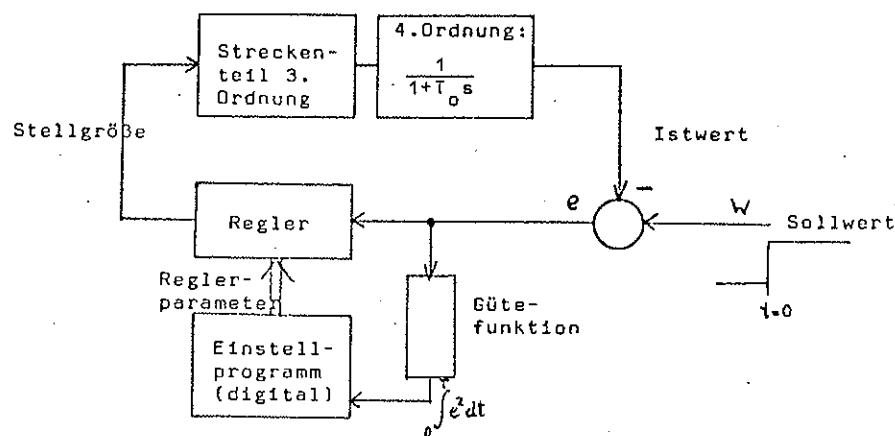


Abb. 2 Reglersynthese durch Parameteroptimierung

AUSWAHL VON DERZEIT LAUFENDEN ODER KÜRZLICH ABGESCHLOSSENEN FORSCHUNGSVORHABEN AN DER HYBRIDRECHENANLAGE

- Korrelationsuntersuchungen von Enzephalogrammen von Kaninchen
- Tragfähigkeit von Ebenen in Abhängigkeit von den Auflagepunkten (part. Differentialgl.)
- Energieoptimale Steuerung eines Erzentladers
- Fixpunkte von Hammerstein-Operatoren
- Transienter Betrieb von Schrittmotoren
- Sekundärströmungen in gekrümmten Rohren
- Periodische Lösungen von autonomen und nichtautonomen nichtlinearen Differentialgleichungssystemen
- Simulation sozioökonomischer Modelle
- Analyse und Diagnose von Elektrokardiogrammen für Reihenuntersuchungen (Screening)
- Analyse und Synthese von Musik und Sprache
- Optimierung von linearen Regelsystemen mit Hilfe stückweise konstanter Steuerfunktionen

