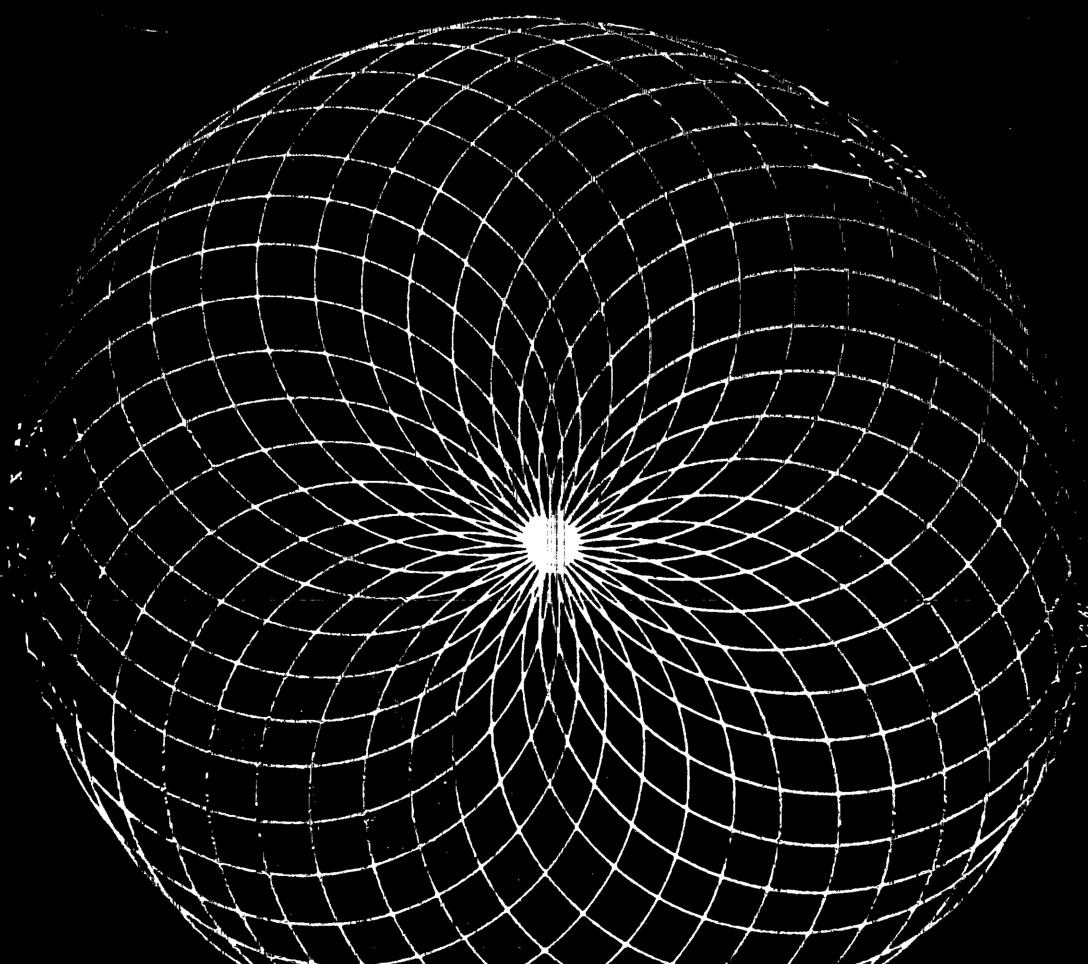
1969 - Volume 8, Number 1

DECUSCOPE

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY



CONTENTS

PAGE

President's Message 1

Spring Meeting Reminder 1

1969 DECUS Executive Board 2

DECUS European Committee 2

DECUS Statistics for 1968 2

A Remote Time Share Terminal With Graphical Output Capabilities 3-9

GRAPHAS 10

Software for Sale or Lease 11,12

Letters 13,14

Programs Available from Authors 14

Programming Notes - LINC-8, PDP-8

PROGOFOP Modification to Use the Type PC01 High Speed Reader/Punch 15,16

Modification to Random Number Generator . . . 17,18

Guide Call and Note on LINC-8 Event Counter 18

General Purpose I/O Panel for a PDP-9 19

A PDP-8/Paper Tape Reader Interface to Simulate Teletype Operation 20

DECUSCOPE HAS BEEN PUBLISHED SINCE APRIL 1962 AND IS THE OFFICIAL NEWSLETTER FOR DIGITAL EQUIPMENT COMPUTER USERS SOCIETY.

IT IS PUBLISHED PERIODICALLY AT THE DECUS OFFICE, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.

TELEPHONE: AC 617, 897-5111, EXT, 2414

EDITOR: MRS. ANGELA J. COSSETTE, DECUS

CIRCULATION: 4,600 COPIES PER ISSUE

PRESIDENT'S MESSAGE

On September 19 and 20 I attended a DECUS European Seminar in Edinburgh, Scotland. I am happy to report that the meeting was considered a success by all. There were about 150 attendees, with liberal representation by DEC. John Oldfield, the Arrangements Chairman, did a fine job in providing excellent accommodations for the meeting at the University of Edinburgh. The meeting was run along the lines of the American meetings with application papers and PDP-8 and PDP-9 workshops. Most of the papers this time were about work done on the PDP-8, since this is the dominent product line in Europe, but indications are the PDP-9 and PDP-10 papers will be forthcoming next time. John Hall, the Papers Chairman, had his hands full trying to fit a large number of papers into a crowded agenda. Here things differed from the American meetings in that all papers were distributed previous to the sessions. Attendees were thus given a chance to read the paper, and the oral presentation was limited to five or ten minutes, and in some cases, not at all. At the end of each session the authors came forward to answer questions from the audience. The sessions were run in parallel, and this met with some disapproval from attendees. Since many were interested in all papers, it was recommended that future meetings be longer with no parallel sessions. It was also decided to continue with one meeting per year in Europe because international travel restrictions are severe in some countries.

My trip was made possible by a recent decision of the Executive Board to provide funds from the DECUS treasury to enable the DECUS President to attend the European meeting each year, and to enable the European Chairman to attend one American meeting per year. It is clearly evident that this has been a wise decision. The American and European DECUS organizations are getting large and vigorous, and we simply must maintain close ties. From the quality of the papers presented, I would say that there is little difference between the workers on both sides of the Atlantic.

With the rapid growth of membership in Europe it is becoming clear that we need a Secretary there. Just recently, DEC approved the hiring of a DECUS Secretary for Europe. She will be headquartered in the Regional DEC office in Geneva. The European DECUS Secretary would work closely with Angela Cossette and be responsible for serving the users in the entire area.

Richard J. McQuillin

1969 SPRING MEETING "CALL FOR PAPERS REMINDER"

The Spring DECUS Meeting will be held on May 12 and 13 at the Colonial Statler Hilton Inn, Wakefield, Massachusetts. A "Call for Papers" was issued recently. Those planning to present papers should have their abstracts into the DECUS office by March 10. Sessions topics are: Biomedicine, Interactive Systems and Computer Graphics, Education, High Energy Physics, Low Energy Physics, Digital Signal Processing, and Business Applications.

1969 DECUS EXECUTIVE BOARD

PRESIDENT	Mr. Richard McQuillin Inforonics, Inc. Cambridge, Massachusetts
PROGRAMMING CHAIRMAN	Mr. Michael S. Wolfberg Moore School of Electrical Engineering University of Pennsylvania Philadelphia, Pennsylvania
MEETINGS CHAIRMAN	Prof. Philip R. Bevington Department of Physics Case Western Reserve University Cleveland, Ohio
EQUIPMENT CHAIRMAN	Mr. Sypko W. Andreae Lawrence Radiation Laboratory University of California Berkeley, California
RECORDING SECRETARY	Dr. Daniel M. Forsyth Department of Psychology University of Vermont Burlington, Vermont
PUBLICATIONS CHAIRMAN	Mr. George Cooper Logic, Inc. Detroit, Michigan
EXECUTIVE SECRETARY	Mrs. Angela J. Cossette DECUS Digital Equipment Corporation Maynard, Massachusetts
DEC REPRESENTATIVE	Mr. John A. Jones Digital Equipment Corporation Maynard, Massachusetts

1968/1969 DECUS EUROPEAN COMMITTEE

CHAIRMAN	Mrs. Gillian D. Spittle Strand Hotels Ltd. 12 Sherwood Street London W.1, England
PAPERS SECRETARY	Mr. John Hall Room G09 Building 443 AERE Harwell Didcot Berkshire, England
ACCOMMODATION SECRETARY	Mr. Birger Kvaavik The Axel Johnson Institute for Industrial Research Nynashamn, Sweden
COMMITTEE MEMBERS	Dr. John Elder DAMTP Silver Street Cambridge, England
	Dr. Hans-Jurgen Trebst Physics Institute Tandem Lab. Erwin-Rommel-Str.1 D-852 Erlangen, Germany
	Dr. Chris Malpus Department of Physiology University of Leeds Leeds, England

DECUS Statistics for Year 1968

	1968	1967
MEMBERSHIP		
New Applications	1472	1189
Total Membership	3476	2008
Delegates	1620	925
Individuals	1856	1049
Non-Member List	602	480
Total on Mailing List	4262	2488
DECUSCOPE		
Circulation	4600	2775
PROGRAM LIBRARY		
Number of programs submitted (120-Accepted 15-In Process)	135	127
Total number of programs in library	428	294
Number of programs obsolete	5	8
Number of requests filled	1712	950
Number of programs involved in requests	10,272	4655
Number of tapes involved in requested programs	30,861	8815
Paper tapes	30,504	8690
DECtapes and LINCtapes	357	125
Approximate number of paper tapes reproduced and verified	31,000	9300
DECUS MEETINGS		
Attendance Figures		
Canadian	103	70
European	150 (approx.)	110
Spring	270	197
Fall	390	200

A REMOTE TIME SHARE TERMINAL WITH GRAPHICAL OUTPUT CAPABILITIES

S. Pardee and P. E. Rosenfeld
Bell Telephone Laboratories
Whippany, New Jersey

For some time, users of time-shared computers have felt the need for a low-cost graphical output device that could be located next to any remote terminal. The device must be able to draw both characters and vectors and to complete a picture of reasonable complexity fast enough so the user will not be inclined to take a coffee break whenever new output is being produced. All this while working over a dialed-up telephone line at a 110-baud rate!

Two major problems had to be solved in order to make this device a reality. First, an inexpensive output device was required that could handle both characters and vectors at speeds greater than can be achieved by mechanical plotters. Second, it was necessary to reduce the transmission time of pictorial information from the central computer to the remote terminal.

Within the last year two solutions to the first problem have become available. One is the large screen storage scope; the other is a hard-copy system that uses a conventional CRT to expose a dry-silver paper. At present the storage scope is somewhat less expensive and considerably smaller in size than the hard-copy device. It also is somewhat faster, as the latter device does not allow the output to be viewed until the paper has been passed through a 15-sec developing cycle.

The solution to the second problem has also appeared recently, in the form of a small, inexpensive computer, such as the PDP-8/L. Placing a small computer at the remote terminal helps to solve the transmission bandwidth problem in the following ways:

1. The computer can be used as a character generator and a vector generator, thus allowing information to be transmitted in an efficiently encoded format rather than in an incremental move form which is quite inefficient.

2. If the picture is stored in the memory of the small computer before it is drawn on the output device, it will be possible to transmit the graphical data in a structured form, making it more compact and thereby further reducing the transmission time. By structured form we mean the use of graphical subroutining, so that an element of a picture need be transmitted only once regardless of the number of times it appears in the picture.

3. By allowing the large central computer to specify where the graphical material it sends is to be stored in the memory of the small remote computer, it becomes possible for the central computer to edit a picture which is stored in the remote computer. Thus if the picture which is presently desired differs only slightly from the last one that was received, only a small amount of new data need be transmitted.

In addition to increasing the transmission efficiency, the small computer provides several other desirable features. One is the flexibility of a programmed controller as opposed to a special purpose hard-wired device. This allows the operation of the terminal to be easily modified if necessary. Another is the possibility of storing several pictures simultaneously in the memory of the small computer, and allowing the user to look alternately at one and then another. This should be helpful in determining if the job is proceeding in the right direction. This last feature becomes unnecessary, of course, if a hard-copy device is used instead of or in addition to the storage scope.

HARDWARE CONFIGURATION

A terminal as described above is now under construction at Bell Telephone Laboratories, and programming for both the small and the large computer has begun. The terminal consists of a PDP-8/L computer with an ASR33 Teletype, a Tektronix 611 storage scope, and a type 103 DATA-PHONE dataset. The scope and the dataset are interfaced to the computer using Digital Equipment M-series modules. The scope interface consists of a 10-bit X register and D/A, an 11-bit Y register and D/A, and a Z-axis control. The origin of the coordinate system is the lower left-hand corner of the scope. Values of X may range from 0 to 1023_{10} and of Y from 0 to 1435_{10}, because of the aspect ratio of the 611 scope. The dataset interface consists of serial-to-parallel and parallel-to-serial converters, timing and control circuits, and a circuit that checks for odd parity. The terminal is built into a standard desk-type console as shown in Figure 1. Consideration was given to replacing the ASR33 with an electronic keyboard and using the storage scope to monitor keyboard activity. However, in order to do this it would have been necessary to remove the primary display from the storage tube when keyboard activity is allowed and this was considered undesirable.

CHARACTER AND VECTOR GENERATION

As indicated earlier, the PDP-8/L is programmed to perform the functions of character and vector generation. Characters will be formed on the storage tube using a 5 x 7 dot matrix, centered on the current electron-beam position. The computer program will cause the beam to move in a fixed sequence through the 35 positions of the matrix, intensifying the appropriate positions to form the desired character. Three of the PDP-8/L's 12-bit words will be required to store the intensification pattern for each character; thus 288 memory locations will suffice to store a font of 96 characters. The program to scan the bean through the matrix and intensify it where necessary is expected to take approximately 100_{10} instructions, so the entire character-generation routine will fit in about 400_{10} words.

Scaling and rotation of the characters will also be provided. At scale 1 the characters will not be resolvable, so this scale will normally not be used. Scaling from 2 to 63 will be provided. Rotation of the characters by 90° in the positive direction will facilitate the labeling of graphs.

Character spacing will be automatic and will be equal to 6 spaces in the character matrix. The receipt of a line feed will cause an automatic carriage return as well as spacing down 10 spaces in the character matrix. The left-hand margin will be taken as the last position of the electron beam before entering the character mode.

Vectors will be produced by generating a sequence of points which best approximate the desired vector. The number of points will be equal to the magnitude of the major component.

OPERATION MODES

The terminal will have two modes of operation: Teletypewriter and Graphic, between which it will switch back and forth. In the Teletypewriter mode the user will communicate with the remote computer via the TELETYPE, with the PDP-8/L acting as a transmission path for ASCII codes between the ASR33 and the remote computer. When the PDP-8/L receives a special control character from the remote computer it will switch to the Graphic mode, and stay in this mode until another special control character switches it back to the Teletypewriter mode. While in the Graphic mode, ASCII code characters received from the remote computer will be interpreted as either graphical commands or data, and will be stored in the memory of the 8/L. Another program running in the 8/L will then interpret the stored graphical data and form a picture on the storage tube. In the initial version, two buffer areas in the 8/L memory will be used to store pictures. These buffers will normally be used alternately, thus allowing the user to view either of the two most recently transmitted displays. However, the user will also be able to alter the sequence in which the buffers are used and thus save the contents of one buffer indefinitely. When the complexity of a picture becomes so great that it will no longer fit in one buffer, both buffers will be combined to form one large one. In the event that a picture is too complex to fit into memory it will still be displayed on the scope, but the complete picture will obviously not be available for later regeneration.

GRAPHIC TRANSMISSION FORMAT

As mentioned before, upon the receipt of a special control character, the terminal will go into the Graphic mode and interpret succeeding characters as graphical op-codes and data until it receives a control character instructing it to leave the Graphic mode. Actually, there are two Graphic modes and therefore three special control characters.

The first mode, which is entered by issuing the control character 034 (FS), is called the Graphical Character mode. In this mode, all characters received after the code 034 and before either of the other two mode-changing control characters are interpreted as ASCII symbols and drawn on the face of the storage scope as such. This allows all 96 printing characters to be drawn on the scope. In addition, the scope will respond in the normal manner to carriage return and new line (line feed).

The second mode, which is entered using control character 035 (GS), is called the Graphical Master mode. In this mode, the succeeding characters are interpreted as graphical op-codes and data. An ASCII character is treated as an op-code if its most significant bit is a 0 and data if it is 1. Since the ASCII characters are seven bits (plus parity) and one bit is used to differentiate between op-codes and data, only six bits can be used for data. For those situations where 12-bit data words are required, two data characters are assembled in the 8/L as shown in Figure 2.

Where negative numbers are required, they will be transmitted in sign and magnitude form (standard PDP-2/S compliment form).

The code used to leave the Graphic mode and return to Teletypewriter mode is 021 (DC1).

In the list of op-codes that follow, the subscript H implies the sign and 5 high-order bits of a 12-bit data-word and the subscript L implies the 6 low-order bits. The op-codes are given as octal numbers that represent a particular ASCII character.

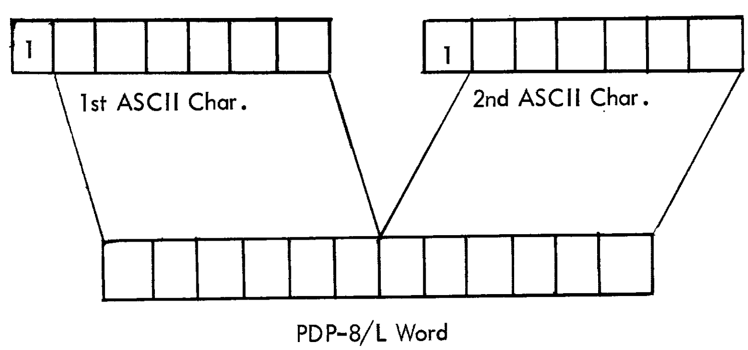


FIGURE 2

CONTROL OP CODES

SET POINTER

This op-code is followed by two data characters that define a 12-bit address in the PDP-8/L memory. This op-code instructs the PDP-8/L to start storing the picture information at the transmitted address.

ASCII Sequence:

$040, Address_H, Address_L$

BEGIN PICTURE

This op-code is followed by two data characters that define an address where a picture begins. The PDP-8/L will start interpreting information at that address.

ASCII Sequence:

$041, Address_H, Address_L$

SUB JUMP

This op-code causes the picture processing program in the PDP-8/L to jump to a graphical subroutine. The location in memory of the beginning of the subroutine is conveyed in the two data characters that follow the op-code. The address of the op-code that immediately follows the SUB JUMP is stored in the location corresponding to the transmitted address. Picture processing continues at the transmitted address +1.

ASCII Sequence:

$042, Address_H, Address_L$

JUMP INDIRECT

This op-code causes picture processing to continue from the address contained in the location corresponding to the transmitted address. This is primarily used to return from a subroutine.

ASCII Sequence:

$043, Address_H, Address_L$

JUMP

This op-code is followed by two data characters defining the memory location where picture processing should continue.

ASCII Sequence:

$044, Address_H, Address_L$

END OF DISPLAY

This op-code causes picture processing to be terminated.

ASCII Sequence:

045

ERASE

This op-code causes the picture presently being displayed to be erased. It does not affect the contents of the PDP-8/L memory.

ASCII Sequence:

046

REPEAT

This op-code is followed by two data characters that, taken together, indicate the number of times that the sequence of op-codes that follow is to be displayed. The sequence is terminated by an END REPEAT op-code.

ASCII Sequence:

$047, N_H, N_L$, ...Op Codes....., 050

END REPEAT

See Repeat

ASCII Sequence:

050

BEAM POSITIONING

SET X

This op-code is followed by two data characters that are combined to produce a 10-bit X address. The blanked beam is positioned to the new X address. Y position is unchanged.

ASCII Sequence:

$051, X_H, X_L$

SET Y

Similar to SET X but for Y position.

ASCII Sequence:

$052, Y_H, Y_L$

SET XY

Similar to SET X except that four data characters are transmitted. The first two are the X position the last two are the Y position. The blanked beam is repositioned to the new X, Y address.

ASCII Sequence:

$053, X_H, X_L, Y_H, Y_L$

INTENSIFY

Causes the beam to intensify at its present position.

ASCII Sequence:

054

SINGLE COMPONENT VECTORS

VISIBLE X

This op-code is followed by two data characters that taken together represent a signed 10-bit vector length (ΔX). A visible vector is then drawn from the present X,Y position to the point $X+\Delta X, Y$. Upon completion of this command the position of the beam remains at $X+\Delta X, Y$.

ASCII Sequence:

$055, \Delta X_H, \Delta X_L$

INVISIBLE X — Same as VISIBLE X except the beam is moved to $X+\Delta X, Y$ without the vector being drawn on the display.

ASCII Sequence:

$056, \Delta X_H, \Delta X_L$

VISIBLE Y — Same as VISIBLE X except the vector is drawn in the Y direction

ASCII Sequence:

$057, \Delta Y_H, \Delta Y_L$

INVISIBLE Y — Same as INVISIBLE X but in the Y direction.

ASCII Sequence:

$060, \Delta Y_H, \Delta Y_L$

ARBITRARY VECTORS (VISIBLE)

VECTOR — This op-code is followed by an indeterminate number of sets of four data characters. The first two data characters of each set represent the 10-bit X component of a visible vector. The last two data characters of each set represent the 11-bit Y component. The vectors are strung head to tail. The series is terminated by occurrence of any other op-code.

ASCII Sequence:

$061, \Delta X1_H, \Delta X1_L, \Delta Y1_H, \Delta Y1_L, \Delta X2_H, \Delta X2_L \ldots$..

SHORT VEC — This is similar to VECTOR except that only two data characters are required for each vector. The first data character of each pair is a 5-bit (plus sign) ΔX and the second is a 5-bit (plus sign) ΔY. This allows a more compact representation in those instances where a series of very short vectors are required to represent a curve.

ASCII Sequence:

$062, \Delta X1, \Delta Y1, \Delta X2, \Delta Y2 \ldots$..

RECTANGLE — This op-code is followed by four data characters, the data characters provide the ΔX and ΔY of a rectangle.

ASCII Sequence:

$063, \Delta X_H, \Delta X_L, \Delta Y_H, \Delta Y_L$

POINT PLOT 1 — This op-code is intended to facilitate point plotting. It is followed by at least five data characters. The first character selects which of the ASCII symbols is to be drawn at the desired points. The next two characters taken together give the X increment between points. The following characters taken in pairs give the ordinates of the desired points. The abscissa for the first point is the current X setting of the electron beam. The series is terminated by the occurrence of any other op-code.

ASCII Sequence:

064, Symbol, $\Delta X_H, \Delta X_L, Y1_H, Y1_L, Y2_H \ldots$..

POINT PLOT 2 — Same as above but both X and Y coordinates are given for each point.

ASCII Sequence:

065, Symbol, $X1_H, X1_L, Y1_H, Y1_L, X2_H \ldots$..

LINE PLOT — Similar to POINT PLOT 1, however symbols are omitted and points are joined by straight lines.

ASCII Sequence:

$066, \Delta X_H, \Delta X_L, Y1_H, Y1_L, Y2_H, Y2_L \ldots$..

CHARACTER INFORMATION

SIZE — This op-code is followed by a single data character. The data character indicates the number of address units between dots in the 5 x 7 dot matrix that makes up a single character. Once a SIZE has been received it remains in effect until another SIZE code is received. Initially SIZE is set to 2.

ASCII Sequence:

067, SIZE

ROTATE — This op-code requires no data character and remains in effect only for the characters that follow the next CHARACTER MODE code. The effect of the ROTATE command is to rotate the text through an angle of $+90^{\circ}$.

ASCII Sequence:
070

NO-OP — Op-codes 71 through 76 are spares and will be assigned as needed in the future.

ASCII Sequence:
077

EXAMPLE OF A GRAPHIC TRANSMISSION

As an example, consider what information would be transmitted to display the picture shown in Figure 3. This picture could be transmitted by sending the following data:

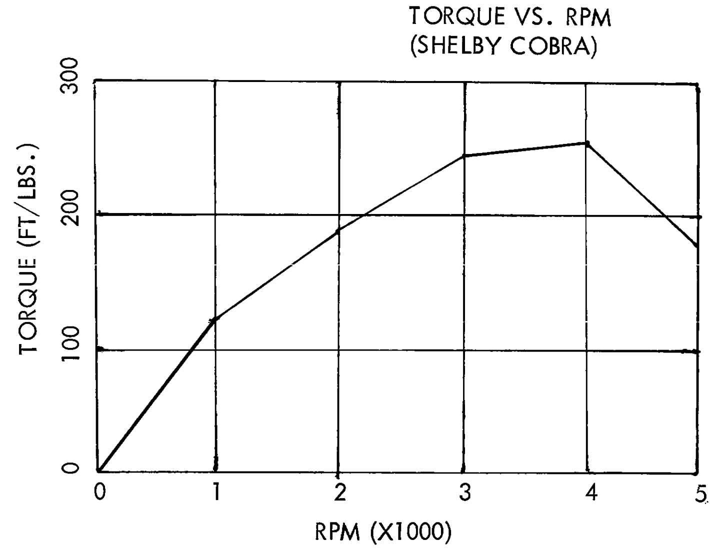


FIGURE 3 GRAPHIC EXAMPLE

OP-CODES & DATA	NO. OF CHAR.	COMMENTS
MASTER MODE	1	
SET POINTER	1	
ADDRESS	2	
ERASE	1	
SET XY	1	
X	2	Position beam to left end of the abscissa
Y	2	
REPEAT	1	
6	2	
VIS. Y	1	
ΔY	2	
INVIS. Y	1	Draw vertical lines of coordinate grid
-ΔY	2	
INVIS. X	1	
ΔX	2	
END REPEAT	1	
SET X	1	Reposition beam to origin
X	2	
REPEAT	1	
4	2	
VIS. X	1	
ΔX	2	Draw horizontal lines of coordinate system
INVIS. X	1	
-ΔX	2	
INVIS. Y	1	
ΔY	2	
END REPEAT	1	
SET XY	2	
X	2	Position for text

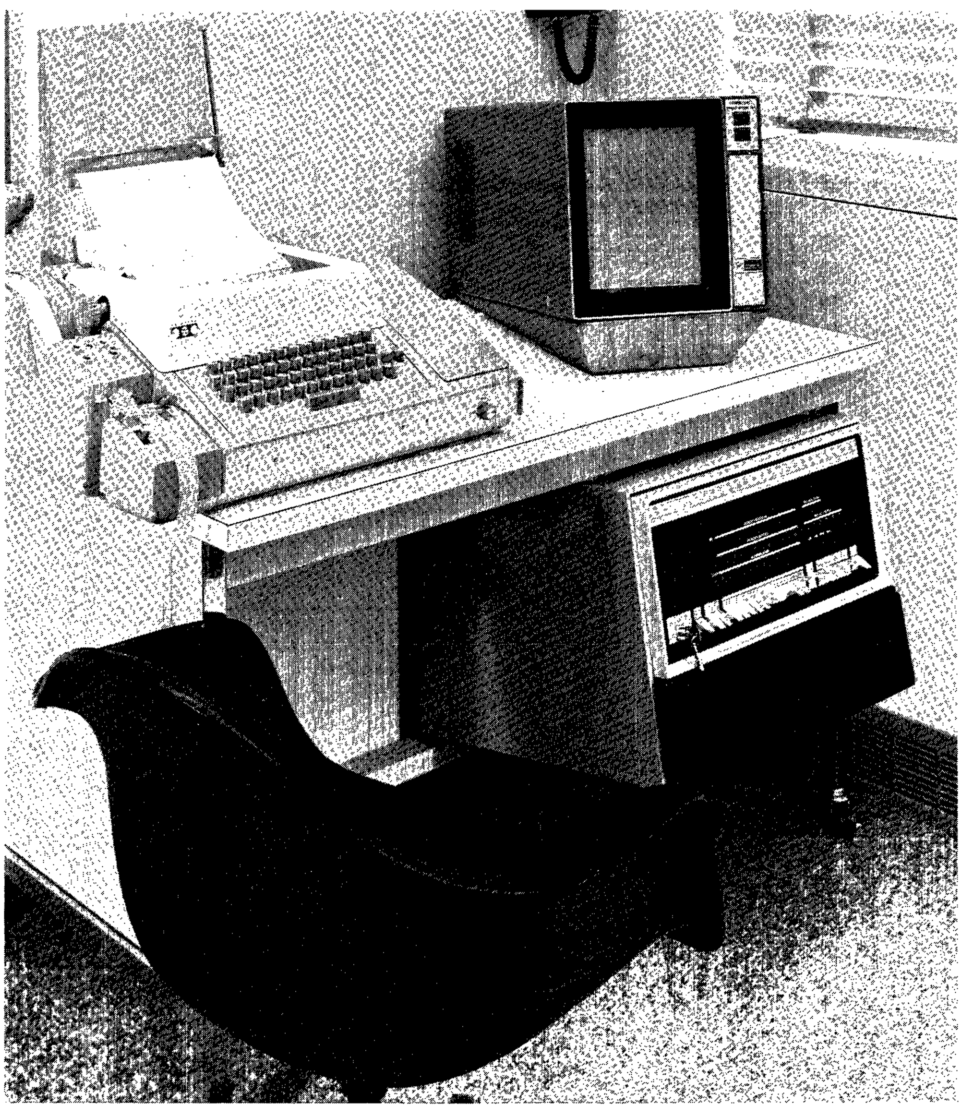
OP-CODES & DATA	NO. OF CHAR.	COMMENTS
Y	2	
CHAR MODE	1	Next 30 characters in character display mode
TORQUE sp VS. sp RPM n1 (SHELBY sp COBRA)	29	
MASTER MODE	1	
SET XY	1	
X	2	Position for Abscissa Label
Y	2	
CHAR MODE	1	Next 26 characters in character display mode
0 spspspsp 1 spspspsp 2 spspspsp 3 spspspsp 4 spspspsp 5	26	
MASTER MODE	1	
SET X,Y	1	
X	2	Position of Abscissa Text
Y	2	
CHAR MODE	1	Next 11 characters in character mode
RPM sp (X1000)	11	
MASTER MODE	1	
XET X,Y	1	
X	2	Position for Ordinate Label
Y	2	
ROTATE	1	Rotate the following 20 characters by 90^o
CHAR MODE	1	
0 spspspsp 100 spspsp 200 spspsp 300	20	
MASTER MODE	1	
SET XY	1	
X	2	Position for Ordinate Text
Y	2	
ROTATE	1	Rotate next 16 characters
CHAR MODE	1	
TORQUE sp (FT/LBS.)	16	
MASTER MODE	1	
SET X	1	
X	2	Reposition at origin
LINE PLOT	1	
ΔX	2	
Y_0	2	
Y_1	2	
Y_2	2	

OP-CODES & DATA	NO. OF CHAR.	COMMENTS
Y_3	2	
Y_4	2	
Y_5	2	
END DISPLAY	1	
BEGIN	1	
ADDRESS	2	
TTY MODE	1	
Total Characters	202	

At Teletypewriter speeds this would require ≈ 20 seconds for transmission and display. If the 1200 bit/sec line capability were used, it would require about 2 seconds to transmit the graph. It should be noted that the vast majority of transmission time was devoted to setting up the grid and labeling the axes. If a second plot were to be sent that uses the same grid and labeling, which is often the case, only the curve would have to be retransmitted since the grid has been saved in core. In that case a new curve could be transmitted in 2.0 seconds or 0.20 seconds depending on the bandwidth available.

CONCLUSION

The Graphic Display System described above meets the requirements outlined at the beginning of this paper: Namely, low cost, efficient transmission codes, flexibility, and the ability to store and retrieve pictures for redisplay. In addition it is fully compatible with ASCII transmission codes and, therefore, can be substituted in place of any ASR33 or ASR35 Teletypewriter presently being used in conjunction with a time-sharing system.



GRAPHAS: A MODIFICATION OF GRAPHA

Karl Lowy and Sandra Furman
University of Rochester
Rochester, New York

The LINC-8 graph assembler Grapha is very useful for labeling of graphs and displaying data in curve and histogram form on the oscilloscope. It appears, however, inconvenient for oscilloscopic demonstration of an appropriately scaled coordinate system and display of data in less conventional form, e.g. those consisting of a set of discrete points.

Graphas incorporates the following modifications of Grapha:

1. It is available through LAP-6.

2. It permits a display of a coordinate system with individually variable scales for abscissa and ordinate.

3. If an extended memory is available, the program permits display of data contained in four consecutive blocks of tape according to the user's choice. The first two blocks contain the abscissae, the last two the ordinates of points to be individually displayed. Registers containing no entries are not displayed so that the solid zero line is absent when the graph is projected.

Graphas retains all features of the original program but was not written to include the plotting subroutine.

Scale, figure and labels can be displayed simultaneously as well as individually, a feature helpful for the photographic procedure.

As an example, figure 1 displays the input-output characteristics of a pulse-generator devised for use under computer control.[1] The linear range of the generator is well characterized and the tracing is seen to be free from a heavy and meaningless zero line.

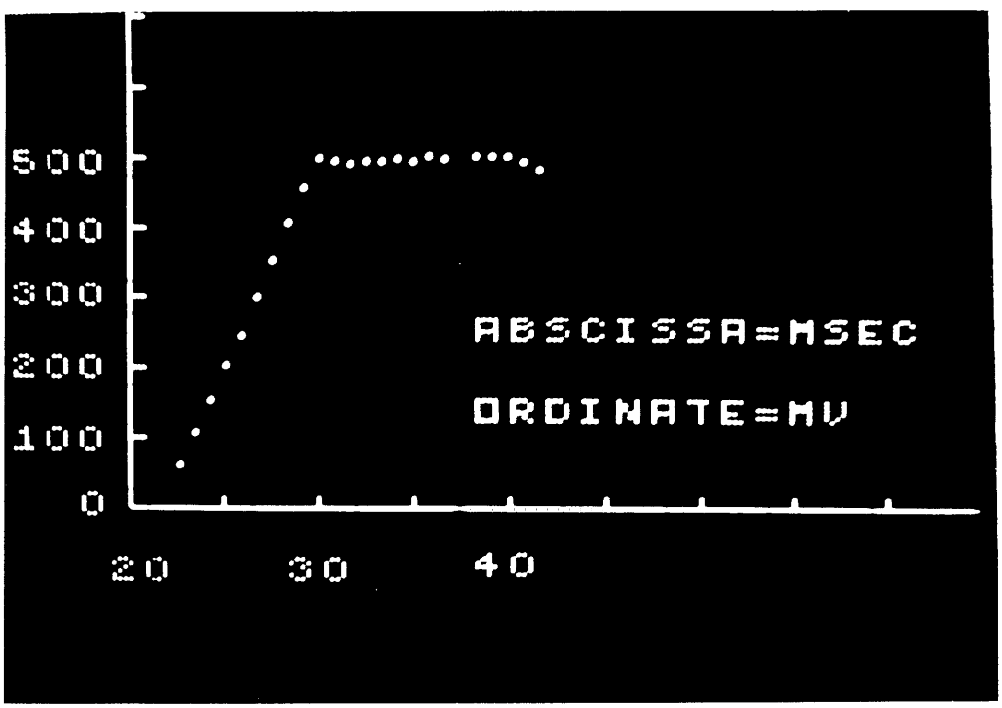
REFERENCES

[1] Karl Lowy and Lawrence Ota. Pulse generator with external amplitude control by computer. Electroenceph. clin. Neurophysiol., in press, 1968.

ACKNOWLEDGEMENTS

This work was supported in part by a Public Health Service program project grant, NB 06827.

This paper was scheduled to be presented at the Spring Symposium, however, circumstances arose preventing it from being presented and also was not received in time to be published in the Proceedings.



SUBMITTED BY DECISION SCIENCE, INC.

Decision Science, Inc. is offering for sale its SIMUL8S programming system consisting of an assembler for and a simulator of the PDP-8/S, PDP-8, PDP-8/I, and PDP-8/L, this available for the IBM 360/40 or larger, 7090, 7094; CDC 3600, 6400, 6600; Univac 1108 or GE 635.

Specifically, SIMUL8S permits programming utilizing cards rather than paper tape, thus simplifying development and programming changes. It provides additional debugging aids to the programmer in that program tracing under on/off timing control, time dumps, and core dumps are available during execution. This time analysis feature can be used to determine the execution time between two points in the program, this being useful to determine compatibility with real time problems.

Complete program development can be achieved by using SIMUL8S for diagnostics, assembling, debugging, executing the program, and/or providing a binary paper tape for direct read in by the PDP-8 series computer, thus eliminating source paper tape punching, assembly and debugging on the smaller computer. SIMUL8S permits software development prior to the installation of the PDP-8 series computer or while it is in use performing its dedicated function.

It can be used to simulate proposed systems for evaluation, thorough checkout of programs by simulating "worst case" conditions since it allows simulation of interface equipment and signals difficult to generate with hardware.

The input language to cards and the printed output includes the basic format and features of the PAL III assembler as well as the page linkage feature of MACRO-8.

For more information contact:

Dr. A. J. Owens, Vice President
Decision Science, Inc.
4508 Mission Bay Drive
San Diego, California 92109

Submitted By GRASON-STADLER COMPANY

SCAT (State Change Algorithm Terminology) is a conversational, real time, time sharing, process control system designed to run on a 4K or larger DEC (Digital Equipment Corporation) PDP-8 computer which is interfaced to a Grason-Stadler "Job Control Multiplexer."

SCAT software consists of two major parts. The first is the Executive and the second is the SCAT Language Processor. The Executive administers the use of the computer. It divides the processor time between the various experiments, or jobs, and the Teletype and other output devices. It allocates storage space for the various programs which are active. The SCAT Language Processor is itself divided into two sections--the SCAT translator and the SCAT interpreter. The SCAT translator accepts the SCAT statements which form the experimental description (the program), checks them for internal consistency and translates them into a code which the SCAT interpreter understands. The SCAT interpreter controls each experiment using the user generated program for guidance.

For more information contact:

Mr. Steven J. Stadler
Grason-Stadler Company
Box 2
West Concord, Massachusetts 01781

Submitted By INFOCOM, Inc.

INFOCOM, Inc. is offering for sale its SAIBOL Compiler. The compiler resembles a COBOL compiler, but was designed specifically for the PDP-8 Family of computers. Although the system has been especially designed for business oriented applications, such as Payroll, Job Costing, Accounts Receivable, Accounts Payable, General Ledger and Inventory Control, it can be equally effective for scientific programming as well.

SAIBOL is an interpretive compiler which allows any PDP-8 Family computer to function as a character-oriented machine. It is designed to reside in a 4K PDP-8 with optional DECtape, DECdisk, high speed paper tape reader and punch and an ASR-33 Teletype. It is core-resident, requiring about 3K of core. The remaining 1K is available for the user program statements.

Some important features include:

1. The system includes a Monitor which allows user programs to be stored on disk.

These programs may be called into core via keyboard commands.

The language includes six classes of commands:

a. Input/Output - for entry of data via keyboard and paper tape, and outputting data to printer and paper tape.

b. Arithmetic - for loading, storing, adding, subtracting, multiplying, and dividing quantities represented as BCD digit strings, and testing the result of arithmetic operations.

c. String manipulation - for moving and comparing strings of alphanumeric characters.

d. File - for filing and retrieving records of information on and from disk storage.

e. Monitor - for calling the Monitor, or requesting other programs on disk in the event of segmented programs.

f. PAL Language - Routines in the PDP-8 PAL III assembly language can be freely intermixed with SAIBOL statements.

The SAIBOL interpretive compiler is a core-resident interpreter which analyzes user statements stored in core. As each statement is analyzed, it is executed by SAIBOL, and the next statement is then accessed.

The SAIBOL interpreter is stored in the upper 23 pages of core. Page 0 is also used in part by the system for common storage and monitoring purposes.

The rest of core (locations 200-2177) is both word-addressable and character addressable. Characters are represented by six bits, and packed two to a word. The character in the high half of the work (bits 0-5) is defined as the first character, and the character in the low half of the work (bits 6-11) is defined as the second character. Thus, user core can be addressed as words 200-2177, or characters 400-4377.

2. Easy to Use and Learn - SAIBOL is conversational in mode through the Teletype. SAIBOL will tell you where you went wrong and how to fix it; a truly turnkey system.

3. Easy to Interface - SAIBOL will interface to nonstandard I/O devices, i.e., if you require upper and lower case printing, color change and a variable width carriage, then SAIBOL will let you talk to a Flexowriter.

4. Precision - Data Fields are accurate to 10 significant digits (ideal for accounting purposes).

5. Internal Power - Interpretive picture strings to direct data manipulation, editing and calculation.

Programs Available from INFOCOM, Inc.

1. Payroll - provides complete payroll accounting, from the maintenance of the basic employee payroll records to the preparation of checks, W-2 Forms and 941 Forms.

2. Job Cost Analysis - provides an easy means of distributing labor and material costs across the appropriate departments for each job, with a complete profit report including the preparation of invoices if desired.

3. Accounts Payable - provides complete accounts payable account from the maintenance of the Master Vendor File to the preparation of checks and the generation of a Check Register File.

4. Accounts Receivable - provides complete accounts receivable accounting for daily updating of the accounts receivable accounts, billing, credit analysis, and activity reporting.

5. Inventory Control - provides a complete inventory control accounting system from the recording of all inventory transactions to the preparation of a Cost of Sales Report.

6. General Ledger - provides for classification, summarization, and recording of the cumulative effects of all transactions on assets, liabilities, revenue and expenses.

7. Display and Classified Ad Billing (Typesetting) - provide a complete display and classified billing system. Information prepared on off-line keyboards is read via the high speed paper tape reader and is stored and manipulated as required.

8. Circulation Package (Typesetting) - provides an integrated draw and billing system for all editions of one or more newspapers. Inputs may be entered on-line, or prepared off-line on punched paper tape and entered via high speed reader.

INFOCOM's newly formed Machine Tool Products Division in Los Angeles is offering a complete line of programs for the preparation of tape for numerically controlled machine tools. NC programs may be used in conjunction with INFOCOM's business programs.

Programs Available for NC are:

1. Two and three axis point to point and two axis contouring programs for basic PDP-8/I or 8/L configurations.

2. Wire wrap program providing tapes for numerically controlled, hand aided, wire wrap systems. Program allows definition of panel layout and use of standard wire list information as input.

3. Full three axis APT language contouring software for expanded PDP-8/I and 8/L. Requires 8K core, 262 disk, high speed paper tape I/O.

4. APT-III implemented on PDP-10.

For more information contact:

Mr. William Landis
INFOCOM, Inc.
20 Walnut Street
Wellesley Hills, Massachusetts 02181

EXCERPT FROM A TYPICAL SAIBOL-8 PAYROLL PROGRAM WITH NO COMMENTS

```
NEXTRECORD, READ 215 WRONGTAPE

        HRS , ACCEPT HOURS HRS
              TAKING REGHOURS MULTIPLY RATE
              GIVING REGPAY 2 NOSIGN
              TAKE OVRHOURS MULTIPLY RATE MULTIPLY
              OVERTIME GIVING OVRPAY 2 NOSIGN
```

EXCERPT FROM A TYPICAL SAIBOL-8 PAYROLL DOCUMENTED WITH COMMENTS

```
NEXTRECORD, READ(EMPLOYEE'S PAY RECORD.  IF MORE
            THAN) 215 (CHARACTERS, GO TO) WRONGTAPE
       HRS, ACCEPT (REGULAR AND OVERTIME) HOURS (IF
            FORMAT ERROR, GO TO) HRS
            TAKE REGHOURS (AND) MULTIPLY (BY) RATE
            GIVING REGPAY (WITH) 2 (DIGITS TO LEFT OF
            DECIMAL POINT AND) NOSIGN
```

LETTERS

"Miss Margaret Noble
Software Quality Control
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

"Dear Miss Noble:

"In attempting to set up a system using the double precision floating point trigonometric subroutines, we have discovered the following errors.

"In the sine subroutine, there is an improper entry for C7. In the LINCtape manuscript, the decimal value of this number is correctly stated as -.004362476, but the DBLFLT value should be:

7770		7770
5610	not	5610
3161		2521

"The arctangent subroutine combines two subroutines in one jumble with nonsense as the result. To get a properly working arctangent subroutine, take the coding on page 17 of the DBLFLT write-up DEC-L8-FLAA-D and make the following changes:

(1) On line 3L+7, replace "ADD 4A+1" with "ADD 4F+1".

(2) On line 3M-3, replace "4K [$\pi/2$" with "4H [0.0".

(3) Eliminate lines 3M through 3M+3. They calculate the tangent, not the arctangent.

(4) Tag the line immediately following this deletion with #3M. The three three-place locations tagged 4A, 4B, and 4K may be eliminated since they are now unnecessary. The decimal value listed for C5 should also be corrected to read 0.1801410; the DBLFLT entry for this constant is correct as it stands.

"The write-up on the arctangent subroutine should read as follows:

"The arctangent of X (over the interval $0 \leq X < \infty$) may be calculated using the following approximation from Hastings:

$$\arctan X \doteq \pi/4 + C_1Y + C_3Y^3 + C_5Y^5 + C_7Y^7 + C_9Y^9$$

where

$$C_1 = 0.9998660$$

$$C_3 = -0.3302995$$

$$C_5 = 0.1801410$$

$$C_7 = -0.0851330$$

$$C_9 = 0.0208351$$

and $Y = \dfrac{X-1}{X+1}$.

"To use this subroutine, store the value of X in the three locations beginning at Tag 4F and jump to the subroutine at 3L. The subroutine returns to the main program with arctan X in the FAC.

"We would be most pleased if these corrections could be listed in the software news section of the next issue of DECUSCOPE.

Sincerely yours,

Paul F. Sullivan
Technical Support Group"
National Aeronautics and Space Administration
Electronics Research Center
Cambridge, Massachusetts

"Dear Angela:

"This is in reference to the problem that we have been having with the DECUS 8-124 card decks. Perhaps others may benefit from our experience.

"Most of our computers, keypunches, and auxilliary equipment use the old 48 character FORTRAN card code. The deck that you have supplied was coded in a 48 character subset of the newer EBCDIC (extended binary coded decimal interchange code) used by IBM on their 360 series and by some other manufacturers. This code differs from the old code in only a few characters and these are listed below.

Symbol	Old F Code	EBCDIC
+	12	12-6-8
=	3-8	6-8
(	12-4-8	12-5-8
)	0-4-8	11-5-8
'	4-8	5-8

Very truly yours,

Lawrence Leipuner"
Brookhaven National Laboratory
Associated Universities, Inc.
Upton, L.I., New York

WANTED

Information relative to interfacing a PDP-8 to a small 40 amplifier analog computer.

Contact: Dr. Martin
Mount Sinai Hospital
Cleveland, Ohio

"Dear Mrs. Cossette:

"This is a general request for information as to the existence of business programs written for the DEC family-of-eight computers. Of special interest are the type of programs which will retrieve information stored in character form on disc or tape, and do statistical studies, generate payroll checks, business reports, etc.

"In addition, does anyone know of a business-oriented compiler written for these machines which will allow a person to write his own business programs in a high level character-manipulating language?

Sincerely,

David A. Starr
Computer Department
LIBERTY INDUSTRIES CORP.
1057 East 3300 South
Salt Lake City, Utah 85106"

"Dear Mrs. Cossette,

"As a final year project in Electronics I hope to design and develop a simple CRT display for the College PDP-8. I will be using a conventional oscilloscope and a single cycle data break to generate a possible display with short vector and point modes, as used in the 338 system.

"I would be most interested to hear from anyone with a similar project in mind.

"Thank you,
Yours sincerely,

David Whiteley
Student Apprentice"
Staffordshire College of Technology
Beaconside, Stafford
ENGLAND

"Dear Mrs. Cossette:

"I would appreciate hearing from anyone who

a. has used the PDP-8 as a recirculating memory for spectral analysis purposes

b. and/or has had experience in interfacing a PDP-8 with an EAI-680 analog computer.

Yours sincerely,

Dr. Denis Poussart
Electrical Engineering Department
Faculty of Sciences
Laval University
Quebec 10, P.Q. CANADA"

PROGRAMS AVAILABLE FROM AUTHORS

COMPUTER - PDP-8/338

Title: Core Display Program

Author: A. M. Romaya, English Electric Computers Limited, Kidsgrove, Stoke-on-Trent, Staffordshire, England

This program allows the user to display, change, dump and punch the contents of any core location by commands initiated from the 338 display light pen and push buttons. The program occupies locations 5ØØØ-7340 of memory field one. The program does not set the push down pointer or the interrupt system and hence field Ø is absolutely free for use by other programs. It is possible to run this program concurrently with another which uses the interrupt system. (This second program should not use the display unless in special modified cases.)

Recently, the program has been extended to include a core transfer facility, Hence, it is possible to transfer a block with start address A and end address B to another part of memory with start address C. This transfer is from one field to another or within each field.

Minimum Hardware: 338 Display System, Character Generator, 8K store, High Speed Punch (optional)

Title: Drawing Applications Program

Author: A. M. Romaya, English Electric Computers Limited, Kidsgrove, Stoke-on-Trent, Staffordshire, England

This program is intended to show the facilities the DEC-338 Display System offers when considered as a drawing board.

The program allows the user to:

1. Draw straight lines or "freehand" over a total of 75 x 75 inch area.

2. Include symbols which may be formed by means of the program.

3. Label the drawing in alphanumeric and other characters.

4. Delete items drawn.

5. Output the display and symbol files created.

6. Input a display file and its symbols for updating.

The program incorporates a tracking cross and raster, and the coordinates of the tracking are shown when required. Control is obtained by a set of light buttons, push buttons and the switch register.

Possible core location changes for adapting the dimensions drawn to special cases are included in the write-up for the program.

Minimum Hardware: 338 Display System, Character Generator

PROGRAMMING NOTES

LINC-8 and PDP-8

To LINC and LINC-8 LAP6 Users:

Washington University's Computer System Laboratory issued a correction to LAP6 thereby creating LAP6A. LAP6-3L users should refer to SOFTWARE NEWS for instructions to modify LAP6-3L into LAP6A-3L.

If you do not use LAP6-3L and did not receive the notice from Washington University you may write to DECUS, Washington University, or exchange LAP6 master tapes for LAP6A tapes by mailing your tapes to Computer System Laboratory, 724 South Euclid Avenue, St. Louis, Missouri 63110.

PROGOFOP MODIFICATION TO USE THE TYPE PC01 HIGH SPEED READER/PUNCH

Glen W. Johnson
Institute of Oceanography
Dalhousie University
Halifax, N. S., Canada

A modification to TYPEIT of PROGOFOP appeared in DECUSCOPE, Vol. 7, No. 1 (1968). This modification has been tried here and it does not seem to work completely. When either RETURN or LINE FEED is typed, the computer reaches the RENINT subroutine without restarting the LINC and when no more interrupts are to come. Since we have an alternate method of punching binary tapes, we have not looked for the solution.

Our LINC-8 has a PDP-8 option, the Type PC01 high speed reader/punch. The following patch listing defines the operate commands OPR Ø and OPR 1. It requires all of the unused space in page zero of PROGOFOP except the auto-index registers.

OPR Ø: PUNCH the character in the LINC A register (bits 4 to 11) on the high speed punch. It does not wait for the punch to finish.

OPR 1: READ the next character from the high speed reader and wait until the character has been obtained. If the power was off, or there was no tape present (run out or not entered), the number 7777 is put in the LINC A register for program information. If a second command is given while no tape is present, the number 377 appears in the LINC A register.

The next LINC command after OPR1 should then be APO(i) to check the sign since the sign bit is normally zero.

If the power was off, turn the reader on and restart the PDP-8 at address 0200 (8) to initialize the reader. Then the LINC may be restarted as needed. If the tape was missing, correct the problem as needed and restart the LINC.

```
/PROGOFOP PATCH TO DEFINE OPR 0 =PUNCH ON HIGH SPEED PUNCH
/                         OPR 1 =READ TAPE ON HIGH SPEED RE
/IF TAPE IS NOT IN READER, LINC ACC IS SET TO -0
/IF A SECOND OPR 1 COMMAND IS GIVEN, 377 IS PUT IN LINC ACC

/GLEN JOHNSON, MAY 1, 1968
/DALHOUSIE UNIVERSITY, HALIFAX, NOVA SCOTIA, CANADA.


GOBACK=1460
OPRATE=1432
ACTOA=1457
IAAC=6171
IACA=6167

          *120
OPRDO,    TAD (JMP I OPSORT+13  /SET UP JMP
          DCA.+1
          0                     /WILL BE JMP I OPSORT + N

OPSORT,   DOPUN                 /OPR 0 -HIGH SPEED PUNCH
          DOHSRD                /OPR 1 -HIGH SPEED READER
          GOBACK
          GOBACK
          GOBACK
          GOBACK
          GOBACK
          GOBACK
          GOBACK
          GOBACK
          GOBACK
```

```
                        /PUNCH CHARACTER FROM LINC ACC.
DOPUN,    ISZ PUNCH               /WAS PUNCH IN USE ?
          JMP OKPUNCH             /NO
PWAIT,    PSF                     /WAIT FOR PUNCH
          JMP.-1                  /TO FINISH WITH
                                  /LAST CHARACTER
OKPUNCH,  IAAC                    /GET CHARACTER
          PLS                     /PUNCH IT, CLEAR FLAG
          CLA CMA
          DCA PUNCH               /SET PUNCH INDICATOR
          JMP GOBACK

                        /FROM PAGE 6-14 OF PDP-8 COURSE WORKBOOK
DOHSRD,   TAD .+2                 /SET UP ENDCHECK
          DCA ENDCK
          RFC                     /START GETTING CHARACTER
                                  /ALSO EQUAL -1764 FOR ABOVE
          RSF                     /IS FLAG = 1 ?
          SKP                     /NO
          JMP .+5                 /YES, GET CHARACTER
          ISZ ENDCK               /INDEX ENDCHECK, IS IT ZERO ?
          JMP .-4                 /NO,CHECK FLAG
          CLA CMA                 /SET ACC
          JMP I A                 /RETURN WITH 7777 IN LINC A REG
                        /ACC IS CLEAR DUE TO DCA ENDCK
          RRB                     /READ CHARACTER AND CLEAR FLAG
          JMP I A                 /PUT IT IN LINC A AND RESTART
ENDCK,    0
A,        ACTOA


PUNTST,   0
          PSF                     /SENSE FLAG
          JMP.+3                  /NONE. IT MAY BE WORKING, SO
                                  /LEAVE IT ALONE.
          PCF                     /CLEAR FLAG
          DCA PUNCH               /CLEAR PUNCH INDICATOR
          RRB                     /CLEAR READER FLAG
          CLA
          JMP I PUNTST            /ALL INTERRUPTS DONE
PUNCH,    0


          *230
          JMS PUNTST              /TEST HIGH SPEED PUNCH/READER

          *OPRATE
          JMP OPRDO               /GO TO USER SUBROUTINE
                                  /WITH N-13 IN ACCUMULATOR



/SYMBOL TABLE OUTPUT OF PASS 2


LG OKPUNC+0004

A       0164
ACTOA   1457
DOHSRD  0147
DOPUN   0136
ENDCK   0163
GOBACK  1460
IAAC    6171
IACA    6167
OKPUNC  0142
OPRATE  1432
OPRDO   0120
OPSORT  0123
PUNCH   0175
PUNTST  0165
PWAIT   0140
```

MODIFICATION TO RANDOM NUMBER GENERATOR DECUS No. 5/8-25

Lawrence L. Feth and Richard V. Wolf
Eye & Ear Hospital and University of Pittsburgh
Pittsburgh, Pennsylvania 15213

"The lack of user-defined functions or subroutines in the DEC-FORTRAN system imposes a severe limitation on FORTRAN programs for the LINC-8 (or PDP-8). Short, machine-language programs stored in the data area will work for simple problems; however, many desirable functions and subroutines are so long that this method fails. After studying the most recent listing of the FORTRAN Operating System (DEC-08-CFA3-LA, March 1967) we have decided that many useful functions can be added to the op-system by replacing a function that will not be used by the FORTRAN program in core. The trig functions, especially ATNF(X), are the most likely candidates for replacement.

"These functions are probably used less often than SQTF(X), EXPF(X), or LOGF(X), and no other routines depend upon them. The techniques for writing new functions are well illustrated in the March 1967 version of the listing which is heavily commented and easy to follow.

"As an example of a user-defined function we have modified P.T. Brady's RANDOM NUMBER GENERATOR (DECUS 5/8-25) so that it replaced ATNF(X). A listing of the modified subroutine is enclosed. Two changes were made in the subroutine. First, we have written a routine that uses X to initialize the random number generator, if X is different from zero. This X is not the first number generated by the routine. A programmer can initialize a random sequence by using RANF(X) with X $\neq$ 0, then continue the sequence from its initial value by using RANF(0.). The sequence will be reproduced every time a particular value of X is used to initialize it.

"For our purposes a positive random floating point fraction is more desirable than a random integer. This led to our second modification. The original subroutine leaves its result in three successive locations FRN1, FRN2, and FRN3. We have inserted location IRN, which contains zero, above FRN1 so that the floating point number returned to the stack by the interpreter is always a fraction. The sequence of operations from locations 4742 to 4745 makes sure that FRN1 is always positive.

"The function name can be changed from ATNF( ) to RANF ( ) by changing location 1605 in the Function Symbol Table from 0124 to 2201 before compiling the program. The name doesn't have to be changed if the programmer can always be sure that the correct routine will be in core when his FORTRAN program uses ATNF( ) to call the random number generator.

"User defined functions may be loaded using the binary loader after the operating system has loaded the FORTRAN object program and halted. The program is started by restarting the operating system at location 201. In our system, using the LINCtape FORTRAN system we load the user defined functions from mag. tape under program control.

```
                 / RANDOM NUMBER FUNCTION FOR USE WITH PDP-8 AND LINC-8
                 / FORTRAN OPERATING SYSTEM
                 / OCCUPIES ATNF[X] LOCATIONS IN OP. SYSTEM
                 *        4656
4656   4501               ENTER      / GO TO INTERPRETER = JMS I Z OLVI
4657   0051               SUBN       / ENTER SUBROUTINE SORT
4660   4663               SORT       / FROM INTERPRETER
4661   0007               ILVE       / EXIT FROM INTERPRETER
4662   5500               RETURN     / EXIT FUNCTION
4663   0051         SORT, SUBN       / SUBROUTINE TO PUT ARG.
4664   5154               SSSB       / INTO X AND BACK ON STACK
4665   0023               FIF        / FLOATING IF TEST:
4666   4671               NEW        / GO TO NEW IF X NEG.
4667   4675               OLD        / GO TO OLD IF X ZERO
4670   4671               NEW        / GO TO NEW IF X POS.
4671   5002          NEW, FLDS       / PUT X ON STACK
4672   0025               X,0025     /
       5402               FSTS       / STORE NO. ON STACK[X] IN
       4755               FRN1       / LOC FRN1 - INITALIZE RANF
       0007          OLD, ILVE       / LEAVE INTERPRETER
                 / BEGIN RANDOM NUMBER SUBROUTINE [ P. T. BRADY - DECUS #5-23
                 / WITH MODIFICATIONS TO OUTPUT POS. FRACTION AND PROVISIONS
                 / FOR INITIALIZATION USING NEW X
       1357               TAD FRN3
       7004               RAL
       0350               AND C7400
       3353               DCA TEMP3
       1356               TAD FRN2
       0347               AND C0177
       1353               TAD TEMP3
       7004               RAL
       7006               RTL
       7006               RTL
```

```
4710  1356      TAD FRN2
4711  3352      DCA TEMP2
4712  7430      SZL
4713  2351      ISZ TEMP1
4714  7000      NOP
4715  1355      TAD FRN1
4716  7104      CLL RAL
4717  7430      SZL
4720  2352      ISZ TEMP2
4721  7410      SKP
4722  2351      ISZ TEMP1
4723  7000      NOP
4724  7100      CLL
4725  1355      TAD FRN1
4726  3355      DCA FRN1
4727  1356      TAD FRN2
4730  7004      RAL
4731  7430      SZL
4732  2351      ISZ TEMP1
4733  7000      NOP
4734  7100      CLL
4735  1352      TAD TEMP2
4736  3356      DCA FRN2
4737  1355      TAD FRN1
4740  7004      RAL
4741  1355      TAD FRN1
4742  1351      TAD TEMP1
4743  7510      SPA
4744  7040      CMA
4745  3355      DCA FRN1
4746  5360      JMP EXIT
4747  0177      C0177,0177
4750  7400      C7400,7400
4751  0000      TEMP1,0
4752  0000      TEMP2,0
4753  0000      TEMP3,0
4754  0000      IRN,    0
4755  0000      FRN1,0
4756  0000      FRN2,0
4757  0000      FRN3,0
4760  4501      EXIT,4501     / RECALL INTERPRETER
4761  5002      FLDS          / PUT RANDOM FRACTION
4762  4754      IRN           / ON STACK
4763  0052      SUBL          / LEAVE SUBROUTINE
          ENTER=4501
          SUBN=0051
          ILVE=0007
          SSSB=5154
          FIF=0023
          FLDS=5002
          FSTS=5402
          SUBL=0052
          RETURN=5500
```

GUIDE CALL AND A NOTE ABOUT THE LINC-8 EVENT COUNTER

Glen W. Johnson
Institute of Oceanography
Dalhousie University
Halifax, Nova Scotia, Canada

GUIDE CALL can be made compatible with the LINC and can be arranged so that LAP6 can never lose the manuscript in its entirety.

Store the following program in a CLEARed memory, then DO: 0704 3400.

p	CONT	
1400	0055	SET i 15
1401	0701	RCG
1402	0056	SET i 16
1403	7200	7/200
1404	6015	jmp 15

If the user makes a habit of leaving LAP6 via EX, LO, or F commands, there will rarely be any problems. If LAP6 is left in some other way, all lines now in memory, but not yet on tape, will be lost when LAP6 is reentered. Unless the tape is damaged, these are the only lines that will be lost.

LAP6 was designed to wipe out the manuscript if it is started at 20 twice without an intervening "exit" command. It was also designed to retain the manuscript if it is started at 17 any number of times. Hence, it is wise to start at 17 when under program control.

In my LINC-8 EVENT COUNTER which appeared on pages 15 and 16 of DECUSCOPE Vol. 7, No. 3, I refer to OPR 0 and OPR 1 commands. These commands for using a high speed reader/punch are described in a DECUS Program with a LAP6 modification (LAP6-1H).

The essential patches to PROGOFOP for OPR i 2 from this program are as follows:

Change the following locations in BN 2

loc	cont		
120	1177	OPRDO, TAD (JMP I.+3+13)	
121	3122	DCA .+1	
122	0000	0	
123	1460	GOBACK*	/OPR 0
124	1460	GOBACK*	/OPR 1
125	2020	INCRMT	
126	1460	GOBACK*	
127	1460	GOBACK*	
130	1460	GOBACK*	
131	1460	GOBACK*	
132	1460	GOBACK*	
133	1460	GOBACK*	
134	1460	GOBACK*	
135	1460	GOBACK*	
.			
.			
.			
177	5536		

*These are available for other OPR commands: just insert the starting address (PDP-8) of the routine.

Change the following location in BN 5

loc	cont	
1432	5120	JMP OPRDO

GENERAL PURPOSE I/O PANEL FOR A PDP-9

Haruhisa Ishida
Denkitsushin University
and University of Tokyo
Tokyo, Japan

INTRODUCTION

One of the features of PDP computers is the ease of interfacing any user-designed I/O devices. A standard way of interfacing requires one or two I/O cables, a mounting panel, a device selector and some other circuit modules. While a large number of I/O devices can be interfaced with the use of an I/O interface logic arrangement such as one offered by DEC, it happens quite often that those I/O devices need not be connected to a mother computer on a permanent basis. This is especially true in a university environment, where researchers or students design and connect experimental devices, use them on-line only occasionally or just for some time and then quit.

It would be desirable to have some sort of general purpose I/O panels so that an arbitrary I/O device can interchangeably be hooked up easily and economically on the spot each time when the need arises. For this purpose, a general purpose I/O panel was designed and attached to a PDP-9 at the University of Tokyo.

A common device code of 37 is assigned to the panel. The terminal arrangement is shown in Figure 1. Available are 35 output signals and 21 input signals associated with IOT37xx instructions. The use of multi-terminal connectors was deliberately avoided, so that one can hook up any loose wires instantly without using expensive multi-terminal connectors. This is permitted within an anticipated frequency range. Care has been taken to insure that there is no direct connection from the panel to the PDP-9 I/O bus. The data channel facility can also be used with this panel and a W104 Bus Multiplexor, although it has not been implemented.

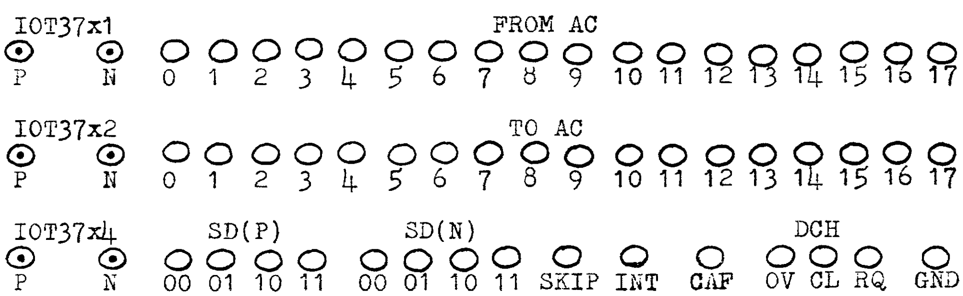


Figure 1 Terminal Arrangement

OUTPUT SIGNALS

IOT pulses (IOT37x1, IOT37x2 and IOT37x4) are likely to be used most often and they are available through six BNC connectors both as positive and negative pulses from a W103 device selector. Decoded subdevice signals in four combinations (SD00, SD01, SD10 and SD11) are also available, all as positive levels from a B155 binary decoder and as negative levels from R107 inverters.

Whenever an IOT37x4 instruction is executed, the 18-bit content of the accumulator appears at FROM-AC terminals 0 to 17 as buffered signals (a positive level for a "1") from W500 high impedance followers. Combining IOT37x4 pulse with subdevice signals, one can implement up to four "write" instructions, although only one write instruction will suffice in most applications. Terminals for a DCH I/O OVFLO signal and a DCH CLEAR FLAG signal are also provided for future use. A positive CAF (clear all flags) pulse from an R107 is included also.

INPUT SIGNALS

Internal connection to input signals is shown in Figure 2.

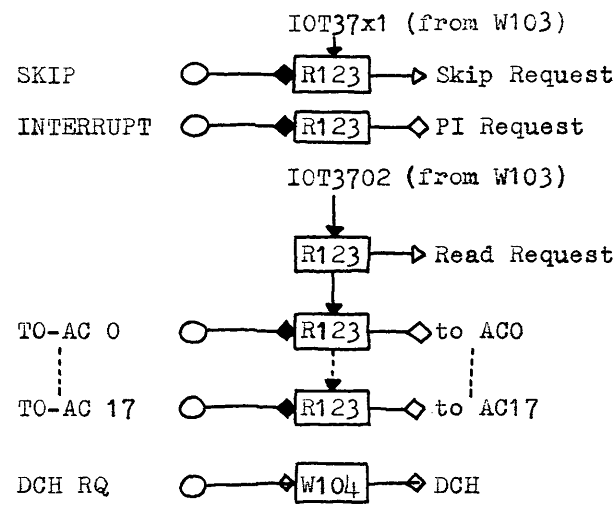


Figure 2 Input Signal Connection

An IOT37x1 instruction becomes a skip instruction when a device flag (a negative level) is connected to the SKIP terminal. When there are two, three or four device flags, one should connect to it an OR output of up to four AND outputs of each of the flags and each of SD00, SD01, SD10 and SD11. Then IOT3701, IOT3721, IOT3741 and IOT3761 become skip instructions. Similarly, flags can be connected to the INTERRUPT terminal to cause interrupts.

The IOT3702 instruction has been wired as a read instruction. Hence one can simply feed up to 18 negative levels to 18 TO-AC terminals 0 to 17 to input data into the accumulator. This one read instruction will suffice in most applications. A DCH RQ signal terminal is also provided.

A PDP-8/PAPER TAPE READER INTERFACE TO SIMULATE TELETYPE OPERATION

Daniel Parrish
D. Eugene Hokanson
Veterans Administration Hospital
and
Department of Medicine
University of Washington
Seattle, Washington

To increase speed and reliability of paper tape input, it was decided to interface a Tally Model R-30 to our system (PDP-8, DECtape and minidisc). The R-30 is a low cost, 30/sec. reader designed for asynchronous operation. The obvious selection of a device code would be 11, to take advantage of the choice between high and low speed tape subroutines which exist in most DEC-supplied software. This was not attractive in our case, however, because most of our applications programs have been written in a conversational mode.[1] Program calls, data locations and control constants are entered either by the operator via the keyboard or by paper tape. The latter method is used frequently for unattended batch processing. To be able to continue operating in both modes, the new reader was given the same device code as the TTY reader/keyboard.

Figure 1 illustrates how this was accomplished. A double-pole, double-throw switch (SW1) is connected to an R200 as shown. The R200 output (0) is connected to pin V of the Tally device selector. Output (1) is coupled by an external diode to pin K of module ME 17. This module is part of the AND gate of the TTY keyboard/reader device selector in the processor. With the switch in the TALLY position, -3V is present on the Tally gate and ground on the keyboard gate, thus disabling the keyboard. The reverse situation exists with the switch in TTY position.

The rest of the interface is straightforward. 10P-2 is stretched and power amplified by a W040 to activate the drive mechanism and move tape. 10P-2 also clears the flag and after 35ms, sets it. This establishes the reader speed at approximately 30 characters/sec. The flag is also cleared by an end-of-tape switch which prevents further reading. The flag is ANDed with 10P-1 to the computer skip bus. 10P-4 is ANDed with itself to strobe characters into the AC input mixer through the hole sensing switches of the reader. This was done to hold the AC input at -3V during switch transition.

With this arrangement, operation of the Tally reader exactly duplicates that of the TTY reader.

KSF	6031	Skip on flag
KCC	6032	Clear AC and reader flag
KRS	6034	Read static
KRB	6036	Clear AC, read buffer (switches) Clear reader flag

The only difference is in setting the reader flag. This is accomplished by a delayed 10P-2 rather than an end of transmission pulse from the TTY buffer.

The system has proved to be very stable in performance and has provided many hours of error-free operation.

[1] DECUSCOPE, 7, No. 3, 1968.

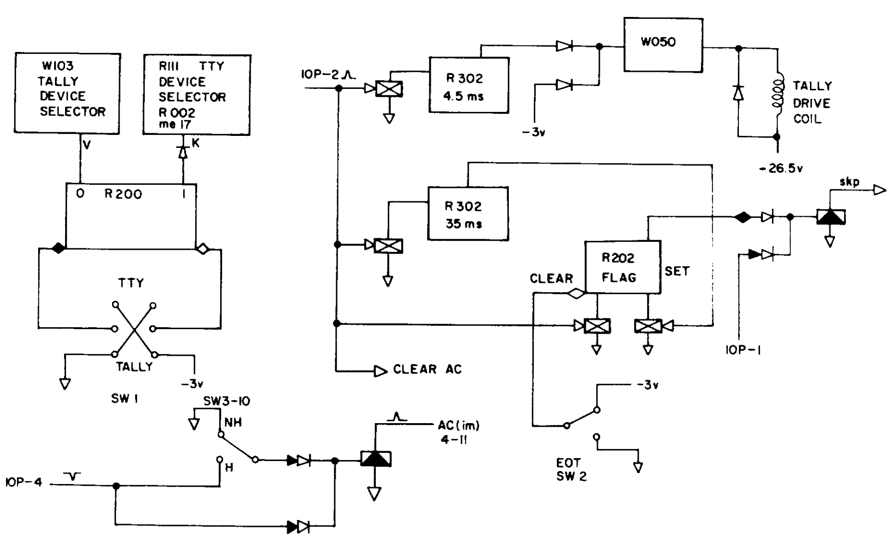


Figure 1

PDP-8/Tally Model R-30 interface. SW 3-10 are the hole-sensing switches of the Tally Reader. SW-2 is an End of Tape sensing switch.

1969 - Volume 8, Number 2

DECUSCOPE

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY

CONTENTS

	PAGE
DECUS Notes	1
Nuclear and Reactor Physics Experiments With an On-Line Computer	2-7
A Fast Approximation Method For Finding Logarithms	8-12
PDP-5 and PDP-8 Hardware Differences Affecting Software Interchange	13
Interface Between SR400 Card Reader and PDP-8	13
Point By Point Plot With LINC-8	14
PDP-9 Programming Note	15
PDP-10 Programming Note	15
PDP-8 Programming Notes	
HLT Everywhere Except the Loaders	16
Changes to DECUS 8-64 and 8-87	16
FOCAL Notes	16,17
FOCAL Points	17,18
Software for Sale or Lease	18
Wanted Note	18
Letters	19
News from DEC	20

DECUSCOPE HAS BEEN PUBLISHED SINCE APRIL 1962 AND IS THE OFFICIAL NEWSLETTER FOR DIGITAL EQUIPMENT COMPUTER USERS SOCIETY.

IT IS PUBLISHED PERIODICALLY AT THE DECUS OFFICE, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.

TELEPHONE: AC 617, 897-5111, EXT, 2414

EDITOR: MRS. ANGELA J. COSSETTE, DECUS

CIRCULATION: 4,600 COPIES PER ISSUE

DECUS NOTES

Watch for the Following DECUS Publications

The following DECUS publications will be available shortly. The automatic distribution of each is indicated. Those not included in automatic distribution may request material directly from the DECUS Office.

1. PROGRAM LIBRARY CATALOG ADDENDUM

   Distribution: All DECUS Members

2. PROCEEDINGS OF FALL 1968 SYMPOSIUM

   Distribution: Speakers and Meeting Attendees
   Charge: $5.00 per issue - Individual Members and Non-Members

3. PROCEEDINGS OF EUROPEAN 1968 SYMPOSIUM

   Distribution: Speakers and Meeting Attendees
   Charge: See No. 2

4. PROCEEDINGS OF CANADIAN 1969 SYMPOSIUM

   Distribution: Speakers and Meeting Attendees
   Charge: See No. 2

NUCLEAR AND REACTOR PHYSICS EXPERIMENTS WITH AN ON-LINE COMPUTER*

B. J. Allen and J. R. Bird
Physics Division, Australian Atomic Energy Commission

Introduction

Much of our knowledge of the properties and interactions of atomic nuclei has come from the measurement of the energy of nuclear radiations. By absorbing alphas, betas, gammas, neutrons, etcetera, in various types of detectors the energy of the incident radiation is converted to a voltage pulse. The height of the pulse if proportional to the energy deposited in the detector and is measured by determining in which of a number of height intervals or "channels" the pulse falls. Such measurements occur in reactor physics studies and other fields of application.

The instrument which performs this operation is called a Multichannel Pulse Height Analyser. By converting the incoming pulse height to a digital code, the analyser obtains a spectrum which relates the energy of the radiation with its frequency of occurrence. The spectrum is normally displayed on an oscilloscope—counts versus pulse height—and a record is printed.

Data handling was not a significant problem in the early pulse height analysers. With 100 or so channels of information, the time taken to examine and interpret data was very much less than the time required to collect it. However, this is often not the case today.

The most sophisticated analysers now offer up to 4,000 data channels, with complete input/output facilities for fast data transfer via paper tape, cards or magnetic tape. Dual pulse height to digital converters enable two-parameter analysis, and different data display modes are available on an oscilloscope. A limited degree of data reduction is provided to allow subtraction and integration of data. These operations are, of course, wired in and selected by switch and are, therefore, not readily modified or extended.

Simultaneously with the advent of the larger analyser systems has been the development and application of small digital computers to the general data taking problem. By coupling pulse height converter and oscilloscope to a central processor and memory bank, a computer can be used as an analyser with versatility and potency beyond that of the most sophisticated "fixed wire" multichannel analyser.

A control program permits peripheral units attached to the computer to be used for a particular application (which need not be pulse height analysis). A change of application requires a change of program only, not internal rewiring. In addition, the computer can readily control external devices.

The diversity of neutron and reactor experiments conducted by the A.A.E.C. Research Establishment has led to varied electronic and data taking requirements. Emphasis therefore has been placed on obtaining a data handling system with maximum versatility. Such a system is the PDP-7 Programmed Data Processor now operating in the Physics Division (Figure 1). Although this article refers specifically to the installation in the Physics Division at Lucas Heights, the comments are applicable to "on-line" computer systems in general.

The Central Processor

The heart of a computer is the central processor which performs logical and arithmetic functions and controls the transfer of data between peripheral units and memory. The present PDP-7 configuration has 8,192 (8K) storage locations in its memory bank, each location containing a "word" or eighteen binary digits (bits). Storage of any number from zero to 262,143 is then possible.

The processor (Figure 2) takes one machine cycle (1.75×10^{-6} sec) to address a location, write its contents into the accumulator, and rewrite the data back into the original location.

Method of Operation

In each time cycle, the processor operates through several control states set by the major state generator. In the FETCH state, the contents of the location specified by the program counter are placed in the memory buffer register. This word consists of a four bit instruction part which is deposited in the instruction register, and a 13 bit address, deposited in the memory address register. The contents of the program counter are then incremented by one in preparation for the next cycle.

The EXECUTE state follows with the loading of the contents of the location specified by the address register into the memory buffer where it is operated on according to the instruction register command. Thus a series of instructions can be followed by storing them in sequential memory locations and setting the program counter to the address of the first instruction. This instruction list then constitutes a program. Programs may be written in SYMBOLIC and FORTRAN II languages, and programming aids are available for editing, assembling and debugging operations.

Coupling the Computer to Experiments

An essential feature of the PDP-7 which makes it applicable to experimental work is its on-line operation through three interrupt modes. Program, priority and data break interrupts are available and operate when the processor senses a signal from a peripheral unit. The occurrence of such a signal indicates that the unit is ready for data exchange with the processor.

Under program interrupt control the ready signal causes a jump from the current operating program to an interrupt routine designed to test and cater for the signalling unit.

Because an interrupt may take some 30-100μsec to clear away, it is often important to ensure that certain peripheral units are not missed. This may be achieved by use of a priority interrupt which allows levels of importance to be assigned to the different peripheral devices. All signals are then remembered and dealt with on a priority basis—even to the extent of interrupts within interrupts.

*Reprinted with permission of the Director, A.A.E.C. Research Establishment, Lucas Heights, N.S.W. Australia.

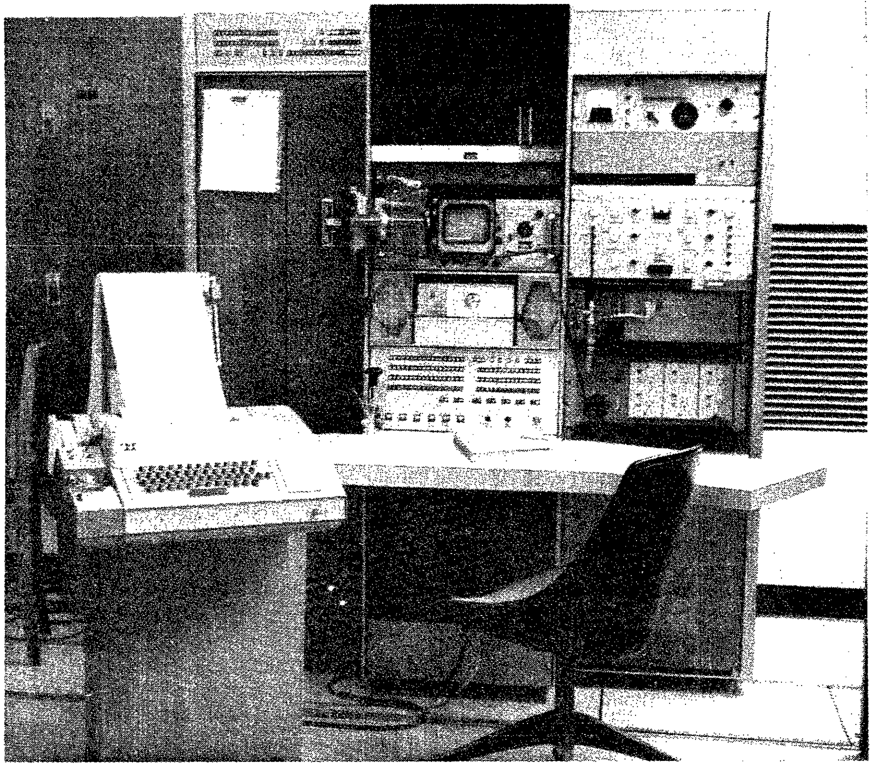


Figure 1. PDP-7 on-line computer.

Data break is a much faster operation. The processor is held for one cycle while data is transferred directly to a specified address. This interrupt mode provides a data transfer rate of up to 570,000 words/sec.

Typical peripheral units are the pulse height converter, oscilloscope, typewriter and paper tape unit. In each case, data transfer occurs through the most suitable of the interrupts discussed above.

The Physics Division system under current development is illustrated in Figure 3.

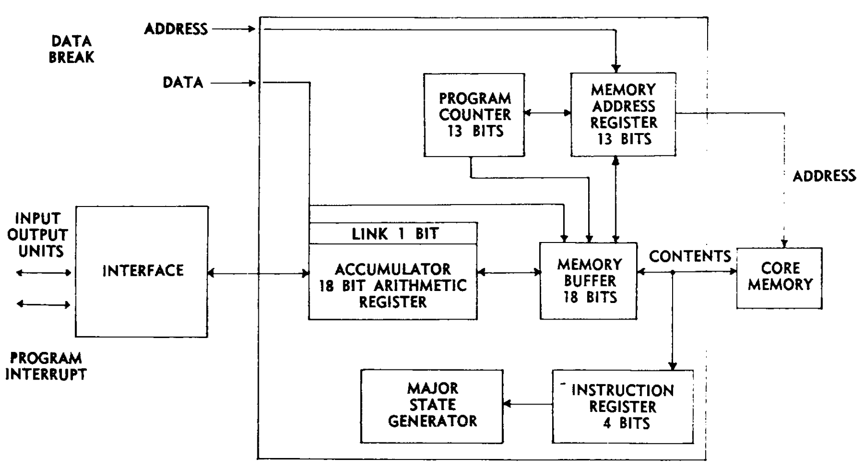


Figure 2. The memory address register enables a core location to exchange information with the accumulator through the memory buffer in 1.75 x 10^{-6} seconds.

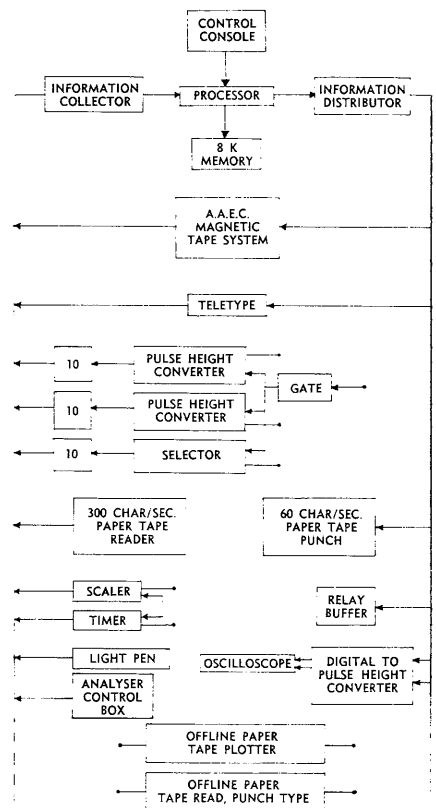


Figure 3. Binary coded data are received through the information collector from the various peripheral units, each identified by a code. Information is transferred through the distributor to a particular unit by calling its code.

Communication

Of primary importance is the ability of the experimentalist to determine the status of his experiment. He must know if his equipment is functioning as intended and if his system is optimised to handle and interpret the incoming data. Relevant regions of interest need to be selected for attention, and resolution and statistics examined. With this information at his fingertips he is in a position to make the best use of facilities in "real time" (the somewhat expensive time during which accelerator, electronics, technicians and data systems are employed).

The most effective method of communication between the experiment and physicist is via the visual display of memory contents on the oscilloscope. A light pen provides a suitable means of response for the experimentalist. Consisting of a shaped light pipe and fast photomultiplier the pen registers an oscilloscope display point and causes the computer to remember the displayed location. The program in operation can then assign this coordinate a higher brightness level for display and undertake any of a multitude of operations as required.

With this technique, simple programming is adequate with much of the decision-making left to the physicist. Nevertheless, the FORTRAN capabilities of the PDP-7 permit relatively complex programming. Theoretical calculations of angular distributions have been made and results displayed on the oscilloscope for direct comparison with reduced experimental data.

EXPERIMENTS

The versatility of the on-line computer is perhaps best illustrated by a survey of some of the current and proposed experiments in the Physics Division. These fall into four catagories—single parameter, multiparameter, multiscaling and correlation experiments.

Single Parameter Pulse Height Analysis

A single parameter experiment involves the measurement of the frequency of observation of one quantity only, e.g., the energy deposited by nuclear radiation in a detector.

A control program, written in symbolic language, allows the PDP-7 to be used in a manner similar to a sophisticated multichannel analyser. The program utilises the program interrupt control to allow data transfer between the processor and peripherals.

An incoming voltage pulse is digitised by the pulse height converter into 1,024 channels (i.e., a 10 bit binary number). Simultaneously with the loading of the binary number into the converter buffer, a ready signal interrupts the operating program. A subroutine determines the source and nature of the interrupt and initiates the parallel transfer of information from the buffer into the accumulator. The contents of the location corresponding to the digitised pulse height is then incremented by one.

The present program accumulates data in regions of 1,024 channels each. Any region can be displayed on the oscilloscope and areas of interest expanded for clarity. Arithmetic manipulations of data may be performed as required.

An important application of this single parameter program is in gamma-ray spectroscopy employing high resolution lithium drifted germanium detectors. For a resolution of 15 keV and with a minimum of three channels to define a peak, the 1,024 channels provide an energy range of 5 MeV.

Figure 4 illustrates a typical 1,024 channel spectrum taken with the germanium detector. The number of gamma-ray versus energy is shown for the Al (p,γ) reaction. The highest energy peak corresponds to an 11.13 MeV gamma-ray. The useful information in such a spectrum is contained in the areas of peaks above the continuous background. A program is available for listing the positions and areas of all peaks which are located by a simple search routine. An alternative method involves the use of a pen to select the region of interest and a FORTRAN II program to compute a least squares background fit under the peak. A subtraction then yields the peak area.

It is frequently useful to take data simultaneously in a number of detector systems. Backgrounds then can be run during an experiment, subtracting from (instead of adding to) the accumulated spectrum. Coincidence and angular distribution systems can have detector outputs recorded in separate sections of memory by tagging, in coincidence, incoming information with a base address from the selector buffer.

Multiparameter Analysis

A multiparameter pulse height analysis program allows experiments with a number of variable parameters to be performed. For example, a dual parameter experiment is the investigation of the (n,γ) reaction where it is of interest to measure both neutron and gamma energies and the number of simultaneous occurrences of these variables $N(E_n, E_\gamma)$. Other applications are the measurement of delayed gamma emission after fission (E_γ, t) and gamma-ray coincidence experiments.

Running the dual pulse height converters in coincidence in the two parameter mode, two conversion channels are provided which digitise the incoming pulse heights to form simultaneously two 10 bit words. These words are combined in predetermined portions to load the buffer with one 12 bit word (Figure 5). The program, on interrupt, increments by one the corresponding address in memory. This location represents the simultaneous values of x and y and is then a co-ordinate in the (x,y) matrix. As the program occupies approximately 1,500 memory locations, the full 4,096 channel data matrix is available only with an 8K store.

Several modes of display of incoming and accumulated data are programmed for the oscilloscope. Figure 6 illustrates the usefulness of these different modes in depicting the (n,γ) reaction. The x axis corresponds to gamma spectra from different neutron time of flight groups (y axis). The isometric display (A) simulates a three dimensional representation of E_n, E_γ, and $N(E_n, E_\gamma)$. By setting upper and lower limits on the number of events per channel, a contour is obtained of those co-ordinates for which N lies in the specified range (D). Markers are positioned and slices along these permit the observation of single parameter spectra (B and C). Because all display modes operate under program interrupt control, it is possible to examine spectra while still accumulating data.

Real time analysis of the (n,γ) data may proceed as follows.

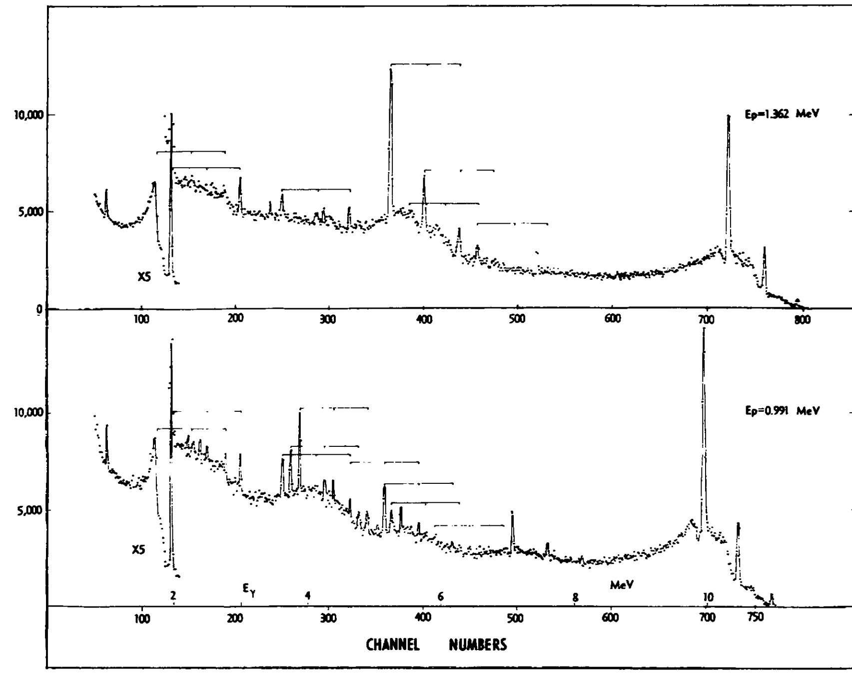


Figure 4. Germanium spectra of gamma-rays from proton capture in aluminium.

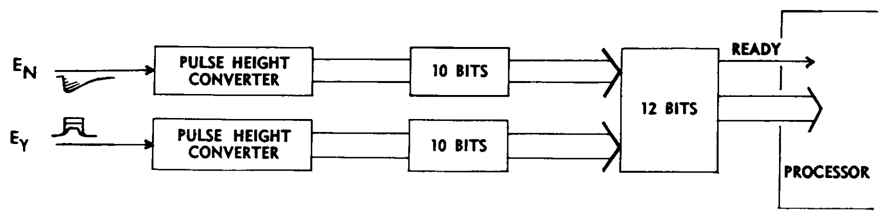


Figure 5. Dual parameter analysis.

The gamma spectra corresponding to events between peaks in the time of flight spectrum are the backgrounds as observed by the gamma spectrometer. Subtraction of these data (after normalisation of neutron flux) from that obtained with "on resonance" events leaves the actual resonance capture gamma spectra. The different neutron resonances may have varied properties which result in dissimilar gamma cascades. Consequently, it is important to distinguish between these events. The ability to manipulate data on-line in this way allows a more accurate representation of the physics involved.

A 16 x 256 data matrix may be used, i.e., 16 gamma spectra of 256 channels each. However, the choice depends on the relative complexities of neutron resonances and gamma cascades.

The obvious advantage of the multiparameter system in this case is the ability to take a number of spectra—including backgrounds—simultaneously. With the relatively low count rate in an (n, γ) experiment this is no meagre saving of real time usage. In addition, electronic drifts are minimised and as in general they would be common to all spectra, corrections can be made readily.

Data Reduction Before Storage

The parameters involved in some experiments may not be independent functions. When this occurs, it is possible to gain accurate results without mammoth storage requirements.

The identification of charged particles by measurement of their rate of energy loss ($\frac{dE}{dx}$) and total energy (E) is a suitable example. To a first approximation of $E \cdot \frac{dE}{dx}$ is a constant and proportional to the mass of the particle.

Now to measure both quantities to better than one percent, a 128 x 128 data matrix is required which far exceeds the available core storage (apart from the storage which may be required for a third parameter).

It is necessary, then, to operate on coincident incoming E and $\frac{dE}{dx}$ values to identify the responsible particle. Then only a single parameter result for each particle need be stored. High speed calculations of this type (i.e., identification of two variables) can often be achieved within the pulse height converter dead time and therefore need not affect the experiment dead time significantly.

Multiscaling

Instead of recording the relative heights of incoming voltage pulses, multiscaling records the number of pulses arriving in time interval Δt at time t_i. A memory address represents t_i and its content is the number of events counted in the time interval.

The pulsed neutron experiments with the Van de Graaff utilise the multiscaling technique. The rate of decay of the neutron flux, incident on a BeO cube, is observed by scaling the output of a neutron detector during sequential time intervals after the pulse. Statistics accumulate by pulse repetition and recycling of time channels.

The PDP-7 configuration shown in Figure 7 requires the use of a fast scaler gated by a crystal controlled timer. On completion of the time interval Δt, the timer stops the scaler and initiates a data break transfer of scaler buffer contents to the address specified by the timer buffer. This operation takes 1.75µsec.

Because the data break overwrites the location contents, it is necessary, on completion of a time cycle, to add this time spectrum to a permanent spectrum held in another area of memory. This addition is achieved by program and requires approximately (15 x N) µsec where N is the number of time channels. At a pulse rate of 1kc/sec, 60 time channels could be catered for per cycle.

Correlations

Neutron flux in a reactor may be studied as a function of space and/or time by correlation techniques. Neutron sensitive ionisation chambers situated in the reactor provide a D.C. voltage which depends on the incident neutron flux. Autocorrelation requires one detector, whose output at time t is compared with the output at all subsequent time intervals. Cross correlations compare the time dependent output of two detectors spatially separated in the core.

Figure 8 illustrates the operation of a cross correlation system. A gate generator triggers the dual pulse height converters which sample the D.C. levels of the two channels simultaneously to provide two six bit words in the 12 bit converter buffer. Increased accuracy can be obtained by alternate converter sampling employing the detector selector buffer to distinguish between the detector outputs. This would permit a maximum of ten bit accuracy in cross correlation work, equal to that normally achieved for autocorrelations.

The contents of the buffer are then loaded into sequential memory locations by a modification of the data break interrupt.

An alternative technique is to deposit data directly on to magnetic tape. At a writing speed of approximately 170 kc/sec, the dead time is of the same order and in series with the pulse height conversion time. Sampling rates of up to 10 kc/sec and relatively unlimited data storage are possible.

On completion of the experiments, the tape is run back through the computer for analysis.

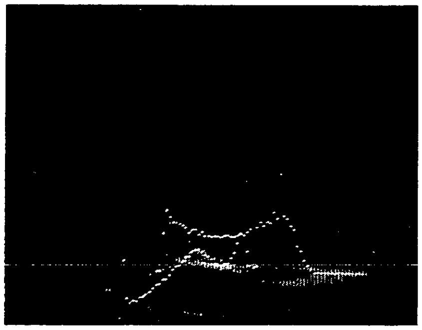
Experiment Control

Computers are frequently used for process control in circumstances where control requirements are well established. In research this is seldom the case, although certain features of an experiment may be suitable for automation.

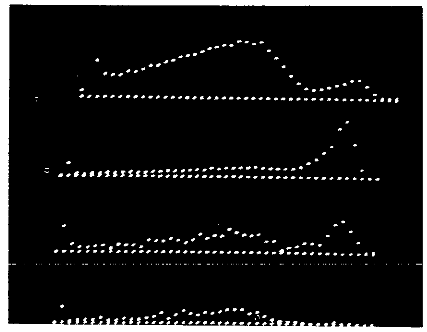
For example, operations such as the sequencing of sets of measurements, including statistical tests of data, changing detector positions (particularly in high radiation and contamination areas) are suitable for computer control if an extensive series of measurements is to be made.

Summary

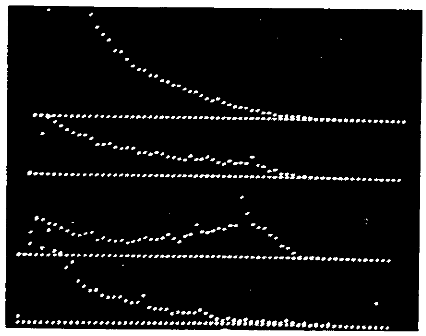
A data processing system such as that described offers superior flexibility in terms of cost and equipment when compared with the sophisticated pulse height analyser. The application of the on-line computer provides an opportunity to develop new and more effective methods in the conduct of research, a fact which is being exploited widely in many fields.



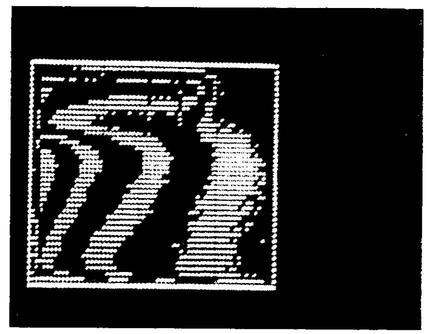
(A) *Three-dimensional display of number of events N(Eγ, t) (y axis) as a function of time (t) (offset y axis) and energy (Eγ) (x axis).*



(B) *Time spectrum-gamma-rays due to the Li(p,γ) reaction arrive first (R.H. peak) followed by Cd(n,γ) events in a cadmium target (L.H. peak).*



(C) *Gamma spectra for selected time intervals illustrate the change in spectral shape between the two time peaks.*



(D) *Contour display of co-ordinates (Eγ, t) with contents within a series of count intervals.*

Figure 6. Dual parameter data matrix showing correlation between arrival time of gamma-rays and gamma-ray energy.

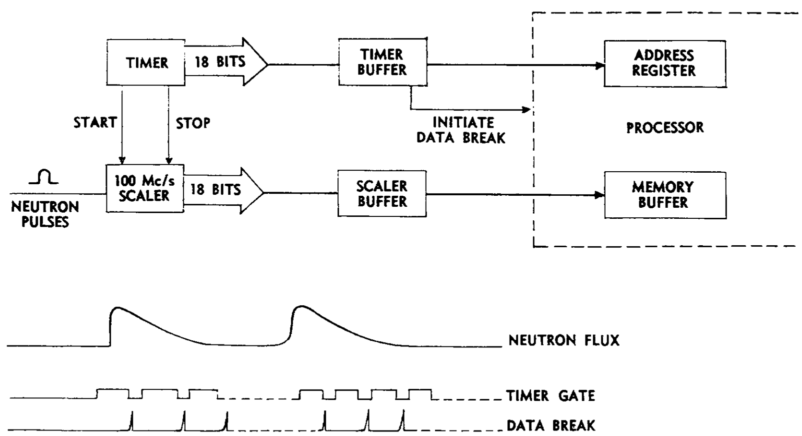


Figure 7. Multiscaling system.

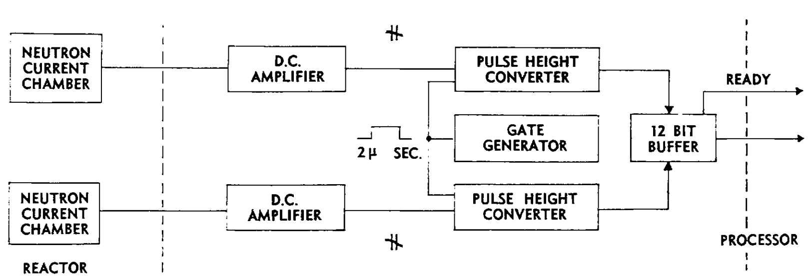


Figure 8. Cross correlation system.

A FAST APPROXIMATION METHOD FOR FINDING LOGARITHMS*

P. D. Siemens
Lawrence Radiation Laboratory, University of California
Livermore, California

In a specific problem of generating a fast log display on a small computer, the need arose to develop a fast algorithm whose error would be less than 10%. To achieve high speed, a linear interpolation method was chosen.

Suppose you wish to find $y_1 = \log_b X$. Assume you have a table values of $\log_b X$ whose y values are consecutive integers, and therefore, whose x values are consecutive integer powers of b.

Consider the diagram shown in Figure 1.

Because of the way we arranged the table,

$$L = n+1-1 = 1$$

$$D = b^{n+1}-b^n = b^n(b-1)$$

$$F = X-b^n$$

$$y_2 = n+Q.$$

By ratios of similar triangles,

$$\frac{Q}{L}=\frac{F}{D}$$

$$\frac{Q}{1}=\frac{X-b^n}{b^n(b-1)}$$

Therefore, $\log_b X \cong y_2 = n+\dfrac{X-b^n}{b^n(b-1)}$.

The error involved in this approximation is

$$E = y_1-y_2 = \log_b X - n - \frac{X}{b^n(b-1)} + \frac{1}{b-1} .$$

To find the maximum error, take the derivative with respect to X and set it equal to zero,

$$\frac{dE}{dX}=\frac{\log_b e}{X} - \frac{1}{b^n(b-1)} = 0.$$

Solving for the X at which maximum error occurs, you get

$$X_{max} = b^n(b-1)\log_b e.$$

Substituting this value into the equation for E, the maximum error is found to be

$$E_{max} = \log_b(b-1) + \log_b \log_b e + \frac{1}{b-1} - \log_b e.$$

Note that for a given b, the maximum error is constant. In particular, for b = 2, the maximum error is 8.6%. Choosing b = 2 seems to be logical since we are working with a binary machine; thus the approximation becomes

$$\log_2 X \cong n + \frac{X-2^n}{2^n} \text{ where } 2^n \quad X \quad 2^{n+1} .$$

Since $2^n \leq X < 2^{n+1}$, the operation $X - 2^n$ is equivalent to setting the most significant '1' in X to a '0'. Also, the division by 2^n is merely n left shifts.

Assuming the machine has a word length of k and that the numbers for which one wishes to find the log are double precision integers, the double precision integer is then 2k bits long. You can now define a normalize operation as one which

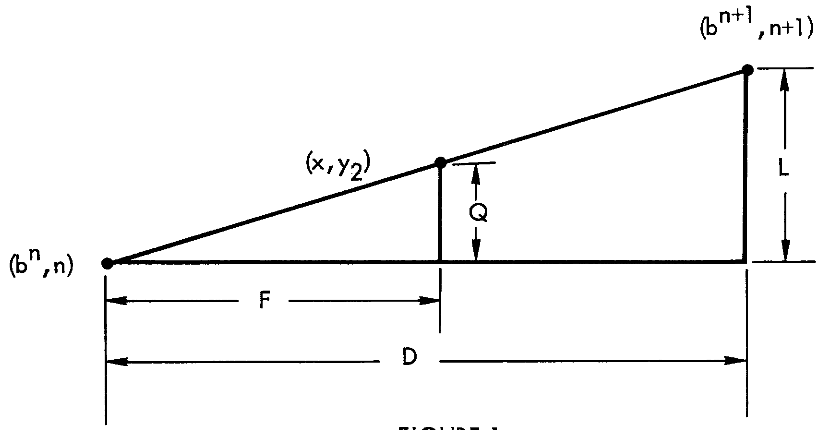


FIGURE 1

*This work was performed under the auspices of the U.S. Atomic Energy Commission.

shifts the double precision word left until the two most significant bits do not agree. For example, to normalize the integer 1 requires 2k-2 left shifts.

To find the log of a double precision integer X, you first normalize X with Q left shifts. Then n = 2k - 2 - Q (remember that $2^n \leq X < 2^{n+1}$). What has been done so far is to find the bit position of the most significant '1' in X, and by normalizing, maximum significance is held.

For convenience, imagine that the double precision integer X is represented in a floating point format M, E, where M is an integer mantissa and E is a power-of-two multiplier of that mantissa. Before normalization, you have X, 0. After normalization, you have X_1, -Q. To perform the subtraction $X - 2^n$ you just want to get rid of the most significant '1' in X. This can be accomplished by two left shifts of the normalized number X_1. In other words, the most significant '1' is shifted out the left end of the word and lost. After these last two shifts, you have in floating point representation, X_2, -(Q+2).

Now divide X_2 by 2^n where 2^n = 1, n in floating point representation. Then

$$\frac{X_2, -(Q+2)}{1, n} = X_2, -(Q+2)-n.$$

But, it was mentioned earlier that n = (2k-2)-Q. Therefore, the floating point representation of the result is X_2, -(Q+2)-n = X_2, -2k. This is equivalent to stating that now you have a double precision fraction whose binary point is at the extreme left of the word. To get $\log_2 X$, all that remains is to add the integer n to the fraction X_2.

In summary, one finds $\log_2 X$ by normalizing X with Q left shifts, so that n = (2k-2) - Q. Shift the normalized number two more left to shift the most significant '1' out the left end, and to the remaining fraction add n, the characteristic, remembering to align the binary points.

An algorithm for the method just described adapts very well to a basic PDP-8 (or PDP-8/S, PDP-8/I). This routine requires 37 storage locations and requires a maximum execution time of 610 µsec on the PDP-8 and an average execution time of 330 µsec (see Figure 2).

Taking advantage of the EAE option improves these figures considerably. This routine requires 18 storage locations and takes a maximum of 40 µsec and an average of 34 µsec to execute on a PDP-8 (see Figure 3). For a PDP-8/I, these times become 32.5 µsec maximum, and 30 µsec average.

The specific problem for which this log routine was developed was a real-time display of pulse-height analysis data. A flicker-free display of up to 1024 data points was needed. There was not enough core storage available to store the calculated logs in a display buffer--the logs had to be calculated as each point was plotted. In addition, decade lines had to be plotted to provide a display of numbers ranging from 0 to 10^6.

Figure 4 shows a subroutine that provides the required display. By using an interlace technique in plotting the data points, only a slight flicker is apparent when displaying 1024 points.

FIGURE 2

```
/ LOG ROUTINE FOR A BASIC PDP-8
/ FIND LOG BASE 2 OF DOUBLE PRECISION INTEGER
/ ASSUME THIS DP INTEGER IS STORED IN MS AND LS
/ THE BINARY POINT OF THE RESULT IS BETWEEN BIT 4 AND BIT 5
.
.
.
        TAD MS          / CHECK FOR ZERO
        SZA CLA
        JMP L2
        TAD LS
        SNA CLA
        JMP OUT         / INTEGER IS ZERO—SET LOG TO ZERO
L2,     DCA ENUM        / CLEAR STEP COUNTER
A,      TAD MS          / NORMALIZE
        RAL
        SPA CLA         / FINISHED
        JMP DNL         / YES
        TAD LS
        CLL RAL
        DCA LS
        TAD MS
        RAL
        DCA MS
        ISZ ENUM        / INCREMENT STEP COUNTER
        JMP A
```

```
DNL,     TAD ENUM          / SUBTRACT FROM 22
         CIA
         TAD N26
         DCA ENUM
         TAD MS            / SHIFT MOST SIG. BIT OUT
         RTL
         AND N7740
         TAD ENUM
         RTR
         RTR
         RTR
OUT,     DCA LOG
.
.
.
MS,      0
LS,      0
ENUM,    0
N26,     26
N7740,   7740
LOG,     0
```

FIGURE 3

```
/ LOG ROUTINE FOR PDP-8'S WITH EAE


         TAD LS
         MQL
         TAD MS
         NMI               / NORMALIZE
         RTL               / SHIFT BIT OUT LEFT END
         MQL               / STORE MANTISSA IN MQ
         SCA               / GET STEP COUNTER
         SNA               / CHECK IF ZERO
         JMP BZ
         CIA               / SUBTRACT FROM 22
         TAD N26
         SHL               / SHIFT INTO AC
         6
BZ,      DCA LOG
.
.
.
LS,      0
MS,      0
N26,     26
LOG,     0
```

FIGURE 4

```
                  / LOG DISPLAY ROUTINE FOR DP INTEGERS
                  / DOUBLE PRECISION FORMAT:
                  /    LOC. N   -- LEAST SIGNIFICANT HALF
                  /    LOC. N+1 -- MOST SIGNIFICANT HALF
                  / DECADE LINES ARE PLOTTED
                  / MAXIMUM NUMBER IS 10**6
                  / SIZE OF BUFFER TO BE DISPLAYED MUST BE A POWER OF 2
                  / MAXIMUM BUFFER SIZE = 1024
                  / CALLING SEQUENCE
                  / CALL   -- JMS
                  / CALL+1 -- STARTING ADDRESS OF BUFFER TO BE DISPLAYED
                  / CALL+2 -- BUFFER SIZE
                  / CALL+3 -- RETURN
                  /
                  / EAE IS REQUIRED
                  / LENGTH = 138 OCTAL
                  / EXECUTION TIME = 2MS + (N*45US.)
                  / PDP-8/I WILL RUN APPROXIMATELY 15% FASTER THAN PDP-8
                  /
                  / USES LOCATION 10 ON PAGE 0


0200  0000  LOGDIS, 0
0201  1600          TAD I LOGDIS            /GET SA
0202  3320          DCA SA
0203  2200          ISZ LOGDIS
0204  1600          TAD I LOGDIS            / GET SIZE
0205  3215          DCA SIZE
0206  2200          ISZ LOGDIS

0207  7344          CLA CMA CLL RAL         / -2
0210  3324          DCA CNT2                /SET UP INTERLACE
0211  3325          DCA DECST
0212  7332          CLA CLL CML RTR         / 2000
0213  7421          MQL                     / CALCULATE X INCREMENT
0214  7407          DVI
0215  7402  SIZE,   HLT
0216  7501          MQA
0217  7104          CLL RAL
0220  3326          DCA XINC

0221  3323  DIPB,   DCA X                   / SET UP STARTING X COORD.
0222  7040          CMA
0223  1320          TAD SA
0224  3010          DCA AUTO                / SET AUTO INDEX
0225  1215          TAD SIZE
0226  7110          CLL RAR
0227  7041          CIA
0230  3321          DCA CNT                 / SET COUNTER

0231  1410  DIPA,   TAD I AUTO              / LOAD AC AND MQ
0232  7421          MQL
0233  1410          TAD I AUTO
0234  7411          NMI                     / NORMALIZE
0235  7006          RTL                     / SHIFT BIT OUT LEFT END
0236  7421          MQL                     / MANTISSA IN MQ
0237  7441          SCA                     / GET STEP COUNTER
0240  7450          SNA
0241  5250          JMP BZ                  / LOG 0 = 0
0242  7041          CIA
0243  1322          TAD N26                 / SUBTRACT FROM 22
0244  7413          SHL                     / SHIFT LOG INTO AC
0245  0006          6
0246  7425          MQL MUY                 / SCALE SO THAT LOG(10**6)
0247  3100          3100                    / EQUALS 1023
```

```
0250  6063  BZ,     DYL                     / LOAD Y
0251  7200          CLA
0252  1323          TAD X
0253  6057          DXS                     / DISPLAY POINT
0254  1326          TAD XINC
0255  3323          DCA X
0256  2010          ISZ AUTO                / INCREMENT INDEX
0257  2010          ISZ AUTO
0260  2321          ISZ CNT
0261  5231          JMP DIPA
0262  7326          CLA CLL CML RTL         / 2
0263  1320          TAD SA                  / ADJUST SA
0264  3320          DCA SA

0265  3327          DCA Y                   / DISPLAY DECADE LINES
0266  1330          TAD M7
0267  3321          DCA CNT

0270  7300  DIP3,   CLA CLL
0271  1325          TAD DECST
0272  3323          DCA X
0273  1327          TAD Y
0274  6063          DYL
0275  1331          TAD N252
0276  3327          DCA Y
0277  1332          TAD M40
0300  3333          DCA CNT3

0301  1323  DIP2,   TAD X                   / PLOT A DECADE LINE
0302  6057          DXS
0303  1334          TAD DECINC
0304  2333          ISZ CNT3
0305  5302          JMP DIP2+1
0306  2321          ISZ CNT
0307  5270          JMP DIP3                / GO SET UP FOR ANOTHER LINE
0310  1335          TAD DESC                / SET UP FOR SECOND SCAN


0311  3325          DCA DECST
0312  1326          TAD XINC
0313  7110          CLL RAR
0314  2324          ISZ CNT2
0315  5221          JMP DIPB
0316  7300          CLA CLL
0317  5600          JMP I LOGDIS

0320  0000  SA,     0
0321  0000  CNT,    0
0322  0026  N26,    26
0323  0000  X,      0
0324  0000  CNT2,   0
0325  0000  DECST,  0
0326  0000  XINC,   0
0327  0000  Y,      0
0330  7771  M7,    -7
0331  0252  N252,   252
0332  7740  M40,   -40
0333  0000  CNT3,   0
0334  0040  DECINC, 40
0335  0020  DESC,   20

                    AUTO=10
```

PDP-5 AND PDP-8 HARDWARE DIFFERENCES AFFECTING SOFTWARE INTERCHANGE

Dave Westlake
Digital Equipment Corporation
Maynard, Massachusetts

Below are the hardware differences between the PDP-5 and the PDP-8 along with a discussion of their affect on programming:

1. The program counter on the PDP-8 is a hardware register.

On the PDP-5 location zero is used as a program counter.

This has no affect whatsoever on software interchange.

2. Interrupt on the PDP-8 deposits the program counter into location zero and transfers control to location 1.

On the PDP-5 the program counter is deposited into location 1 and control is transferred to location 2.

Software interchange involving interrupt cannot be made between the two computers without making appropriate changes to the instructions in the interrupt handler.

3. Extended memory programming differs in that the mnemonic, CDF, changes <u>all</u> memory references on the PDP-5 and only indirect memory references on the PDP-8.

A hardware option is available which when installed on the PDP-5 changes the CDF (Change Data Field) instruction to conform to that of the PDP-8. Otherwise, programs using extended memory addressing would not be interchangeable.

4. The instructions, complement the accumulator (CMA) and rotate the accumulator (RAL, RTL, RAR, RTR) can be combined into one instruction on the PDP-8 but not on the PDP-5.

All PDP-8 programs should be carefully searched for these instructions, which should then be separated into two instructions before using the programs on the PDP-5.

5. Due to the fact that the PDP-5 and PDP-8 use entirely different DECtape controls (the 555 and the TC01), DECtape control software is <u>not</u> interchangeable.

Also, if anyone ever desires to hook up a disk to a PDP-5, it would be doubtful that the PDP-8 disk software would work.

6. Neither CML nor STL may be combined with rotate instructions on the PDP-5.

<u>EAE Instructions</u>

These are distinctly different in that the PDP-5 uses IOT code 6000 while the 8 uses operate class 7000. Therefore, these instructions must be changed when using PDP-8 programs on the 5.

INTERFACE BETWEEN SR400 CARD READER AND PDP-8 COMPUTER

Peter H. Castine
Los Alamos Scientific Laboratory
Los Alamos, New Mexico

An Uptime Corporation Card Reader Model SR400 has been interfaced to a PDP-8 and has performed satisfactorily for six months. Several PDP owners have experssed interest in our experience and our interface. Therefore, this brief description has been prepared.

The card complement is:

2, W103's
8, R107's
2, R121's
1, R111
8, R202's
1, R602
1, R302

This list does not include the AC input mixers.

The omission of level changers and the quantity of inverters are worth noting. Uptime logic is quite similar to R series, and therefore, the signals from the SR400 to the interface are fed directly into R107's. On the other hand, the Uptime signals are slow (1 to 2 μsec rise time) and three stages of inversion are used to insure fast pulses of the proper polarity into the DCD gates of the data registers. R111's without collector clamped loads are used to transmit signals to the SR400.

Two twelve-feet lengths of ribbon cable are used to connect the interface to the SR400. No terminations have been used with these cables, and no troubles have arisen from the lack of terminations. This can be attributed to the slow response of Uptime logic.

The I/O instruction set is:

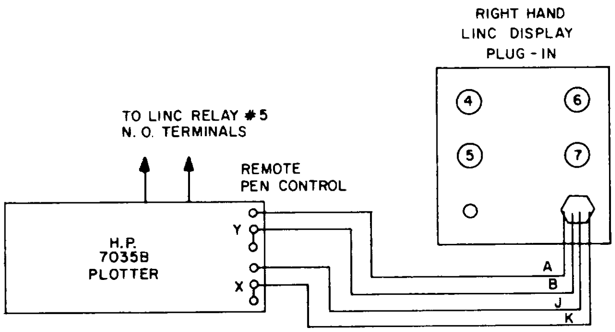
Skip if reader read. Clear flags and data register.
Skip if card done. Clear card done flag.
Read a card.
Skip of data register ready. Clear data flag. Clear register.
Transfer data register to AC.
Offset a card in output hopper.

Interested parties may contact the author for details.

POINT BY POINT PLOT WITH LINC-8

E. Duffin
Hospital of the University of Pennsylvania
Philadelphia, Pennsylvania

It has been found that relatively good, but slow, point by point plots can be obtained from a LINC-8 by adding an inexpensive ($1000) Hewlett Packard X-Y plotter (type 7035B). One small change has been made in the plotter pen by substituting a felt tip pen for the one normally used. (This is done to prevent splattering as the pen is lifted and dropped for each point.) The interconnections are simple and illustrated below:



The basic rules for operating this system are:

1. Give a display command and keep in a loop for a length of time sufficient to let the pen position itself.

2. Turn on the relay and the display again long enough for the pen to drop and make a point.

3. Raise the pen and display a little until it actually lifts off.

4. Go get next point.

5. Be sure the display channel is turned off so that the screen is not burned.

Next is a typical plotting program that will plot out data that is in quarter 4 of the LINC. Knob 7 will control the plotting rate (start with it at the extreme left).

```
      ⊟ 20
      SET i 1     ]   horizontal value
      0
      SET i 2     ]   # pts to plot
      - 400
      SET i 3     ]   data starts at 2000
      3777
      CLR         ]   raise pen.
      ATR
# 1S  SAM 7       }   Set up plotting rate.
      STC 1R
      SET 4
      1R
      SET 12          Set up plotting rate.
      1R
      SET i 13
      - 40
      SET i 5
# 1R  0           }
      LDA i 3
      DIS i 1
      SET 4
      1R              Display pt. long enough for
      DIS 1           pen to position itself.
      XSK i 4
      JMP p-2
      XSK i 5
      JMP p-6
      LDA i       }
      1
      ATR
      LDA 3
      SET 12          Display long enough for pen
      1R              to lower to paper.
      DIS 1
      XSK i 12
      JMP p-2
      XSK i 13
      JMP p-6
      CLR         }
      ATR
      LDA 3
      SET 12
      1R              Raise pen and resume display
      DIS 1           until pen gets off paper.
      XSK i 12
      JMP p-2
      XSK i 13
      JMP p-6
      XSK i 2         do all pts?
      JMP 1S          no
      HLT             yes
```

PDP-9—PROGRAMMING NOTE

N. Martin Sandler
Applied Physics Division
Atomic Energy of Canada Limited
Chalk River, Ontario, Canada

It is sometimes desirable to know the current step in long real-time experiment. We developed two routines when our PDP-9 was required to move a remote stepping switch used as a multiplexer. We wanted information about the switch during a run without using extra hardware.

On an interrupt wait is possible to display an octal number on the REGISTER indicators. A wait of seconds duration is sufficient. Loading the AC with the number and strobbing the I/O bus in a wait loop with a dummy I/O instruction (preferably not a "read" or "skip on flag") does the job. An example of such a loop is:

```
....
ISZ     SHWSTP    /setting number to be shown
LAC     SHWSTP    /put in AC
ION               /turn on interrupt
IOT               /dummy I/O command
JMP     .-1       /repeat
```

This sequence applies power to the proper REGISTER lights four out of every five microseconds while waiting. On interrupt, location Ø always has the address of the dummy I/O instruction, IOT.

Another way to show a change of step is to have the link on for one step and off for the next. Again, a wait of seconds is sufficient.

```
....
LAC     Ø         /get return address
RAL               /set link as before
CML               /change it
RAR               /rotate back
DAC     Ø         /reset location Ø
ION               /turn on interrupt
DBR               /re-store
JMP     I Ø       /JMP* Ø if using MACRO-9
```

The two routines may be combined when separate indicators are needed. The instruction LAC SHWSTP must then be put in just before ION. If DBR is not used, the instructions RAR and DAC Ø are not necessary.

PDP-10 PROGRAMMING NOTE

Gary Efron
Princeton University
Princeton, New Jersey

```
TITLE GCD          ;THIS IS A FORTRAN SUBROUTINE
ENTRY GCD          ;TO FIND THE GREATEST COMMON DIVISOR OF
GCD:Ø              ;2 NUMBERS USING THE EUCLIDEAN ALGORITHM.
MOVEM 1,A          ;FORMAT:CALL GCD(X,Y,Z) WHERE
MOVEM 2,B          ;X,Y,Z ARE DEFINED AS INTEGERS.
MOVEM 3,C          ;X,Y ARE INPUT. Z IS OUTPUT (GCD).
MOVE 1,@(16)       ;GET ARGUMENT ONE.
MOVE 2,@1(16)      ;GET ARGUMENT TWO.
CAMGE 1,2          ;IS ARG 1>ARG 2?
EXCH 1,2           ;YES. SWAP THEM SO DIVISION CAN TAKE PLACE.
IDIV 2,1           ;DIVIDE. REMAINDER WILL BE IN AC3.
JUMPE 3,.+4        ;IF AC3=Ø,GCD IS IN AC1,GO TO STORE.
MOVE 2,1           ;OLD DIVISOR BECOMES NEW DIVIDEND.
MOVE 1,3           ;REMAINDER BECOMES NEW DIVISOR.
JRST .-4           ;REPEAT DIV. AND SWAP TILL AC3=Ø
MOVEM 1,@2(16)     ;STORE ANSWER IN ARGUMENT THREE.
MOVE 1,A
MOVE 2,B
MOVE 3,C
JRA 16,3(16)       ;EXIT
A:Ø
B:Ø
C:Ø
END
```

HLT EVERYWHERE EXCEPT THE LOADERS

P. F. Jones
Nimrod Division
Science Research Council
Rutherford High Energy Lab
Chilton, Didcot, Berkshire, England

The following program used recently places HLT everywhere except for the loaders.

```
7611    0000    LOC,
7617    1300    TAD 7700
7620    3611    DCA I LOC
7621    2211    ISZ LOC
7622    5217    JMP 7617

7700    7402    HLT

SA :  7617
```

The program starts depositing at location 0000 and continues throughout the memory until a halt is deposited in 7611 whereupon it returns to location 7402 and repeats. Everything beyond 7611 is thus protected.

The program occupies gaps in the binary loader (except for 7700 which is part of it). It could therefore be loaded with the HELP Loader as part of the RIM and binary loaders; ready for use when they have been loaded.

One or two variants are possible. By omitting 7617, it will deposit 0000 instead, thus clearing the memory. Also since everything beyond LOC is protected, LOC can be moved to protect more of the last page. Finally, because the initial contents of LOC determines where the depositing starts, some of the lower pages can be omitted. This will not work, however, when depositing zero as the program returns to location 0 when it has put 0 in LOC.

CHANGES TO DECUS 8-64 AND 8-87

Edward P. Steinberger
Digital Equipment Corporation
Maynard, Massachusetts

It has been brought to my attention by DEC Field Service that some family-of-eight users with TC∅1 DECtape controls have been using DECUS 8-64 (PDP-8 DECtape Programming System) and DECUS 8-87 (XMAP) and experiencing trouble. Typically, a DECtape drive will "rock" in search mode trying to find a particular block. The DECtape diagnostics indicate no hardware problems when searching. Upon analyzing the programs in question, it was discovered that the DECtape on the TC∅1 was not being programmed according to hardware specifications. Because of the characteristics of the TU55 transports, when the transport is "turned around," the unit may not be "up-to-speed" at the point at which turn-around was initiated. To allow for this, the programmer should wait until the DECtape has gone one whole block beyond the block desired before turn-around is initiated. (This is usually encountered when executing a "search-reverse.") The search routines in these programs wait until the next block number is encountered, not until the next block is traversed, before turn-around is initiated. Thus, at times the desired block is not found and the DECtape rocks. To correct this situation the contents of the following locations (in the respective programs) should be changed from 771∅ (SPA CLA) to 775∅ (SPA SNA CLA) to allow the DECtape to go one block further on tape:

XMAP	-	∅435
XEDIT	-	2235
XPAL	-	2435
XMACRO	-	5235
XLOAD	-	7435
XLIST	-	1∅35
XDUP	-	?
XSYM	-	∅435

FOCAL NOTE

Ed Duffin
University of Pennsylvania Hospital
Philadelphia, Pennsylvania

Following is a procedure for saving FOCAL programs on the LINCtape without the user storing all of FOCAL at the same time.

1. Load FOCAL and write program.
2. Set RSW to 7600.
3. Press LOAD ADDRESS and START on PDP-8 console.
4. Type UPDATE and answer questions:

   NAME: User's choice
   SA (octal) 200
   MEMORY LOCATIONS: 0 3301,5277

5. To use FOCAL again, type either FOCAL or ESCAPE, LOAD ADDRESS and START at 200 via PDP-8 console.

To run a program that has been saved in the preceding way:

1. Load FOCAL, if it is not already running.
2. Set RSW to 7600.
3. Press LOAD ADDRESS and START on PDP-8 console.
4. Type name of FOCAL program that is to be run.
5. If program does not start automatically, type CONTROL C and GO.
6. To run another FOCAL program, repeat starting at Step 2.

FOCAL PROGRAMMING NOTE

FOCAL commands translated into French by the Iroquois Falls and Calvert District High School, Iroquois Falls, Ontario Canada.

TRANSLATION TABLE

Commandements Francais pour le calculateur electronique "IGOR"

French commands for the electronic calculator "IGOR"

	ENGLISH	FRENCH	LETTER
1.	SET	ORGANIZE	O
2.	FOR	QUAND	Q
3.	IF	SI	S
4.	DO	FAIS	F
5.	GOTO	VA	V
6.	COMMENT	COMMENTE	C
7.	ASK	DEMANDE	D
8.	TYPE	TAPE	T
9.	LIBRARY	ENTREPOSE	E
10.	ERASE	BIFFE	B
11.	WRITE	INSCRIS	I
12.	MODIFY	MODIFIE	M
13.	QUIT	ARRETE	A
14.	RETURN	RETOURNE	R

Ce n'est pas parfait
Mais "IGOR" est intelligent
Il comprendra

This is not perfect but "IGOR" is intelligent, it understands.

NOTE: "IGOR" refers to PDP-8/I

FOCAL POINTS

FOCAL POINTS was inaugurated in DECUSCOPE, Vol. 7, No. 2. The purpose of this column is to serve as a forum of comments and/or programs written in DEC's new family-of-8 conversational language, FOCAL. These programs and examples will be kept in printed form only and will usually be distributed in groups. Existing categories are Engineering, Plotting, Mathematics, Education, Accounting, Physics, and Statistics. Other group headings will no doubt be added rapidly. Anyone desiring a set or sets of these write-ups should write the DECUS Office, Maynard, Massachusetts 01754, indicating the category desired. Abstracts of additional routines follow. Each program submitted either to an existing category or as a new category should be submitted with a listing, an example of execution, and a brief abstract which will be printed in this column. We may also reproduce some results of their operation.

Reference the DECUS Library Catalog Addendum 1, DECUS No. 8-163, for complete list of FOCAL routines available. Recently submitted FOCAL Routines.

Engineering

Submitted by: David H. Tyrrell, Middlesex County College, Edison, New Jersey 08817

Δ-Y Complex - This program does a DELTA-WYE transformation for A-C circuits.

Y-Δ Complex - This program does a WYE-DELTA transformation for A-C circuits.

Series Resonant Circuit Analysis - This program computes resonant frequency, bandwidth, Q, and values of inductive and capacitive reactance at resonance for a given R-L-C series circuits. It also produces, upon request, a table of impedance and phase angle for 10 points each side of the resonant frequency. Distance between points is determined by a user inputted DELTA-F.

Mathematics

Submitted by: David H. Tyrrell, Middlesex County College, Edison, New Jersey 08817

Rectangular to Polar Conversion - Converts complex numbers in rectangular form to polar form.

Polar to Rectangular Conversion - Converts complex numbers in polar form to rectangular form.

Submitted by: Ron Dorman, Georgia Institute of Technology, Nuclear Research Center, Atlanta, Georgia

Rootfinder Program - The Rootfinder program is a simple procedure, for use in determining the real roots of any suitable function. The program uses a conventional search to find root containing intervals followed by a binary search (successive approximation method) to converge on the root value.

Determinot Program - The Determinot program is a simple program which may be used to find the determinant of a square matrix of dimension 2 x 2 to 6 x 6. The method used in finding the determinant is based on the definition of the determinant and involves an N! summation of products of N matrix terms with the proper inversion sign.

Statistics

Submitted by: M. J. McKeown, University of Chicago, Department of Obstetrics and Gynecology, Chicago, Illinois

Simple Statistics Routines -

1. One-Sample Statistics:
   Two-Sample Statistics:
   Welch Procedure

2. One-way Analysis of Variance

3. Sheffe's Contrast Between Means

Submitted by: D. E. Wrege, Georgia Institute of Technology

STRIP FOCAL: A Data Display and Storage Program - This program, written for FOCAL W. on an 8k machine, accepts data from paper tape on the high speed reader, and displays it on the Type 34 display unit. The data is stored in upper core, or on the disk, using the FNEW array. An ERASE or ERASE ALL command will not wipe out the stored data. Several sets of data may be stored in different sections of the array with the user keeping track of the indecies.

Physics

Submitted by: Bryan W. McGhee, Georgia Institute of Technology, Atlanta, Georgia

Monte Carlo Solution to Neutron Penetration Problem - The display is a one axis display (Z coordinate only) of each scattering event - though scattering is calculated in 3-dimensions. The axis shifts upward for each new neutron to facilitate ease of following collisions.

Accounting

Submitted by: D. E. Wrege, Georgia Institute of Technology

GRADE: A Grade Averaging and Display Program - This program will average grades from a number of quizes, taking into account weighting factors of relative importance between quizes, and plot a histogram of the number of grades in a given percentile.

Software for Sale or Lease

SUBMITTED BY
CALCOMP, California Computer Products, Inc.

The following six subroutines constitute the basic software packages available for the Calcomp Plotter.

Subprogram AXIS - Provides the capability of drawing an annotated axis for linear graphs.

Subprogram SCALE - A functional subroutine which scans a set of values and computes a "desirable" scale factor to present the information at the desired size.

Subprogram LINE - Provides the capability of connecting data points with straight lines and/or producing centered symbols at specified data points.

Subprogram NUMBER - Provides the capability of converting a machine format number to decimal representation and plotting the number to the specified precision.

Subprogram SYMBOL - Provides the capability of drawing the full character set of the particular computer at any specified location, size, and orientation. In addition, a number of additional characters are included for data point plotting.

Subprogram PLOT - Provides the basic capability of producing the necessary string of plotter commands to move the pen from one point to another. For on-line plotters, this subroutine includes the I/O Commands for driving the plotter; for off-line systems, all tape formatting, tape writing, and checking are included. This subroutine is the heart of the plotting package, and all data, whether from other subprograms in this group or user programs, are passed to this subroutine in the form of straight line segments for plotting.

In addition to the basic six routines described above; a FORTRAN program called SAMPLE is normally provided to exercise the subroutine package for initial installation and check-out.

	Calcomp Product No.	Computer	Interface	Tape Density	Plotter Series
1.	34000	PDP-8	DEC Interface	N/A	500
2.	35000	PDP-9	DEC Interface	N/A	500
3.	36000	PDP-10	DEC Interface	N/A	500
4.	36350	PDP-10	Calcomp 760 Off-Line	500/2	500
5.	Time Sharing	PDP-10	Calcomp 210	N/A	500

For more information contact:

Richard H. Hinkley
California Computer Products, Inc.
118 Cedar Street
Wellesley, Massachusetts 02181

WANTED

PRG Program for PDP-8. Program is used to generate reports. Also, Generalized SORT Program. Please contact:

Dan LaRoe
Digital Equipment Corporation
1625 West Mockingbird Lane
Suite 309
Dallas, Texas 75235

"Dear Mrs. Cossette:

"This letter is really intended for your DECUS members.

"I want everyone to know about the new product group which has been formed within DEC with two major responsibilities: 1) to support users of our older product lines, i.e., PDP-1, 4, 5, 6 and the Classic LINC and 2) to make DEC supplies available to all our customers.

"This new group is called "Traditional Products." It is the responsibility of Traditional Products to make the newly developed computer options available to users of our older systems. For example, we want all PDP-8 options to work with PDP-5, PDP-9 options on PDP-7 and PDP-4, PDP-10 options on PDP-6, etc. As new options, software, etc. are developed by DEC, "TP" will offer them to our "traditional" customers. We also work very closely with our special systems group to insure that these customers can still obtain one-of-a-kind equipment.

"We will also provide a listening ear to customers who might have a problem with an older computer, option, etc. Basically, I want all our customers to know that there is a department within DEC directly responsible for the older computer products.

"We will make certain that our customers can continue to expand their systems, take advantage of new hardware developments, obtain latest software, and get assistance when necessary.

"Further, DEC will now look after customer purchased supplies with real dedication. We are now maintaining adequate levels of supply (paper tape, DECtape, LINC tape, etc.), and we will provide same week deliveries. We are developing new storage devices and better supplies and will sell these at a fair price with a guarantee to their quality."

Robert Lane
Manager
Traditional Products
Digital Equipment Corporation
Maynard, Massachusetts

"Dear Mrs. Cossette:

"The description of the Fast Fourier Transform by Mr. Rothman was very well written and easily understood. The flow chart was also very useful. However, one minor modification to the flow chart should reduce the number of complex exponentiations to be calculated by about half the number required in the given flow chart.

"In box 8 of the flow chart, U and K are calculated. However, their values do not change within the internal loop composed of boxes 8, 9, 10, and 11. The low order bits do change, but these are shifted out. Therefore, U and K could be calculated in box 7, just prior to the entry to the internal loop. At the same time, W^k could be calculated and saved so that it need not be calculated every time box 8 is entered. Figure 1 reproduces that part of the flow chart under consideration and figure 2 gives the modified flow chart section.

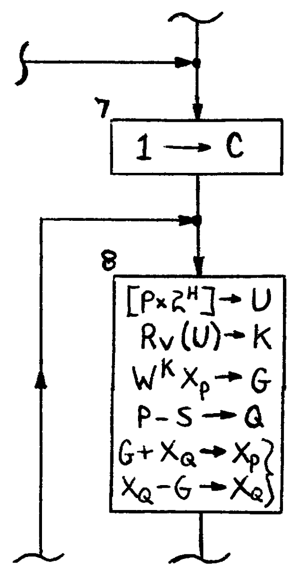


FIGURE 1 - ORIGINAL FLOW CHART

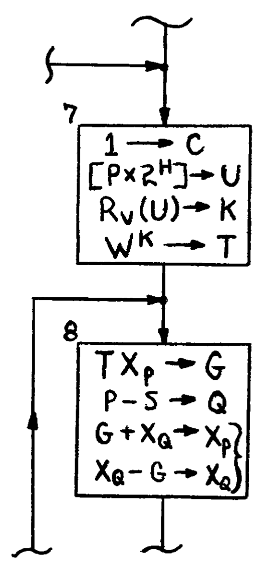


FIGURE 2 -- MODIFIED FLOW CHART

Steven R. Deller
Systems Analyst"
The Medical School
Northwestern University
Chicago, Illinois

PDP-8/PDP-9 TRANSLATOR AND PDP-9 STATISTICS PACKAGE

A translator package that will make it possible to adapt a major part of the code written for its PDP-8 family of computers in PAL III, PAL D or MACRO-8 assembly language to medium size PDP-9 computer line MACRO-9 assembly language will soon be available from Digital Equipment Corporation.

In addition, DEC is now offering a statistics package, called STATPAC for use with its PDP-9 and PDP-9/L computers.

The new translator is not designed to produce a program that runs on the PDP-9 by simulating the PDP-8, but rather to translate a major part of the source code and to indicate those parts that should be reviewed in the light of the greater word length and more powerful instruction set of the PDP-9.

The statistics package is a collection of modular statistical programs. It is written in FORTRAN and is designed to operate within the PDP-9 Keyboard Monitor System.

The package consists of five modules: control, input, descriptive statistics, stepwise linear regression, and multiple linear regression.

According to the staff of the Programming Department, other modules are being considered for later addition to the package. They include: analysis of variance, factor analysis, descriminant analysis, restrictions capability, transformation capability, and two-way tabulation capability.

KEYBOARD DISPLAY ADDED TO PDP-10 LINE

A keyboard display terminal, providing quicker response in an interactive computing environment than is possible with teletypewriter devices, has been added to the PDP-10 product line.

The VT03 display console operates similarly to a conventional teleprinter and incorporates "carriage return" and "line feed" characters for position control. It is virtually noiseless and accepts data at the rate of 1200 baud as compared to a teleprinter rate of 110 baud.

The full-duplex console features a local memory for display refreshing thus eliminating the demand on processor time usually required for this function. The VT03 displays up to 960 characters arranged in 12 rows of 80 characters each.

Among others, the display unit features an alphanumeric keyboard, editing capability from the keyboard or computer, audible end-of-line and incoming message tones and plug-in boards for easy maintenance.

The unit is priced at $7900. An interface option, priced at $400, is available which allows the user to generate hard copy remotely via standard Teletype devices.

First deliveries of the new unit are scheduled for this summer.

The VT03 keyboard display joins a comprehensive line of PDP-10 options which include, among other items, mass storage devices, card handling and line printing equipment, display and plotter systems and data communication equipment. More than 1.5 million console hours have been logged on PDP-10 systems in some 50 worldwide installations in such environments as commercial time-sharing, manufacturing, banking, universities and research.

INCREMENTAL MAGTAPE FOR PDP-8 FAMILY OF COMPUTERS

A variety of tape control units and transports for use with its PDP-8 family of computers have just been introduced by Digital Equipment Corporation.

The Control Unit Type TR02 is designed to perform a programmed data transfer between the PDP-8 family processor and an incremental write only or incremental write and synchronous read tape transport (seven or nine channel).

Both single and dual controls are available in the TR02 series. Each is available as read/write or write only. The prices range from $2,100 for a single, write only control to $3,600 for a dual, read/write control.

The transports of the TU22, TU25, and TU28 series are either write only or read/write units of either seven or nine channels.

The synchronous incremental writing speed is 700 characters per second maximum. Read transfer rates are 5K to 13.9K characters per second, with bit densities of 200, 556, 800 bits per inch available.

Prices range from $5,550 for a seven channel write only 200 bits per inch to $9,400 for a read/write nine channel 800 BPI.

PDP-9 PERIPHERAL DEVICES

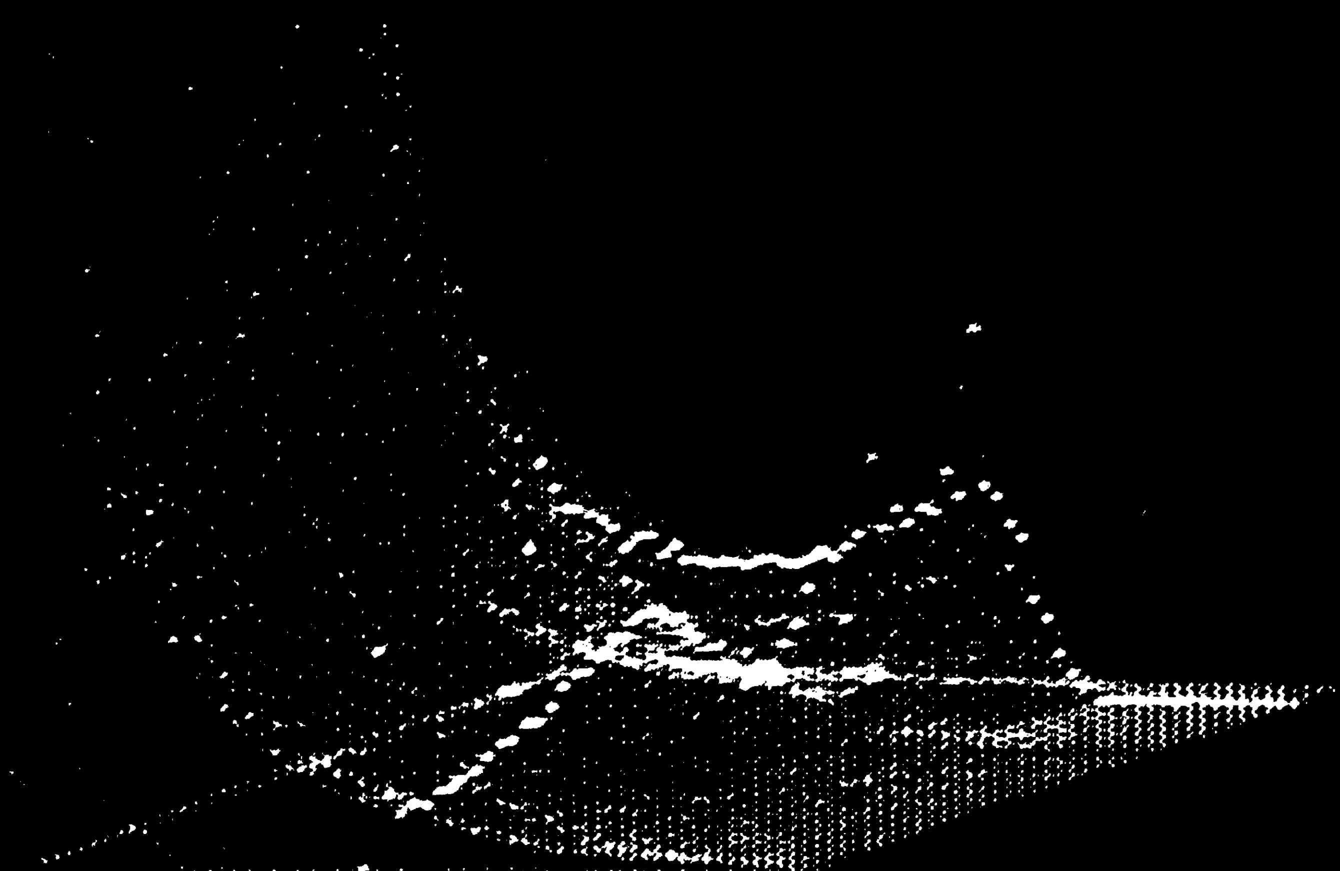
Two new peripheral devices for its PDP-9 family of medium-sized computers were announced recently.

The first, a medium-speed card reader and control, also can be used with DEC's PDP-8 family of small computers. This device reads standard 12-row, 80 column punched cards, by columns, beginning with column one. A single select instruction starts the card moving past the read station. Once in motion, all 80 cards are read. A total of 200 cards can be read per minute.

The control with the reader assembles 12 bits from each card column into its 12-bit binary data buffer. A program interrupt is requested when data from each column is ready in the buffer. The data is transferred from the control to the accumulator of the PDP-9 or PDP-9/L under program control. Data bits (holes) in each column are sensed by a light source/photo diode system. The card reader flag is set when the data from each column is ready in the data buffer. Also, the flag is set if there has been an error, and when the card has completely passed through the read station. The flag may be cleared without further action.

The second device is a low cost disk storage and control unit. The control unit can take up to eight disks, each with a capacity of 262,144 eighteen-bit words for a total capacity of 2,097,152 words of this length. The control unit operates through the three-cycle data break channel of the PDP-9 or PDP-9/L computer. A transfer rate of either 16 or 32-microseconds a word is switch selectable at the control. Average access time is 16.67 microseconds, and write lockout is provided in increments of 16,384 eighteen-bit words. The control uses integrated circuitry.

For more information on the above articles, contact the DEC Sales Office in your area.



Printed in U.S.A.

1969—Volume 8 Number 4

DECUSCOPE

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY

CONTENTS

	PAGE
News from DECUS-Europe	1
A Little on DECUS Meetings	1
DECUS Mailing Service	1,2
Mailing List Update	2
Radiation Treatment Planning Project	2
Digital Control of Bevatron External-Proton-Beam Transport Magnets	3-7
Remote Operation of a PDP-8 Computer	8,9
Wanted - For Sale Items	9
PDP-8 Programming Notes	
Disappearing DECtape Bootstrap	10
The Use of FORTRAN Subroutines in PDP-8 FORTRAN (4K)	10
FOCAL Programming Note	10,11
PDP-7/9 Programming Notes	
A Fast, Small Subroutine to Zero Arrays in PDP-7/9 FORTRAN	11
Programs for Masking and Processing Nonstandard Paper Tape Input to PDP-7/9 FORTRAN	11-13
Software for Sale or Lease	13,14
Program Available from Author	14
TP-1351 Cartridge Tape Unit	15,16
Letters	16-19
News from DEC	20

NEWS FROM DECUS-EUROPE

Authors presenting papers at the 5th European DECUS Seminar in Stockholm, Sweden in September are submitting mini-papers (one or two-page versions of papers to be presented) in advance of the meeting. The mini-papers will be bound in one volume and will be available at the end of August. Copies will be distributed free of charge to members attending the meeting. Others may request copies at a cost of 15 shillings ($1.80) each by writing to:

DECUS Representative
Digital Equipment Co. Ltd.
Arkwright Road
Reading, Berkshire, England

The complete version of the papers will be published in a Proceedings. These are distributed automatically to meeting attendees. For those not attending copies are available at 2 pounds ($5.00) each by writing to the:

DECUS Office
Digital Equipment Corporation
146 Main Street
Maynard, Massachusetts 01754

A LITTLE ON DECUS MEETINGS

The last DECUS Spring Meeting held in Wakefield was well attended (665 registrants). This was the first time that five simultaneous sessions were attempted. Although, we managed to get all the presentations in during the two days, attendees generally felt that they missed many papers they wanted to hear because of so many sessions running simultaneously.

The R.E.S.I.S.T.O.R.S. group gathered a considerable amount of attention with their demonstrations on the computers displayed at the meeting.

Our applause to the Biomedical Session speakers and participants as the feedback on these sessions was generally excellent. Holding the Biomedical Sessions in another building did prove somewhat inconvenient at times, however, we were limited in space available in the main hotel building. It just proves that DECUS is growing so fast, we are even outgrowing some hotels.

For the Fall 1969 Meeting it is hoped to correct the space inadequacy. The meeting will be held at the Flamingo Hotel, Las Vegas, Nevada on November 17 and 18. We overlap with the FJCC by one day, but these were the only days close to the FJCC available to us. Information flyers regarding housing accommodations have been mailed to all members and a "Call for Papers" has gone out to the complete mailing list. Please try to get your abstract and mini-paper in as soon as possible if you are planning to make a presentation at the meeting. We have taken a suggestion from our European counterparts in instituting the mini-papers concept for the U.S. Meetings. All prospective speakers have been asked to submit mini-papers, a condensed version of the final paper, for review prior to acceptance. The mini-papers of accepted presentations will be bound and distributed at the meeting.

Registration forms for the meeting sessions will be sent shortly. Registration in advance is advised, it will lessen the load on the registration desk and help eliminate delays in opening the sessions.

We will again allow literature displays. Those interested in displaying brochures, flyers, etc., should send a sample of the material to be displayed to the DECUS Office. The only charge associated with the display of literature is a handling charge of $5.00 per brochure that is, each type of brochure, and the limit per brochure is 500 pieces.

DECUS MAILING SERVICE

At a recent DECUS Board meeting, the charging policy for the mailing service was revised somewhat to enable requests for specialized listings. The new policy reads as follows:

> A company or organization may specify specialized lists (i.e. by area or product line) at a fee of $.25 per label, the minimum charge being $100.00. The total cost of the specialized list shall not exceed the cost of the complete mailing list.
>
> For the complete mailing list, the fee is $.10 per label. The total charge will be determined by the number of individuals on the mailing list at the time of the request.

DECUS provides its mailing list in label form and only literature approved by the DECUS Board can be mailed under this service.

If an organization is interested in using this service, they should send a sample of the material to be mailed to the DECUS Office for approval. Once approved, DECUS will notify the organization of the cost of the labels. The organization will then forward all material to be mailed stuffed in envelopes to DECUS. DECUS will affix labels to envelopes, seal and mail. The organization must also assume the cost of postage. Any questions regarding this service should be directed to the DECUS Executive Secretary, DECUS, Maynard, Massachusetts 01754.

MAILING LIST UPDATE

We are now in the process of updating our mailing list for verifying address and adding new application and configuration codes to the list. Cards have been sent to all non-members and individual members, and forms are being sent to all delegates. Please try to complete these cards and forms as soon as possible. Your cooperation in this matter is important in order to maintain an accurate and up-to-date system.

Angela Cossette
DECUS Executive Secretary

RADIATION TREATMENT PLANNING PROJECT

The interim specifications for the radiation treatment planning program are described in the following paragraphs. The system is being developed for the Royal Marsden Hospital, England, by David Cope, Digital Equipment Company Ltd., Reading, England. Any questions regarding the system should be directed to Mr. Cope.

OVERALL DESCRIPTION

The function of the program is to assist in producing the data necessary to set up equipment producing X or γ radiation prior to using this radiation for the treatment of cancerous tissue.

At present a radiotherapist works from a traced body outline of the patient and a library of X-ray beams. Using his previous experience, the radiotherapist positions various beams at certain positions and angles in such a way as to produce a high, even distribution in the area of the tumor with as low doses as possible elsewhere. This distribution is produced by superimposing over the patient outline traced, graphical representations of the beams. The resulting contour map of doses is then drawn manually. Although the manual method works very satisfactorily it is extremely laborious and time consuming; and if all the mathematical corrections are applied rigorously, it may take two hours or so to produce one-dose distribution.

This program is intended to speed up this operation, enabling the production of a dose distribution in four minutes or so. The facilities available are:

1. Store on DECtape a representation of X or γ ray beams. Approximately 400 beams may be stored on one tape.

2. Store on DECtape a representation of the patient outline together with a summary of the last four treatment plans made for that patient. A minimum of 125 outlines may be stored on one tape.

3. To digitize and convert to an acceptable format the analog input from a tracing device known as a Rho-Theta transducer used to trace a drawing of the patient outline, internal structure and tumor area.

4. To display, in a suitable scale on a storage oscilloscope, the patient outline.

5. To permit the display in non-storage mode of from 1-6 beams, and to permit adjustment to each beam of position, angle, radius and relative weighting. The display of beams is scaled by the same factor as the patient outline.

6. From the results of step 5, to display requested isodose curves (lines of equal dose).

7. To store the data specifying the position, angle, radius and weighting of the beams (subsequently referred to as the Plan) on DECtape immediately after the appropriate patient outline file.

8. To print a summary of the Plan on the Teletype.

A similar program is in operation on the P.C. (Programmed Console) which is a 4K computer, without on-line mass storage, having an instruction set somewhat similar to the LINC. The proposed hardware configuration was decided upon by the UK Institute of Cancer Research in consultation with DEC and in the light of past experience with the P.C. at the Biomedical Computer Laboratory of Washington University, St. Louis, Missouri. This version of the program implements the functions described above.

GENERAL SPECIFICATIONS

Machine Requirements

A PDP-8 or PDP-8/I with 8K of core, one TC01 control, two TU55 tape transports, one Teletype, one KV8/I controller with write-through option, one Tektronix 611 display, one A to D converter of at least 10 bits, one 4-channel A/D multiplexer, one Rho-Theta transducer, and four control potentiometers.

Machine Options

EAE is regarded as being extremely desirable. Although certain sections of the program are limited by I/O and operator action, the displays of beams and isodose curves require considerable calculations and it is estimated that EAE will reduce the execution time of these sections by a factor of about 3.

System Requirements

Although the program could be written to stand alone, it was decided to run it under the Disc/DECtape Monitor System for ease of calling various sections of the program and also for ease of further expansion and/or modification.

Resident Programs

The program is self-contained except for the monitor.

DIGITAL CONTROL OF BEVATRON EXTERNAL PROTON-BEAM TRANSPORT MAGNETS*

Don M. Evans
Lawrence Radiation Laboratory
University of California
Berkeley, California

SUMMARY

A computer-based digitally-structured, data acquisition and control system has been developed for on-line programmed control of external beam transport magnets at the Bevatron. The control algorithm achieves flexibility to accommodate all possible modes of bevatron operation by separating the operating cycle of the accelerator into several zones. This method allows independent processing of Bevatron field within each zone, and the introduction of zone-related time-variable functions in each of the magnet currents.

Evolution of the hardware is discussed and use of small computers is considered for each well-defined project, over a time-shared single large computer. The technique utilized to provide time-sharing of peripheral hardware between small computers is described.

Pertinent details of methods used to reject noise in transmission of both analog and digital data are examined.

Communication between digital systems and the human operator is discussed in terms of minimizing the number of operator responses required to establish a series of functions to be performed by the system.

INTRODUCTION

With the progressive expansion of the external-proton-beam facilities at the Bevatron, the operator-controlled beam transport magnet system (Figure 1) has become sufficiently complex to warrant serious consideration of a replacement for the multiple-control--"several knobs/magnet" technique. Additional flexibility also seemed advisable in the control of these pulsed magnets over a wide range of possible operating modes of the accelerator. Accordingly, a digital system has been created and placed into full-time, on-line service in control of the pulsed magnets.

GENERAL CONSIDERATIONS

At the outset, a decision was made to push the boundary between the analog and the digital as far from the center of the system as possible. This defines the system as strictly digital, with coupling from the analog world as soon as possible for data input, but retaining the data in digital form to the last possible point. To implement this requires the transmission of data in digital form.

*Work done under the auspices of the U.S. Atomic Energy Commission.

Presented at the National Particle Accelerator Conference, Washington, D.C., March 1969, and reprinted with permission of the University of California, Lawrence Radiation Laboratory, Berkeley, California.

AEC Contract No. W-7405-eng-48. UCRL-18563.

The power supplies to be controlled are 3-phase full-wave rectifiers of two types. Most are SCR, and four are magamp with shunt transistor actuators. In any case, the digital system responsibility rests with providing a reference voltage to the regulator. The reference must be essentially noise-free, and must appear as a continuous signal insofar as no discontinuities may be present in magnet current due solely to the digitally quantized reference amplitude or to the selected update rate. The time allowed between reference changes is at least the period of the gate triggers to the SCR's. Three milliseconds was selected as the minimum period required between control adjustments. This decision specifies much of the balance of the system. The number of magnets and the extent of the calculations are required to define the computer to be selected and the data rate specified for the transmission system. In our case, the maximum number of pulsed magnets was set at 35, since only those of the pulsed variety require careful tracking with Bevatron field and therefore require on-line operation. There is provision for 64 magnet control channels, dc and pulsed combined.

The Bevatron operating cycle has been divided into a number of independent zones. In general, a zone is defined as the region between points of discontinuity in the $\dot{B}$. An example of this type of zone assignment is shown in Figure 2. Zones may be arbitrarily defined as well. Regions between accelerator current markers, and regions bounded by limits in real-time are examples.

CONTROL ALGORITHM

The calculations required for each magnet at each 3-ms update point must satisfy the requirement of independent processing of each magnet current and must track to the required proportion of Bevatron field within each zone of operation. That is,

$$I_{mag} = \rho_z \int \dot{B} \tag{1}$$

where ρ_z is the zone-dependent, operator-determined proportionality factor which provides proper rate of rise within a zone, or attains the proper current level at the end of a zone; either point of view can be taken by the operator. The calculations must also allow introduction of zone-related time-variable functions in each magnet current.

Considering all of these factors, all data processing is performed relative to the beginning of the current zone, so that:

$$CW \quad CW^i_\phi + (B-B^i_\phi)x^i + (t-t^i_\phi)y^i \tag{2}$$

where CW is the control word to the magnet, B is the Bevatron field value, X is the operator-specified ratio (ρ specified above), t is the time quantized to 3 ms update periods, and y is the operator-specified time-dependent slope. Subscript ϕ denotes the value of the parameter at the beginning of the Ith zone. If it were intended to run "open-loop" without considering slow-drift correction or ripple-detection, the operator could operate directly upon x and (or) y to achieve control. To "close" the loop to the extent that magnet-power-supply drift or reference-related gain change effects can be removed from the operating system and thereby reduce operator "trimming", our x and y include information about the control-word/magnet-current transfer function. A dimensional analysis from (2) reveals:

$$X\left[\frac{\text{control word}}{\text{gauss}}\right] = A \cdot BR\left[\frac{\text{control word}}{\text{ampere}} \cdot \frac{\text{ampere}}{\text{gauss}}\right], \qquad (3)$$

where A is the transfer function (digital input to analog output of power supply), and BR is the operator-specified magnet current to the Bevatron field ratio, (B ratio). Similarly: we have

$$y\left[\frac{\text{control word}}{\Delta\ \text{time}}\right] = A.S\left[\frac{\text{control word}}{\text{ampere}} \cdot \frac{\text{ampere}}{\Delta\ \text{time}}\right], \qquad (4)$$

where S is the operator-specified magnet-current change per unit time.

During active control changes, the magnet under control is retained in the open-loop mode, the operator request is used to modify BR or S directly, and the subroutine to calculate new x and y is called repeatedly as operator data varies. This apparently provides direct connection of the control knob to the magnet reference. Upon completion of the active control request, the magnet is again inserted into the closed-loop control mode.

Since the closed-loop is not used to provide a constant current in each magnet on a pulse-to-pulse basis, but rather must maintain a constant ratio of magnet current to Bevatron field, this closed-loop control method is thought of more as a "holding" technique. Since the initial placement into full-time control, several unusual (though not unanticipated) modes of control have been requested. These involve providing certain step functions to be applied to magnet references at various times to provide beam-channel switching or temporary fields to allow short-term extraction techniques to be applied.

This accommodation is provided by software capability to remove from or insert into the normal update cycle any magnet at anytime. While out of the update routine, the magnet may be directed to any predetermined current level.

HARDWARE REQUIREMENTS

Figure 3 is an abbreviated block diagram of the hardware. Magnet-current data are obtained by scanning transductors on the controlled magnets with an analog multiplexer and analog-digital converter and by sampling the current value of each magnet transductor every 3 ms. There is provision for 64 analog inputs. This magnet current data is saved at the end of each zone, as are control words, in order to check the transfer function, A, in each zone, and to allow calculation of new xy values as A varies. Variation beyond allowed tolerances initiates a notice to the operator of drift or ripple problems with a magnet.

It should also be noted that the value of the Bevatron field must be used in control-word calculations. The integral of $\dot{B}$ is obtained by digitally integrating the output of a voltage-frequency converter in a 21-bit up/down scaler, the contents of which is read into core each 3 ms. Provision is made for 64 digital channels to be scanned. Other digital inputs include magnet status, operator control data and radiation-chain status.

Combining the methods of handling parameters into a reasonable hardware system results in the core of the magnet control system becoming a three-part direct memory access (DMA) device which provides completely asynchronous input/output of the three channels of data: control word out, converted analog data in, and pure digital data in. The three DMA devices are identical in most respects. The small differences will be described as they are encountered.

The PDP-8 computer utilized by the system is a 12-bit, 1.5-μs, 4k machine. Our control is 1 part in 4k, as is our monitoring. Therefore, all I/O is 12-bit data. Each of the DMA devices has a table in memory specifically dedicated to it, the starting location being defined by the contents of a dedicated pointer location in core. The pointer location is defined by hard-wiring in the DMA device, whereas the program can modify the starting location of the table to be operated on by data.

At the tick of the external 3-ms clock, each of the three devices starts a cycle, which begins by consulting the pointer in core and transferring this address into an external register. The maximum number of channels in each device is 64_{10}. This allows us to define the end of data transfer by detecting 77_8 in the address register. By blending all the operations associated with addressing during data transfer, the least six bits of the external address register point not only to the "slot" in the core table, but also to the channel of the device being serviced. This provides complete dedication of specific locations in core for parameters involved in calculations. The pointer in core can be program-modified to change the length of the table or its absolute core location, but during ordinary control, the program is not required to exercise any significant control over the DMA devices, once having started them. Since the three devices are asynchronous and each requires a different time to service one channel and move on to the next, there is a priority established such that no anomolies exists as the devices vie for access to the computer.

TRANSMISSION DEVICE

The magnet power supplies under control are located at widely separated areas of the Bevatron, such that the difference in chassis ground potential during the Bevatron cycle approaches 4V due to magnetically induced currents as observed from the computer location. Commutation noise from SCR power supplies exceeds this level by about three times. The method of data transmission selected must have common-mode rejection capable of operating under these conditions, and further, must be able to reject asymmetrical noise induced into the system by asymmetry in lead-dress, wire-length or even maintenance-oscilloscope-connection ground loops. The time required to transmit control words to all magnets, separates the first transmission from the last in real time and introduces a small error, since the B-field measurement used in calculations of control words, is the same for all magnets. Therefore, the parallel transmission of all data bits is desirable. To live in a noisy real-life environment, the receiver data-buffer logic was selected from the Digital Equipment Corporation Industrial Series (K-series), which utilizes a Miller integrator capacitor in the logic gates. The next question to be met is that of common-mode noise. Differential current sources driving a twisted pair, random-lay cable with a characteristic impedance of 105Ω, provides high data reliability when adjacent line pairs in a cable are shorted together (the most likely failure mode). This technique, with differential line receivers, combines advantages of high data rates and dc operation. The data and address line data rate is 100k Hz and other lines are operated dc to provide interlocking. Twelve-bits of parallel data, together with 6-bits of parallel address lines are presented to all receiver buffer registers at once, as is a strobe pulse. That buffer which decodes the address line uses the strobe to update itself and in turn, returns the strobe as a data-received pulse, which is used at the DMA device to advance the address register and obtain the next piece of data for the next magnet in sequence. Should the transmitted address be outside the range of addresses of magnets under control, or should a failure occur at a remote receiver location, the absence of the return accepted pulse creates a system error, and diagnostics are called into service to determine the offending channel. Individual 12-bit DAC's are located at the remote receiver locations to be connected by minimal-length cabling to the power-supply regulators as reference. Provision is made to allow direct remote analog monitoring of the DAC outputs in place of magnet transductors for diagnostic purposes.

ANALOG DEVICE

The central element in the analog monitoring system is the 12-bit analog-digital converter, which is fed from an analog FET multiplexer with up to a 64-channel capacity. It is not expected to exceed 35 channels of magnet-current monitoring, since to obtain 12-bit accuracies at reasonable cost, the successive approximation ADC consumes 4 μs bit. An ADC complete signal requests a DMA cycle direct to core to insert the converted value into the table entry as defined by the address register in the external device. The least 6 bits of the register, also defines the magnet for which current is being converted. Satisfactory insertion of the data into core calls for advancing to the next magnet and table address and a new ADC conversion of this next entry. The program responsibility for analog entries presently rests with a graphic CRT display of the instantaneous value of magnet current for the magnet selected for display. The program must also store away necessary values from the analog table at the end of each zone, to be able to calculate the transfer function for each magnet in each zone of the preceding Bevatron acceleration cycle.

Because of the presence on the analog lines, of the common-mode signals described in the transmission device, differential-input follow and hold amplifiers are used with a one-point analog ground at each remote receiver location. This allows the differential lines into the amplifiers to vary independently within the common-mode capability of the amplifiers without creating unreal analog values to be acquired which would then create a problem in calculation of transfer function on a pulse-to-pulse basis. Should the analog system include random noise in the data, the slow closed loop becomes a random-hunt system and could introduce a larger error margin than that expected from magnet power-supply drifts.

THE DIGITAL DEVICE

This device operates similarly to the analog device, except at a higher throughput, because it simply scans data on digital lines, which are static as a rule. There are some exceptions, and to accommodate asynchronous data lines, a strobe is available which is similar to the transmission device. This DMA device does not require a return accepted pulse to advance its address register, but rather dwells on a digital channel a fixed time and moves on. If the strobe is used to synchronize variable data, it is assumed the data is on the lines within 1 μs after the strobe. One input that requires the strobe is the digital integrator for field value. The maximum frequency of count pulses to the scaler is approximately 2.0 MHz. Data must be taken from the scaler only when all 21 bits are quiescent. Therefore, if counts are occuring, the nearest one within the tolerance allowed by the digital system is synchronized to the digital-device strobe line and used to place the scaler data onto the digital input channel. On the other hand, if no (or few) counts are coming in, the digital strobe itself causes the data in the 21-bit scaler to be placed on the data channel.

ZONE DEFINITION-INTERRUPTS

The zone orientation of the control system has been described, and the method used to define zones is now discussed. A hierarchy of priority interrupts has been established. The PDP-8 has provision for only one interrupt bus, so an external device has been developed, with arm, disarm capability for 16 levels of interrupt. To expedite service of interrupts, the level number is strobed to the accumulator under program control to be used as an address modifier to get to the service routine for the specific level in the shortest time. Each level, of course, can have any number of individual devices on it, each capable of causing an interrupt. An example is that of the Bevatron timing-pulse interrupts, in which case, there are 12 different pulse inputs on each of two levels. Each DMA device is assigned a level of interrupt to provide early discovery of any malfunction or error in the devices. Zones are defined as extending from any accelerator-derived pulse to any other. Those normally used are the $\dot{B}$ derived pulses such as Mag On, Flattop On, etc. Upon detection of a zone-change interrupt, the program has the responsibility to save zone-end parameters, transfer the contents of a minimal number of tables to obtain the new x and y values for the new zone and continue on until the next interrupt. The analog DMA device requires the longest time to complete its cycle of operation. Its cycle-complete interrupt is utilized to indicate the completion of table freshening, and releases the table to program calculation for preparing control words to be transmitted during the next 3-ms cycle.

OPERATOR CONTROL

One of the most difficult areas of endeavor in an on-line control system is to insure that the operator has control of the situation and not vice versa. One of the major purposes in developing this system was to lessen the burden on the operator and create a more flexible tool for his benefit. It is proposed for Phase II of this systems development, that the control loop be closed on the beam position (via suitable sensors) throughout the extent of the external-proton-beam facility. At present, however, the loop is closed through scintillators, television monitors, and the operators. That is, the entire system starts out as a single knob in the hands of the operator. By a series of push buttons and graphic displays (Figure 2), an operator may request a specific magnet by its name, i.e. X2Q7A, and may place a control request, which can be either as a request to change "BR" of Eq. (3) or, as a dI/dt request to change "S" in (4). In either case, the knob is used to insert a change in the proper direction. To closely define the region of the acceleration cycle in which the change will take place, the operator also has defined the zone by switch selection and light-panel response. That is, a lighted numeral physically located on a drawing of the magnet response curve clearly indicates the zone under control.

Any graduations on the control knob are meant only for the convenience of the operator to keep track of how far he has turned the knob. After exercising control of any magnet in any zone, before losing the original parameters, the operator must take one of two options. He may "SAVE" the new data and thus throw out the old parameters associated with that magnet in that zone, since the last SAVE, or he may "RESTORE" the parameters to those that were being used just prior to the current control request. This feature allows an operator to escape from an adjustment that has removed all evidence of beam from a scintillator. By periodically writing magnet parameters on disc, rapid set-up retrieval is accomplished to accommodate experimenter or accelerator operating mode changes.

To prevent accidental change in magnet number or zone request during an active control period, these options are disabled during control. When controlling a doublet quadrupole magnet, it has been found that duplex control is desirable and therefore, when the first of a doublet has been selected by the operator, a second knob is connected to the second magnet in the doublet.

SMALL COMPUTERS AS CONTROL DEVICES

The question of small vs large computers to exercise a control function would be better phrased as dedicated vs time-shared. When a decision is to be made as to type of computer to be utilized to accomplish an extensive calculation function, the level of sophistication of command structure and memory size are paramount. Time is important, but not of highest priority. When operating in a control capacity, however, close scrutiny of the update period and the number of controlled devices, together with the extent of the data acquisition required to provide necessary feedback often results in the selection of a computer just large enough to accommodate the active control software. This leads to serious questions regarding future expansion capability and provision for general-purpose data handling.

The point of emphasis here, is that a computer can suffer from "time-saturation". That is, regardless of memory size and calculation capability, a practical machine can perform a finite number of input-output functions in a given time. All the additional capabilities are to no avail. Even a so-called "faster" machine is all too often speedier in internal operations and contemporary with its slower kin in the region of input/output.

These considerations point to the use of dedicated machines to perform the control function, leaving the sophistication of elaborate display and general-purpose use to another machine dedicated to that function. This approach speaks not at all to the "size" of the computer, but rather to a careful appraisal of the extent of data transfer, the complexity of the external interface, and the calculation requirements of the control function.

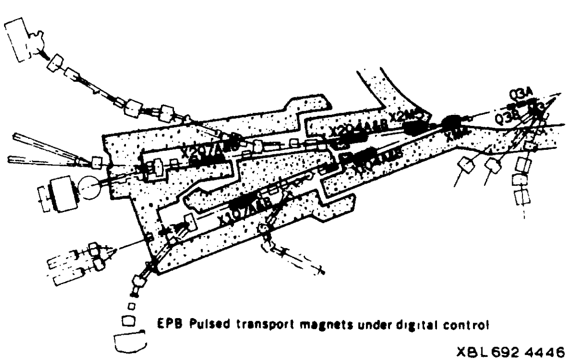


Fig. 1 The EPB Transport-Magnet System

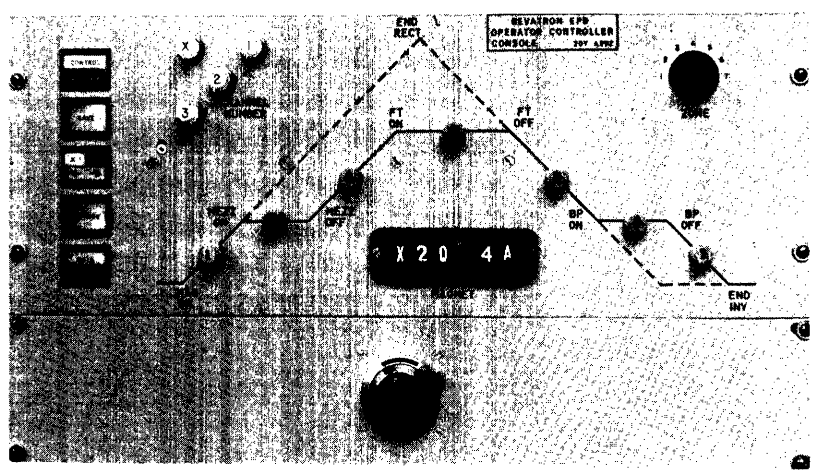


Fig. 2 Operator Control Panel

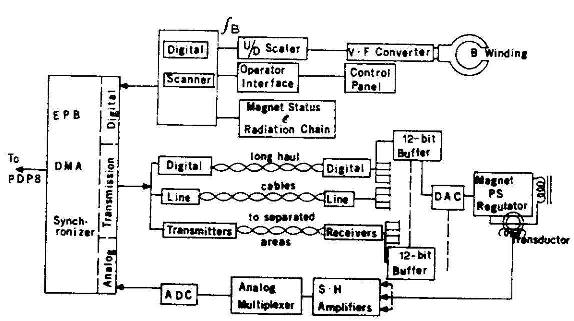
DEDICATED COMPUTER IN COMBINATION

Allowing the above arguments to rest, the question of integrating dedicated machines into a system arises. The PDP-8 in the EPB magnet-control system stands alone in exercising control. To provide backup to continue magnet operation in event of processor failure, a second PDP-8 was obtained. The aim was to provide the operating crew at the Bevatron with monitoring and diagnostic information relating to other facets of the accelerator. This is provided in a general-purpose data-acquisition subsystem without tying the second machine down so as to endanger its backup role in magnet control.

Complete I/O bus multiplexing is provided between PDP-8's (Figure 4) such that any device can be connected to either computer. The only completely separate computer-related hardware is the interrupt and clock devices. This technique provides time-sharing capability in accessing magnet disc for data storage or retrieval as well as magnetic tape and display. Each multiplex switch has busy flags to prevent stealing when inappropriate, but not when priority warrants the theft.

ACKNOWLEDGMENTS

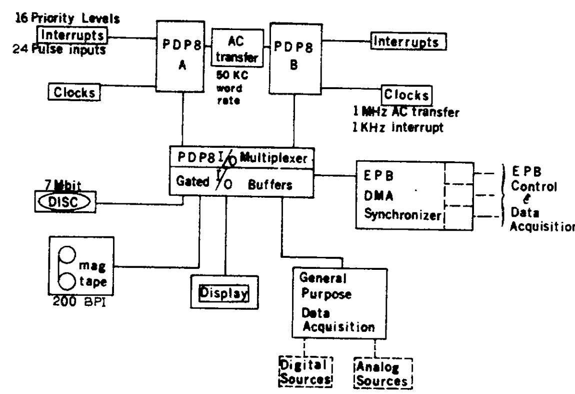
The development of the EPB magnet control system is due to the many contributions of R. A. Belshe, J. B. Greer, J. R. Guggemos, C. D. Pike and T. M. Taussig.



External Proton Beam Transport Magnet Control System

XBL 692 4450

Fig. 3   EPB Transport-Magnet Control System



Basic Duplex Computer System

XBL 692 4449

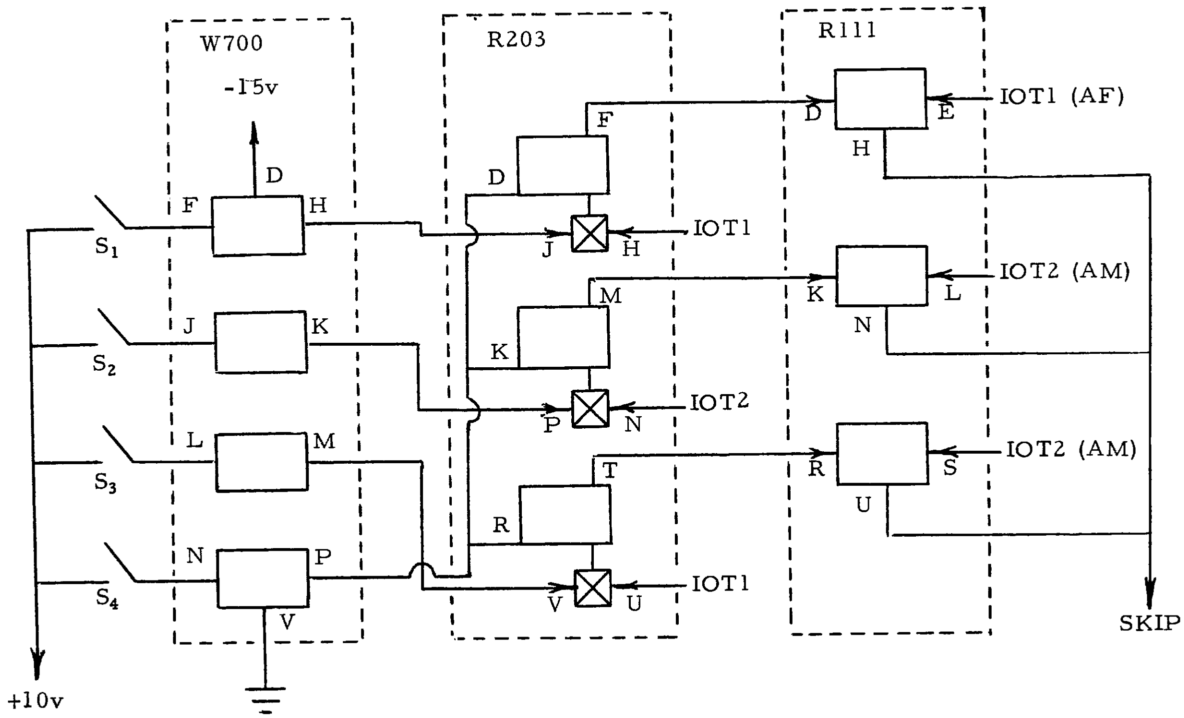
Fig. 4   Basic Duplex Computer System

REMOTE OPERATION OF A PDP-8 COMPUTER

Kenneth B. Wiberg
Department of Chemistry
Yale University
New Haven, Connecticut

In using a small computer for on-line data collection, it is often convenient to be able to operate at a reasonable distance from the computer (e.g. in an adjacent room). If control over the computer is to be exercised during the course of the experiment, it is frequently necessary to place a second Teletype adjacent to the experimental setup. This is expensive and in some ways inconvenient.

Control can also be exercised using remote switches, particularly in response to questions displayed on an oscilloscope adjacent to the instrument in question. This is a very quick and easy mode of operation. An inexpensive switch input may be constructed according to the following diagram.



This provides three levels of direction (S_1, S_2, and S_3) which might be designated, for example, as YES, NO and CONTINUE respectively. The fourth switch (S_4) is a reset to clear the flip-flops. The function of the fourth switch could be combined with the other switches using dp dt switches and additional sections of the W700 switch filter so that release of the switch will clear the flip-flop. However, we have found the four-switch operation to be convenient since, if a switch is pressed prematurely, the operator still has a chance to properly set the instrument before pressing the release switch S_4.

A typical section of program utilizing this switch might be:

```
        :
BEGIN,  CLA
        DCA I XVAL
        TAD I ORDI
        DCA I VALU
        TAD LIST
        JMS I GIAN
        6703
        JMP BEGIN
        JMS SWIN
        :
SWIN,   0
        CLA CMA         / SET AC TO -1 FOR SW1
        6701
        JMP .+2
        JMP RELEAS
        IAC             / SET AC TO ZERO FOR SW2
        6702

        IAC             / SET AC TO ONE FOR SW3
RELEAS, 6703            / HAS SWITCH BEEN CLEARED?
        JMP .-1         / NO: LOOP
        JMP I SWIN      / YES: RETURN
```

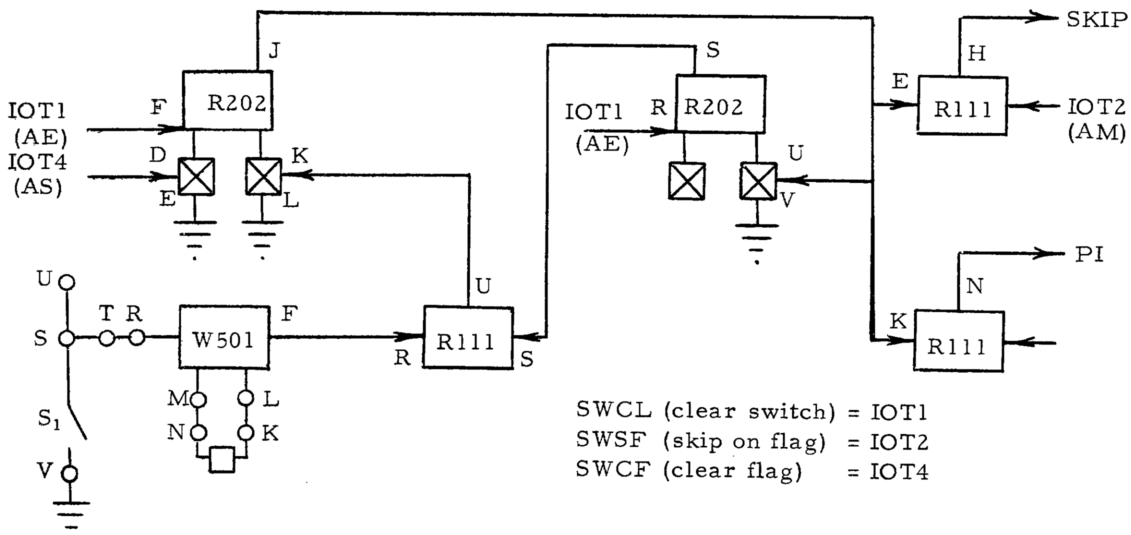
The section of the program starting with BEGIN sets up the locations needed for the oscilloscope output subroutine GIAN (contained in DECUS 5/8-23). After the list has been displayed, the switches are tested (here given the device code 70). If no switch has been activated, the list is displayed again. When a switch is activated, the program jumps to the subroutine SWIN which determines which switch has been used and leaves the code in the AC (-1 for SW1, 0 for SW2 and +1 for SW3). Then it enters a waiting loop until the switch has been cleared using SW4. (It may be noted that two IOT pulses are required to activate the IOS, the first sets the flip-flop and the second provides the skip command.)

Many experiments require time synchronization, and this also may be provided using a simple arrangement:

Here, the IOT1 pulse clears both R202 flip-flop and makes them ready to be activated by a signal from the W501 Schmidt trigger. The IOT2 pulse checks the flag which is set by closure of switch S_1. Closure of S_1 sets the first R202 flip-flop and this in turn sets the second R202 flip-flop. The IOT4 pulse is used to clear the first flip-flop. However, since the second is not cleared, the input R111 gate is disabled and further closure of S_1 has no effect. A typical series of statements making use of this switch is:

SWCL
SWSF
JMP.-1
SWCF

The third element of a remote control center is a facility for entering numbers into the computer in response to questions displayed on the oscilloscope. This may be done by reading the contents of binary-coded decimal switches into the accumulator via R123 input bus gates. We use one switch for the sign, four switches for the number and one switch to indicate the position of the decimal point. The switches are read as two words. A subroutine for interpreting the contents of the switches, inserting a BCD representation of the number into a TEXT string for CRT display and converting the number into floating point form will be submitted to the DECUS Library.



SWCL (clear switch) = IOT1
SWSF (skip on flag) = IOT2
SWCF (clear flag) = IOT4

PDP-8 PROGRAMMING NOTES

DISAPPEARING DECTAPE BOOTSTRAP

Jimmy C. English
Lawrence Radiation Laboratory
Livermore, California

It is often convenient to have a disappearing bootstrap loader so that the monitor can be loaded on the last page without disturbing any other core locations. The following program is less than half as long as the DEC Bootstrap loader which makes it especially applicable to a hardware loader. The program works on version 8E of the System Builder, but not on version 8G. Version 8G cannot be patched because the program uses locations 147-151, however; if any future versions of the System Builder reserve these locations, the monitor bootstrap could be shorter, simpler, as well as disappearing.

I Completely Disappearing

Loc	Contents		
7745	1353		TAD MVB
7746	6766	DO,	DTXA DTCA
7747	6771		DTSF
775Ø	5347		JMP -1
7751	1354		TAD WC
7752	5346		JMP DO
7753	Ø6ØØ	MVB,	Ø6ØØ
7754	Ø22Ø	WC,	Ø22Ø
7755	7577	CA,	7577

The DECtape System Builder version 8E, must be modified as follows:

	Loc	Contents	Change To
BLOCK Ø	147	ØØØØ	6771
	15Ø	ØØØØ	5347
	151	ØØØØ	52ØØ

These changes should be made on the DECtape after the system is built.

THE USE OF FORTRAN SUBROUTINES IN PDP-8 FORTRAN (4K)

Robert Corbett
C. L. McLane High School
Fresno, California

Frequently, it is desirable to use subroutines with PDP-8 FORTRAN. The simplest method by which this may be done is to link a PAL subroutine to the FORTRAN program by means of a PAUSE statement. This method, however, is undesirable when complicated arithmetic manipulation is used in the subroutine. To overcome this fault, the following method was developed to allow a single FORTRAN subroutine to be called:

1. Punch the FORTRAN main routine and subroutine on separate tapes. When a subroutine call is desired punch "PAUSE 1"; when a return is desired type "PAUSE 4". Do not use an END statement in the main routine.

2. Compile the main routine. When the end of the program is reached halt the computer. Examine the contents of register 76. Remember this number!

3. Place the subroutine in the reader and restart compilation at Ø2Ø4.

4. Punch and assemble the following PAL routine.

```
XXXX= the contents of register 76 plus one
*1
         Ø
         JMP I CONT
CONT,    3256
         Ø
         TAD TEMP
         DCA 14
         JMP I 4

*3256
         TAD 14
         DCA TEMP
         TAD XXXI
         DCA 14
         JMP I 4
XXXI,    XXXX
TEMP=Ø
$$$$$
```

In order to use the routine, load the FORTRAN Operating System and the above PAL routine with the binary loader. Then load the FORTRAN routine as a normal routine.

NOTE: This method will not work on the PDP-5 and will cause unusual conditions to result if certain errors occur.

FOCAL PROGRAMMING NOTES

Thomas L. Nichols
U.S. Army Natick Laboratories
Natick, Massachusetts

The following patch allows FOCAL W programs to use the LINC-8 analog to digital converter.

Calls to the function are in the format:

SET A=FADC(K)

where A is any FOCAL variable, and K is the decimal number of the channel to be sampled. The function returns values in the range of -256 to +255, corresponding to minus and plus 1 volt respectively.

Execution of each function call requires approximately 10 milliseconds, limiting the maximum sampling rate to 100 SPS.

Minor changes to the patch will allow the execution of any single LINC instruction (stored at LINSTR) by FOCAL programs. Such a function could be designated FNEW by placing its starting address in FNTABF + 15 (location 413 in FOCAL W).

```
                /SAM PATCH FOR FOCAL
                /8-68.
                /TLN 28 FEB 69
                *27
0027    5200    5200
                *404
0404    5201    5201 /ADC FUNCTION
                EFUN3I=100
                INTEGER=52
                FLAC=44
                IACF=6175
                ICON=6141
                ISSP=6165
                IAAC=6171
                *5201
5201    4452    XFADC, JMS I INTEGER
5202    1377    TAD (100 /SAM INST
5203    3370    DCA LINSTR
5204    4212    JMS LINCDO
5205    7200    CLA
5206    6171    IAAC
5207    3045    DCA FLAC+1
5210    4232    JMS  FLOAT
5211    5500    JMP I EFUN3I
                /DO 1 LINC INSTR.
5212    0000    LINCDO, 0
5213    7200    CLA
5214    1376    TAD (1370
5215    6165    ISSP /SET LINC P
5216    7200    CLA
5217    1375    TAD (3
5220    6175    IACF /SET IBI
5221    7200    CLA
5222    1374    TAD (10
5223    6141    ICON
5224    1373    TAD (2
5225    6141    ICON /GO DO LINSTR
5226    7000    NOP
5227    1372    TAD (-1
5230    6141    ICON /DESEL LINC
5231    5612    JMP I LINCDO
                /SR TO FLOAT FAC
5232    0000    FLOAT, 0
5233    3046    DCA 46
5234    1371    TAD (13
5235    3044    DCA 44
5236    4407    JMS I 7
5237    7000    FNOR
5240    0000    FEXT
5241    5632    JMP I FLOAT
                *5370
5370    0000    LINSTR,0
5371    0013    $
5372    7777
5373    0002
5374    0010
5375    0003
5376    1370
5377    0100
```

PDP-7/9 PROGRAMMING NOTES

A FAST, SMALL SUBROUTINE TO ZERO ARRAYS IN PDP-7/9 FORTRAN

Garry A. Flint
Langley Porter Neuropsychiatric Institute
San Francisco, California

The DO loop code generated in PDP-7/9 ADVANCED SOFTWARE FORTRAN is expensive in both execution time and core space when used to zero arrays. SUBROUTINE ZERO occupies only 35 machine words and zeros 1, 2 and 3 dimensional vectors at least 7 and in extreme cases more than 68 times faster than with FORTRAN code. When using more than two DO loops to ZERO arrays of any size, SUBROUTINE ZERO is also very economical in terms of space savings. This subroutine has been submitted to the DECUS Program Library.

PROGRAMS FOR MASKING AND PROCESSING NONSTANDARD PAPER TAPE INPUT TO PDP-7/9 FORTRAN

Garry A. Flint
Langley Porter Neuropsychiatric Institute
San Francisco, California

Many data collection devices produce paper tape output with a format unacceptable to the standard FORTRAN read instructions. Program TREAD, and bit manipulation functions IAN, IOR, NDTO, and LCS, allow paper tapes of any format to be processed in PDP-7/9 FORTRAN. The functions also enable the user to pack or unpack strings of bits in any format. Here is a description of these programs:

SUBROUTINE TREAD (IDATA(1), NWORDS, IS) (43 words)

This subroutine reads paper tape in any format. Each row of punched holes or blank tape is read into one machine word (bits 10 to 17). IDATA(1) is the first word of the input vector. The first two words of IDATA are used by FORTRAN. The paper tape data starts in IDATA(3). NWORDS is the number of paper tape characters to be read. IDATA must be dimensioned at least IDATA(NWORDS + 2). TREAD returns either when read is initiated (IS = 0), or when read is completed (IS = 1). Thus, the user can double buffer the input data. The end of the tape can be detected by a 6 in bits 15 to 17 in IDATA(1) or by the user's criterion determined from the data input.

The program below shows the employment of the double buffering option in the analysis of paper tape.

```
      DIMENSION  ID(204)
C     PROGRAM TO BUFFER PAPER TAPE INTC ID
C     THE MAXIMUM BUFFER SIZE IS 100
C     SET THE BUFFER SIZE, N
      N = 64
```

```
C     LOAD THE TOP BUFFER AND WAIT
      CALL TREAD (ID(1),N,1)
C     BRANCH TO SET PARAMETERS TO ALTERNATE BUFFERS
C     SET TO START READ INTO BOTTOM BUFFER
      ISW = 2
10    GO TO (20,30),ISW
C     CHANGE SWITCH
20    ISW = 2
C     SET FIRST CELL OF TOP BUFFER
      M = 1
C     SET LIMITS OF DATA IN BOTTOM BUFFER
      NL = N+5     (2+N+2+1)
      NT = 2*N+4   (2+N+2+N)
      GO TO 40
30    ISW = 1
C     SET FIRST CELL OF BOTTOM BUFFER
      M = N+3    (2+N+1)
C     SET LIMITS OF DATA IN TOP BUFFER
      NL = 3
      NT = N+2
C     INITIATE READ AND CONTINUE
40    CALL TREAD (ID(M),N,0)
C     ANALYSIS LOOP ON DATA IN INACTIVE BUFFER
      DO 100  I = NL,NT
      IN = ID(I)
C     CHECK FOR END OF DATA CODE, 177(8)
C     NDTO IS DESCRIBED BELOW
      IF (IN.EQ.NDTO(000,177)) GO TO 60
      . . . . . . . . . . .
      ANALYZE PAPER TAPE DATA
      . . . . . . . . . . .
100   CONTINUE
C     RETURN FOR NEXT BUFFER LOAD
      GO TO 10
60    TERMINATE ANALYSIS
      . . . . .
      END
```

FUNCTION NDTO (N,M) (33 words)

By means of NDTO one can set up octal masking words in FORTRAN. The first three octal digits are defined by N and the last three octal digits are defined by M.

The statement:

IN = NDTO(070,177)

results in the octal number 070177 being stored in IN. This is the equivalent to DATA IN/ 28799/.

FUNCTION LCS (N,M)* (32 words)

LCS shifts the contents of location N, M bits in a left circular rotation. This function permits efficient packing and unpacking of information in a machine word. The statement:

IA = LCS (N,M)

generated a four-word calling sequence that is replaced with

```
LAC N
CLL
CLQ!LRS 22 - M
OMQ
```

which are executed upon return. Note that the value of M remains fixed after the first call to the LCS function. Here is an example:

If	IA = 207703 (octal)
and	IA = LCS (IA,6)
Then	IA = 770320 (octal)

FUNCTION IAN (N,M) (26 words)

The calling sequence is replaced by the following sequence of instructions:

```
LAC N
AND M
NOP
NOP
```

The generated code executes a logical AND on return.

FUNCTION IOR (N,M) (26 words)

The calling sequence is replaced with the following sequence of instructions:

```
LAC N
LMQ
LAC M
OMQ
```

which are executed on return. The execution of this function results in an inclusive OR such that 0 0 = 0, 1 0 = 1, 0 1 = 1, and 1 1 = 1.

*The idea for LCS originated from a similar program written by J. P. Chandler and G. P. Eckley for the CDC 3600, Indiana University.

These functions for packing and unpacking data in FORTRAN are economical in space and time, and are also useful in compacting data for storage.

Here are a few examples:

Example 1

If events for independent data sources were coded in real time on paper tape in the following FORMAT,

Bit 8 - 0 = data character, 1 = time character
Bits 6,7 - data source 0, 1, 2, and 3
Bits 1-5 - events 0 - 31

then the data can be read into a vector called IND and decoded in the following way:

```
C     SET UP MASK FOR THE EVENT CODES
      MSK = NDTO(000,037)
      DO 100 I=1,M
C     SEPARATE TIME AND DATA EVENTS
      IF (IND(I) - 128)20,10,10
10    (INTERPRET TIME CODE)
      . . . . . . . . . . . .
      GO TO 100
20    N = IND(I)
C     GET EVENT AND SAVE IN IE
      IE = IAN(N,MSK)
C     ROTATE IN 13 BITS SO THAT
      N = LCS(N,13)
C     THE SOURCE CODE CAN BE MASKED OFF
      IN = IAN (N,3)+1
C     BRANCH TO THE APPROPRIATE ANALYSIS
      GO TO (11,12,13,14),IN
      . . . . . . . . . . . .
      ANALYSIS
      . . . . . . . . . . . .
100   CONTINUE
```

Example 2

Suppose that, in the analysis program, we want to separate the time between events for each data source and construct vectors for each source where the time between events is stored in Bits 0 to 14, and the event occurring at the end of the time interval is stored in Bits 15 to 17. If the event and time for source 1 is IE1 and IT1 respectively, then

```
C     MOVE THE TIME LEFT THREE BITS
      IT1 = LCS(IT1,3)
C     AND 'OR' THE EVENT AND STORE THE NTH EVENT
      IS1(N) = IOR(IT1,IE1)
      N = N + 1
```

SOFTWARE FOR SALE OR LEASE

Submitted by: Infotec, Inc.

Infotec, Inc. offers for sale an assembler for the PDP-8 computers which operates on the IBM-1130 computer. Source programs in PAL-III language are normally punched into cards and read by the 1130 card reader. The 1130 program assembles the PDP-8 program, prints out diagnostics, lists the DEC program on the printer and punches out a paper tape for input to the DEC binary loader. Sale price is $1,500. Those in the vicinity of Rye, New York, may use the Infotec service bureau at a cost of five cents per printed line of assembly listing.

For further information contact:

Michael J. Kelly
Infotec, Inc.
22 Purchase Street
Rye, New York 10580
Tel: (914) 967-1325

Submitted by: Strategic Time-Sharing, Inc.

Strategic Time-Sharing, Inc. of New York City is currently providing time-sharing services (STIDAC - STI Direct Access Computing) on the first available commercial TSS-8 system. Languages provided include BASIC, FORTRAN, FOCAL, ASSEMBLY, DDT and EDITOR.

STI's time-sharing computers have 24K words of core storage, operate with a cycle time of 1.5 microseconds, and have the capability of accommodating up to 32 simultaneous users. A high-speed swapping disk with 17 milliseconds average access time is used for system programs and active users programs. Large bulk storage is provided by removable disk packs, each of which holds 8.2 million characters. Magnetic tape and line printers are also available. The printer may be used for lengthy printouts which are not practical to transmit over telephone lines to the customer's terminal.

STIDAC prices are as follows:

$6.50/hour terminal connect time
$ .03/unit CPU usage
$1.75/unit storage - 2048 characters/unit

Unlimited use and full-time accessibility are available to heavy time-sharing users at a cost of only $1000 per month for a dedicated port (or channel) into the computer. STI will also operate complete systems dedicated to specific users or industries, for those who need a computer or communications system but want to avoid the programming and operational problems associated with an in-house data processing system.

In addition to conventional time-sharing services, STI also offers what it calls "On-Time Processing." Data is entered into the computer from an on-line terminal, which permits the file information to be queried on a real-time basis. The reports or other information is processed later and supplied to the user at the time he requires--be it the next day, week, or month.

For more information contact:

Strategic Time-Sharing, Inc.
132 West 31 Street
New York, New York 10001

or

Call (212) 736-6266

DECUSCOPE HAS BEEN PUBLISHED SINCE APRIL 1962 AND IS THE OFFICIAL NEWSLETTER FOR DIGITAL EQUIPMENT COMPUTER USERS SOCIETY.

IT IS PUBLISHED PERIODICALLY AT THE DECUS OFFICE, DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.

TELEPHONE: AC 617, 897-5111 EXT. 2414

EDITOR: MRS. ANGELA COSSETTE, DECUS

CIRCULATION: 5,300 COPIES PER ISSUE

PROGRAM AVAILABLE FROM AUTHOR

Computer - LINC-8

Title: Wisconsin Programmed Medicine Interviewer (WPMI)

Author: Dr. W. V. Slack and Lawrence J. Van Cura, University of Wisconsin Medical Center, 1300 University Avenue, Madison, Wisconsin

This is a conversational program written for use with the Laboratory INstrument Computer (LINC). It is designed for verbal interaction between computer and people involved in the practice of medicine--patients and medical personnal. Patient interviewing and patient instruction (operations traditionally performed by physicians and nurses) have been conducted on the LINC with WPMI. The program can also be used for interviewing and instruction in non-medical situations.

Generally, the program operates as follows: questions and statements are displayed on the cathode-ray screen and responses are made on the computer keyboard. Printed summaries of patients' histories and physicians' physical examination findings are generated by Teletype upon completion of each interview. All responses are saved on magnetic tape for future computer processing both for patient care and clinical research.

There are 3 frame formats used in WPMI. These are designated mode 1, mode 2 and mode 3 and differ on the basis of the responses available to the respondent. Mode 1 frames have 4, fixed, mutually exclusive multiple choices--"YES," "NO," "DON'T KNOW" and "DON'T UNDERSTAND" which are numbered 1, 2, 3 and 4 and displayed beneath the question.

With mode 3 frames, the interview writer can set up his own 4, mutually exclusive responses. Otherwise, mode 3 and mode 1 frames are identical.

The computer, with its capability of rapid branching, can present questions and statements as a function of the responses made. Thus, the following are possible--questions presented multiple times with varied wording before an affirmative or negative response is accepted; "YES" answers followed by complex groups of questions in turn involving and branching and designed to cover the item in depth; and "NO" answers resulting in the instantaneous skipping of irrelevant questions. All of these enable information to be elicited from respondents in great detail.

Further, the computer can exert a useful control over the interviewing process. Programmed to proceed only after an appropriate response; to reinforce the respondent's progress through the interview with meaningful words of encouragement; to explain and teach the meaning of concepts not understood and to delve further into the subject of a question whose answer is initially unknown; the computer can increase the likelihood of successful completion of the interview with valid data having been obtained.

For more detailed information contact either Dr. Slack or Mr. Van Cura at the address given above.

TP-1351 CARTRIDGE TAPE UNIT

P. M. Aebersold
Tennecomp, Inc.
Oak Ridge, Tennessee

Tennecomp's inexpensive cartridge magnetic tape unit is a high-speed I/O device capable of replacing most paper tape I/O. Read and record operations transfer approximately 200 twelve-bit words per second between the tape and computer memory. The unit is extraordinarily flexible in that all operations are software controlled including formatting of information stored on tape. Cartridges are standard continuous-loop audio type loaded with high-quality 1/4 inch digital tape verified by Tennecomp to be error free. Record operations are file protected to avoid accidental destruction of valuable programs or data.

The TP-1351 allows rapid access to programs; the user merely selects the cartridge with the desired program, sets a switch to the desired program track, and starts the read operation as he would the binary loader. Editing and assembling are speeded up orders-of-magnitude by using the TP-1351 for storage of symbolic text. Data storage and retrieval are efficiently handled without operator intervention by file programs, and users may recall selected data sets by code word identification.

HARDWARE

The TP-1351 includes the tape transport, interface, connecting cables, and tape cartridges. The tape transport is a heavy duty, 7 1/2 inch per second audio type with a four channel read/write head. Track selection is by means of a four-position rotary switch, but an option is available allowing software as well as manual track selection.

The interface consists entirely of DEC Flip-Chip modules and installation requires only inserting the interface into the standard I/O bus. Serial information read or written by the tape unit is transferred to or from the computer by means of IOT pulses and the skip facility. Instructions are provided for sensing a recorded bit, for recording a bit, for sensing the reflective load point marker, for starting or stopping the unit, and for selecting read or record mode of operation. These instructions are all obtained from one device selector card by microprogramming.

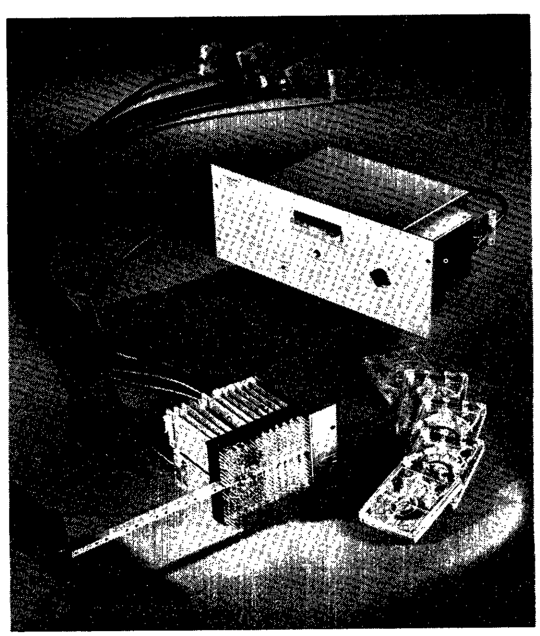
Continuous-loop tape cartridges with load point markers are provided in lengths of 25 seconds (4K per track for convenient program access) and 100 seconds (16K per track). Cartridges have provisions for inserting a small plastic pin which enables the write circuitry. Any attempt to write on a cartridge without an inserted pin turns on a warning light on the front panel of the transport and fails to alter previous contents of the tape.

SOFTWARE

Program read and record routines reside simultaneously in the last page of computer memory. Recording a program requires initial and final addresses to be supplied in a manner similar to using the binary punch routine, while reading takes this information from the tape. Both operations calculate a checksum, and the read program compares its checksum with the recorded one. A read-compare routine allows recorded programs to be verified without chancing the loss of core information.

The Symbolic Editor and PAL-III Assembler have been modified to use the TP-1351 as a text storage medium. Standard high-speed paper tape I/O options have been replaced with magnetic tape I/O routines since most TP-1351's are supplied to minimum systems users.

A complete software package for file operations is supplied in symbolic form for relocation according to user needs. Data file operations have calling sequences specifying initial and final addresses for data transfer and keep checksum tabulations for error detection. Protection against reading or writing past the load point marker is built into all operations. Under control of the user's calling sequence, data files may be recorded



with identifying code words and searches may be made for a file with a particular identifier. In addition, a skip routine allows a number of files specified in the calling sequence to be passed over, and a load point routine finds the reflective marker for initialization of recording or searching operations.

ACCESSORIES

An automatic loader for the PDP-8 series, Tennecomp's TP-1346, features push button loading of any program up to 56 words long. In conjunction with the TP-1351 tape unit, the automatic loader bootstraps in the read program for the tape and starts it up. The read program can on option start up the program just read from tape. Thus, regardless of the state of the computer memory, users are able to select any program on tape and have it operating in less than a minute by pushing a single button.

INFORMATION

For further information, contact:

P. M. Aebersold
Tennecomp, Inc.
P.O. Box J
Oak Ridge, Tennessee 37830
(615) 482-3491

LETTERS

"Mrs. Angela Cossette
DECUS
Maynard, Massachusetts 01754

"Dear Angela:

"If it is convenient, you might mention something in the next newsletter about the group I am working with to code the FOREGROUND/BACKGROUND/8 operating system.

"At present I am distributing a short periodic letter* to each of the people who have shown interest in the system, mainly for internal communication about the design and coding of the system. I would welcome others who might be interested. It would really help us to "recruit" if the Foreground/Background/8 paper were published in the newsletter, as well.

"At present there are about 11 people on the list, and I could use more. I think that this approach might be a workable one for other DEC users, who would band together in a coding effort for some special purpose software usable by each member.

Sincerely,

John Alderman
Georgia Institute of Technology
Nuclear and Biological Sciences Division
Atlanta, Georgia 30332"

*The letter referenced in the above, follows.

"Dear Sir:

"Please excuse the form letter. I have had a very strong response to my appeal for programming effort from users in development of the Foreground/Background/8 (F/B/8) operating system. From comments received at the meeting, it is apparent that the F/B/8 system has wide appeal. Thus, I would like to elicit some additional information from each of the prospective users as to computer configurations, and features desired for such a system:

1. Should the Disk/DECtape monitor be preserved in the system? If not, what other system is desirable? How about SERF as the language? Other suggestions?

2. Should the F/B/8 system operate with other peripherals and configurations than stated in the paper? It might be possible to make it run in a 4K machine, with much slower service for the foreground program. How about use of DECtape (also slow)?

3. Should there be a "standard" EXECUTIVE, or should there be a source distributed, which may be modified by each user?

4. There has been some interest in a DDT rather than an ODT like debugging language for use with the system. Please comment.

5. How is the communication between the users to be obtained? I am willing to act as a publisher of the informational newsletter of this type, but eventually we should use the DECUS organization for this purpose.

6. How are we to get DEC to maintain the finished package? I am sure that you recognize that the maintenance function for such a piece of software is quite different from the programming function, and must be designed in as part of the overall program from the beginning.

7. How are we to handle the assignment and management of the coding effort? Any piece of the work must be done according to the specifications, and on schedule. Checkout must be carried out in the same spirit of urgency. My plan at present is to have each of the pieces checked out by other users, and recirculate the comments back to the originator. This is not a negligible effort in management, so I would like to get some volunteers for looking after parts of this, as well as the coding.

"I would appreciate your comments on any of the above, and any other thoughts that you might have. I enclose a copy of the original paper, in case you didn't get one. (I just found out that the PDP-12 does not have the TSS-8 trap, sorry.) In about another month, I will write again with some more specific proposals for the group, and hopefully, we can converge on a set of realizable goals, which then may be fulfilled by the community-of-8-users, resulting in the F/B/8 system operating as desired.

Sincerely,

John Alderman
Georgia Institute of Technology
Nuclear and Biological Sciences Division
Atlanta, Georgia 30332"

"Mr. David Westlake
Digital Equipment Corporation
Maynard, Massachusetts

"Dear Mr. Westlake:

Re: Your article in DECUSCOPE 1969, Volume 8, Number 2. PDP-5 and PDP-8 Hardware Differences Affecting Software Interchange.

"On item 1 you wrote correctly the differences between PDP-8 and PDP-5 program counters. However, you mention that "this has no affect whatsoever on software interchange." Is this correct? A PDP-8 program modifying the contents of register 0 would not work too well on the PDP-5. Would you clear this matter up for me since I plan to use on a PDP-5 the DECUS 8-58 one page DECtape routine for TC552 which modifies the contents of register 0.

"I look forward to your reply.

"Sincerely yours,

Louis Siegel
Senior Technical Associate
The University of Rochester
School of Medicine and Dentistry
Department of Radiation Biology and Biophysics
Rochester, New York 14620"

"Mr. Louis Siegel
The University of Rochester
School of Medicine and Dentistry
Department of Radiation Biology and Biophysics
Rochester, New York 14620

"Dear Mr. Siegel:

Re: Your letter of 28 April 1969

"Modifications to location zero on the PDP-8 are normally made in conjunction with interrupt routines. (The DECUS program 8-58 for TC552 DECtape control appears to be an example of such programming.) Since interrupt goes to location 1 on the PDP-5, obviously PDP-8 interrupt programs must be modified before being used on the 5.

"Sincerely,

David A. Westlake
Software Maintenance
Digital Equipment Corporation
Maynard, Massachusetts 01754"

"Dear Mrs. Cossette:

"Users of the Four Word Floating Point Package (Digital 8-20-F) will have noticed the lengthy execution times of the FMPY and FDIV operations in this package (approximately 3.4 msec. in each case).

"Since there is not an EAE version of this package available I have written a short patch which utilizes the EAE to speed up the FMPY operation to approximately 0.9 msec. This patch is of particular use with programs DECUS No. 8-103 A, B, C, D, since the extended functions frequently use the FMPY operation.

"I hope this may be of some help to other users.

Location	Contents	MNEMONIC
		* 6333
6333	0000	MULTIP, 0
6334	3337	DCA . + 3
6335	1362	TAD MP2CON
6336	7425	MQL MUY
6337	0000	0
6340	3364	DCA MPSCON
6341	7501	MQA
6342	5733	JMP I MULTIP

MP2 CON = 6362
MPSCON = 6364

"Yours sincerely,

Bryan D. Young
University Department of Medical Cardiology
Royal Infirmary
Glasgow, Scotland

"Dear Mrs. Cossette:

"An error was found in P. D. Siemens' "A Fast Approximation Method for Finding Logarithms." In calculating the maximum error, the procedure is correct, and for b = 2, the value of E_{max} is :086. However, percentage of error is found by using $E = \frac{Y_1 - Y_2}{Y^1}$ instead of $E = Y_1 - Y_2$. Using this method, the maximum error is at least 16%.

"Sincerely,

Paul F. Sullivan
National Aeronautics and Space Administration
Electronics Research Center
Cambridge, Massachusetts 02139"

"Dear Mrs. Cossette:

"An interface designated as SEL-2 has been designed in the Electrical Engineering Department, University of Saskatchewan to use an IBM Selectric typewriter with a PDP-8 or 8/I computer. This provides such features as higher printing speed (approximately 15 characters per second), two case (upper and lower) capability, ease of altering the character style (the print heads are easily interchangeable), back space and ribbon shift which are not normally available on a Teletype.

"The present version of SEL-2 implements the following five instructions:

a. Skip on Print Flag.
b. Clear the typewriter buffer, print and keyboard flags.
c. Load the typewriter buffer and print.
d. Skip on keyboard flag.
e. Read the typewriter buffer.

"The transfers between the accumulator and the typewriter buffer always take place as 6 bit parallel words. As printing from the computer and typing from the keyboard at the same time are not possible, the same 6 bit buffer is used as print and keyboard buffer.

"When any key on the keyboard is struck, the corresponding 6 bit code is loaded into the buffer and the keyboard flag is set. When printing, the desired 6 bit code is loaded into the buffer and this automatically initiates the print cycle. The print flag is set when the buffer is ready to accept another character for printing. While implementing a carriage return or tab, the print flag is not set until the appropriate interlock switches have been closed in the typewriter to ensure proper printing.

"The print section is operated as a closed loop, i.e. the print magnets are energized by the print instruction and are released only after receiving a feedback signal from the print mechanism, so that variations in timing from character to character do not affect the printing operation. The closed loop operation is also believed to increase the useful life of the typewriter apart from providing the maximum speed because the next print cycle can be initiated before the clutches in the print mechanism are disengaged thus reducing frequency of engaging and disengaging.

"The DEC Symbolic Editor program has been modified so that text editing can be done from the Selectric keyboard. The modifications to the editor program also allow the user to go directly to the DECtape library system from the Teletype keyboard so that the input from the IBM keyboard can be stored on the DECtape as files after editing and retrieved to print on the Selectric without manipulating the console switches.

"This type of on-line typing and editing system is hoped to make the task of preparing documents such as a thesis which requires error free neat typing easier and pleasant.

"Keen interest of Dr. A. R. Boyle and the funds provided by the Canadian Hydrographic Service for this work are gratefully acknowledged.

"Yours faithfully,

R. Krishna
University of Saskatchewan
Department of Electrical Engineering
Saskatoon, Canada"

"Dear Mrs. Cossette:

"I am enclosing a patch to KM9V4B which will enable it to use the CR03B card reader as a batch processing device. Due to limitations in space, the patched version of CDB in the monitor cannot detect if the Hopper is empty and then return to the non-batch monitor. This will not cause much inconvenience though. Also, it has not been tested under API.

"We have also written a standard CR03B card reader handler. I shall submit that directly to the PDP-9 software group in case they wish to release it. I have written a letter to James Murphy in which I informed him that the versions of Time and Time10 are not suitable to European 50 cycle current. You might make a note of this in DECUS.

"Yours in peace,

Avram-Chaiyim Miller
Medische Faculteit Rotterdam
Wytemaweg 2a, Postbus 1738
Rotterdam, Netherlands"

```
        .TITLE PATCH
/
        .ABS
/
CRSF=7Ø6721
CRCS=7Ø67Ø4
CRRB=7Ø6712
CRSB=7Ø6722
/
/
    .LOC 2251
    ION
/
    .LOC 3Ø56
    CRSF     /WAS CRSF
/
    .LOC 3Ø6Ø
    JMP .+4 /WAS SECOND SKP SETUP
/
    .LOC 3143
    NOP      /WAS SKP ON HOPPER EMPTY
/
    .LOC 3155
    SKP      /WAS SKP ON CR READY
/
    .LOC 3162
    CRSB     /WAS CRSB, DIFFERENT READER
/
    .LOC 32Ø4
    CRRB
    RTR
    SZL
    JMP 3223         /CRDONE--CARD DONE INNPT
    AND 353Ø         /AND (7777
    CRCS     /CLEAR FLAG CAUSED BY COLUMN DONE
/
    .LOC 3223
    CRCS     /CLEAR FLAG CAUSED BY CARD DONE
/
    .END
```

PIP V7A

"Dear Mrs. Cossette:

"Recently, I have written several I/O handlers for our PDP-9 equipped with API. Because the interrupt rates for some of the devices concerned were high I was distrubed by the time required to process an API interrupt from a device connected both to the API and PI as described in applications note No. 1 for the Monitors Manual: Techniques for Special Device Handlers. Since the programs concerned would not operate under PI (it is too slow) I looked for a simple way to delete the PI capabilities from my I/O routines without making them vulnerable to undecipherable errors if a PI were to occur for the device concerned. If the IOT skip were just deleted from the skip chain then the .SETUP (CAL followed by 16) would not function properly.

"The solution to the above problem was to construct during SYSGEN a false skip, e.g. 7Ø57ØØ (which never skips) and later use it in the .SETUP for the I/O handler so that should the device cause an interrupt on the PI (not API) an IOPSØ3 would result. The interrupt handler may then use the entry and exit coding given in Figure A instead of the usual coding given in Figure B.

"The shortened handler requires 17 less locations and 29 less micro-seconds to execute. The only loss is the ability to function under PI which is already impossible with the interrupt rates associated with these devices.

"Yours truly,

Steven R. Deller
Department of Surgery
Northwestern University
The Medical School
Chicago, Illinois 60611"

FIGURE A

```
DEVINT      ØØØ             /Enter here by JMS (API instruction)
            DAC SAVEAC#     /Save accumulator
            DEVCF           /Clear device flag
            .
            .
            .
            .
            LAC SAVEAC      /Restore AC
            DBR             /Debreak and restore
            JMP* DEVINT     /Return
```

FIGURE B

```
PIENT    DAC    SAVEAC          /PI save AC
         LAC*   (Ø)             /Save PC link, ex mode,
         DAC    DEVOUT          /and Mem Prtct
         LAC    DEVION          /Force ION
         JMP    DVSTON
DEVINT   JMP    PIENT           /PI ENTRY
         DAC    SAVEAC          /API ENTRY, SAVE AC
         LAC    DEVINT          /Save PC, Link, Ex mode,
         DAC    DECOUT          /Mem Prtect
         IORS                   /check status of PIC
         SMA!CLA                /for restoration at dismissal
         LAW    -40
         TAD    DEVION          /ION or IOF in AC
DVSTON   DAC    DVSWCH
         DEVCF                  /Clear Flag
         ION                    /Enable PIC
         .                      /for other interrupt
         .
         .
         .
DVDISM   LAC (JMP DEVPIC)       /Restore PIC entry
         DAC    DEVINT
         LAC    SAVEAC          /Restore AC
DVSWCH   ION                    /ION or IOF
         DBR
         JMP*   DEVOUT
```

NEWS FROM DEC

LAB-8/L SIGNAL AVERAGING SYSTEM

A new, lower-priced version of the LAB-8 computerized signal averaging system has been announced. Designated the LAB-8/L, the new signal averaging system is built around the DEC PDP-8/L small computer and is priced comparably with the majority of hard-wired signal averagers on the market.

Unlike the hard-wired averagers, the inclusion of the general-purpose computer makes the LAB-8/L highly programmable. The total system includes the PDP-8/L computer, a special analog-to-digital laboratory peripheral device, a multiplexer, a real-time clock, a display control, a paper tape punch and reader, a Teletype, and special signal averaging software. First deliveries are scheduled for this summer.

MODIFICATIONS TO COMPUTERIZED NC TAPE PREPARATION SYSTEM

Modifications providing increased precision and flexibility to its Quickpoint-8 numerical control tape preparation system, have been announced by Digital Equipment Corporation. Introduced here last year, Quickpoint-8 is a software package designed around a DEC PDP-8 family general-purpose computer. It allows the user to define random and/or geometric patterns such as grids, bolt-like circles, and various arrays with symbolic definitions and store and recall these at any location on the piece part.

Among the modifications is the fractional input feature which allows the programmer to mix fractional and decimal quantities freely in a command line equation. Quickpoint-8 also has the new capability to add and subtract while defining geometric as well as coordinate commands.

The system's character editor has also been modified to simplify programming corrections by eliminating the need for complete line erasure when correcting typing errors.

Additional modifications have been made in Quickpoint pattern repeats, error messages, maximum number of computations, Z-axis input, offset commands, and NRT commands.

Quickpoint-8 is particularly suitable in the sheet metal or machine shop. A Quickpoint-8 system outputs a punched paper tape for virtually any point-to-point machine tool. Quickpoint avoids the need for additional manual programmers and saves tape preparation time on very simple jobs through its editing and calculating capabilities. It also virtually eliminates the need for manual computation of point coordinates. The system also includes a larger buffer area and memory capacity to accommodate particularly large NC program preparation.

SPECIAL PURPOSE TIME-SHARING SYSTEM

A new special purpose or single language small computer time-sharing system using the conversational language, FOCAL*, was introduced recently by Digital Equipment Corporation.

The system is designed specifically for educational and engineering applications, but because it lowers the time-sharing terminal cost, it is expected to have a variety of other applications. The maximum number of terminals in the system is seven. With the addition of more core memory and other hardware options, the number of terminals can be increased to 16 or more, and the system made general purpose, giving it the capability of using a variety of computer languages.

The key to the system, which can be installed in any DEC PDP-8, PDP-8/I or PDP-8/L computer equipped with disk storage, is FOCAL. Disk capacity can vary from 32,768 words to more than a million words, depending on user requirements. In all cases, FOCAL permits the storage of programs in a common library on the disk.

The system is designed so that if a user designs a program requiring more than his alloted space in core memory, that program can be broken into segments, and the needed segments stored in the disk storage unit. When it is time to call up this program, FOCAL first removes the segment that is in the core memory, and takes the remaining segments from the disk storage unit, automatically chaining the program together in its proper sequence. There is common storage of up to five variables between segments.

LOWER COST TYPESET-8 SYSTEM ANNOUNCED

A new smaller version of its completely computerized typesetting system, TYPESET-8, which produces punched paper tape containing all hyphenation, justification, and format commands needed to drive typesetting machines was announced. The new system, is built around DEC's PDP-8/L general purpose computer.

The new unit comes complete with typesetting programs and can set hot metal with or without space bands. It has the capability to drive various photo composition machines including the Photon 713, Linofilm Quick, and Compigraphic 4962.

DEC's larger TYPESET-8 system, in addition to the capabilities of the newer system, can also store 300-500 display formats and drive the Photon 513, 560, 713, the Mergenthaler Linofilm, Superquick, and the Harris Intertype Machines Fototronic 480, and 1200.

For more information on the above articles, contact the DEC Sales office in your area.

*Trademark of Digital Equipment Corporation