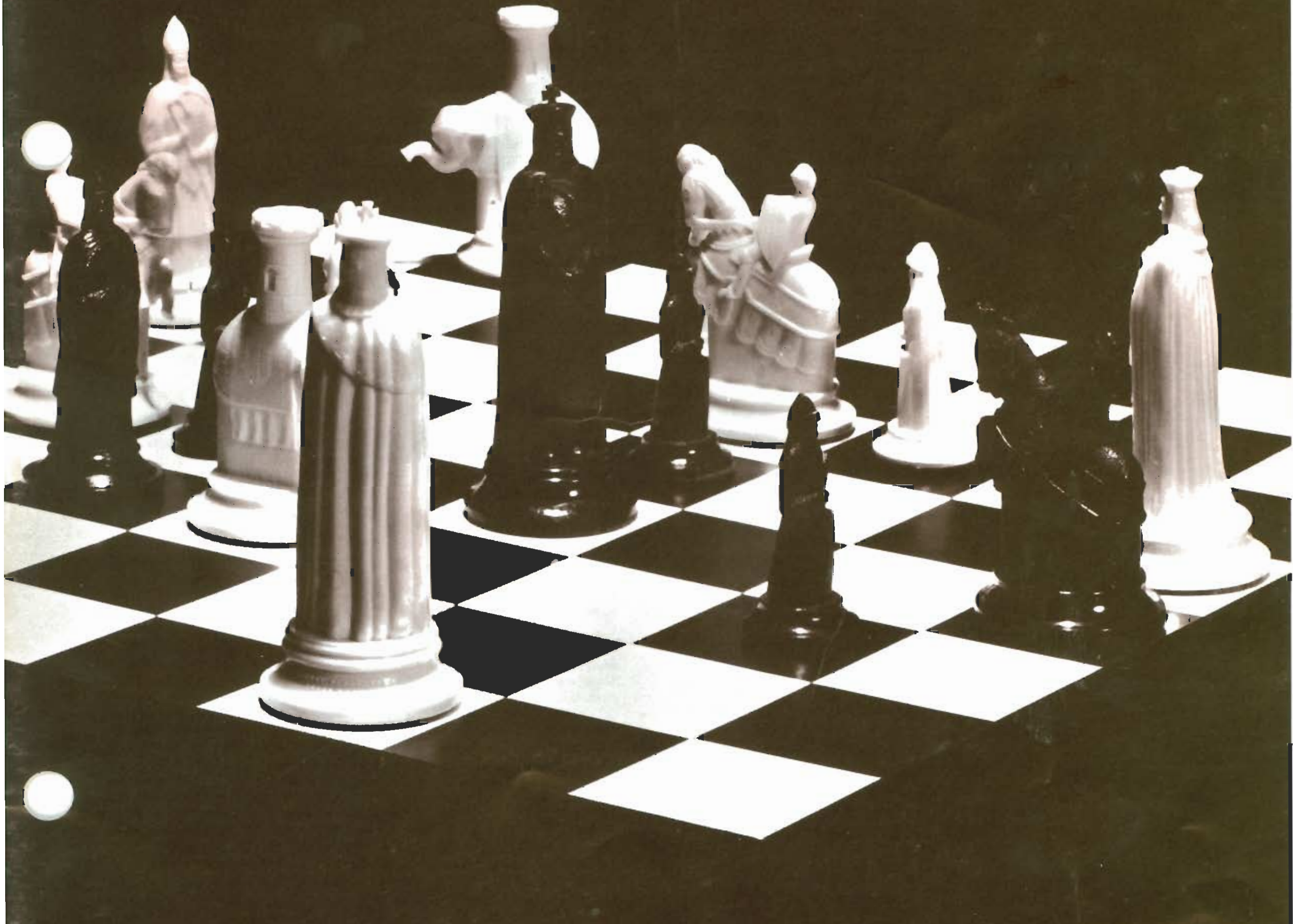
HEWLETT-PACKARD

KEYBOARD

VOL. 6 NO. 4



NEW FIELD EDITORS

To make it easier for our readers in many European countries to communicate with *KEYBOARD*, we are very pleased to welcome the following new field editors to the *KEYBOARD* staff. Readers living in the countries and areas now separately covered may send any correspondence and contributions for *KEYBOARD* publication to the appropriate editor below:

FIELD EDITOR	COUNTRY/AREA
Werner Hascher	East Bloc
Dick Adaway	England
Elisabeth Caloyannis	France
Jaap Vegter	Holland
Elio Doratio	Italy
Terry Adams	Scandinavia
Jose L. Barra	Spain
Heinz Neiger	Switzerland

Ed Hop, who formerly covered all of Europe, is still field editor for Germany. Please see the complete list on this page for the full addresses of the new editors.

OVERVIEW

It is a pleasure to announce in this issue the results of the 1974 Calculator System Application Contest, U.S.A. branch. Two U.S.A. winners were chosen instead of one, for reasons discussed in the article on page 2, which includes a list of all contestants and abstracts of their articles. The two winning articles also appear in this *KEYBOARD*, starting on pages 4 and 8. The outside-U.S.A. branch winners will be selected and announced as soon as possible. The results are later than expected because of the deadline extension to September 17.

If you have a need for modular furniture to allow assembling your HP calculator system in an aesthetically-pleasing array for your professional office environment, a solution is available. See the description of the new units on page 6.

The CROSSROADS this time features "The Art of Science", and deals with alternate approaches to solving mathematical problems. Several sample problems are included; we hope you will enjoy solving them.

Many of our readers find that the various ideas in Programming Tips help them save calculator memory space, time, and operator errors. We would welcome any ideas you have for HP programmable calculators. Send them to the field editor nearest you or directly to the *KEYBOARD* Loveland address.

A B Sperry

TABLE OF CONTENTS

Page

Features

The Crossroads

Programming Tips

Announcements

APPLICATIONS INFORMATION
FOR HEWLETT-PACKARD CALCULATORS
PUBLISHED AT P.O. BOX 301,
LOVELAND, COLORADO 80537

Editor: A.B. Sperry
Art Director: L.E. Braden
Artist/Illustrator: H.V. Andersen

Field Editors: **ASIA**--Jaroslav Byma, Hewlett-Packard Intercontinental, 3200 Hillview Avenue, Palo Alto, California 94304; **AUSTRALASIA**--Bill Thomas, Hewlett-Packard Australia Pty. Ltd., 31-51 Joseph Street, Blackburn, 3130 Victoria, Australia; **CANADA**--Larry Gillard, Hewlett-Packard Canada Ltd., 6877 Goreway Drive, Mississauga, Ontario L4V 1L9; **EAST BLOC, EUROPE**--Werner Hascher, Hewlett-Packard Ges.m.b.H., Handelskai 52/3, A-1205 Vienna, Austria; **ENGLAND**--Dick Adaway, Hewlett-Packard Ltd., 224 Bath Road, Slough SL 1 4DS, Bucks, England; **FRANCE**--Elisabeth Caloyannis, Hewlett-Packard France, Quartier de Courtaboeuf, Boite Postale No. 6, F-91401 Orsay, France; **GERMANY**--Ed Hop, Hewlett-Packard GmbH, Herrenberger Strasse 110, 703 Böblingen, West Germany; **HOLLAND**--Jaap Vegter, Hewlett-Packard Benelux N.V., Weerdestein 117, NL-1011 Amsterdam, Holland; **ITALY**--Elio Doratio, Hewlett-Packard Italiana Spa, Via Amerigo Vespucci 2, I-20124, Milano, Italy; **JAPAN**--Akira Saito, Yokogawa-Hewlett-Packard Ltd., 59-1, Yoyogi 1-chome, Shibuya-ku, Tokyo 151; **LATIN AMERICA**--Ed Jaramillo, Hewlett-Packard Intercontinental, 3200 Hillview Avenue, Palo Alto, California 94304; **SCANDINAVIA**--Terry Adams, Hewlett-Packard Sverige AB, Enighetsvägen 3, Fack, S-161 20 Bromma 20, Sweden; **SOUTH AFRICA**--Denis du Buisson, Hewlett-Packard South Africa (Pty.) Ltd., 30 de Beer Street, Braamfontein; **SPAIN**--José L. Barra, Ataio Ingenieros SA, Enrique Larreta 12, Madrid 16, Spain; **SWITZERLAND**--Heinz Neiger, Hewlett-Packard Schweiz AG, Züricherstrasse 20, CH-8952 Schlieren, Zürich, Switzerland; **EASTERN U.S.A.**--Stan Kowalewski, Hewlett-Packard Co., W120 Century Road, Paramus, New Jersey 07652; **MIDWESTERN U.S.A.**--Jerry Reinker, Hewlett-Packard Co., 330 Progress Road, Dayton, Ohio 45449; **SOUTHERN U.S.A.**--Bob McCoy, Hewlett-Packard Co., P. O. Box 28234, Atlanta, Georgia 30328; **WESTERN U.S.A.**--Robert C. Reade, Hewlett-Packard Co., 3939 Lankershim Boulevard, North Hollywood, California 91604.

NOTE TO CLINICAL PATHOLOGY LAB DIRECTORS

If you own a 9810A Calculator, are you aware that the following items are available to help you increase the utility of your calculator in your lab?

1. HP 9810 Clinical Pathology Pac:
   Manual part number (p/n) 09810-75350; recorded magnetic cards p/n 09819-75350.
   - Pac consists of routines for blood gas analysis, electrophoresis calculations, lipoproteins, LDH isoenzymes, red cell indices, Schilling Test, Urea Clearance, Creatinine Clearance, Phosphatase Calculations, Fecal fat determination, total blood volume, CO_2 content and tension of human blood, Hemoglobin oxygen saturation; plus mean, standard deviation, standard error, histogram, and linear regression routines.
2. HP 9810 Programming Manual for the Clinical Laboratory:
   Manual (only) p/n 09810-75340.
   - Manual consists of 10 chapters (lessons) and 3 useful appendices treating the use and programming of the 9810 and peripherals in the Clin-Lab environment. Written by Doctors William R. Dito, Russell H. Clark, and Gerald G. Hoffman, this manual has been used at numerous ASCP tutorial workshops. (Request data brochure #5952-2375).
3. HP 9810 Clinical Lab Radioimmunoassay Pac:
   Pac p/n 09810-75262.
   - Pac consists of a modular program which accommodates up to 20 known concentrations with up to 4 replicates each for standard curve preparation. A variance-weighted least squares regression is performed using the Logit of % Bound (B/Bϕ) versus the log of concentration. Any number of patient unknowns can be automatically calculated from the standard curve. Optional use of the HP 9863 Punched Tape Reader and HP 9862 Graphic Plotter is provided. (Request Data Brochure #5952-2428).
4. HP Radioimmunoassay Theory for Health Care Professionals:
   Manual (only) p/n 09830-75250.
   - Manual is a comprehensive monograph treating all aspects of RIA theory, chemistry and mathematics. Written by RIA researcher and clinician, Richard C. Rodgers, this document contains one of the most understandable descriptions to be found anywhere plus a complete bibliography.

These items plus many other useful pacs on general statistics and mathematics are available through your local Hewlett-Packard Calculator salesman. Simply call him and request any of the above. Also, have him send you the new 9810 Software Information Sheet (#5952-8908) and General Calculator/Peripheral Brochure (#5952-2404). It is our sincerest wish to help Clinical Laboratorians in all ways we can.

DATA INTERCHANGE PUZZLE (9820A/9821A)

Texas A & M University Professors James N. Shapiro and Anthony F. Gangi submitted the following puzzle which should appeal to HP 9820A and 9821A users. Although this is not a contest, you may find it an interesting problem.

It is easy to sequentially interchange data among several registers in the HP 9820A or 9821A Calculator if an extra register such as X is used as temporary storage while interchanging data among, say, the A, B, and C registers. But how do you interchange the data among two or more registers without having to use an extra register? The answer to this puzzle won a bet for one of the above professors. Send us your answer as to whether this can be done, and if so, how.

1974 Calculator System Application Contest two U.S.A. Winners

KEYBOARD congratulates Keith Mitchell, Seattle, Washington, and Dr. Alan A. Wray, Fayetteville, Arkansas, for their prize-winning entries in the 1974 Calculator System Application Contest, U.S.A. branch. A pocket calculator, either HP-45 or HP-80, will go to each of these winners. The outside-U.S.A. branch of the contest is still open as this is being written; a winner will be announced as soon as possible after the extended closing date, September 17.

Mr. Mitchell submitted his winning article describing a set of programs for the HP 9830A used in design of airplane stability augmentation systems. His article is reproduced in this issue, starting on page 8.

Dr. Wray entered his HP 9810A program which plays nontrivial games of regular chess against a human opponent. The versatility and logic levels used constitute a historic first for this application of a programmable calculator with a memory of comparable size. His article starts on page 4.

When the smoke cleared away and all of the entries had been received in the U.S.A. branch of the contest, the hard part began--judging the 34 articles. Applications ranged from satellite orbit mapping and plotting to underwater pipelaying; from curve-fitting inheritance parameters to calculating the temperature at the center of a can of soup in sterilization; and from design and plotting of sundial calibrations to simulating a Morse code practice unit.

Since the contest was for unusual applications, some of the more common fields of calculator use were not difficult to eliminate. Many of the longest-surviving entries had nearly equal unusual qualities; the final winners were chosen for both their uniqueness in programmable calculator history and their sophisticated applications of the calculators' logic capability. The decision to award duplicate prizes was made because our judges felt it impossible to eliminate either of the candidate entries which, while representing opposite ends of the application spectrum and programmed on machines of different power, both use the full capabilities of their respective calculators.

We regret that more prizes could not be awarded in the U.S.A. branch of the contest. To the other contestants, we give our sincere thanks for participating. We hope to give each of you an opportunity to participate again in the 1975 contest.

The quality of most of the U.S.A. entries was excellent; KEYBOARD will publish as many of them as space permits in future issues. Meanwhile, shown below are the titles of all contest entries, the authors' names in alphabetical order, and abstracts of the articles.

1. PIPER CHEROKEE 6 LOAD AND BALANCE DETERMINATION (9100B)
   by Donald L. Brown, Madison, Wisconsin
   Using weight data according to FAA rules as input, this program calculates and prints the center of gravity and weight for a Piper Cherokee 6 airplane.
2. MICROWAVE SEMI-AUTOMATIC NETWORK ANALYZER SYSTEM (9820A)
   by Ronald L. Chilluffo and John M. Eardley, Washington, D.C.
   This paper describes a calculator-based data acquisition system used in conventional microwave network analysis and for various automated instrumentation applications.
3. VERTICAL CURVE (9100B)
   by James K. Delmarter, Bakersfield, California
   This paper describes a program for running vertical alignment for streets, sewers, etc., incorporating vertical curves.
4. GRADE STORAGE PROGRAM - GRADEBOOK (9100B)
   by J.E. Dooley, Florence, South Carolina
   These programs store student grades on magnetic cards and allow rapid calculation of averages, etc., after any number of tests.
5. SINGLE GENE AND MULTIFACTORIAL MODELS OF INHERITANCE (9810A)
   by Margaret M. Freedman, St. Louis, Missouri
   This program set calculates best fit parameters and chi-square goodness of fit for a single gene and a multifactorial model of inheritance where the trait has multiple thresholds.
6. SATELLITE ORBIT COMPUTATION (9810A)
   by Lt. Peter M. Fried, Madison, Wisconsin
   This paper describes a system of 9810A programs which calculate and plot orbits of satellites on either a Mercator projection or a Hammer equal-area projection of the world, which it also plots. Rise and set times are calculated and printed. A satellite's position can be plotted for a specified time.
7. COX-SPRAGUE SYSTEM OF SCORING YACHT RACES (9810A)
   by Peter E. Galloway, Norwalk, Connecticut
   This program calculates and records scores using the Cox-Sprague method for yachts participating in a series of two or more races.
8. HISTOGRAM GENERATOR AND PLOTTER (Program 1) (9810A)
   by Dr. Walter J. Gamble, Boston, Massachusetts
   This program generates a histogram plot with up to 279 cells. Plottable statistics calculated include mean, standard deviation, standard error of mean.
9. CUMULATIVE PERCENT PLOTTER (Program 2) (9810A)
   by Dr. Walter J. Gamble, Boston, Massachusetts
   This program plots a cumulative percent curve, derived from stored data of Histogram program (Program 1 above). Has option to draw horizontal lines at user-defined percentages.
10. PROJECT MILESTONE TRACKING AND CHARTING (9830A)
    by John J. Gassner, Florissant, Missouri
    This paper describes a set of programs that allow the user to automatically prepare and plot schedules for planning of project milestones, and to automatically modify and update these schedules to reflect milestone completion and/or slippages whenever required.
11. POLYPHASE TEST BOARD (9100B)
    by Robert E. Gifford, P.E., Fort Wayne, Indiana
    This paper describes a 9100B calculator-based automatic test system for multiphase kilowatt-hour meters.
12. POINT COUNT BIDDING FOR BRIDGE BEGINNERS (9810A)
    by Alfred H. Hausrath, San Bernardino, California
    This program deals simulated bridge hands, checks each bid for correct bidding sequence, and evaluates the point count.

13. A PLOTTING PROGRAM FOR STEREOSCOPIC PROJECTION DRAWINGS (9100B)
by George W. Hopkins and Prof. Roland V. Shack, Tucson, Arizona
This program calculates and plots pairs of stereoscopic perspective projection drawings.

14. STEREO PLOTTING PROGRAM (9810A)
by James L. Jackson, San Mateo, California
This program makes two perspective drawings of a supposed object from slightly different viewpoints to create a stereogram. Has independent matrix and translation capabilities.

15. HP 9830A PLAYS BLACKJACK
by Gregory R. Janes, Seattle, Washington
This program provides a standard blackjack card game useful for demonstration and promotional purposes as well as a 'fun' game. Results are printed out as game progresses.

16. CALENDAR/DATE PROGRAM (9830A)
by Gregory R. Janes, Seattle, Washington
This program calculates and prints on the 9866A any user-specified month. It also outputs the day of the week for a specified date, or the number of days between two specified dates.

17. HEXAPAWN (9820A)
by Dr. Wilbur E. Jorgenson, Livermore, California
This program allows the calculator to play a game of Hexapawn against a human opponent. The calculator learns to improve its strategy as more games are played in a series.

18. SATELLITE POPULATION FORECAST (9810A)
by Preston M. Landry, Colorado Springs, Colorado
This program computes expected satellite population for any time span desired, and the associated 99% confidence interval. Output may be used to assist estimation of future tracking system workload and computational requirements.

19. OPTIMIZING OPERATION OF A CHEMICAL PROCESSING COMPLEX (9830A)
by K. W. Lessey, St. Helens, Oregon
This paper describes an application of linear programming with the HP 9830A, solving 32 equations with 60 variables including energy consumption, steam production, ammonia, urea, and nitrogen solution production and demand, etc., to optimize operation of a fertilizer manufacturing plant.

20. MAINTENANCE SPARE PARTS INVENTORY (9830A)
by K. W. Lessey, St. Helens, Oregon
This system of parts inventory control maintains 6000 items, using the HP 9830A Inventory Control package, with additional programs to generate purchase orders for low-inventory parts and to notify the maintenance group that certain parts should be repaired and returned to the warehouse.

21. MACHINE INVENTORY SIMULATION (9810A)
by David P. Lilly and G. McCalley, Sr., Aiken, South Carolina
This paper describes a simulation to determine machine time in production, considering machine life and replacement time, reorder points, and inventory levels.

22. AIRPLANE STABILITY AUGMENTATION SYSTEM DESIGN (9830A)
by Keith Mitchell, Seattle, Washington
This article describes a series of linked, completely general purpose control system design programs, written without specialization to airplanes, which enable extremely rapid and convenient design of multiloop control systems.

23. ANALYSIS OF ANALOG DATA FROM MECHANICAL TESTS (9830A)
by Dr. William Oldfield, Westerville, Ohio
This article describes an application of a 9830A interfaced with a high-speed transient recorder. The system analyzes data and outputs load to yield, fracture points of sample, and other statistics. Output plots are made.

24. IMPROVED COURSE GRADING WITH A 9820A CALCULATOR
by Prof. Lloyd D. Partridge, Memphis, Tennessee
This program processes raw test scores to give grades in standard deviations from the mean, histogram of distribution, count of number of grades in user-selected range and card-stored permanent record. Easy correction of errors is provided.

25. DIFFUSION OF HEAT IN A FINITE CYLINDER (9820A)
by Joseph D. Ramsay, Camden, New Jersey
This program solves a set of 25 finite-difference differential equations by the Runge-Kutta method, as an approach for simulating heat transfer in sterilizing processes.

26. SATELLITE POSITION FROM ELEMENTS (9810A)
by Billy G. Sanders, Ent Air Force Base, Colorado
This program computes geodetic latitude and longitude at 5-minute intervals from satellite element sets. Output is a printed listing of time since epoch, altitude, latitude and longitude at each point, and plot of satellite's ground trace.

27. EXTENSION OF CIRCUIT ANALYSIS TO 18 NODES VIA A SPARSE MATRIX APPROACH (9100B)
by Richard Q. Schmidt, Huntington, New York
This article describes a method for increasing the capacity of a given calculator regarding number of nodes of a network that are solvable. An increase in the 9100B CNAP capability from 8 to 12 nodes AC or to 18 nodes DC can be achieved with this method.

28. CODE PRACTICE OSCILLATOR (9820A)
by Ronald W. Shimanek, Kokomo, Indiana
This program allows the calculator to act as an international Morse code practice unit. Definable keys represent dots, dashes, and other required functions.

29. PIPELAYING OFFSHORE (9830A)
by Reddy Talusani, Houston, Texas
This article describes a trilateration application of the HP 9830A as an aid in keeping a pipelaying vessel on an accurate course.

30. 9820A DESIGNS CUSTOM SUNDIALS
by John Taylor, Boulder, Colorado
This article describes a series of four calculator programs which can be used to calculate and plot calibrations for line-shadow or point-shadow types of horizontal or vertical sundials.

31. USE YOUR PLOTTER AS A DIGITIZER (9810A)
by Dr. Rodes Trautman, Greenport, New York
This program allows the plotter to be used as a digitizer to make measurements of graphic material.

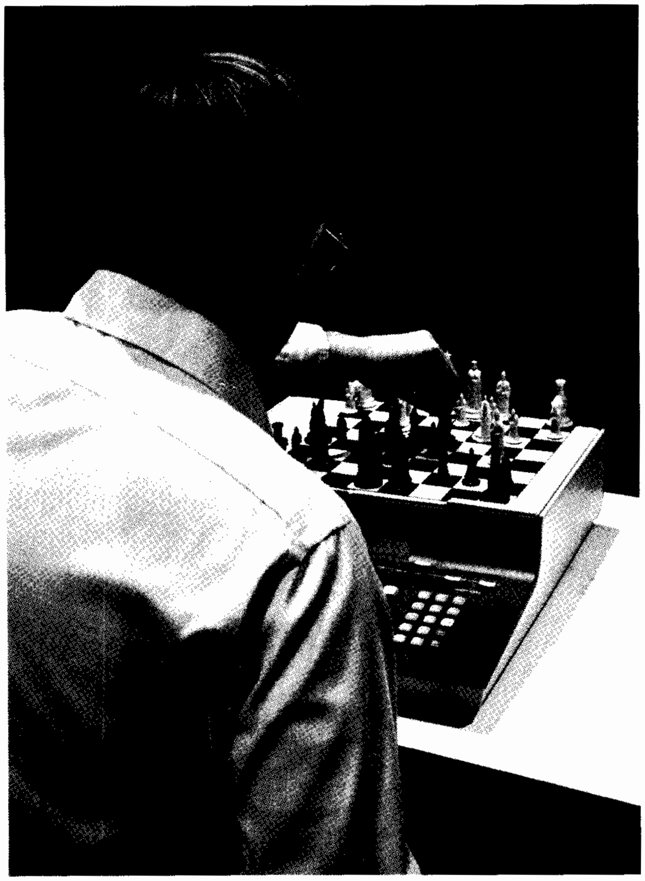
32. GEODETIC POSITION/STATE PLANE COORDINATES CONVERSIONS (9810A)
by Tom R. Wagemaker, St. Paul, Minnesota
This program converts data from geodetic position to state plane coordinates and vice versa. Also converts coordinates from one state plane coordinate zone to coordinates in another zone.

33. EXAMINATION AND SURVEY EVALUATION (9830A)
by Prof. Michael A. Wartell, Denver, Colorado
This series of programs constitute an examination evaluation and grading system for several types of examinations.

34. CHESS-PLAYING PROGRAM FOR THE 9810A
by Dr. Alan A. Wray, Fayetteville, Arkansas
This program allows the 9810A to act as a chess player in a game against a human opponent. Calculator chooses best alternative for each move. All regular moves are enabled, including queening pawns, castling, and capturing *en passant*.

A Chess-Playing Program for the 9810A

by Alan A. Wray, Ph.D.



INTRODUCTION

Schemes for mechanical chess players have appeared many times, including some long before the required technology existed, but the first detailed ideas for programming a digital computer to play chess were described by Claude E. Shannon in a 1949 talk (*Philosophy Magazine*, Vol. 41, pp. 356-375, March, 1950; and *Scientific American*, February, 1950). His basic ideas have been expanded and improved so that programs for large scale computers now exist which play at a competent novice's level (about a 1500 rating on the International Chess Federation scale). Just as large scale computation reached a level of sophistication in the 1950's sufficient for chess-playing programs to be implemented, recent advances in desk-top computation have produced programmable calculators of chess-playing capacity. Indeed, in addition to the intellectual interest in having an electronic opponent that is actually smaller than oneself, the desk-top advantages of high convenience and low cost are obviously important for game playing with a computing device.

Experience with the program described below indicates that the HP 9810A with 111 registers (option 001) and 2036 steps (option 003) has sufficient memory and speed for non-trivial play of the full game of chess, including castling, queening, and capture *en passant*, and, while the machine's play is of course significantly weaker than that of large-scale computers, it offers an interesting opponent for non-expert humans. For instance, some sort of trap or other positional maneuver is always necessary to gain material because the machine is very alert, never making obvious blunders nor ever failing to take advantage of its opponent's.

PROGRAM STRUCTURE

The chess-playing program is divided into two parts (though some subroutines are used by both): one searches for and provisionally makes each legal move allowed to the machine, and the other evaluates the position resulting from each provisional move. In the interest of size and speed, the program does not directly investigate possible responses by the opponent, further responses by the machine, etc., but it calculates the results of exchanges which begin with the capture of the threatened pieces on both sides. The program also considers the positional value of a given move based upon what squares are controlled by each player, with the four central squares considered most valuable and those farther from the center of progressively lower value.

The move which gives the most favorable position for the machine according to the evaluation procedure is then communicated to the opponent, and the machine's internal storage of the piece positions is updated. If no legal move is found, then the machine is either checkmated or stalemated, and the program halts with a status light. Otherwise, the machine accepts the opponent's move and continues with its next move.

It is usual in chess programs for large computers to include a substantial number of sequences containing the accepted best moves of the common openings. The HP 9810A program instead relies on the evaluation algorithm used throughout the game with some slight modifications that take effect for the early moves. For instance, moves by major pieces (rooks, queen, and king) are discriminated against, and attacks by the machine are given reduced values for the first few moves. In this way acceptable opening sequences are produced by the program without filling memory with explicit descriptions of openings.

USER INTERACTION

In addition to the calculator and chess program, the user needs magnetic cards containing the input data (originally recorded with a FORMAT XTO). This data contains the following:

1. For input to registers 1 to 64, the initial board position with squares numbered sequentially in rank-major order from the opponent's leftmost king-row square (KR1 if opponent is Black and QR1 if White) and with contents in the following numerical code: opponent's king = -6, opponent's queen = -5, opponent's rook = -4, opponent's bishop = -3, opponent's knight = -2, opponent's pawn = -1, empty square = 0, machine's pawn = 1, etc.
2. For input to register 95, a number in the range 0 to 1 that controls the aggressiveness of the machine's play; in the example game below this number was set to 0.1.
3. For input to registers 96 to 101, the values to be assigned to the opponent's pieces. All should be negative, and the magnitude indicates relative values among the types of pieces as well as the value of pieces compared to positional values (which are of order 1). In the game below these values were: opponent's king = -1200, queen = -900, rook = -500, bishop = knight = -300, pawn = -100.
4. For input to registers 103 to 108, the values of the machine's pieces. In the game below these were: pawn = 110, knight = bishop = 330, rook = 550, queen = 990, king = 1320 (kings are of course actually priceless, but the program requires a finite king value for its calculations).
5. For input to all other registers, 0.

With this data on cards, play is begun by loading the chess program and pressing END, CONTINUE. The program requests the above data by executing FMT XFR. After the user inserts the data cards, the game starts.

Moves are communicated between the machine and its opponent by using X and Y as coordinates on the chessboard. However, when the machine castles, the resulting display is according to standard chess notation. X and Y are zero; if Z is zero, castling is queen-side (0-0-0), otherwise it is king-side (0-0). The human opponent castles or captures *en passant* by a two-move sequence provided by the program.

The opponent chooses, by entering an appropriate number, what piece his pawn becomes when it reaches the eighth rank. However, the machine's pawns advanced to the eighth rank always become queens.

The opponent may suspend play by entering his final move for the session, then pressing SET FLAG, CONTINUE. The register contents can then be recorded with FORMAT XTO. Play can be resumed later by restoring the registers with FORMAT XFR and pressing GO TO 12, CONTINUE. The program thus allows a "sealed move" suspension.

EXAMPLE GAME

The following game was played with the 9810A as White and the author playing Black. Chess buffs will note a Giuoco Piano type opening, followed by inexpert play on both sides. The game is not presented as an example of great chess by man or machine, but as an example of the level of sophistication of modern programmable calculators.

1	P-K4	P-K4
2	N-KB3	N-QB3
3	B-B4	B-B4
4	N-B3	P-Q3

Normally preferred is 4 P-B3 for pawn strength in the center or 4 P-Q3 to free the queen bishop.

5	Q-K2	B-KN5
6	P-KR3	P-Q4

This gambit by Black is a daring attempt at control of the center, but it would be dangerous against a stronger opponent.

7	PxP	BxN
8	PxB?	N-Q5

To maintain defensive pawns, 8 QxB would be preferred, but the machine chooses to keep pressure on Black's king pawn and king file.

9	Q-K4	Q-B3
10	B-K2	Q-KN3
11	QxPch	N-K2
12	QxBP	NxP(B2)ch

Offense is absorbing both sides, and the result is a wide-open game with many threats.

13	K-B1	NxR
14	QxB	Q-N3
15	Q-B4	0-0
16	N-R4	Q-R4

White should be getting his queen bishop into the fray rather than vainly pursuing the Black queen.

17	P-Q6	QR-B1
18	PxN	RxQ

The pin on the queen was damaging, but White gets fair compensation.

19	BxR?	R-K1

Queening with 19 PxRch would be preferred, but the machine does not foresee the gain of a rook in exchange for its knight and advanced pawn.

20	N-B3	RxP
21	N-Q5	R-K4
22	N-B3	N-B7
23	P-B4	R-K8ch

White's failure to root out the knight leads to trouble.

24	K-N2	RxR
25	KxR	N-K8
26	B-Q5	Q-N3
27	P-B3??	Q-N3

The bishop pawn is better defended with 27 N-Q1, leaving the bishop's vital defense of White's KN2.

28	BxNP	Q-N7mate

White's fate was sealed, and this last capture is little compensation. At the end the machine is only one point behind in material, but its immobile queen bishop and some poor moves resulting from lack of foresight led to its demise.

CONCLUSION

The spectacular number-crunching ability of recent programmable calculators is certainly well documented, but, as the chess-playing 9810A has shown, a formidable logical capability has also been attained. Clearly this capability goes beyond game playing and finds applications in many fields.

Alan A. Wray obtained his B.S. in Physics at the University of Arkansas in 1969. He served in the U.S. Air Force in 1972-73, and was awarded a Ph.D. in Physics by the California Institute of Technology in 1973. He is currently seeking a research position in mathematical physics.

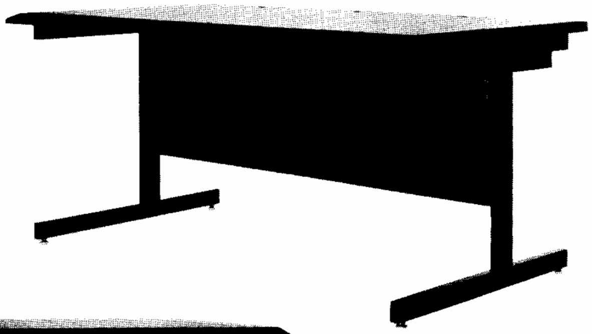
Office Furniture for Calculator Systems

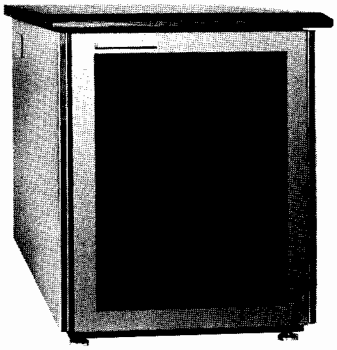
Systems Furniture Company of Gardena, California, now offers personalized, modular furniture for Hewlett-Packard computing calculator systems. This furniture combines the requirements for system packaging and versatility with an aesthetically pleasing exterior.

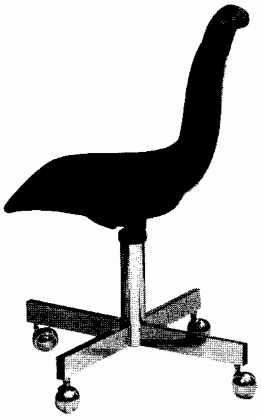
The custom furniture for HP computing calculator systems includes an operator station, mass memory console, and chair. The Systems Furniture Company has designed the furniture for HP computing calculators to be compatible with its standard product line. This combines maximum versatility with professional appearance in a complete computing calculator furniture system for the office.

Since all of these furniture products are modular, lending themselves to numerous configurations, a high degree of customization to individual needs is possible. Systems Furniture Company does not limit its efforts to merely conforming to individual needs with contemporary furniture designed for HP computing calculators, or to making this furniture modular, or to making it compatible with the SFC standard product line. Systems Furniture Company takes one more step to fill custom needs by either matching your paint chip or accepting your order for any one of three standard colors. The standard color choices are Garnet Rose, Harvest Gold, or Rocky Mountain Blue.

The basic building block of the HP computing calculator system furniture is the operator station. Its leather formica top with a solid walnut edge is accentuated by your color choice. The full-width modesty panel adds a finishing touch.







The operator station is designed to accommodate any Hewlett-Packard desktop programmable calculator, with enough room for any two additional peripherals. Three cable-access holes on the table top run to a fully enclosed cable trough attached to its bottom surface. Matching top-fitted plugs provide a neat table-top appearance. The cable trough is designed to connect with the cable-access opening in the mass memory console.

The mass memory console has a smoked plexiglass door, through which the mass memory status lights are visible. Total accessibility to all component parts of the 9880A/B system, a new level of quietness, and modular construction reflect the careful engineering.

The chair combines simplicity with comfort. Your choices of color and material are limited only by your taste.

Systems Furniture Company has combined contemporary styling, custom design for electronic equipment, and human engineering in a product line to meet a variety of individual system needs.

More information is available from:

Systems Furniture Company
2012 West 139th Street
Gardena, California 90249
Attention: Lee Kennedy

THE Crossroads

THE ART OF SCIENCE

By John Nairn, Ph. D.*

"The purpose of computing is insight, not numbers"
R. Hamming

At first glance the title of this article, The Art of Science, may appear to be composed of contradictory terms. In fact, a popular method of categorizing a given discipline is to use a spectrum whose endpoints are labeled art and science. Thus, cooking is considered to be mostly an art, whereas medicine is recognized as having started as an art and, through its history, having shifted its position on the spectrum toward the end labeled science. If in this context we take art to mean that subjective or speculative side of a discipline (insight or intuition, if you will), and science to mean the objective or verified body of knowledge associated with that discipline, then such a categorization is indeed useful. Unfortunately, the two disciplines whose names have been taken to label the ends of this spectrum, Art and Science, suffer a great injustice from its use, because by implication, their positions on the spectrum are taken to be the end points. The concept that each should contain no elements of the other causes difficulty for both. Only when the appropriate amount of science (objective theory) is blended with art (subjective creativity) can works of the caliber of DaVinci, Mondrian, and Cézanne result. Conversely, the great advances made by such men as Newton, Gauss, and Einstein would never have seen the light of day, if their great scientific abilities had not been matched by an equal ability for intuition and insight.

In many fields such as statistics, civil engineering, clinical laboratory operations, finance, accounting, and a host of others, the advent of electronic calculating devices has offered a means of quickly and easily reducing complex theoretical relationships to numerical results for a specific case of interest. And indeed, this is the use for which a large percent of the electronic calculators are employed. But they also offer a powerful research tool on the frontiers of these disciplines where that body of theoretical relationships is being analyzed and expanded. Like any tool, however, the calculator in research work can be used to benefit, or it can be abused. I suppose that the mathematical purist could view the advent of electronic calculating devices in theoretical research as a splended opportunity for total corruption of the discipline. In school I had two professors, heads of the physics and math departments respectively, who carried on a friendly rivalry. The physics professor would refer to the math department as the "rigor mortis" department (rigor to the point of beating the problem to death); and the math professor would refer to the physics department as the "about" department (getting about the right answer is close enough). In reality, they both understood the way in which empirical investigation and rigorous theory go hand in hand to advance theoretical knowledge. Calculators and computors offer an invaluable tool for making available to the researcher a wealth of empirical data relating to a problem under investigation. This data may then be used to gain insight to the correct theoretical approach to the problem. It is only when the copious supply of data is allowed to become an end in itself, and not a means to theoretical insight, that numerical results can become detrimental to theoretical investigations. The trick is for the researcher to know when to use the calculator and when to apply the art of science to generalize his results and obtain a theoretical relationship.

Shakespeare often used the technique of comic relief to give his audience a breather from the heavy emotional state induced by his tragedies. After my longwinded tirade, I suppose I should, in the manner of the great bard, offer some clarification by means of examples. And my favorite examples are those in the guise of mathematical recreations (commonly known as puzzles). Below are a number of puzzles whose solutions can be obtained from a purely analytical approach (if you are so inclined), but have been chosen to exemplify the use of a mixture of empirical and analytic techniques. In a later article I will give the answers to these puzzles, together with a discussion of what I consider to be a good mix between an analytical and numerical approach to each of them. I encourage you to try to solve them and will also publish in the *KEYBOARD* issue after next any interesting approaches to their solutions that I receive. I will also be discussing a related topic of the use of analysis vs. so-called "brute force" methods in writing programs. (By the way, be aware that the last problem contains an unsuspected twist).

1. In 1066, Harold II of England led an army to meet the Norman invaders at the Battle of Hastings. It is said that Harold was at the head of thirteen ranks of men, each rank of the same size and marching in square array. When they reached Hastings, Harold had to join his men in order to form one large square array for the battle (a tactic which, by the way, got him killed !). How many men were in Harold's army?
2. Seven sailors, stranded on a deserted island, decide to gather the only food on the island - coconuts. After they collect coconuts all day, they are too tired to divide them then, and decide to wait until morning. During the night, one sailor wakes up and decides to take his share now. He divides the pile into seven equal portions, finds one extra coconut which he throws to a nearby monkey, buries his share, puts the other six piles back into one large pile, and goes back to sleep. During the night, each of the other sailors in turn wakes, takes 1/7 of the pile he finds, throws the extra coconut to the monkey, and sneaks back to sleep (a very untrustworthy lot!). The next morning, the sailors wake up (each trying to look as innocent as possible) and divide the remaining pile seven ways, throwing the extra coconut to the monkey. How many coconuts were in the original pile?
3. A truck needs to cross an 800 mile stretch of desert. Unfortunately it can only carry enough gasoline to travel 500 miles. The driver (an out of work Ph.D. in mathematics) realizes he can still cross the desert by driving out into the desert, leaving part of his gas supply, and returning for more. Fortunately, there is no gasoline shortage at his initial departing station. By building up caches this way, how much gas is required to cross the desert? How many miles does he travel altogether? What is the widest desert he could cross using this method?
4. Six men are at a party. By the time they go home they are all quite drunk. As each one leaves he takes a coat from the coat rack, not knowing (much less caring) whose coat he takes. What is the probability that at least one of the six men will get his own coat? What is the probability if there are N men at the party?
5. What is the probability that three lines whose lengths are chosen at random in the range zero to one will form a triangle?
6. I play a game in which I throw a number of dice one at a time, and stop when any two of the dice show the same number. What is the most probable number of dice on the table at the end of the game? That is, on which throw is the game most likely to end?
7. What is the probability that if I randomly break a stick into three pieces, the pieces will form a triangle?

---

*Hewlett-Packard Company, Calculator Products Division, Loveland, Colorado

Airplane Stability Augmentation System Design on the HP 9830A

by Keith Mitchell

INTRODUCTION

The modern transport airplane is designed to fly at extremes of Mach and altitude at which artificial pilot aids become necessary to provide adequate response and stability. These aids normally consist of closed loop control systems sensing airplane motion variables and feeding back suitably processed electrical signals to the aerodynamic control surfaces to augment the pilot's manual inputs.

The typical computations required by control system design fall into the class requiring fairly heavy 'number crunching'. Consequently, providing a design office with adequate computing facilities having the rapid access and turnaround required for short design cycles has been a persistent problem. Batch operation with large scale machines has been too slow and time sharing with terminals capable of document quality plotting has been too expensive for routine use.

The arrival of desk top calculators such as the HP 9830 having high quality plotting and printing peripherals has revolutionized computing possibilities. Recently the entire high speed control system design for a prototype medium weight short takeoff and landing transport airplane (STOL) has been performed on a single dedicated HP 9830. The design efficiency achieved due to the combination of economy and accessibility has been higher than hitherto experienced with any other machine of any size.

Techniques routinely used for airplane control system design combine classical and modern theory. The airplane must be considered as a strongly cross-coupled multivariable dynamic system. This leads naturally to a state variable formulation of the equations of motion. Modern decoupling techniques are routinely used and optimal control occasionally is necessary. The single most useful technique is still the root locus applied to specific transfer functions obtained from the state model.

A single control system design may require consideration of 200 Mach-altitude combinations, each one leading to a different state model. Clearly, any hand manipulation of data is out of the question.

The article describes a series of linked, completely general purpose control system design programs, written without specialization to airplanes, which enable rapid and convenient design of multiloop control systems.

EQUIPMENT USED

1. HP 9830A Calculator with 7904 words of memory
2. HP 9865A Tape Cassette
3. HP 9866A Thermal Printer
4. HP 9862A Plotter
5. HP Matrix Operations ROM
6. HP Plotter Control ROM
7. HP String Variables ROM

PROGRAM ORGANIZATION

The airplane must be treated for control system design purposes as a multi-input, multi-output coupled system, and the obvious choice for a standard representation is the state variable form:

$$x = Ax + Bu \qquad y = Cx + Du$$

where u, x, and y are the system control, state and output vectors respectively; and A, B, C, and D are constant matrices of dimension suitable to ensure conformability. The 9830 matrix operations make handling vector equations very straightforward, and so whenever possible the programs are written to use algorithms involving matrix operations rather than operations on individual matrix elements.

The programs are almost all linked via a standard common block containing a state model in the form of the A, B, C, and D matrices, together with dimensions and identification. The augmentation of state models in general changes the dimensions by increasing the size of the state vector, so means must be available for loading state models from tape into common blocks of various standard sizes while preserving the positions of elements within the arrays.

Once a set of simultaneous linear differential equation coefficients has been entered at the keyboard it may be transformed to state variable form and stored on tape, and is then available to any other program without having to be reentered. Two cassette drives were found to be the minimum for convenient application of this principle if excessive cassette exchanging was to be avoided. A disc would be extremely useful for organizing a system of programs such as this.

All programs are written to operate automatically on state models of any size up to the maximum, so that, for example, transfer function and time history plotting programs may be used on the state models of basic airplanes and on those of airplanes augmented by the addition of filters and feedback loops without regard to the dimensions of the specific state model in use. To achieve this requires the central and perhaps most interesting program. This enables a form of state variable vector algebra to be performed by means of which a total system block diagram combining any number of state models, feedback matrices from either state or output vectors to any model input and similar topological features may be reduced systematically to the standard form of a single state model. This program employs the keys to facilitate the selection of the steps in the reduction process. The reduction is similar in practice to the systematic simplification of a scalar block diagram by loop elimination.

Whenever operator interaction with a program is not required it is written to loop automatically with tape input and output so that if necessary the machine may be run unattended and many cases computed per run. This batch mode is useful for keeping the machine occupied overnight.

PROGRAM DESCRIPTION

Transformation to State Variable Form

A set of simultaneous second order linear constant coefficient differential equations of the form

$$[Ls^2 + Ms + N]p = Sq$$

where s is the Laplace operator, and p and q are vectors of dependent and independent variables respectively and L, M, N, and S are constant matrices, is transformed to state variable form. The program uses ideas gleaned from reference 1 but does not follow the procedure detailed therein. The algorithm is constrained to choose a state vector from the elements of p and upper derivatives of p but is not restricted to systems with invertible L.

Obtaining State Model Transfer Functions

From a state model input from tape all transfer functions between the input and both state and output vectors are found simultaneously by matrix operations.

This program is only used for low order systems, since seldom are all possible transfer functions needed, and the method is not numerically sound beyond sixth or seventh order when the system A matrix has a large spread in the magnitude of its elements.

The characteristic polynomial is found by a matrix form of Böcher's formula and the numerators by Leverrier's algorithm. The method of Danilevsky is better than Böcher's formula but the latter is extremely convenient to code. Neither, however, is suitable for high order systems and other methods are used.

Numerator and denominator polynomials are rooted by a standard Newton-Bairstow algorithm.

Root Locus Plotting

The program for plotting a root locus from a set of open loop poles and zeros is coded for the keys. At the designer's discretion he may choose to enter, edit, print or plot poles or zeros. He may automatically plot a complex branch from its origin at a complex pole or a real axis departure point. Whenever a branch point is plotted, the coordinates, relative damping, natural frequency, and feedback gain associated with it are printed.

The branches are computed by iteration using a gradient technique to satisfy the angle criterion and an increment between successive points determined by the user. No polynomial rooting is performed and so the execution is quite fast. Systems with a total of 35 poles and zeros have been plotted satisfactorily.

Real axis departure points are determined by a linear search technique which finds local gain maxima and feedback gains for branches which lie along the real axis may be printed out if desired.

Time Histories from Transfer Functions

For a quick look at transient responses of augmented systems this program plots a single response to any transformable forcing function, by evaluating residues from poles and zeros and summing the individual modal responses. The main use is in conjunction with the root locus program to verify that a chosen pole configuration exhibits a satisfactory time response. The maximum total number of poles and zeros is similar to that of the root locus program and is determined by roundoff error rather than storage limitations.

Eigenvalues and Eigenvectors

As the dimension of the state approaches 9 or 10, more powerful methods are needed for accuracy. The best current method for finding eigenvalues of large systems is by the double sided QR transformation. The procedure followed is to scale the A matrix transform to Upper Hessenburg form and apply the QR transformation. With the 12 decimal digits of precision available on the 9830 this technique is good up to the limits of memory available (about 20th order without resorting to the storage of intermediate results on disc or tape).

Eigenvectors are computed by inverse iteration and are used primarily for finding state transition matrices for time solution of the state equations. They are incidentally useful in determining eigenvalue sensitivity to changes in A matrix elements. This is useful in preliminary airframe design.

Computed data is stored on tape in groups of three files per flight condition. The first file contains the state model as previously described. The second contains eigenvalues and characteristic polynomial and the third contains eigenvectors. Tapes in this format are standard input for the subsequent plotting programs.

Transfer Functions of High Order Systems

The current method of obtaining transfer function numerators of high order systems is given in reference 2. The method requires that the difference between the characteristic polynomials of the A matrix and a suitable modification of the A matrix be computed. All characteristic polynomials are calculated by multiplying together eigenvalues found by QR, and the difference polynomial is rooted by writing it as a matrix and again finding eigenvalues.

State Variable Algebra

By selection of the appropriate key the user may perform functions useful in reducing to a single state model a complex combination of state models, feedback and feedforward matrices.

Two state models may be added when the output of one conforms with the input of another; feedback may be added to an input from either output or state vectors; elements of an input vector may be included in the output vector of the same model to preserve them through the addition of feedback. State models of any dimensions up to the maximum may be loaded and stored and moved into various locations within main memory. Any array may be printed by entering the name, and the common block size of a stored model may be changed.

Time Histories from State Models

The time response of the state and output vectors to any control time function expressible as a sequence of time points may be plotted. The time scale is selected by the user, but all other scaling is completely automatic. Any or all elements of the input, state and output vectors may be plotted in any order, with axis labels and titles input from the keyboard or tape. Initial conditions may be placed on the state.

The program uses an exact recurrence equation of the form:

$$x(t + T) = \Phi x(t) + \Gamma u$$

$$y = Cx + Du$$

where Φ is the state transition matrix e^{AT} and Γ is the integral exponential

$$\int_0^T e^{A\tau} d\tau$$

Both of these matrices are computed rapidly from the eigenvalues and eigenvectors by similarity transformation rather than by summing a matrix exponential time series as is sometimes done.

Points for plotting are stacked on a plot tape if desired so that the program may be run overnight with many cases.

DESIGN EXAMPLE

This example illustrates the design of a two-loop augmentation system with a second order filter for the lateral-directional axes of a transport airplane.

For the airplane model under consideration:

$$A = \begin{bmatrix} -.1147 & -5.74 \times 10^{-4} & .0462 & -.998 \\ -4.826 & -1.775 & 1.173 & .457 \\ 0 & 1 & 0 & 0 \\ 1.17 & -.1948 & -1.456 & -.0852 \end{bmatrix}$$

$$B = \begin{bmatrix} .0402 & .00287 \\ -.0856 & 1.792 \\ 0 & 0 \\ -2.303 & .257 \end{bmatrix}$$

$$x = \begin{bmatrix} \text{BETA (Sideslip)} \\ \text{P (Roll Rate)} \\ \text{PHI (Bank Angle)} \\ \text{R (Yaw Rate)} \end{bmatrix}$$

$$u = \begin{bmatrix} \text{DR (Rudder Deflection)} \\ \text{DA (Aileron Deflection)} \end{bmatrix}$$

From this the transfer function between rudder (DR) and yaw rate (R) is calculated as: $\frac{R(S)}{DR(S)}$

$$\frac{-2.303\,(S + 1.85)\,(S^2 + 2(.019)\,(.349)\,S + (.349)^2)}{(S + 2.46)\,(S + 1.99)\,(S^2 + 2(-.0048)\,(1.28)\,S + (1.28)^2)}$$

The denominator has a second order factor with negative damping indicating instability. This mode is present in most airplanes and is generally referred to as the "Dutch Roll". The principal task of the yaw damper is to stabilize this mode. The time response of the airplane to a 10 degree aileron deflection of 2 seconds duration is shown in Figure 1. The very low damping of the system is apparent on the sideslip response which oscillates with almost constant amplitude.

A bandpass filter of the form: $\frac{3.6S}{(S + 3.6)\,(S + .36)}$ is combined with the R/DR transfer function and a root locus plotted in Figure 2. The filter is included to reduce feedback to zero in steady state turns and avoid interference with higher frequency flexible body modes. A gain of .41 gives adequate damping.

The filter state model is combined with that of the airplane and the transfer function between aileron and roll-rate calculated. The feedback gain for the second loop closure is determined by a second root locus plot and is chosen at a value giving a sufficiently rapid roll rate response.

A block diagram of the final system is shown in Figure 3 and the corresponding time history in Figure 4. The oscillation has been eliminated and the roll rate (P) now reaches a steady value in less than 1 second.

The flying qualities of the airplane have been improved from near-uncontrollable to the point at which they are conventional and quite acceptable.

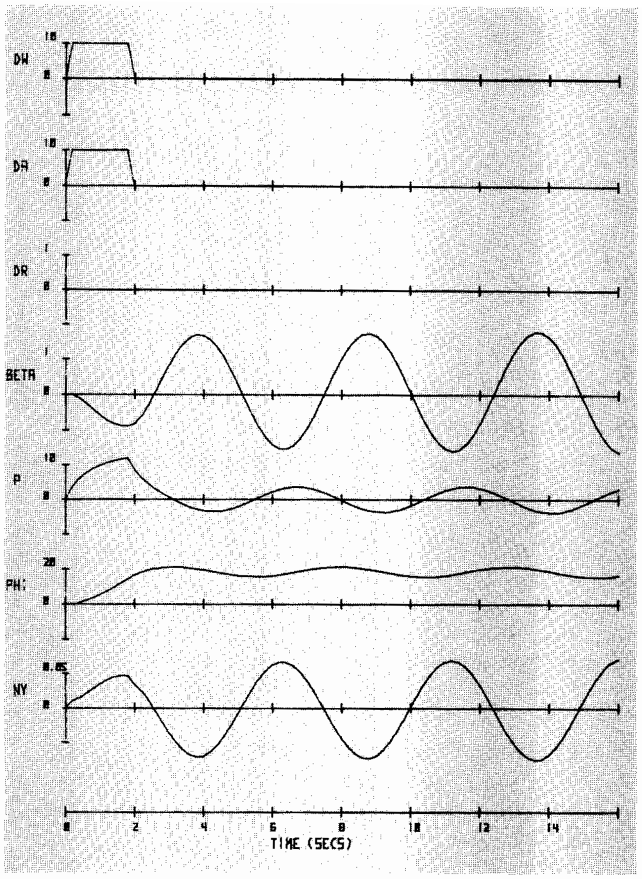


Figure 1. Wheel Pulse, Basic Airplane

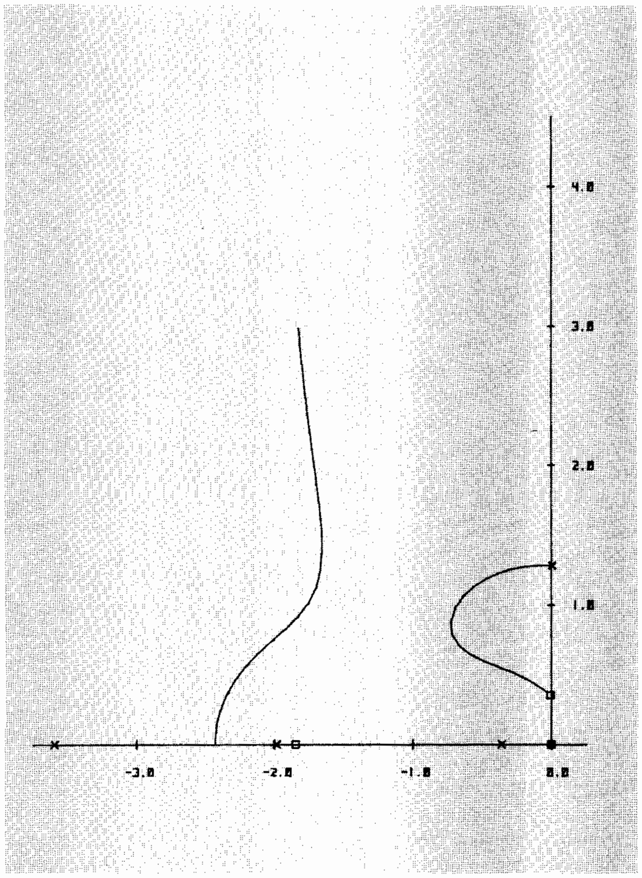


Figure 2. Root Locus Closing Loop Yaw Rate into Rudder Including Washout and Lag Filters

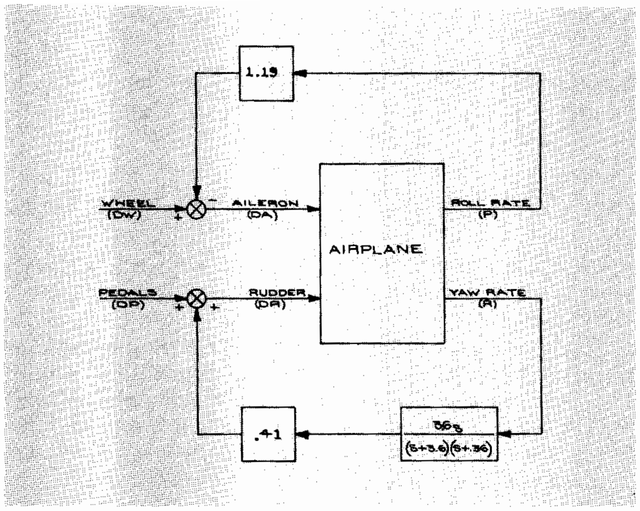


Figure 3. Block Diagram of Fully Augumented System

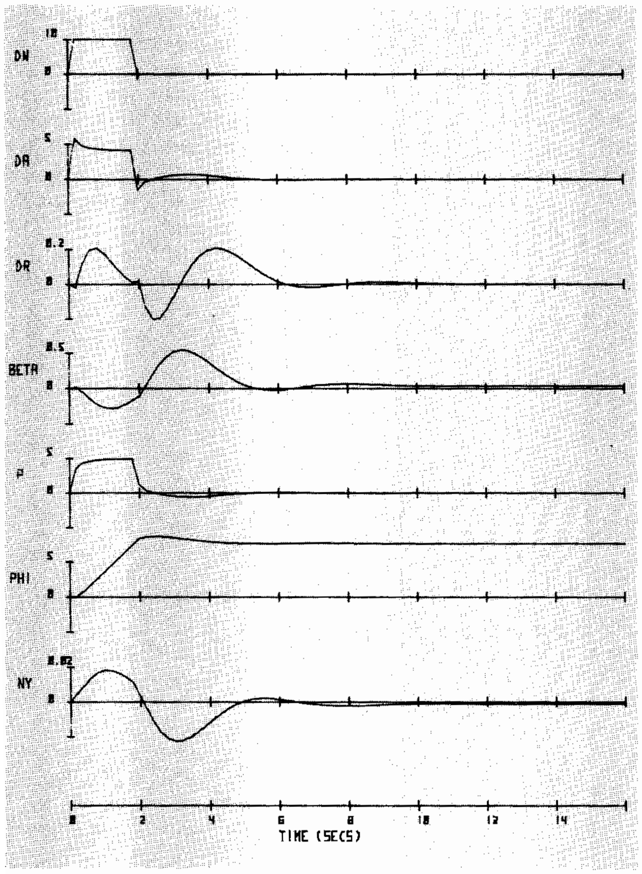


Figure 4. Wheel Pulse, Yaw and Roll Dampers

REFERENCES

1. Wolovich, W.A.; "The Determination of State-Space Representations for Linear Multivariable Systems", presented at the 2nd IFAC Symposium on Multivariable Technical Control Systems, Duesseldorf, Germany, October, 1971.
2. Patel, R.V.; "On the Computation of Numerators of Transfer Functions of Linear Systems", IEEE Trans. Automat. Contr., Vol. AC-18, p. 400-401, August, 1973.

CURRICULUM VITAE

Mr. Mitchell was born and educated in Great Britain receiving the degree of B.Sc. in Electrical Engineering from Manchester University in 1956 and Postgraduate Diploma in Electronics from Southampton University in 1959.

Prior to coming to the U.S.A. in 1961 he was employed as a controls engineer by the Sperry Gyroscope Company in both Brentford and Bracknell and Smith's Aviation Division, Cheltenham.

From 1962 to the present the author has been with The Boeing Co. engaged in the design of control systems for several aerospace vehicles and in the adaptation of modern control techniques including decoupling, state identification and optimization to routine design.

Mr. Mitchell's main interests are music and photography. He plays saxophone, clarinet, vibes and piano and specializes in the photography of performing musicians.

programming tips

BASIC INTEGER TO ALGEBRAIC (9820A/9821A)

D.L. Schacher of Tel-Instrument Electronics Corp., Carlstadt, New Jersey, sent in this program tip.

Recently a problem was encountered in translating a BASIC program into algebraic for the 9820A. BASIC says that the function INT "gives the largest integer ≤ the expression", while the MATH ROM INT "eliminates fractional part of value; does not affect sign or integer value." For positive numbers, there is no difference, but for negative numbers, INT (BASIC)(-1.5) = -2, while INT (Algebraic)(-1.5) = -1. Thus, when rewriting a BASIC program for the 9820A or 9821A where negative values may occur, instead of INT(X), write INT[X-(0> X)], to maintain the same meaning.

HIGH SPEED FILE IDENTIFICATION (9820A/9821A)

Our thanks go to Koichi Tanaka of Yokogawa-Hewlett-Packard, Ltd., in Tokyo, Japan for the following time-saving tip.

IDENTIFY FILE (IDF) is often used to determine the construction of the cassette files. Although Program A (below) will accomplish this, it works very slowly when the files are large.

My program, B, works rapidly because of using high speed search capability. It takes about half as much time to identify files using B as using A.

Program A

```
0:
FXD 0;♦FDF 0⊢
1:
♦IDF A,B,C,X⊢
2:
PRT A,C,B,X;SPC
⊢
3:
GTO 1⊢
4:
END ⊢
Σ23521
R1442
```

Program B

```
0:
FXD 0;♦FDF 0⊢
1:
♦IDF A,B,C,X⊢
2:
IF X>5;♦BKS ;♦
FDF A+2;PRT A,C,
B,X;SPC ;♦BKS ;
GTO 1⊢
3:
PRT A,C,B,X;SPC
;GTO 1⊢
4:
END ⊢
Σ20131
R1437
```

EXAMPLE

```
 17   ← File no.
  2   ← File type
 85   ← Current size
100   ← Absolute size

 18
  0
  0
100

 19
 20
 99
109
```

"DO" LOOPS (9820A/9821A)

D.L. Schacher of Tel-Instrument Electronics Corp., Carlstadt, New Jersey submitted another interesting program tip for the 9820A or 9821A.

Fig. 1 below shows the normal manner of programming two nested loops on a 9820A or 9821A which, while efficient, does not always indicate the broad picture of what is being accomplished. In FORTRAN, for example, the "DO" loop indicates what is to be done, without getting involved in the question of how to do it.

It is possible to write "DO" loops on a 9820A or 9821A, as shown in Figs. 2 and 3, using a "DO" key on the UDF ROM. This "DO" key is somewhat better than the FORTRAN "DO", as it can be nested up to 11 deep, and can have positive or negative initial, final, and incremental values.

The "DO" key (Fig. 2) uses 5 passing parameters: P1 is the DO loop number (from 1 to 11), to allow keeping track of which loop ends where; P2 is the variable, P3 is the initial value, P4 is the final value, and P5 is the increment or decrement. The line before each CLL DO must have CFG N: SFG 12, where N is the loop number; while the end of the loop is signified by IF FLG N = 0; GTO (CLL DO line). Fig. 3 gives a sample program using the "DO" key, while Fig. 4 is the resulting printout.

```
0:
4→A⊢
1:
FXD 2;PRT 3↑A→Z⊢
2:
1→B⊢
3:
FXD 0;PRT 2↑B→Z⊢
4:
B+1→B;IF B≤3;
GTO -1⊢
5:
A-1→A;IF 1≤A;
GTO -4⊢
6:
"END";SPC 8;END
⊢
```

Fig. 1 Conventional loop nesting

```
0:
"DO ";P2+P5→P2;
IF FLG 12;P3→P2;
CFG 12⊢
1:
IF (P5/ABS P5)(P
2+P5-P4)≤0;GTO +
2⊢
2:
SFG P1⊢
3:
END ⊢
```

Fig. 2 Defined "DO" key

```
0:
CFG 1;SFG 12⊢
1:
CLL DO 1,A,4,1,-
1⊢
2:
FXD 2;PRT 3↑A→Z⊢
3:
CFG 2;SFG 12⊢
4:
CLL DO 2,B,1,3,1
⊢
5:
FXD 0;PRT 2↑B→Z⊢
6:
IF FLG 2=0;GTO -
2⊢
7:
IF FLG 1=0;GTO -
6⊢
8:
SPC 8;END ⊢
```

Fig. 3 Mainline program for "DO" loops

```
81.00
    2
    4
    8
27.00
    2
    4
    8
 9.00
    2
    4
    8
 3.00
    2
    4
    8
```

Fig. 4 Output of "DO" routine

"TABLE" IDENTIFICATION (9820A/9821A)

We are indebted to D.L. Schacher of Tel-Instrument Electronics Corp., Carlstadt, New Jersey, for the following useful tip.

The one-line program below determines for which trigonometric units (degrees, radians, or grads) the Math ROM trigonometric functions are set in the 9820A or 9821A calculator. This is used in the beginning of a subroutine, before setting the table needed for the subroutine. The trigonometric functions can then be reset to the original TABLE 1, 2, or 3 before leaving the subroutine.

```
0:
(SIN 180=0)+2(0>
SIN 180)+3(SIN 1
80>0)→X⊢
```

9830A INTERRUPT SYSTEM

David A. Ripley of General Dynamics, Albuquerque, New Mexico, submitted this programming tip which you may find useful.

If you have written programs containing nested or lengthy "DO loops" you probably know that there is no interrupt system for the 9830A as such. For example, you cannot alter your program flow any way short of stopping the program, changing a statement, and continuing from there. This forces the programmer to do one of two things: either display each result or wait for termination, assuming the loop is not "hung up".

The following sample may be useful to you as it allows a physical action on your part to cause branching. It uses the STAT (status) command found on pp 3-4 and A-3 of the Extended I/O ROM manual to allow for such interrupts, i.e., by opening or closing the tape transport door. Only one command is needed to perform this function. See statement 120 of the example. Note that this statement assumes there will be a tape inserted into the transport and ready (not on clear leader). If the tape is on clear leader or if the transport is empty, a different value will be returned with "STAT".

Execution time can be drastically reduced for multiple calculations by this method as compared to displaying or printing each result.

Comparative Times

DISP Statement	STAT Statement
100 items approx. 20 sec.	100 items approx. 2 sec.

The STAT statement can be used at any time to allow branching by the simple IF statement. There are many other applications for this statement, such as printing totals, etc., without terminating execution.

EXAMPLE

```
10 REM* SET UP FOR LOOP CALCULATIONS *
20 FOR I=1 TO 1000
30 REM
40 REM** CALCULATION SECTION **
50 Y=INT(RND(I)*1000)
60 REM
70 REM** CHECK STATUS OF TAPE TRANSPORT **
80 REM** IF STATUS = 11 DOOR IS OPEN **
90 REM***** PRINT CALCULATIONS IF STATUS NOT EQUAL TO 11 (DOOR SHUT) **
100 REM**** STAT CHECK ASSUMING TAPE IN TRANSPORT AND READY **
110 REM
120 IF STAT10=11 THEN 140
130 PRINT "Y ";Y;" I ";I
140 NEXT I
150 END
```

MONETARY FORMATTING (9830A)

The following programming tip was suggested by Bob McCoy of the Hewlett-Packard sales office in Atlanta, Georgia.

When the output of your computation on the HP 9830A is in monetary units such as dollars, it is convenient to have the dollar sign preceding the figure, as well as having the digits grouped in threes separated by commas, especially when six or more digits appear to the left of the decimal. The routine shown below will insert the dollar sign and commas as required, according to the number of digits in the output. The input must be a minimum of .XX, and the routine requires the Extended I/O ROM (or appropriate DEXP command on the Mass Memory), and the String Variables ROM.

EXAMPLE

```
$ 0.50
$-0.50
$ 0.02
$-0.02
$ 123.00
$-123.00
$ 123,456.00
$-123,456.00
$ 123,456,789.00
$-123,456,789.00
```

```
10 DIM A$[20],B$[20]
20 FIXED 2
30 INPUT A
40 OUTPUT (A$,*)A
50 FOR I=1 TO 20
60 B$[I,I]=""
70 NEXT I
80 X=LEN(A$)-2
90 A=17
100 B$[18,20]=A$[X-2,X]
110 X=X-3
120 IF X>4 THEN 150
130 B$[A-X+1,A]=A$[1,A-(A-X)]
140 GOTO 220
150 B$[A-2,A]=A$[X-2,X]
160 X=X-3
170 A=A-3
180 IF X <= 1 THEN 220
190 B$[A,A]=","
200 A=A-1
210 GOTO 120
220 A$[1]="$"
230 A$[2]=B$[A-X+1]
240 IF POS(A$,"-")=0 THEN 260
250 A$[1,2]="$-"
260 PRINT A$[1,21-(A-X)]
270 GOTO 30
280 END
```

HP-65 KEY NOTE

Many owners of HP programmable desktop calculators in the U.S.A. also own the new HP-65 programmable pocket calculator. If you are in this category, you may already know about the new newsletter which is available at no charge as part of the HP-65 package.

The newsletter, called HP-65 KEY NOTE, is distributed to all owners who send in their "free subscription" card. This newsletter, which will be published four times a year, has goals of giving HP-65 owners programming and operating tips, answers to questions, and information on new programs, developments, and accessories.

The first issue of HP-65 KEY NOTE, Summer 1974, has been mailed. If you own one of the new HP-65's in the U.S.A. but did not have the opportunity to send in a subscription form, write to the editor at the following address:

Editor
HP-65 KEY NOTE
19310 Pruneridge Drive
Cupertino, California 95014

The editor will welcome any contributions of HP-65 programs, programming techniques, or other ideas you may have for publication in HP-65 KEY NOTE.