HEWLETT-PACKARD

# KEYBOARD

# options

VOL. 3 NO. 4

## NEW HP *KEYBOARD* FIELD EDITORS

*KEYBOARD* Vol. 3 No. 3 introduced two new field editors for the western U.S.A. and European areas. Now we welcome to the *KEYBOARD* staff five additional field editors for the following areas:

Canada . . . . . . . . . . . Brian Wiley
Japan . . . . . . . . . . . . Akira Saito
South Africa . . . . . .Denis Du Buisson
Australia . . . . . . . . . . . Bill Thomas
Latin America . . . . Daniel Mazar-Barnett

Readers living in these areas should submit their articles, programs, and programming tips to the appropriate field editor. Suggestions for types of material you would like to see in *KEYBOARD* or comments regarding applications you feel should be emphasized will be welcomed by any member of the editorial staff.

## TABLE OF CONTENTS




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

Editor: A.B.SPERRY — Art Director: L.E.BRADEN

Field Editors:

AUSTRALASIA: Bill Thomas, Hewlett-Packard Australia Pty. Ltd.
61 Alexander Street
Crows Nest, 2065, New South Wales

CANADA: Brian Wiley, Hewlett-Packard Canada Ltd.
275 Hymus Blvd.
Pointe Claire, Quebec

EUROPE: Eberhard Beck, Hewlett-Packard GmbH
Herrenberger Strasse 110
703 Boblingen, West Germany

JAPAN: Akira Saito, Yokogawa-Hewlett-Packard Ltd.
59-1, Yoyogi 1-chrome
Shibuya-ku, Tokyo 151

LATIN AMERICA: Daniel Mazar Barnett, Hewlett-Packard Intercontinental
3200 Hillview Avenue
Palo Alto, California 94304

SOUTH AFRICA: Dennis du Buisson, Hewlett-Packard South Africa (Pty.) Ltd.
30 de Beer Street
Braamfontein

WESTERN U.S.A.: Robert C. Reade, Hewlett-Packard Co.
3939 Lankershim Boulevard
North Hollywood, California 91604

KEYBOARD
VOLUME 3 NUMBER 4

## TO HEWLETT-PACKARD *KEYBOARD* READERS

This is your personal copy of *KEYBOARD*, which is published to support all HP calculator owners and users. There is no charge for a subscription, the only qualification being that we are informed of your calculator ownership by the reply card shipped with each unit.

If you own both 9100A/B calculators and the new Model 10, you may find the programming tip in this issue on converting 9100A/B programs to the Model 10 to be especially useful.

*KEYBOARD* will welcome any programs or programming tips for HP calculators which you feel are of general interest.

## EXTENDED MEMORY STAT PAC

The EXTENDED MEMORY STAT-PAC, Part No. 09100-70875, is now available. This PAC contains programs for the solution of many complex statistical problems including:

SECTION I ANALYSIS OF VARIANCE

- I-1 Two-Way AOV with Replicates ($r \leq 10$)
- I-2 Two-Way AOV with Tukey's Test ($r \leq 10$)
- I-3 Two-Way Design - Unbalanced (A and B Levels $\leq 10$)
- I-4 Nested AOV - Balanced (Levels $\leq 4$)
- I-5 Nested AOV - Unbalanced (Levels $\leq 3$)
- I-6 Three Way AOV - Factorial Split Plot (Treatment Combinations $\leq 175$)
- I-7 Latin Square ($k \leq 16$)

SECTION II ANALYSIS OF COVARIANCE

- II-1 One-Way ANOCOV - Equal Number of Observations (Treatments $\leq 47$)
- II-2 One-Way ANOCOV - Unequal Number of Observations (Treatments $\leq 103$)

SECTION III REGRESSION ANALYSIS

- III-1 Multiple Linear Regression (Order $\leq 17$)
- III-2 Polynomial Regression (Order $\leq 14$)
- III-3 Stepwise Regression (Order $\leq 5$)

These programs were chosen and written by the Statistics Laboratory of Colorado State University. They result from the extensive experience this laboratory has gained in the statistical data reduction field. All solutions are complete and include full printing of input data and computed results. All programs have been written for use with either HP 9100A or HP 9100B Calculators, but do assume availability of a 9120A Printer.

To order a copy of the 9101A Extended Memory Stat-Pac, or an Extended Memory, check the appropriate box on the KEYBOARD reply card.

# options for the model 10

The Hewlett-Packard Model 9810A is an extremely versatile, self-contained desk-top calculator. It allows the owner to select the memory capability which is best suited to his general needs, as well as best matched to the types of specialized computations needed in his particular work. Aside from the versatility added by the peripheral accessory units such as the Model 9862A Plotter and the Model 9860A Marked Card Reader, which are external to the calculator, the greatest aid to the Model 10's flexibility consists of the built-in options and the optional plug-in function blocks.

## Built-in Options

The Model 10 is available at the time of purchase with a variety of memory options, as well as a choice of having or not having a built-in thermal printer. The basic machine comes with 500 program steps and 51 data storage registers. It can be supplied with a total of 1012 steps or 2036 steps (Option 002 or 003), and 111 total data storage registers (Option 001). The thermal printer (Option 004) can also be supplied factory-installed with the calculator.

By considering the complexity of problems to be solved, the relative need for storage, and permanent record requirements, the buyer may select the best combination of internal options for his particular needs at the time of purchase. If the immediate requirements are not great but they may increase later, any of the internal options can be installed at any time. Although some economies are achieved by including options at the time of purchase, their initial absence may be offset by the convenience and time savings of having the calculator available at the earliest date, notwithstanding a limited budget.

## Optional Function Blocks

Past experience with the Model 9100 series calculators has indicated that most calculator owners would prefer to have some degree of flexibility in selecting the left-hand block of keys. Although the mathematics-oriented keys on the Model 9100 were significant for most engineering and scientific applications, individuals working primarily in business or statistics would find keys defined for their disciplines more useful.

The plug-in function blocks on the Model 9810A Calculator have been designed for maximum flexibility in optimizing its capabilities for many diversified applications.

Three slots are provided in the top of the Model 10 to receive the plug-in function blocks, as shown in Figure 1. Slot #1 on the left holds a function block which defines each key in the left-hand keyboard section. Three function blocks are currently available for the left-hand slot. Each block has a matching template which snaps into place over the left keyboard section, labeling the key functions as they are defined by the particular plug-in block. The keys protrude through the overlay, and are used in the same way as the other keys on the calculator.

The *MATHEMATICS BLOCK*, Model 11210A, adds 26 mathematics functions and further programming features to the basic calculator. Some of these are logarithms and exponential functions, trigonometric functions in degrees, radians or grads, coordinate transformation, vector arithmetic, manipulation of complex numbers, and automatic rounding to any power, automatic scaling for X-Y plotting, and a user-definable function.

The *STATISTICS BLOCK*, Model 11214A adds single-keystroke statistical computations to the basic calculator. These include $\chi^2$, $t$, linear, multi-linear, and parabolic regression; random number generation; accumulation of sums, sums of products, and sums of squares, for up to five variables; maximum/minimum search; and an exclusive 'correct' key to remove erroneous data.

The *DEFINABLE FUNCTIONS BLOCK,* Model 11213A, enables up to nine user-definable functions to be stored at one time. Each function can be executed by a single keystroke. Stored functions can be easily deleted or changed at will. A 'protect' feature saves functions from being accidentally erased or changed when other programs are entered into the calculator. Three special editing keys, FIND, DELETE, and INSERT, greatly simplify editing programs (both the definable functions and the programs in the rest of the calculator's memory).

These "single" blocks may be used in Slot #1 only, and may not be combined with other blocks or used in other slots.

Additional "single" blocks currently available include the following:

The *PLOTTER BLOCK*, Model 11215A, provides automatic plotter scaling, axis drawing, alpha-numeric plotting, plotting of the X-register contents as displayed, and setting of the display to fixed or floating point.

The *TYPEWRITER BLOCK*, Model 11212A, provides complete keyboard control of the typewriter, including tab clear and set, upper-lower case, and ribbon shift.

The *PERIPHERAL CONTROL BLOCK*, Model 11264A, provides the control for the digitizer, as well as ASCII bit parallel, character serial input and output capability.

The *PRINTER ALPHA BLOCK*, Model 11211A, enables the printer to print alphabetic characters and messages, such as user instructions, and to insert mnemonics into program listings.

These four blocks may be used in the middle slot, #2, or the right-hand slot, #3 with the exception of the *Printer Alpha* block which may be used in slot #3 only. The functions of two of these may be combined in one block as follows:

*Plotter* and *Printer Alpha*

The combination block offers the same capability as the blocks used singly. However, it allows the calculator system operator to utilize several blocks at the same time. For example, in order to operate the Printer Alpha, Plotter, and Typewriter functions in a program, he would require the combination block listed above, in addition to the appropriate single block. The combination block also provides for future expansion to include other peripherals as they become available.

OPTIONAL FUNCTION BLOCKS

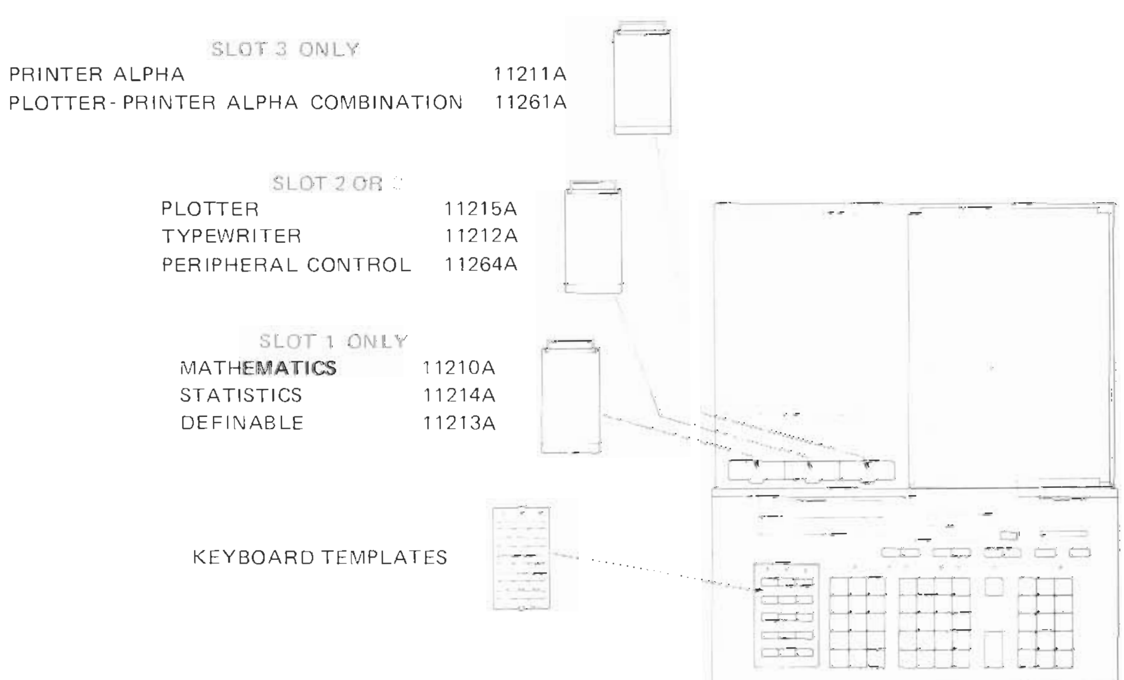



Figure 1

# BUSINESS ROUTINES FOR THE MODEL 10

Many short calculating routines are commonly needed in solving problems such as present worth of a single payment or a series of payments, the size of payments necessary to repay a given present amount, the number of days between two given dates, and others.

A series of routines of this type is now being designed for the HP Model 10 Calculator. The routines will be useful in a number of business areas, such as engineering economy studies, new equipment studies, banking, investment analysis, and financial management.

The business routines can be used to define keys on the Model 11213A Definable Function plug-in block for the Model 10, or they can be used alone, or combined with other steps to form programs for solving specific problems.

Twenty or more routines will become available soon in the form of a packet. Ordering information will be announced when the packet is completed. A partial listing and description of the contents is given below, as well as an example of one of the routines.

### Future Compound Amount of a Single Payment

This routine calculates the amount of principal and interest accumulated after n periods if amount P is invested at interest rate i.

### Future Compound Amount of a Series of Payments

Calculates the amount F to which a series of n payments will grow invested at interest rate i, or the total amount of prospective saving, F, n periods hence in order to justify spending amount A per period for n periods if money is worth i percent.

### Periodic Amount Necessary to Total F in Future

This routine calculates the amount that must be invested each period for n periods at interest i to amount to F, or it calculates what periodic amount is justified to spend for n periods in order to avoid spending amount F, n periods from the present with interest at i percent.

### Periodic Amount Necessary to Repay Present Amount P

Calculates the periodic amount A necessary to pay off present debt P in n periods with interest at i percent, or the periodic amount A which can be withdrawn for n periods from a present deposit P at interest i percent, or the periodic savings A for n periods which must be anticipated to justify an additional present expenditure of P if money is worth i percent.

### Present Worth of Single Payment

This routine calculates the present value P of a single future payment F to be made n periods from the present, if money is worth i percent; or the amount P which must be invested at present to amount to F in n periods at interest rate i percent.

### Present Worth of Series of Payments

Calculates the present value P of a series of n periodic payments, each payment being A, if interest is i percent; or the investment P justified in order to make a prospective saving of A per period for n periods if money is worth i percent; or the amount P owed on a loan with payments of A per period and n more periods to be paid.

### Days Difference

This routine calculates the number of days between any two dates, leap year included.

### Julian Routine

Calculates the Julian date for any day of the year.

### Interest Factor Routine

Calculates interest factors – daily, monthly, quarterly, etc.

### Months Difference

Calculates the number of months between any two dates. Part of a month is counted as one full month. Especially handy when calculating installment loan rebates.

### Simple Interest Payments

Calculates monthly, quarterly, semi-annual or annual payments when interest is calculated on the declining balance.

### Lister Routine

This routine lists columns of numbers as they are being added or subtracted, and maintains two sets of totals. Since it makes the calculator act like an adding machine, the entire routine is included here.

**EQUIPMENT NEEDED**

- [X] PRINTER
- [ ] 9860A MARKED CARD READER
- [ ] 9861A TYPEWRITER
- [ ] 9862A PLOTTER

**TOTAL REGISTERS**

- [X] 51
- [ ] 111

**TOTAL PROGRAM STEPS**

- [X] 500
- [ ] 1012
- [ ] 2036

**ROM'S**

Printer Alpha

| STEP | USER INSTRUCTION | DISPLAY x | y | z |
|---|---|---|---|---|
| 1. | Set FIX 2 (or number of decimal places desired) | | | |
| 2. | Enter a NUMBER TO ADD OR SUBTRACT*. Press CONTINUE. *To subtract press CHG SIGN key after entering number and before pressing CONTINUE. Repeat this step until totals are wanted. | | | |
| 3. | To print totals, press SET FLAG, CONTINUE when calculator is stopped. | | | |
| 4. | To clear grand total press END, CONTINUE. | | | |

## EXAMPLES

```
         15.00
        132.00
         46.00
        -17.00
         81.00
       -105.00
SUB
        152.00
GRAND
        152.00

         32.00
         46.00
        -81.00
         44.00
        116.00
SUB
        157.00
GRAND
        309.00
```

## PROGRAM STEPS

| STEP | KEY | CODE | STEP | KEY | CODE |
|---|---|---|---|---|---|
| 0000 | CNT | 47 | 0027 | FMT | 42 |
| 0001 | CNT | 47 | 0028 | XFR | 67 |
| 0002 | CLR | 20 | 0029 | 0 | 00 |
| 0003 | XTO | 23 | 0030 | PNT | 45 |
| 0004 | 0 | 00 | 0031 | FMT | 42 |
| 0005 | XTO | 23 | 0032 | FMT | 42 |
| 0006 | 1 | 01 | 0033 | G | 15 |
| 0007 | CNT | 47 | 0034 | a | 13 |
| 0008 | 1 | 01 | 0035 | A | 62 |
| 0009 | STP | 41 | 0036 | N | 73 |
| 0010 | IFG | 43 | 0037 | D | 63 |
| 0011 | 2 | 02 | 0038 | FMT | 42 |
| 0012 | 2 | 02 | 0039 | XTO | 23 |
| 0013 | CNT | 47 | 0040 | + | 33 |
| 0014 | CNT | 47 | 0041 | 1 | 01 |
| 0015 | PNT | 45 | 0042 | XFR | 67 |
| 0016 | + | 33 | 0043 | 1 | 01 |
| 0017 | XTO | 23 | 0044 | PNT | 45 |
| 0018 | + | 33 | 0045 | PNT | 45 |
| 0019 | 0 | 00 | 0046 | CLX | 37 |
| 0020 | GTO | 44 | 0047 | XTO | 23 |
| 0021 | 8 | 10 | 0048 | 0 | 00 |
| 0022 | FMT | 42 | 0049 | GTO | 44 |
| 0023 | FMT | 42 | 0050 | 8 | 10 |
| 0024 | YTO | 40 | 0051 | END | 46 |
| 0025 | 1/X | 17 | | | |
| 0026 | B | 66 | | | |

GREY AREAS DENOTE PROGRAM STEPS UTILIZING THE **PRINTER ALPHA ROM.** PLACE CONTINUE'S IN THESE AREAS IF THIS ROM IS NOT AVAILABLE.

# USE OF THE HEWLETT-PACKARD 9100A/B CALCULATOR FOR AUTOMATIC RECORDING AND PROCESSING OF DATA FROM PSYCHOLOGICAL EXPERIMENTS AND TESTS

by Erich A. Pfeiffer, Ph.D.

## INTRODUCTION

The administering of psychological tests and the recording and processing of data from such tests is rather time-consuming. While certain tests will always require the attendance of a trained experimenter, others lend themselves well to automation of the procedure. In the past, the automation of such tests has been achieved in one of two ways: with devices especially designed for this purpose, like the TAPAC (Totally Automated Psychological Assessment Console) (1) or by the use of a general purpose digital computer (2). Both methods are very expensive.

A system for the automatic administration of any psychological test has to include the following basic parts:

1. A stimulus presentation section which presents a suitable stimulus to the subject.
2. A response manipulandum which accepts the response of the subject.
3. A control and processing device which controls the stimulus presentation and scores and records the subject response.

These three parts correspond to a certain extent to the three basic building blocks of a programmable electronic calculator: the display, the keyboard and the arithmetic unit (3).

The purpose of this paper is to show that it is indeed possible to use the Hewlett-Packard 9100A/B Calculator for the automation of certain types of psychological experiments and tests. The limitations of the calculator for this application will be discussed and, as an example, a program for the automatic measurement of reaction times will be described.

## PROBLEM

The type of psychological experiments that seem to be most suitable for adaptation to the HP 9100A/B calculators are simple response and vigilance tasks and number recall or digit span tests. In a response task, a stimulus is presented to the subject, who responds in one or more possible ways, depending on the type of test. Each stimulus presentation requires a response by the subject. In a vigilance task, on the other hand, stimuli with varying properties are presented and the subject is required to respond only to those stimuli -- usually a very small fraction of the total -- that have certain characteristics. In number recall tests, a number or sequence of numbers is presented to the subject who is asked to repeat the numbers, sometimes in a reversed order, immediately or at a certain time after the presentation.

The response variables to be recorded are the latency of the response and whether the response was correct. The automation of such tasks by means of a programmable calculator requires the automatic presentation of a suitable stimulus, and the recording and processing of the subject response.

## SOLUTION

The 9100A/B has no alphabetic capabilities but presents a display of three lines of numbers. The stimuli, therefore, must be presented in numerical form. The calculator also has a so-called error light which indicates the execution of an illegal operation (such as the logarithm of zero). During the execution of a program, this light will not interfere with subsequent program steps, but it will remain on until a key of the calculator is depressed following a STOP instruction, or the STOP key is pressed. The error light, therefore, is very well suited to be used as a simple stimulus.

The response of the subject can be entered into the calculator by using suitable keys of the keyboard. During the execution of a program, however, all keys are locked out with the exception of the STOP key. In order to interact with the calculator while a program is executed (as is necessary for measuring the response latency) this key must be pressed and released before any other key can be used.

The latency of a response can be measured in two different ways. A simple counting loop can be executed at a rate of about 800 per second, allowing the measurement of

Modified from Pfeiffer, Erich, The Use of Programmable Desk Calculators for the Recording and Processing of Data from Psychological Experiments, *Computers and Biomedical Research*, Vol. 3 No. 6, Dec. 1970 with permission of Academic Press, New York, N. Y.

the latency in increments of about 1.2 milliseconds. During the execution of this loop no register display can occur. This method is, therefore, limited to tests which use the error light as a stimulus or tests in which the subject responds after the stimulus presentation has been terminated.

Another possibility of measuring the latency is to use the PAUSE instruction. This instruction causes the content of the registers to be displayed for a time interval of about 150 milliseconds. In conjunction with a counting loop, this instruction can be used to measure the response latencies, but only in increments of 150 milliseconds. The response task, therefore, has to be designed in such a way that the latencies are of an appropriate order of magnitude for these increments. When the subject responds, the calculator must branch out of the counting loop. With the HP 9100A calculator, this can be achieved by including an IF FLAG instruction in the counting loop while the subject responds by pressing the keys STOP, SET FLAG, and CONTINUE in sequence. Because of a different internal organization in the 9100B calculator, another method for initiating the branching has to be used for this model (which, however, also works on the 9100A). In this case, the counting loop does not contain a conditional branching instruction, but the subject responds by pressing the keys STOP, END, and CONTINUE. Pressing the END key causes the calculator to branch from the counting loop to storage location 00, from whence the program continues.

For experiments not requiring latency measurements, the stimulus can be presented by means of the PAUSE and STOP instructions. The subject responds by pressing one or more number keys (or SET FLAG) followed by CONTINUE. Conditional branch instructions in the program can then be used to score the subject response. Using the calculator in this fashion requires that the subject press a number of keys in a certain sequence. This process can be aided by providing a simple cardboard mask over the keyboard which covers the unused keys. A more elegant way would be to have the keys pressed by small solenoids (e.g., Dormeyer P-8 series) activated by a simple sequencer. Unless the HP 9101A Extended Memory is available, the storage capability of the calculator usually will require that the collected data be processed after each trial and only the results stored.

Within the boundaries set by the limitations of the calculator regarding the display of stimuli and the measurement of latencies, the experiments and data analysis that can be performed are limited only by the size of the memory of the calculator and the imagination of the programmer. Very powerful programs for immediate analysis of the collected data are possible, especially if a HP 9120A Printer and/or a HP 9125A Plotter are available.

The program library for the HP 9100A/B calculators contains numerous programs that can be used as subroutines in the development of more extensive testing and analysis programs. A random number program, 09100-70816, for instance, can be used to randomize the stimuli or to provide a random time delay in the stimulus presentation while numerous statistical programs can be used as subroutines for the immediate analysis of the results of the tests.

The ability to easily modify program steps permits the experimenter to manipulate experimental variables such as the number of trials, frequency of stimuli and the distribution of stimuli with time.

## EXAMPLE

In order to demonstrate the feasibility of using the HP 9100A/B calculator for this purpose, a program for the measurement and analysis of reaction time has been written. This program uses the error light as a stimulus. The reaction time is measured over a prescribed number of successive trials by presenting the stimulus with a random time delay. The subject responds by pressing STOP and initiates the next trial by pressing END and CONTINUE. After the prescribed number of trials have been executed, the following variables are calculated and displayed sequentially in groups of 3 or printed out in 6 lines on the HP 9120A Printer:

Maximum reaction time in milliseconds.
Minimum reaction time in milliseconds.
Geometric mean of reaction time in milliseconds.
Number of trials.
Mean of the natural logarithms of the reaction time.
Standard deviation of the natural logarithms of the reaction time.

The experimenter can manipulate the number of trials and the mean time of stimulus delay by simple modifications of the appropriate program instructions.

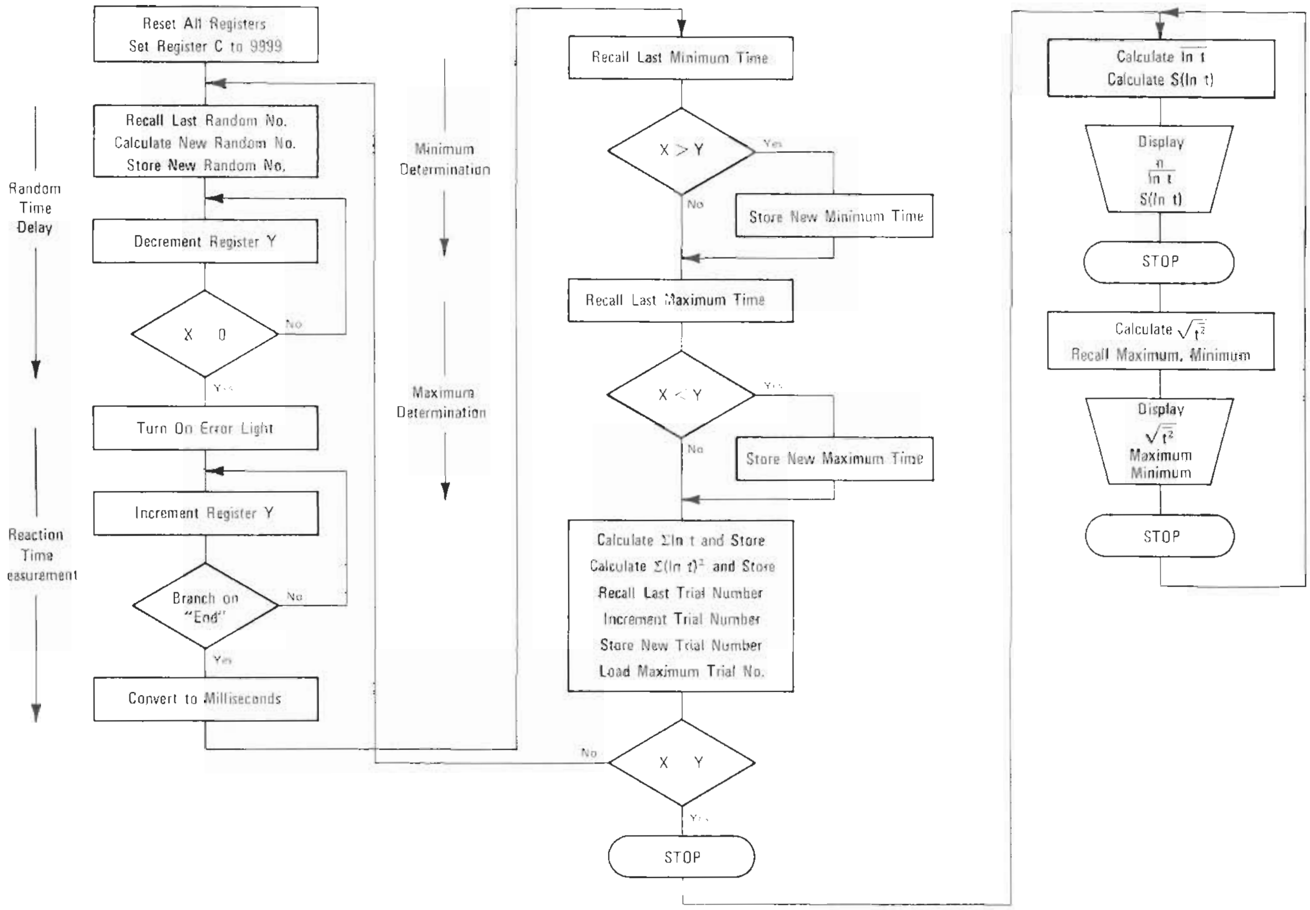
A flow diagram for this program is shown in Figure 1. Although almost all the storage capability of the Model 9100A is required for this particular program, more complex analysis programs can be executed by separating the recording and analysis parts of the program. This requires that the second half of the program be loaded after the prescribed number of trials have been performed.

## REFERENCES

1. Edwards, A.E. (1967). Prophylactic Psychopharmacology and Age Correlated Intellectual Changes. *The Gerontologist* 7:2, 20-23.

2. Crow, T.J. and Kelman, G.R. (1968). Use of general purpose digital computer for psychological testing. *J. Psychol.* 201, 4-5.

3. Osborne, T.E. (1968). Hardware Design of the Model 9100A Calculator. *Hewlett-Packard J.* 20:1, 10-13.




Erich A. Pfeiffer was born and got his training in Germany where he received the M.S. in 1956 and the degree of a Doctor of Engineering from the University of Stuttgart in 1962. He heads the Biomedical Instrumentation Section in the Western Research Support Center of the Veterans Administration in Sepulveda, California, where he acts as an instrumentation consultant for research projects in the life and behavioral sciences.

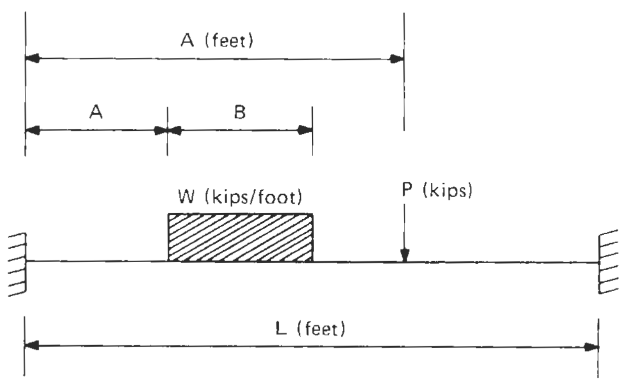
# FRAME ANALYSIS*

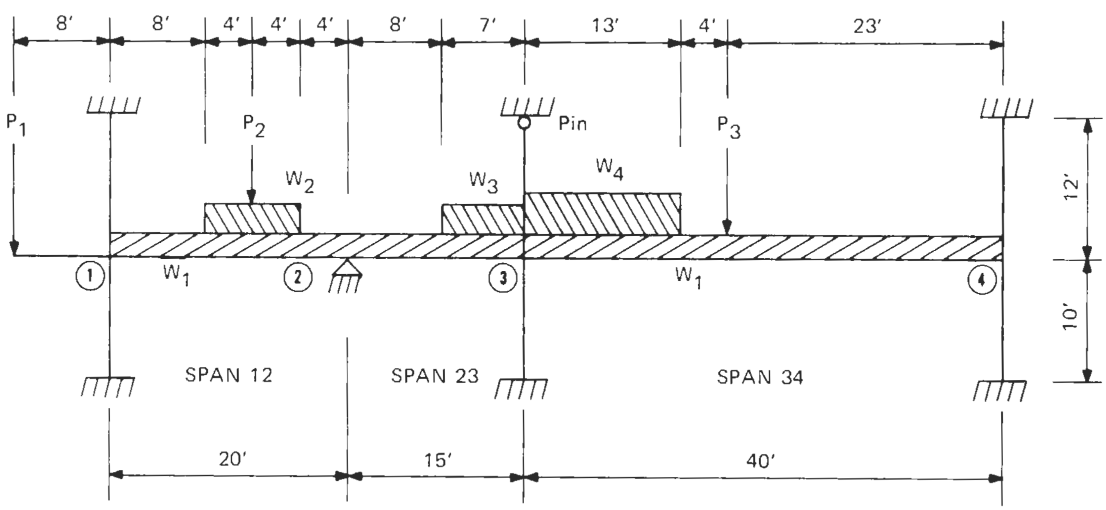
by
**Ben T. Felder, P.E.**

This program will analyze one floor of a structure with a maximum of 6 bays plus 2 cantilevers. Columns are assumed fixed above and below. Any number of concentrated and uniform or partial uniform loads may be applied. Five cycles of moment distribution are employed.

NOMENCLATURE

A: Distance from left support to load (feet)
B: Length of uniform load (feet)
P: Concentrated load (Kips)
L: Length of beam or column (feet)
W: Uniform load (Kips per foot)
EI: Column or beam stiffness factor where
E = Modulus of elasticity, I = Moment of inertia.




EXAMPLE

$P_1$ = 10 Kips
$P_2$ = 10 Kips
$P_3$ = 20 Kips

$W_1$ = 2 Kips/foot
$W_2$ = 2 Kips/foot
$W_3$ = 2 Kips/foot
$W_4$ = 4 Kips/foot

**For more information on this program, contact: Wm. F. Freeman Associates*
*309 N. Hamilton Street*
*High Point, North Carolina 27261*

## USER INFORMATION

1. For columns pinned at ends, modify the EI factor by 3/4.

2. Where the beam is continuous over a support but unrestrained, such as at joint 2 in the example, enter the column lengths as 0, EI as 1.

3. Enter all data as positive (+).

4. Final distributed column moments are positive (+) when the fiber to the right of the joint is in tension. Final distributed beam moments are positive (+) when the bottom fiber is in tension.

5. No correction for sidesway is made.

6. Any number of loads can be applied. The fixed end moments are calculated and stored as each load is entered. After the last load is entered, the accumulated fixed end moments are distributed by five cycles of moment distribution and the results printed.

7. Joints are numbered 1, 2, 3, etc.

8. Spans are numbered 12 (joint 1 to 2), 23 (joint 2 to 3), etc.

Ben T. Felder is Chief Structural Engineer and an associate of Wm. F. Freeman Associates, High Point, North Carolina. He received a BS in civil engineering from the Citadel, Charleston, S.C., in 1962. Mr. Felder served as an assistant bridge engineer with the California Division of Highways for three years before joining Freeman Associates. He is a registered P.E. in California, North Carolina and South Carolina, He is an associate member of ASCE and a member of NSPE.

### ADDITIONAL STRUCTURES PROGRAMS AVAILABLE

A complete library of structures programs is now available for the Model 9100B Calculator. This Structures Pac bears HP Part No. 09100–74200. Programs contained in it are described in the Calculator Program Catalog, a copy of which is free on request.

| EQUIPMENT NEEDED | TOTAL REGISTERS | TOTAL PROGRAM STEPS | ROM'S |
|---|---|---|---|
| [X] PRINTER | | | |
| [ ] 9860A MARKED CARD READER | [ ] 51 | [ ] 500 | 1 Math |
| [ ] 9861A TYPEWRITER | [X] 111 | [ ] 1012 | 2 |
| [ ] 9862A PLOTTER | | [X] 2036 | 3 Printer Alpha |

| STEP | USER INSTRUCTION | DISPLAY |
|---|---|---|
| 1. | PRESS: FIX ( ), 3. | |
| 2. | PRESS: END, FMT, GTO. Insert magnetic cards in numerical order. | |
| 3. | Enter the following data in the indicated order. PRESS: CONT after each sequence of data entry. | |

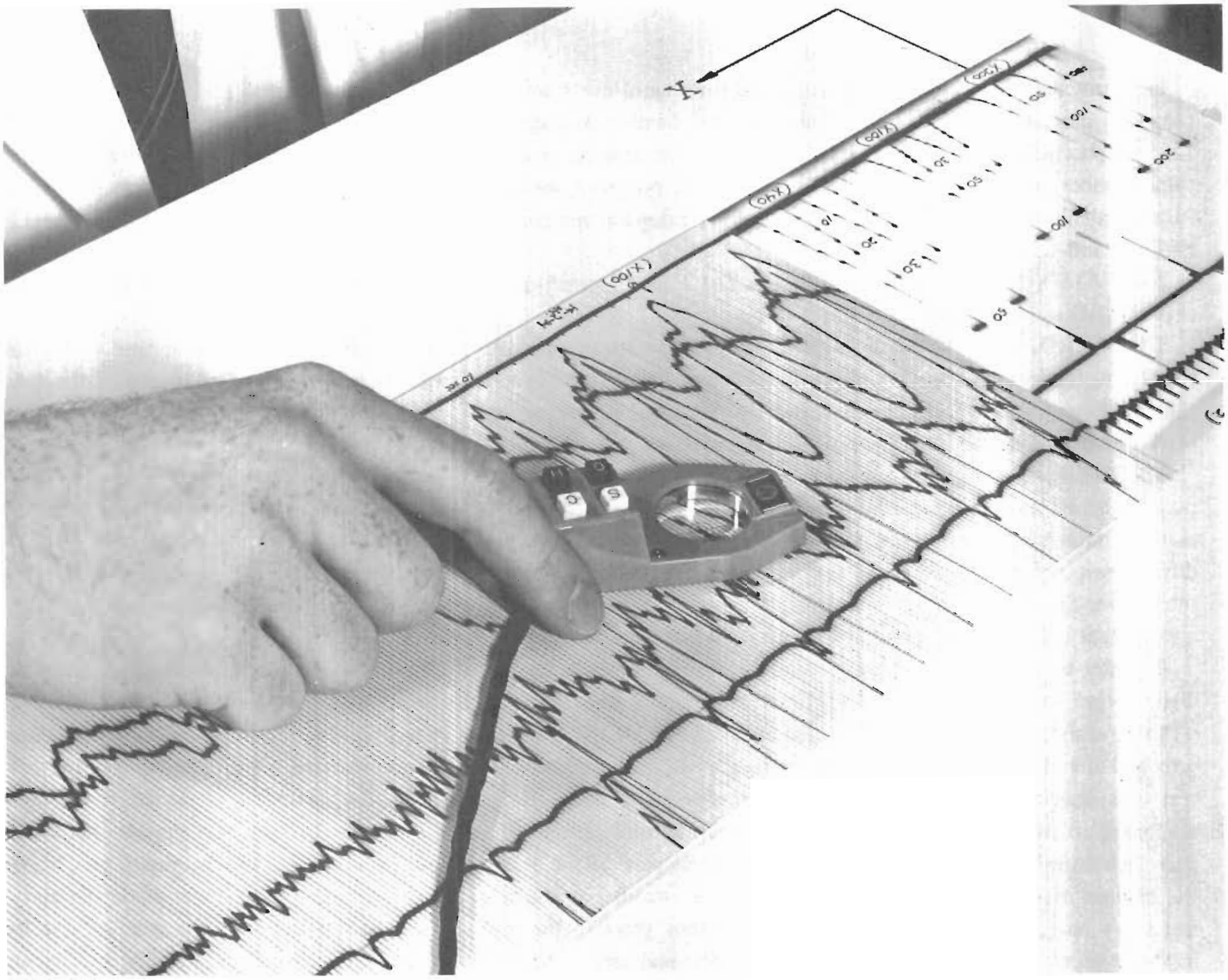
## EXAMPLE

| STEP | USER INSTRUCTION | DISPLAY |
|---|---|---|
| 1. | Enter the number of framed joints. PRESS: CONT | 4 |
| 2. | Enter the cantilever length at the left end. PRESS: CONT | L=8 (feet) |
| 3. | Enter the beam EI and length at spans indicated by the display. PRESS: CONT after each. | L=20 (feet) EI=1 Span Number<br>(Etc. for each beam.) |
| 4. | Enter the cantilever length at the right end. PRESS: CONT | 0 |
| 5. | Enter the upper and then lower column EI and length at the joints indicated by the display. PRESS: CONT after each. Enter 0 in X and 1 in Y if no column, or for unrestrained support. | (a) L=12 EI=1<br>(b) L=10 EI=1<br>(c) L=0 EI=1<br>(d) L=0 EI=1<br>(Etc. for each joint.) |
| 6. | Enter the concentrated load at the left cantilever. PRESS: CONT | P=10 (Kips) |
| 7. | Enter the concentrated loads and location in each span indicated and PRESS CONT after each entry. To go to next beam enter 0 in X and PRESS CONT. | A=12 P=10<br>0 0<br>(Next Span) |
| 8. | Enter the concentrated load at the right cantilever. PRESS: CONT | 0 |
| 9. | Enter the uniform loads and location in each span indicated and PRESS CONT after each entry. To go to next beam, PRESS CONT. | B=8 A=8 W=2 (kips/ft)<br>(CONT)<br>B=20 A=0 W=2<br>(CONT, CONT – to next span) |
| 10. | After all uniform loads have been entered, the program will calculate the distributed moments through five cycles of moment distribution, print the results and cycle back to Step 6. See the following example printout. | |

## EXAMPLE

```
ENTER NO.
OF FRAMED JOINTS
          4.000*

ENTER CANTILEVER
LENGTH AT JT 1
          8.000*

ENTER BEAM
EI AND L AT
INDICATED SPANS
         12.000
          1.000
         20.000

         23.000
          1.000
         15.000

         34.000
          1.000
         40.000

ENTER CANTILEVER
LENGTH AT END
          0.000*

ENTER COLUMN EI
AND L TOP
AND BOTT OF
JOINTS INDICATED
          1.000
          1.000
         12.000

          1.000
         10.000

          2.000
          1.000
          0.000

          1.000
          0.000

          3.000
          0.750
         12.000

          1.000
         10.000
          4.000
          1.000
         12.000

          1.000
         10.000

ENTER CONC LOAD
AT CANTILEVER
         10.000*

ENTER CONC LOADS
AND LOCATION AT
INDICATED SPANS
         12.000
         10.000
         12.000

         23.000
         34.000
         20.000
         17.000

ENTER CONC LOAD
AT CANTILEVER
          0.000*

ENTER UNIF LOADS
AND LOCATION AT
INDICATED SPANS
         12.000
          2.000
          0.000
         20.000

          2.000
          8.000
          8.000

         23.000
          2.000
          0.000
         15.000

          2.000
          8.000
          7.000

         34.000
          2.000
          0.000
         40.000
          4.000
          0.000
         13.000

BEAM MOMENTS
TO LT AND RT OF
INDICATED JOINTS
          1.000
         80.000
        137.963

          2.000
         69.826
         69.826

          3.000
        169.035
        555.147

          4.000
        382.650
          0.000

COLUMN MOMENTS
TOP AND BOTT OF
JOINTS INDICATED
          1.000
        -26.347
        -31.616

          2.000
          0.000
          0.000

          3.000
        148.505
       -237.608

          4.000
       -173.932
        208.718

PRESS END CONT
FOR NEW PROB
CONT FOR ADD LDS
```

# DATA REDUCTION USING THE 9107A DIGITIZER



The newest peripheral in the 9100 series, the digitizer is used in a wide variety of applications. Some of these are waveform analysis from oscilloscope trace photographs, acceleration and velocity determination from displacement plots, and cardiac and pulmonary data analysis to determine volume and flow rates. The 9107A will provide fast, accurate solutions to your problems involving measurement of area, line length, or distance between points.

The 9107A is the only digitizer with a direct calculator interface. It may be incorporated into an existing system merely by plugging into the rear of the 9100. X, Y coordinates may be entered into the calculator as single samples for selective analysis, or the continuous mode may be used to automatically enter up to 100 points per second. The reference origin may be set anywhere on the digitizing surface. An origin shift feature allows entry of data up to ±99 inches in both X and Y without resetting the reference origin. Therefore, long strip charts or maps which are larger than the 17 X 17 platen may be analyzed conveniently.

Analog data in virtually any graphical form may be placed upon the digitizing surface, or platen for analysis. The free-moving hand-held cursor provides the operator with a mechanically unencumbered curve follower. Data is entered with an accuracy of 0.01 inches, which is well within the accuracy of most analog recordings. To improve the relative accuracy of a small recording, the original curve can be digitized, smoothed, and replotted several times up-size on the 9125 Plotter. The replotted curve can then be placed on the platen for analysis.

# FAST FOURIER TRANSFORM ROUTINES

by

M. L. Boyd, J. K. Beard, Ph. D., J. W. Wauson

PART NO. 09100-70419
9100B ONLY

## I. Introduction and General Description

Programs using various fast Fourier transform algorithms have found widespread application in recent years. The three programs presented here provide a complete and flexible package for computing Fourier transforms of an input array of 128 complex points. With minor modifications to the existing programs, arrays of fewer points can be used, as long as the total number of points is a power of two. These programs were designed to be run on a system consisting of the HP 9100B calculator, 9125 plotter, and the 9101A extended memory. The three basic routines given here are the FFTSO, FFTRO, and REORDER programs.

The FFTSO program does a same order fast Fourier transform. This algorithm is the fundamental forward (negative exponential) transform.

The FFTRO, or reverse order, algorithm does a backward (positive exponential) transform. The output of the forward transform, however, is an array in the Fourier domain which is in scrambled order. This is a consequence of the nature of the fast Fourier transform algorithm and the necessity of overstoring the input array. Thus, in order to obtain the result of the transform in the Fourier domain, it is necessary to reorder the output of the FFTSO routine.

The REORDER program also contains a conjugating subroutine so that the output of the FFTSO forward transform may be back-transformed using the same SO algorithm. A reordering routine without the conjugating subroutine may be easily obtained by simply replacing the subroutine branching statements with continue statements. The FFTRO routine differs from the SO algorithm in that its input array must be in a scrambled form. Thus the reordering routine must be run before the FFTRO program in order to produce the correct input, and after the FFTSO program in order to produce a sequential output. These points are more fully discussed in the User Instructions Section.

In order to make the best use of available storage, both of the transform routines have been written to operate on six digit integer arguments. This is fairly common practice even on full scale computers and has little effect on the accuracy of the transform. In these programs the repeatability of the transforms is well within the limits of the plotter accuracy provided that the input arrays are properly scaled. This limitation on the arguments makes it possible to store both the real and imaginary parts of a complex number in a single storage register. Two integers of six or fewer digits are combined to form a floating point number with the units' place of the exponent used to separate the two parts. The real part of a complex number is given by the integral part of the floating point number and the imaginary part is given by the decimal fraction. The sign of the real part is encoded as the sign of the floating point number while the sign of the imaginary part is encoded as an 8 or a 9 in the tens' place of the exponent of the floating point number. An 8 will occur in the tens' place of the exponent only when the real part of the complex number is zero. The encoding and decoding is accomplished by separate subroutines within the FFTRO and the FFTSO programs. Examples of complex numbers and their coded equivalents are given in Table I.

It is suggested that in writing auxiliary routines to store functions, perform operations in the Fourier domain, and plot the transforms, the user employ the encoding and decoding subroutines as found in the FFTSO and FFTRO programs. In addition it should be noted that the complex points of the data array are stored in registers 1 through 128, inclusive, of the extended memory.

## II. Basic Algorithm

The number of points in the data array, N, is a power of two,

$$N = 2^M \tag{1}$$

where M = 7, N = 128. The index of a data point, i, can therefore be expressed in binary form,

$$i = i_0 + 2^1 i_1 + 2^2 i_2 + \ldots + 2^{M-1} i_{M-1} = \sum_{P=0}^{M-1} 2^P i_P \ , \tag{2}$$

---

**Editor's Note:**

The three Fast Fourier routines described in this article can be ordered through the local Hewlett-Packard Sales Office.

HEWLETT-PACKARD

# KEYBOARD

# cumulative index

## volumes 1, 2 and 3

## ARTICLES



## PROGRAMS



## PROGRAMMING TIPS

## TEACHER'S CORNER



## AUTHOR INDEX

NOTES: F = Article; P = Program; PT = Programming Tip; TC = Teacher's Corner

## IT'S THE LITTLE THINGS THAT COUNT

Have you thought about the convenience of the Model 10 Calculator provided by the ability to tailor it to different types of applications in seconds? This is accomplished easily, without tools, by merely changing plug-in blocks. From solving statistics problems most conveniently using the Stat block, you can convert your Model 10 to a specialized math-oriented calculator by plugging the Math block in place of Stat, and snapping the Math cover plate over the left-hand keys. The cost of plug-in blocks is not high, and they customize your unit to do a large number of different applications most economically.

The optional thermal printer fits inside the Model 10 so that the only evidence of its presence is the output tape with printed instructions, input data and labeled answers. There is no external cable or outside attachment to the calculator, leaving all of the four external connector receptacles available to connect other peripherals.

Loading paper is a pleasure, too -- a secretary will agree it's much simpler than changing any typewriter ribbon. Simply drop the roll in the receptacle, and press the PAPER key until the end of the tape appears in the printer window. Paper is economical, and the emitted sound while printing is at a pleasantly low level.

The status light and other indicator lights perform useful functions to simplify Model 10 operation. When this light comes on during program execution, it shows that an error has occurred. Either the dynamic range of the calculator has been exceeded, such as obtaining 'infinity' by dividing by zero, or a procedural error has occurred. Mathematical errors do not stop the program execution, but procedural errors, such as designating a non-existent address or incomplete subroutine branch address will stop it, making it easy to locate the programming error.

Other lights show readiness of the Model 10 to accept loading of magnetic cards; condition of the KEYLOG key; whether the machine is in floating or fixed point; whether the Model 10 is in the PROGRAM or RUN mode, and whether the Math block is set for operating in grads, radians, or degrees.

Many more convenient features of the Model 10, such as the BACK STEP key for easy editing, save the user time and money. Remember, it's the little things that count!

where

$$0 \leq i \leq N-1, \; 0 \leq i_p \leq 1, \; 0 \leq P \leq M-1,$$

and the index of the transformed output, K, will be given in the form

$$K = K_{M-1} + 2^1 K_{M-2} + 2^2 K_{M-3} + \cdots + 2^{M-1} K_o = \sum_{P=O}^{M-1} 2^P K_{M-P-1} \tag{3}$$

where

$$0 \leq K \leq N-1, \qquad 0 \leq K_p \leq 1, \qquad 0 \leq P \leq M-1$$

The continuous Fourier transform,

$$A(W) = \int_{-\infty}^{\infty} F(t)\, \mathrm{EXP}\,(-jwt)\, dt \tag{4}$$

is approximated to the accuracy of the sampled input by the discrete sum

$$N \cdot A(K) = \sum_{i=O}^{N-1} X(i)\, \mathrm{EXP}\,(-j\, 2\pi\, iK/N) \tag{5}$$

where $j = \sqrt{-1}$. The sum on i is expressed in terms of the bits $i_p$, and the trigonometric identity

$$\mathrm{EXP}\,(-j\, 2\pi\, iK/N) = \mathrm{EXP}\,(-j\, \frac{2\pi}{2^M} \sum_{p=O}^{M-1} 2^P i_p \sum_{q=O}^{M-1} 2^{M-q-1} K_q) = \mathrm{EXP}\,(-j\, 2\pi \sum_{q=O}^{M-1} 2^{-q-1} K_q \left[ \sum_{P=O}^{q} 2^P i_p + \sum_{P=q+1}^{M-1} 2^P i_p \right])$$

$$= \mathrm{EXP}\,(-j\, 2\pi \sum_{q=O}^{M-1} 2^{-q-1} K_q \sum_{p=O}^{q} 2^P i_p) = \prod_{q=O}^{M-1} \mathrm{EXP}\,(-j\, 2\pi \frac{K_q}{2^{q+1}} \sum_{P=O}^{q} 2^P i_p) \tag{6}$$

is used in EQN. (5) to yield

$$N \cdot A(K) = \sum_{i_o=O}^{1} \mathrm{EXP}\,(-j\, 2\pi\, K_o i_o/2) \sum_{i_1=O}^{1} \mathrm{EXP} \left[ -j\, 2\pi\, K_1\, (i_o + 2^1 i_1)/4 \right] \cdots \sum_{i_{M-1}=O}^{1} \mathrm{EXP}\,(-j\, 2\pi\, i\, K_{M-1}/N)\, X(i) \tag{7}$$

The sum is performed on one bit of i at a time, $i_{M-1}$ first, $i_{M-2}$ second, etc., each time overstoring the original data with computed values. Each sum on $i_p$ is computed for $k_q = 0$ and $k_q = 1$ and the two values are stored back at the locations corresponding to $i_p = 0$ and $i_p = 1$, respectively. Thus summing on a bit of i in effect replaces a bit of the address of the data array with the corresponding bit of K. Reordering is required because of the order of the bits in i and K given in Equations (2) and (3).

In the RO algorithm, the back transform is the sum

$$X\,(i) = \sum_{K=O}^{N-1} A(K)\, \mathrm{EXP}\,(+j\, 2\pi\, iK/N) \tag{8}$$

The SO sum is inverted bit by bit according to

$$X\,(i) = \sum_{K_{M-1}=0}^{1} \mathrm{EXP}\,(+j\, 2\pi\, iK_{M-1}/N) \sum_{K_{M-2}=0}^{1} \mathrm{EXP}\,(+j\, 2\pi \frac{K_{M-2}}{4} \sum_{p=0}^{M-2} 2^p i_p) \cdots \sum_{K_0=0}^{1} \mathrm{EXP}\,(+j\, 2\pi\, K_0 i_0/2)\, A(K)\,, \tag{9}$$

where the sum on K was expressed as sums on the bits of K and the trigonometric identity (6) was used to distribute the complex exponential as in Equation (7).

| Real Part | Imaginary Part | Coded Form |
|---|---|---|
| +111111 | +222222 | 1.11111222222E05 |
| -111111 | +222222 | -1.11111222222E05 |
| +111111 | -222222 | +1.11111222222E95 |
| -111111 | -222222 | -1.11111222222E95 |
| +222 | +111111 | 2.22111111000E02 |
| +222 | +111 | 2.22000111000E02 |
| 0 | +1 | 1.00000000000E-06 |
| 0 | -1 | 1.00000000000E84 |
| -1 | -1 | -1.00000100000E90 |
| 0 | -1.1 | 1.10000000000E85 |
| +1 | +1 | 1.00000100000E00 |

TABLE I
Examples of Coded Data used in FFT.

## III. Description of Examples and General Applications

Several simple example programs were used to test the transform routines and are shown here to provide the user with a general idea of how the transforms can be used. The first such program generates a simple square pulse, or aperture function, of a specified extent. The input array in this case consists of zero values outside the aperture and values of 1000 inside the aperture. The input values are all real with zeros stored as the imaginary parts. These functions, with the corresponding transforms normalized and plotted in decibels, are shown in Fig. 1A and 1B. The scale on the transforms is 25dB/inch. Bizonally and trizonally shaded apertures and their transforms are shown in Fig. 1C and 1D. The large input values ranging from 500 to 1500 were used to limit the effects of round-off error which becomes significant in an integer transform if the input integer values are very small. The transform of a constant function will overflow if the value of the constant is greater than $(10^7-1)/128$. Thus, normalizing the input array of this value will avoid overflow and provide reasonably good accuracy for most transforms.

The graphs in Fig. 1 demonstrate several well known aspects of antenna design, especially the sidelobe suppression achieved with shaded apertures. These transforms were all obtained using the FFTSO program on the input array and then running the reordering program. The same results would also have been obtained by reordering the input and then using the FFTRO program. The second example is shown in Fig. 2. The input array shown in Fig. 2(A) consists of 128 equally spaced samples of the function (sin x)/x where $-6\pi \leqslant x \leqslant 6\pi$. The output shown in Fig. 2(B) was generated by reordering the input and running the FFTRO program.

Although these examples are quite simple and involve only real input arrays, the transform routines themselves are very flexible and are designed to use complex data points. The direct transforms are themselves quite useful in fields such as antenna and array design, where they can be used to find the far-field radiation patterns of various aperture functions; filter design and calculation of filter response functions; statistical analysis, where they can be used to find characteristic functions and probability density functions, etc. In addition, the basic transform routines can be used to develop programs for power spectra, cross power spectra, autocorrelation and cross correlation functions, convolutions, covariance, autocovariance and cross variance functions.

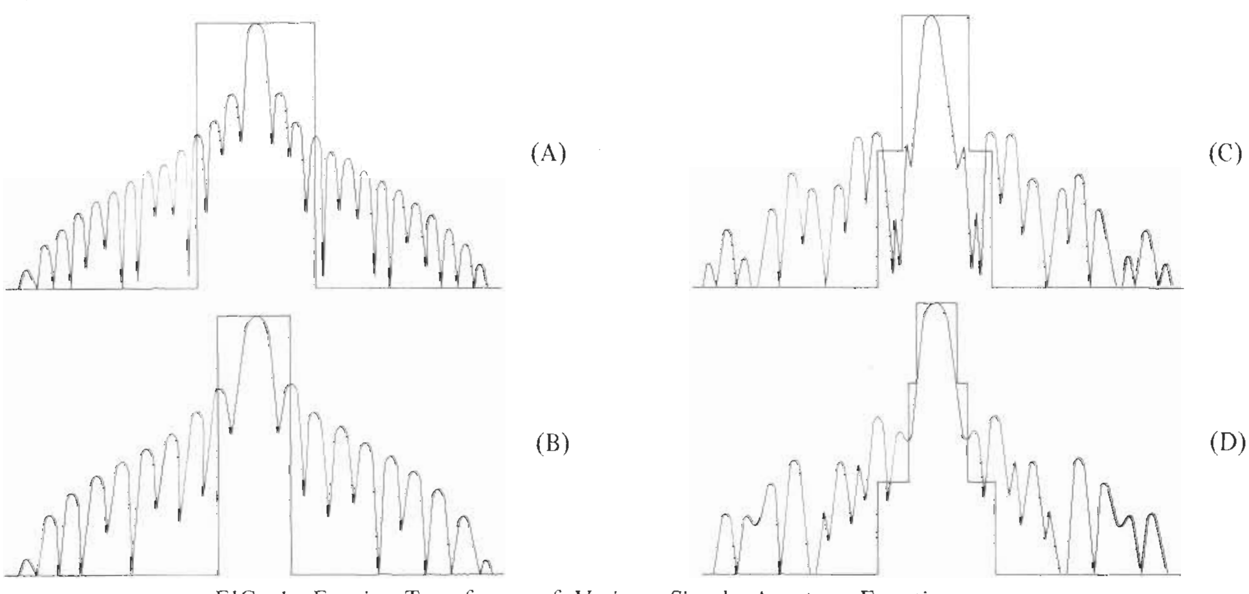
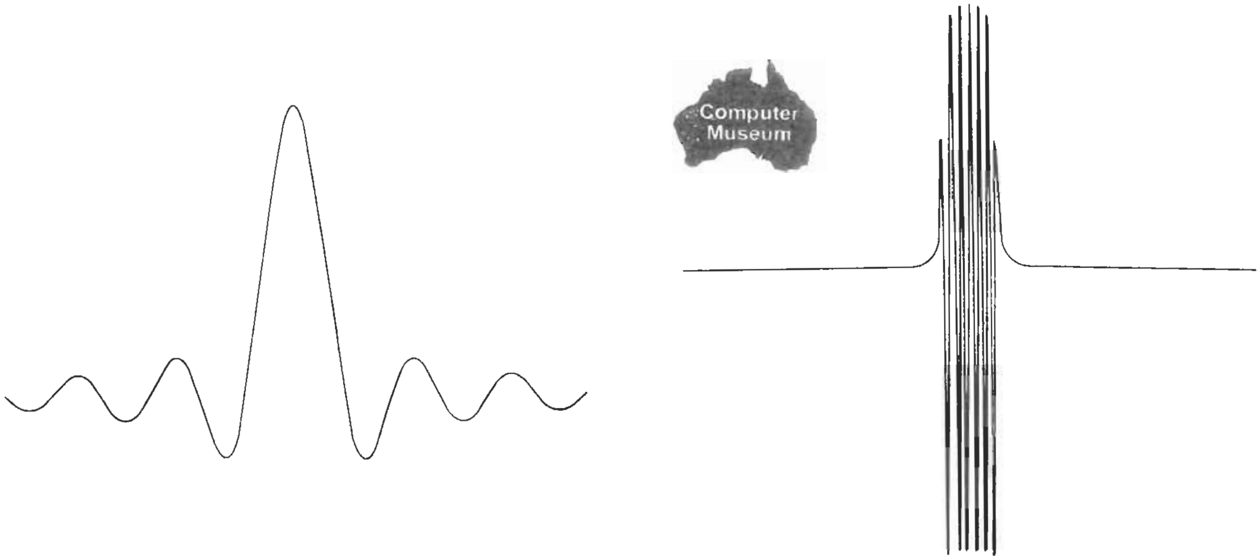



FIG. 1. Fourier Transforms of Various Simple Aperture Functions.



(A) (B)

FIG. 2. Sin X/X and its Fourier Transform.

## IV. User Instructions

A general outline of the use of the FFTRO, FFTSO and REORDER programs is given in Fig. 3. The input array consists of 128 complex points stored in registers 1 through 128 of the extended memory. These points can be made to be totally real or totally imaginary by encoding zeros for the real or imaginary parts. A user program is required to generate the input array of samples from the function to be transformed. It is suggested that the encoding subroutine found in the FFTRO and FFTSO programs be used to code the input points. This subroutine should be called with the real part of the complex number in the X register and the imaginary part in the Y register. These values should be properly scaled at this point, as the encoding subroutine requires that they be no longer than six digits and converts them to integers by truncation. As was previously mentioned, it is a good idea to use fairly large values for the input array so that errors introduced by successive truncations are kept to a minimum. The encoding subroutine returns the coded complex number to the Y register. This value should then be stored in the proper storage register by the user's program.

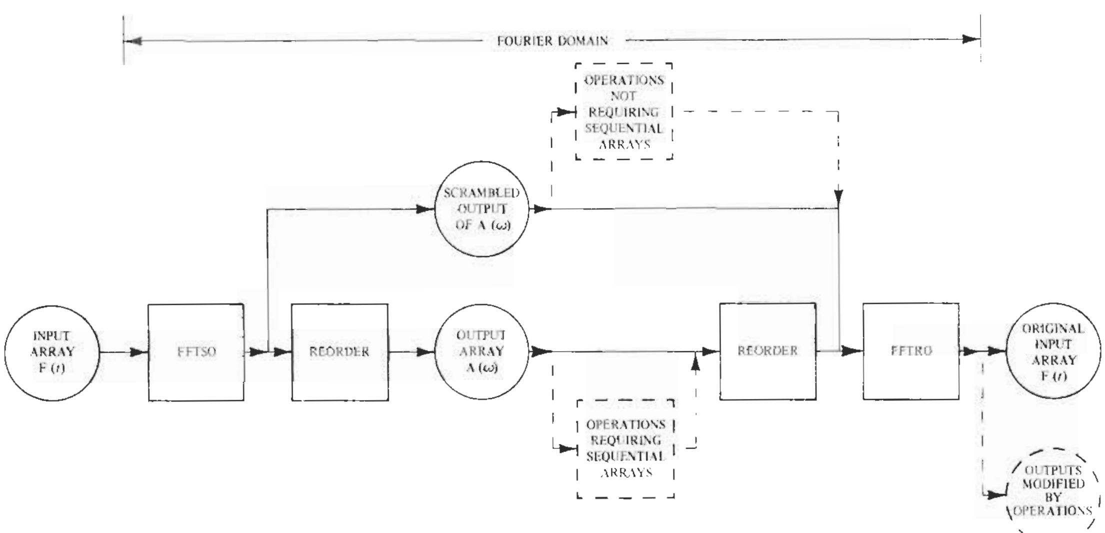



FIG. 3. Flow Chart for FFT Operations.

After the input array has been calculated and stored, it may be transformed using the routines provided. Use of the FFTSO program and then the REORDER program overstores the input array with the transformed array. Since the REORDER program also conjugates, the output array is actually the complex conjugate of the transform of the original array. As was noted previously, this feature can be easily changed and is included as a matter of convenience to enable the user to back-transform using the same SO algorithm. Using the notation given in Section II, the results of running the FFTSO program and then the REORDER program are given in Table 2. Reordering the input and then running the FFTRO routine gives the same results.

As is implied by the indices of the arrays in the table, some of the output arrays are not in sequential order even after the reordering program has been run. This is a natural result of the digital Fourier transform algorithm and is easily remedied by having an output plot routine that either plots the contents of registers 1 through 128 sequentially or plots registers 65 through 128 and then 1 through 64. Again, in writing output or plot routines it is suggested that the decoding subroutine be used in the same form as found in the FFTRO and FFTSO programs. This routine takes the coded complex number given in the X register and returns the decoded real part in the Y register and the imaginary part in the Z register.

The execution times for the FFTRO and the FFTSO programs are practically the same and are on the order of 3.5 minutes for an input array of 128 points. The REORDER program takes approximately 45 seconds to run for the same size array. There are no displays during execution of any of the programs.

| Input | Output |
|---|---|
| X(J) | N·A*(K) |
| A*(K) | X(J) |
| A(K) | X*(N–J+1) |
| X*(N–J+1) | N·A(K) |
| X(N–J+1) | N·A*(N–K+1) |
| A*(N–K+1) | X(N–J+1) |
| A(N–K+1) | X*(J) |
| X*(J) | N·A(N–K+1) |

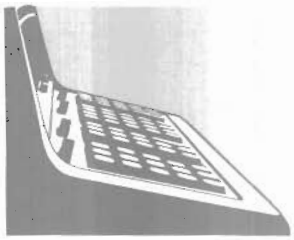
Table 2

Output of transform and recording programs for various input arrays (the asterisk represents the complex conjugate).

Michael L. Boyd received his B.A. in Math and Physics from Austin College in Sherman, Texas. He worked at the Southwest Center for Advanced Studies (now The University of Texas at Dallas) before coming to Applied Research Laboratories in 1968. He is currently employed as a staff scientist in the Sonar Research Division of ARL in the Propagation and Scattering Section.

James K. Beard received his B.S.E.E. from The University of Texas in 1962, his M.S.E.E. from the University of Pittsburgh in 1963 and his Ph.D. in Electrical Engineering from The University of Texas in 1968. He is currently employed as a staff scientist in the Signal Physics Division of Applied Research Laboratories. Before coming to ARL in 1968, Dr. Beard was employed by Westinghouse Electric Corporation and by Tracor, Incorporated. He is a member of Phi Eta Sigma, Eta Kappa Nu, Tau Beta Pi, and Sigma Xi.

J. Wesley Wauson is currently an undergraduate student at The University of Texas at Austin where he is majoring in Math and Computer Science. He is employed as a laboratory research assistant in the Sonar Research Division of Applied Research Laboratories, Austin, Texas.



# BESSEL FUNCTIONS

by

J. Robert Cooke, Ph. D.

**PART NO.**
**09100-70068**

The power series representation for $J_n(x)$ (Hp Program 09100-70025) is useful only for small x values ($x < 10$) due to the slowness of convergence and large roundoff error.* Asymptotic expansions are useful for large x when $|x| >> |n|$ The method of generating the Bessel functions, due to Miller, overcomes these restrictions.

Bessel Functions of First and Second Kind of Orders 0 and 1

$$[J_0(x), \quad J_1(x), \quad Y_0(x), \quad Y_1(x), \quad x > 0]$$

The Bessel functions of the first kind are obtained by choosing $\hat{J}_{k+1} = 0$ and $\hat{J}_k = 1$ which may be used to obtain trial values $\hat{J}_p$ using the recurrence relation

$$\hat{J}_{p-1} = (2p/x)\ \hat{J}_p - \hat{J}_{p+1}$$

with descending order where $k > \max(n, x)$. Normalization is obtained from the relation

$$J_0 + 2\sum_{m=1}^{\infty} J_{2m} = 1$$

The Bessel function of the second kind of order zero is obtained from

$$Y_0(x) = (2/\pi)\left\{J_0(x)\ \left[\ln \frac{x}{2} + \gamma\right] + 2\sum_{m=1}^{\infty} \frac{(-1)^{m-1}}{m} J_{2m}(x)\right\}.$$

The Bessel function of the second kind of order one is obtained from the Wronskian

$$J_1(x)\,Y_0(x) - J_0(x)\,Y_1(x) = 2/(\pi\, x)$$

Usually the error is less than one unit in the 10th significant figure, but for all $0 < x \leqslant 100, 0 \leqslant n \leqslant 50$ (when convergent), the error is less than five units in the tenth digit.

| *Argument | Error bound for $J_0$, $J_1$, $J_2$ (for 09100-70025) |
|---|---|
| $x \leqslant 5$ | less than 1 unit in 10th digit |
| $5 \leqslant x \leqslant 7$ | less than 2.5 units in 9th digit |
| $7 \leqslant x < 10$ | less than 4 units in 8th digit |

### References

*"Generation of Bessel Functions on High Speed Computers," Irene Stegun and Milton Abramowitz, Mathematical Tables and Other Aids to Computation, Volume 11, 1957, pp. 255-257*

*Milton Abramowitz and Irene Stegun, Handbook of Mathematical Functions, National Bureau of Standards, 4th printing, 1965, Chapter 9*

**Editor's Note:** Dr. Cooke has also written the other Bessel Function programs described below. These can be ordered through the local HP sales office.

Part Number 09100-70069

Asymptotic Expansions for Bessel Functions of the First and Second Kinds and the Modified Bessel Functions of the First and Second Kinds.

$$[J\nu(x), \quad Y\nu(x), \quad I\nu(x), \quad K\nu(x), \quad x >> \nu \geqslant 0, \quad \nu \text{ real}]$$

Part Number 09100-70070

The Bessel Function of the First Kind, the Modified Bessel Function of the First Kind, the Spherical Bessel Function of the First Kind and the modified Spherical Bessel Function of the First Kind.

$$[J_n(x), \quad I_n(x), \quad j_n(x), \quad i_n(x) \quad x > 0, n \geqslant 0, \text{ n integer}$$

Part Number 09100-70071

Bessel Functions of the Second Kind, Spherical Bessel Functions of the Second Kind, and Modified Spherical Bessel Functions of the Third Kind.

$$[Y_n(x), \quad y_n(x), \quad k_n(x), \quad x > 0, n \geqslant 0, \text{ n integer}]$$

Programs 09100-70070 and 09100-70071 use Miller's recurrence relations.

J. Robert Cooke received his Ph.D. from North Carolina State University in Biological and Agricultural Engineering with a minor in Applied Mathematics. He is an Associate Professor of Agricultural Engineering at Cornell University, and is a member of the Bioengineering Committee of ASAE.

EQUIPMENT NEEDED [X] 9100A [X] 9100B [ ] 9120 [ ] 9125 [ ] 9160 [ ] 9101 [ ] 9104 [ ] 9106

[ ] DEGREES [X] RADIANS [ ] FLOATING [ ] FIXED DECIMAL WHEEL AT [ ] PRESS [ ] [ ] [ ] ON 9120

| | | | | |
|---|---|---|---|---|
| 1. | ENTER PROGRAM (Starting Address is 00) | | | |
| 2. | PRESS: END | | | |
| 3. | PRESS: CONTINUE | 0 | 0 | 0 |
| 4. | ENTER DATA | X | | |
| 5. | PRESS: CONTINUE | X | $Y_0(x)$ | $Y_1(x)$ |
| 6. | If desired, PRESS: RCL | $J_0(x)$ | $J_1(x)$ | $Y_1(x)$ |
| 7. | To run another value return to Step 2. | | | |

Note: Program Step 3c may be changed to Pause for diagnostic purposes.

## PROGRAM STEPS

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | CLR | 20 | 30 | DIV | 35 | 60 | DIV | 35 | 90 | RUP | 22 |
| 01 | STP | 41 | 31 | b | 14 | 61 | YE | 24 | 91 | X | 36 |
| 02 | XEY | 30 | 32 | X | 36 | 62 | f | 15 | 92 | DN | 25 |
| 03 | 2 | 02 | 33 | IFG | 43 | 63 | DIV | 35 | 93 | + | 33 |
| 04 | DIV | 35 | 34 | 3 | 03 | 64 | YE | 24 | 94 | 2 | 02 |
| 05 | YTO | 40 | 35 | 8 | 10 | 65 | e | 12 | 95 | X | 36 |
| 06 | d | 17 | 36 | SFL | 54 | 66 | DIV | 35 | 96 | π | 56 |
| 07 | 3 | 03 | 37 | AC+ | 60 | 67 | UP | 27 | 97 | DIV | 35 |
| 08 | 0 | 00 | 38 | XEY | 30 | 68 | \|Y\| | 55 | 98 | YTO | 40 |
| 09 | RUP | 22 | 39 | d | 17 | 69 | EEX | 26 | 99 | c | 16 |
| 0a | + | 33 | 3a | DIV | 35 | 6a | 9 | 11 | 9a | e | 12 |
| 0b | 2 | 02 | 3b | c | 16 | 6b | 8 | 10 | 9b | X | 36 |
| 0c | RUP | 22 | 3c | CNT | 47 | 6c | X<Y | 52 | 9c | 1 | 01 |
| 0d | d | 17 | 3d | X | 36 | 6d | 4 | 04 | 9d | UP | 27 |
| 10 | INT | 64 | 40 | b | 14 | 70 | d | 17 | a0 | π | 56 |
| 11 | X | 36 | 41 | XEY | 30 | 71 | EEX | 26 | a1 | DIV | 35 |
| 12 | DN | 25 | 42 | RUP | 22 | 72 | 1 | 01 | a2 | d | 17 |
| 13 | + | 33 | 43 | - | 34 | 73 | 3 | 03 | a3 | DIV | 35 |
| 14 | YTO | 40 | 44 | YTO | 40 | 74 | YE | 24 | a4 | RDN | 31 |
| 15 | c | 16 | 45 | b | 14 | 75 | b | 14 | a5 | - | 34 |
| 16 | CLR | 20 | 46 | 1 | 01 | 76 | \|Y\| | 55 | a6 | f | 15 |
| 17 | c | 16 | 47 | YE | 24 | 77 | X>Y | 53 | a7 | DIV | 35 |
| 18 | UP | 27 | 48 | c | 16 | 78 | 5 | 05 | a8 | 2 | 02 |
| 19 | 1 | 01 | 49 | - | 34 | 79 | 0 | 00 | a9 | RUP | 22 |
| 1a | XTO | 23 | 4a | GTO | 44 | 7a | d | 17 | aa | X | 36 |
| 1b | b | 14 | 4b | 1 | 01 | 7b | LN | 65 | ab | c | 16 |
| 1c | X>Y | 53 | 4c | c | 16 | 7c | XEY | 30 | ac | XEY | 30 |
| 1d | 5 | 05 | 4d | CHS | 32 | 7d | . | 21 | ad | END | 46 |
| 20 | 6 | 06 | 50 | 2 | 02 | 80 | 5 | 05 | | | |
| 21 | YTO | 40 | 51 | 0 | 00 | 81 | 7 | 07 | | | |
| 22 | c | 16 | 52 | RDN | 31 | 82 | 7 | 07 | | | |
| 23 | c | 16 | 53 | GTO | 44 | 83 | 2 | 02 | | | |
| 24 | XEY | 30 | 54 | 0 | 00 | 84 | 1 | 01 | | | |
| 25 | 2 | 02 | 55 | 7 | 07 | 85 | 5 | 05 | | | |
| 26 | - | 34 | 56 | DN | 25 | 86 | 6 | 06 | | | |
| 27 | DIV | 35 | 57 | f | 15 | 87 | 6 | 06 | | | |
| 28 | π | 56 | 58 | YE | 24 | 88 | 4 | 04 | | | |
| 29 | X | 36 | 59 | f | 15 | 89 | 9 | 11 | | | |
| 2a | DN | 25 | 5a | + | 33 | 8a | + | 33 | | | |
| 2b | COS | 73 | 5b | b | 14 | 8b | f | 15 | | | |
| 2c | UP | 27 | 5c | + | 33 | 8c | X | 36 | | | |
| 2d | c | 16 | 5d | XEY | 30 | 8d | 4 | 04 | | | |

## EXAMPLES

$Y_1(.01) = -6.\ 367\ 859\ 628(1)$
$Y_0(.01) = -3.\ 005\ 455\ 637(0)$
$J_1(.01) = 4.\ 999\ 937\ 500(-3)$
$J_0(.01) = 9.\ 999\ 750\ 001(-1)$
$Y_1(17.5) = 9.\ 857\ 279\ 871(-2)$
$Y_0(17.5) = -1.\ 604\ 111\ 925(-1)$
$J_1(17.5) = -1.\ 634\ 199\ 694(-1)$
$J_0(17.5) = -1.\ 031\ 103\ 982(-1)$
$Y_1(25.0) = -9.\ 882\ 996\ 480(-2)$

The range of the argument may be extended by entering a positive, *even* integer in the z register. For example, 80→Z and x = 75→X yields (approx. 45 sec)

$Y_1(75) = -3.521\ 378\ 521(-2)$
$Y_0(75) = -8.536\ 904\ 763(-2)$
$J_1(75) = -8.513\ 999\ 503(-2)$
$J_0(75) = 3.464\ 391\ 381(-2)$

Program B* using an asymptotic expansion is much faster for large arguments provided the argument is much larger than the order.

*Part Number 09100-70069

# NUCLEAR REACTION KINEMATICS; TRANSFORMATION OF DIFFERENTIAL CROSS SECTION DATA AND ANGLES

PART NO. 09100-73301

by J. D. A. Roeders, Ph. D.

In nuclear physics it is often necessary to calculate the energies of particles emitted in nuclear reactions, e.g., for the identification of peaks in particle spectra. Also in many cases the differential cross sections measured in the laboratory system have to be transformed to the center of mass system in order to facilitate the comparison of experiment with theory.

We use the following notation:

Consider the reaction x(a,b)y
$m_a$ = mass of incident particle
$m_x$ = mass of target nucleus
$m_b$ = mass of outgoing particle, observed at laboratory angle $\theta$
$m_y$ = mass of residual nucleus
$E_a$ = laboratory energy of incident particle
Q = Q-value of the reaction
$\bar{\theta}$ = the center of mass angle corresponding to $\theta$

### Program I (Nuclear reaction kinematics)

This program calculates for non-relativistic energies the energy $E_b$ of particle b emitted at laboratory angle $\theta$. The following expression is used (Reference 1):

$$E_b^{\frac{1}{2}} = \frac{(m_a m_b E_a)^{\frac{1}{2}} \cos\theta \pm \left\{ m_a m_b E_a \cos^2\theta + (m_y+m_b)\,[m_y Q + (m_y-m_a)E_a] \right\}^{\frac{1}{2}}}{m_y+m_b} \qquad (1)$$

For endoergic reactions $E_b$ is double-valued for $E_a$ between the threshold energy

$$E_s = -Q\,(m_y+m_b)\,/\,(m_y+m_b-m_a) \text{ and } E_o = -Qm_y/(m_y-m_a)$$

When $E_a$ is greater than $E_o$, $E_b$ is single valued as is also the case for exoergic reactions; in expression (1) the plus sign should be taken.
In certain cases, e.g. endoergic reactions with $E_s < E_a < E_o$, not all values of $\theta$ between $0^0$ and $180^0$ are possible: particles cannot be emitted at angles exceeding an angle $\theta_m$ defined by

$$\cos^2\theta_m = \frac{-(m_y+m_b)\,[m_y\,Q + (m_y-m_a)E_a]}{m_a m_b E_a} \qquad (2)$$

The output of the program depends on the various possibilities mentioned above. If $E_a < E_s$ for the reaction considered, only $E_s$ is calculated. When $E_s \leqslant E_a \leqslant E_o$ and $\theta < \theta_m$ the program calculates the two possible values of $E_b$ ($E_b^+$ and $E_b^-$), while for energies exceeding $E_o$ the single value of $E_b$ is given. The program is organized in such a way that $\theta$ has to be entered after $E_a$. At the position where $\theta$ has to be entered, the display shows the maximum value of $\theta$ (= $\theta_m$ or $180^0$) which is possible for the reaction considered.

---

**Editor's Note:** The complete program is available through the HP Calculator Program Catalog.

---

*References:* 1. *D. Halliday: "Introductory nuclear physics" chapter 13 (John Wiley, New York)*

2. *J.B. Marion and A.S. Ginzbarg: "Tables for the transformation of angular distribution data from the laboratory system to the center of mass system" (Shell Development Company, Houston)*

**Program II (Transformation of differential cross section data and angles).**

This program calculates the center of mass angle $\bar{\theta}$ corresponding to the laboratory angle $\theta$ according to Reference 2:

$$\sin(\bar{\theta} - \theta) = k \sin\theta \qquad \text{where } k^2 = \frac{m_a m_b}{m_x m_y}\left[1 + \frac{m_a + m_x}{m_x}\frac{Q}{E_a}\right]^{-1}$$

Also is calculated the ratio $G(k,\theta) \equiv \bar{\sigma}/\sigma$ of the differential cross sections in the center of mass system and in the laboratory system.

$$G(k,\theta) = \frac{(1 - k^2 \sin^2\theta)^{\frac{1}{2}}}{[k\cos\theta + (1 - k^2\sin^2\theta)^{\frac{1}{2}}]^2}$$

For $k < 1$ all values of $\theta$ are possible; for $k \geqslant 1$ values of $\theta$ are restricted to the angular range from $\theta^0$ to $\theta_m$, where $\theta_m$ is given by eq. (2) or by

$$\operatorname{tg}\theta_m = 1/(k^2 - 1)^{\frac{1}{2}}$$

The output of the program consists of $G(k,\theta)$ and $\bar{\theta}$, but when $\theta > \theta_m$ for the reaction considered, only $\theta_m$ is calculated. When $k^2 < 0$ which means that $E_a < E_s$ (endoergic reactions) the threshold energy for that reaction is calculated.

---

## EXAMPLE

Equipment needed: 9100A/B and 9120A

Reaction $^{12}C(d,n)^{13}N$ $m_a = 2$ $m_x = 12$ $m_b = 1$ $m_y = 13$ $Q = -0.28$ MeV For this reaction $E_s = .3267$ MeV and $E_0 = .3309$ MeV.

**Program I**

| | |
|---|---|
| 1. $E_a = .20$ MeV | |
| $E_s = .326667$ | .326667 |
| $E_s = .326667$ | .326667 |
| $E_a = .2$ | .2 |
| 2. $E_a = .33$ MeV | |
| $E_a = .33$ | .33 |
| $\theta_m = 62^0\ .5764$ | 62.576351 |
| | 0. |
| $\theta = 60^0$ | |
| $E_b^+ = .001625$ | .001625 |
| $E_b^- = .000314$ | .000314 |
| | 0. |
| 3. $E_a = 1.00$ MeV | |
| $E_a = 1$ | 1. |
| $\theta_m = 180^0$ | 180. |
| | 0. |
| $\theta = 150^0$ | |
| $E_b = .413241$ | .413241 |
| | 0. |
| | 0. |

**Program II**

| | |
|---|---|
| 1) $E_a = .20$ Mev | |
| $E_a = .2$ | .2 |
| $E_a = .2$ | .2 |
| $E_s = .3267$ | .3267 |
| 2) $E_a = .33$ Mev | |
| $\theta = 90^0$ | |
| $\theta = 90$ | 90. |
| $\theta = 90$ | 90. |
| $\theta = 62.5764$ | 62.5764 |
| $\theta = 60^0$ | |
| $\bar{\theta} = 137.3342$ | 137.3342 |
| $G(k,\theta) = 0.3580$ | .3580 |
| $\theta = 60$ | 60. |
| 3) E = 1.0 Mev | |
| $\theta = 150^0$ | |
| $\bar{\theta} = 153.9562$ | 153.9562 |
| $G(k,\theta) = 1.2938$ | 1.2938 |
| $\theta = 150$ | 150. |

---

J.D.A. Roeders studied experimental physics at the University of Groningen, The Netherlands, and obtained his Ph.D. in Nuclear Physics at that university in 1971. He is a member of the scientific staff of "Kernfysisch Versneller Instituut" (Nuclear Accelerator Institute) of the University of Groningen.

# BESSEL FUNCTIONS OF THE FIRST KIND

by
James A. Cochran, Ph.D. and Charles M. Nagel

**PART NO. 09100-70072**

This program calculates the Bessel function of the first kind $J_\nu(z)$ for given positive argument z and non-negative order $\nu$. The technique that is employed is particularly suited for use on a desk calculator with programming capabilities.
The user may recall the three-term recursive relation satisfied by the Bessel functions $J_\nu(z)$ and $Y_\nu(z)$:

$$f_{\nu+1}(z) = \left(\frac{2\nu}{z}\right) f_\nu(z) - f_{\nu-1}(z) \qquad (1)$$

Let us consider an upward recurrence starting with the initial values $f_\nu(z) = 0$ and $f_{\nu+1}(z) = 1$. The general solution of (1) may be written as a linear combination of Bessel functions of the first and second kinds, and hence

$$f_\nu(z) = \gamma J_\nu(z) + \delta Y_\nu(z) = 0 \; , \qquad (2)$$

$$f_{\nu+1}(z) = \gamma J_{\nu+1}(z) + \delta Y_{\nu+1}(z) = 1 \; , \qquad (3)$$

$$\vdots$$

$$f_{\nu+n}(z) = \gamma J_{\nu+n}(z) + \delta Y_{\nu+n}(z) \; . \qquad (4)$$

Since $|Y_{\nu+n}(z)|$ increases [and $J_{\nu+n}(z)$ decreases] extremely rapidly with increasing order $n >> z$, it is clear that eventually equation (4) becomes

$$f_{\nu+n}(z) \doteq \delta Y_{\nu+n}(z)$$

to any desired numerical accuracy. The constant $\delta$ can be determined from (2), (3), and the familiar Wronskian relation as

$$\delta = -\left(\frac{\pi z}{2}\right) J_\nu(z)$$

thus yielding finally

$$f_{\nu+n} \doteq -\left(\frac{\pi z}{2}\right) J_\nu(z)\, Y_{\nu+n}(z) \qquad (5)$$

Equation (5) is the key result. It shows how an appropriate upward recurrence using (1) followed by an independent evaluation of $Y_{\nu+n}(z)$ can lead to an approximate value of $J_\nu(z)$. In this program, values of $Y_{\nu+n}(z)$ are determined using the three-term Debye expansion

$$Y_\nu(\nu \operatorname{sech} \alpha) \sim -\frac{\exp[\nu(\alpha - \tanh \alpha)]}{\sqrt{\frac{1}{2}\pi\nu \tanh \alpha}} \sum_{k=0}^{2} (-1)^k \frac{U_k(\coth \alpha)}{\nu^k}$$

$U_0(t) = 1,$
$U_1(t) = (3t - 5t^3)/24,$
$U_2(t) = (81t^2 - 462t^4 + 385t^6)/1152.$

Depending upon the numerical accuracy desired, the appropriate value of $\nu_0 = \nu + n$ is selected from the empirical curves shown below. Note especially, that in the case $\nu > z$, $\nu_0$ should be increased say by the smallest integer greater than or equal to the difference $(\nu - z)$.

EQUIPMENT NEEDED [X] 9100A [X] 9100B [X] 9120 [ ] 9125 [ ] 9160 [ ] 9101 [ ] 9104 [ ] 9106

[ ] DEGREES [X] RADIANS [X] FLOATING [ ] FIXED DECIMAL WHEEL AT [ ] PRESS [X] [Y] [Z] ON 9120

| | | | | |
|---|---|---|---|---|
| 1. | ENTER: PROGRAM at +00 | | | |
| 2. | PRESS: CONTINUE | 0 | 0 | 0 |
| 3. | ENTER: DATA | $\nu_0$ | z | $\nu$ |
| | NOTE: Select $\nu_0$ from graph for desired accuracy. The order of the Bessel Function ($\nu$) and $\nu_0$ must differ by an integer. | | | |
| 4. | PRESS: CONTINUE | $\nu$ | z | $J_\nu(z)$ |
| 5. | To run another value, return to step 2. | | | |

James A. Cochran

Charles M. Nagel Jr.

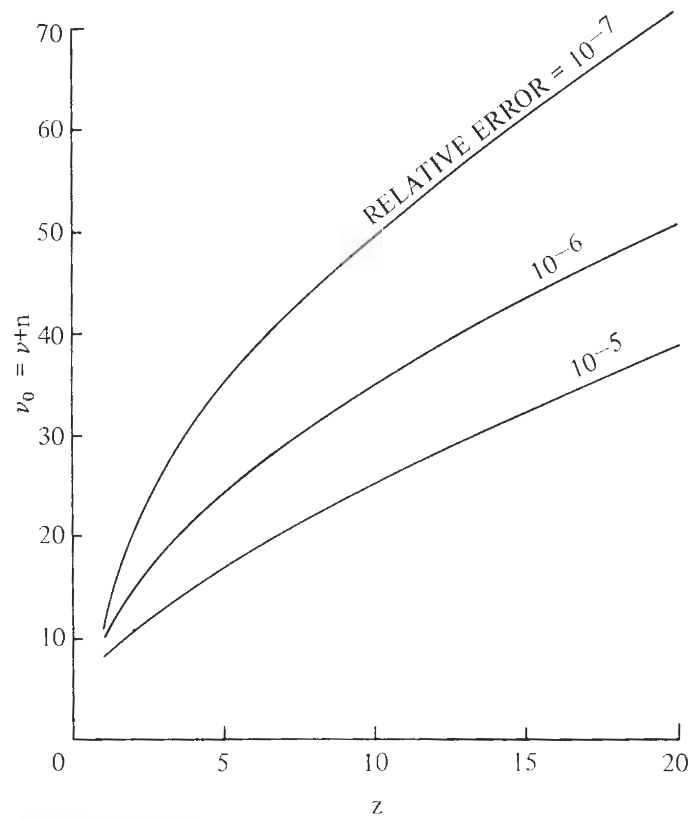
James A. Cochran received his BS (1956) and MS (1957) in physics, and Ph.D. (1962) in mathematics, all from Stanford. He joined Bell Telephone Laboratories, Whippany, New Jersey in 1962 as a Member of Technical Staff. Dr. Cochran has worked on various problems concerned with antenna analysis and the application of electromagnetic theory to microwave systems. He currently supervises an applied mathematics group doing basic systems research.

He is active in church, library, and scout work, and is an avid tennis player and camper. He has authored a book (in press) on Integral Equations, and more than a dozen published technical papers. Dr. Cochran is a member of Phi Beta Kappa and Sigma Xi honorary fraternities, as well as the American Mathematical Society and SIAM.

Charles M. Nagel Jr. joined Bell Telephone Laboratories, Whippany, New Jersey in 1965 as a Member of Technical staff. He received his BS (1964) and MS (1967) in physics, both from Stevens Institute of Technology.

Mr. Nagel was killed in a subway accident in early 1971. At that time he was working in optical communications theory. Prior to that he did research in numerical aspects of waveguide propagation and the study of electromagnetic scattering.

Mr. Nagel was a devoted husband and the father of two small children. He had published several technical papers, and was a member of Tau Beta Pi and Pi Delta Epsilon honorary fraternities and the American Association for the Advancement of Science.




SMOOTHED CONTOURS OF EQUAL RELATIVE ERROR IN $J_\nu(Z)$ FOR $\nu \leq Z$. FOR $\nu > Z$ INCREASE $\nu_0$ BY THE SMALLEST INTEGER $\geq (\nu - Z)$.

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00 | CLR | 20 | 20 | EXP | 74 | 40 | a | 13 | 60 | + | 33 | 80 | DIV | 35 |
| 01 | STP | 41 | 21 | RDN | 31 | 41 | UP | 27 | 61 | b | 14 | 81 | b | 14 |
| 02 | PNT | 45 | 22 | X | 36 | 42 | 3 | 03 | 62 | X | 36 | 82 | RDN | 31 |
| 03 | AC+ | 60 | 23 | RDN | 31 | 43 | 8 | 10 | 63 | YTO | 40 | 83 | YTO | 40 |
| 04 | DIV | 35 | 24 | √ | 76 | 44 | 5 | 05 | 64 | a | 13 | 84 | b | 14 |
| 05 | DN | 25 | 25 | DIV | 35 | 45 | X | 36 | 65 | 0 | 00 | 85 | X | 36 |
| 06 | YTO | 40 | 26 | 1 | 01 | 46 | 4 | 04 | 66 | XTO | 23 | 86 | DN | 25 |
| 07 | d | 17 | 27 | RUP | 22 | 47 | 6 | 06 | 67 | b | 14 | 87 | - | 34 |
| 08 | YTO | 40 | 28 | DIV | 35 | 48 | 2 | 02 | 68 | 2 | 02 | 88 | DN | 25 |
| 09 | c | 16 | 29 | RDN | 31 | 49 | - | 34 | 69 | UP | 27 | 89 | CHS | 32 |
| 0a | UP | 27 | 2a | YTO | 40 | 4a | a | 13 | 6a | π | 56 | 8a | GTO | 44 |
| 0b | X | 36 | 2b | b | 14 | 4b | X | 36 | 6b | DIV | 35 | 8b | 7 | 07 |
| 0c | 1 | 01 | 2c | UP | 27 | 4c | 8 | 10 | 6c | DN | 25 | 8c | 0 | 00 |
| 0d | XEY | 30 | 2d | UP | 27 | 4d | 1 | 01 | 6d | √ | 76 | 8d | DN | 25 |

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | - | 34 | 30 | X | 36 | 50 | + | 33 | 70 | UP | 27 | 90 | a | 13 |
| 11 | DN | 25 | 31 | YTO | 40 | 51 | a | 13 | 71 | d | 17 | 91 | DIV | 35 |
| 12 | √ | 76 | 32 | a | 13 | 52 | X | 36 | 72 | UP | 27 | 92 | e | 12 |
| 13 | UP | 27 | 33 | 5 | 05 | 53 | 1 | 01 | 73 | 1 | 01 | 93 | DIV | 35 |
| 14 | ARC | 72 | 34 | X | 36 | 54 | 1 | 01 | 74 | + | 33 | 94 | UP | 27 |
| 15 | HYP | 67 | 35 | 3 | 03 | 55 | 5 | 05 | 75 | f | 15 | 95 | c | 16 |
| 16 | TAN | 71 | 36 | - | 34 | 56 | 2 | 02 | 76 | X=Y | 50 | 96 | PNT | 45 |
| 17 | UP | 27 | 37 | DN | 25 | 57 | DIV | 35 | 77 | 8 | 10 | 97 | PNT | 45 |
| 18 | DN | 25 | 38 | X | 36 | 58 | f | 15 | 78 | d | 17 | 98 | END | 46 |
| 19 | - | 34 | 39 | 2 | 02 | 59 | DIV | 35 | 79 | YTO | 40 | | | |
| 1a | f | 15 | 3a | 4 | 04 | 5a | DIV | 35 | 7a | d | 17 | | | |
| 1b | X | 36 | 3b | DIV | 35 | 5b | DN | 25 | 7b | 2 | 02 | | | |
| 1c | RDN | 31 | 3c | f | 15 | 5c | + | 33 | 7c | X | 36 | | | |
| 1d | CHS | 32 | 3d | DIV | 35 | 5d | 1 | 01 | 7d | e | 12 | | | |



## EXAMPLES

From the empirical graph, a value of $\nu_0$ = 50 should suffice to calculate $J_{10}(10)$ with a relative error of the order of $10^{-7}$. The Handbook of Mathematical Functions[1] gives $J_{10}(10)$ = 0.207486107.

| | | |
|---|---|---|
| $\nu_0$ | 5.0 | 01 |
| $J_\nu(z)$ | 2.074861276 | -01 |
| z | 1.0 | 01 |
| $\nu$ | 1.0 | 01 |

To calculate the value of $J_{5.5}(z)$ at its first turning point, z = 6.95974575 with similar accuracy, the graph suggests that $\nu_0$ be set to $\nu_0$ = 40.5 (Note that the order of the Bessel function and $\nu_0$ must differ by an integer). The Royal Society Math Tables[2] shows $J_{5.5}(6.95974575)$ = 0.36355735.

| | | |
|---|---|---|
| $\nu_0$ | 4.05 | 01 |
| $J_\nu(z)$ | 3.635573875 | -01 |
| z | 6.95974575 | 00 |
| $\nu$ | 5.5 | 00 |

Calculating the value of $J_{5.5}(z)$ at its first zero, z = 9.35581211 with the same value of $\nu_0$ yields a value of zero to better than $10^{-9}$.

| | | |
|---|---|---|
| $\nu_0$ | 4.05 | 01 |
| $J_\nu(z)$ | 2.462823743 | -10 |
| z | 9.35581211 | 00 |
| $\nu$ | 5.5 | 00 |

*References*

[1]*Handbook of Mathematical Functions, M. Abramowitz and I. A. Stegun editors, National Bureau of Standards Applied Mathematics Series 55, 1964.*

[2]*Royal Society Math Tables, Bessel Functions Part III, Zeros and Associated Values, F. W. J. Olver editor, Cambridge University Press, 1960.*

MODEL 10

# CONVERTING A DECIMAL NUMBER TO BINARY CODE

**PART NO. 09810-75881**

When a student is studying the concept of number bases he is required to put out quite a bit of effort converting from one base to another. If an easy way were available for the student to check his work, more effort could be directed toward understanding rules governing the number system under investigation. Due to the familiarity with the decimal system, he is faced with the laborious mechanics of base conversion.

Program 09810-75881 was designed to help the student and teacher in this area. The program uses the newest member of the HP Calculator family; the 9810A with printer, printer/alpha and math ROM (Read Only Memory) options. The program takes any whole decimal number greater than zero and converts it to its binary equivalent. The algorithm successively factors out the largest power of two and stores away the corresponding binary bits in memory for later recall.

The program is written in a conversational mode, i.e., operating instructions for operation are printed out. This allows the student to use the program without referring to a write-up. When student inputs numbers that fall outside the set defined by the program, the calculator prints out a message for the operator suggesting that the instructions be reread.

With this program the student can perform duplicate operations with decimal numbers and then convert answers to binary equivalent to check against answers obtained using the base under study. For example, if two binary numbers are to be multiplied the answer can be checked by carrying out the same problem in decimal and using program number 09810-75881 to convert the answer to the binary equivalent.

An additional program for the conversion of any number in any base to its decimal equivalent, Part Number 09810–75882, is available also. An additional set of programs allowing binary math to be performed on HP Desk Top Calculators can also be prepared.

## EQUIPMENT NEEDED

Model 10 Calculator (basic), Column Printer, Model 11210A
Mathematics Block, Model 11211A Printer Alpha Block

## USER INSTRUCTIONS

1. PRESS: END
2. ENTER: PROGRAM
3. PRESS: CONTINUE
4. Follow Printed Instructions on Tape.

**Editor's Note:** The complete program is available through the HP Calculator Program Catalog.

EXAMPLE

```
THIS PROGRAM
CONVERTS ANY
WHOLE DECIMAL
NUMBER GREATER
THAN ZERO TO A
BINARY NUMBER

INPUT YOUR
NUMBER

          0.300

READ
INSTRUCTIONS
AGAIN

THIS PROGRAM
CONVERTS ANY
WHOLE DECIMAL
NUMBER GREATER
THAN ZERO TO A
BINARY NUMBER

INPUT YOUR
NUMBER

         1.230*

READ
INSTRUCTIONS
AGAIN

THIS PROGRAM
CONVERTS ANY
WHOLE DECIMAL
NUMBER GREATER
THAN ZERO TO A
BINARY NUMBER
```

```
INPUT YOUR
NUMBER

        -2.000*

READ
INSTRUCTIONS
AGAIN

THIS PROGRAM
CONVERTS ANY
WHOLE DECIMAL
NUMBER GREATER
THAN ZERO TO A
BINARY NUMBER

INPUT YOUR
NUMBER

       200.000*

THE BINARY
NUMBER IS

         1.000
         1.000
         0.000
         0.000
         1.000
         0.000
         0.000
         0.000
```

```
INPUT YOUR
NUMBER

       201.000*

THE BINARY
NUMBER IS

         1.000
         1.000
         0.000
         0.000
         1.000
         0.000
         0.000
         1.000
INPUT YOUR
NUMBER

         1.000*

THE BINARY
NUMBER IS

         1.000
INPUT YOUR
NUMBER

         0.000*

THE BINARY
NUMBER IS

         0.000
INPUT YOUR
NUMBER
```

# PROGRAMMING TIPS

## CONVERTING 9100A/B PROGRAMS TO MODEL 10

Users with 9100A/B programs for their specific applications may need to write versions of these to use on their Model 10 calculators. This conversion is a fairly simple process if a few basic rules are followed, and it precludes the need for designing a completely new program. Here is a suggested procedure:

1. Before starting, it is best to make a table showing a cross-reference between the 9100 registers used for data storage and corresponding Model 10 registers you arbitrarily assign. This table should be referred to in making all of the storage register changes.

The choice of corresponding registers is optional, except for the two alpha registers which have exactly the same function in both the 9100A/B and the Model 10. Use the relationship:

| 9100A Register | 9810A Register |
|---|---|
| e | b |
| f | a |

These are the accumulator registers, which are cleared by a CLR instruction.

A sample table is shown below. Note that except for a and b, three spaces must be left for addressing each register. The a and b registers require only one step for storage or recall, as in the 9100A/B.

| 9100 Register | 9810A Register |
|---|---|
| +0 | 000 |
| 1 | 001 |
| 2 | 002 |
| ⋮ | ⋮ |
| 9 | 009 |
| -0 | 010 |
| -1 | 011 |
| -2 | 012 |
| ⋮ | ⋮ |
| -9 | 019 |
| a | 020 |
| b | 021 |
| c | 022 |
| d | 023 |
| e | b |
| f | a |
| -a | 024 |
| -b | 025 |
| -c | 026 |
| -d | 027 |
| -e | 028 |
| -f | 029 |

Table 1 Storage Register Cross Reference

2. Rewrite the program on a Model 10 program form (Part No. 09810-90016). Leave four blank steps following each conditional (IF) statement and GO TO statement, and use four digits for each address. Abbreviated addressing is possible in some cases, as explained in the Model 10 operating manual, but using the complete four-digit address is less error-susceptible. The basic Model 10 with 500 program steps has adequate memory for any existing 9100A/B program not using transcendental functions.

Another table which is useful in converting programs should be made up as you go along with the conversion. This shows GO TO and conditional branching addresses:

| | Jump-to Address | |
|---|---|---|
| Jump No. | 9100 | 9810 |
| 1 | 5a | 0081 |
| 2 | 72 | 0111 |
| 3 | b3 | 0156 |

Numbering each jump exit and entry, both in the original program and the converted one, helps to assure proper branching.

3. Programs for the 9100 including trigonometric, logarithmic, or exponential functions require the Math block for the converted Model 10 version.

If you have the basic Model 10 without plug-in blocks, you can convert |Y|, AC+, AC- and RCL by using short substituting step sequences:

| \|Y\| | | AC+ | | AC- | | RCL | |
|---|---|---|---|---|---|---|---|
| Step | Code | Step | Code | Step | Code | Step | Code |
| CLX | 37 | XTO | 23 | XTO | 23 | | |
| X>Y | 53 | + | 33 | - | 34 | b | 14 |
| XEY | 30 | a | 13 | a | 13 | XEY | 30 |
| CHS | 32 | YTO | 40 | YTO | 40 | a | 13 |
| XEY | 30 | + | 33 | - | 34 | | |
| CNT | 47 | b | 14 | b | 14 | | |

When |Y|, AC+, AC-, or RCL occur directly following an IF statement, branch to a subroutine to do the above conversions.

4. Hyperbolic functions can be calculated by subroutines in the 9810A, with the Math block, using the identity:

$$\text{arctanh } x = \frac{1}{2} \ln \left(\frac{1+x}{1-x}\right), \quad x^2 < 1$$

5. Since there is no ROLL DOWN key on the Model 10, substitute two ROLL UP's for each 9100 ROLL DOWN.

6. In plotter programs a FMT, DN instruction causes the 9125 plotter pen to move to the designated coordinates and then drop. The same instruction causes the Model 62 pen to drop at its original location and draw a line to the designated point. FMT, UP should therefore be substituted for the first FMT, DN instruction in the 9100/9125 program.

7. Take note of any 9100A/B programming tricks that are machine-dependent. These tricks may not behave in the same way when transferred to the Model 10.

## DIVISION BY 2 IN X REGISTER

Jan Cooper of Applied Research Laboratories, The University of Texas at Austin, discovered a technique for dividing a positive number in the X register by 2 without disturbing the contents of the Y and Z registers, as well as saving a step compared to the sequence UP, 2, DIV, DN.

The sequence is $e^x$, $\sqrt{x}$, ln x. This works with numbers from about $10^{-10}$ to 230.2 on the Model 9100A/B, and to 227.9 on the Model 10 with the Math block. Results are accurate up to 10 digits over most of this range.

## ROUNDING NUMBERS

This programming tip was submitted by E. R. Head of the Dukane Corporation, St. Charles, Illinois.

In some financial calculating programs it is necessary to round off certain calculated dollar values – round them off in fact, as well as in display. This routine will round off a number in the Y register of the 9100A/B or the Model 10 to the desired number of decimal places. It takes advantage of the automatic rounding that occurs as digits go "spilling" out the end of the register, past the guard digits.

In effect one simply adds on a number large enough to cause the unwanted digits to "spill" out the end. Adding $10^9$ to a number will cause the first two digits to the right of the decimal to fall in the guard digit position, rounding off all decimals in excess of two. Adding $10^8$ will save three decimal places, etc. By the choice of the exponent, one can round off to two decimals, no decimals, or round off to the nearest tens, hundreds, or other desired magnitude. The rounded number is retrieved by subtracting the same power of ten that was added.

EXAMPLE

| STEP | CODE | DISPLAY x | y | z |
|---|---|---|---|---|
| EEX | 26 | | 23.445 | |
| P* | 09 | 1.000 000 000 09 | | |
| + | 33 | | | |
| - | 34 | | 23.45 | |

*P = (11 – n), where n = desired number of digits to right of decimal after rounding.

## MODEL 10 PROGRAM LIBRARIES

You can order the following Model 10 program libraries through the local Hewlett-Packard sales office:

| Description | Part Number |
|---|---|
| 9810A MATH PAC VOL. I[1] | 09810–70000 |
| 9810A MATH SUBROUTINE PAC[2] | 09810–70100 |
| 9810A STAT PAC VOL. I[3] | 09810–70800 |
| 9810A SURVEYING PAC VOL. I | 09810–74100 |
| 9810A CARDIO-PULMONARY PAC | 09810–75325 |
| 9810A GENIE CIVIL I (French Structures) | 09810–77200 |
| 9810A VERMESSUNG I (German Surveying) | 09810–78000 |
| 9810A SURVEY PAC VOL. I (British) | 09810–78200 |
| 9810A TOPOGRAPHIE I (French Surveying) | 09810–78300 |

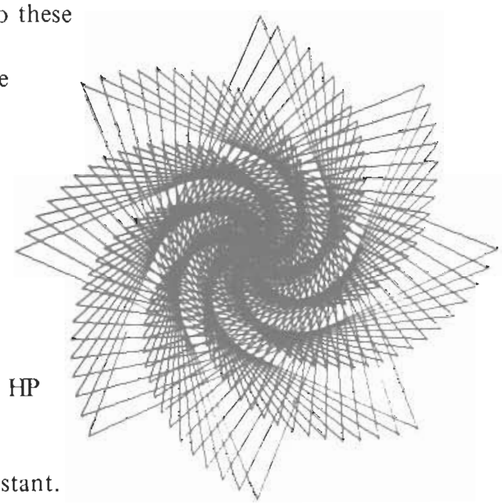
[1] One copy supplied with purchase of Model 9810A.

[2] One copy supplied with purchase of Model 11213A User Definable Function Block.

[3] One copy supplied with purchase of Model 11214A Statistics Block.

# CALCULATOR ART CONTEST

The automatic generation of artistic designs using a programmed system has become even more intriguing with the advent of the HP Model 10 Calculator. *KEYBOARD* has received many requests for another calculator art contest similar to the one held last year (see *KEYBOARD* Vol. 3, Nos. 1 and 2). In response to these requests, there will be another contest.



The 1972 contest will allow enough time for the calculator artists in every country to submit entries. The deadline for receiving entries at Loveland will be May 15, 1972.

To encourage students to participate, there will be a student branch of the contest with winners and prizes separate from the general branch.

## CONTEST RULES

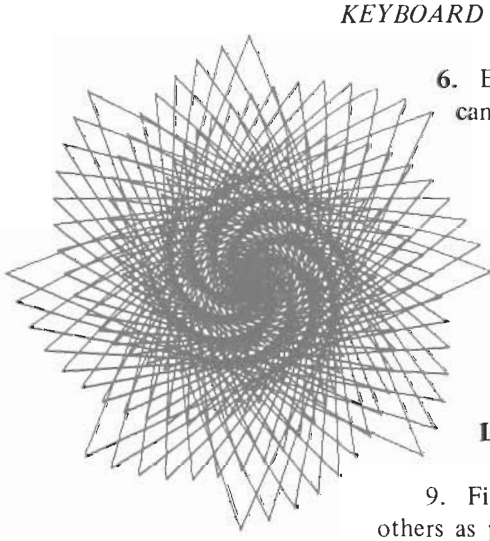
1. Entries may be any aesthetic designs made with an HP calculator and plotter of the 9100 or 9800 series.

2. There is no limit to the number of entries per contestant.

3. Entries must be in either black or red, on plain white paper at least 8½ X 11 inches (21,5 X 28 cm), preferred size, but not larger than 11 X 17 inches (28 X 43 cm).

4. Entries may be in a maximum of two colors. Two-color entries must include one complete two-color plot and a separate plot of each color.

5. Each entry must include a short description of how the plot was made, complete user instructions, a written copy of the program, and either recorded magnetic card(s) or marked cards. *KEYBOARD* will replace the magnetic cards.



6. Entries will become the property of Hewlett-Packard, and cannot be returned.

7. Entries must be mailed either flat with a stiff cardboard protector or in a mailing tube. They may not be folded.

8. Entries may be sent either to the nearest *KEYBOARD* field editor or directly to Loveland. All entries must be received by a field editor not later than April 15, 1972, or by the *KEYBOARD* editor at Loveland not later than May 15, 1972.

9. First, second, and third-place winning entries, and a many others as possible in both the main contest and the student branch, will be published in *KEYBOARD* Vol. 4 No.3, Aug. 1972.