
Interface

herausgegeben von der
Abt. Hybridrechenanlage des
EDV-Zentrums der
Technischen Universität Wien

Nummer 12
November 1978



Analoge Simulation und Darstellung
einer bewegten Glocke

INHALTSVERZEICHNIS

	Seite
Hybridrechenanlage bekommt AutoPATCH-Erweiterung	3
EAI Users Group Meeting 1978	5
Aktuelle Mitteilungen	6
Kurse	8
Neue Assembler-Skripten	10
Fragen? Probleme?	11
IMACS Symposium	12
Arbeitsweise und Verwendung der Rep-op Timer an der Hybridrechenanlage	15
Modemverbindungen an der Hybridrechenanlage	20
Rechnen mit Interruptunterprogrammen	23
Graphische Datenverarbeitung	26
Analoge Simulation der mechanischen Vorgänge beim Läuten einer Glocke	30
Bildhafte Darstellung eines Laufkrans am Oszillographen	33
Periodensuche und Periodenanalyse von veränderlichen Sternen	36

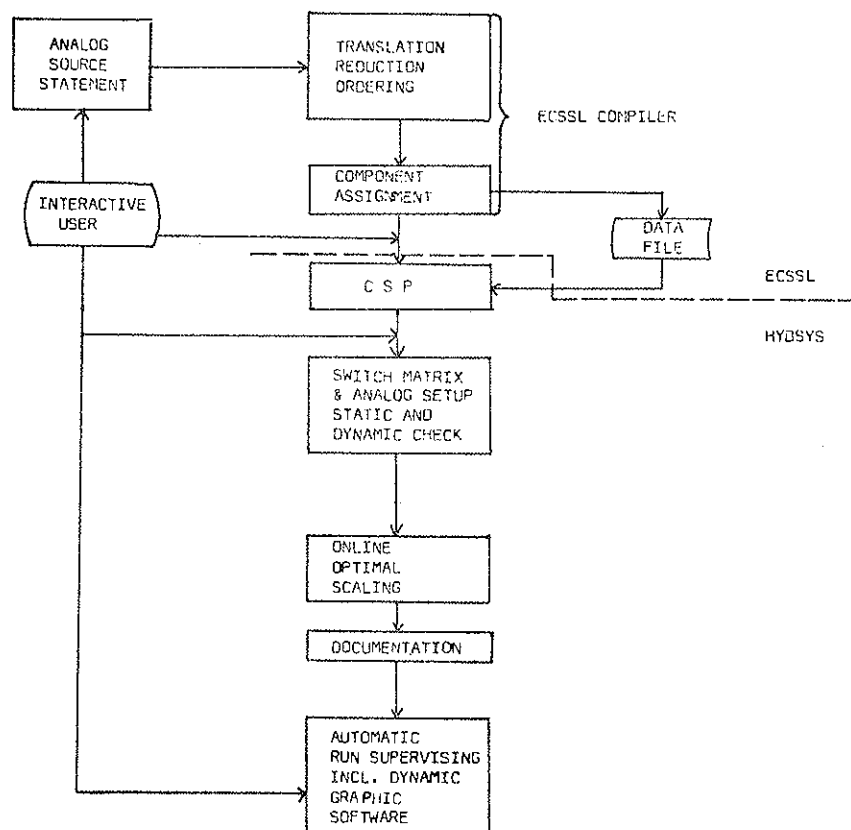
Redaktion: Irmgard Husinsky

Eigentümer, Herausgeber, Verleger: EDV-Zentrum der Technischen Universität Wien, Abteilung Hybridrechenanlage, Vervielfältigung: Österreichische Hochschülerschaft Technik, für den Inhalt verantwortlich: Dipl.Ing. W. Kleinert, alle: Gußhausstraße 27-29, 1040 Wien
Telex: 76875 rzthw a

Hybridrechenanlage bekommt AutoPATCH-Erweiterung !

Nach einer mehr als eineinhalb Jahre dauernden Planungs- und Vorbereitungsphase konnte Anfang Oktober das Bundesministerium für Wissenschaft und Forschung die Genehmigung zur Vertragsunterzeichnung über eine AutoPATCH-Erweiterung der Hybridrechenanlage EAI PACER 600 an der TU Wien erteilen.

Noch heuer wird EAI den hybriden Compiler ECSSL liefern, der vorerst auf der CYBER 74 laufen soll. Die analoge 192×320 Schaltmatrix soll samt den dazugehörigen Controllern im Frühjahr 1979 installiert werden. Die Implementierung der gesamten AutoPATCH-Software - das sind neben dem hybriden Compiler vor allem der blockungsfreie Schaltungserstellungsalgorithmus (CSP = Connection Statement Processor) sowie die Programmmoduln zum Laden und Testen der Matrix - am PACER 600 wird in Kooperation mit der Firma EAI von der Abteilung Hybridrechenanlage des EDV-Zentrums der TU Wien durchgeführt werden.



Programmmoduln der AutoPATCH-Software an der TU Wien

Die Abbildung zeigt das Ablaufdiagramm einer hybriden Simulation unter Verwendung des TU AutoPATCH Systems. Die Problemeingabe erfolgt über Karten oder Terminal in der gleichungsorientierten ECSSL-Syntax. Der Compiler übersetzt, reduziert und ordnet die Gleichungen und führt eine Zuordnung der Gleichungsstrukturen zu den AutoPATCH "Macros" (einem neuen Typ analoger Rechenelemente, wobei mehrere Verstärker und Logikelemente zu einem "Macro" zusammengefaßt sein können) durch. Das Ergebnis der Compilation ist ein Datenfile mit der Struktur der Schaltung, zu dem das an der Hybridrechenanlage entwickelte Programmsystem HYBSYS (siehe INTERFACE Nr. 11) Zugriff hat. HYBSYS verarbeitet die Schaltstruktur zu direkten Schaltbefehlen für die Schaltmatrix und generiert alle anderen notwendigen Informationen zum Einstellen des Analogrechners. Die Durchführung eines statischen oder dynamischen Tests sowie einer optimalen on-line Skalierung kann ebenso auf Wunsch des Benutzers erfolgen wie das automatische Zeichnen des Schaltplanes oder die automatische Laufüberwachung der hybriden Simulation inklusive graphischer Dokumentation.

Aus finanziellen Gründen kann vorerst die geplante Ersetzung der Logik durch einen PLP (= Parallel Logic Processor) nicht durchgeführt werden, es wird aber alles für seinen nachträglichen Einbau vorbereitet. Noch nicht entschieden wurde über andere Ersatzmöglichkeiten für die parallele Logik, wie deren Simulation durch einen extrem schnellen sequentiellen 3-Adressrechner oder den Aufbau von Hardwarelogikmoduln. Vorerst ist daran gedacht, das AutoPATCH-Steckbrett mit festverdrahteten Logikprogrammen für Standardanwendungen auszustatten.

Über den weiteren Ausbau des AutoPATCH-Systems durch eine time-sharing-fähige Software für hybriden Multi-Userbetrieb und eine spezielle, an der TU zu entwickelnde Hardware werden wir in einer der nächsten Ausgaben des INTERFACE berichten.

At the beginning of October, after more than one and a half years of planning and preparation, the Austrian Ministry for Science and Research consented to the signing of the contract for an AutoPATCH-expansion of the EAI PACER 600 installation at the Hybrid Computation Centre of the Technical University of Vienna. The hybrid compiler ECSSL will be supplied by EAI this year, the switch matrix will be installed in spring 1979. The diagram shows the program moduls of the AutoPATCH software at the TU Vienna.

W. Kleinert

EAI USERS GROUP MEETING 1978

In Barcelona trafen einander im Oktober 49 Benützer von EAI-Rechenanlagen aus 9 europäischen Ländern und den USA. Die Teilnehmer berichteten über den Betrieb und die Entwicklungen an ihren Rechenanlagen. Dabei zeigte sich das Bestreben, vorhandenen Devices den Zugriff von mehreren Rechnern zu ermöglichen. Dadurch, sowie aus Gründen der Entlastung einer CPU für Real-Time Prozesse wurden wieder Rechnerkoppelungen vorgestellt. Konkretisiert wurden diesmal bereits Pläne für den Einsatz von Microprozessoren, deren zunehmende Bedeutung unübersehbar ist, wenngleich die Anwendungen unterschiedlichster Art sind.

Die TU Wien präsentierte einen Film über den grundlegenden Aufbau des Betriebssystems JCS/TS 7 und einen Überblick über die bevorstehende, weltweit erstmals projektierte Installation eines EAI AutoPATCH-Systems sowie die damit zusammenhängende, zum Teil bereits durchgeführte, zum Teil in Arbeit befindliche Entwicklung auf den Gebieten der Hard- und Software.

EAI berichtete über ihre Produkte sowie über die Verteilung der etwa 750 installierten Hybridrechner, wobei besonders der hohe Anteil an Installationen an Universitäten und ähnlichen Institutionen auffiel.

A. Blauensteiner

aktuelle mitteilungen

ÖFFNUNGSZEITEN:

Die Hybridrechenanlage ist von

Montag bis Freitag

von

8 Uhr bis 20 Uhr

geöffnet. Die Closed-Shop-Eingabe ist zu jeder vollen Stunde.

PERSONELLES:

Anstelle von Herrn Helmut Hummer, der in den nächsten 8 Monaten den Zivildienst absolviert, ist Frau Elisabeth Berger als Programmiererin an der Hybridrechenanlage angestellt. Herr Hummer ist jedoch nach Vereinbarung (für Benützerauskünfte) erreichbar.

NEU: 4-QUADRANTEN-DCA'S

VORTEILE DER VERWENDUNG VON DCA'S

Seit Juli 1978 ist der Analogrechner EAI 680 mit 20 4-Quadranten-DCA's ausgerüstet. Sie befinden sich in den Potpositionen 0 - 19. Die bisher schon vorhandenen 2-Quadranten-DCA's belegen jetzt die Potpositionen 20 - 39.

Diese 4-Quadranten-DCA's sind digital setzbare Potentiometer, die sowohl auf positive als auch auf negative Werte gesetzt werden können, während die 2-Quadranten-DCA's hardwarebedingt nur auf positive Werte gesetzt werden können. Beide Arten von DCA's führen intern eine Vorzeichenumkehr durch.

Diese Vorzeichenumkehr wird jetzt von den entsprechenden Routinen der Hybrid Linkage Software berücksichtigt, sodaß die in den Unterprogrammen angegebenen Potentiometerwerte tatsächlich den Werten entsprechen, mit denen multipliziert wird. Bei den 2-Quadranten-DCA's müssen diese Werte also immer negativ sein. Gegenüber den Servopotentiometern bringen die DCA's folgende entscheidende Vorteile:

- das Setzen eines DCA (FORTRAN-Aufruf) dauert nur ca. 17 μ sec (Servopotentiometer bis zu 2 sec!)
- DCA's können unabhängig vom Mode des Analogrechners gesetzt werden.

MASSENSPEICHERERWEITERUNG GEPLANT:

Im November 1978 wird anstelle der defekten Fixed Head Disk eine weitere Dual-Magnetplattenspeichereinheit mit einer fest eingebauten und einer auswechselbaren Platte (je 1143K Worte) installiert werden. Das wird eine Vergrößerung der Districts im Betriebssystem JCS/TS 7 mit sich bringen und somit mehr Platz zum Abspeichern von Daten- und Programmfiles für die Benutzer.

kurse

Folgende Kurse werden im Wintersemester 1978/79 an der Hybridrechenanlage abgehalten werden:

RH1 GERÄTETECHNIK EAI PACER 600

Zentraleinheit, Interruptsystem, I/O System (inklusive Interface, Analogrechner und AutoPATCH)

Dauer: 3 Tage

Termin: 1979/01/15 bis 1979/01/17

Vortragender: Dipl.Ing. W. Kleinert

RH2 BENÜTZUNG DES BETRIEBSSYSTEMS JCS/TS 7

Optimale Ausnützung der Möglichkeiten des Systems, Systemstruktur

Dauer: 1 Woche

Termin: 1978/11/13 bis 1978/11/17

Vortragende: Dipl.Ing. A. Blauensteiner und Dipl.Ing.F.Blöser

RH5 PROGRAMMIERUNG IN HOI

mit Übungen

Dauer: 2 Tage

Termin: 1978/11/30 bis 1978/12/01

Vortragender: Dipl.Ing. W. Kleinert

RH6 EAI ASSEMBLER

mit Übungen

Dauer: 1 Woche

Termin: 1979/01/22 bis 1979/01/26

Vortragender: Dipl.Ing. F. Blöser

RH7 SOFTWAREUNTERSTÜTZUNG FÜR DIE BENÜTZUNG DES PACER 600 ALS PLOTTER SYSTEM

Dauer: 2 Tage

Termin: 1978/11/07 bis 1978/11/08

Vortragender: Dipl.Ing. F. Blöser

RH9 EINFÜHRUNGSKURS IN DIE HYBRIDE PROGRAMMIERUNG

Dieser Kurs wird nach Bedarf, mindestens aber alle zwei Monate für wissenschaftliche Benützer, die keinerlei Vorkenntnisse auf dem Gebiet der hybriden Programmierung besitzen und an einer möglichst raschen Problemlösung interessiert sind, abgehalten. Der Kurs ist als Anregung zum Selbststudium gedacht.

Dauer: 2 Tage

Vortragender: D. Solar

Nähere Auskünfte bei Herrn D. Solar (1040 Wien, Gußhausstraße 27-29, 4.Stock, Zimmer 1401, Tel.: 65-37-85/747 DW).

Der genaue Kursort sowie die Kurszeiten werden noch rechtzeitig im Schaukasten der Hybridrechenanlage sowie durch eine Aussendung von Einladungen bekanntgegeben.

Nähere Auskünfte und Anmeldungen zu den Kursen telefonisch oder persönlich bei Herrn M.Schandl (1040 Wien, Gußhausstraße 27-29, 4.Stock, Zimmer 1404/05, Tel.: 65-37-85/803 DW).

Vom Institut für Technische Mathematik werden folgende Kurse abgehalten:

AH1 AH2 PROGRAMMIEREN VON ANALOGRECHENANLAGEN

Grundlagen und Anwendungsmöglichkeiten des Analogrechners mit praktischen Übungen an den Rechnern EAI-18Ø und EAI-68Ø

Termin: 1979/01/29 - 1979/02/02

AH3 AH4 HYBRIDE SPRACHEN UND VERFAHREN

Grundprinzipien und Einsatzgebiete des Hybridrechnens, aufbauend auf Vorkenntnisse über Analogrechnen, wie sie in den Kursen AH1 und AH2 vermittelt werden. Simulationssprache CSMP, hybride Programmiersprache HOI, FORTRAN-Hybrid-Ergänzungen.

Termin: 1979/02/19 - 1979/02/23

Auskünfte und Anmeldungen zu den Kursen bei Dipl.Ing. Frank Rattay (Klappe 899), Institut für Technische Mathematik, Gußhausstraße 27-29, 1040 Wien).

NEUE ASSEMBLER-SKRIPTEN:

Zur Programmierung im EAI-Assembler sind nun zwei Skripten erhältlich:

EAI Assembler I

Grundlagen der Assemblerprogrammierung

Dieses Skriptum ist eine Überarbeitung des bisherigen Assemblerskriptums und dient zum Erlernen des Assemblers im Rahmen des Betriebssystems JCS/TS 7. Viele Beispiele machen die Anwendung der einzelnen Befehle deutlich.

EAI Assembler II

Hinweise zur fortgeschrittenen Programmierung

Dieses Skriptum ist neu und enthält Hinweise zur Ausnützung sämtlicher Eigenschaften des EAI Assemblers für eine optimale und effektive Programmierung in Hinsicht auf Kernspeicherplatzersparnis, Rechenzeitreduzierung und Programmübersichtlichkeit. Außerdem wird auf spezielle Programmstrukturen, die innerhalb des Betriebssystems JCS/TS 7 möglich sind, sowie auf Interruptprogrammierung eingegangen.

Durch umfassende Befehls- und Stichwortverzeichnisse haben sich die Skripten zum Nachschlagen für den Assemblerprogrammierer bewährt. Anhand von Programmbeispielen wird die praktische Anwendung des Assemblers demonstriert.

Die Skripten sind zum Preis von je S 40,-- in der Programmberatung erhältlich.

I. Husinsky

fragen ? probleme ?

Für Auskünfte und Unterstützung bei Problemen steht Ihnen täglich die Programmberatung zur Verfügung. Bei tieferliegenden Fragen bzw. Schwierigkeiten bitten wir Sie, sich direkt an den zuständigen Mitarbeiter bzw. dessen Vertreter zu wenden:

BETRIEBSSYSTEM

Betriebssystem, Systemorganisation	Blauensteiner	Blöser
Prozessoren, Compiler	Blauensteiner	Kleinert
JCSRTL, JCSSLIB	Blöser	Blauensteiner
CYBER-Datenverbindung	Blauensteiner	Blöser
Rechnerverbindungen (über Datenträger)	Blöser	Blauensteiner

ANWENDERSOFTWARE

Programmbibliothek, Softwareorganisation	Husinsky	Blöser
Graphische Software	Blöser	Husinsky
USER-Files (Utilities), FORDIN-Arithmetik	Husinsky	Blauensteiner
Terminalprogrammierung	Husinsky	Blauensteiner
CSMP (Digitale Simulation)	Hummer	Solar
Statistik	Solar	Husinsky
POINT-Integrationspaket	Blöser	Oberhuber (Institut für Numerische Mathematik)

HYBRIDE PROGRAMMIERUNG

Hybride Analysen	Solar	Hummer, Berger
AutoPATCH	Solar	Kleinert
HYBSYS, HYSMPK	Solar	Husinsky
Hybrid Linkage	Hummer, Berger	Solar
HYINFO, EAI 680 Fehler	Hummer, Berger	Solar
Kurs RH 9	Solar	Husinsky

BENÜTZERINFORMATION

Dokumentation, Manuals	Husinsky	Blauensteiner
INTERFACE	Husinsky	Kleinert

LITERATURDIENST

RECON	Hummer	Blöser
-------	--------	--------

BETRIEBSMITTEL

Ansuchen, Vergabe, Accounting	Husinsky	Schandl
-------------------------------	----------	---------

TECHNISCHER BETRIEB

Operating, Closed-Shop, Reservierungen, Steckbrettvergabe, Kursanmeldungen, Schaukasten	Schandl	Kondraschew
---	---------	-------------

I M A C S SYMPOSIUM

Vom 27. bis 29. September 1978 wurde an der Technischen Universität Wien unter der Patronanz von IMACS (International Association for Mathematics and Computers in Simulation) ein Symposium über "Simulation of Control Systems" abgehalten. Die wissenschaftliche Vorbereitung wurde von

a.o.Univ.Prof.Dr.I. Troch
o.Univ.Prof.Dr.A. Weinmann
Dipl.Ing.W. Kleinert

wahrgenommen. Hauptschwerpunkte dieser Veranstaltung waren Probleme der Modellbildung und Redundanz, was u.a. auch in den Themen der (eingeladenen) Hauptvorträge zum Ausdruck kam:

- R. Tomovics, Universität Belgrad:
Control of Large Systems
- J.P. Landauer, EAI Inc. West Long Branch:
Dynamic System Engineering Using Hybrid Computers
- E. Pavlik, Karlsruhe:
Interdependence of Process Model and Simulation Tool
- H. Rzehak, Hochschule der Bundeswehr München:
Redundancy in Hardware and Software of Process Computers
- D. Heppner, Rechenzentrum TU Braunschweig:
Comparative Study of Various Computer Structures

Eine sehr kurzfristige Absage des Vortragenden der Eröffnungssitzung (Prof. R. VICHNEVETSKY, Rutgers University) war die Ursache, daß an Stelle eines wissenschaftlichen Eröffnungsvortrages (Difficult Problems in the Simulation of Optimal Control Systems) eine zwanglose Vormittagsparty stattfand. Die sehr angenehme äußere Atmosphäre des Kontaktraumes der TU Wien ermöglichte nicht nur ein Kennenlernen der Tagungsteilnehmer untereinander, es wurden vielmehr bereits hier manche wissenschaftliche Diskussionen begonnen und Erfahrungen ausgetauscht.

In 44 Einzelvorträgen wurden vor allem praktische Aspekte der Modellbildung und Simulation behandelt, wobei Schwerpunkte der Anwendungen auf den Gebieten der chemischen Prozesse, Reaktortechnik, Elektrotechnik und Biologie lagen. Im einzelnen wurden folgende Vorträge gehalten:

1. THEORETICAL ASPECTS OF MODELLING AND SIMULATION

1.1. Pulse Modulated Control Systems

- R. Dillmann, P.M. Frank, K.A. Turban (Gesamthochschule Duisburg):
Mathematical Model of PFM Controllers and Its Simulation
- S.G. Tzafestas (Universität Patras):
Pulse Width and Pulse Frequency Modulated Control Systems

1.2 Design of Large Systems

- E. Toczyłowski (Politechnisches Institut Warschau):
Large Scale Steady State Process Simulation in Design of Supervisory Control
- F. Furuta, H. Kajiwara, T. Okutani, K. Tsuruoka, T. Miyasugi, T. Yoshida (Tokyo Institute of Technology):
Simulation in Computer Aided Design for Large Complex Systems
Film on the Control of a Double Inverted Pendulum

1.3. Miscellaneous Theoretical Aspects

- R. Petrovic, A. Senborn, M. Vujosevic (Institut für Automation Belgrad):
A New Algorithm for the Near Optimum Spares Allocation Subject to Multiple Constraints
- A. Kuzucu, A. Roch (École Polytechnique Fédérale de Lausanne):
Suboptimal Feedback Control of Nonlinear Systems
- E. Eitelberg (Universität Karlsruhe):
Numerical Solution of Stiff Systems with a Diagonal Splitting Method
- V.G. Beljakov (Nilschotmash Moskau):
Methods of Evaluating the Equivalent Speed of Analog Processors

2. SOFTWARE FOR SIMULATION OF CONTROL SYSTEMS

2.1 Data Transmission and Handling

- G. Dieterich (Standard Elektriz Lorenz AG, Stuttgart):
A Vector Space Approach to Synchronous Data Transmission
Part 1: Theory - Leading to a Learning Receiver
- K. Hanauer (Universität Karlsruhe):
A Vector Space Approach to Synchronous Data Transmission
Part 2: Simulation on a Hybrid Computer
- F. Bouille (Université Pierre et Marie Curie, Paris):
Contribution to the Simulation of Data Base Access Control
- G. Voehr (Technische Universität Wien):
Simulation of a System that is Described by a Statistic Nonlinear Differential Equation

2.2 Software for General Control Systems

- B. Wiegele (Technische Universität München):
Computer Assisted Learning of Methods and Problem Solving in Control Engineering
- B. Danev, A.L. Kukleva (Bulgarische Akademie der Wissenschaften):
Transformation of Control System Mathematical Models
- W. Gitt, H.J. Herrmann (Physikalisch-Technische Bundesanstalt Braunschweig):
Determination of Locus Curves for Model Parameters and Control Lay-Outs by Digital Simulation with Optimization

2.3 Software Aspects for the Simulation of Specific Control Systems

- P. Till, G. Tsamis (HTBLVA Wien 1 und ELIN Wien):
Computer Simulation of Thermal Systems by Network-Analysis Programs
- R. Mezencev, R. Lepeix (Ecole Nationale Supérieure de Mécanique Nantes):
Hybrid Simulation of a Nonlinear Hydro Pneumatic Damper for Ships

- G. Delory, H. Ghestem (Universität Lille):
Programmed Driving Sequential Machines by Whole or Fractional Control

3. MODELLING AND SIMULATION OF SPECIFIC SYSTEMS

3.1. Electrical Engineering

- C. Weisang, H. Jaschek (Universität Saarbrücken):
Multiconductor Transmission Line Macro-modelling to Simulate the Behaviour of a High Tension DC Line Unit
- H. Theuerkauf (Ludwigsburg):
Digital Simulation of Converter-Fed A-C-Drives Using Modular Models

3.2 Chemical Processes

- K.H. Fasol, M. Granau (Ruhr-Universität Bochum):
Modelling and Simulation of a Chemical-Process Plant Blast-Supply System
- A. Evers (Technische Universität Twente Enschede):
Optimal Control and Sensivity Analysis of a Chemical Process

3.3. Nuclear Engineering

- A. Hoeld, D. Beraha (GRS mbH Garching):
A Nonlinear Digital Simulator for BWR Nuclear Power Plants Within the Scope of the Application of the Optimal Digital Control Methods
- M.C.R. de Keyser, A.R. van Cauwenberghe (University of Ghent):
Simulation and Self-Tuning Control in a Nuclear Power Plant
- H. Jaschek, M. Seiermann (Universität Saarbrücken):
Modelling and Simulation of a Nuclear Power Plant with a Boiling Water Reactor

3.4 Transportation Systems

- R. Meisinger (MBB München):
Analog Simulation of Magnetically Levitated Vehicles on Flexible Guideways
- U.M. Rao, D.P. Atherton, R. Balasubramanian (University of New Brunswick, Canada):
Simulation and Estimation Studies of a Satellite Attitude Control System
- P. Kümmel (Universität Stuttgart):
Hybrid Memory for Automatic Air Traffic Control
- P. Grepper (Swiss Federal Institute of Technology Zürich):
A Simulation and Control Study of a Transport Aircraft during Helical Approach and Straight-In Landing
- J. Homola (Rudy Letov National Enterprise Prag):
Application of Simulation Methods in Aircraft Engineering

3.5 Biology

- W. Hofmann (Universität Salzburg):
Multicompartment Models for the Hybrid
Simulation of the Indoor Natural Radiation
Environment and the Resulting Dose Distri-
bution in Man
- J.A. Spriet, G.C. Vansteenkiste (University of
Ghent and Brussels):
A New Approach towards Measurement and
Identification for Control of Fermentation
Systems
- S. Lobodzinski, W. Pillmann (Technische
Universität Wien):
Simulation of the Aorta by Means of an
Adaptive Model
- J. Morawski, K. Wiklik (Institute of Sport,
Warschau):
The Applications of Analog and Hybrid
Simulation in Sports

3.6. Various Industrial Processes

- W. Drexler, W. Rademacher, L. Willner (Betriebs-
forschungsinstitut Düsseldorf):
Design of a Sub-Optimum Control System and
Hybrid Simulation of a Tandem Cold-Rolling
Mill
- D. Franke (Hochschule der Bundeswehr Hamburg):
Modelling and Simulation of High-Temperature
Creep-Stress Dynamics in the Process of Soli-
dification, with Application to Steel Casting

- L. Blach, R. Wasniowski, T. Wisniewski (Tech-
nische Universität Wrocław):
Mathematical Model and Algorithms of Optimal
Control of Coal Milling Process in the Ball
Mill

3.7 Miscellaneous Applications

- A.K. Kippo, P. Uronen (University of Oulu,
Finnland):
Simulation of a State-Controller for a
Recovery Boiler
- K. Barra, I.I. Lepsenyi (Universität Budapest):
Stochastic Simulation and Automation of a
Production Line
- Y. Ercan, Y. Goegues, Y. Paker (Universität
Ankara):
Modelling and Time Optimum Control of a Two
Capacity Thermal System with Heat Engines
and Pumps
- B. Danev, S.P. Patarinsky, G.V. Tzvetkova (Bul-
garische Akademie der Wissenschaften):
Manipulators Control Systems Design via
Successive Simulations
- G. Gunawardene, M.J. Grimble (Sheffield Poli-
technic):
Development of a Static Model for a Sendzimir
Cold Rolling Mill

Der Großteil dieser Vorträge wurde in dem bei North-Holland erschienenen
Proceedings-Band dieses Symposiums veröffentlicht, von einigen Vorträgen
wird eine erweiterte Fassung in "Mathematics and Computers in Simulation"
(Transactions of IMACS) erscheinen.

Abschließend sei darauf hingewiesen, daß es sich hier einmal mehr gezeigt
hat, wie günstig die zentrale Lage Wiens für eine derartige Veranstaltung
ist: Von den 116 Tagungsteilnehmern kamen 6 aus Übersee, 16 aus Jugoslawien,
Polen, UdSSR, CSSR und Ungarn, 12 aus Österreich, 43 aus der BRD, 18 aus
Belgien und den Niederlanden, und weitere 21 aus Frankreich, England, der
Schweiz, Schweden, Finnland, Italien, Griechenland und der Türkei.

Als für die Organisation Hauptverantwortliche möchte ich nicht versäumen,
auch an dieser Stelle allen damit befaßten Personen meinen Dank dafür aus-
zusprechen, daß diese Tagung in den Räumen der TU Wien durchgeführt werden
konnte.

I. Troch

ARBEITSWEISE UND VERWENDUNG DER REP-OP TIMER AN DER HYBRIDRECHENANLAGE

1. DER MANUELLE REP-OP TIMER DES ANALOGRECHNERS

Der Analogrechner EAI 680 ist mit einem an sich recht flexiblen "Rep-op Timer" für repetitive und iterative Rechenoperationen ausgestattet. Bei diesem "Rep-op Timer" handelt es sich im Kern um einen Dezimalzähler mit einem jeweiligen Bereich von 000 bis 999. Abb.1 zeigt das Blockschaltbild des "Rep-op Timers".

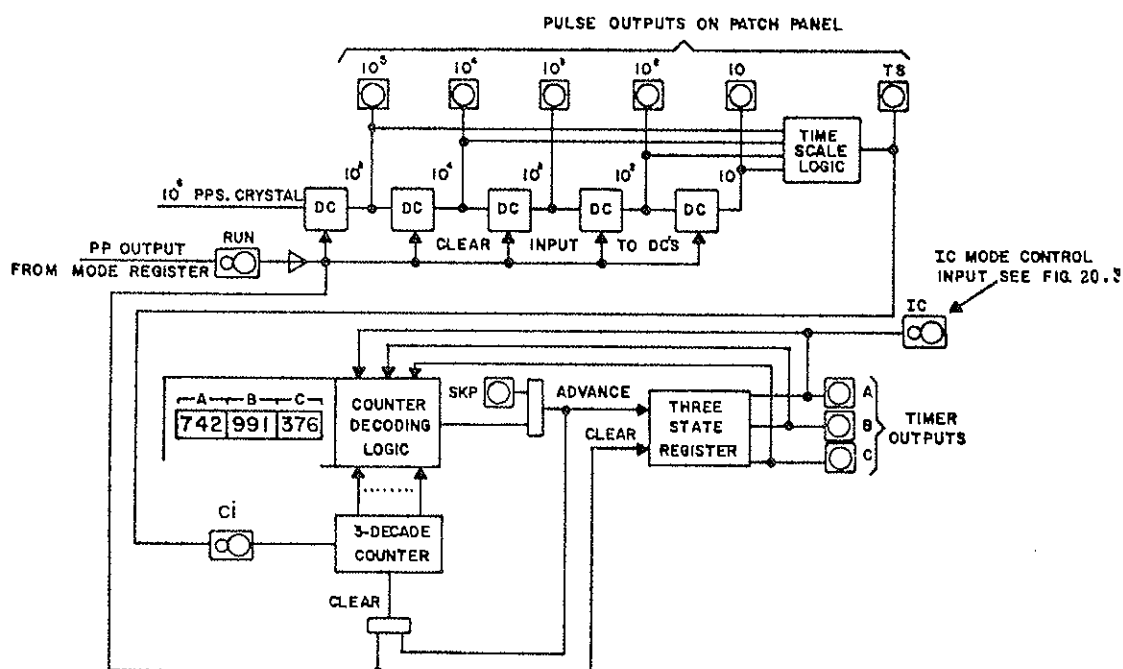


Abb.1: Blockschaltbild des EAI-680 Rep-op Timers

Der Eingang am linken oberen Rand, der mit 10^6 pps crystal" bezeichnet ist, kommt von der zentralen 1MHz-Clockfrequenz des Analogrechners. Diese wird in eine Kette von kaskadierten Dezimalzählern ("DC") geführt. Jeder dieser Dezimalzähler gibt einen Ausgangsimpuls nach jeweils 10 Eingangsimpulsen; dadurch ist die Ausgangsfrequenz jedes Dezimalzählers genau zehnmals langsamer als diejenige des vorhergeschalteten. Diese Pulsraten ("zentrale Frequenzen") haben eigene Ausgänge am Logikteil des Steckbretts und werden häufig für die Synchronisation bei Datenübertragungen vom und zum Digitalrechner benützt.

Das "TS" (Time Scale) Signal wird durch entsprechende Schaltungen in Abhängigkeit von der gewählten Zeitbasis (NS, FS, NM, FM) aus den zentralen Frequenzen abgeleitet. So erscheint z.B. in "Normal Seconds" die Frequenz 10^0 am TS-Ausgang, in "Fast Milliseconds" hingegen 10^5 .

Normalerweise ist der TS-Ausgang intern mit dem Zählereingang (CI, "Carry In") des Rep-op Timers verbunden. Dieser zählt die CI-Impulse, bis deren Anzahl mit dem Wert des entsprechenden Zählradschalters auf der 680-Konsole übereinstimmt.

Wird dieser Wert erreicht, erzeugt die Decoderlogik des Zählers ein Ausgangssignal, das den Zähler wieder auf 0 setzt und das dreistufige nachgeschaltete Zustandsregister auf den nächsten Zustand setzt: von A nach B, von B nach C, von C nach A (falls am Zählrad für C der Wert 0 eingestellt ist, springt das Register gleich von B nach A).

Der Rep-op Timer wird noch zusätzlich durch die Eingänge SKP (Skip) und RUN kontrolliert.

Der SKP Eingang wirkt gerade so, als ob der Zähler bereits den eingestellten Wert erreicht hätte: also Löschen des Zählers und Setzen des Zustandsregisters. Dies kann mitunter vorteilhaft verwendet werden, wenn das Ende eines Operate-Zyklus nicht durch eine vorgegebene Zeit, sondern eine andere Bedingung (z.B. Nulldurchgang einer analogen Variablen) bestimmt werden soll.

Der RUN-Eingang gewährleistet den oben beschriebenen Betrieb des Timers, wenn er auf logisch "1" liegt. Wenn er logisch "0" ist, löscht er alle zentralen Frequenzzähler (DC), den 3-Dezaden Zähler des Rep-op Timers und das Zustandsregister, das in diesem Fall auf den Zustand "A" zurückspringt. Wie die Abb.1 zeigt, ist der RUN-Eingang normalerweise nur dann gesetzt, wenn die PP-Drucktaste an der Console gedrückt ist. Der Run-Eingang wird häufig zur Synchronisation eines repetitiven Betriebes des Analogrechners vom Digitalteil her (z.B. durch eine Controlline) verwendet.

Für die häufigste Betriebsart des Analogrechners - den repetierenden Operate-IC-Betrieb der Integrierer zur Erzeugung eines schnell ablaufenden stehenden Bilds von Zustandsgrößen am Oszillographen - sind also keinerlei Steckverbindungen am logischen Steckbrett nötig. Es genügt, den Rechner bei Hand oder vom Digitalrechner her in den "PP-Mode" zu bringen. Die Steuerung der Integrierer erfolgt durch Standardverbindungen ("normally closed contacts").

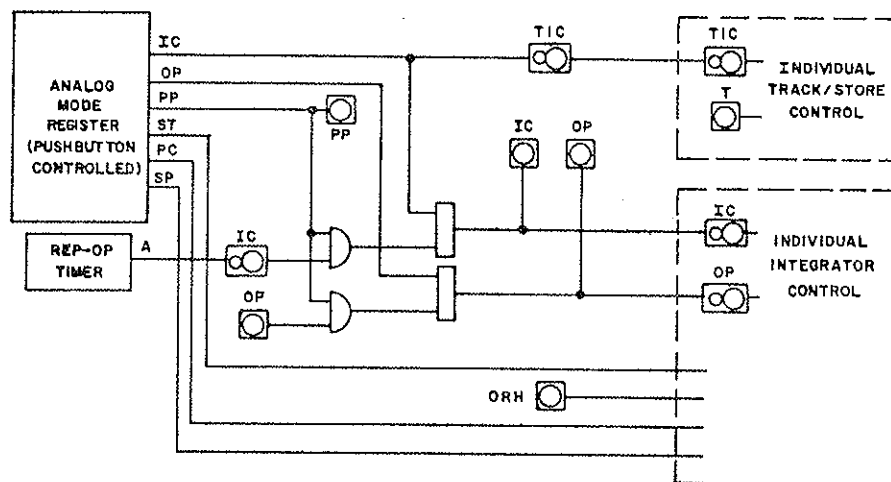


Abb.2: Mode Register und Mode Control Logik

Der Rep-op Timer Ausgang A gibt standardmäßig die IC-Zeit der Integrierer an, die Summe von B und C (=not A) die Operate-Zeit. Wie aus Abb.2 ersichtlich, ist der Steuereingang "OP" normalerweise unbesteckt und wirkt daher wie eine logische "1". Will man mit einem Hold-Intervall rechnen, so muß entweder der Rep-op Timerausgang "C" auf ORH (Override hold) oder das invertierte C-Signal auf den Eingang "DP" gelegt werden. Dabei ist zu beachten, daß in diesem Fall

die Zeitkonstanten nur NS oder FS betragen sollten, da die Integrierer unseres Analogrechners mit mechanischen Relays (Schaltzeit > 1ms) in Hold gebracht werden, deren Kontakte bei dauerndem Umschalten im zeitlichen Grenzbereich durchbrennen.

Der Rep-op Timer am EAI 680 ist also im allgemeinen für "normalen" Repetitivbetrieb beim reinen Analogrechnen recht gut geeignet. Er besitzt aber zwei, zum Teil gravierende Nachteile. Erstens ist die maximal einstellbare Rechenzeit (Operate) mit 200 Sekunden begrenzt, was in manchen Anwendungsfällen zuwenig ist und nur mit zusätzlichen Schaltungen (Zähler, Direktansteuerung des CI-Einganges) behoben werden kann. Der zweite, weit größere Nachteil besteht darin, daß der Rep-op Timer nur händisch auf bestimmte Werte gesetzt werden kann. Dieser Nachteil würde sich natürlich besonders auf den geplanten AutoPATCH-betrieb äußerst störend auswirken. Aber auch schon jetzt muß der Rep-op Timer in manchen Hybridprogrammen, so z.B. beim Arbeiten mit der interaktiven automatischen Skalierungssoftware HYBSYS, durch die Realtime Clock ersetzt werden, die dann wiederum in anderen Teilen des Programms abgeht.

2. NEUENTWICKLUNG: DIGITAL SETZBARER ABCD-TIMER

Ausgehend von diesen Problemen der Rechenpraxis ist an der Hybridrechenanlage eine neue hybride Hardware entwickelt worden: der ABCD-Timer. Es handelt sich dabei um einen voll software-unterstützten, äußerst flexiblen, digital setzbaren 12-bit Binär Zähler (0 bis 4095) mit Ein- und Ausgängen, die am Logikteil des Analogrechners abgegriffen werden können (logische Trunks 10-15). Der ABCD-Timer kann sowohl neben, oder auch als Ersatz für den normalen Rep-op Timer eingesetzt werden.

Im folgenden soll die Hardware näher beschrieben werden.

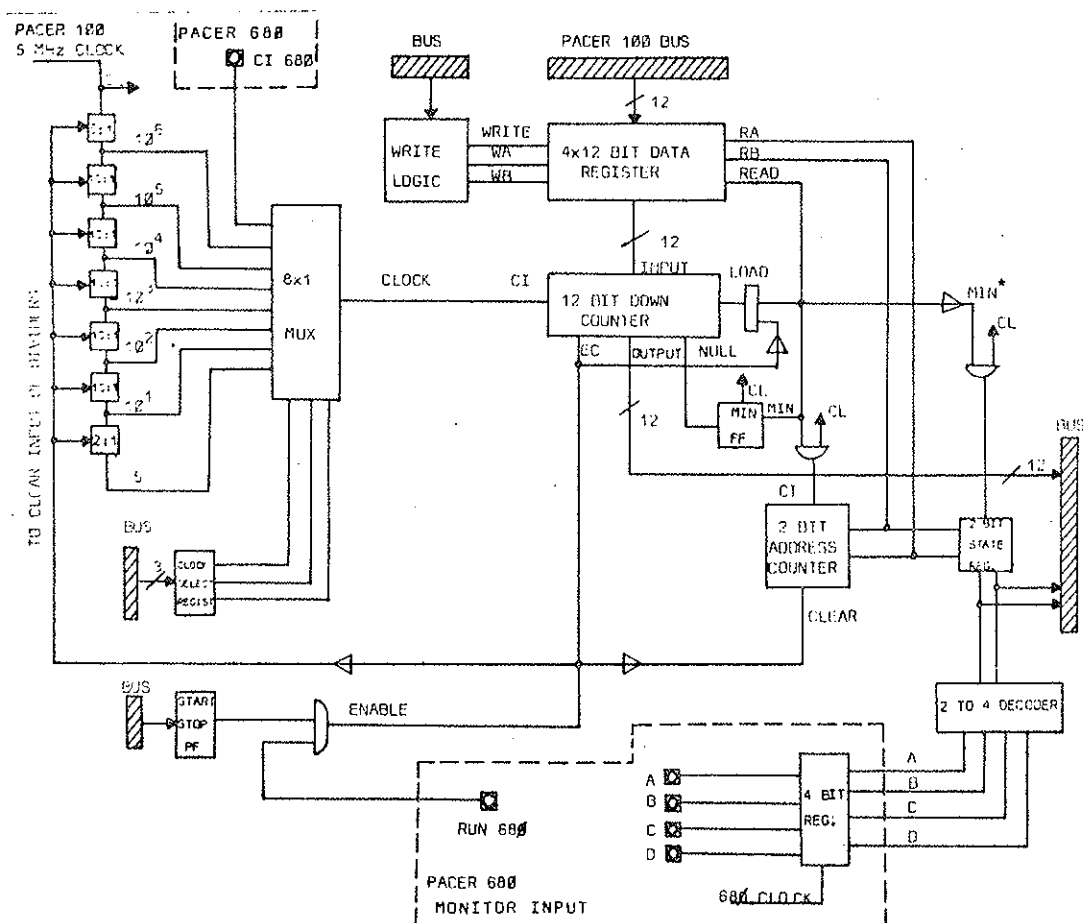


Abb. 3: Blockschaltbild des ABCD-Timers

Abb. 3 zeigt das Blockschaltbild des ABCD-Timers. Er ist in der Hauptsache auf einer Play-pack Karte aufgebaut, einem Controller im EAI Norm-Format, der die gesamte Steuerlogik für den PACER 100 Bus enthält und auf nicht bestückten IC-Sockeln Platz für den Aufbau einer Steuerlogik enthält. Die strichliert umrandeten Teile befinden sich auf einer ebenfalls neu geschaffenen Interfacekarte im Monitor Input Chassis des Analogrechners. Zwischen beiden Karten wurde ein 6 m langes Flachbandkabel verlegt. Die Datenübertragung über dieses Kabel wurde mit (in Abb. 3 nicht eingezeichneten) integrierten unsymmetrischen Leitungstreibern (SN75123) und -empfängern (SN75124) durchgeführt und die Signalleitungen ordnungsgemäß angepaßt. Diese, in IBM 360 Systemen verwendete Schaltungskombination ermöglicht Übertragungsgeschwindigkeiten auf diese Distanz von 25 Mbit/sec, was natürlich mehr als ausreichend ist. Die Ausgänge A, B, C, D sind bereits mit der internen 680-Clock synchronisiert. Das "Herz" des ABCD-Timers ist ein 12bit Binär-Zähler (3 x SN74191), der als "Down"-Counter geschaltet ist. Kontrolliert wird dieser Zähler durch den Eingang "EC" (Enable Count), der wiederum durch den Zustand des Signals ENABLE definiert ist. Dieses Signal ist nur dann logisch "1", wenn sowohl das Start/Stop FF über DF-Word am Digitalrechner als auch der Eingang RUN 68D am Logikteil des Analogrechners auf "1" gesetzt sind. Letzteres ist auch bei Nichtbestecken der entsprechenden Trunk-Buchse der Fall. Ist ENABLE logisch "0", dann stoppt der Zähler und wird mit dem ersten Wert (RA = 0, RB = 0) des 4-stufigen 12bit Registers geladen.

Das mit IC's des Typs SN74LS670 aufgebaute Register erlaubt gleichzeitiges Lesen und Schreiben. (Das Schreiben erfolgt durch die WRITE Logik und benötigt eine Sequenz DO, DF.) Die Adressen RA und RB werden dekodiert, und am Analogrechner ist das Signal A gesetzt. Wird nun ENABLE = 1, dann beginnt der Zähler zu arbeiten. Er zählt mit einer durch CLOCK bestimmten Frequenz. Diese kommt aus einem 8 x 1 Multiplexer (SN74151), der je nach dem Zustand des mit DF-Word setzbaren Clock Select Registers eine der sieben Frequenzen, die durch Zähler von der zentralen PACER 100 Clock abgeleitet werden, oder eine beliebige Frequenz, die vom Analogrechner kommt, (CI 680) anwählt.

Erreicht der 12bit Zähler den Wert Null, so wird der 2bit Adresszähler um 1 erhöht und der nächste Wert aus dem 4 x 12 bit Datenregister geladen. Ist dieser Wert gleich Null, so wiederholt sich der Vorgang beim nächsten 5 MHz Clockimpuls. Das 2bit Zustandsregister, das den Decoder steuert, wird allerdings nur dann getriggert, wenn der neugeladene Zählerwert ungleich Null ist (MIN* = 1). Ist zum Beispiel der Timer auf die Werte 10, 0, 10, 0 geladen, dann ist der Ausgang "A" am Analogrechner 10 Zeiteinheiten lang "1". Er bleibt es auch noch für weitere 200 ns. Dann wird für 10 Zeiteinheiten der Ausgang "C" aktiviert, B und D bleiben die ganze Zeit Null. Hat das Adressregister den Wert 3 erreicht, beginnt es wieder bei Null, der Timer arbeitet also zyklisch. Zu jedem Zeitpunkt ist es möglich, den laufenden Wert des Zustandsregisters und des Zählerstandes vom Digitalrechner auszulesen. Das erlaubt die Erzeugung von programmgesteuerten Impulsfolgen variabler Länge aus den Ausgängen A bis D, wobei das Datenregister immer wieder während des Zählens mit neuen Werten geladen wird.

Es soll noch darauf hingewiesen werden, daß wegen der Verwendung von unterschiedlichen Logikfamilien (die Interfacekarten verwenden TTL bzw. Low-Power Schottky, der Analogrechner RTL-Schaltkreise) die Ausgänge des ABCD-Timers nicht mit mehr als je 3 Eingängen belastet werden sollten. Gegebenenfalls müssen sie mit einem Ungatter zwischengepuffert werden.

Geplant ist noch die folgende Erweiterung der Hardware: Nach Setzen eines bestimmten Bits im DF-Word wird die Zählerfrequenz noch einmal mit einem 8-stelligen Binärzähler, der separat mit einem beliebigen Wert geladen werden kann, geteilt. Die Zählerfrequenz erscheint in diesem Fall auch auf der Buchse "D" am Logikteil des Steckbretts. Diese Erweiterung erlaubt es, ohne weiteren Aufwand, auch bei allgemeinen Zeittransformationen der analogen Variablen eine feste Anzahl von Datenpunkten zur Diskretisierung analoger Variablen (= Darstellung hybrider Variablen) unabhängig von der gewählten Zeitkonstanten zu verwenden.

SOFTWAREUNTERSTÜTZUNG DES ABCD-TIMERS

Beim digital setzbaren ABCD-Timer können also maximal vier verschiedene Werte gesetzt werden (z.B. 5, 10, 15, 0) und eine bestimmte Frequenz (z.B. 10^3) gewählt werden. Darauf werden die Ausgänge A, B, C, D repetierend am Analogrechner jeweils auf "1" gesetzt, also in unserem Beispiel A 5 msec, B 10 msec, C 15 msec lang, der Ausgang D wird nie gesetzt.

Wie bereits oben erwähnt stehen dem Benutzer des ABCD-Timers eine Reihe von Unterprogrammen aus dem erweiterten Hybrid-Linkage Package zur Verfügung (LOAD,D9 HYBRID). Die Subroutine TIMER mit dem Aufruf

```
CALL TIMER (IFRQ, IA, IB, IC, ID)
```

setzt das Start Flip Flop zurück, wählt eine der 8 Frequenzen und lädt die Datenregister auf die Werte IA bis ID. Mit

```
CALL TSTART
```

wird der Timer gestartet, der ab dann unabhängig vom Digitalprogramm repetierend läuft. Mit

```
CALL TSTOP
```

kann der Timer in den Zustand "A" versetzt und gestoppt werden. Die Subroutine TSTATE mit dem Aufruf

```
CALL TSTATE (INR, IVAL)
```

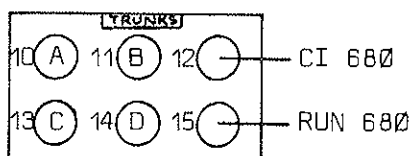
liefert die Information, welcher Timer-Zustand (INR = 1,...,4) gerade auf welchen Wert (IVAL = 0,...,4095) abgearbeitet ist. Mit

```
CALL TLOAD (INR, IVAL)
```

wird das Datenregister des Timers INR auf den Wert IVAL gesetzt, der Ablauf des Timers wird dabei aber nicht verändert. Schließlich wird mit

```
CALL TIMING (N, IFIELD, IFRQ)
```

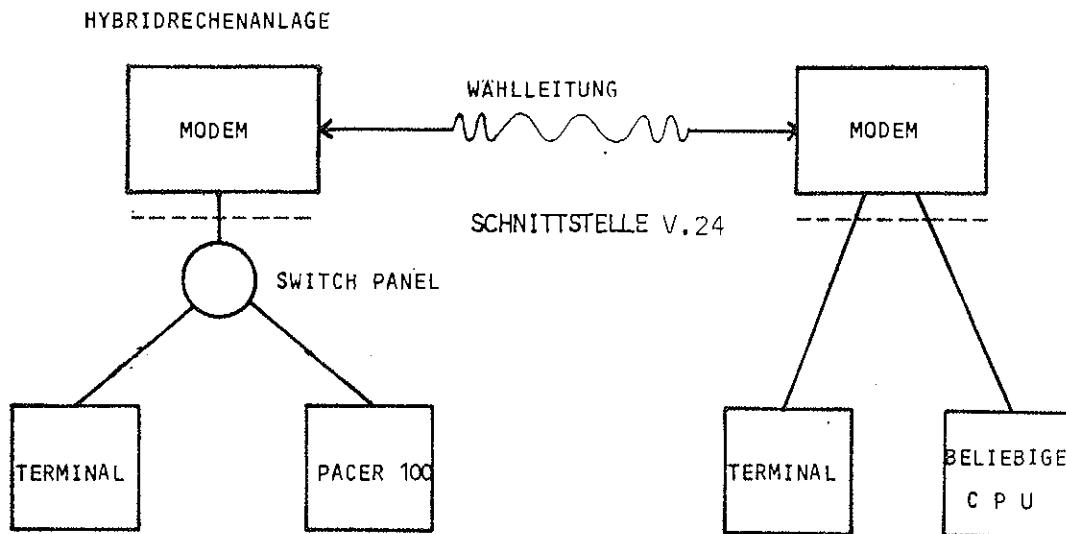
eine Impulsfolge durch ein (2-dimensionales) Feld IFIELD (4,N) von je N Werten auf den Ausgängen A bis D mit bestimmter Frequenz erzeugt. Die entsprechenden Buchsen befinden sich am Logikteil des Steckbretts unmittelbar neben den A/D SAMPLE CONTRDL Buchsen.



Genauere Angaben über die Softwareunterstützung des ABCD-Timers können den Kurzbeschreibungen entnommen werden.

W. Kleinert

MODEMVERBINDUNGEN AN DER HYBRIDRECHENANLAGE



Ein Modem (Modulator-Demodulator), wie er jetzt auch an der Hybridrechenanlage zur Verfügung steht, ermöglicht die Datenübertragung auf Fernsprechleitungen, bei unserem Gerät mit der Geschwindigkeit von 300 Baud. Dadurch werden keine besonderen Anforderungen an die Güte der Übertragungsleitung gestellt und daher können das interne Telephonnetz der Technischen Universität sowie, unter Berücksichtigung der postalischen Bestimmungen, auch das öffentliche Wählnetz verwendet werden.

Die Übertragung erfolgt im binären Frequenzmodulationsverfahren und asynchron, wobei das Frequenzband der Fernsprechleitung in zwei voneinander unabhängige Kanäle A und B aufgeteilt zu denken ist (Duplexbetrieb). Wir arbeiten halb-duplex, d.h. es kann immer nur eine Station senden oder empfangen. Die rufende Station sendet auf Kanal A(1080Hz), die gerufene auf Kanal B(1750Hz).

Die Schnittstelle zwischen Modem und Dateneneinrichtung (Terminal, Controller) entspricht der CCITT Norm V.24 (Spannungsinterface).

TECHNISCHE EINRICHTUNGEN AN DER HYBRIDRECHENANLAGE

Das im letzten INTERFACE beschriebene Switch Panel ermöglicht die Anschaltung des Modems an die V.24 Schnittstellen verschiedener Geräte.

Es sind zwei Schaltwege möglich:

- Terminal: zur Zeit ist ein EAI Video C101 angeschlossen
- PACER 100 CPU: zur Benützung des Rechners durch externe Datenendstellen

TECHNISCHE EINRICHTUNGEN DER GEGENSEITE

Die Gegenstation muß über einen kompatiblen Modem verfügen und kann daran ein

- Terminal oder einen
- Rechner

mit entsprechender Schnittstelle betreiben.

EINSATZMÖGLICHKEITEN

Wir haben derzeit die Möglichkeit, als Terminal Verbindungen zu folgenden Rechnern herzustellen:

CYBER 74 TU Wien (NOS/BE)

CYBER 73 Uni Wien (NOS/BE)

ITEL AS/5 Frascati /Italien (RECON-Informationssystem, siehe auch INTERFACE Nr.10)

Diese Verbindungen sind nur für die rechenzentrumsinterne Benützung gedacht.

Umgekehrt besteht die Möglichkeit für alle Terminalstationen mit entsprechender Schnittstelle (die bisher z.B. nur mit CYBER 73/74 gearbeitet haben), sich auch an die Hybridrechenanlage anzuschalten. Dieser Anschluß erfolgt allerdings nicht automatisch durch Anwahl der Telefonnummer eines Multiplexer-Boards, sondern wird nach vorheriger Vereinbarung handvermittelt. Dieser

Einsatz ist für Hybridbenutzer interessant, die eine bereits vollständig vorbereitete Simulation von einem entfernt aufgestellten Terminal (z.B. mit Hilfe der digitalen Unterstützungssoftware HYBSYS) steuern wollen. Dazu eignet sich natürlich am besten ein Terminal mit graphischer Ausgabemöglichkeit, auf dem man die Ergebnisse on-line zeichnen kann. Bei Verwendung entsprechender Software können diese Ergebnisse auch durch die COPY-Option wahlweise auf den Data Plotter kopiert werden (siehe INTERFACE Nr. 10).

Ebenso ist die Benutzung digitaler interaktiver Programme (z.B. Source Processor) denkbar. Nähere Auskünfte erteilt Herr Hummer.

H. Hummer

DATENVERBINDUNG IN DIE WIENER STADTHALLE

In der oben geschilderten Weise konnte Mitte Oktober beim Kongress WID '78 der Österreichischen Ärztekammer und der Österreichischen Dentistenkammer eine Datenfernverbindung zwischen der Wiener Stadthalle und der Hybridrechenanlage hergestellt werden. Dies ermöglichte das Arbeiten mit Programmen, Daten und graphischer Ausgabe bei einer Vorführung der rechnergesteuerten Unterstützung bei kieferorthopädischen Problemen.

RECHNEN MIT INTERRUPTUNTERPROGRAMMEN

Da im Betriebssystem JCS/TS 7 einige Änderungen bezüglich des Rechnens mit Interruptunterprogrammen durchgeführt wurden, sei hier eine neue kurze Anleitung gegeben.

Interruptunterprogramme unterscheiden sich syntaktisch nicht von anderen Unterprogrammen, sie werden jedoch nicht vom Programm (softwaremäßig), sondern nach einer Aktion der Hardware (General Purpose Interrupt Eingänge (GPI) am Analogrechner) exekutiert. Sie können also den Programmablauf am Digitalrechner zwischen beliebigen Operationen zu nicht vorher definierten Zeitpunkten unterbrechen.

Die Interruptunterprogramme müssen im System JCS/TS 7 statt des normalen FORTRAN-Rücksprungs RETURN einen Aufruf der Systemroutine RETURN (d.h. CALL RETURN) enthalten. Soll ein Interruptunterprogramm nicht nur durch Interrupts, sondern auch über CALL-Statements aufgerufen werden können, so muß zusätzlich ein Rücksprung RETURN programmiert werden. In diesem Fall ist die Steuerung der Rücksprünge, eventuell über COMMON-Variable, genau zu überlegen. Verwendet ein Interruptunterprogramm weitere Unterprogramme, die auch das Hauptprogramm verwendet, kann es durch den normalen Bindevorgang des Core Image Generators (CIG) zu Exekutionsfehlern kommen.

Beispiel:

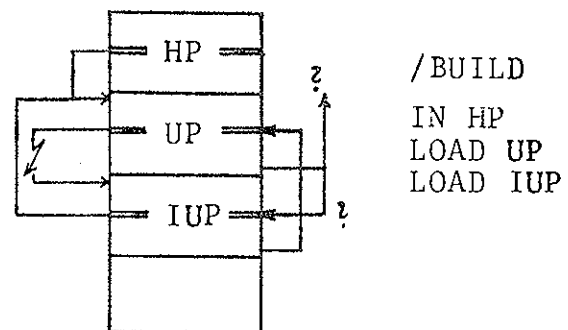
Steuerkarten

```
/FOR HP  
EXTERNAL IUP  
CALL HYBRID  
CALL INTER (3, IUP)  
CALL SI  
1 CALL UP  
GOTO 1  
END
```

```
/FOR UP  
SUBROUTINE UP  
RETURN  
END
```

```
/FOR IUP  
SUBROUTINE IUP  
CALL UP  
CALL RETURN  
END
```

Aktionen des Programms



↓ normaler Einsprung

⚡ Einsprung durch Interrupt

Da UP nur einmal zum Hauptprogramm geladen wird, könnte im Falle, daß IUP die Exekution von UP unterbricht und z.B. selbst UP aufruft, die erste Rücksprungadresse von UP zum Hauptprogramm überschrieben werden. Dies würde zu einer Endlosschleife von IUP-Aufrufen führen.

Durch Verwendung der Steuerkarte

SPECIAL,D. NAMEXX

beim Bindevorgang des Programms durch den CIG werden alle vom Programm NAMEXX benötigten Unterprogramme auf den durch nachfolgende LOAD-Karten spezifizierten Files noch einmal geladen, unabhängig davon, ob sie bereits für das Hauptprogramm geladen wurden.

Werden in einem Hauptprogramm mehrere Interruptunterprogramme verwendet, die SPECIAL gebunden werden müssen, können die Objektprogramme unter einem Sammelnamen übersetzt werden. Dieser Name ist dann auf der SPECIAL-Steuerkarte anzuführen.

Daß mehrere Interruptunterprogramme denselben Satz an Unterprogrammen verwenden dürfen, ist deshalb möglich, weil Interruptunterprogramme einander definitionsgemäß nicht unterbrechen können.

Als Beispiel sei die Steuerkartenreihenfolge für das SPECIAL-Binden von INTERS angegeben. MAINCI ist das neu zu erstellende exekutable Programm, PRO das vorher abgespeicherte Objektprogramm und FILE1...FILEX gespeicherte, von PRO und INTERS benötigte Unterprogramme. Die Skizze daneben zeigt den korrekten Exekutionsablauf des vorigen Beispiels.

Steuerkarten

```

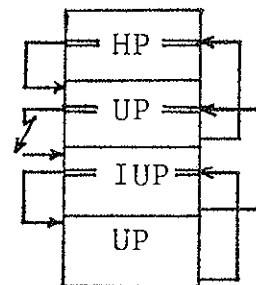
/BUILD MAINCI
IN PRO
LOAD,D. FILE1
:
LOAD,D. FILEX
SPECIAL,D. INTERS
LOAD,D. FILE1
:
LOAD,D. FILEX
  
```

Aktionen des Programms

Hauptprogramm

benötigte
Subroutinen
Interruptroutine

im IUP benötigte
Subroutinen



```

/BUILD
IN HP
LOAD UP
SPECIAL IUP
LOAD UP
  
```


Interruptunterprogramme müssen immer SPECIAL gebunden werden, denn es gibt zumindest Systemunterprogramme, die von HP und IUP gemeinsam aufgerufen werden.

Werden im Hauptprogramm und Interruptunterprogramm Programme des Hybrid Linkage Programmpaketes aufgerufen, müssen die entsprechenden LOAD-Karten sowohl beim Hauptprogramm als auch beim Interruptunterprogramm eingefügt werden.

Die Programme der Run-Time-Library (JCSRTL) und der System Library (JCSLIB) werden automatisch sowohl zum Hauptprogramm als auch zum SPECIAL-gebundenen Unterprogramm geladen.

Nur in Sonderfällen kann man bei der Benützung von Interruptunterprogrammen die SPECIAL-Steuerkarte sparen. Dies wäre z.B. möglich, wenn sichergestellt ist, daß Unterbrechungen des Hauptprogramms durch Interruptunterprogramme nur in Programmteilen wie

```
1  CONTINUE      oder      STOP      oder      CALL WAITSL(I)
      GOTO 1      (Sonderbe-
                  deutung im
                  hybriden
                  FORTRAN)
```

möglich sind. Im letzten Fall dürfte natürlich keines der Interruptunterprogramme das Unterprogramm WAITSL verwenden.

Bei Beachtung der gegebenen Hinweise können in Interruptunterprogrammen hybride Unterstützungsprogramme, I/O sowie Massenspeicher- und Datentransfers unbeschränkt durchgeführt werden. Generell ist folgendes für das Binden von Programmen bezüglich der Rechenarithmetik zu beachten: es dürfen exklusive immer nur das Hauptprogramm oder die Interruptunterprogramme

- FPP-Real rechnen (Normalfall) bzw. Complex oder mit doppelter Real-Genauigkeit (also mit LOAD, D9 DC/FPP),
- NONFPP-Real rechnen (also mit LOAD, D9 NONFPP) bzw. Complex oder mit doppelter Genauigkeit (sowohl Real als auch Integer, also mit LOAD, D9 DC/NON).

H. Hummer

GRAPHISCHE DATENVERARBEITUNG

EINSATZ AN DER HYBRIDRECHENANLAGE

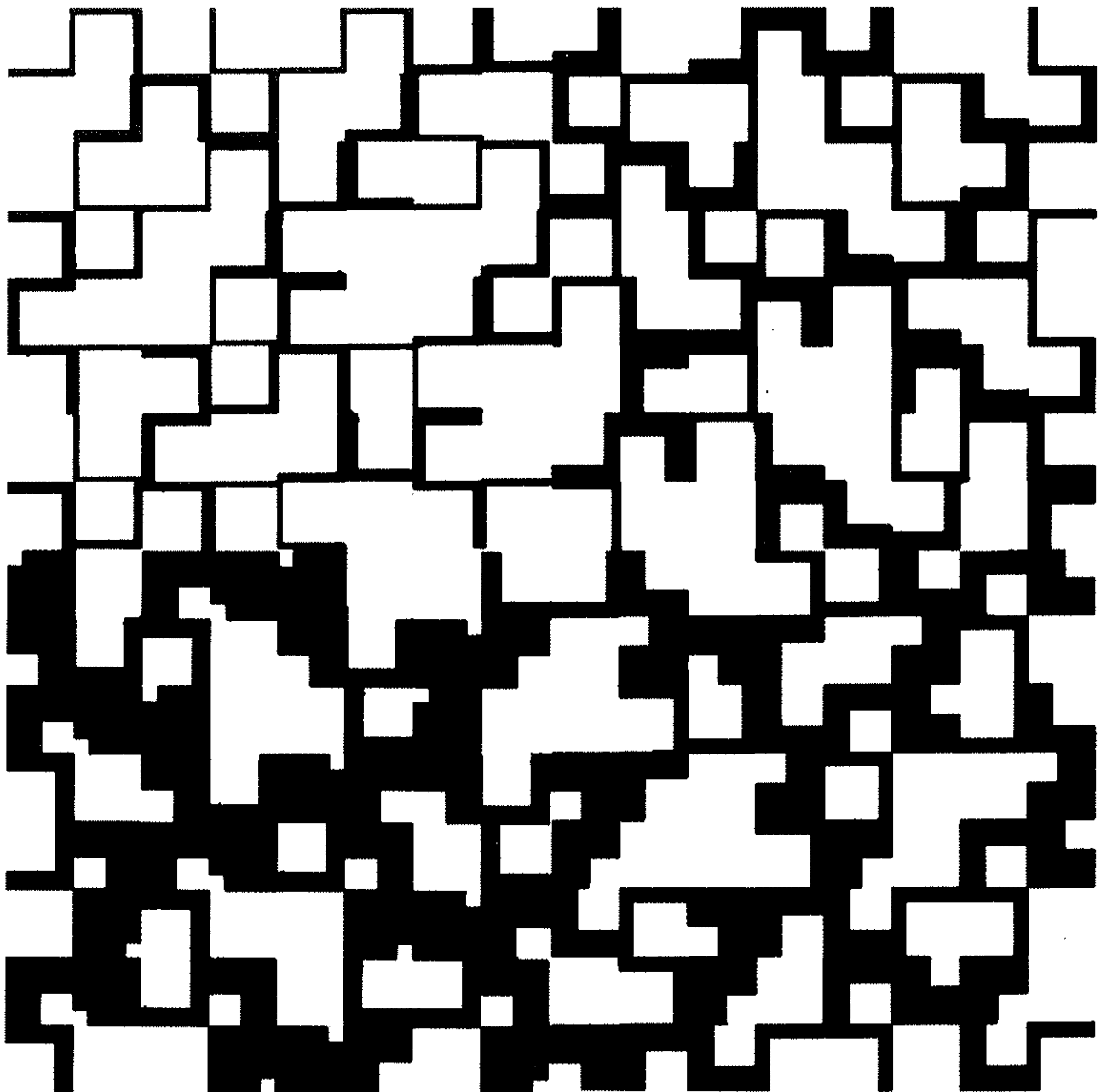
Die Tatsache, daß an der Hybridrechenanlage der TU Wien schon seit langem ein Data Plotter zur Verfügung steht, hat immer wieder dazu geführt, daß Benutzer die Ergebnisse von Berechnungen nicht, wie sonst leider sehr oft üblich, in langen, unübersichtlichen Tabellen, sondern in Form von übersichtlichen Kurven, Histogrammen, etc. mithilfe dieses Plotters dargestellt haben. Aus solchen Zeichnungen lassen sich quantitative und qualitative Informationen gewinnen, die den Ansprüchen der Computer-Benutzer oft viel mehr entgegenkommen als seitenlange Tabellen. Seit längerer Zeit ist es außerdem möglich, an der Hybridrechenanlage ein Graphic Display Terminal zu benutzen, mit dessen Hilfe eine gewisse Interaktivität zwischen Benutzer und Computerprogramm erreicht werden kann. Besonders geeignet ist dieses Graphic Display auch für die Programmentwicklung, da die graphische Ausgabe auf dem Bildschirm nicht nur schneller ist als auf dem Plotter, sondern auch eine Papierersparnis mit sich bringt.

Der graphischen Datenverarbeitung ist also an der Hybridrechenanlage immer schon große Aufmerksamkeit geschenkt worden. Im Zuge der laufenden Verbesserung der Grundsoftware für den Plotter und das Graphic Display wurde eine software-mäßige Hard Copy-Möglichkeit für Bildschirmzeichnungen geschaffen, bei der beliebige Zeichnungen auf dem Display auf Wunsch des Benutzers durch Tastendruck auch auf dem Plotter ausgegeben werden. Mit der Entwicklung und Implementierung des IPP1 (Interpolations- und Plotterpaket für eindimensionale Probleme) wurde den Benutzern ein umfangreiches Programmpaket für die graphische Darstellung von Funktionen und ebenen Kurven zur Verfügung gestellt. Im Frühjahr dieses Jahres wurde schließlich eine vorläufige Version des IPP2 (Interpolations- und Plotterpaket für Flächen) implementiert. Für die nächste Zeit ist eine schnelle graphische Ausgabe über einen Fernseh Bildschirm geplant, wobei es möglich sein wird, einzelne Punkte und Linien wieder zu löschen, ohne das restliche Bild zu zerstören. Beim Graphic Display Terminal mit Speicherröhre kann ja nur der ganze Bildschirm gelöscht werden.

MÖGLICHKEITEN UND ANWENDUNGEN

Einen guten Überblick über Möglichkeiten und Anwendungen der graphischen Datenverarbeitung gab der Kurs "Computer Graphics-State of the Art Techniques and Applications", der von Dr. Stephen R. Levine von den Lawrence Livermore Laboratories (Kalifornien) im Juni in München gehalten wurde. Im folgenden soll ein kurzer Bericht über diesen Kurs gegeben werden.

Bei der graphischen Datenverarbeitung werden u.a. computergesteuerte Zeichengeräte (Plotter und große Zeichentische) für die Ausgabe, Digitalisierte für die Datenerfassung und graphische Bildschirmgeräte mit Lichtstift, positionierbarem Fadenkreuz u.ä. für die interaktive Verarbeitung eingesetzt. Die Anwendungen umfassen Management Informationssysteme, computerunterstützte Entwurfstechnik (u.a. für Autos, Schiffe und Flugzeuge), Gebäude- und Städteplanung, Simulationen für Trainingszwecke, automatische Prozeß- und Fertigungssteuerung, Verarbeitung von kartographischen Daten, Entwurf integrierter Schaltungen, Flugüberwachungssysteme und Computerkunst, um nur einige zu nennen. (Gerade auf dem letztgenannten Gebiet wurden von Herrn O.Putz vom Institut für zeichnerische und malerische



O. Putz: Strukturwandlung

Darstellung der TU Wien, Vorstand Prof. R. Göschl, schon Arbeiten an der Hybridrechenanlage durchgeführt.) Die Einsatzgebiete der graphischen Datenverarbeitung werden immer mehr, wobei besonders der Möglichkeit der interaktiven Verarbeitung eine große Bedeutung zukommt. Neue Anwendungsmöglichkeiten bieten sich dort, wo gewisse Kontroll- und Bedienungsfunktionen durch einen Bearbeiter erforderlich sind und wo vollautomatische Verarbeitungskonzepte versagen.

HARDWARE-MÖGLICHKEITEN

Während sich die Anwendung der graphischen Datenverarbeitung in der Vergangenheit häufig auf die Benutzung eines Plotters beschränkte, da interaktive graphische Bildschirmgeräte zu teuer, die Bildgenerierung zu langsam und die Rechner zu wenig leistungsfähig waren, führt die Verbilligung

der Hardware sowie die Vergrößerung des Angebots an Geräten und speziellen Softwarepaketen zu einer immer breiteren und stärkeren Nutzung der graphischen Datenverarbeitung. Für beinahe jeden Zweck steht heute ein geeignetes Gerät zur Verfügung. (Das bedingt aber umgekehrt auch, daß man sich vor der Anschaffung eines Geräts oder eines kompletten graphischen Datenverarbeitungssystems genau über die geplante Verwendung im klaren sein muß.) Die Palette der angebotenen Ausgabegeräte reicht von Plottern (in den verschiedensten Größen), elektrostatischen Druckern über Kathodenstrahlröhren, Speicherröhren und Plasma Panel Displays bis zu speziellen Geräten für die Ausgabe auf Dias, Film oder Mikrofilm. Selbstverständlich gibt es Plotter, bei denen man software-mäßig unter bestimmten Schreibstärken bzw. Farben auswählen kann, ebenso können Darstellungen auf Bildschirmgeräten auch in Farbe und in einer Vielzahl von unterschiedlichen Intensitätsstufen erfolgen.

UNTERSTÜTZUNG VERSCHIEDENARTIGER HARDWARE

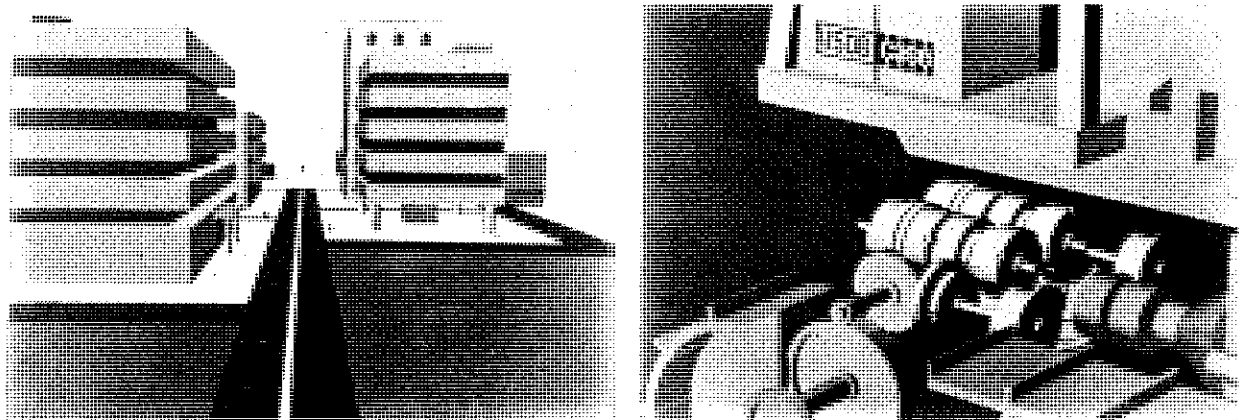
Probleme können sich bei der Koppelung verschiedenartiger Geräte ergeben, und zwar nicht so sehr bei der Hardware wie bei der Software. Bei einem genau spezifizierten Anwendungsbereich kann unter Umständen auf ein fertiges Programmsystem zurückgegriffen werden. Im Normalfall hat man ein anwendungsunabhängiges graphisches Grundsoftwarepaket zur Verfügung, auf dem dann eine höhere Software aufbaut. (Den Programmpaketen IPP1 und IPP2 wurde die Grundsoftware der Firma CalComp zugrunde gelegt, die sehr weit verbreitet ist.) Man ist übrigens gerade bemüht, diese Grundsoftware international zu normen.

Stehen verschiedenartige Ausgabegeräte in einem Rechenzentrum zur Verfügung, so ergibt sich das Problem, wie die Grundsoftware diese Geräte unterstützt. Hier kann man z.B. nach dem Prinzip des kleinsten gemeinsamen Nenners vorgehen und jede Ausgabe auf das Ziehen einer Linie zurückführen. In diesem Fall braucht nur die elementarste Routine zum Verbinden zweier Punkte auf das betreffende Gerät zugeschnitten sein, da alle anderen Routinen (zum Zeichnen von Achsen, Texten, etc.) auf diese Routine zurückgreifen. Der Nachteil besteht darin, daß man nicht von speziellen Eigenschaften des Geräts Gebrauch machen kann. Eine andere Möglichkeit besteht darin, einen Zwischenfile (Display File) anzulegen, der von den verschiedenen Interpretern für die verschiedenen Ausgabegeräte verarbeitet werden kann. Hier können spezielle Geräteeigenschaften ausgenutzt werden, doch ist diese Vorgangsweise im allgemeinen langsamer.

SOFTWARE TECHNIKEN

Zahlreich wie die Geräte für die graphische Datenverarbeitung sind auch die angewandten Techniken. Sie reichen von einfachen Verfahren für die näherungsweise Darstellung einer Linie durch Punkte in einem Punktraster über Verfahren zur Zeichengenerierung bis zu verschiedenen Transformationen der darzustellenden Objekte. Im Bereich der graphischen Darstellung dreidimensionaler Objekte kommt dem Problem der verdeckten Linien und der verdeckten Flächen sowie der Perspektive eine große Bedeutung zu. Spezielle Verfahren existieren z.B. auch zur Vermeidung von gezackten Linien bei Raster Displays (Anti-Aliasing), zur Erzeugung "falscher" Intensitätsstufen auf Geräten mit nur zwei Intensitätsstufen (Dithering), zur Darstellung von Lichtreflexionen auf einer Oberfläche sowie zur Erzeugung von Schatteneffekten. Allerdings sind eine Vielzahl der Verfahren nur bei einer bestimmten Gruppe von Ausgabegeräten anwendbar.

Äußerst wertvoll war die Veranschaulichung der diversen Verfahren anhand von Dia- und Filmvorführungen. Sie sollten die verschiedenen Anwendungsbereiche der graphischen Datenverarbeitung aufzeigen und die Verwendung der beschriebenen Verfahren demonstrieren. Besonders interessant waren farbige Echtzeitsimulationen für die Pilotenausbildung, wie sie tatsächlich von einigen Luftfahrtgesellschaften verwendet werden. Start-, Flug- und Landemanöver können so beliebig oft trainiert werden, wobei etwa auch die Sichtverhältnisse für den Piloten variiert werden können (z.B. Nebel, Nachtflug, etc.).



Beispiele für den Einsatz der graphischen Datenverarbeitung in Städteplanung und Maschinenbau. (Die Bilder sind Reproduktionen von einem Mikrofiche aus den Kursunterlagen)

ZUKÜNFTIGE ENTWICKLUNG

Was die Zukunft der graphischen Datenverarbeitung betrifft, so wird die Entwicklung schnellerer und billigerer Hardware sowie besserer und komfortablerer Software dazu führen, daß die graphische Datenverarbeitung in noch weiterem Umfang als bisher in Industrie, Wissenschaft, Technik und Verwaltung Anwendung findet. Eine besondere Bedeutung könnte der graphischen Datenverarbeitung in ferner Zukunft zukommen, wenn die Haushalte an Kabelnetze angeschlossen sind. Wenn man, wie heute schon verschiedentlich prophezeit wird, das TV-Gerät in Verbindung mit einer Tastatur als Bildschirmterminal eines Großrechners für diverse Dienste verwenden kann, so ist es auch denkbar, Graphiken beliebiger Art abzurufen und auf dem Fernsehschirm darzustellen.

FORSCHUNGSPROJEKT AN DER TU WIEN

Seit 1977 läuft am Institut für Datenverarbeitung, Vorstand Prof. R. Eier, das vom Fonds zur Förderung der wissenschaftlichen Forschung und vom Gewerbefonds unterstützte Projekt "Graphische Datenverarbeitung" (Kontaktperson: J. Weiß, Klappe 787). Die dafür zur Verfügung stehende Anlage besteht aus einem vollausgebauten SIEMENS 330 Zentralrechner mit Standardperipherie und einer großen Palette von graphischen Peripheriegeräten. Für die graphische Eingabe gibt es einen Digitizer, Joy Stick und eine Rollkugel, für die Ausgabe einen vierfärbigen Bildschirm (mit Lightpen und Tastatur), ein Graphic Display und einen Vierfarbenplotter. Unterstützt werden diese Geräte mit dem interaktiven Softwaresystem GMB300 Version 3. Die Forschungsschwerpunkte umfassen unter anderem geräteunabhängige graphische Systeme, Dialogunterstützung, intelligente graphische Terminals, Umschlüsselung von Vektor- in Rasterdarstellungen, dreidimensionale Dateneingabe und -manipulation. Außerdem besteht eine Mitarbeit bei den Softwarenormierungsbestrebungen von SIGGRAPH (Special Interest Group for Computer Graphics der ACM).

F. Blöser

ANALOGUE SIMULATION DER MECHANISCHEN VORGÄNGE

BEIM LÄUTEN EINER GLOCKE

Fortgeschrittenenpraktikum für hybride Analogrechner
Gruppe Mak, Ringhofer

Ähnlich dem im letzten INTERFACE vorgestellten Praktikumsbeispiel der rein analogen Simulation des Schwingungsverhaltens eines Autos bei der Fahrt über eine Teststrecke soll auch diese Simulation in erster Linie als Übungsaufgabe zur Anwendung der Komponenten des Analogrechners gesehen werden. Bei diesem Beispiel wurde eine vom Analogrechner gesteuerte figürliche Darstellung der bewegten Glocke am Oszillographen gefordert, die neben dem Erlernen gewisser programmieretechnischer Besonderheiten den Vorteil eines einfachen und schnellen Erfassens der in Echtzeit programmierten Bewegung bietet.

Simuliert soll das Anläuten einer Glocke werden, d.h. daß auf die sich zunächst in Ruhe befindliche Glocke zeitweise eine über den Glockenstrang wirkende Kraft ausgeübt wird, die dann die Bewegungen von Glocke und Klöppel hervorruft. Das Zweimassensystem kann als Doppelpendel aufgefaßt werden und genügt folgenden Gleichungen:

$$\begin{aligned} (m_1 i_1^2 + m_2 a^2) \ddot{\phi}_1 + m_2 s_2 \ddot{\phi}_2 \cos(\phi_1 - \phi_2) + m_2 s_2 \dot{\phi}_2^2 \sin(\phi_1 - \phi_2) + \\ (m_1 s_1 + m_2 a) g \sin \phi_1 = f(t) \end{aligned} \quad (1)$$

$$i_2 \ddot{\phi}_2 + a s_2 \ddot{\phi}_1 \cos(\phi_1 - \phi_2) - a s_2 \dot{\phi}_1^2 \sin(\phi_1 - \phi_2) + s_2 g \sin \phi_2 = 0 \quad (2)$$

Dabei bedeuten

m_1, m_2	Masse von Glocke, Klöppel
i_1, i_2	Trägheitsradien bezüglich der Aufhängepunkte A_1, A_2
$f(t)$	Zugkraft am Strang ($f(t) = 0$ oder $f(t) = c$)
g	Erdbeschleunigung

Die anderen Parameter von (1), (2) sind aus Abb.1 ersichtliche Abmessungen

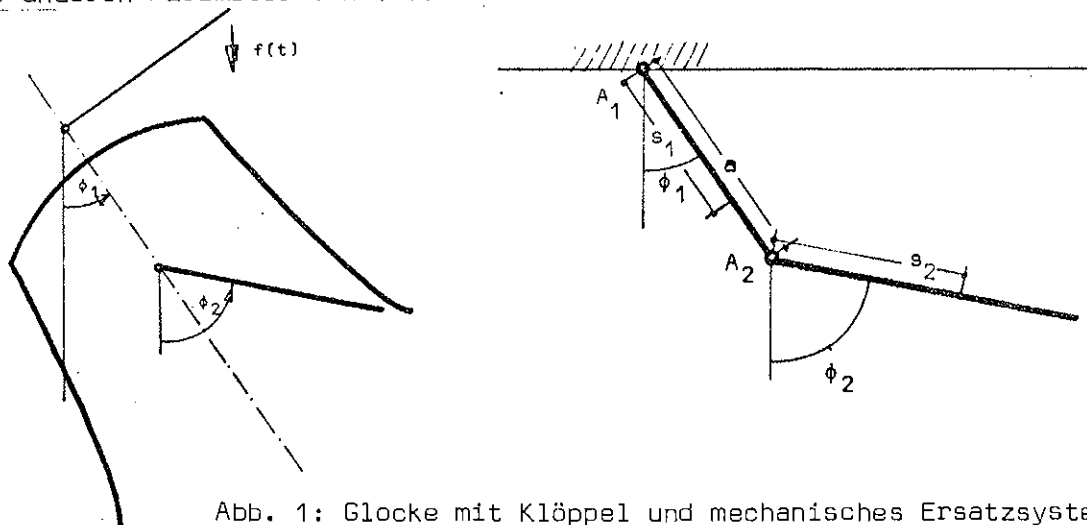


Abb. 1: Glocke mit Klöppel und mechanisches Ersatzsystem

Die Zugkraft kann mittels eines Handschalters abwechselnd 0 oder c gesetzt werden, wodurch die ungedämpfte Glockenbewegung angeregt wird. Um den Stoßvorgang beim Anschlag des Klöppels an die Glocke zu vereinfachen, wurde angenommen, daß die Bewegung der Glocke vom Stoß unbeeinflusst bleibt, während sich die Relativgeschwindigkeit vom Klöppel zur Glocke umkehrt:

$$(\dot{\phi}_1 - \dot{\phi}_{2 \text{ alt}}) = -(\dot{\phi}_1 - \dot{\phi}_{2 \text{ neu}})$$

Es ergibt sich

$$\dot{\phi}_{2 \text{ neu}} = 2\dot{\phi}_1 - \dot{\phi}_{2 \text{ alt}} \quad (3)$$

In der Anlogschaltung läßt sich der Übergang $\dot{\phi}_{2 \text{ alt}} \rightarrow \dot{\phi}_{2 \text{ neu}}$ in einfacher Weise dadurch erreichen, daß der Integrierer, der ϕ_2 liefert, kurz in den IC-Zustand geht, wobei $\dot{\phi}_{2 \text{ neu}}$ nach (3) als neuer Anfangswert übergeben wird. Das logische Signal für die Klöppelanschlagbedingung erhält man durch Abfrage von

$$|\phi_1 - \phi_2| < d$$

mit geeignet gewähltem d (d wurde durch Handpot solange variiert, bis der Anschlag des Klöppels am Oszillographen korrekt war).

Als Skalierungshilfe wurde auch ein CSMP-Programm geschrieben. Dazu müssen zunächst die impliziten Gleichungen (1) und (2) explizit nach $\ddot{\phi}_1$ bzw. $\ddot{\phi}_2$ aufgelöst werden, da die auftretenden algebraischen Schleifen wohl vom Analogrechner, nicht aber vom CSMP verarbeitet werden können. Außerdem hätte eine reine CSMP-Simulation noch den Nachteil einer komplizierten Programmierung des Stoßvorganges und es würde eine Rechengeschwindigkeit resultieren, die eine Echtzeitsimulation kleinerer Glocken nicht erlaubt.

Dem Display der Glocke liegt eine negative Kosinusschwingung zugrunde, die im oberen Teil abgeflacht ist.

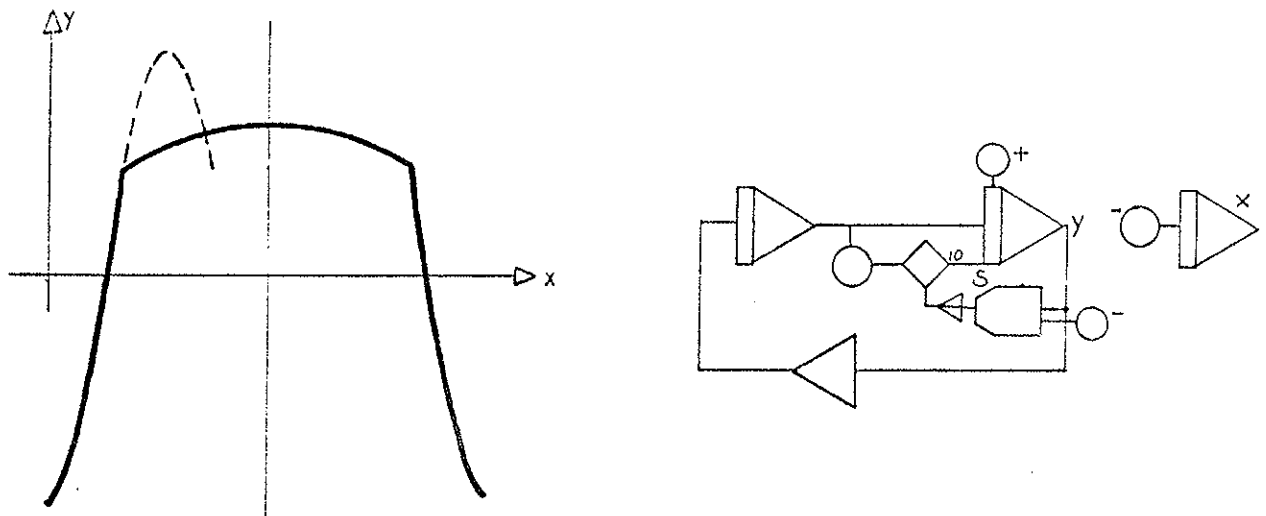


Abb. 2: Darstellung der Glocke aus abgeflachter negativer Kosinusfunktion

Für den Klöppel kann die bereits für die x-Ablenkung der Glocke benötigte Zeitrampe verwendet werden. Die Parameterdarstellung des ruhenden Klöppels lautet (verschoben):

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ -t \end{pmatrix}$$

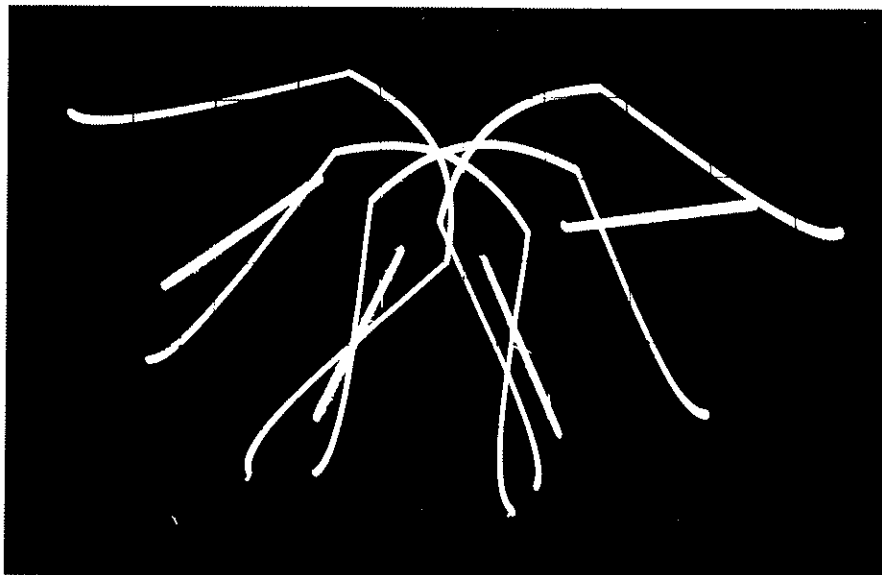
Um einen Punkt (x,y) um den Winkel ϕ um den Ursprung zu drehen, läßt sich die Transformation

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

verwenden. Bei der Darstellung der um ϕ_1 gedrehten Glocke ist natürlich zuerst eine Parallelverschiebung der Glocke so vorzunehmen, daß der Drehpunkt A_1 in den Ursprung fällt. Bei der Darstellung des schwingenden Klöppels muß noch berücksichtigt werden, daß der Aufhängepunkt A_2 bereits die Drehbewegung der Glocke mitmacht und daher die Gerade, die den Klöppel symbolisiert, durch die folgende Gleichung gegeben ist:

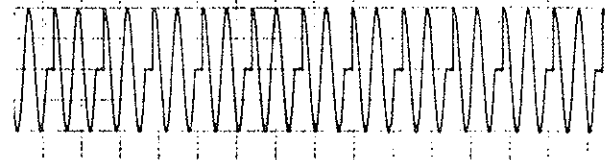
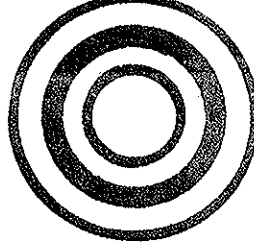
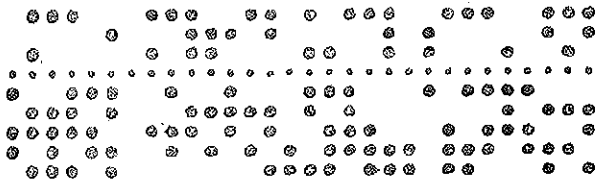
$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \sin \phi_1 \\ -a \cos \phi_1 \end{pmatrix} \begin{pmatrix} \cos \phi_2 & -\sin \phi_2 \\ \sin \phi_2 & \cos \phi_2 \end{pmatrix} \begin{pmatrix} 0 \\ wt \end{pmatrix}$$

Das Gesamtbild am Oszillographen wurde durch abwechselndes Aufschalten von Glocke und Klöppel mit einer Frequenz von 100 Hz erzielt, wodurch vom Betrachter eine einheitliche Bewegung wahrzunehmen war.



Das Beispiel wurde unter anderem mit folgenden Daten gerechnet:

$$\begin{array}{llll} m_1 = 1200\text{kg} & a = s_1 = 0.6\text{m} & i_1 = 0.9\text{m} & g = 9.81\text{m}\cdot\text{sec}^{-2} \\ m_2 = 200\text{kg} & s_2 = 0.6\text{m} & i_2 = 0.8\text{m} & c = 200 \text{ N} \end{array}$$



benützerforum

BILDHAFTE DARSTELLUNG EINES LAUFKRANS AM OSZILLOGRAPHEN

Beispiel aus dem Fortgeschrittenenpraktikum
für Analogrechner (Gruppe Jereb G., Nehfort A.)
F. Breitenecker
Institut für Technische Mathematik
Technische Universität Wien

Einen wesentlichen Moment bei der Behandlung technisch-physikalischer Probleme bildet die anschauliche Darstellung der Methoden und Ergebnisse. Sie ist vor allem für den Nichtfachmann wichtig, der oft mit Zahlen oder technischen Ausdrücken nichts anzufangen weiß, sich jedoch durch praktische Beispiele oder eine Demonstration gerne überzeugen läßt. An der Hybridrechenanlage wurde von Herrn Breitenecker ein Verfahren zur optimalen Steuerung eines Laufkrans entwickelt [1]. Zur Demonstration dieses Verfahrens wurde von den Verfassern eine analoge Schaltung erstellt, die am Oszillographenschirm ein realistisches Bild dieses Laufkrans erzeugt. Alle wesentlichen Funktionen eines natürlichen Laufkrans (Fahren der Katze, Pendeln des Greifers, Heben, Senken, Öffnen und Schließen des Greifers) können - durch andere Schaltungen gesteuert - nachvollzogen werden. Da zusätzlich zur optimalen Steuerung eine Handsteuerung existiert, kann man sehr leicht das optimale Verfahren mit eigenen Versuchen vergleichen.

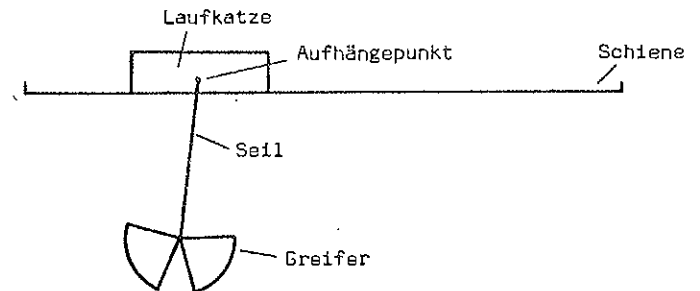
Darstellung am Oszillographenschirm

Um ein beliebiges Bild am Oszillographenschirm darzustellen, verwendet man folgende Methode: Man generiert Ablenkspannungen, die den Leuchtpunkt des Oszillographen entlang den Konturen des Bildes führen. Diesen Vorgang wiederholt man etwa 50mal pro Sekunde und erhält dadurch ein stehendes, flimmerfreies Bild. Teile des Bildes können sich bewegen, wenn man die Zeichnung in jedem Durchlauf etwas verändert. Die Geschwindigkeit dieser Bewegung soll nur so groß sein, daß das Auge ihr mühelos folgen kann.

Das Bild

Das Bild besteht aus vier Teilen: Schienen, Laufkatze, Seil und Greifer. Die Schienen werden nicht verändert. Die Laufkatze kann sich über die Schienen bewegen. Seil und Greifer müssen ihr folgen, wobei außerdem beide um den Aufhängepunkt gedreht werden können. Dieser leuchtet zur besonderen Markierung etwas stärker. Die Länge des Seils kann verändert

werden. Der Greifer kann geöffnet und geschlossen werden. Der Drehpunkt des Greifers leuchtet ebenfalls etwas stärker.



Prinzip der Erzeugung der Ablenkspannungen

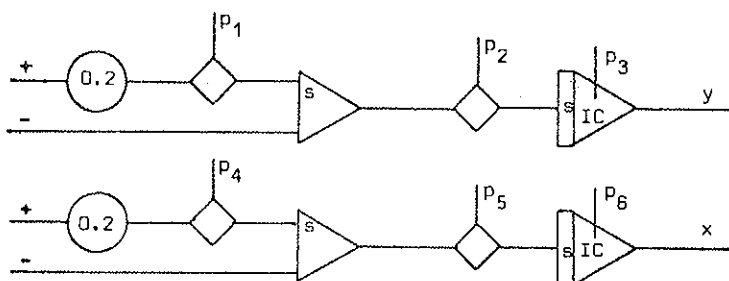
Am einfachsten darzustellen ist eine senkrechte oder waagrechte Linie. Diese erhält man, indem man einen Integrierer über einen konstanten Betrag integrieren läßt. Die Länge der Linie ist abhängig von diesem Betrag und der Integrationszeit.

Während eines vorangehenden IC-Intervalls erscheint am Beginn der Linie ein stärker leuchtender Punkt. Wiederholt man diese "Zeichnung" mindestens 50mal in der Sekunde, so erhält man eine stehende Linie.

Möchte man sowohl eine waagrechte als auch eine senkrechte Linie zeichnen, kann man einen zweiten Integrierer verwenden. Die IC-Zeit des einen Integrierers wird als DP-Zeit des anderen verwendet, und umgekehrt.

Will man eine Figur zeichnen, die aus senkrechten und waagrechten Linien besteht, so muß man diese Linien "aneinanderfügen", d.h. man muß z.B. zuerst nur einen Integrierer integrieren lassen, dann diesen in HOLD versetzen und den nächsten weiterintegrieren lassen. Auf diese Art kann man eine Treppe zeichnen. Für eine geschlossene Figur ist es

zusätzlich notwendig, daß beide Integrierer sowohl in positiver als auch in negativer Richtung integrieren können.



Mit p_1 und p_4 wird die Integrationsrichtung umgeschaltet, mit p_2 bzw. p_5 der Integrierer in einen dem HOLD entsprechenden Zustand versetzt. Mit p_3 und p_6 werden die Integrierer in IC versetzt. Eine Ablaufsteuerung sorgt für die Erzeugung der logischen Pegel.

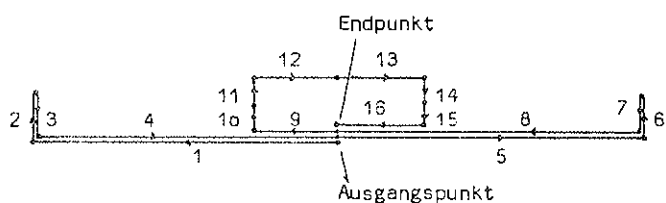
Die Schaltung für die Darstellung besteht aus zwei Teilen: Zeichnung der Schiene und Katze, Darstellung von Seil und Greifer.

Jeder dieser Teile besteht aus zwei Integrierern mit zugehörigen Summieren und Schaltern, der zweite enthält zusätzlich aufgrund der benötigten runden Konturen zwei Transformationsschaltungen. Die Ausgangsspannungen dieser beiden Teile werden addiert.

Zeichnung von Schiene und Katze

Zur Darstellung von Schiene und Katze sollen möglichst geschlossene, symmetrische Figuren gezeichnet werden.

Daher werden Schiene und Katze nach folgendem Plan gezeichnet

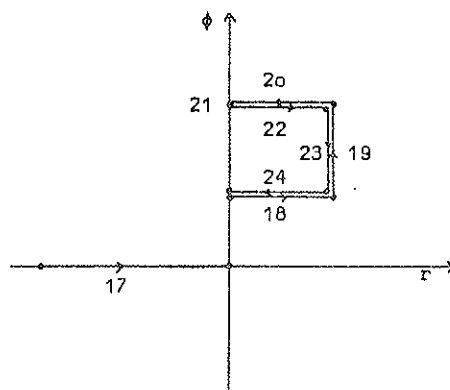
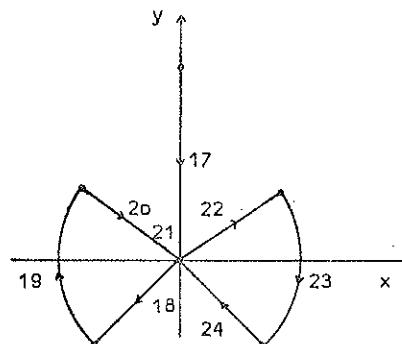


Die Zahlen geben die Zustände in der Reihenfolge an, wie sie durchlaufen werden. Während eines Zustandes wird eine Linie gezeichnet. Die Länge der Linie richtet sich nach der Dauer des Zustandes. Der Ablauf einer bestimmten Zeit ist daher das Kriterium für den Übergang von einem Zustand zum nächsten. (Die Schiene wird zur Erhöhung der Helligkeit doppelt gezeichnet). Für die Dauer der Zustände werden vier Zeitkonstanten τ_1 benötigt. Für die Ablaufsteuerung wird ein Vier-bit-Binärzähler verwendet.

Zeichnung von Seil und Greifer

Die Zeichnung des Greifers besteht aus Kreisteilen und Radien dieses Kreises. Durch Transformation in das Polarkoordinatensystem ergeben sich zur Darstellung nur Radiusänderungen bei konstantem Winkel und Winkeländerungen bei konstantem Radius. Daher kann man zur Erzeugung von r und ϕ ebenfalls die oben beschriebene allgemeine Methode anwenden. Die Seillänge wird bestimmt durch den Anfangswert des r -Integrierers.

Durch Ausnutzung der vorhandenen Symmetrieeigenschaften erhält man daher folgenden Plan:

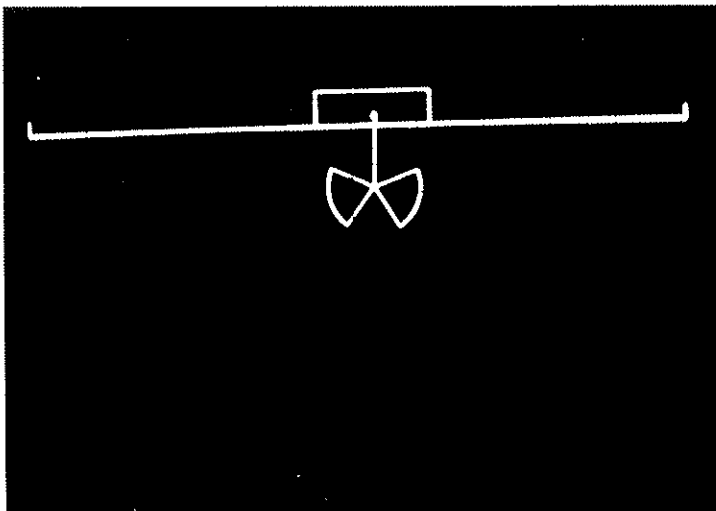


Zustand 17 wird beendet, wenn der r -Integrierer Null erreicht. Durch den Anfangswert dieses Integrierers ist daher die Seillänge bestimmt. Die Drehung um den Aufhängepunkt erfolgt in einer Transformationsschaltung.

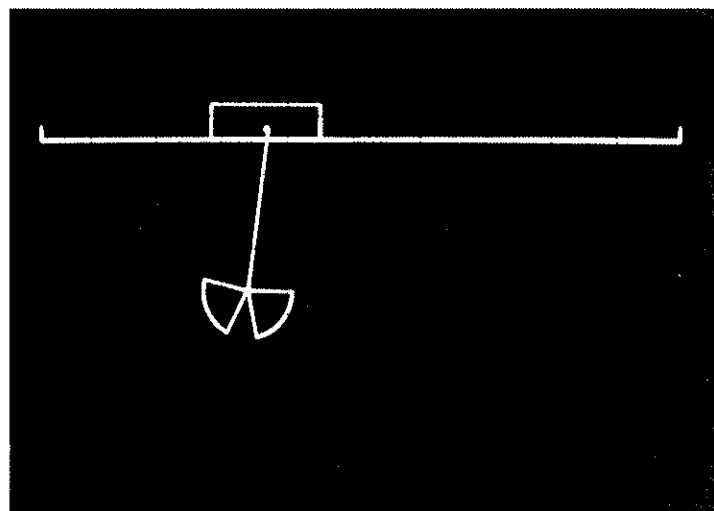
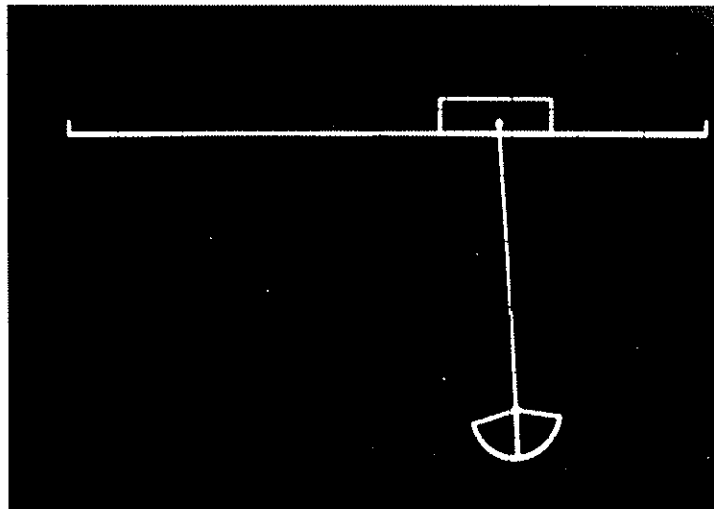
Ablaufsteuerung

Die Ablaufsteuerung wurde mit der parallelen Logik des Analogrechners ausgeführt. Sie entspricht einem Schrittschaltwerk, das durch Zeitkonstanten weiterschaltbar wird. In jedem Zustand werden die entsprechenden logischen Pegel (IC, HOLD, Richtung) für die Integrierer erzeugt. Das Schrittschaltwerk besteht aus einem 4-bit und einem 3-bit Binärzähler und zwei Flip-Flops. Als Zeitkonstanten werden monostabile Kippstufen verwendet. Während der Zeichnung von Katze, Seil und Greifer wird der Weg s zugeschaltet.

Mit einer Simulationsschaltung, die die physikalischen Gegebenheiten nachbildet, kann der Bewegungsablauf beim Fahren der Katze, beim Einholen von Gut und beim Entleeren des Greifers nachgebildet werden. Eine entsprechende Anordnung wurde ausprobiert und zeigte gute Resultate.



Die Bilder geben einen Überblick über die Möglichkeiten der Darstellung des Laufkrans. Die Position der Laufkatze, die Länge des Seils, der Öffnungswinkel und der Pendelwinkel von Seil und Greifer können dynamisch verändert werden.



Literatur

- [1] F.Breitenecker, H.Schafhauser: Modell eines Erzentladers. INTERFACE 4, 1975

PERIODENSUCHE UND PERIODENANALYSE VON VERÄNDERLICHEN STERNEN

Johann Nittmann
 Institut für Astronomie
 Universität Wien

1 EINLEITUNG

Einen Schwerpunkt der naturwissenschaftlichen Forschung am Institut für Astronomie der Universität Wien stellen die magnetischen Sterne dar. Diese Sterne fallen in den Spektralklassenbereich B7V - F5V und besitzen peculiare Eigenschaften, deshalb auch der Kurzname Ap-Sterne. Eine periodische Variation der Intensität der Spektrallinien, der Helligkeit und eines zum Teil beträchtlichen Magnetfeldes zeichnet diese Sterne aus. Zudem zeigen einige dieser Sterne eine kurzzeitige periodische Helligkeitsänderung mit einer Periodenlänge von ca. 2 Stunden, die der Haupthelligkeitsperiode überlagert ist. Diese kurzzeitigen Helligkeitsänderungen kleinster Amplitude liegen an der Grenze der Beobachtungsgenauigkeit. Um hier eine Periodizität in den stark streuenden Daten zu finden, müssen mathematische Methoden die Periodensuche übernehmen.

2 LAF LER - KINMAN METHODE

Aus physikalischen Überlegungen oder anderwertiger Information weiß man meist den ungefähren Bereich, in dem man einen Trend in den Daten erwarten kann. Existieren n Beobachtungswerte x_i ($i=1, n$) mit dem Mittelwert $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$, so bildet man für eine gewählte

Periode

$$\theta = \frac{\delta^2}{\sigma^2} = \frac{\sum_{i=1}^{n-1} (x_{i+1} - x_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

Liegt kein Trend in den Daten vor, wird dieser Parameter $\theta \geq 2$ sein. Erhalten wir aber Parameterwerte, die weit unter 2 sind, so kann mit großer Wahrscheinlichkeit von einem Trend in den Daten gesprochen werden. Eine genauere Diskussion erfolgt in NITTMANN (1978).

Im ersten Computerlauf wird der gesamte vermutete Periodenbereich mit grober Periodenschrittweite untersucht. Die dabei aufscheinenden Minima des Parameters θ können in einem zweiten Lauf weiter untersucht werden, bis schließlich das absolute Minimum bei kleinster sinnvoller Schrittweite gefunden wird. Die Berechnung, sowie die graphische Darstellung der θ -Werte erfolgte an der CDC CYBER 73, vor allem, da keine besonderen Anforderungen an die graphische Darstellung gestellt wurden.

Tritt nun der Fall ein, daß mehrere Perioden etwa gleichgute Werte besitzen, müssen die Perioden weiter untersucht werden, bzw. eine andere Methode zur Periodensuche angewendet werden. Bevor wir eine Methode zur weiteren Untersuchung besprechen, soll eine zweite Methode zur Periodensuche angeführt werden.

3 FOURIERANALYSE NACH DEEMING

Im Gegensatz zur klassischen Fourieranalysemethode sind bei der Methode von DEEMING keine äquidistanten Zeitwerte bzw. x -Werte notwendig. Die komplexe Fouriertransformierte $F(v)$ einer Funktion $f(t)$ ist

$$F(v) = \int_{-\infty}^{+\infty} f(t) e^{i2\pi vt} dt$$

definiert. Diese Fouriertransformierte können wir in der Praxis nicht bilden, da wir erstens nur innerhalb eines endlichen Zeitbereiches beobachten können, zweitens nur zu diskreten Zeitpunkten eine Messung erfolgen kann. DEEMING definiert deshalb eine endliche Fouriertransformierte $F_N(v)$ und ein "Spektral Window" $\delta_N(v)$

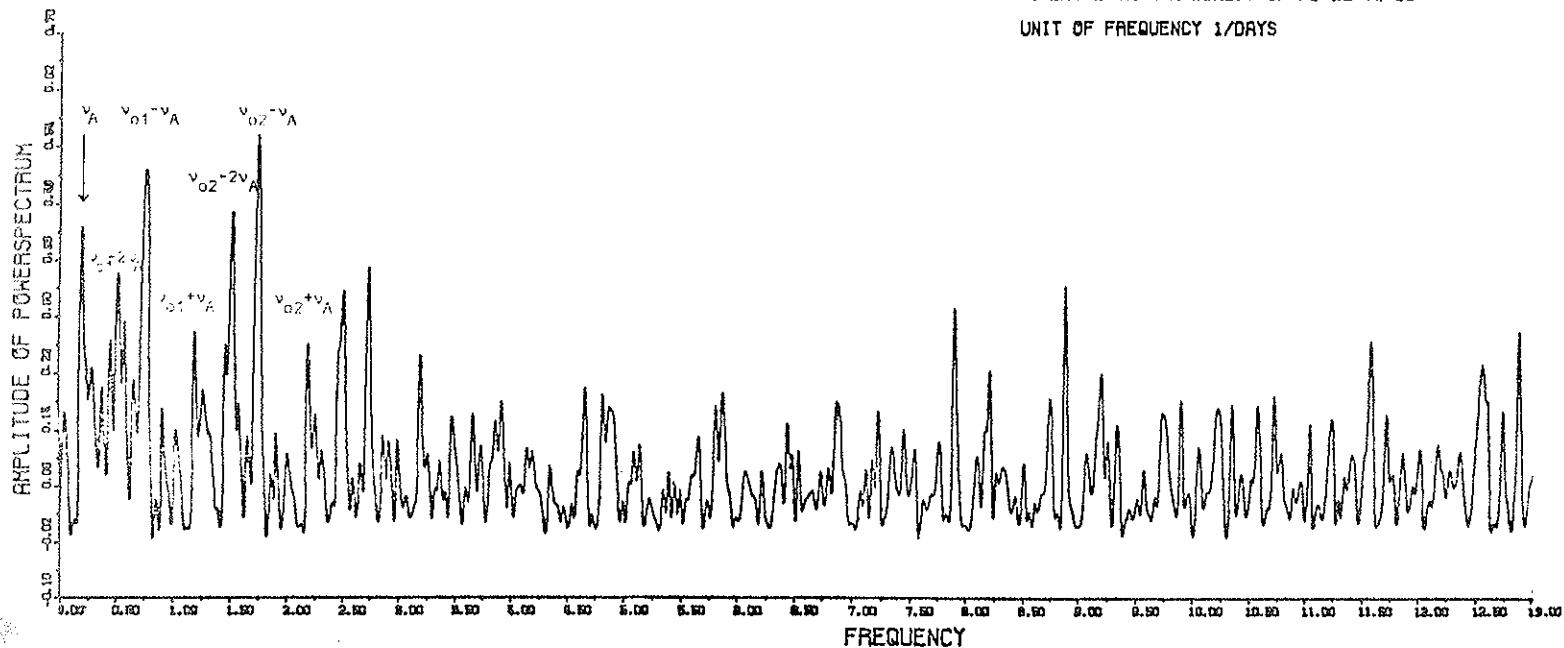
$$F_N(v) = \sum_{k=1}^N f(t_k) e^{i2\pi vt_k}$$

$$\delta_N(v) = \sum_{k=1}^N e^{i2\pi vt_k}$$

$F_N(v)$ ist nun die Faltung der wahren Fouriertransformierten $F(v)$ mit dem Spectral Window $\delta_N(v)$, welches all die Informationen enthält, die durch die Anordnung der x bzw. t -Werte in $F_N(v)$ entstehen

$$F_N(v) = F(v) * \delta_N(v) \equiv \int_{-\infty}^{+\infty} F(v-v') \delta_N(v') dv'$$

Enthält $\delta_N(v)$ einen Peak an der Stelle v_0 und existiert ein Peak an der Stelle v_A im "Powerpektrum", so werden an den Stellen $v_0 + v_A$ und $v_0 - v_A$ zwei weitere Peaks im "Powerpektrum" aufscheinen.



SPECTRAL WINDOW OF DATA

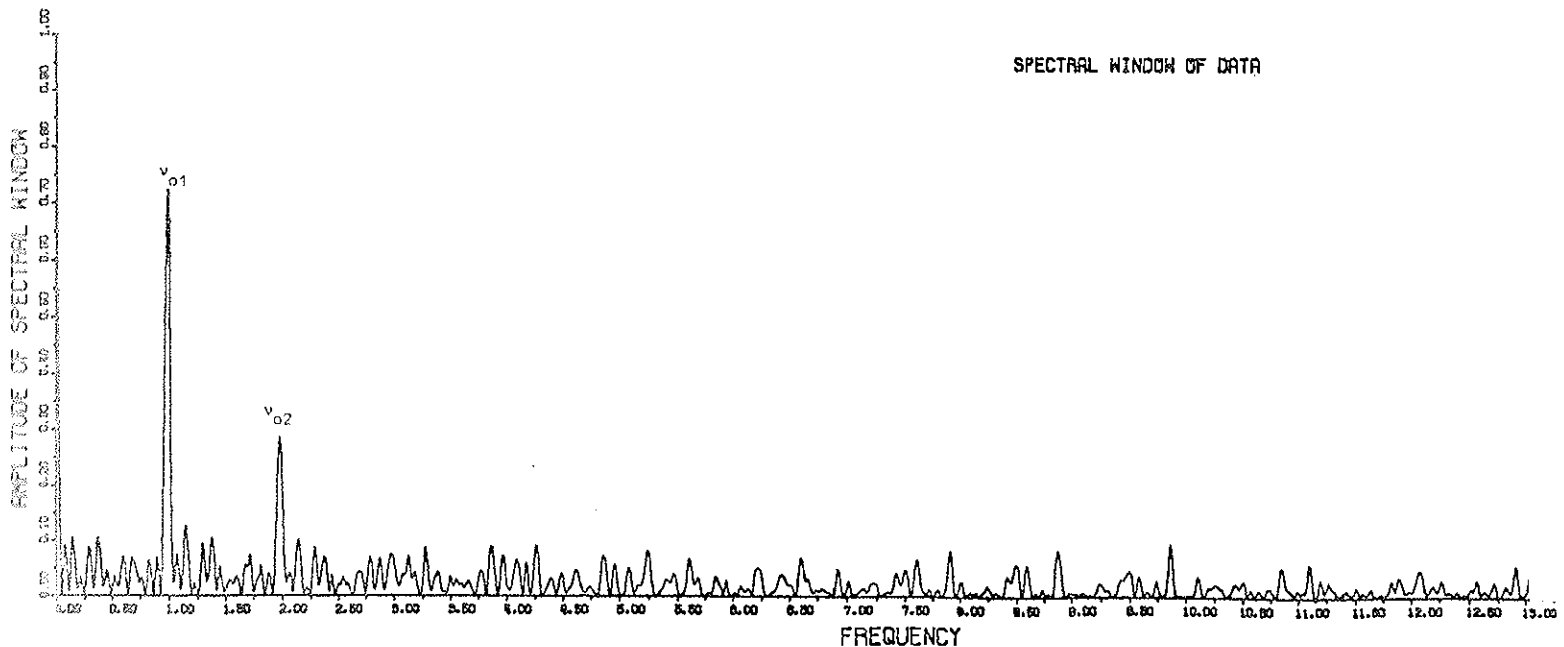


Abbildung 1

Abb.1 zeigt das Fourierspektrum von Radialgeschwindigkeitmessungen des Ap-Sternes HD 224801. Die Periode von 3.74 Tagen ist leicht auffindbar. Die Verwendung eines Plotters mit der notwendigen Softwareunterstützung ist von grundlegender Bedeutung für die Anwendung dieser Methode. Beides konnte an der Hybridrechenanlage des EDV-Zentrums der Technischen Universität Wien gefunden werden.

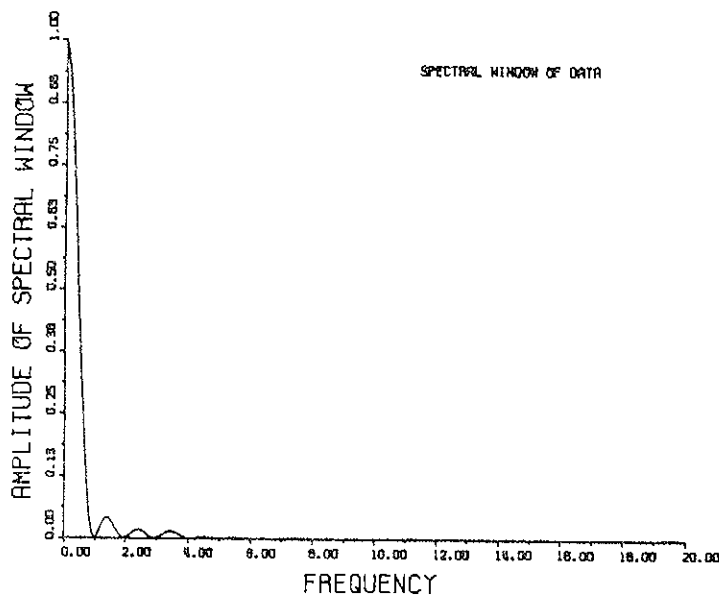
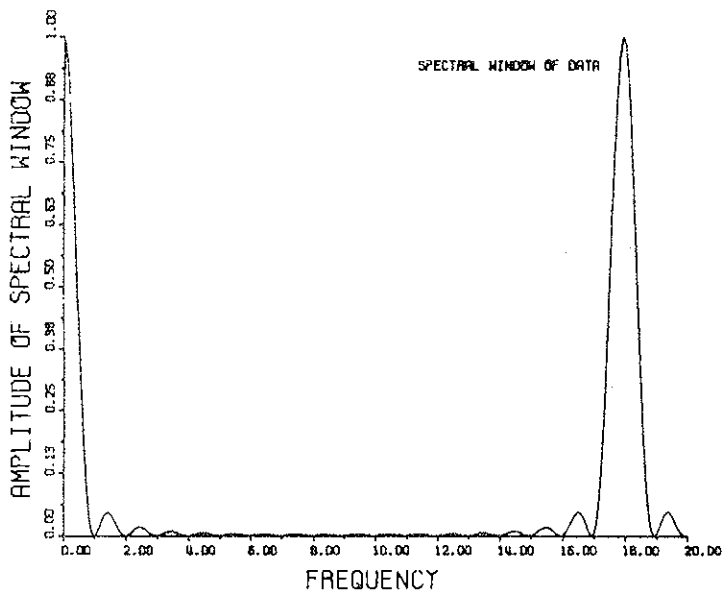
4 SPLINEGLÄTTUNG UND FOURIERANALYSE

Erhalten wir bei der Anwendung der beiden vorhin besprochenen Periodensuchmethoden kein eindeutiges Ergebnis, so müssen die Daten weiter untersucht werden. Hierfür wurde vom Autor die reichhaltige Software, die an der Hybridrechenanlage zur Verfügung steht, verwendet (Interpolations- und Plotterpaket IPP1). Die durch die beiden obigen Perioden-

suchmethoden gefundenen Perioden sollen zunächst graphisch dargestellt werden. Eine Splineglättung soll der erste Schritt einer weiteren Untersuchung sein. Als zweiter Schritt soll das Fourierspektrum der Periode und der Splineglättung gleichzeitig in einer Zeichnung geplottet werden.

Die Splineglättung nach Späth bietet die Möglichkeit der Wahl des Glättungskoeffizienten. Durch die Verwendung eines Graphic Display (Elektronix 40D6) kann eine rasche inter-

aktive Änderung des Glättungskoeffizienten durchgeführt werden. Durch den Vergleich der Fourierspektren der Meßwerte sowie der Glättungskurve konnte die Größe der Streuung erkannt werden. Hochfrequente Amplituden im Powerspektrum der Splinefunktion konnten nur von der Datenanordnung verursacht werden. Um hier sicher zu gehen, wurde auch das "Spektral Window" der Periode geplottet. Abb.2 und Abb.3 zeigen verschiedene Testdaten, die mit dieser Methode untersucht wurden. Die Testdaten wurden mit Hilfe des Unterprogrammes MLAREN (Zufallszahlengenerator) erzeugt.



GLÄTTUNGSKOEFFIZIENT = 0.020

GLÄTTUNGSKOEFFIZIENT = 0.200

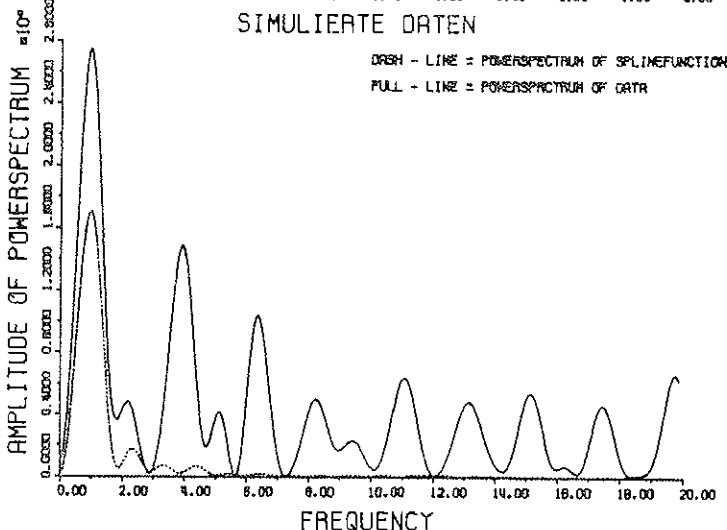
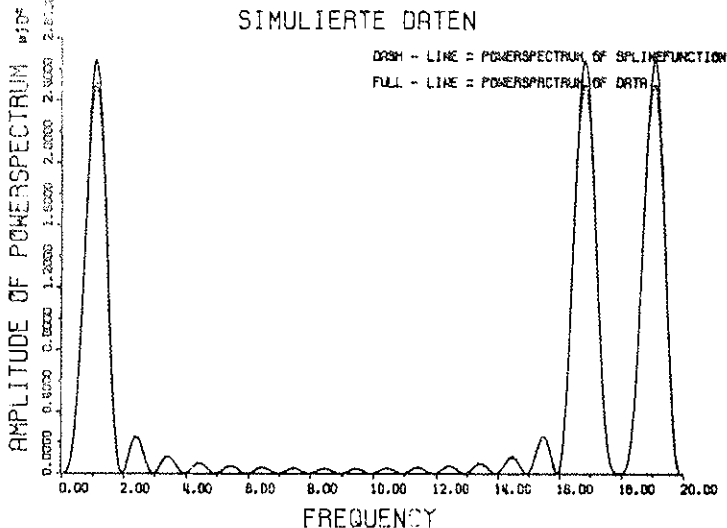
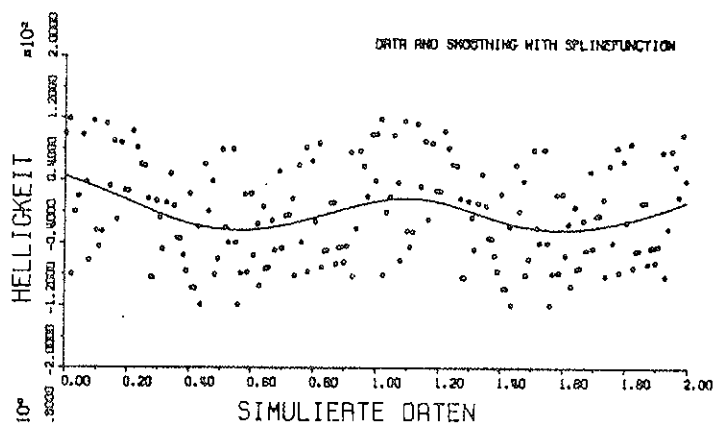
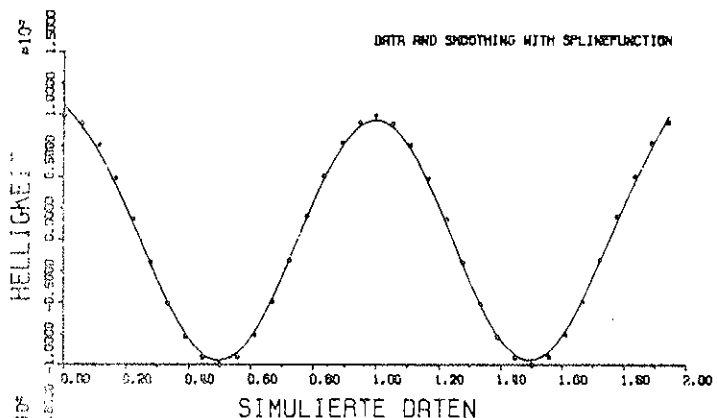


Abbildung 2

18 äquidistante Beobachtungswerte ohne Streuung

Abbildung 3

94 nicht äquidistante Beobachtungswerte mit "idealer" Streuung

Mit diesen drei Methoden wurden neue Helligkeitsmessungen des Ap-Sternes HD 2248D1 untersucht. Abbildung 4 zeigt das Ergebnis der $2^h 07^m$ Periode des Johnson U-Filters.

Der Autor möchte sich beim gesamten Team der Hybridrechenanlage für die hilfsbereite Unterstützung während der Test- und Produktionsphase bedanken.

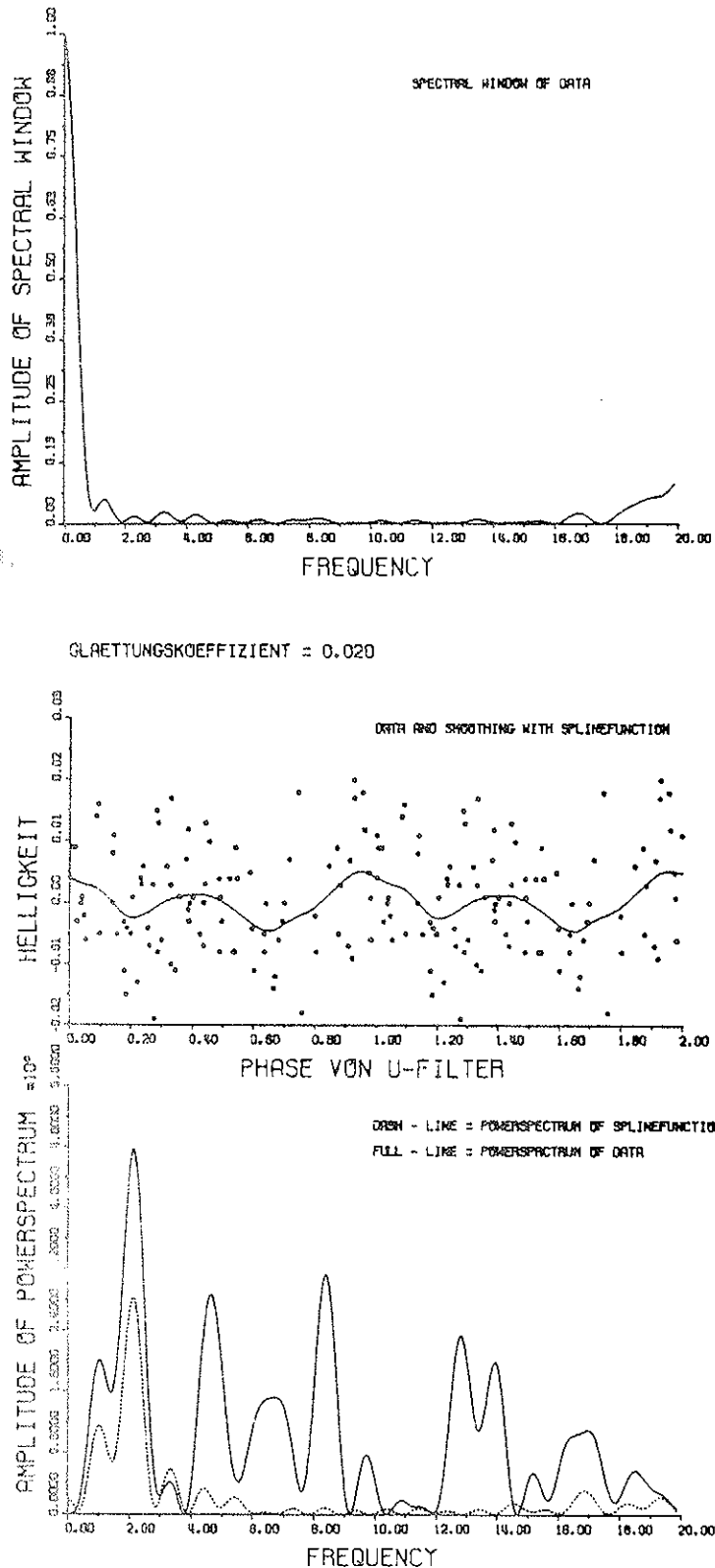


Abbildung 4

Praktisches Beispiel:
93 Beobachtungswerte
des Johnson U-Filters,
Periode $2^h 07^m$

SUMMARY

The investigation of the existence of a short period brightness variation in magnetic stars is only possible with the aid of mathematical period search methods. We applied two different mathematical methods, first on simulated data then on observed data. Using the LAFLEER-KINMAN method we calculated a test parameter for a number of trial periods. For test parameters much smaller than 2 there is a trend in our data. Using the fourier-analysis method of DEEMING for unequally spaced data, a second period search method was available. From the simulated data we found good agreement between these two independent methods. In case of no definite period or more than one period further analysis was necessary. For these purposes the author used the sophisticated software of the Hybrid Computer Centre. For the first step we used a Spline function for smoothing the observed data. Next we represented the fourier spectra of the data and the smoothed curve simultaneously in one plot, this enabled us to distinguish between the real information in our data and the noise.

LITERATUR

- F. BLOESER, C.W. OBERHUBER (1977): Das Interpolations- und Plotterpaket IPP1; Institut für Num. Mathematik Bericht Nr. 28/77
- T.J. DEEMING (1975): Fourier Analysis with Unequally-Spaced Data; Ap.a.S.S. 36, 137
- J.LAFLEER, T.D. KINMAN (1965): An RR Lyrae Star Survey with the Lick 20-Inch Astrograph; Ap.J. Sup. 11, 216
- J. NITTMANN (1978): Kurzzeitige Helligkeitsschwankungen des Ap-Sternes HD 2248D1; Dissertation, Universität Wien
- K.D. RAKDSCH (1973): Magnetische Sterne; Acta Phys. Austria

INTERFACE November 1978