AD-A061 029 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO F/G 10/2

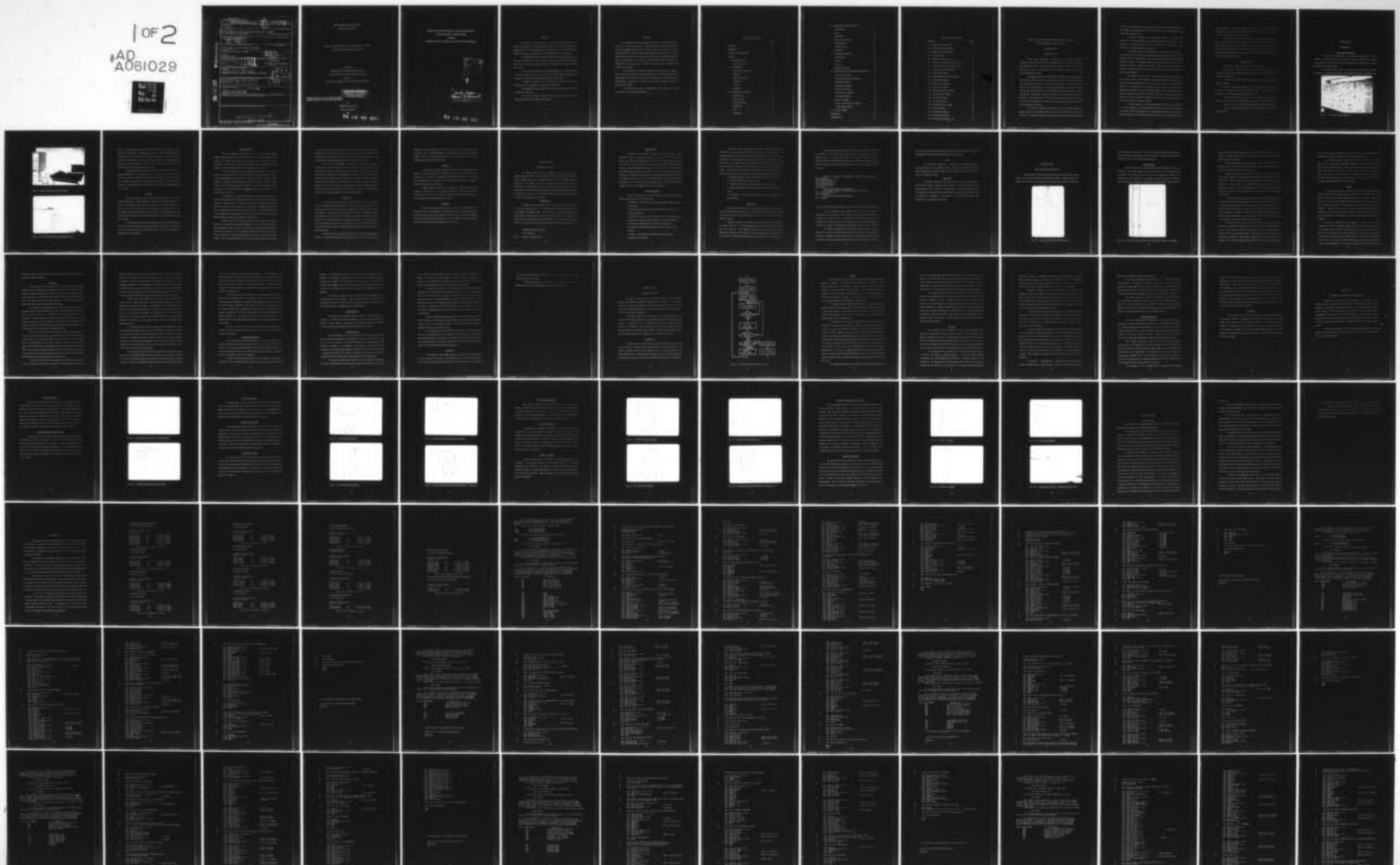
DIGITAL COMPUTER, INTERACTIVE GRAPHICS CONTROL OF AN ELECTRICAL--ETC(U)

1978 L R DAVIS

UNCLASSIFIED AFIT-CI-78-115 NL

1 OF 2

AD A061029



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: CI 78-115	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): Digital Computer, Interactive Graphics Control of an Electrical Power System.	5. TYPE OF REPORT & PERIOD COVERED: Thesis
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Captain Lawrence R. Davis	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: AFIT Student at the University of Oklahoma, Norman, OK	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: AFIT/CI WPAFB OH 45433	12. REPORT DATE: 1978
	13. NUMBER OF PAGES: 126 Pages
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): 134p.	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

LEVEL

16. DISTRIBUTION STATEMENT (of this Report)

Master's thesis,

Approved for Public Release, Distribution Unlimited

DDC RECEIVED OCT 4 1978 F

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

AFIT-CI-78-115

18. SUPPLEMENTARY NOTES

SEP 22 1978

JOSEPH P. HIPPS, Major, USAF
Director of Information, AFIT

APPROVED FOR PUBLIC RELEASE AFR 190-17.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

ORIGINAL CONTAINS COLOR PLATES: ALL DDC REPRODUCTIONS WILL BE IN BLACK AND WHITE.

012 200

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

DIGITAL COMPUTER, INTERACTIVE GRAPHICS CONTROL OF AN ELECTRICAL POWER SYSTEM

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirements for the

degree of

MASTER OF SCIENCE in ELECTRICAL ENGINEERING

By

LAWRENCE RAY DAVIS

Norman, Oklahoma

1978

DIGITAL COMPUTER, INTERACTIVE GRAPHICS CONTROL

OF AN ELECTRICAL POWER SYSTEM

A THESIS

APPROVED FOR THE SCHOOL OF ELECTRICAL ENGINEERING

By

John Fagan

Marion E. Council

C. R. Hader

PREFACE

This thesis is the description of the design of a digital control for an electric power system. The input-output terminal is an innovative, interactive graphics terminal. The techniques and intricasies of such a design are described in detail so the user may make modifications and improvements to the system.

The thesis goes further to propose various power system applications for this type control system. Not only is the system described here an important Power System educational tool but can become a powerful experimental research tool as well.

I am extremely indebted to the foresightedness of Dr. John E. Fagan, my faculty advisor, and Dr. M. E. Council for conceiving such a system and procuring the equipment to make the design realization possible.

I thank Dr. Fagan for his constant help and advice on matters concerning both the design and the preparation of this thesis.

I also thank Dr. Council and Dr. C. R. Hayden for serving as members of my graduate committee.

Last but never least, I thank my wife, Judi, for constant care and understanding while I was engaged in this effort.

ABSTRACT

This thesis presents the details of the design for the digital computer control of an electric power system. The input-output medium is the innovative, interactive graphics terminal. The combination enables a man-machine interface with interaction at a level not achieveable in earlier control systems. The system easily solves the problem of scale encountered with the large power system by storing representations of the system on disk and recalling them as needed. The displays also present the system variables from remote points to enable better monitoring and control.

The collection of displays described in this thesis make possible a powerful and unique research and educational facility. Experimental research on digital control of an electric power system is prohibitive on a real power system. Educationally, it provides the student state-of-the-art training in control of electric power systems.

The information provided is applicable to the design of any digital control system using interactive graphics.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

FIGURES Page

DIGITAL COMPUTER, INTERACTIVE GRAPHICS CONTROL OF AN ELECTRICAL POWER SYSTEM

CHAPTER ONE

INTRODUCTION

Power System generation, transmission, and distribution networks coupled by switching and load center substations make up a system that covers vast geographical areas. Without adequate planning for protection and control during contingencies, this network system may collapse, as was the case in the New England power outages.

Contingencies, such as the removal of a transmission line or generator due to faults caused by weather conditions, may cause a system unbalance that cannot be simply corrected. As the system's automatic controls operate to protect system equipment from overload and destruction, the human contoller monitors and makes decisions as to load shedding, generation adding, or system rescheduling, attempting to keep as much of the system supplying power as possible. If the decisions are in error because of inadequate system protection, controllers lack of information, and/or lack of controller training, the result will be a possible system collaspe.

In an effort to enhance the control of an electric power system, the digital computer and related devices are being used to better control system reactions. The computer's ability to recall instructions and make pre-programmed decisions at the proper instant make it a very effective power system controller. The adaptibility and speed of the computer points to future closed loop operation of the electric power system.

A problem encountered with computer interface is the real time presentation of system configuration and data for the human controller. Also needed is an input medium for real time reaction to contingencies by the controller. The solution is interactive graphics.

Interactive graphics provides near real time communication between a human operator, the computer, and, ultimately, the system being controlled. The controller sees a representation of the system configuration and data that's updated constantly. Any changes desired are commanded by opting to a list of commands or circuit actions on the display. The interface between man and machine is as direct as seeing and acting; interaction at a level not achieveable in earlier control systems.

The problem of scale for the control and monitoring of an electrical power system is not a minor problem. The physical dimensions of most power systems precludes representation on a small scale in a control room environment. However, the compactness of the computer and its ability to store data has enabled the reduction of the system representation and its associated data to a control cabinet and the graphics terminal. Without physical movement, the operator has at his command the complete system on a series of displays, called as needed by pointing to the graphics screen.

Of prime importance to control and monitoring are the current values of a systems variables. For the large power system the collection of data from very remote points and presentation in a meaningful, readable form is nearly an insurmountable task. With graphics oriented control, the task is reduced to a size easily presentable. The graphics displays present remotely sensed data, changed to digital form, at positions on the configuration display representing its origin. The data can be changed to per unit if desired.

It is with this in mind that this project was originated; The Control of a Power System Simulator with a digital computer connected to an interactive graphics terminal. Specifically, the project was to first draw the one-line diagram representations of the simulator and then coordinate the control of the displays to ultimately perform actions on the system. This paper attempts to explain the details of designing such a system.

Chapter Overview

Chapter One contains the introduction to the problem.

Chapter Two contains descriptions of the hardware necessary to accomplish the digital-simulator interface. The limations of the hardware and its reliability are discussed.

Chapter Three contains brief descriptions of the code necessary to build a graphics display.

Chapter Four details considerations, design, and optimizing of the software for completing the one-line diagram displays. Aspects of subpicture numbering and trouble shooting ideas are also presented.

Chapter Five describes in detail the main program that controls and performs actions as directed. It approaches the program in chronological order.

Chapter Six is the results in the form of photographs of the displays.

Chapter Seven concludes with some future applications for this type of control system.

CHAPTER TWO

HARDWARE

Power System Simulator

The power system simulator is an excellent educational tool. Power systems are excessively large and expensive. A physical realization on a small scale enables the student to "see" the basics and perform experiments that would be prohibitive on a real system.

The simulator (Fig. 2.1) is a small power system consisting of scaled generators, a transmission system, sub-transmission system, substations, and

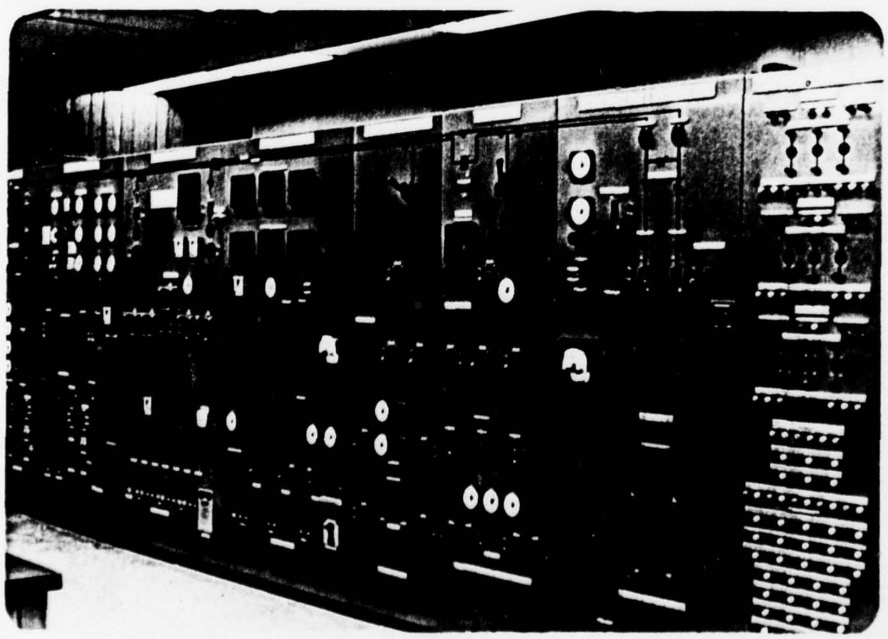


Fig. 2.1. The Power System Simulator.

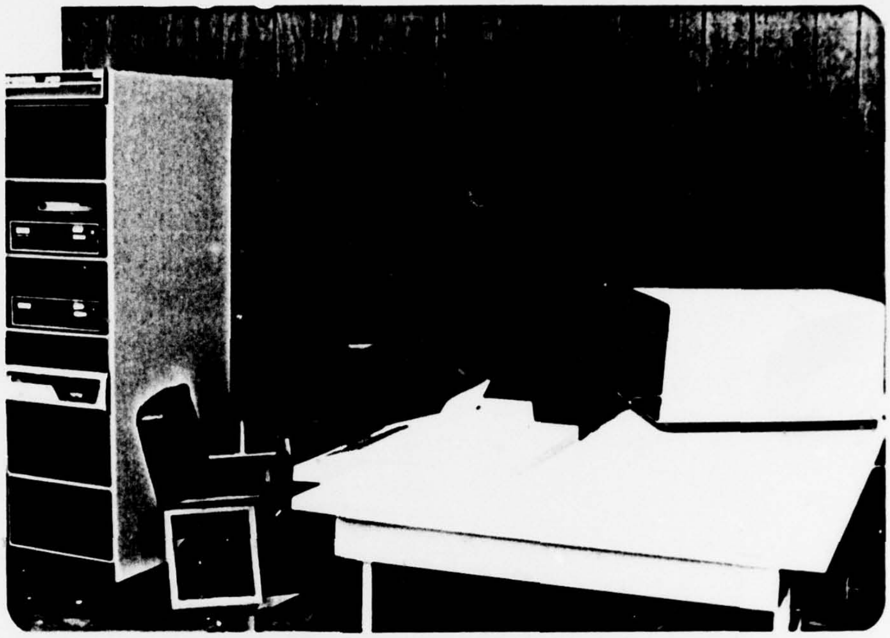


Fig. 2.2. PDP11T34 Digital Computer System.

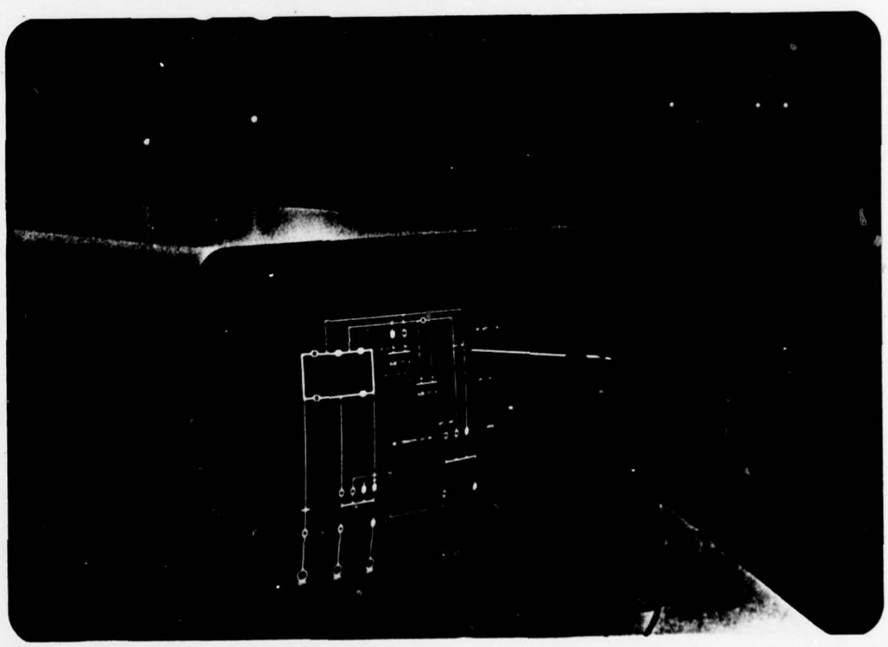


Fig. 2.3. VT-11 and Graphics Grid Coordinate Axis.

loads. The generation is three dc motor/synchronous generator sets of 1 KW, 1 KW, and 3 KW capacity. Additional tie capacity may be supplied through the simulator system interconnection. The high voltage (600v) transmission system connects the generation to two distribution substations where residential and industrial loads are simulated. There is also a network system fed from both substations, and a load center substation as well.

The circuit breakers and certain control switches are accompanied by a special modification. The breakers have a remotely or manually controlled contactor in series. This allows the breaker and circuits to be closed and opened by the computer instructions. The contactor's terminals are routed to a control panel where special buffer circuits change five volt computer originated signals to 115 v signals.

Computer

The digital computer used for the interface and to run the VT11, graphics terminal is a Digital Equipment Corporation PDP11T34 (Fig. 2.2). It includes a DECWRITER typewriter input terminal and two RK05 magnetic disk cassettes. The CPU is connected to the peripherals by a UNIBUS system. The data transfer takes place over this bus. The memory section contains 64K words of semi-conductor memory of which less than 28K is needed to perform the task described here.

The operating system is RSX-11M which has a multi-tasking capability. The task or program described here runs as a single task, however. The graphics terminal doesn't require full time servicing from the CPU, so multi-tasking is possible while it is operating.

Graphics Terminal

The VT11 graphics terminal (Fig. 2.3) is a single color, variable intensity, random position scan CRT terminal for real time graphics display. The screen is 17" diagonal and defines 1023 coordinate points in both x- and y-directions. A light pen for user interaction with the CPU is provided, too. The terminal contains a display processor that is integral to the system.

The VT11 display processor can operate as a peripheral on the UNIBUS system just as any other device. It can also access the memory directly and fetch its data independently of the CPU. The memory accessed is called a display buffer and must exist in the lower 28K of memory. The buffer must be set up by software as an initialization process. The VT11 will also issue interrupts to the CPU when it detects the light pen has ben pointed to a sensitive area of the screen (a light pen "hit").

The light pen is an infrared light detector. When the software directs a screen intensification for the light pen within the detection angle of the pen, and interrupt is communicated. The processor "remembers" the beam position and subpicture number it was tracing when the "hit" occurs. Global flags, positional data, and subpicture number are then passed to the CPU. This information is used as it is programmed to be used by the CPU to change the display buffer. Thus, the graphics display is changed as a response to a light pen hit.

Integral to the VT11 is a group of specialized hardware to accomplish quickly and accurately certain graphics displays. One is the vector generator. The calculations necessary to draw a vector or line between two points is done internally with hardware. The line display is difficult because the beam must follow any slope and light enough points of the square grid pattern to look straight. There is also a problem of accuracy of start and stop points. Since there is some delay between the time the beam is turned off and when it actually disappears on the screen, the delay must be accounted for by the circuitry. To avoid flicker of the presentation, the vectors must also be drawn quickly. Four vector types are available: solid, long dash, short dash, and dot dash.

The text or alpha-numeric characters are generated by special hardware, also. As the text mode is initiated a special sweep generator is set up that sweeps the beam up and down in a small rectangle. A ROM contains digital words for each character so the appropriate dots in the matrix are intensified to finally cause the representation to appear. As an added feature the sweep is slanted with ramp signals to cause the characters to appear as italics. The text generator has 96 ASCII characters and an additional 31 special characters such as Greek letters and math symbols. The variety of characters available enable excellent clarity and distinguishability of textual presentation.

Limitations

There are some limitations to the system, none of which are critical. One is a very slight flicker of the display. Since the beam traces the pattern line by line and point by point, the more that is displayed, the longer it takes to trace. The phosphor on the screen holds this problem to a minimum by holding the display long enough for the next trace to re-illuminate the display. The flicker becomes pronounced with a display buffer size of around 1000 words. This does limit the data to be displayed on one picture. This is easily fixed by using more pictures, each with less detail. However, optimization of the buffer is a necessity and will be discussed later.

Another limitation is compiler and task size. The graphics language is Fortran. The Fortran compiler (8K words) must be resident in memory while compiling. So the programs may have to be sectioned into subroutines to obtain the object files. A similiar problem, to a lesser degree, exists with the task builder or link step. These limitations have been relieved with the recent addition of 32K words of memory for total of 64K.

Interface

The interface circuitry is designed by the University of Oklahoma and built by a number of students. This hardware changes the five volt signals of the computer to 115 volt level commands for the simulator contactors. The analog data sampling hardware interface for ultimate digital display on the graphics terminal has been built by this team.

Analog to digital converters are necessary to convert the analog data samples to digital form. The software subroutine to handle the AR11 A-D converters was written by OU students. Minor changes were made to make it more compatible to the main program and system.

Reliability

All of the system hardware has proven capable of performing the task described herein. Its reliability is demonstrated best by its past performance. There has been only one major malfunction in the past years operation. There has been some difficulties with the light pens. They are unable to take rough handling and if damaged may cause spurious light pen hits to occur.

CHAPTER THREE

GRAPHICS SOFTWARE

The software available for the graphics is provided as part of the computer package by DEC. It is a collection of Fortran callable subroutines that build a display buffer that contains instructions for the display processor. Functions on the display such as line drawing, point placement, subpicture numbering, and text placement are then performed when the task is ran. RSX-11M, version 2 is the operating system used with the more sophisticated version 3 becoming available soon. A description of the subroutines and concepts necessary to complete a display or picture follows.

Initialization

The beginning of each picture must be the creation of a common block to contain the display buffer. It is standard and called DFILE and contains the array IBUF(n), the display buffer. The next call is a graphics initializing call, called INIT(n). Its purpose is to clear the screen, initialize all control flags and system variables, and link the VT11 to the UNIBUS. The INIT(n) call may be used afterward at any point in a program to re-initiate a new display. (See figure 3.1)

```
COMMON/DFILE/IBUF(1100)
CALL INIT(1100)
```

Fig. 3.1. Example of initiation calls.

Display Buffer

The heart of the graphics display is the display buffer. All the information used by the display processor in the VT11 is in the display buffer which is the common array, IBUF (see Fig. 3.1). The dimensioned size of the array is limited only by the requirement to be resident in the lower 28K of memory. However, it should be kept at a minimum to avoid flicker. The dimensioned size of the buffer has no effect on the flicker. It is the actually used portion that is important. This doesn't mean that the array should be dimensioned large, because the memory is reserved, even if its not used, wasting memory you may need. If the buffer size is too small, an error message stating such will be issued to the teletype terminal. The program is now ready for drawing.

Basic Subroutines

The basic subroutines used to build pictures are APNT, VECT, LVECT, RDOT, and FIGR. Each is described briefly:

a. APNT(x,y) - causes the beam to be positioned at absolute point x,y on the screen.

b. RDOT(x,y) - causes the beam to be displaced x and y, relative to the present position.

c. VECT(x,y) - causes a line to be drawn from present beam position, to a point x,y relative to present position. Also called short vector for use in short distances - not accurate for long distance.

d. LVECT(x,y) - same as VECT except used for long lines - more accurate.

e. FIGR( ) - will cause lines to be drawn between relative coordinates in an array list.

Each of the above subroutines have optional integer arguments that determine the parameters for the beam for that point, line, or figure. The arguments appear in the order l,i,f,t and must be included only if a change occurs. A succession of two commas indicates no change for one parameter but a change for later one. If a change is made to the i, f, or t parameter, the prior parameters must be included, in order, or two commas used in each place to represent no change. For instance, to change the f parameter only, the sequence could be (x,y,l,i,f),(x,y,,i,f), (x,y,,,f), or (x,y,l,,f). The meaning of the parameters follows:

l - light pen sensitivity. If zero, stays the same, if greater than zero, it is sensitive, if less than zero, it is not.

i - intensity of point, line, or figure. Scale from one to eight.

f - flash of presentation. Same code as l except replace sensitive with flash.

t - type of line. 1 for solid, 2 for long dashed, 3 for short dashed, and 4 for dot dashed.

Subpictures

Essential to the optimization of the graphics display and manipulation of the the presentation is the concept of subpictures. They are similar to Fortran subroutines but have other characteristics that must be known to successfully program a display.

Basically, a subpicture is some part of an overall picture or nested part of another subpicture. It is a defined group of fortran and graphics statements that draw that part. The SUBP(n) call with one argument is placed at the beginning of the group. The argument n is a unique integer that is associated with the subpicture. The ESUB call defines the end of the subpicture. (see Fig. 3.2)

Once a representation of a figure, for example a transformer, is defined as a subpicture, it may be copied at other points on the picture. The call is SUBP(n,m) where n is a new unique integer and m is the subpicture being copied. ON(n) and OFF(n) routines cause subpicture n to be added or removed from the display. By defining subpictures and turning them on or off, the display is changed as desired.

```
        .
        .
        .
C       THIS IS A SAMPLE OF BUILDING A SUBPICTURE NUMBERED 25.
CALL SUBP(25)
CALL APNT(100.,200.,-1,-4)
CALL VECT(100.,550.)
CALL RDOT(-100.,0.,,-4)
CALL VECT(256.,-300.)
CALL ESUB
C       THE END OF SUBPICTURE NUMBER 25.
C       COPY SUBPICTURE 25 ELSEWHERE AS SUBPICTURE 26.
CALL APNT(200.,452.,-1,-4)
CALL SUBP(26,25)
C       TURN OFF SUBPICTURE NUMBER 26.
CALL OFF(26)
        .
        .
        .
```

Fig. 3.2. Sample of creating subpicture 25 and copying as 26. Then turning off 26.

From an optimal display buffer's point of view, the concept of subpictures is very important. Each subpicture starting place is known to the processor and a jump to this starting address is made when copying. Because of the capability to jump and use parts of the buffer again and again the buffer size is kept at a minimum. The importance is great when dealing with mini-computers.

The OFF(n) subroutine uses the jump instruction also. When a subpicture is turned off the processor jumps to the end of the subpicture and continues. Invariably, the start and end point of the subpicture isn't the same. The results of turning a subpicture off are therefore very humorous if there is a routine containing relative plotting following the turned off subpicture. So, follow all subpictures to be turned off with an absolute plotting call.

Text

The TEXT(XXX) subroutine is called for the presentation of the characters described in the hardware chapter. The arguments are manipulated to present sequential text, special characters, and carriage return. The STAT(n) call enables or disables the italics.

Conclusion

The graphics subroutines are fully described in the manual DEC-11-AMLEAA-D, Graphics Extensions User's Guide. Detailed description of the aforementioned calls and others is contained there. If a person desired to learn the language, the manual contains several faulty examples that are recommended as starters for a beginner. The mistakes in the examples provide ample opportunity for learning trouble shooting.

CHAPTER FOUR

ONE-LINE DIAGRAM DISPLAYS

The beginning of the design process for the interactive graphics control system was to build the displays to represent the power system simulator. In addition to limitations of the hardware, the design had to consider the ultimate

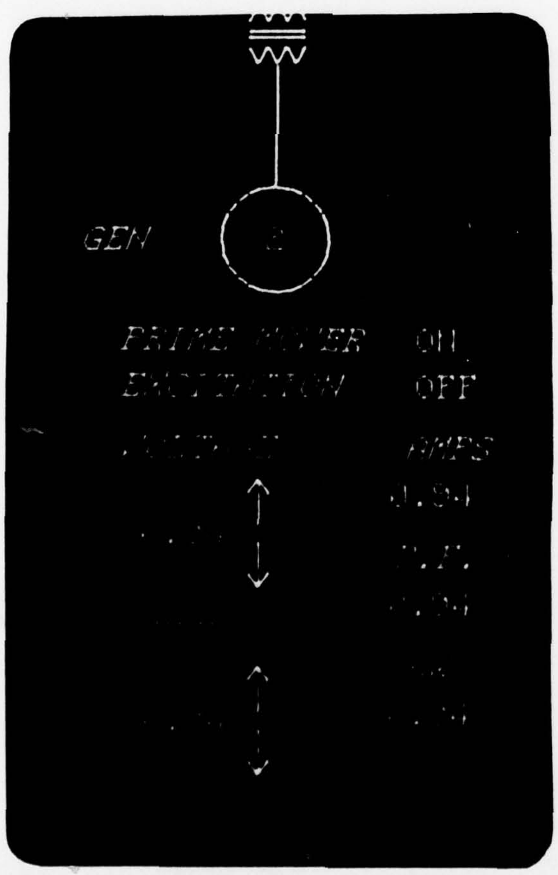


Fig. 4.1. Close-up of Generator Control Section.

user, the student system operator. The one-line diagram representations had to be easy to use, easily recognizable, coherent with regard to voltage levels, and easily adaptive to hardware and software requirements.

Considerations

Since the system is to be educationally used by students with little or no experience in interactive graphics control, the system has to be as simple to use as possible. The switches and circuit breakers are designed to change to the opposite state when pointed at. For raising or lowering voltage or frequency, arrows pointing up or down leave no doubt as to their purpose. Additionally, the

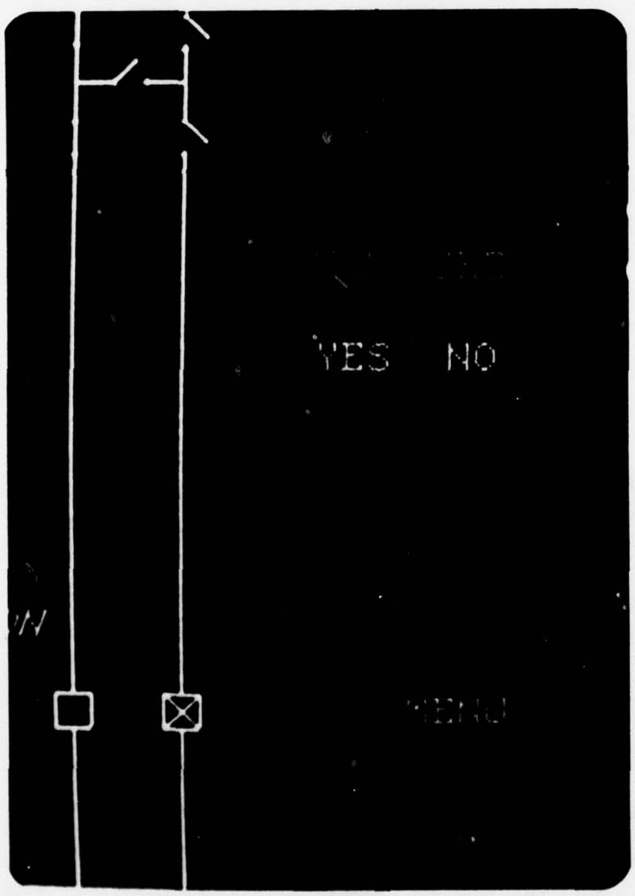


Fig. 4.2. Close-up of CB's, switches, "YOU SURE?", and "MENU" for displays.

magnitude being changed is displayed directly to the left of the arrows. The result is a simple, easily read and understood group of displays forming the system. (see Fig. 4.1, and 4.2)

Users will be familiar with the simulator before using the system. It is important that the layout of the one-line diagrams closely resemble the simulator, so the user wastes little time learning the system.

Ideally, transmission, distribution, and secondary voltage levels should be represented together. This would allow switching at equal voltage levels on the same picture. This is desirable from the standpoint of sequential switching and evaluation of system status. For identical reasons, the substations are each represented separately. The network system and, loads large enough to deserve the attention, each have their own display, too. See Chapter Six for photographs of each display.

Since the complete design was original, it offered excellent opportunity to avoid man-machine interface problems. By considering right-handed users and the right side light pen connection, the data displays and variable controls were oriented for ease of viewing. The different text types for titling and commands avoids confusion. The symbols used for circuit breakers and switches leave no doubt as to their being open or closed. A special attempt was made to place data and commands to leave little question of their meanings.

Every attempt was made to make the commands as direct as possible by avoiding sub-commands for actions. Instead of pointing to a voltage regulator, then to a raise or lower command in the margin, then to a surety command, this system has arrows as described earlier. After pointing to the arrow, a safety command is included, too. The sequence, avoiding sub-commands, is more direct and saves time. The fewer light pen hits needed to perform an action; the quicker the action is performed. While this method of presentation is more wasteful in presentation, ease of use and time saved more than compensates.

As mentioned above, a safety "switch" is included to avoid accidental light pen hits. The switch enables the operator to make sure the command directed was the intended one. A stop in the main control program causes the words "YOU SURE?", with "YES" and "NO" answers beneath the question, to appear after a hit on a circuit action. At this point nothing else will happen on the screen until the question is answered by a hit on the desired answer. Thus, mistakenly dropping generation or load is avoided.

Design

As described, the display on the screen has to be built by directing the electron beam with the graphics software subroutines. The problem here is to have the beam at the correct coordinate position for the line or figure to be drawn. This was solved by drawing the desired display to scale on graph paper with small grid markings. The coordinates were then easily picked off the drawing. Another by-product of the scale drawings was the ability to predetermine the layout and placement. Changes to effect better display could be made much easier on the drawing than after programming the display on the computer.

After the drawings were completed, critiqued, and corrected, subroutines to draw any repetitive figures such as transformers, generators, and circuit breakers were written. Part of the process involved further scaling of the figures to determine the most asthetic presentation. The scaled drawings were very necessary for placement but in some instances the scale would look different on the screen. To avoid such scaling modifications the figure subroutines were made as general as possible. For instance an argument to the circle subroutine is the desired radius and several of the subroutines have arguments for light pen sensitivity, intensity, and flash.

Optimizing

At this point, the programming involved writing, in order, the graphics subroutine calls to duplicate the drawings on the screen. The order becomes important for optimal memory use. The task itself can be made shorter by proper ordering of the calls to avoid moving the beam needlessly and repeating certain calls. The display buffer can be made smaller as a result of proper ordering. Several ideas and facts are instrumental in optimizing.

The subpicture concept keeps the size of the buffer much lower by utilizing the same area of the buffer many times. The SUBP(n,m) routine does use several words; so, there is a trade-off when copying subpictures of few statements. While the copying of subpictures keeps the buffer small, the beam still has to trace out more patterns; so, the refresh rate is still longer. For instance, a buffer size of 900 words with many subpicture copies may cause more flicker than one of 1100 words with few subpicture copies.

One of the most efficient means of saving buffer and decreasing flicker is to limit the use of certain calls. Since the APNT and LVECT routines use two words of buffer to point and draw long vectors, a minimal use of these routines helps. By using RDOT, except when a APNT is needed after an ON-OFF subpicture, the buffer size for this type call is halved. The almost total use of the VECT routine causes some gaps to appear at the end of long lines but these are not distracting. After viewing the number of lines on a display, it is easy to see a sizable buffer savings is made by using the VECT routine.

Similarly, the TEXT subroutine displays two symbols per digital word. A line of text with an odd number of characters and spaces wastes a half word.

Shortening the text by one character will free a whole word in this situation. Also, once the graphics is in the text mode, it stays there for successive characters on a line. By presenting all text horizontally, a savings is made. As an example, if "RING BUS" is to be displayed, write both words on the same line and not separate lines. Note the total characters and spaces is eight, an even number. Another savings concerning text is made by displaying all italics at once. This is to avoid using the STAT(n) call many times if the order of display is careless.

Another way to save buffer words is to group like calls together to avoid a system, change mode word. This is particularly applicable to the line drawing routines, VECT and LVECT. For displays with many lines the goal is to connect as many end points as possible. Of course, the first consideration of this fact is in the initial scaled drawings. An additional savings is realized by not having to move the beam invisibly, via an APNT or RDOT call, to another start point prior to the next line. So as many as three words may be saved by proper ordering and layout.

To demonstrate the effects of not optimally ordering, the network system display was built without ordering. All the horizontal lines were displayed first which meant a RDOT call between each VECT call. The vertical lines were then drawn. This display has one of the largest buffers and not nearly the degree of presentation of the other large displays. APNT and LVECT calls (two words in length) were used originallly, also, but had to be replaced with RDOT and VECT calls to reduce the buffer to its present size.

A savings will be made by predetermining what the most likely start and stop point for the general figures will be. The subroutine can start and stop there on the screen avoiding the need for an invisible beam movement prior to calling the figure. The flow of ordering through a display should be dictated by these start and stop points for the copied subpictures. As an example, if all transformers start at the top center and finish at the bottom center, to approach the transformer with a line from the bottom would require an additional invisible beam movement to top center to continue the display. Two extra calls are required to draw from bottom to top, rather than top to bottom. The direction of the flow is important, too.

One consideration while organizing the displays applies to optimizing the main control program later. The displays are saved on disk to be called as needed by the main program. If a display isn't complete and must be added to by the main program, considerable task is used that could have been performed by the display building task. Modifications to the display necessary in the main program are affected by turning on and off subpictures already completed by the display building tasks. The example here is the "YOU SURE?" switch. Each display building task includes this subroutine and the main program just turns it on when needed.

Overall, a conscious effort to avoid moving the beam invisibly by planning ahead for layout and text position will pay off in a large display. Ordering is important.

Subpicture Numbering

As was mentioned earlier, each subpicture must receive a number and it must be different from all other subpicture numbers in the display buffer. To avoid errors it is important to keep track of each assigned subpicture number. As an aid, each separate display has a different hundreds digit prefixing its numbers. A list for each display is found in the appendix.

Another numbering system exists for all subpictures involved with interaction. This was required because some circuit breakers appear in several displays. The system had to have an efficient method of updating that circuit breaker on each display. At the same time the circuitry handling the change command had to assign a channel number to each circuit action it performed. To facilitate the handling of both channels and subpicture numbers, they were made equal and are numbered from one to one hundred See SUBNUM.TXT in the appendix.

The analog to digital converter channels were each assigned a data collecting point on the simulator. These assignments become important when displaying the data on the screen because a subpicture number is associated with each data placement. The appendix contains these listings.

Saving Displays

The last step in completing a display is to save the display buffer on a disk file. The SAVE routine will accomplish this task to a file named in the argument. When needed it is restored to the main program buffer via a RSTR call. All these files are designated "XXXXXX.DPY" in this system.

Trouble Shooting

Once the program is written, it is ready to be typed in for input. The only input terminal available for the computer is teletype to disk files. This step isn't as easy as it sounds. A thorough knowledge of the computer's RSX-11M Utilities is necessary. A description of the utilities isn't pertinent to this discussion and would prove quite lengthy if included. After the Fortran source file is completed and edited, the source is compiled and the task or link step is completed. The task is then ran to check the results. See the appendix.

The first run is usually quite humorous. Due to mistakes in coordinates and typos while editing, the display seldom resembles the expected. Since much of the display is relatively plotted, mistakes magnify themselves throughout the display. This makes the exact location of the error difficult to discover. A solution to this problem lies in completing the display one-quarter at a time and, before running, having a fresh copy of the source to trace through.

This difficulty is aided by the small compiler and lack of error detection particularly in the arguments of the graphics subroutines. The errors probably won't be detected until the "run" stage. Errors listed at either compile or task building time may not contain sufficient error detail to explain an error. Try to build the task anyway. If it builds, go ahead and run it even though error messages are present. The actual error will then usually be more apparent. The error detection and messages aren't easy to work with. An appreciation of this fact will save some hair.

Another difficulty when dealing with errors is finding the codes. The fortran codes are in the fortran book, task builder errors in that book, and general operator errors in the operator's manual. The utilities errors are under the appropriate chapter in the utilities manual.

One important error is the "FATAL ERROR T" error while compiling. Its code is found somewhere in the middle of the fortran book. It means there isn't enough memory for compiling. As a solution, break the program in half and call the second half as a subroutine from the first. This is but one example of the problems encountered when trouble shooting errors.

Conclusion

The building of the simulator displays involves more than just putting lines and figures on the screen. Consideration must be given to man-machine interface, optimal ordering of statements and drawings, judicious use of certain calls, and efficient numbering of subpictures. The finished product is a simple, meaningful, flicker free display.

The source program for each display is attached in the appendix with detailed comment cards to explain the beam position throughout.

CHAPTER FIVE

CONTROL PROGRAM

The main control program is the master program for the interface between the interactive graphics terminal and the simulator. It coordinates commands issued from the graphics terminal with simulator commands. It keeps the displays updated with current switch positions and data. This program must be on-line to the control the simulator.

This chapter explains the main program in a chronological manner, as the CPU would see the program. Each of the eleven displays are handled basically the same. A diagram in Fig 5.1 illustrates the flow for the display handling sections. The variables for the displays are the subpicture numbers that were optimally ordered to minimize these variables. The optimal ordering also allows the standardized handling of each display to minimize the main program.

Initialization

The first step must be to initiate the graphics display by defining the common block for the display buffer and calling INIT to initialize the buffer. The buffer is initialized each time a new display is desired. This accomplishes turning the old display off, too. The INIT call is followed by the RSTR call that brings the new display into the buffer from disk. See CONTROL.FTN source in the appendix.

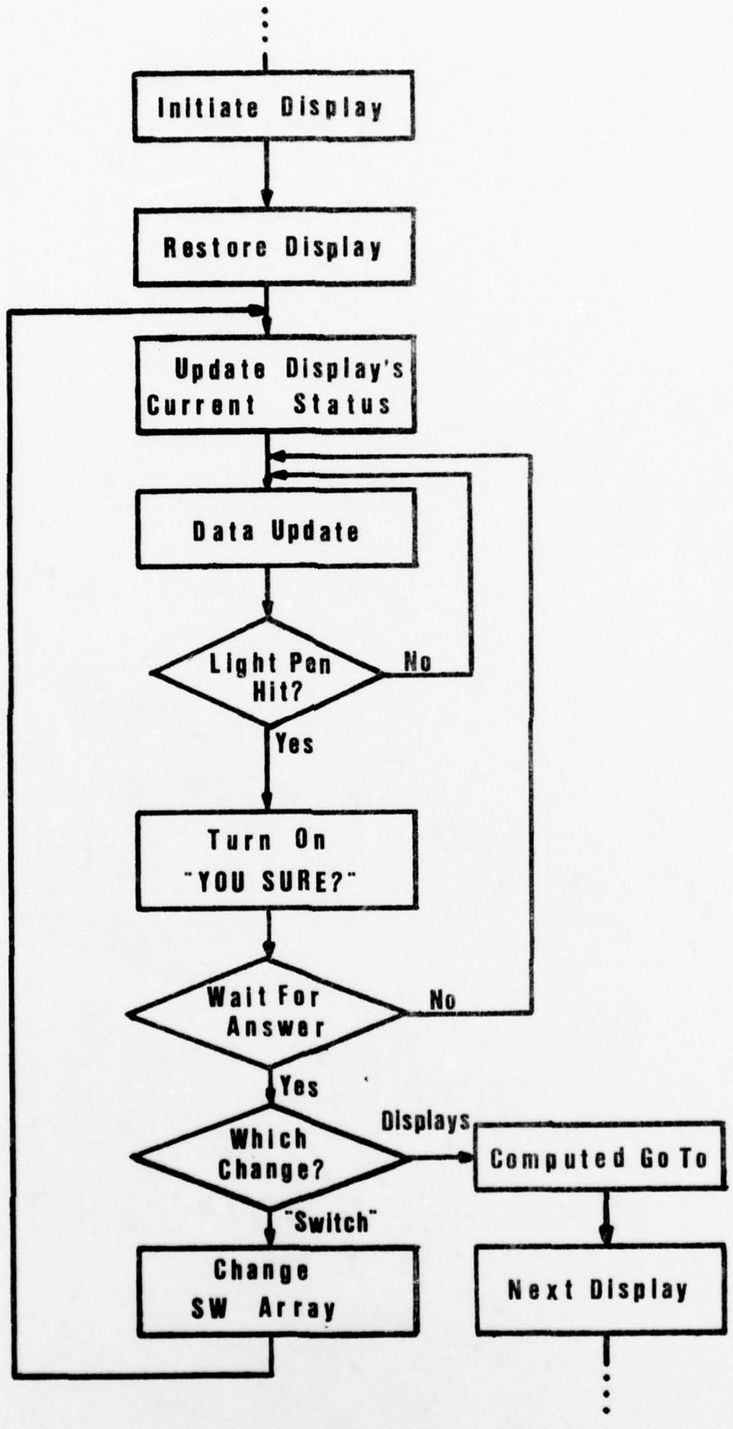


Fig. 5.1. Flow Chart for Main Control Program.

Update

Since each display is stored with fixed initial switch and circuit breaker positions, a display, newly loaded into the buffer, must be updated with the systems latest status. This applies to transmission line switches and circuit breakers as well as generation control switches. (In the following paragraphs a "switch" will refer to a binary operation of which the above three types are all included, unless otherwise noted.) This subroutine is the return point after a switch change command; it accomplishes the display change.

A subroutine called UPDATE accomplishes the update by passing arguments that are optimally ordered to enable the update of only those switches on the current display. The optimality achieved results in a few of the many switches being updated at once, so that the reaction time from new display to updated display is almost instantaneous.

The subroutine uses an array of length 100 to store the information necessary to indicate switch positions. A switch is a binary operation, so the array is a Logical array where each element is one byte long and contains only .TRUE. or .FALSE. values. This saves memory. The subroutine checks the value of the array elements corresponding to subpicture numbers for switches in the current display. (Switch subpicture numbers are identical to array subscripts to save arithmetic and time.) If the value of the array element is .FALSE., it turns the subpicture on that exists in the original display and turns its companion subpicture off. The array is initialized all .FALSE. to begin so that all circuit breakers start open and most functional switches, closed. It is the switch's initial position subpicture that is numbered less than one hundred of which only 76 are now in use.

The companion subpicture is the opposite switch position subpicture. To make identification simple, they are numbered the same as the original, plus one hundred. So, if UPDATE checks an array element and discovers it is .TRUE., it turns off subpicture N and on subpicture N+100. If the N+100 subpicture receives a light pen hit, a check must be made to see if the subpicture number is greater than 100 and if it is, subtract 100 so the update and change will occur properly.

One subtlety remains. The circuit breakers use only one light pen sensitive main subpicture and a non-sensitive companion. The companion is just the "X" to indicate it is open. Its absence indicates closed. This means only turning on or off one subpicture every change of circuit breaker position. All circuit breakers are numbered consecutively so different sections of UPDATE handle circuit breakers and switches. The reason for this was to avoid duplication the graphics statements required to draw two semi-identical subpictures in the display program. The result is a smaller buffer size (less flicker) but slightly lengthier update subroutine.

The Loop

After updating the new display, the loop for monitoring data and waiting for light pen hits is entered. The first step is to insure the system's event flag for a light pen hit is clear. The system subroutine CLREF accomplishes this. The event flag isn't cleared automatically after each light pen hit so it must be cleared at an appropriate time by the subroutine. The clearing at the beginning saves clearing at each different end of the loop. Next to be presented is the data.

The simulator's data to presented on the display must be obtained from a channel on the analog to digital converters. A subroutine called SINCON accomplishes this. Each display's data channels are a consequtive sequence. The end points of the sequence provide the arguments for a DO loop to check the channel value and then put it on the screen. The necessary alphanumberic fields were given position on the displays in the display building programs. Now, only their subpicture number is referenced by the NMBR routine. The subpicture numbers are also consecutively numbered and equal to the channel number plus a constant. This enables the same DO loop to be used for all main displays.

To detect a light pen hit, the LPEN subroutine must be called. If no light pen hit occurs when LPEN is called, the event flag stays set to zero. At this point, a check for the status of the event flag determines whether to continue with the light pen command or go back to the top of the loop. It is easy to see that data changes and light pen hits may not be instantaneous, depending on which part of the loop the CPU is handling at the time.

If there has been a light pen hit during the call to LPEN, the loop is extended, the "YOU SURE?" subpicture with answers is turned on, the light pen event flag cleared, and another sub-loop is entered to wait for the answer of the question. The sub-loop is on a LPEN call and if the answer is NO, the control returns to the top of the loop.

If YES, the loop is extended and the "YOU SURE?" subpicture is turned off. At this point the subpicture number, N, where the original hit occurred, is tested for its value. If less than 200, the command is for switch change. The next few statements determine whether the N is greater than 100, and if it is, 100 is subtracted from it. The array element corresponding to N, SW(N), is checked for .TRUE. or .FALSE. and the array element is changed to the opposite value. The return of this section of the loop is to the UPDATE call where the display is changed. This completes the loop for a switch change hit and the cycle is repeated.

If the value of N is greater than 200, some change to the display or a change of display itself has been directed. The loop is then exited for a section dealing with peculiarities for the current display.

The group of statements forming the loop is con ined in a subroutine called CHECK. The arguments of CHECK are the end points of two sets of subpicture number sequences for use in two UPDATE calls. Another pair of end points describe sequences for data channels. The last argument is passed back to the main program and is the subpicture number where a light pen hit occurred.

The CHECK subroutine is used for all displays execpt the introduction and high voltage transmission displays. The introduction control loop is more simple so it doesn't need the detail of CHECK. The high voltage transmission display has the date and time displayed which causes its loop to be different. The loops for these displays function the same as CHECK, however.

Display Peculiarities

The section dealing with peculiarities of a display currently contains statements to discern which display is desired next. Every display has a different list of displays available from that display. The switch is text on the screen in standard capitals that has a subpicture number greater than 200 and is light pen sensitive. Any text not a switch is in italics. A light pen hit on the text causes the loop to be exited and the subpicture number to be passed to this section.

Each display's subpicture numbers are prefixed with a different hundreds digit. The exception is the circuit breakers and functional switches. One of the reasons for this is to be able to easily change pictures or perform some action peculiar to a single display. If the subpicture numbers for a display are consecutively numbered from some hundred plus one, and the hundred is subtracted, the new subpicture numbers are consecutive from one. This is done to provide the variable for a computed GO TO statement.

The arguments of the computed GO TO statement have special significance. The INIT statement that begins each display's section in the main program has the statement number identical to the display's subpicture number, hundreds digit. For example, the current display has a switch SUBSTATION-4 with subpicture number 1105 and the substation-4 display's hundreds digit is 3. A hit on SUBSTATION-4 would cause 1105 to be passed out of the loop. In the next step, variable NB would be calculated as equal to 1105-1100=5. Thus, the fifth argument in the computed GO TO statement would be 300 and the next step executed would be statement 300, the substation-4 display initiator. It is in this manner that all display changes are made.

Conclusion

The complete cycle of control in the main program has been discussed. From display initiation, to changes on that display, to new displays, the main program uses optimal ordering of subpicture numbers to efficiently change the displays. The loop provides the opportunity for data update and reception of light pen commands, continuously. The last accomplished function is to turn off the program. The program is exited by a light pen hit on the word EXIT found on the introduction display.

CHAPTER SIX

PRESENTATION OF RESEARCH PRODUCT

Photographs of the one-line diagram displays, as they appear on the graphics screen, are presented in this chapter to illustrate the results of the programs. The control of the diagrams is dynamic and impossible to demonstrate in this medium. The results of the control program may be followed on the photographs, as the system was set up to provide power to both substations from generator one and two and the interconnection when the photos were taken. Some of the finer details of each display is included for better understanding of the displays.

There are eleven displays. Two of the displays contain information that may be obscured by varying the intensity control; so these have two photographs to show both intensity levels.

Introduction Display

The introduction display (Fig. 6.1) contains a paragraph of text to explain the system operation. In an educational situation, similar displays may be developed to contain instructions for experiments. To the right of the display is a list or menu of available displays and the exit switch. Each word of the menu is light pen sensitive and is referred to as a "switch." This display is presented initially, automatically, when the control program is initiated. This display is selected by the "MENU" switch in the other displays. Pages 102 through 104 in the appendix contains listings of the files necessary to create this display.

High Voltage Transmission Display

This display (Fig. 6.2) contains all the high voltage lines and buses. It is a general layout of the entire system. It would be referenced to get the overall view of generation and supply for the system. Larger scale displays of the components are obtained by selecting the display's, word switch with the light pen. Due to its size, no data is presented on this display. Defining routing is its primary function. Pages 52 through 59 of the appendix contains the file listings for this display.

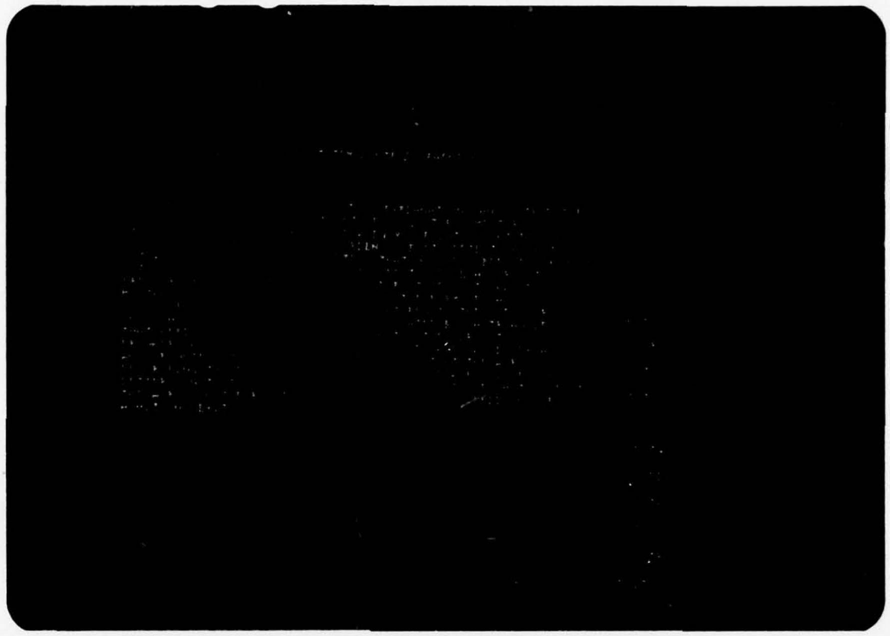


Fig. 6.1. The Introduction Display with Display Menu.

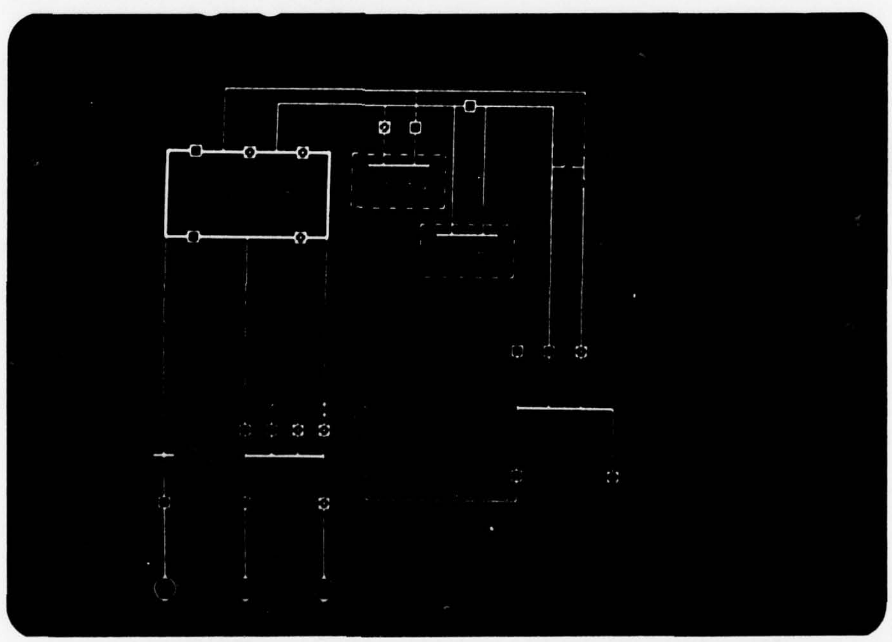


Fig. 6.2. The High Voltage Transmission Display.

Substation-4 Display

This display (Fig. 6.3) gives a layout of the substation and its connected load. The dot-dashed lines indicate breaks between this and other displays. So the display picks up with the high side of the transformer. In the photograph the network system and the residential loads are being fed from this station. The appendix contains the files for this display beginning on page 60.

Industrial Load Display

The display (Fig. 6.4) gives the large scale layout of the connected industrial load. The layout starts at the substation-4 bus for continuity of presentation. The arrows and data near the regulator are for control of the regulator from the console and monitoring of the load. The switches on the individual loads are not just manual switches, so that the load may be shed from the terminal. This is not the case for the capacitive load, however. The appendix contains the files for this display beginning with page 80.

Substation-6 Display

Fig. 6.5 and 6.6 show the display with intensity down and up, respectively. Note the manual switch, when closed, will give warning of itself on the low intensity display by being brighter than when it's open. This substation is feeding both loads in the configuration in the photo. The numbers on the incoming lines are their identification. The appendix contains the files for this display on pages 65 through 69.

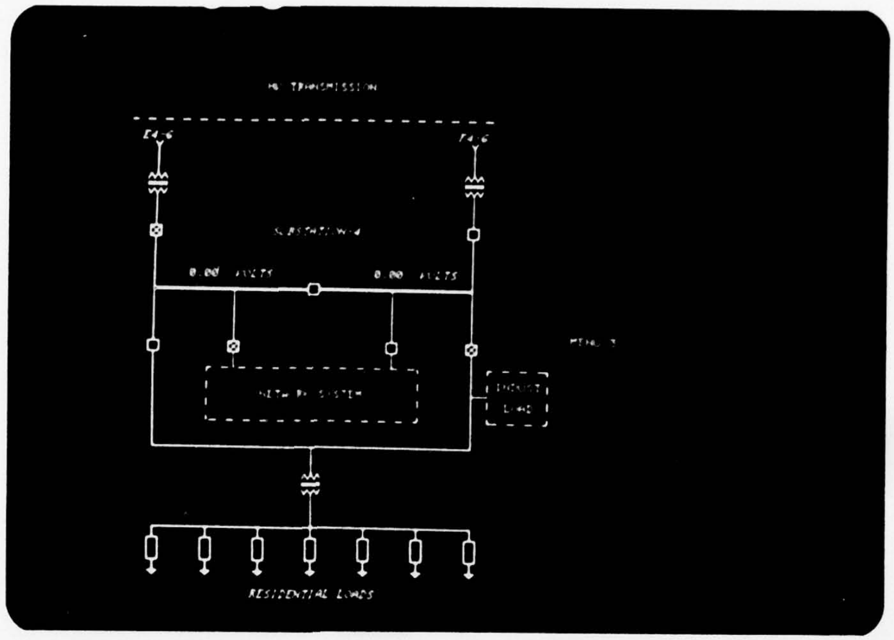


Fig. 6.3. The Substation-4 Display.

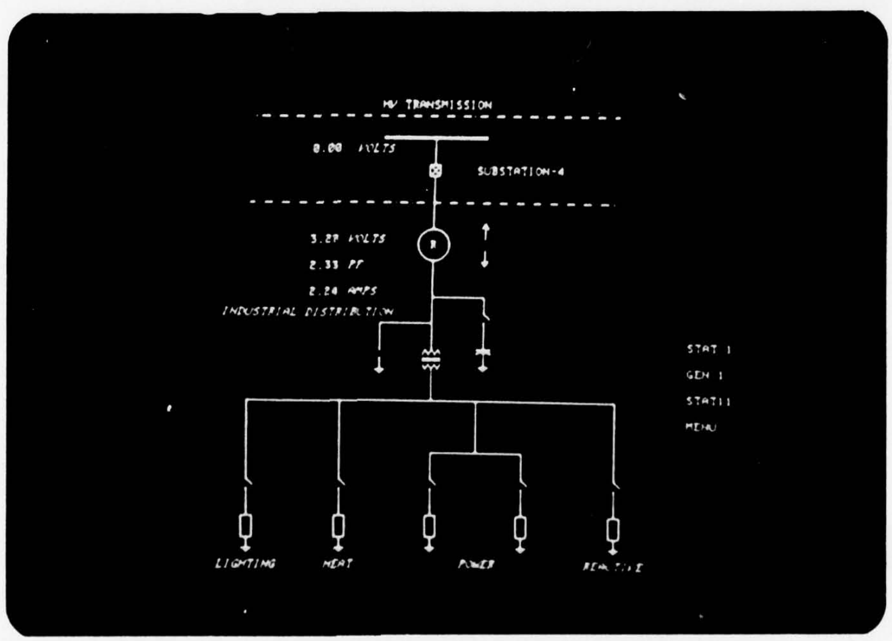


Fig. 6.4. The Industrial Load Display.

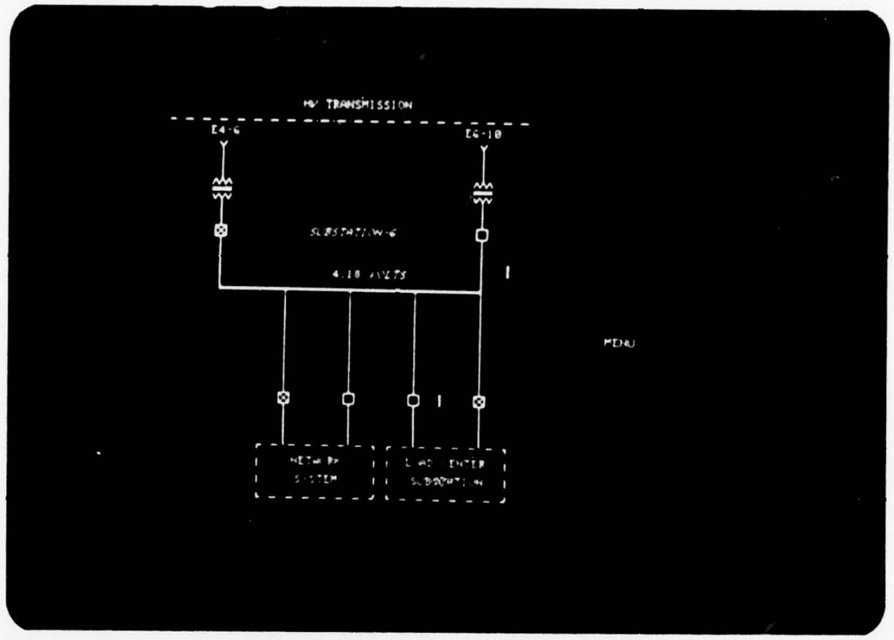


Fig. 6.5. Graphics Controlled Substation-6 Display.

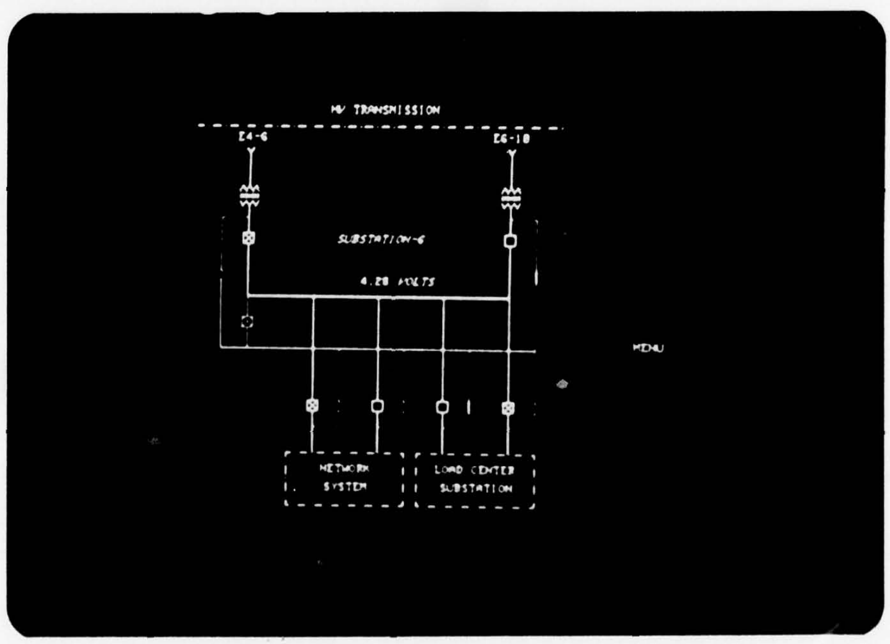


Fig. 6.6. Manual and Controlled Substation-6 Display. Intensity up.

Network System Display

The network system (Fig. 6.7) is different from the other displays because it is fed from two different substations. Both buses and their voltages are included for continuity of presentation. Each phase's voltage and current is displayed for the secondarybeing fed from both substations on one feeder each. The appendix contains information on this display beginning on page 75.

Load Center Display

This display (Fig. 6.8) depicts a secondary selective system. Again the substation-6 bus and bus voltage is included for clearity. Note that the configuration indicated in the photo would result in no power supplied to any load unless the right feeder breaker is closed. The position of the breakers in this system would allow selective load management at the secondary level. Data is presented to enable monitoring of load. The appendix contains the files for this display on pages 70 through 74.

Station 11 Display

These photographs (Figs. 6.9, and 6.10) are large scale depictions of the controlable and manual features of this switching station. The display incorporates the features of the displays discussed earlier, particularly the substation-4 display. The peak load generator is presently a dummy generator but was included for future use. Pages 85 through 89 of the appendix contain the files for this display.

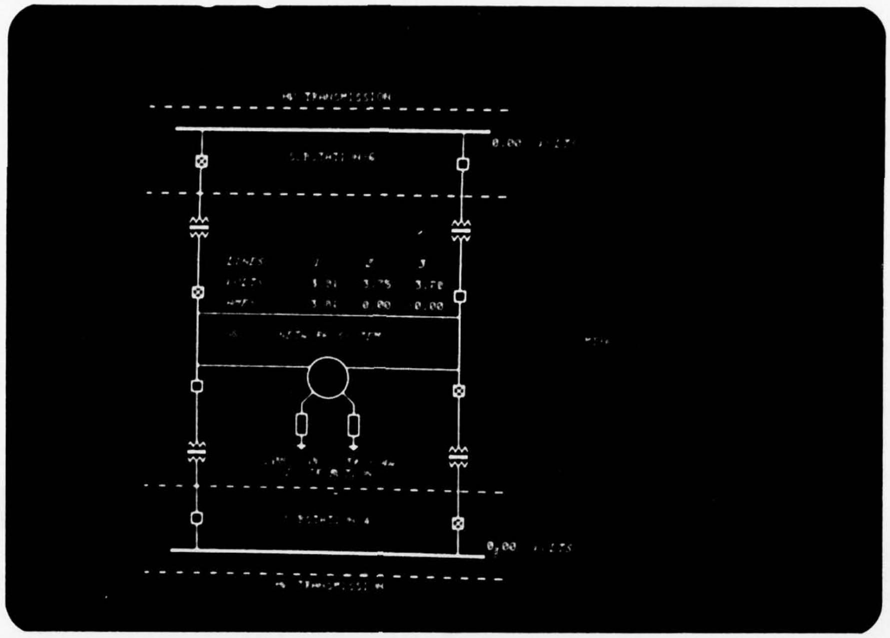


Fig. 6.7. The Network System Display.

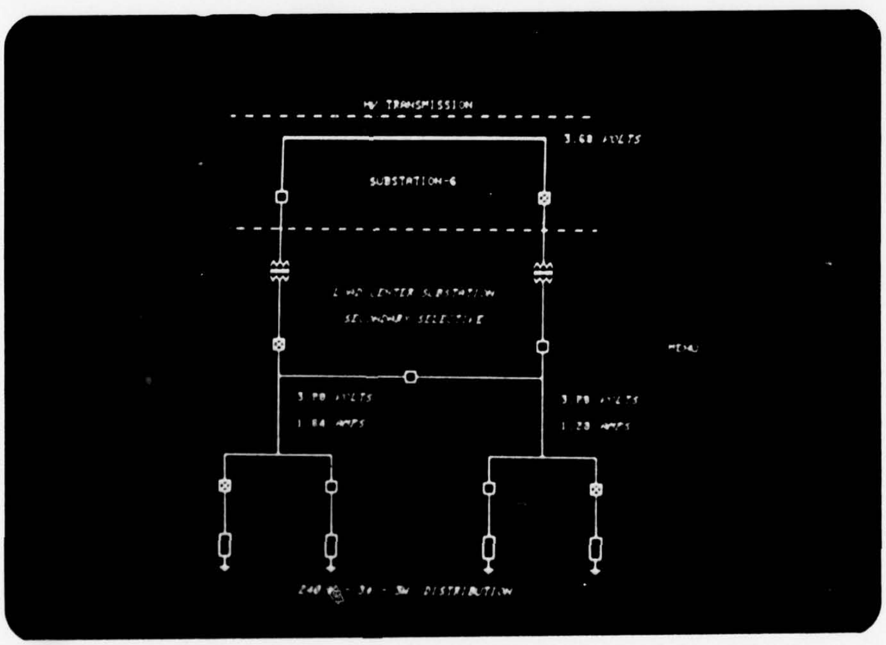


Fig. 6.8. The Load Center Display.

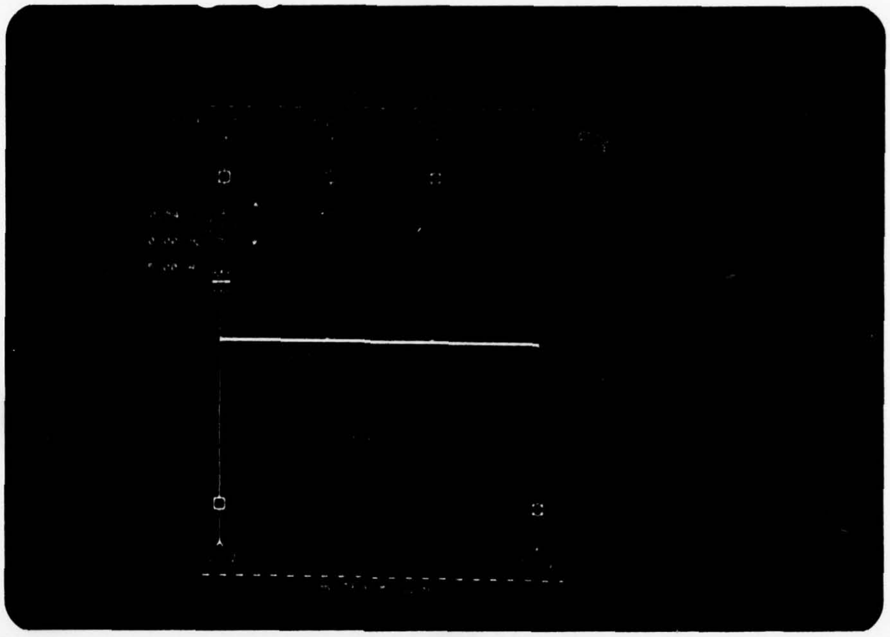


Fig. 6.9. Graphics Controlled Station 11.

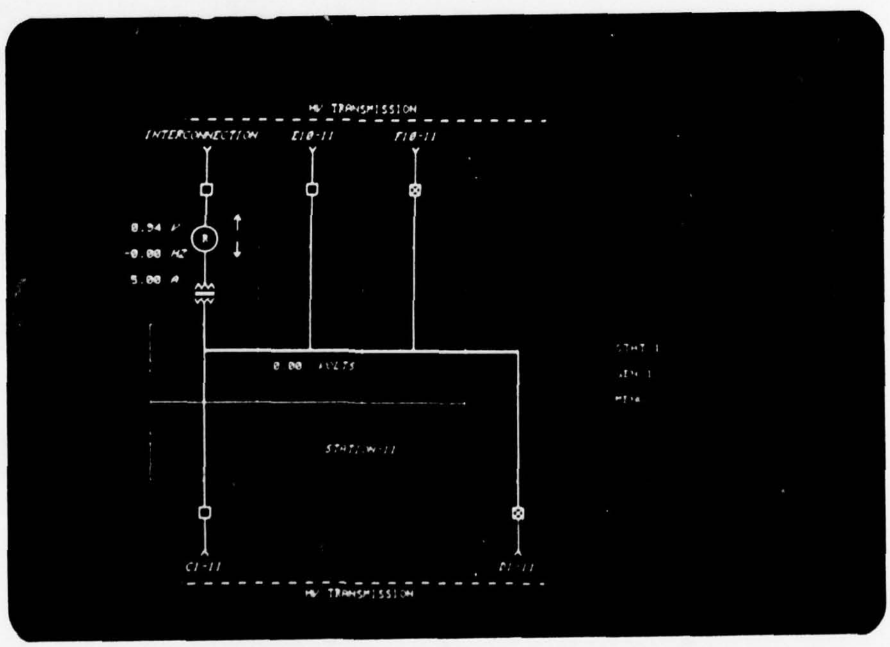


Fig. 6.10. Manual and Controlled Station 11. Intensity up.

Station 1 and Generator 1 Displays

These two displays (Figs. 6.11, and 6.12) contain large scale depictions of the generation control stations. The control switches in each case are identical. The prime mover is the switch for the DC motor and is activated by a hit on the word "ON" or "OFF" , at which time the opposite position will be displayed. The word displayed indicates its current condition. The excitation switch operates the same. The arrows for voltage control raises or lowers the excitation and the arrows for frequency varies the power input to the DC motor or prime mover, simulating increasing the steam to a steam turbine. The generator-1 depiction also includes all the data for the other suppliers making generator one an ideal swing bus generator. The indications on the data presentations are all identical because the A-D channels are all tied to a power supply until the hardware to the simulator is finished. The appendix on pages 90 through 94 and 105 through 110 contains the files for this display. Fig. 6.14. shows a user closing the circuit breaker to the Station Service load on the Station 1 display.

Synchronizing Display

This display (Fig 6.13) contains the controls for all the suppliers and a route for synchronizing the generators at any of several points. Space has been allowed below the word synchronizing for the inclusion of a remote controlled synchronizing display. One method of accomplishing a syncroscope function would be to use the three lights system and simulate the lights with small circles on the screen. Another method would actually display the moving voltage vectors, superimposed. Later a switch for automatic synchronizing could be added, too. Files for this display are in the appendix, pages 95 through 101.

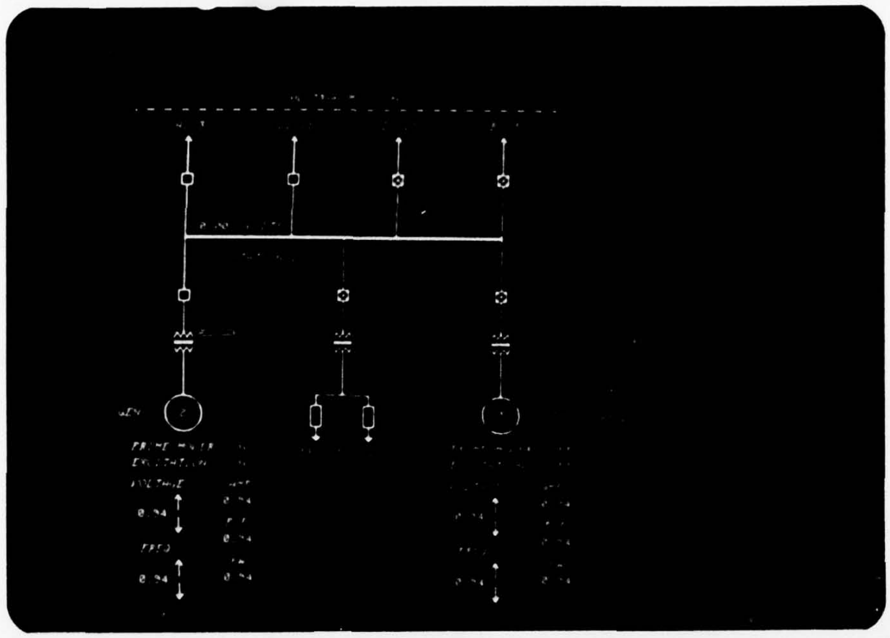


Fig. 6.11. Station 1 Display.

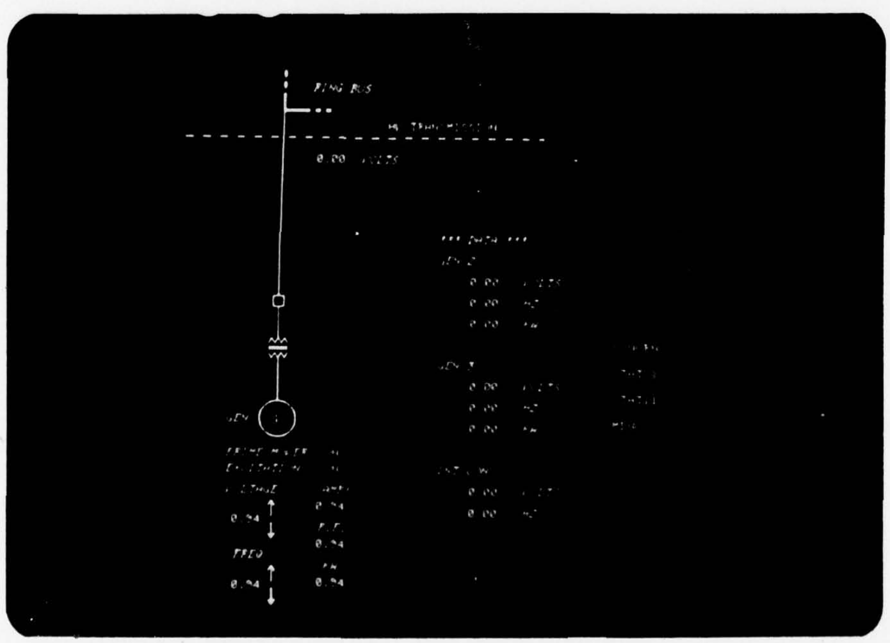


Fig. 6.12. Generator 1 Display.

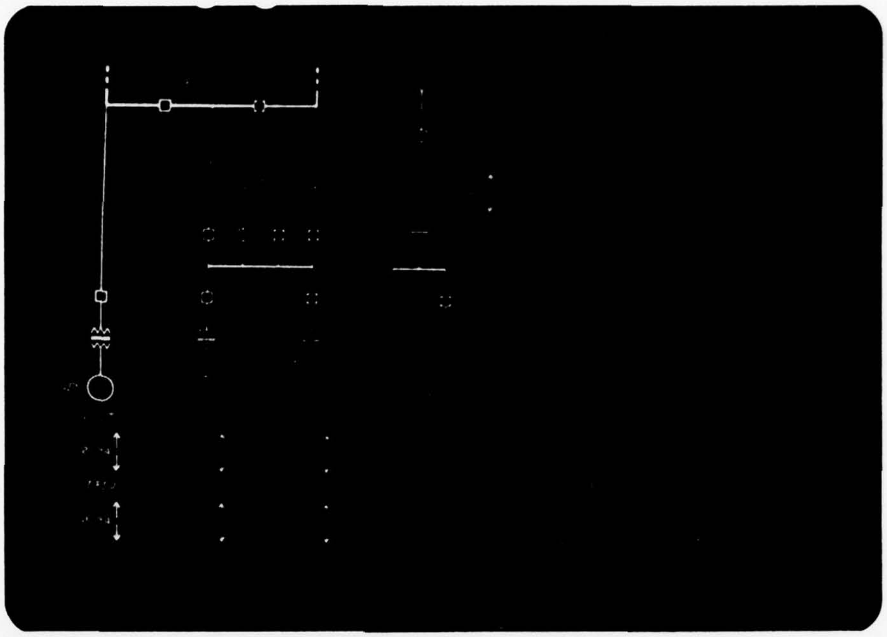


Fig. 6.13. Synchronizing Display.

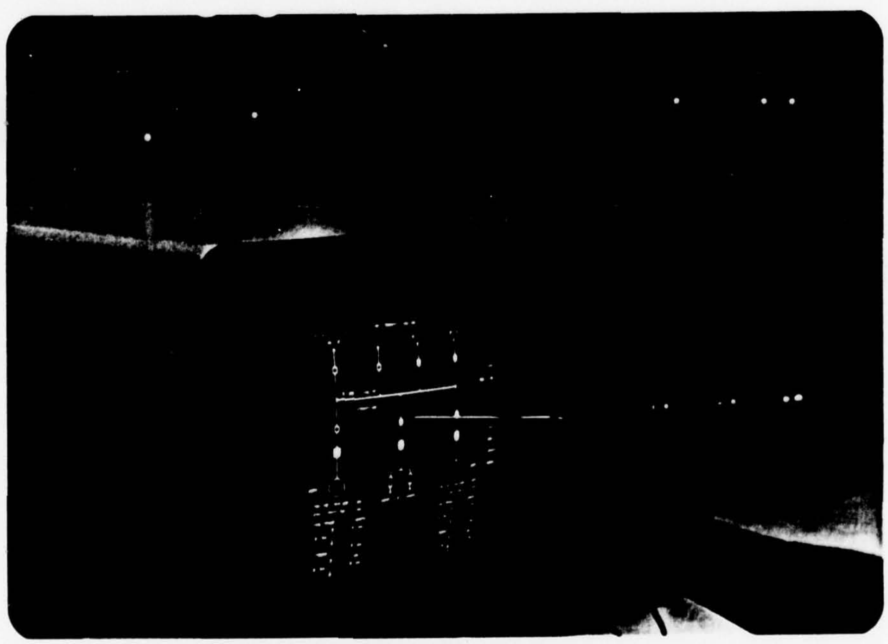


Fig. 6.14. Adding Load to Station 1. Display Control in Use.

CHAPTER SEVEN

CONCLUSIONS

The collection of displays described in this thesis make possible a powerful and unique research and educational facility.

The combination of the PDP11T34 and Power System Simulator into a computer controlled power system provides a very unique research tool for control of power systems. The possibilities and capabilities of the computer to control a complete power system are just beginning to be explored. While an actual power system does not allow experimental research because of cost and possible service loss if failure occurs, the system developed here does.

The ability of the computer to completely control a power system has yet to be completely explored. With this system it would be possible to write software to start from zero power production, bring generators up, synchronize them, add load, close breakers, provide the relaying and circuit breaking signals, open lines, issue manual operation commands, and shut down the system. The coordinated use of real time stability and load flow calculations, coupled with real time, short-run load forecasts would enable the computer to supply the operational necessities of a system automatically. Of course, the computer has to be programmed by very knowledgeable engineers so that the decisions made by the computer are sound ones. The computer really only provides the capability of not forgetting past mistakes and storing all of present knowledge if properly programmed. The ability to provide an operator with that capability should not be overlooked.

Thus, the foundation for further research has been laid in the form of a computer controlled simulator. Its ultimate worth belongs to the future and the foresight of those who use it.

The educational value of the computer controlled simulator is unquestionable. It is state of the art or better for many power companys. The system provides the power systems engineer with a current system with which to learn the operation of the power system. The student will be "at home" with any company after completion of his education.

The opportunities for the digital or circuit design engineering student to apply his knowledge to the hardware interface has been great and will continue as more and more research is advanced. While the power systems student learns with "hands on" the equipment, all students learn about digital control systems.

After some consideration of the capabilities of the graphics terminal and the computer, it has come to mind that on even more powerful tool may exist here than this paper describes. By using the computer or groups of micro-compters to run real time simulations of a real power system and combining the outputs of these simulations to the graphics terminal, a real power system could be simulated and controlled from the graphics terminal. This would void the need of the power system simulator itself and the resulting interface. It would allow the student to control a simulated, real power system.

It could be used effectively for research on new software to perform the control functions for the power system. Additionally the system planning divisions at power utilities level would be able to study contingencies presently impossible to preview. The human element could be included since the operator would be ultimately controlling during contingencies, especially if the system was not fully automated. The contribution to controller training would be invaluable.

The recent advances in computers and computer graphics would enable the design of such a system to become a reality. The possibilities for such a system simulator would be great. Hopefully, the start as outlined in this thesis will not be the finish.

APPENDIX

The appendix is the users manual for the software comprising the digital control system. The complete users manual exists as files on magnetic disk and thus should be updated as progress and improvements are made. The documentation is contained in TXT files, source in FTN files, and task builder commands in CMD files.

Each program, eleven graphics display and one control, has a text file (e.g., HVTRNS.TXT) where information necessary to use that program is contained. These files preceed each applicable program section in this manual. Two additional text files document common subpicture numbers (SUBNUM.TXT) and list A-D channel assignments (ADCHNL.TXT) and are listed at the end.

Following each text file are the Fortran source file(s) (e.g., HVTRNS.FTN) necessary for that program. The file containing the subroutines for the display sources, DSPSUB.FTN, is listed after the last dispay source program. Likewise, the subroutine file, CNTLSB.FTN, for the control program follows the control source listing. The command files are listed in each program section, too.

A list of the existing file directories immediately follows this introduction. Note objective (.OBJ) and task (.TSK) files have been deleted where no longer needed. The minimal number of disk blocks needed is 400. More will be needed when the tasks are built. An explanation of the file name prefix accompanies each section. The order of sections in the list is the same as the order of the listings in the remainder of the appendix.

The High Voltage Transmission Files.

```
PIP>HVTRNS.*;*,HVREST.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:40

HVTRNS.TSK;1          80.    C  10-OCT-77 16:09
HVTRNS.DPY;14         5.        19-OCT-77 21:09
HVTRNS.TXT;7          5.        22-NOV-77 13:13
HVTRNS.FTN;4          14.       19-OCT-77 21:21
HVTRNS.CMD;2          1.        22-NOV-77 10:41
HVTRNS.DPY;13         5.        30-AUG-77 09:44
HVREST.FTN;2          8.        19-OCT-77 21:21

        TOTAL OF 118. BLOCKS IN 7. FILES
```

The Substation-4 Files.

```
PIP>SUBST4.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:36

SUBST4.CMD;2          1.        22-NOV-77 10:49
SUBST4.TXT;3          4.        22-NOV-77 12:09
SUBST4.FTN;1          11.       30-AUG-77 09:44
SUBST4.DPY;11         3.        30-AUG-77 09:50

        TOTAL OF 19. BLOCKS IN 4. FILES
```

The Substation-6 Files.

```
PIP>SUBST6.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:36

SUBST6.CMD;1          1.        22-NOV-77 10:50
SUBST6.FTN;1          13.       23-AUG-77 00:40
SUBST6.TXT;6          4.        22-NOV-77 12:40
SUBST6.DPY;14         3.        23-AUG-77 16:04

        TOTAL OF 21. BLOCKS IN 4. FILES
```

The Load Center Files.

```
PIP>LDCTR.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:37

LDCTR.FTN;1           11.       25-AUG-77 12:27
LDCTR.TXT;5           4.        22-NOV-77 12:41
LDCTR.DPY;11          3.        25-AUG-77 12:34
LDCTR.CMD;2           1.        22-NOV-77 10:52

        TOTAL OF 19. BLOCKS IN 4. FILES
```

The Network System Files.

```
PIP>NETSYS.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:37

NETSYS.DPY;5          3.          25-AUG-77 13:11
NETSYS.CMD;1          1.          22-NOV-77 10:53
NETSYS.FTN;1         12.          25-AUG-77 13:02
NETSYS.TXT;5          4.          22-NOV-77 12:42

        TOTAL OF 20. BLOCKS IN 4. FILES
```

The Industrial Load Files.

```
PIP>INDLD.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:37

INDLD.CMD;1           1.          22-NOV-77 10:56
INDLD.FTN;1          11.          25-AUG-77 16:16
INDLD.DPY;12          3.          25-AUG-77 16:22
INDLD.TXT;5           4.          22-NOV-77 12:42

        TOTAL OF 19. BLOCKS IN 4. FILES
```

The Station 11 Files.

```
PIP>STAT11.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:38

STAT11.TXT;4          4.          22-NOV-77 12:37
STAT11.DPY;10         4.          25-AUG-77 13:39
STAT11.CMD;1          1.          22-NOV-77 10:58
STAT11.FTN;1         12.          25-AUG-77 16:22

        TOTAL OF 21. BLOCKS IN 4. FILES
```

The Generator 1 Files.

```
PIP>GEN1.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:38

GEN1.TXT;3            4.          22-NOV-77 12:46
GEN1.DPY;11           4.          25-AUG-77 14:20
GEN1.FTN;1           10.          25-AUG-77 14:12
GEN1.CMD;1            1.          22-NOV-77 11:03

        TOTAL OF 19. BLOCKS IN 4. FILES
```

The Synchronizing Files.

```
PIP>SYNCRN.*;*,SYREST.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:39

SYNCRN.CMD;1          1.          22-NOV-77 11:04
SYNCRN.DPY;3          5.          25-AUG-77 15:06
SYNCRN.FTN;1          10.         22-AUG-77 20:07
SYNCRN.TXT;3          6.          22-NOV-77 12:49
SYREST.FTN;1          6.          25-AUG-77 14:59

        TOTAL OF 28. BLOCKS IN 5. FILES
```

The Introduction Files.

```
PIP>INTRO.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:40

INTRO.CMD;1           1.          22-NOV-77 11:08
INTRO.FTN;1           5.          22-AUG-77 08:25
INTRO.DPY;6           3.          22-AUG-77 20:35
INTRO.TXT;3           3.          22-NOV-77 12:53

        TOTAL OF 12. BLOCKS IN 4. FILES
```

The Station 1 Files.

```
PIP>STAT1.*;*,ST1FIN.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:41

STAT1.TXT;4           5.          22-NOV-77 12:46
STAT1.CMD;1           1.          22-NOV-77 11:00
STAT1.FTN;1           12.         25-AUG-77 16:33
STAT1.DPY;11          4.          29-AUG-77 14:42
ST1FIN.FTN;1          3.          25-AUG-77 13:55

        TOTAL OF 25. BLOCKS IN 5. FILES
```

The Display Subroutine Files.

```
PIP>DSPSUB.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:41

DSPSUB.FTN;1          12.         06-AUG-77 19:47
DSPSUB.OBJ;1          31.         10-OCT-77 15:58

        TOTAL OF 43. BLOCKS IN 2. FILES
```

The Control Program Files.

```
PIP>CONTROL.*;*,CNTLSB.*;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:43

CONTROL.FTN;1        11.           10-OCT-77 15:41
CONTROL.TSK;1        66.      C    19-OCT-77 21:29
CONTROL.CMD;1        1.            22-NOV-77 11:11
CONTROL.TXT;2        3.            22-NOV-77 11:33
CONTROL.FTN;4        11.           22-NOV-77 15:18
CNTLSB.FTN;2         5.            22-NOV-77 13:04
CNTLSB.OBJ;1         8.            10-OCT-77 15:50

          TOTAL OF 105. BLOCKS IN 7. FILES
```

The text files for A-D channels and Subpicture Numbers.

```
PIP>ADCHNL.TXT;*,SUBNUM.TXT;*/LI


DIRECTORY DK1:[200,200]
22-NOV-77 15:44

ADCHNL.TXT;3         5.            18-NOV-77 15:51
SUBNUM.TXT;2         6.            21-NOV-77 08:23

          TOTAL OF 11. BLOCKS IN 2. FILES
```

THE 'HVTRNS.FTN' AND 'HVREST.FTN' FILES CONTAIN THE FORTRAN SOURCE FOR THE HV TRANSMISSION DISPLAY. THEY USE SUBROUTINES CONTAINED IN A FILE, 'DSPSUB.FTN'. EDIT THE FILES USING THE EDIT UTILITY.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
          >FOR HVTRNS=HVTRNS
 THEN     >FOR HVREST=HVREST
 THEN     >FOR DSPSUB=DSPSUB
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
          >FOR HVTRNS=HVTRNS/LI:1
THEN      >FOR HVREST=HVREST/LI:1
THEN      >FOR DSPSUB=DSPSUB/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
          >TKB @HVTRNS
```

THIS CAUSES FILES, 'HVTRNS.OBJ', 'HVREST.OBJ', AND 'DSPSUB.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'HVTRNS.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'HVTRNS.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11 AND BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
          >RUN HVTRNS
```

IF THE ABOVE SEQUENCE IS FOLLOWED, A NEW FILE IS CREATED CALLED 'HVTRNS.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE HV TRANSMISSION DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBER	DESCRIPTION
201	GEN 1 TEXT FOR SW
202	STAT 1 TEXT FOR SW
203	MENU TEXT FOR SW
204	SUB ST 4 TEXT FOR SW
205	SUB ST 6 TEXT FOR SW
206	ST 11 TEXT FOR SW
.	.
.	.
.	.
239	TIME
240	TIME
250	GEN 1 CIRCLE
251	40 UNIT BUS @ GEN 1
252	GEN 2 CIRCLE
253	RING BUS 180 LONG
254	GEN 3 CIRCLE
255	VERTICAL BUS FOR RING BUS
256	SUB ST 4 BOX
257	SUB ST 6 BOX
261 THRU 270	RING BUS SEGMENTS
271	ST 11 BUS
276	'GEN' 1 TEXT
277	'GEN' 2 TEXT
278	'GEN' 3 TEXT

```
C
C       PROGRAM TO DRAW THE HIGH VOLTAGE TRANSMISSION LINE PICTURE.
C
        COMMON/DFILE/IBUF(1100)
        DIMENSION TIM(4)
        CALL INIT(1100)
C
        CALL RDOT(0.,50.,-1,-1)                    ! 0,50
C
C       START SUBPICTURE OF THE GENERATORS.
C
        CALL SUBP(250)                             !SUBP: GEN , TRANS
C
C       DRAW A CIRCLE OF RADIUS 20 FOR GENERATOR ONE.
C
        CALL CIRCLE(20.,-1,4,-1)
C
        CALL RDOT(20.,20.,0,4)                     ! 20,70
        CALL VECT(0.,130.)                         ! 20,200
        CALL ESUB                                  ! END OF GEN SUBP
C
C       DRAW THE TWO CIRCUIT BREAKERS FOR GENERATOR ONE.
C       ONE BREAKER IS OPEN AND ONE CLOSED.
C
        CALL RDOT(-10.,0.,0,-4)                    ! PB FOR CB
        CALL SUBP(6)                               !SUBP:CL CB GEN 1
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(106)                             !SUBP: OP CB GEN 1
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(20.,220.,-1,-4)
C
C       DRAW TRANSMISSION LINE C1-3 AND BUS FOR GENERATOR 1.
C
        CALL VECT(0.,79.)                          ! 20,299
        CALL RDOT(-20.,1.,,-4)
        CALL SUBP(251)                             !SUBP: BUS FOR GEN 1
        CALL HBUS(40.,5)
        CALL ESUB
        CALL RDOT(-20.,1.,-1,-4)                   ! 20,301
        CALL VECT(0.,418.)                         ! 20,719
C
C       DRAW RING BUS WITH CB'S STARTING AT THE LOWER RIGHT CORNER.
C
        CALL RDOT(-1.,0.,-1,-4)
        CALL SUBP(255)                             !SUBP: VBUS MASTER
        CALL VBUS(164.,5)                          ! 19,882
        CALL ESUB
        CALL RDOT(1.,-1.,,-4)
        CALL SUBP(261,251)                         !SUBP: UP LFT RNG BUS
        CALL RDOT(0.,-10.,,-4)                     ! 60,870
        CALL SUBP(1,6)                             !SUBP: CLSD CB RNG BUS 1
        CALL SUBP(101,106)                         !SUBP: OPEN CB RNG BUS 1
        CALL APNT(80.,880.,-1,-4)
        CALL SUBP(262,251)                         !SUBP: UP MID RNG BUS
        CALL SUBP(263,251)                         !SUBP: UP MID RNG BUS
        CALL RDOT(0.,-10.,,-4)                     ! 160,870
        CALL SUBP(2,1)                             !SUBP: CLSD CB RNG BUS 2
        CALL SUBP(102,101)                         !SUBP: OPEN CP RNG BUS 2
        CALL APNT(180.,880.,-1,-4)
        CALL SUBP(264,251)                         !SUBP: RNG BUS
        CALL SUBP(265,251)                         !SUBP: RNG BUS
```

```
      CALL ESUB
C
C     DRAW LINE A1-3 WITH CB'S.
C
      CALL APNT(160.,340.,-1,-4)
      CALL SUBP(9,1)                          !SUBP: CLSD CB, A1-3
      CALL SUBP(109,101)                      !SUBP: OPEN CB, A1-3
      CALL APNT(170.,360.,-1,-4)
      CALL VECT(0.,360.)                      ! 170,720
C
C     DRAW LINE B1-3 WITH CB'S.
C
      CALL RDOT(140.,-380.,-1,-4)             ! 310,340
      CALL SUBP(10,1)                         !SUBP: CLSD CB, B1-3
      CALL SUBP(110,101)                      !SUBP: OPEN CB, B1-3
      CALL APNT(320.,360.,-1,-4)
      CALL VECT(0.,359.)                      ! 320,720
C
C
C     DRAW LINES F3-4 AND F6-11 WITH SWITCHES AT SECTIONALIZING
C     STATION 10.
C
      CALL RDOT(-200.,160.,-1,-4)
      CALL VECT(0.,118.)                      ! 120,1000
      CALL VECT(690.,0.)                      ! 810,1000
      CALL VECT(0.,-100.)                     ! 810,900
C
C     DRAW THE OPEN AND CLSD SWITCHES AT STAT 10.
C
      CALL SUBP(160)                          !SUBP: OPEN SW F6-10
      CALL SWOPV(4)
      CALL ESUB
      CALL OFF(160)
      CALL SUBP(60)                           !SUBP: CLSD SW F6-10
      CALL SWCLV(4)
      CALL ESUB
C
C     CONTINUE WITH LINE AT SECTIONALIZING STATION 10.
C
      CALL APNT(810.,880.,-1,-4)
      CALL VECT(0.,-40.)                      ! 810,840
      CALL SUBP(161,160)                      !SUBP:OPEN SW LR
      CALL OFF(161)
      CALL SUBP(61,60)                        !SUBP: CLSD SW LR
      CALL APNT(810.,820.,-1,-4)              ! PB FOR LINE F10-11
      CALL VECT(0.,-300.)                     ! 810,520
      CALL RDOT(-10.,-20.,,-4)
      CALL SUBP(16,1)                         !SUBP: CLSD CB LN F10-11
      CALL SUBP(116,101)                      !SUBP: OPEN CB LN F10-11
      CALL APNT(810.,500.,-1,-4)              ! PB FOR LINE TO BUS
      CALL VECT(0.,-98.)                      ! 810,402
C
C     DRAW THE BUS AT STATION 11.
C     THE SECOND BUS AND DETAIL IS ON STATION 11 PICTURE.
C
      CALL RDOT(-120.,-2.,-1,-4)              ! 690,400
      CALL SUBP(271,253)                      !SUBP: BUS @ ST 11
      CALL VECT(0.,-120.)
C
C     DRAW LINE E10-11 AND SWITCHES AT STATION 10.
C
      CALL APNT(220.,881.,-1,-4)              ! PB FOR LN E3-4
      CALL VECT(0.,89.)                       ! 220,970
      CALL VECT(360.,0.)                      ! 580,920
      CALL RDOT(0.,-10.,,-4)
      CALL SUBP(18,1)                         !SUBP: CB CL LN E4-6
```

```
      CALL RDOT(0.,-10.,,-4)                       ! 260,870
      CALL SUBP(3,1)                               !SUBP: OPEN CB RNG BUS 3
      CALL SUBP(103,101)                           !SUBP: OPEN CB RNG BUS 3
      CALL APNT(280.,880.,-1,-4)
      CALL SUBP(266,251)                           !SUBP: UP RT RNG BUS
      CALL RDOT(-300.,-160.,,-4)                   ! 20,720
      CALL SUBP(268,251)                           !SUBP: LW LFT RNG BUS
      CALL RDOT(0.,-10.,,-4)
      CALL SUBP(4,1)                               !SUBP: CLSD CB RNG BUS 4
      CALL SUBP(104,101)                           !SUBP: OPEN CB RNG BUS 4
      CALL APNT(80.,720.,-1,-4)
      CALL SUBP(253)                               !SUBP: LW MID RNG BUS
      CALL HBUS(180.,5)
      CALL ESUB
      CALL RDOT(0.,-10.,,-4)                       ! 160,710
      CALL SUBP(5,1)                               !SUBP: CLSD CB RNG BUS 5
      CALL SUBP(105,101)                           !SUBP: OPEN CB RNG BUS 5
      CALL APNT(280.,720.,-1,-4)                   ! PB FOR BUS
      CALL SUBP(270,251)                           !SUBP: LW RT RNG BUS
      CALL RDOT(1.,-2.,,-4)
      CALL SUBP(267,255)                           !SUBP: RT RNG BUS
C
C     COPY GENERATOR SUBPICTURE AT THE DESIRED POSITION FOR
C     GENERATOR 2.
C
      CALL APNT (150.,50.,-1,-4)
      CALL SUBP(252,250)                           !SUBP: GEN 2 TO CB
      CALL RDOT(-10.,0.,0,-4)                      ! 140,200 PB FOR CB
      CALL SUBP(7,1)                               !SUBP: CLSD CB FOR GEN 7
      CALL SUBP(107,101)                           !SUBP: OPEN CB FOR GEN 7
      CALL APNT(170.,220.,-1,-4)
      CALL VECT(0.,120.)                           !CONNECTION TO CB A1-3
C
C     DRAW BUS AT STATION 1.
C
      CALL RDOT(0.,-40.,-1,-4)                     ! PB FOR BUS
      CALL HBUS(150.,5)
      CALL RDOT(-20.,-250.,-1,-4)                  ! 300,50
      CALL SUBP(254,250)                           !SUBP: GEN 3 TO CB
      CALL RDOT(-10.,0.,,-4)                       ! 340,200
      CALL SUBP(8,1)                               !SUBP: CLSD CB @ 8
      CALL SUBP(108,101)                           !SUBP: OPEN CB AT 8
      CALL APNT(320.,220.,-1,-4)
      CALL VECT(0.,120.)                           !CONNECTION TO CB B1-3
C
C     WRITE THE TEXT ON THE GENERATORS.
C
      CALL RDOT(-320.,-330.,,-5)
      CALL SUBP(276)                               !SUBP: GEN 1 TEXT
      CALL TEXT('GEN')
      CALL ESUB
      CALL APNT(15.,38.,1,-5)
      CALL SUBP(201)                               !SUBP: SW FOR GEN 1
      CALL TEXT('1')
      CALL ESUB
      CALL APNT(150.,10.,-1,-5)
      CALL SUBP(277,276)                           !SUBP: GEN 2 TEXT
      CALL APNT(165.,38.,0,-5)
      CALL TEXT('2')
      CALL APNT(300.,10.,0,-5)
      CALL SUBP(278,276)                           !SUBP: GEN 3 TEXT
      CALL APNT(315.,38.,0,-5)
      CALL TEXT('3')
      CALL APNT(220.,270.,1,-4)
      CALL SUBP(202)                               !SUBP: SW FOR ST 1
      CALL TEXT('ST-1')
```

```
      CALL SUBP(118,101)                       !SUBP: CB OP LN E4-6
      CALL APNT(600.,970.,-1,-4)
      CALL VECT(150.,0.)                       ! 750,970
      CALL VECT(0.,-70.)                       ! 750,900
      CALL SUBP(162,160)                       !SUBP: SW OP ST-10 UL
      CALL OFF(162)
      CALL SUBP(62,60)                         !SUBP: SW CL ST-10 UL
      CALL APNT(750.,880.,-1,-4)
      CALL VECT(0.,-40.)                       ! 750,840
      CALL SUBP(163,160)                       !SUBP: SW OP ST-10 LL
      CALL OFF(163)
      CALL SUBP(63,60)                         !SUBP: SW CL ST-10 LL
C
C     DRAW CENTER SECTIONALIZING SWITCH FOR ST 10.
C
      CALL APNT(750.,860.,-1,-4)
      CALL VECT(20.,0.)                        ! 770,860
      CALL SUBP(164)                           !SUBP: SW CL MID ST 10
      CALL SWCLH(4)
      CALL ESUB
      CALL OFF(164)
      CALL SUBP(64)                            !SUBP: SW OP MID ST 10
      CALL SWOPH(4)
      CALL ESUB
      CALL APNT(790.,860.,-1,-4)
      CALL VECT(20.,0.)                        ! 810,860
      CALL RDOT(-60.,-40.,-1,-4)               ! 750,820
      CALL VECT(0.,-300.)                      ! 750,520
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(17,1)                          !SUBP: CL CB E10-11
      CALL SUBP(117,101)                       !SUBP: OP CP E10-11
      CALL APNT(750.,500.,-1,-4)
      CALL VECT(0.,-99.)                       !750,401
C
C     WRITE THE TITLE OF THE PICTURE IN ITALICS.
C
      CALL STAT(-1)
      CALL APNT(400.,500.,-1,-5)
      CALL TEXT('HV TRANSMISSION')
C
C     CALL SUBROUTINE TO COMPLETE PICTURE .
C
      CALL THERST
      STOP
      END
```

```
C
C       A SUBROUTINE TO FINISH THE HV TRANSMISSION LINE
C       PICTURE.  THE SUBROUTINE IS NECESSITATED BY THE FORTRAN
C       COMPILER NOT BEING ABLE TO HANDLE ALL THE NECESSARY
C       PROGRAM AT ONCE IN MEMORY.
C
        SUBROUTINE THERST
        DIMENSION TIM(4)
C
C       DRAW THE INTERCONNECTION REPRESENTATION.
C
        CALL APNT(633.,550.,-1,-5)
        CALL TEXT('INT CON')
        CALL STAT(1)
        CALL APNT(680.,500.,-1,-4)
        CALL SUBP(13,1)                         !SUBP: CL CB INT CON
        CALL SUBP(113,101)                      !SUBP: OP CB INT CON
        CALL APNT(690.,500.,-1,-4)
C       DRAW LINE C1-11 WITH CIRCUIT BREAKERS.
C
        CALL VECT(0.,-220.)                     ! 690,280
        CALL RDOT(-10.,-20.,-1,-4)
        CALL SUBP(14,1)                         !SUBP: CB CL LN C1-11
        CALL SUBP(114,101)                      !SUBP: CB OP LN C1-11
        CALL APNT(690.,260.,-1,-4)
        CALL VECT(0.,-40.)                      ! 690,220
        CALL VECT(-290.,0.)                     ! 400,220
        CALL VECT(0.,180.)                      ! 400,400
        CALL VECT(-180.,0.)                     ! 220,400
        CALL VECT(0.,-40.)                      ! 220,360
        CALL RDOT(-10.,-20.,-1,-4)
        CALL SUBP(11,1)                         !SUBP: CB CL LN C1-11
        CALL SUBP(111,101)                      !SUBP: CB OP LN C1-11
        CALL APNT(220.,340.,-1,-4)
        CALL VECT(0.,-39.)                      ! 220,301 ST 1 BUS
C
C       DRAW LINE D1-11 WITH CIRCUIT BREAKERS.
C
        CALL RDOT(50.,0.,-1,-4)                 ! 270,301 PB FOR D1-11
        CALL VECT(0.,39.)                       ! 270,340
        CALL RDOT(-10.,0.,,-4)
        CALL SUBP(12,1)                         !SUBP: CB CL LN D1-11
        CALL SUBP(112,101)                      !SUBP: CB OP LN D1-11
        CALL APNT(270.,360.,-1,-4)
        CALL VECT(0.,20.)                       ! 270,380
        CALL VECT(110.,0.)                      ! 380,380
        CALL VECT(0.,-180.)                     ! 380,200
        CALL VECT(490.,0.)                      ! 870,200
        CALL VECT(0.,60.)                       ! 870,260
        CALL RDOT(-10.,0.,,-4)
        CALL SUBP(15,1)                         !SUBP: CB CL LN D1-11
        CALL SUBP(115,101)                      !SUBP: CB OP LN D1-11
        CALL APNT(750.,370.,1,-5)
        CALL SUBP(206)                          !SUBP: SW FOR ST-11
        CALL TEXT('ST-11')
        CALL ESUB
C
C       DRAW THE CONNECTIONS TO SUBSTATION 4 WITH CIRCUIT BREAKERS.
C
        CALL APNT(425.,970.,-1,-4)
        CALL VECT(0.,-30.)                      ! 410,940
```

```
      CALL RDOT(-10.,-20.,,-4)
      CALL SUBP(19,1)                            !SUBP: CB CL ST 4 LF
      CALL SUBP(119,101)                         !SUBP: CB OP ST 4 LF
      CALL APNT(425.,920.,-1,-4)
C
C     DRAW A SUBPICTURE OF THE SUBSTATION BOX.
C
      CALL SUBP(256)                             !SUBP: SUB ST BOX
      CALL VECT(0.,-60.)                         ! 425,860
      CALL RDOT(-30.,0.,,-4)                     ! 395,860
      CALL HBUS(120.,4)                          ! 515,860
      CALL RDOT(30.,20.,,-4,,4)                  ! 545,880
      CALL VECT(0.,-100.)                        ! 545,780
      CALL VECT(-180.,0.)                        ! 395,780
      CALL VECT(0.,100.)                         ! 395,880
      CALL VECT(180.,0.)                         ! 545,880
      CALL RDOT(-60.,-20.,,-4,,1)                ! 485,860
      CALL VECT(0.,60.)                          ! 485,920
      CALL ESUB
      CALL RDOT(-10.,0.,,-4)
      CALL SUBP(20,1)                            !SUBP: CL CB SUB ST-4
      CALL SUBP(120,101)                         !SUBP: OP CB SUB ST-4
      CALL APNT(485.,940.,-1,-4)
      CALL VECT(0.,60.)                          !485,1000
      CALL RDOT(-85.,-190.,1,-5)
      CALL SUBP(204)                             !SUBP: SW FOR SUB ST-4
      CALL TEXT('SUB ST-4')
      CALL ESUB
C
C     DRAW THE REPRESENTATION OF SUBSTATION 6 WITH BUS.
C
      CALL APNT(560.,970.,-1,-4,,1)
      CALL VECT(0.,-180.)                        ! 560,790
      CALL SUBP(257,256)                         !SUBP: BOX FOR SUB ST-6
      CALL VECT(0.,180.)                         ! 620,970
      CALL RDOT(-85.,-290.,1,-5)                 ! 535,680
      CALL SUBP(205)                             !SUBP: SW FOR SUB ST-6
      CALL TEXT('SUB ST-6')
      CALL ESUB
C
C     PLACE THE NUMBER FOR THE TIME.
C
      CALL APNT(885.,850.,-1,-5)
      CALL NMBR(239,TIM(1),4,'(A4)')
      CALL NMBR(240,TIM(2),4,'(A4)')
C
C     ADD 'MENU' AS THE SWITCH TO REGAIN THE INTRODUCTION MENU.
C
      CALL APNT(930.,500.,1,-5)
      CALL SUBP(203)
      CALL TEXT('MENU')
      CALL ESUB
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
      CALL SUBP(90)                              !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                              !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
```

```
C       DRAW THE 'NO' SUBPICTURE.
C
        CALL SUBP(92)
        CALL APNT(950.,700.,1,-5,-1)
        CALL TEXT('NO')
        CALL ESUB
C
        CALL ESUB
C
        CALL OFF(90)
C
C       SAVE THE PICTURE IN A FILE CALLED 'HVTRNS.DPY'.
C
        CALL SAVE('HVTRNS.DPY')
C
        RETURN
        END
```

THE FOLLOWING IS HVTRNS.CMD:

```
HVTRNS/-CP,TI:/SH=HVTRNS,DSPSUB,HVREST,GLIB/LB
/
ASG=GRO:1
//
```

THE 'SUBST4.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE SUBSTATION 4 DISPLAY. IT USES SUBROUTINES CONTAINED IN A FILE CALLED 'DSPSUB.FTN'. EDIT THE FILES USING THE EDIT UTILITY.
TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
                >FOR SUBST4=SUBST4
THEN            >FOR DSPSUB=DSPSUB
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
                >FOR SUBST4=SUBST4/LI:1
THEN            >FOR DSPSUB=DSPSUB/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
                >TKB @SUBST4
```

THIS CAUSES FILES, 'SUBST4.OBJ' AND 'DSPSUB.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'SUBST4.TSK', TO BE CREATED BY THE 'TKB' COMMAND.
THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'SUBST4.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.
TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
                >RUN SUBST4
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'SUBST4.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE SUBSTATION 4 DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
301	HV TRANSMISSION TEXT FOR SW
302	NETWORK SYSTEM TEXT FOR SW
303	INDUST LOAD TEXT FOR SW
304	MENU TEXT FOR SW
.	.
.	.
.	.
340	TRANSFORMER UPPER LEFT
341	TRANSFORMER UPPER RIGHT
342	LEFT BUS
343	RIGHT BUS
344	RESIDENTIAL SECONDARY TRANSF
345	RESIDENTIAL LD 6
346	RESIDENTIAL LD 5
347	RESIDENTIAL LD 4
348	RESIDENTIAL LD 3
349	RESIDENTIAL LD 2
350	RESIDENTIAL LD 1
351	RESIDENTAIL ST LTG

```
C
C       A PROGRAM TO DRAW THE PICTURE FOR SUBSTATION 4.
C
        COMMON/DFILE/IBUF(1050)
        CALL INIT(1050)
C
C       DRAW SUBPICTURE OF TRANSFORMER AT HV END.  DASHED LINES SHOW
C       WHERE THE PICTURE PICKS UP FROM FORMER PICTURE OR WHERE THE
C       REMAINDER OF THE PICTURE IS.
C
        CALL APNT(100.,920.,-1,-4)
        CALL VECT(700.,0.,-1,,-1,4)                          ! 800,920
        CALL APNT(115.,880.,-1,-5,-1,1)
        CALL STAT(-1)
        CALL TEXT('E4-6')
        CALL APNT(350.,970.,1,-5)
        CALL STAT(1)
        CALL SUBP(301)
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
        CALL APNT(715.,880.,-1,-5)
        CALL STAT(-1)
        CALL TEXT('F4-6')
        CALL APNT(365.,700.,-1,-5)
        CALL TEXT('SUBSTATION-4')
        CALL STAT(1)
C
C       DRAW A SUBPICTURE OF THE TRANSFORMERS.
C
        CALL APNT(150.,720.,-1,-4)
        CALL SUBP(340)                             !SUBP: TOP TRANS'S
        CALL VECT(0.,60.,-1,4)
        CALL TRANSF
        CALL VECT(0.,50.)
        CALL ESUB
        CALL ARROWD(-1,4,-1)
C
C       DRAW THE TOP CIRCUIT BREAKER ON THE LEFT.
C
        CALL APNT(140.,700.,-1,-4)
        CALL SUBP(21)                              !SUBP: TP LF CL CB
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(121)                             !SUBP: TP LF OP CB
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(150.,700.,-1,-4)
        CALL VECT(0.,-200.)
        CALL APNT(140.,480.,-1,-4)
        CALL SUBP(22,21)                           !SUBP: LWR LF CL CB
        CALL SUBP(122,121)                         !SUBP: LWR LF OP CB
        CALL APNT(150.,480.,-1,-4)
        CALL VECT(0.,-180.)                        ! 150,300
        CALL VECT(600.,0.)                         ! 750,300
        CALL VECT(0.,180.)                         ! 750,480
        CALL APNT(740.,480.,-1,-4)
        CALL SUBP(23,21)                           !SUBP: LWR RT CL CB
        CALL SUBP(123,121)                         !SUBP: LWR RT OP CB
        CALL APNT(750.,500.,-1,-4)
        CALL VECT(0.,200.)                         ! 750,700
        CALL APNT(740.,700.,-1,-4)
```

```
      CALL SUBP(24,21)                               !SUBP: UPR RT CL CB
      CALL SUBP(124,121)                             !SUBP: UPR RT OP CB
      CALL APNT(750.,720.,-1,-4)
      CALL SUBP(341,340)                             !SUBP: UP TRANS RT
      CALL ARROWD(-1,4,-1)
C
C     DRAW THE BUS WITH CIRCUIT BREAKER.
C
C     DRAW A SUBP OF THE LEFT OF THE BUS.
C
      CALL APNT(150.,600.,-1,-4)
      CALL SUBP(342)                                 !SUBP: LFT SIDE BUS
      CALL HBUS(290.,5)
      CALL ESUB
      CALL APNT(440.,590.,-1,-4)
      CALL SUBP(25,21)                               !SUBP: MID BUS CL CB
      CALL SUBP(125,121)                             !SUBP: MID BUS OP CB
      CALL APNT(460.,600.,-1,-4)
      CALL SUBP(343,342)                             ! SUBP: RT SIDE BUS
      CALL APNT(600.,599.,-1,-4)
      CALL VECT(0.,-99.)                             ! 600,500
      CALL APNT(590.,480.,-1,-4)
      CALL SUBP(26,21)                               !SUBP: LWR RT MID CL CB
      CALL SUBP(126,121)                             !SUBP: LWR RT MID OP CB
      CALL APNT( 600.,480.,-1,-4)
      CALL VECT( 0.,-30.,)                           ! 600,550
C
C     DRAW THE  NETWORK SYSTEM AS A BOX WITH A
C     DASHED LINE BOUNDARY.
C
      CALL APNT(650.,450.,-1,-4,-1,4)
      CALL VECT(0.,-100.)
      CALL VECT(-400.,0.)
      CALL VECT(0.,100.)
      CALL VECT(400.,0.)
      CALL APNT(300.,450.,-1,-4,-1,1)
      CALL VECT(0.,30.)                              ! 300,480
      CALL APNT(290.,480.,-1,-4)
      CALL SUBP(27,21)                               !SUBP: LWR LF MID CL CB
      CALL SUBP(127,121)                             !SUBP: LWR LF MID OP CB
      CALL APNT(300.,500.,-1,-4)
      CALL VECT(0.,99.)                              ! 300,599
C
C     WRITE THE TEXT FOR THE NETWORK SYSTEM.
C
      CALL APNT(350.,390.,1,-5)
      CALL SUBP(302)                                 !SUBP: SW FOR NETWORK
      CALL TEXT('NETWORK SYSTEM')
      CALL ESUB
C
C     DRAW THE REPRESENTATION OF THE INDUSTRIAL LOAD.
C
      CALL APNT(750.,400.,-1,-4)
      CALL VECT(30.,0.)
      CALL APNT(780.,450.,-1,-4,-1,4)
      CALL VECT(110.,0.)
      CALL VECT(0.,-100.)
      CALL VECT(-110.,0.)
      CALL VECT(0.,100.)
      CALL APNT(795.,410.,1,-5,-1,1)
      CALL SUBP(303)                                 !SUBP: SW FOR INDUS LD
      CALL TEXT('INDUST')
      CALL ESUB
      CALL APNT(810.,370.,-1,-5)
      CALL TEXT('LOAD')
C
```

```
C        DRAW THE RESIDENTIAL LOADS WITH THE TRANSFORMER.
C
         CALL APNT(450.,150.,-1,-4)
         CALL SUBP(344,340)                        !SUBP: RES SEC TRANS
         CALL APNT(150.,150.,-1,-4)
         CALL VECT(600.,0.)
         CALL SUBP(345)                            !SUBP: LOAD 6
         CALL LOAD
         CALL ESUB
         CALL APNT(650.,150.,-1,-4)
         CALL SUBP(346,345)                        !SUBP: LOAD 5
         CALL APNT(550.,150.,-1,-4)
         CALL SUBP(347,345)                        !SUBP: LOAD 4
         CALL APNT(450.,150.,-1,-4)
         CALL SUBP(348,345)                        !SUBP: LOAD 3
         CALL APNT(350.,150.,-1,-4)
         CALL SUBP(349,345)                        !SUBP: LOAD 2
         CALL APNT(250.,150.,-1,-4)
         CALL SUBP(350,345)                        !SUBP: LOAD 1
         CALL APNT(150.,150.,-1,-4)
         CALL SUBP(351,345)                        !SUBP: LOAD ST LTNG
         CALL APNT(330.,20.,-1,-5)
         CALL STAT(-1)
         CALL TEXT('RESIDENTIAL LOADS')
         CALL STAT(1)
C
C
         CALL APNT(200.,620.,-1,-5)
         CALL NMBR(317,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' VOLTS')
         CALL STAT(1)
         CALL APNT(550.,620.,-1,-5)
         CALL NMBR(318,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' VOLTS')
         CALL STAT(1)
C
C        ADD THE MENU SWITCH.
C
         CALL APNT(930.,500.,1,-5)
         CALL SUBP(304)
         CALL TEXT('MENU')
         CALL ESUB
C
C
C        WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C        FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
         CALL SUBP(90)                             !SUBP: R U SURE
         CALL APNT(870.,750.,-1,-5,1)
         CALL TEXT('YOU SURE?')
C
C        DRAW THE 'YES' SUBPICTURE.
C
         CALL SUBP(91)                             !SUBP: CK IF YES
         CALL APNT(880.,700.,1,-5,-1)
         CALL TEXT('YES')
         CALL ESUB
C
C        DRAW THE 'NO' SUBPICTURE.
C
         CALL SUBP(92)
         CALL APNT(950.,700.,1,-5,-1)
         CALL TEXT('NO')
         CALL ESUB
C
```

```
      CALL ESUB
C
      CALL OFF(90)
C
C     SAVE THE PICTURE IN A FILE CALLED 'SUBST4.DPY'.
C
      CALL SAVE('SUBST4.DPY')
C
      STOP
      END
```

THE FOLLOWING IS CONTAINED IN FILE SUBST4.CMD:

```
SUBST4/-CP,TI:/SH=SUBST4,DSPSUB,GLIB/LB
/
ASG=GRO:1
//
```

THE 'SUBST6.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE SUBSTATION 6 DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR SUBST6=SUBST6
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR SUBST6=SUBST6/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @SUBST6
```

THIS CAUSES A FILE, 'SUBST6.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'SUBST6.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'SUBST6.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN SUBST6
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'SUBST6.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE SUBSTATION 6 DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
401	HV TRANSMISSION TEXT FOR SW
402	LOAD CENTER TEXT FOR SW
403	NETWORK SYSTEM TEXT FOR SW
404	MENU TEXT FOR SW
.	.
.	.
.	.
420	LEFT SIDE TRANSFORMER
421	RIGHT SIDE TRANSF
422	RIGHT SIDE BOX
423	LEFT SIDE BOX

THE FOLLOWING IS THE CONTENTS OF SUBST6.CMD:

```
SUBST6/-CP,TI:/SH=SUBST6,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

```
C
C       PROGRAM TO DRAW THE PICTURE FOR SUBSTATION-6.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       DRAW TRANSFORMER AND TEXT FOR INCOMING LINES.
C
        CALL APNT(100.,920.,-1,-4)
        CALL VECT(700.,0.,-1,,-1,4)                 ! 800,920
        CALL APNT(175.,890.,-1,-5,-1,1)
        CALL TEXT('E4-6')
C
C       DRAW THE SWITCH FOR HV TRANSMISSION PICTURE.
C
        CALL APNT(350.,940.,1,-5)
        CALL SUBP(401)                              !SUBP: HV-TRNS SW
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
C
C       TEXT FOR OTHER INCOMING LINE.
C
        CALL APNT(663.,890.,-1,-5,-1,1)
        CALL TEXT('E6-10')
C
C       TEXT FOR PICTURE LABEL 'SUBSTATION-6'.
C
        CALL APNT(365.,700.,-1,-5)
        CALL STAT(-1)                               !TURN ON ITALICS.
        CALL TEXT('SUBSTATION-6')
        CALL STAT(1)
C
C       DRAW A SUBPICTURE OF THE TRANSFORMER, TO BE COPIED LATER.
        CALL APNT(200.,720.,-1,-4)
        CALL SUBP(420)                              !SUBP: LT TRANSF
        CALL VECT(0.,50.)
        CALL TRANSF
        CALL VECT(0.,60.)
        CALL ARROWD(-1,4,-1)
        CALL ESUB
C
C       DRAW THE CIRCUIT BREAKER ON THE LEFT TOP, CB #1.
C
        CALL APNT(190.,700.,-1,-4)
        CALL SUBP(34)                               !SUBP: CL CB #1
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(134)                              !SUBP: OP CB #1
        CALL CBOPN(4)
        CALL ESUB
C
C       CONTINUE WITH CONNECTIONS TO THE BUS AND THE MAIN BUS ITSELF.
C
        CALL APNT(200.,700.,-1,-4)
        CALL VECT(0.,-100.)
        CALL HBUS(500.,5)
        CALL VECT(0.,100.,-1,4)
C
C       DRAW THE CIRCUIT BREAKER  #2.
C
        CALL RDOT(-10.,0.,,-4)
```

```
      CALL SUBP(35,34)                         !SUBP: CL CB #2
      CALL SUBP(135,134)                       !SUBP: OP CB #2
C
C     COPY THE TRANSFORMER.
C
      CALL APNT(700.,720.,-1,-4)
      CALL SUBP(421,420)                       !SUBP: RT TRANSF
C
C     BEGIN DRAWING THE CIRCUIT BREAKERS AND CONNECTIONS TO THE
C     BOXES FOR THE NETWORK SYSTEM AND THE LOAD CENTER.
      CALL APNT(325.,600.,-1,-4)
      CALL VECT(0.,-200.)                      ! 325,400
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(32,34)                         !SUBP: CL CB #4
      CALL SUBP(132,134)                       !SUBP: OP CB #4
      CALL APNT(325.,380.,-1,-4)
      CALL VECT(0.,-80.)
C
      CALL APNT(450.,600.,-1,-4)
      CALL VECT(0.,-200.)
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(33,34)                         !SUBP: CL CB #5
      CALL SUBP(133,134)                       !SUBP: OP CB #5
      CALL APNT(450.,380.,-1,-4)
      CALL VECT(0.,-80.)
C
      CALL APNT(575.,300.,-1,-4)
      CALL VECT(0.,80.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(38,34)                         !SUBP: CL CB #6
      CALL SUBP(138,134)                       !SUBP: OP CB #6
      CALL APNT(575.,400.,-1,-4)
      CALL VECT(0.,200.)
C
      CALL APNT(700.,600.,-1,-4)
      CALL VECT(0.,-200.)
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(37,34)                         !SUBP: CL CB #7
      CALL SUBP(137,134)                       !SUBP: OP CB #7
      CALL APNT(700.,380.,-1,-4)
      CALL VECT(0.,-80.)
C
C     DRAW THE BOXES FOR THE OTHER SYSTEMS.
C
      CALL RDOT(50.,0.,-1,-4)
      CALL SUBP(422)                           !SUBP: BOX
      CALL VECT(0.,-100.,-1,4,-1,4)                    !750,200
      CALL VECT(-225.,0.,-1)                   ! 525,200
      CALL VECT(0.,100.)                       ! 525,300
      CALL VECT(225.,0.)                       ! 750,300
      CALL ESUB
C
C     WRITE THE TEXT IN THE RIGHT BOX.
C
      CALL APNT(560.,260.,1,-5,-1,1)
      CALL SUBP(402)                           !SUBP: SW FOR LD CTR
      CALL TEXT('LOAD CENTER')
      CALL APNT(570.,225.,1,-5)
      CALL TEXT('SUBSTATION')
      CALL ESUB
C
C     COPY THE LEFT BOX AND ADD THE TEXT.
C
      CALL APNT(500.,300.,-1,-4)
      CALL SUBP(423,422)                       !SUBP: BOX
      CALL APNT(340.,260.,1,-5,-1,1)
```

```
      CALL SUBP(403)                          !SUBP: SW FOR NTWK
      CALL TEXT('NETWORK')
      CALL APNT(347.,225.,1,-5)
      CALL TEXT('SYSTEM')
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
      CALL SUBP(90)                           !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                           !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
C     DRAW THE 'NO' SUBPICTURE.
C
      CALL SUBP(92)
      CALL APNT(950.,700.,1,-5,-1)
      CALL TEXT('NO')
      CALL ESUB
C
      CALL ESUB
C
C     THE MANUAL BUS WITH THE TIE CIRCUIT BREAKER IS DRAWN NEXT.
C     THE LINES ARE DRAWN AT A INTENSITY OF TWO SO THAT IT WILL
C     SHOW ONLY WHEN IT DESIRED TO OPERATE IN THE MANUAL MODE.
C
      CALL APNT(200.,750.,-1,-1,-1,1)
      CALL VECT(-50.,0.)
      CALL VECT(0.,-100.)
C
C     DRAW THE SUBPICTURES OF THE OPEN AND CLOSED MANUAL SWITCHES.
C     THE SWITCHES OPEN AND CLOSE ONLY ON THE SCREEN.
C
C
      CALL SUBP(71)                           !SUBP: SW OP UP LT
      CALL SWOPV(1)
      CALL ESUB
      CALL SUBP(171)                          !SUBP: SW CL UP LT
      CALL SWCLV(1)
      CALL ESUB
      CALL OFF(171)
      CALL APNT(150.,630.,-1,-1,-1,1)
C
C     CONTINUE WITH THE LINE CONNECTION TO THE BUS.
C
      CALL VECT(0.,-130.,-1,1)
C
C     DRAW THE DIM BUS.
C
      CALL HBUS(600.,1)
C
C     CONTINUE WITH THE MANUAL SWITCHES AND CONNECTIONS.
C
      CALL APNT(700.,750.,-1,-1)
      CALL VECT(50.,0.)
      CALL VECT(0.,-100.)
      CALL SUBP(72,71)                        !SUBP: SW OP UP RT
      CALL SUBP(172,171)                      !SUBP: SW CL UP RT
      CALL OFF(172)
      CALL APNT(750.,630.,-1,-1)
      CALL VECT(0.,-230.,-1,1)                ! 750,400
```

```
      CALL SUBP(73,71)                          !SUBP: SW OP LWR RT
      CALL SUBP(173,171)                        !SUBP: SW CL LWR RT
      CALL OFF(173)
      CALL APNT(750.,380.,-1,-1)
      CALL VECT(0.,-30.,-1,1)
      CALL VECT(-50.,0.)                        ! 700,350
C
      CALL APNT(625.,500.,-1,-1)
      CALL VECT(0.,-100.)
      CALL SUBP(74,71)                          !SUBP: SW OP LWR MID RT
      CALL SUBP(174,171)                        !SUBP: SW CL LWR MID RT
      CALL OFF(174)
      CALL APNT(625.,380.,-1,-1)
      CALL VECT(0.,-30.,-1,1)
      CALL VECT(-50.,0.)
C
      CALL APNT(500.,500.,-1,-1)
      CALL VECT(0.,-100.)
      CALL SUBP(75,71)                          !SUBP: SW OP LWR MID LT
      CALL SUBP(175,171)                        !SUBP: SW CL LWR MID LT
      CALL OFF(175)
      CALL APNT(500.,380.,-1,-1)
      CALL VECT(0.,-30.,-1,1)
      CALL VECT(-50.,0.)
C
      CALL APNT(375.,500.,-1,-1)
      CALL VECT(0.,-100.)
      CALL SUBP(76,71)                          !SUBP: SW OP LWR LT
      CALL SUBP(176,171)                        !SUBP: SW CL LWR LT
      CALL OFF(176)
      CALL APNT(375.,380.,-1,-1)
      CALL VECT(0.,-30.)
      CALL VECT(-50.,0.)                        ! 325,350
C
C     DRAW THE BUS TIE CIRCUIT BREAKER.
C
      CALL APNT(200.,500.,-1,-1)
      CALL VECT(0.,40.)
      CALL RDOT(-10.,0.)
      CALL SUBP(36)                             !SUBP: BUS TIE CB
      CALL CBCLD(1)
      CALL ESUB
      CALL SUBP(136)                            !SUBP: BUS TIE CB OP
      CALL CBOPN(1)
      CALL ESUB
      CALL APNT(200.,560.,-1,-1)
      CALL VECT(0.,40.)
C
C
C     ADD A MENU SWITCH.
      CALL APNT(930.,500.,1,-5)
      CALL SUBP(404)
      CALL TEXT('MENU')
      CALL ESUB
C
C     ADD THE BUS VOLTAGE.
C
      CALL APNT(400.,620.,-1,-5)
      CALL NMBR(958,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL STAT(1)
C
      CALL OFF(90)
C
C     SAVE THE PICTURE IN A FILE CALLED 'SUBST6.DPY'.
C
      CALL SAVE('SUBST6.DPY')
C
      STOP
      END
```

THE 'LDCTR.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE LOAD CENTER DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR LDCTR=LDCTR
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR LDCTR=LDCTR/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @LDCTR
```

THIS CAUSES A FILE, 'LDCTR.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'LDCTR.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'LDCTR.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN LDCTR
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'LDCTR.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE LOAD CENTER DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
501	HV TRANSMISSION TEXT FOR SW
502	SUBSTATION 6 TEXT FOR SW
503	MENU TEXT FOR SW
504	SUB ST 6 BUS VOLTAGE
505	LEFT SIDE SECONDARY VOLTAGE
506	LEFT SIDE SECONDARY AMPS
507	RIGHT SIDE SECONDARY VOLTS
508	RIGHT SIDE SECONDARY AMPS
.	.
.	.
.	.
520	TRANSFORMER UPPER LEFT
521	TRANSFORMER UPPER RIGHT
522	LOAD #1
523	LOAD #2
524	LOAD #3
525	LOAD #4

THE FOLLOWING IS CONTAINED IN THE FILE LDCTR.CMD:

```
LDCTR/-CP,TI:/SH=LDCTR,DSPSUB,GLIB/LB
/
ASG=GRO:1
//
```

```
C
C       PROGRAM TO DRAW THE LOAD CENTER SUBSTATION.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       START AT THE TOP OF THE LEFT LOAD AND PROCEED CLOCKWISE.
C
        CALL APNT(200.,300.,-1,-4,-1,1)
        CALL VECT(0.,200.,-1)
C
C       DRAW CIRCUIT BREAKER SS-1.
C
        CALL RDOT(-10.,0.,,-4)
        CALL SUBP(39)                           !SUBP: CL CB #SS-1
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(139)                          !SUBP: OP CB #SS-1
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(200.,520.,-1,-4)
        CALL SUBP(520)                          !SUBP: TRANS ON LT
        CALL VECT(0.,110.)                      ! 200,630
        CALL TRANSF                             ! 200,670
        CALL VECT(0.,110.)                      ! 200,780
        CALL ESUB
C
C       DRAW THE #6 CIRCUIT BREAKER FROM SUBSTATION 6.
C
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(38,39)                        !SUBP: CL CB #6
        CALL SUBP(138,139)                      !SUBP: OP CB #6
        CALL APNT(200.,800.,-1,-4)
        CALL VECT(0.,100.)                      ! 200,900
C
C       DRAW THE BUS OF SUBSTATION 6.
C
        CALL HBUS(500.,5)                       ! 700,900
C
C       CONTINUE WITH THE RIGHT PORTION.
C
        CALL VECT(0.,-100.,-1,4)                ! 700.,800
        CALL RDOT(-10.,-20.,,-4)
        CALL SUBP(37,39)                        !SUBP: CL CB #7
        CALL SUBP(137,139)                      !SUBP: OP CB #7
        CALL APNT(700.,520.,-1,-4)
        CALL SUBP(521,520)                      !SUBP: TRANS ON RT
        CALL APNT(690.,500.,-1,-4)
        CALL SUBP(40,39)                        !SUBP: CL CB #SS-3
        CALL SUBP(140,139)                      !SUBP: OP CB #SS-3
        CALL APNT(700.,500.,-1,-4)
        CALL VECT(0.,-200.)                     ! 700,300
C
C       DRAW THE DASHED LINE ACROSS THE PICTURE TO DEPICT THE PORTION
C       THAT IS ALREADY DEPICTED ON THE SUBSTATION 6 PICTURE.
C
        CALL APNT(800.,730.,-1,-4,-1,4)
        CALL VECT(-700.,0.)                     ! 100,750
C
C       DRAW A DASHED LINE ABOVE THE BUS TO DEPICT THE FACT THAT THE
C       HV TRANSMISSION LINE PICTURE CONTAINS THE GENERATION WHICH IS
```

```
C       THE INPUT TO THE PICTURE.  THE TEXT WILL BE THE SWITCH FOR
C       CHANGING TO THIS PICTURE.
C
        CALL APNT(100.,940.,-1,-4,-1)
        CALL VECT(700.,0.)                        ! 800,950
        CALL APNT(350.,950.,1,-5,-1,1)
        CALL SUBP(501)                            !SUBP: SW FOR HVTRNS
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
C
C       WRITE THE TEXT FOR SUBSTATION 6 AND SENSITIZE IT FOR THE
C       LIGHT PEN.
C
        CALL APNT(365.,810.,1,-5)
        CALL SUBP(502)                            !SUBP: SW FOR ST-6
        CALL TEXT('SUBSTATION-6')
        CALL ESUB
C
C       DRAW THE SECONDARIES STARTING ON THE RIGHT.
C
        CALL APNT(300.,300.,-1,-4)
        CALL VECT(-200.,0.)                       ! 100,300
        CALL VECT(0.,-50.)                        ! 100,250
        CALL RDOT(-10.,-20.,-1,-4)
        CALL SUBP(42,39)                          !SUBP: CL CB #1
        CALL SUBP(142,139)                        !SUBP: OP CB #1
        CALL APNT(100.,230.,-1,-4)
        CALL VECT(0.,-60.)
        CALL SUBP(522)                            !SUBP: LOAD
        CALL LOAD
        CALL ESUB
C
C       WRITE THE NAME OF THE PICTURE IN ITALICS.
C
        CALL STAT(-1)
        CALL APNT(297.,600.,-1,-5)
        CALL TEXT('LOAD CENTER SUBSTATION')
        CALL APNT(318.,550.,-1,-5)
        CALL TEXT('SECONDARY SELECTIVE')
C
C       DRAW THE TIE BREAKER.
C
        CALL APNT(200.,450.,-1,-4)
        CALL VECT(240.,0.)                        ! 440,450
        CALL RDOT(0.,-10.,-1,-4)
        CALL SUBP(41,39)                          !SUBP: CL CB #SS-2
        CALL SUBP(141,139)                        !SUBP: OP CB #SS-2
        CALL APNT(460.,450.,-1,-4)
        CALL VECT(240.,0.)                        ! 700,450
C
        CALL APNT(300.,300.,-1,-4)
        CALL VECT(0.,-50.)                        ! 300,250
        CALL RDOT(-10.,-20.,-1,-4)
        CALL SUBP(43,39)                          !SUBP: CL CB #2
        CALL SUBP(143,139)                        !SUBP: OP CB #2
        CALL APNT(300.,230.,-1,-4)
        CALL VECT(0.,-60.)                        ! 300,170
        CALL SUBP(523,522)                        !SUBP: LOAD #2
C
        CALL APNT(800.,300.,-1,-4)
        CALL VECT(-200.,0.)
        CALL VECT(0.,-50.)
        CALL RDOT(-10.,-20.,-4,-4)
        CALL SUBP(44,39)                          !SUBP: CL CB #3
        CALL SUBP(144,139)                        !SUBP: OP CB #3
        CALL APNT(600.,230.,-1,-4)
```

```
      CALL VECT(0.,-60.)                         ! 600,170
      CALL SUBP(524,522)                         !SUBP: LOAD #3
C
      CALL APNT(800.,300.,-1,-4)
      CALL VECT(0.,-50.)
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(45,39)                           !SUBP: CL CB #4
      CALL SUBP(145,139)                         !SUBP: OP CB #4
      CALL APNT(800.,230.,-1,-4)
      CALL VECT (0.,-60.)                        !800,170
      CALL SUBP(525,522)                         !SUBP: LOAD #4
C
C
C     WRITE THE TEXT AT THE BOTTOM TO DESCRIBE THE DISTRIBUTION
C     SYSTEM.
      CALL APNT(235.,40.,-1,-5)
      CALL TEXT('240 V - 3',-1,'B',' - 3W  DISTRIBUTION')
C
C     ADD THE MENU SWITCH.
C
      CALL APNT(930.,500.,1,-5)
      CALL STAT(1)
      CALL SUBP(503)
      CALL TEXT('MENU')
      CALL ESUB
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
C
      CALL SUBP(90)                              !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                              !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
C     DRAW THE 'NO' SUBPICTURE.
C
      CALL SUBP(92)
      CALL APNT(950.,700.,1,-5,-1)
      CALL TEXT ('NO')
      CALL ESUB
      CALL ESUB
C
      CALL OFF(90)
C
C     BUS VOLTAGE ON SUBSTATION 6.
C
      CALL APNT(720.,890.,-1,-5)
      CALL NMBR(958,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
C
C     LEFT SIDE SECONDARY VOLTAGE AND CURRENT.
C
      CALL APNT(220.,400.,-5,-5)
      CALL STAT(1)
      CALL NMBR(956,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL APNT(220.,350.,-1,-5)
      CALL STAT(1)
```

```
      CALL NMBR(955,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' AMPS')
C
C     RIGHT SIDE SECONDARY VOLTAGE AND CURRENT.
C
      CALL APNT(720.,400.,-1,-5)
      CALL STAT(1)
      CALL NMBR(957,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL APNT(720.,350.,-1,-5)
      CALL STAT(1)
      CALL NMBR(954,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' AMPS')
C
C     SAVE THE PICTURE IN A FILE CALLED 'LDCTR.DPY'.
C
      CALL SAVE('LDCTR.DPY')
C
      STOP
      END
```

THE 'NETSYS.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE NETWORK SYSTEM DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR NETSYS=NETSYS
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR NETSYS=NETSYS/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @NETSYS
```

THIS CAUSES A FILE, 'NETSYS.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'NETSYS.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'NETSYS.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN NETSYS
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'NETSYS.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE NETWORK SYSTEM DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
601	TOP HV TRANSMISSION TEXT FOR SW
602	SUBSTATION 6 TEXT FOR SW
603	SUBSTATION 4 TEXT FOR SW
604	BOTTOM HV TRANSMISSION TEXT FOR SW
605	MENU TEXT FOR SW
.	.
.	.
.	.
620	SUBSTATION 6 BUS
621	SUBSTATION 4 BUS
622	TRANSF LOWER RIGHT
623	TRANSF UPPER RIGHT
624	TRANSF LOWER LEFT
625	TRANSF UPPER LEFT
626	LOAD #1
627	LOAD #2

```
C
C       PROGRAM TO DRAW THE NETWORK SYSTEM.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       START WITH THE TEXT AT THE TOP.
C
        CALL APNT(350.,950.,1,-5,-1,1)
        CALL SUBP(601)                          !SUBP:HVTRNS SW
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
C
C       DRAW DASHED LINE TO INDICATE WHERE SEPARATION OF PRIOR PICTURES.
C
        CALL APNT(100.,940.,-1,-4,-1,4)
        CALL VECT(700.,0.)                      ! 800,940
C
C       DRAW THE BUS FOR STATION 6 AS A SUBPICTURE.
C
        CALL APNT(750.,900.,-1,-4,-1,1)
        CALL SUBP(620)                          !SUBP: HBUS ST-6
        CALL HBUS(-600.,5)
        CALL ESUB
C
C       CONTINUE WITH TEXT AND DASHED LINES.
C
        CALL APNT(365.,840.,1,-5,-1,1)
        CALL SUBP(602)                          !SUBP: SUBST-6 SW
        CALL TEXT('SUBSTATION-6')
        CALL ESUB
        CALL APNT(100.,780.,-1,-4,-1,4)
        CALL VECT(700.,0.)
C
C       WRITE THE TITLE OF THE PICTURE IN ITALICS.
C       ADD THE STREET MAINS OF THE NETWORK IN ONE-LINE DIAGRAM FORM.
C
        CALL APNT(700.,550.,-1,-4,-1,1)
        CALL VECT(-500.,0.)
        CALL STAT(-1)
        CALL APNT(350.,500.,-1,-5)
        CALL TEXT('NETWORK SYSTEM')
        CALL APNT(700.,450.,-1,-4)
        CALL VECT(-214.,0.)                     ! 486,450
        CALL APNT(414.,450.,-1,-4)
        CALL VECT(-214.,0.)                     ! 200,450
C
C       WRITE THE TEXT ON THE DISTRIBUTION SYSTEM.
C
        CALL APNT(325.,260.,-1,-5)
        CALL TEXT('208/120V - 3',-1,'B',' - 4W')
        CALL APNT(365.,238.,-1,-5)
        CALL TEXT('DISTRIBUTION')
        CALL STAT(1)
C
C       DRAW THE REPRESENTATION OF SUBSTATION-4 AND
C       HV TRANSMISSION AS ABOVE.
C
        CALL APNT(100.,220.,-1,-4,-1,4)
        CALL VECT(700.,0.)
        CALL APNT(365.,150.,1,-5,-4,1)
        CALL SUBP(603)                          !SUBP: SUBST-4 SW
```

```
      CALL TEXT('SUBSTATION-4')
      CALL ESUB
      CALL APNT(750.,100.,-1,-4)
      CALL SUBP(621,620)                    !SUBP: HBUS ST-4
      CALL APNT(100.,60.,-1,-4,-1,4)
      CALL VECT(700.,0.)
      CALL APNT(350.,30.,1,-5,-1,1)
      CALL SUBP(604,601)                    !SUBP: HVTRNS SW
C
C     START ON RIGHT SIDE AND DRAW CIRCUITS FROM BOTTOM TO TOP.
C
      CALL APNT(700.,100.,-1,-4,-1,1)
      CALL VECT(0.,50.)
      CALL RDOT(-10.,0.,,-4)
      CALL SUBP(27)                         !SUBP: CL CB #7 ST-4
      CALL CBCLD(4)
      CALL ESUB
      CALL SUBP(127)                        !SUBP: OP CB #7 ST-4
      CALL CBOPN(4)
      CALL ESUB
      CALL APNT(700.,170.,-1,-4)
      CALL SUBP(622)                        !SUBP: LWR RT TRANS
      CALL VECT(0.,95.)                     ! 700,265
      CALL TRANSF
      CALL VECT(0.,95.)
      CALL ESUB
C
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(28,27)                      !SUBP: CL CB #4
      CALL SUBP(128,127)                    !SUBP: OP CB #4
      CALL APNT(700.,420.,-1,-4)
      CALL VECT(0.,160.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(29,27)                      !SUBP: CL CB #2
      CALL SUBP(129,127)                    !SUBP: OP CB #2
      CALL APNT(700.,600.,-1,-4)
      CALL SUBP(623,622)                    !SUBP: UP RT TRANS
      CALL RDOT(-10.,0.,,-4)
      CALL SUBP(33,27)                      !SUBP: CL CB #5 ST-6
      CALL SUBP(133,127)                    !SUBP: OP CB #5 ST-6
      CALL APNT(700.,850.,-1,-4)
      CALL VECT(0.,50.)                     ! 500,900
C
C     START ON LEFT SIDE AND DRAW CIRCUITS FROM BOTTOM TO TOP.
C
      CALL APNT(200.,100.,-1,-4,-1,1)
      CALL VECT(0.,50.)
      CALL RDOT(-10.,0.,,-4)
      CALL SUBP(26,27)                      !SUBP: CL CB #8 ST-4
      CALL SUBP(126,127)                    !SUBP: OP CB #8 ST-4
      CALL APNT(200.,170.,-1,-4)
      CALL SUBP(624,622)                    !SUBP: LWR LT TRANS
C
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(30,27)                      !SUBP: CL CB #3
      CALL SUBP(130,127)                    !SUBP: OP CB #3
      CALL APNT(200.,420.,-1,-4)
      CALL VECT(0.,160.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(31,27)                      !SUBP: CL CB #1
      CALL SUBP(131,127)                    !SUBP: OP CB #1
      CALL APNT(200.,600.,-1,-4)
      CALL SUBP(625,622)                    !SUBP: UP LT TRANS
      CALL RDOT(-10.,0.,,-4)
      CALL SUBP(32,27)                      !SUBP: CL CB #4 ST-6
      CALL SUBP(132,127)                    !SUBP: OP CB #4 ST-6
```

```
      CALL APNT(200.,850.,-1,-4)
      CALL VECT(0.,50.)                          ! 500,900
C
C     DRAW THE ONE-LINE REPRESENTATION OF THE MANHOLE RING BUS.
C
      CALL APNT(411.,430.,-1,-4)
      CALL CIRCLE(40.,-1,4,-1)
C
C     DRAW THE REPRESENTATION OF THE LOADS.
C
      CALL APNT(421.,401.,-1,-4)
      CALL VECT(-21.,-21.)
      CALL SUBP(626)                             !SUBP: LOAD #1
      CALL LOAD
      CALL ESUB
      CALL APNT(479.,401.,-1,-4)
      CALL VECT(21.,-21.)
      CALL SUBP(627,626)                         !SUBP: LOAD #2
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
C
      CALL SUBP(90)                              !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                              !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
C     DRAW THE 'NO' SUBPICTURE.
C
      CALL SUBP(92)
      CALL APNT(950.,700.,1,-5,-1)
      CALL TEXT('NO')
      CALL ESUB
      CALL ESUB
C
      CALL OFF(90)
C
C     ADD THE MENU SWITCH.
C
      CALL APNT(930.,500.,1,-5)
      CALL SUBP(605)
      CALL TEXT('MENU')
      CALL ESUB
C
C     PLACE THE DATA TO BE MONITORED.
C
      CALL STAT(-1)
      CALL APNT(250.,640.,-1,-5)
      CALL TEXT('LINES')
      CALL APNT(416.,640.,-1,-5)
      CALL TEXT('1')
      CALL APNT(516.,640.,-1,-5)
      CALL TEXT('2')
      CALL APNT(616.,640.,-1,-5)
      CALL TEXT('3')
      CALL APNT(250.,600.,-1,-5)
      CALL TEXT('VOLTS')
      CALL APNT(250.,560.,-1,-5)
      CALL TEXT('AMPS')
      CALL STAT(1)
```

```
      CALL APNT(400.,600.,-1,-5)
      CALL NMBR(959,VOLTS,5,'(F5.2)')
      CALL APNT(500.,600.,-1,-5)
      CALL NMBR(960,VOLTS,5,'(F5.2)')
      CALL APNT(600.,600.,-1,-5)
      CALL NMBR(961,VOLTS,5,'(F5.2)')
      CALL APNT(400.,560.,-1,-5)
      CALL NMBR(962,VOLTS,5,'(F5.2)')
      CALL APNT(500.,560.,-1,-5)
      CALL NMBR(963,VOLTS,5,'(F5.2)')
      CALL APNT(600.,560.,-1,-5)
      CALL NMBR(964,VOLTS,5,'(F5.2)')
      CALL APNT(740.,870.,-1,-5)
      CALL NMBR(958,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL STAT(1)
      CALL APNT(740.,110.,-1,-5)
      CALL NMBR(950,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
C
C     SAVE THE PICTURE IN A FILE CALLED 'NETSYS.DPY'.
C
      CALL SAVE('NETSYS.DPY')
C
      STOP
      END
```

THE FOLLOWING IS THE CONTENTS OF NETSYS.CMD:

```
NETSYS/-CP,TI:/SH=NETSYS,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

THE 'INDLD.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE INDUSTRIAL LOAD DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR INDLD=INDLD
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR INDLD=INDLD/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @INDLD
```

THIS CAUSES A FILE, 'INDLD.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'INDLD.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'INDLD.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN INDLD
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'INDLD.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE INDUSTRIAL LOAD DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
701	HV TRANSMISSION TEXT FOR SW
702	SUBSTATION 4 TEXT FOR SW
703	VOLTAGE REGULATOR RAISE ARROW
704	VOLTAGE REGULATOR LOWER ARROW
705	STATION 1 TEXT FOR SW
706	GENERATOR 1 TEXT FOR SW
707	STATION 11 TEXT FOR SW
708	MENU TEXT FOR SW
.	.
.	.
.	.
720	LIGHTING LOAD
721	HEATING LOAD
722	POWER #1 LOAD
723	POWER #2 LOAD
724	REACTIVE LOAD

```
C
C       PROGRAM TO DRAW THE INDUSTRIAL LOADING PICTURE.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       START AT THE TOP WITH THE REPRESENTATION OF THE HV TRANSMISSION
C       AND SUBSTATION 4 WHICH PROVIDE THE SOURCE AND PROTECTION FOR
C       THIS LINE.
C
        CALL APNT(345.,950.,1,-5,-1,1)
        CALL SUBP(701)                          !SUBP: HVTRANS SW
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
C
C       DRAW DASHED LINE DEPICTION OF BOUNDARY BETWEEN HV TRANSMISSION,
C       SUBSTATION 4, AND INDUSTRIAL LOAD.
C
        CALL APNT (800.,940.,-1,-4,-1,4)
        CALL VECT(-700.,0.)                     ! 100.,940
        CALL APNT(100.,780.,-1,-4)
        CALL VECT(700.,0.)                      ! 800,780
C
C       DRAW BUS, CIRCUIT BREAKER, AND SWITCH FOR SUBSTATION 4.
C
        CALL APNT(350.,900.,-1,-4,-1,1)
        CALL HBUS(200.,5)
        CALL APNT(450.,900.,-1,-4)
        CALL VECT(0.,-50.)                      ! 450,850
        CALL RDOT(-10.,-20.,-1,-5)
        CALL SUBP(23)                           !SUBP: CL CB #9 ST-4
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(123)                          !SUBP: OP CB #9 ST-4
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(450.,830.,-1,-4)
        CALL VECT(0.,-100.)
        CALL APNT(530.,830.,1,-5)
        CALL SUBP(702)                          !SUBP: ST-4 SW

        CALL TEXT('SUBSTATION-4')
        CALL ESUB
C
C       DRAW THE DEPICTION OF THE VOLTAGE REGULATOR.
C       ALSO INCLUDE A SET OF ARROWS THAT WILL BE USED TO
C       CHANGE THE OUTPUT VOLTAGE OF THE REGULATOR.
C
        CALL APNT(421.,700.,-1,-4)
        CALL CIRCLE(30.,-1,4,-1)
        CALL APNT(443.,690.,-1,-5)
        CALL TEXT('R')
        CALL APNT(550.,710.,1,-4)
        CALL SUBP(703)                          !SUBP:  RAISE REG VOLT
        CALL VECT(0.,30.)
        CALL ARROWU(1,4,-1)
        CALL ESUB
        CALL APNT(550.,690.,1,-4)
        CALL SUBP(704)                          !SUBP: LOWR REG VOLT
        CALL VECT(0.,-30.,1,4)
        CALL ARROWD(1,4,-1)
        CALL ESUB
```

```
C
C       CONTINUE WITH THE FEEDER, TO THE TRANSFORMER.
C
        CALL APNT(450.,400.,-1,-4)
        CALL VECT(0.,60.)
        CALL TRANSF
        CALL VECT(0.,170.)
C
C       DRAW THE REPRESENTATION OF THE CAPACITOR.
C
        CALL APNT (450.,600.,-1,-4)
        CALL VECT(100.,0.)
        CALL VECT(0.,-30.)
        CALL SUBP(158)                      !SUBP: CL SW CAP
        CALL SWCLV(4)
        CALL ESUB
        CALL OFF(158)
        CALL SUBP(58)
        CALL SWOPV(4)
        CALL ESUB
        CALL APNT(550.,550.,-1,-4)
        CALL VECT(0.,-50.)
        CALL CAPCTR
        CALL VECT(0.,-30.)
        CALL GND
C
C       DRAW THE LIGHTNING PROTECTOR.
C
        CALL APNT(450.,550.,-1,-4)
        CALL VECT(-100.,0.)
        CALL VECT(0.,-50.)
        CALL RDOT(0.,-1.,-1,6)
        CALL RDOT(0.,-18.,-1,6)
        CALL RDOT(0.,-1.,-1,-4)
        CALL VECT(0.,-20.)
        CALL GND
C
C       DRAW THE LOADS, SWITCHES, AND TEXT.
C
        CALL APNT(800.,400.,-1,-4)
        CALL VECT(-700.,0.)
        CALL VECT(0.,-150.)
        CALL SUBP(153,158)                  !SUBP: CL SW LTG LD
        CALL OFF(153)
        CALL SUBP(53,58)                    !SUBP: OP SW LTG LD
        CALL APNT(100.,230.,-1,-4)
        CALL VECT(0.,-30.)
        CALL SUBP(720)                      !SUPB: LTG LD
        CALL LOAD
        CALL ESUB
C
C
        CALL APNT(275.,400.,-1,-4)
        CALL VECT(0.,-150.)
        CALL SUBP(154,158)                  !SUBP: CL SW HEAT LD
        CALL OFF(154)
        CALL SUBP(54,58)                    !SUBP: OP SW HEAT LD
        CALL APNT(275.,230.,-1,-4)
        CALL VECT(0.,-30.)
        CALL SUBP(721,720)                  !SUPB: LTG LD
C
C
        CALL APNT(537.,400.,-1,-4)
        CALL VECT(0.,-100.)
        CALL APNT(625.,300.,-1,-4)
        CALL VECT(-175.,0.)
```

```
      CALL VECT(0.,-50.)
      CALL SUBP(155,158)                           !SUBP: CL SW PWR #1 LD
      CALL OFF(155)
      CALL SUBP(55,58)                             !SUBP: OP SW PWR #1 LD
      CALL APNT(450.,230.,-1,-4)
      CALL VECT(0.,-30.)
      CALL SUBP(722,720)                           !SUPB: PWR #1 LD
C
C
      CALL APNT(625.,300.,-1,-4)
      CALL VECT(0.,-50.)
      CALL SUBP(156,158)                           !SUBP: CL SW PWR #2 LD
      CALL OFF(156)
      CALL SUBP(56,58)                             !SUBP: OP SW PWR #2 LD
      CALL APNT(625.,230.,-1,-4)
      CALL VECT(0.,-30.)
      CALL SUBP(723,720)                           !SUPB: PWR #2 LD
C
C
      CALL APNT(800.,400.,-1,-4)
      CALL VECT(0.,-150.)
      CALL SUBP(157,158)                           !SUBP: CL SW LTG LD
      CALL OFF(157)
      CALL SUBP(57,58)                             !SUBP: OP SW STG LD
      CALL APNT(800.,230.,-1,-4)
      CALL VECT(0.,-30.)
      CALL SUBP(724,720)                           !SUPB: REACT LD
C
C     WRITE TEXT ON LOADS.
C
      CALL STAT(-1)
      CALL APNT(40.,80.,-1,-5)
      CALL TEXT('LIGHTING')
      CALL APNT(242.,80.,-1,-5)
      CALL TEXT('HEAT')
      CALL APNT(502.,80.,-1,-5)
      CALL TEXT('POWER')
      CALL APNT(740.,80.,-1,-5)
      CALL TEXT('REACTIVE')
      CALL APNT(50.,560.,-1,-5)
      CALL TEXT('INDUSTRIAL DISTRIBUTION')
      CALL STAT(1)
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
      CALL SUBP(90)                                !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                                !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
C     DRAW THE 'NO' SUBPICTURE.
C
      CALL SUBP(92)
      CALL APNT(950.,700.,1,-5,-1)
      CALL TEXT('NO')
      CALL ESUB
C
      CALL ESUB
C
      CALL OFF(90)
```

```
C
C        PLACE THE DATA TO BE MONITORED.
C
         CALL APNT(200.,870.,-1,-5)
         CALL NMBR(950,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' VOLTS')
         CALL STAT(1)
         CALL APNT(200.,700.,-1,-5)
         CALL NMBR(951,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' VOLTS')
         CALL STAT(1)
         CALL APNT(200.,650.,-1,-5)
         CALL NMBR(952,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' PF')
         CALL STAT(1)
         CALL APNT(200.,600.,-1,-5)
         CALL NMBR(953,VOLTS,5,'(F5.2)')
         CALL STAT(-1)
         CALL TEXT(' AMPS')
C
C        WRITE A MENU OF OTHER PICTURES TO BE USED.
C
         CALL STAT(1)
         CALL MENU(930.,500.,-50.,705,'STAT 1','GEN 1','STAT11','MENU')
C
C        SAVE THE PICTURE IN A FILE CALLED 'INDLD.DPY'.
C
         CALL SAVE('INDLD.DPY')
C
         STOP
         END
```

THE FOLLOWING IS THE CONTENTS OF THE INDLD.CMD FILE:

```
INDLD/-CP,TI:/SH=INDLD,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

THE 'STAT11.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE STATION 11 DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR STAT11=STAT11
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR STAT11=STAT11/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @STAT11
```

THIS CAUSES A FILE, 'STAT11.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'STAT11.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'STAT11.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN STAT11
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'STAT11.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE STATION 11 DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
801	TOP HV TRANSMISSION TEXT FOR SW
802	BOTTOM HV TRANSMISSION TEXT FOR SW
803	VOLTAGE REGULATOR RAISE ARROW
804	VOLTAGE REGULATOR LOWER ARROW
805	STATION 1 TEXT FOR SW
806	GENERATOR 1 TEXT FOR SW
807	MENU TEXT FOR SW
.	.
.	.
.	.

```
C
C       PROGRAM TO DRAW THE STATION 11 PICTURE.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       START WITH ALL HORIZONTAL LINES BEGINNING AT THE TOP
C       HV TRANSMISSION SWITCH.
C
        CALL APNT(340.,950.,1,-5,-1,1)
        CALL SUBP(801)                          !SUBP: HVTRNS SW
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
        CALL APNT(800.,940.,-1,-4,-1,4)
        CALL VECT(-700.,0.)
        CALL STAT(-1)
        CALL APNT(30.,900.,-1,-5,-1,1)
        CALL TEXT('INTERCONNECTION')
        CALL APNT(305.,900.,-1,-5)
        CALL TEXT('E10-11')
        CALL APNT(505.,900.,-1,-5)
        CALL TEXT('F10-11')
        CALL APNT(668.,840.,-1,-1)
        CALL TEXT('PEAK LOAD &')
        CALL APNT(690.,810.,-1,-1)
        CALL TEXT('EMER GEN')
        CALL APNT(250.,700.,-1,-1)
        CALL VECT(100.,0.)
        CALL RDOT(100.,0.,,-1)
        CALL VECT(100.,0.)
        CALL RDOT(100.,0.,,-1)
        CALL VECT(100.,0.)
        CALL APNT(50.,550.,-1,-1)
        CALL VECT(100.,0.)
        CALL RDOT(0.,-50.,,-4)
        CALL HBUS(600.,5)                               ! BRIGHT BUS
        CALL VECT(0.,-300.,-1,4)
        CALL RDOT(0.,50.,-1,-1)
        CALL VECT(-100.,0.)
        CALL RDOT(0.,150.,-1,-1)
        CALL HBUS(-600.,1)                      ! DIM BUS
        CALL RDOT(0.,-150.,-1,-1)
        CALL VECT(100.,0.)
        CALL APNT(375.,300.,-1,-5)
        CALL TEXT('STATION-11')
        CALL APNT(110.,80.,-1,-5)
        CALL TEXT('C1-11')
        CALL APNT(710.,80.,-1,-5)
        CALL TEXT('D1-11')
        CALL STAT(1)
        CALL APNT(800.,60.,-1,-4,-1,4)
        CALL VECT(-700.,0.)                     ! LOWER DASHED LN
        CALL APNT(340.,30.,1,-5,-1,1)
        CALL SUBP(802)                          !SUBP: HVTRNS SW LWR
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
C
C       DRAW THE VERTICAL LINES STARTING WITH LINE C1-11
C       AND WORKING UPWARD.
C
        CALL APNT(150.,120.,-1,-4,-1,1)
```

```
      CALL ARROWU(-1,4,-1)
      CALL VECT(0.,60.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(14)                          !SUBP: CL CB C1-11
      CALL CBCLD(4)
      CALL ESUB
      CALL SUBP(114)                         !SUBP: OP CB C1-11
      CALL CBOPN(4)
      CALL ESUB
      CALL APNT(150.,200.,-1,-4)
      CALL VECT(0.,390.)
      CALL TRANSF
      CALL VECT(0.,60.)
      CALL APNT(125.,715.,-1,-4)
      CALL CIRCLE (25.,-1,4,-1)
      CALL APNT(143.,705.,,-5)
      CALL TEXT('R')
      CALL APNT(210.,730.,1,-4)              ! PR FOR ARROWS
      CALL SUBP(803)                         !SUBP: RAISE REG VOLT
      CALL VECT(0.,30.)
      CALL ARROWU(1,4,-1)
      CALL ESUB
      CALL APNT(210.,710.,1,-4)
      CALL SUBP(804)                         !SUBP: LWR REG VOLT
      CALL VECT(0.,-30.)
      CALL ARROWD(1,4,-1)
      CALL ESUB
      CALL APNT(150.,740.,-1,-4)
      CALL VECT(0.,60.)
      CALL RDOT(-10.,0.-1,-4)
      CALL SUBP(13,14)                       !SUBP: CL CB INT CON
      CALL SUBP(113,114)                     !SUBP: OP CB INT CON
      CALL APNT(150.,820.,-1,-4)
      CALL VECT(0.,60.)
      CALL ARROWD(-1,4,-1)
C
C     CONTINUE WITH E10-11, DOWNWARD.
C
      CALL APNT(350.,880.,-1,-4)
      CALL ARROWD(-1,4,-1)
      CALL VECT(0.,-60.)
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(17,14)                       !SUBP: CL CB E10-11
      CALL SUBP(117,114)                     !SUBP: OP CB E10-11
      CALL APNT(350.,800.,-1,-4)
      CALL VECT(0.,-300.)
C
C
      CALL APNT(550.,500.,-1,-4)
      CALL VECT(0.,300.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(16,14)                       !SUBP: CL CB F10-11
      CALL SUBP(116,114)                     !SUBP: OP CB F10-11
      CALL APNT(550.,820.,-1,-4)
      CALL VECT(0.,60.)
      CALL ARROWD(-1,4,-1)
C
C     FINISH LINE D1-11.
C
      CALL APNT(740.,180.,-1,-4)
      CALL SUBP(15,14)                       !SUBP: CL CB D1-11
      CALL SUBP(115,114)                     !SUBP: OP CB D1-11
      CALL APNT(750.,180.,-1,-4)
      CALL VECT(0.,-60.)
      CALL ARROWU(-1,4,-1)
C
```

```
C        CONTINUE WITH THE MANUAL, DIM COMPONENTS.
C        START WITH THE PEAK LOAD AND EMERGENCY GENERATOR AND
C        WORK TO THE LEFT.
C
         CALL APNT(726.,765.,-1,-4)
         CALL CIRCLE(25.,-1,1,-1)
         CALL APNT(743.,755.,-1,-1)
         CALL TEXT('4')
         CALL APNT(750.,740.,-1,-1)
         CALL VECT(0.,-240.)
         CALL APNT(650.,700.,-1,-1)
         CALL VECT(0.,-80.)
         CALL SUBP(165)                            !SUBP: CL SW EMER GEN
         CALL SWCLV(1)
         CALL ESUB
         CALL OFF(165)
         CALL SUBP(65)                             !SUBP: OP SW EMER GEN
         CALL SWOPV(1)
         CALL ESUB
         CALL APNT(650.,600.,-1,-1)
         CALL VECT(0.,-260.)
         CALL SUBP(166,165)                        !SUBP: CL SW D1-11
         CALL OFF(166)
         CALL SUBP(66,65)                          !SUBP: OP SW D1-11
         CALL APNT(650.,320.,-1,-1)
         CALL VECT(0.,-70.)
C
C        CONTINUE WITH MANUAL CONNECTION TO F10-11.
C
         CALL APNT(450.,700.,-1,-1)
         CALL VECT(0.,-80.)
         CALL SUBP(167,165)                        !SUBP: CL SW F10-11
         CALL OFF(167)
         CALL SUBP(67,65)                          !SUBP: OP SW F10-11
         CALL APNT(450.,600.,-1,-1)
         CALL VECT(0.,-200.)
C
C        CONTINUE WITH MANUAL CONNECTION TO E10-11.
C
         CALL APNT(250.,700.,-1,-1)
         CALL VECT(0.,-80.)
         CALL SUBP(168,165)                        !SUBP: CL SW E10-11
         CALL OFF(168)
         CALL SUBP(68,65)                          !SUBP: OP SW E10-11
         CALL APNT(250.,600.,-1,-1)
         CALL VECT(0.,-200.)
C
C        CONTINUE WITH MANUAL CONNECTION TO C1-11 AND INTERCONNECTION.
C
         CALL APNT(50.,550.,-1,-1)
         CALL VECT(0.,-70.)
         CALL SUBP(169,165)                        !SUBP: CL SW E10-11
         CALL OFF(169)
         CALL SUBP(69,65)                          !SUBP: OP SW E10-11
         CALL APNT(50.,460.,-1,-1)
         CALL VECT(0.,-120.)
         CALL SUBP(170,165)                        !SUBP: CL SW C1-11
         CALL OFF(170)
         CALL SUBP(70,65)                          !SUBP: OP SW C1-11
         CALL APNT(50.,320.,-1,-1)
         CALL VECT(0.,-70.)
C
C        WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C        FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
C
         CALL SUBP(90)                             !SUBP: R U SURE
```

AD-A061 029   AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO   F/G 10/2
DIGITAL COMPUTER, INTERACTIVE GRAPHICS CONTROL OF AN ELECTRICAL--ETC(U)
1978   L R DAVIS
UNCLASSIFIED   AFIT-CI-78-115   NL

2 OF 2
AD
A061029

END
DATE
FILMED
1 -79
DDC

```
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                           !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
C     DRAW THE 'NO' SUBPICTURE.
C
      CALL SUBP(92)
      CALL APNT(950.,700.,1,-5,-1)
      CALL TEXT('NO')
      CALL ESUB
C
      CALL ESUB
C
      CALL OFF(90)
C
C     PLACE THE DATA TO BE MONITORED.
C
      CALL APNT(0.,725.,-1,-5)
      CALL NMBR(946,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' V')
      CALL STAT(1)
      CALL APNT(0.,675.,-1,-5)
      CALL NMBR(947,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' HZ')
      CALL STAT(1)
      CALL APNT(0.,625.,-1,-5)
      CALL NMBR(948,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' A')
      CALL STAT(1)
      CALL APNT(265.,460.,-1,-5)
      CALL NMBR(965,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL STAT(1)
C
C     ADD THE MENU OF OTHER PICTURES.
C
      CALL MENU(930.,500.,-50.,805,'STAT 1','GEN 1','MENU')
C
C     SAVE THE PICTURE ON A FILE CALLED 'STAT11.DPY'.
C
      CALL SAVE('STAT11.DPY')
C
      STOP
      END
```

THE FOLLOWING IS THE CONTENTS OF FILE, STAT11.CMD:

```
STAT11/-CP,TI:/SH=STAT11,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

THE 'GEN1.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE GENERATOR 1 DISPLAY. EDIT THE FILE USING THE EDIT UTILITY. IT USES SUBROUTINES CONTAINED IN THE FILE, 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
>FOR GEN1=GEN1
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
>FOR GEN1=GEN1/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @GEN1
```

THIS CAUSES A FILE, 'GEN1.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'GEN1.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'GEN1.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN GEN1
```

IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED 'GEN1.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE GENERATOR 1 DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBERS	DESCRIPTION
1001	HV TRANSMISSION TEXT FOR SW
1002	SYNCRN TEXT FOR SW
1003	STATION 1 TEXT FOR SW
1004	STATION 11 TEXT FOR SW
1005	MENU TEXT FOR SW
1006	GENERATOR 1 VOLTAGE RAISE ARROW
1007	GENERATOR 1 VOLTAGE LOWER ARROW
1008	GENERATOR 1 FREQUENCY RAISE ARROW
1009	GENERATOR 1 FREQUENCY RAISE ARROW
.	.
.	.
.	.

```
C
C       PROGRAM TO DRAW THE GENERATOR 1 STATION.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       START AT THE SECTION OF THE RING BUS.
C
        CALL APNT(300.,1020.,-1,-4)
        CALL VBUS(-10.,5)
        CALL RDOT(0.,-10.,-1,-5)
        CALL VBUS(-10.,5)
        CALL RDOT(0.,-10.,-1,-4)
        CALL VBUS(-30.,5)
        CALL RDOT(-1.,-1.,-1,-4)
        CALL HBUS(50.,5)
        CALL RDOT(10.,0.,-1,0-4)
        CALL HBUS(10.,5)
        CALL RDOT(10.,0.,-1,-4)
        CALL HBUS(10.,5)
        CALL STAT(-1)
        CALL APNT(350.,980.,-1,-5)
        CALL TEXT('RING BUS')
        CALL STAT(1)
C
C       DRAW THE HV TRANSMISSION SWITCH WITH DASHED LINE.
C
        CALL APNT(500.,910.,1,-5)
        CALL SUBP(1001)                         !SUBP: HVTRNS SW
        CALL TEXT('HV TRANSMISSION')
        CALL ESUB
        CALL APNT(800.,900.,-1,-4,-1,4)
        CALL VECT(-700.,0.)
C
C       DRAW THE CONNECTION LINE G1-3 AND GENERATOR 1.
C
        CALL APNT(266.,365.,-1,-4,-1,1)
        CALL CIRCLE(35.,-1,4,-1)
        CALL APNT(293.,355.,-1,-5)
        CALL TEXT('1')
        CALL APNT(300.,400.,-1,-4)
        CALL VECT(0.,80.)
        CALL TRANSF
        CALL VECT(0.,60.)
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(6)                            !SUBP: CL CB GEN 1
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(106)                          !SUBP: OP CB GEN 1
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(300.,600.,-1,-4)
        CALL VECT(0.,348.)
C
C       WRITE ALL THE TEXT FOR GENERATOR ONE.
C
        CALL APNT(200.,290.,-1,-5)
        CALL STAT(-1)
        CALL TEXT('PRIME MOVER')
        CALL APNT(200.,260.,-1,-5)
        CALL TEXT('EXCITATION')
```

```
      CALL APNT(200.,220.,-1,-5)
      CALL TEXT('VOLTAGE       AMPS')
      CALL APNT(290.,180.,1,-4)
      CALL SUBP(1006)                          !SUBP: RAISE VOLT GEN 1
      CALL VECT(0.,30.)
      CALL ARROWU(1,4,-1)
      CALL ESUB
      CALL APNT(290.,170.,1,-4)
      CALL SUBP(1007)                          !SUBP: LWR VOLT GEN 1
      CALL VECT(0.,-30.)
      CALL ARROWD(1,4,-1)
      CALL ESUB
      CALL APNT(375.,150.,-1,-5)
      CALL TEXT('P.F.')
      CALL APNT(215.,100.,-1,-5)
      CALL TEXT('FREQ')
      CALL APNT(290.,60.,1,-4)
      CALL SUBP(1008,1006)                     !SUBP: RAISE FREQ GEN 1
      CALL APNT(290.,50.,1,-5)
      CALL SUBP(1009,1007)                     !SUBP: LWR FREQ GEN 1
      CALL APNT(385.,80.,-1,-5)
      CALL TEXT('KW')
      CALL STAT(1)
C
C     DRAW THE PRIME MOVER AND EXCITATION SWITCHES FOR GENERATOR 1.
C
      CALL APNT(390.,290.,1,-5)
      CALL SUBP(151)                           !SUBP: GEN 1 DC SUP ON
      CALL TEXT('ON')
      CALL ESUB
      CALL OFF(151)
      CALL SUBP(51)                            !SUBP: GEN 1 DC SUP OFF
      CALL TEXT('OFF')
      CALL ESUB
      CALL APNT(390.,260.,1,-5)
      CALL SUBP(152,151)                            !SUBP: GEN 1 EXCI
      CALL OFF(152)
      CALL SUBP(52,51)                         !SUBP: GEN 3 EXCIT OFF
C
C     PLACE THE DATA TO BE MONITORED IN CORRECT POSITION.
C
      CALL STAT(1)
      CALL APNT(200.,165.,-1,-5)
      CALL NMBR(931,VOLTS,5,'(F5.2)')
      CALL APNT(200.,45.,-1,-5)
      CALL NMBR(932,VOLTS,5,'(F5.2)')
      CALL APNT(360.,190.,-1,-5)
      CALL NMBR(933,VOLTS,5,'(F5.2)')
      CALL APNT(360.,120.,-1,-5)
      CALL NMBR(934,VOLTS,5,'(F5.2)')
      CALL APNT(360.,50.,-1,-5)
      CALL NMBR(935,VOLTS,5,'(F5.2)')
C
C     WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C     FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
      CALL SUBP(90)                            !SUBP: R U SURE
      CALL APNT(870.,750.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE 'YES' SUBPICTURE.
C
      CALL SUBP(91)                            !SUBP: CK IF YES
      CALL APNT(880.,700.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
C
```

```
C         DRAW THE 'NO' SUBPICTURE.
C
          CALL SUBP(92)
          CALL APNT(950.,700.,1,-5,-1)
          CALL TEXT('NO')
          CALL ESUB
C
          CALL ESUB
C
          CALL OFF(90)
C
C         WRITE DATA ON THE RIGHT SIDE OF PICTURE.
C
          CALL STAT(-1)
          CALL APNT(600.,700.,-1,-5)
          CALL TEXT('*** DATA ***')
          CALL APNT(600.,660.,-1,-5)
          CALL TEXT('GEN 2')
          CALL APNT(650.,620.,-1,-5)
          CALL STAT(1)
          CALL NMBR(936,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  VOLTS')
          CALL APNT(650.,580.,-1,-5)
          CALL STAT(1)
          CALL NMBR(937,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  HZ')
          CALL APNT(650.,540.,-1,-5)
          CALL STAT(1)
          CALL NMBR(940,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  KW')
C
C         WRITE THE DATA FOR GEN 3.
C
          CALL APNT(600.,460.,-1,-5)
C
C
C
          CALL TEXT('GEN 3')
          CALL APNT(650.,420.,-1,-5)
          CALL STAT(1)
          CALL NMBR(941,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  VOLTS')
          CALL APNT(650.,380.,-1,-5)
          CALL STAT(1)
          CALL NMBR(942,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  HZ')
          CALL APNT(650.,340.,-1,-5)
          CALL STAT(1)
          CALL NMBR(945,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  KW')
C
C         WRITE THE DATA FOR INTERCONNECTION.
C
          CALL APNT(600.,260.,-1,-5)
          CALL TEXT('INT CON')
          CALL APNT(650.,220.,-1,-5)
          CALL STAT(1)
          CALL NMBR(946,VOLTS,5,'(F5.2)')
          CALL STAT(-1)
          CALL TEXT('  VOLTS')
```

```
      CALL APNT(650.,180.,-1,-5)
      CALL STAT(1)
      CALL NMBR(947,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT('  HZ')
      CALL APNT(200.,355.,-1,-5)
      CALL TEXT('GEN')
      CALL STAT(1)
      CALL APNT(350.,850.,-1,-5)
      CALL NMBR(967,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL STAT(1)
C
C     ADD THE MENU FOR SWITCHES FOR OTHER PICTURES.
C
      CALL MENU(930.,500.,-50.,1002,'SYNCRN','STAT 1','STAT11','MENU')
C
C     SAVE THE PICTURE IN A FILE CALLED 'GEN1.DPY'.
C
      CALL SAVE('GEN1.DPY')
C
      STOP
      END
```

THE FOLLOWING IS CONTAINED IN THE FILE, GEN1.CMD:

```
GEN1/-CP,TI:/SH=GEN1,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

THE 'SYNCRN.FTN' AND 'SYREST.FTN' FILES CONTAIN THE FORTRAN SOURCE FOR THE SYNCHRONIZING DISPLAY. EDIT THE FILES USING THE EDIT UTILITY. THESE FILES USE SUBROUTINES CONTAINED IN A FILE CALLED 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
                >FOR SYNCRN=SYNCRN
AND THEN                >FOR SYREST=SYREST
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
                >FOR SYNCRN=SYNCRN/LI:1
AND THEN        >FOR SYREST=SYREST/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
                >TKB @SYNCRN
```

THIS CAUSES FILES, 'SYNCRN.OBJ' AND 'SYREST.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'SYNCRN.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'SYNCRN.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11 AND BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
                >RUN SYNCRN
```

IF THE ABOVE SEQUENCE IS FOLLOWED, A NEW FILE IS CREATED CALLED 'SYNCRN.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE SYNCHRONIZING DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBER	DESCRIPTION
1101	GEN 1 VOLTAGE RAISE ARROW
1102	GEN 1 VOLTAGE LOWER ARROW
1103	GEN 1 FREQUENCY RAISE ARROW
1104	GEN 1 FREQUENCY LOWER ARROW
1105	GEN 2 VOLTAGE RAISE ARROW
1106	GEN 2 VOLTAGE LOWER ARROW
1107	GEN 2 FREQUENCY RAISE ARROW
1108	GEN 2 FREQUENCY LOWER ARROW
1109	GEN 3 VOLTAGE RAISE ARROW
1110	GEN 3 VOLTAGE LOWER ARROW
1111	GEN 3 FREQUENCY RAISE ARROW
1112	GEN 3 FREQUENCY LOWER ARROW
1113	INT CON VOLTAGE RAISE ARROW
1114	INT CON VOLTAGE LOWER ARROW
1115	HV TRANSMISSION TEXT FOR SW
1116	STATION 1 TEXT FOR SW
1117	GEN 1 TEXT FOR SW
1118	ST 11 TEXT FOR SW
1119	MENU TEXT FOR SW
.	.
.	.
.	.
1120	VERTICAL BUS DASH 10 UNITS
1121	VERTICAL BUS DASH 10 UNITS
1122	VERTICAL BUS ON LEFT
1123	HORIZ. BUS 100 LEFT
1124	HORIZ. BUS 100 RIGHT
1125	VERTICAL BUS ON RIGHT
1126	GEN AND TRANSF 1
1127	GEN AND TRANSF 2
1128	GEN AND TRANSF 3
1129	ST 11 BUS
1130	GEN 1 TRANSF ONLY
1131	INT CON TRANSF
1135	CIRCLE ONLY
1136	INT CON CIRCLE

THE FOLLOWING IS THE CONTENTS OF THE FILE, SYNCRN.CMD:

```
SYNCRN/-CP,TI:/SH=SYNCRN,SYREST,DSPSUB,GLIB/LB
/
ASG=GR0:1
//
```

```
C
C       PROGRAM TO DRAW THE SYNCHRONIZING PICTURE.
C
        COMMON/DFILE/IBUF(1050)
        CALL INIT(1050)
C
C       START WITH THE RING BUS SECTION.
        CALL APNT(150.,1020.,-1,-4)
        CALL SUBP(1122)                             !SUBP: VBUS SECTION
        CALL SUBP(1120)                             !SUBP: VBUS 10
        CALL VBUS(-10.,5)
        CALL ESUB
        CALL RDOT(0.,-10.,-1,-4)
        CALL SUBP(1121,1120)                        !SUBP: VBUS 10
        CALL RDOT(0.,-10.,-1,-4)
        CALL VBUS(-30.,5)
        CALL ESUB
        CALL SUBP(1123)                             !SUBP: HBUS 100
        CALL HBUS(100.,5)
        CALL ESUB
        CALL RDOT(0.,-10.,-5,-5)
        CALL SUBP(4)                                !SUBP: CL CB RNG BUS LT
        CALL CBCLD(4)
        CALL ESUB
        CALL SUBP(104)                              !SUBP: OP CB RNG BUS LT
        CALL CBOPN(4)
        CALL ESUB
        CALL APNT(270.,950.,-1,-4)
        CALL HBUS(160.,5)
        CALL RDOT(0.,-10.,-1,-4)
        CALL SUBP(5,4)                              !SUBP: CL CB RNG BUS RT
        CALL SUBP(105,104)                          !SUBP: OP CB RNG BUS RT
        CALL APNT(450.,950.,-1,-5)
        CALL SUBP(1124,1123)                        !SUBP: HBUS ON RT
        CALL RDOT(0.,70.,-1,-4)
        CALL SUBP(1125,1122)                        !SUBP: VBUS SECT ON RT
C
C       WRITE THE DESIGNATION FOR THE GENERATORS AND REGULATOR.
C
        CALL APNT(143.,405.,-1,-5)
        CALL TEXT('1')
        CALL APNT(343.,405.,-1,-5)
        CALL TEXT('2')
        CALL APNT(543.,405.,-1,-5)
        CALL TEXT('3')
        CALL APNT(743.,805.,-1,-5)
        CALL TEXT('R')
C
C       TEXT FOR RING BUS.
C       ALSO FOR THE GENERATORS.
C
        CALL STAT(-1)
        CALL APNT(290.,980.,-1,-5)
        CALL TEXT('RING BUS')
        CALL APNT(70.,405.,-1,-5)
        CALL TEXT('GEN')
        CALL APNT(270.,405.,-1,-5)
        CALL TEXT('GEN')
        CALL APNT(470.,405.,-1,-5)
        CALL TEXT('GEN')
C
```

```
C       DRAW GEN 1, TRANSFORMER, AND CIRCUIT BREAKER.
C
        CALL APNT(126.,415.,-1,-4)
        CALL SUBP(1126)                          !SUBP: GEN1 & TRANS
        CALL SUBP(1135)                          !SUBP: CIRCLE ONLY
        CALL CIRCLE(25.,-1,4,-1)
        CALL ESUB
        CALL RDOT(24.,25.,-1,-4)
        CALL SUBP(1130)                          !SUBP: TRANS ONLY
        CALL VECT(0.,50.)
        CALL TRANSF                              ! GEN 1 TRNS
        CALL VECT(0.,50.)
        CALL ESUB
        CALL ESUB
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(6,4)                           !SUBP: CL CB GEN 1
        CALL SUBP(106,104)                       !SUBP: OP CB GEN 1
        CALL APNT(150.,600.,-1,-4)
        CALL VECT(0.,350.)
C
C       DRAW GEN 2, TRANSFORMER, AND CIRCUIT BREAKERS UP TO THE RING BUS.
C
        CALL APNT(325.,415.,-1,-4)
        CALL SUBP(1127,1126)                     !SUBP: GEN 2 & TRNS
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(7,4)                           !SUBP: CL CB GEN 2
        CALL SUBP(107,104)                       !SUBP: OP CB GEN 2
        CALL APNT(350.,600.,-1,-4)
        CALL VECT(0.,100.)
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(9,4)                           !SUBP: CL CB A1-3
        CALL SUBP(109,104)                       !SUBP: OP CB A1-3
        CALL APNT(350.,720.,-1,-4)
        CALL VECT(0.,230.)
C
C       DRAW GEN 3, TRANSFORMER, AND CIRCUIT BREAKERS TO RING BUS.
C
        CALL APNT(525.,415.,-1,-4)
        CALL SUBP(1128,1126)                     !SUBP: GEN 3 & TRNS
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(8,4)                           !SUBP: CL CB GEN 3
        CALL SUBP(108,104)                       !SUBP: OP CB GEN 3
        CALL APNT(550.,600.,-1,-4)
        CALL VECT(0.,100.)
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(10,4)                          !SUBP: CL CB B1-3
        CALL SUBP(110,104)                       !SUBP: OP CB B1-3
        CALL APNT(550.,720.,-1,-4)
        CALL VECT(0.,230.)
C
C       DRAW STATION 11 BUS THEN THE LINES C1-11 AND D1-11.
C
        CALL APNT(350.,650.,-1,-4)
        CALL HBUS(200.,5)
        CALL APNT(416.,650.,-1,-4)               !START C1-11
        CALL VECT(0.,50.)
        CALL RDOT(-10.,0.,-1,-4)
        CALL SUBP(11,4)                          !SUBP: CL CB C1-11 ST-1
        CALL SUBP(111,104)                       !SUBP: CL CB C1-11 ST-1
        CALL APNT(416.,720.,-1,-4)
        CALL VECT(0.,80.)                        ! 416,800
        CALL VECT(234.,0.)                       ! 650,800
        CALL VECT(0.,-250.)                      ! 650,550
        CALL VECT(50.,0.)                        ! 700,550
        CALL VECT(0.,30.)
        CALL RDOT(-10.,0.,-1,-4)
```

```
      CALL SUBP(14,4)                              !SUBP: CL CB C1-11 ST11
      CALL SUBP(114,104)                           !SUBP: OP CB C1-11 ST11
      CALL APNT(700.,600.,-1,-4)
      CALL VECT(0.,50.)
      CALL SUBP(1129,1123)                         !SUBP: ST-11 BUS
      CALL VECT(0.,-50.,-1,4)                      ! START D1-11
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(15,4)                              !SUBP: CL CB D1-11 ST11
      CALL SUBP(115,104)                           !SUBP: OP CB D1-11 ST11
      CALL APNT(800.,580.,-1,-4)
      CALL VECT(0.,-80.)                           ! 800,500
      CALL VECT(-200.,0.)                          ! 600,500
      CALL VECT(0.,250.)                           ! 600,750
      CALL VECT(-116.,0.)                          ! 484,750
      CALL VECT(0.,-30.)                           ! 484,720
      CALL RDOT(-10.,-20.,-1,-4)
      CALL SUBP(12,4)                              !SUBP: CL CB D1-11 ST-1
      CALL SUBP(112,104)                           !SUBP: OP CB D1-11 ST-1
      CALL APNT(484.,700.,-1,-4)
      CALL VECT(0.,-50.)
C
C     DRAW INTERCONNECTION.
C
      CALL APNT(750.,650.,-1,-4)
      CALL SUBP(1131,1130)                         !SUBP: INT CON TRNS
      CALL APNT(726.,815.,-1,-4)
      CALL SUBP(1136,1135)                         !SUBP: IN CON CIR
      CALL APNT(750.,840.,-1,-4)
      CALL VECT(0.,50.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(13,4)                              !SUBP: CL CB INT CON
      CALL SUBP(113,104)                           !SUBP: OP CB INT CON
      CALL APNT(750.,910.,-1,-4)
      CALL VECT(0.,70.)
      CALL ARROWD(-1,4,-1)
      CALL APNT(700.,1000.,-1,-5)
      CALL TEXT('INT CON')
      CALL APNT(650.,400.,-1,-5)
      CALL TEXT('SYNCHRONIZING')
C
C     CALL THE REST OF THE PICTURE.
C
      CALL RESTOF
C
      STOP
      END
```

```
C
C       SUBROUTINE TO FINISH THE SYNCHRONIZING PICTURE.
C
        SUBROUTINE RESTOF
C
C       DISPLAY THE DATA FOR THE VARIOUS SOURCES.
C
        CALL APNT(115.,350.,-1,-5,-1,1)
        CALL TEXT('VOLT')
        CALL APNT(180.,300.,1,-4)
        CALL SUBP(1101)                         !SUBP: RAISE GEN1 VOLT
        CALL VECT(0.,30.)
        CALL ARROWU(1,4,-1)
        CALL ESUB
        CALL RDOT(0.,-40.,-1,-4)
        CALL SUBP(1102)                         !SUBP: LWR GEN 1 VOLT
        CALL VECT(0.,-30.)
        CALL ARROWD(1,4,-1)
        CALL ESUB
        CALL APNT(120.,220.,-1,-5)
        CALL TEXT('FREQ')
        CALL APNT(180.,170.,1,-4)
        CALL SUBP(1103,1101)                    !SUBP: RAISE GEN1 FREQ
        CALL RDOT(0.,-40.,1,-4)
        CALL SUBP(1104,1102)                    !SUBP: LWR GEN1 FREQ
C
        CALL APNT(315.,350.,-1,-5)
        CALL TEXT('VOLT')
        CALL APNT(380.,300.,1,-4)
        CALL SUBP(1105,1101)                    !SUBP: RAISE GEN 2 VOLT
        CALL RDOT(0.,-40.,-1,-4)
        CALL SUBP(1106,1102)                    !SUBP: LWR GEN 2 VOLT
        CALL APNT(320.,220.,-1,-5)
        CALL TEXT('FREQ')
        CALL APNT(380.,170.,1,-4)
        CALL SUBP(1107,1101)                    !SUBP: RAISE GEN 2 FREQ
        CALL RDOT(0.,-40.,1,-4)
        CALL SUBP(1108,1102)                    !SUBP: LWR GEN2 FREQ
C
        CALL APNT(515.,350.,-1,-5)
        CALL TEXT('VOLT')
        CALL APNT(580.,300.,1,-4)
        CALL SUBP(1109,1101)                    !SUBP: RAISE GEN 3 VOLT
        CALL RDOT(0.,-40.,1,-4)
        CALL SUBP(1110,1102)                    !SUBP: LWR GEN 3 VOLT
        CALL APNT(520.,220.,-1,-5)
        CALL TEXT('FREQ')
        CALL APNT(580.,170.,1,-4)
        CALL SUBP(1111,1101)                    !SUBP: RAISE GEN3 FREQ
        CALL RDOT(0.,-40.,1,-4)
        CALL SUBP(1112,1102)                    !SUBP: LWR GEN3 FREQ
C
        CALL APNT(815.,850.,-1,-5)
        CALL TEXT('VOLT')
        CALL APNT(880.,800.,1,-4)
        CALL SUBP(1113,1101)                    !SUBP: RAISE INT VOLT
        CALL RDOT(0.,-40.,1,-4)
        CALL SUBP(1114,1102)                    !SUBP: LWR INT VOLT
        CALL APNT(820.,720.,-1,-5)
        CALL TEXT('FREQ')
        CALL STAT(1)
C
```

```
C        WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C        FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
         CALL SUBP(90)                                 !SUBP: R U SURE
         CALL APNT(890.,750.,-1,-5,1)
         CALL TEXT('YOU SURE?')
C
C        DRAW THE 'YES' SUBPICTURE.
C
         CALL SUBP(91)                                 !SUBP: CK IF YES
         CALL APNT(900.,700.,1,-5,-1)
         CALL TEXT('YES')
         CALL ESUB
C
C        DRAW THE 'NO' SUBPICTURE.
C
         CALL SUBP(92)
         CALL APNT(970.,700.,1,-5,-1)
         CALL TEXT('NO')
         CALL ESUB
C
         CALL ESUB
C
         CALL OFF(90)
C
C        PLACE THE NUMBERS FOR THE DATA TO BE DISPLAYED.
C
         CALL APNT(100.,285.,-1,-5)
         CALL STAT(1)
         CALL NMBR(931,VOLTS,5,'(F5.2)')
         CALL APNT(100.,155.,-1,-5)
         CALL NMBR(932,VOLTS,5,'(F5.1)')
         CALL APNT(300.,285.,-1,-5)
         CALL NMBR(936,VOLTS,5,'(F5.2)')
         CALL APNT(300.,155.,-1,-5)
         CALL NMBR(937,VOLTS,5,'(F5.1)')
         CALL APNT(500.,285.,-1,-5)
         CALL NMBR(941,VOLTS,5,'(F5.2)')
         CALL APNT(500.,155.,-1,-5)
         CALL NMBR(942,VOLTS,5,'(F5.1)')
         CALL APNT(800.,785.,-1,-5)
         CALL NMBR(946,VOLTS,5,'(F5.2)')
         CALL APNT(800.,690.,-1,-5)
         CALL NMBR(947,VOLTS,5,'(F5.1)')
C
C        SETUP THE MENU.
C
         CALL MENU(930.,500.,-50.,1115,'HVTRNS','STAT 1','GEN 1',
      1 'STAT11','MENU')
C
C        SAVE THE PICTURE IN A FILE CALLED 'SYNCRN.DPY'.
C
         CALL SAVE('SYNCRN.DPY')
C
C
         RETURN
         END
---
```

```
PIP>TI:=INTRO.TXT
        THE 'INTRO.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE
INTRODUCTION DISPLAY.  EDIT THE FILE USING THE EDIT UTILITY.
        TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

                >FOR INTRO=INTRO

        TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

                >FOR INTRO=INTRO/LI:1

        TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

                >TKB @INTRO

        THIS CAUSES A FILE, 'INTRO.OBJ', TO BE CREATED BY THE 'FOR'
COMMAND AND A FILE, 'INTRO.TSK', TO BE CREATED BY THE 'TKB' COMMAND.
THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE.  THIS
IS FILE, 'INTRO.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY
TO BUILD THIS TASK.
        TURN ON THE VT-11, THEN BUILD THE DISPLAY TO BE SAVED BY TYPING:

                >RUN INTRO

        IF THE ABOVE SEQUENCE IS FOLLOWED A NEW FILE IS CREATED CALLED
'INTRO.DPY' THAT CONTAINS THE SAVED DISPLAY.

        THE INTRODUCTION DISPLAY HAS SUBPICTURES OF TEXT FOR
SWITCHING PURPOSES ONLY.  THEY ARE USED FOR CHANGING DISPLAYS.
```

SUBPICTURE NUMBER	DESCRIPTION
1	HVTRNS TEXT FOR SW
2	SYNCRN TEXT
3	STAT1 TEXT
4	STAT11 TEXT
5	GEN 1 TEXT
6	SUBST4 TEXT
7	IND LD TEXT
8	SUBST6 TEXT
9	NETSYS TEXT
10	LD CTR TEXT
11	EXIT TEXT

THE FOLLOWING IS CONTAINED IN THE FILE, INTRO.CMD:

```
INTRO/-CP,TI:/SH=INTRO,DSPSUB,GLIB/LB
/
ASG=GRO:1
//
```

```
C
C       PROGRAM TO WRITE THE INTRODUCTION AND ENTRY MENU FOR THE
C       CONTROL PROGRAM.
C
        COMMON/DFILE/IBUF(1000)
        CALL INIT(1000)
C
C       WRITE THE TEXT FOR EXPLANATION STARTING WITH TITLE.
C
        CALL APNT(250.,850.,-1,-5)
        CALL STAT(-1)
        CALL TEXT('POWER STSTEM CONTROL GRAPHICS')
        CALL STAT(1)
C
C       WRITE TEXT.
C
        CALL APNT(50.,750.,-1,-5)
        CALL TEXT('THE GRAPHICS THAT FOLLOWS THIS EXPLANATION CAN
     1 CONTROL THE',1,'POWER SYSTEM SIMULATOR IN FRONT OF YOU.  THE
     2 COMPUTER IS',1,'PROGRAMMED TO DETECT WHERE AND WHEN YOU POINT
     3 LIGHT PEN ON',1,'SENSITIZED PORTIONS OF THIS SCREEN.  IF IT
     4 DETECTS A 'HIT',',1)
        CALL TEXT('IT WILL THEN EXECUTE SOME ACTION THAT HAS ALSO BEEN
     1 PROGRAMMED',1,'INTO THE COMPUTER.  THE ACTION MAY BE TO CLOSE
     2 A CIRCUIT',1,'BREAKER, RAISE A VOLTAGE, OR CHANGE PICTURES.  TO
     3 AVOID',1,'MISTAKES, A DOUBLE CHECK IS ADDED IN THE FORM OF
     4  A',1)
        CALL TEXT('"YOU SURE" QUESTION.  POINT THE LIGHT PEN TO THE
     1 "YES" OR',1,'"NO", AS DESIRED, FOR THE CHANGE INDICATED BY
     2 A TRIANGLE',1,'THAT APPEARS AROUND THE LIGHT PEN HIT.  THE
     3 CIRCUIT',1,'BREAKERS AND SWITCHES WILL OPEN IF CLOSED AND
     4 VICE VERSA',1)
        CALL TEXT('WHEN POINTED AT.  TO CHANGE PICTURES POINT AT THE
     1 WORD',1,'DESCRIBING THE PICTURE DESIRED.  ONLY THOSE WORDS IN
     2 BLOCK',1,'LETTERS, NOT ITALICS, ARE SENSITIVE.  TO CONTINUE "HIT"
     3 ONE OF',1,'THE WORDS ON THE RIGHT.  THE PROGRAM CAN BE STOPPED
     4 ONLY BY',1,'A HIT ON "EXIT".')
C
C       CREATE THE MENU TO BE USED TO CHOOSE WHICH PICTURE TO START FROM.
C
        CALL MENU(930.,550.,-50.,1,'HVTRNS','SYNCRN','STAT1','STAT11',
     1 'GEN 1','SUBST4','IND LD','SUBST6','NETSYS','LD CTR')
        CALL MENU(930.,50.,50.,11,'EXIT')
C
```

```
C
C         WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C         FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
          CALL SUBP(90)                               !SUBP: R U SURE
          CALL APNT(870.,750.,-1,-5,1)
          CALL TEXT('YOU SURE?')
C
C         DRAW THE 'YES' SUBPICTURE.
C
          CALL SUBP(91)                               !SUBP: CK IF YES
          CALL APNT(880.,700.,1,-5,-1)
          CALL TEXT('YES')
          CALL ESUB
C
C         DRAW THE 'NO' SUBPICTURE.
C
          CALL SUBP(92)
          CALL APNT(950.,700.,1,-5,-1)
          CALL TEXT('NO')
          CALL ESUB
C
          CALL ESUB
C
          CALL OFF(90)
C
C         SAVE THE PICTURE IN A FILE CALLED 'INTRO.DPY'.
C
          CALL SAVE('INTRO.DPY')
          STOP
          END
```

THE 'STAT1.FTN' AND 'ST1FIN.FTN' FILES CONTAIN THE FORTRAN SOURCE FOR THE STATION 1 DISPLAY. EDIT THE FILES USING THE EDIT UTILITY. THESE FILES USE SUBROUTINES CONTAINED IN A FILE CALLED 'DSPSUB.FTN'. IF THIS FILE HAS NOT BEEN PREVIOUSLY COMPILED, IT MUST BE COMPILED TOO.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
          >FOR STAT1=STAT1
AND THEN  >FOR ST1FIN=ST1FIN
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
          >FOR STAT1=STAT1/LI:1
AND THEN  >FOR ST1FIN=ST1FIN/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
>TKB @STAT1
```

THIS CAUSES FILES, 'STAT1.OBJ' AND 'ST1FIN.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'STAT1.TSK', TO BE CREATED BY THE 'TKB' COMMAND. THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'STAT1.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TURN ON THE VT-11 AND BUILD THE DISPLAY TO BE SAVED BY TYPING:

```
>RUN STAT1
```

IF THE ABOVE SEQUENCE IS FOLLOWED, A NEW FILE IS CREATED CALLED 'STAT1.DPY' THAT CONTAINS THE SAVED DISPLAY.

THE FOLLOWING IS A LIST OF THE SUBPICTURE NUMBERS FOR THE STATION 1 DISPLAY. THE FIRST SET CONTAINS THE SUBPICTURE NUMBERS THAT WILL BE LIGHT PEN SENSITIVE. THE ORDER DETERMINES THE ORDER OF THE COMPUTED GO TO STATEMENT'S ARGUMENTS IN THE MAIN PROGRAM. THE REMAINDER ARE NOT SENSITIVE AND ARE FOR COPYING PURPOSES ONLY.

SUBPICTURE NUMBER	DESCRIPTION
901	HV TRANSMISSION TEXT FOR SW
902	GEN 3 RAISE VOLTAGE ARROW
903	GEN 3 LOWER VOLTAGE ARROW
904	GEN 3 RAISE FREQUENCY ARROW
905	GEN 3 LOWER FREQUENCY ARROW
906	GEN 2 RAISE VOLTAGE ARROW
907	GEN 2 LOWER VOLTAGE ARROW
908	GEN 2 RAISE FREQUENCY ARROW
909	GEN 2 LOWER FREQUENCY ARROW
910	SYNCRN TEXT FOR SW
911	GEN 1 TEXT FOR SW
912	STATION 11 TEXT FOR SW
913	MENU TEXT FOR SW
.	.
.	.
.	.
920	GEN 2 AND TRNSF
921	TRANSFORMER
922	GEN TEXT
923	ST SERV LOAD
924	AG SERV LD
925	ST SERV TRNSF
926	GEN 3 AND TRNSF

```
C
C         PROGRAM TO DRAW STATION 1.
C
          COMMON/DFILE/IBUF(1050)
          CALL INIT(1050)
C
C         START WITH THE TOP OF THE PICTURE AND WORK DOWNWARD.
C
          CALL APNT(340.,950.,1,-5,-1,1)
          CALL SUBP(901)                               !SUBP: HVTRNS SW
          CALL TEXT('HV TRANSMISSION')
          CALL ESUB
          CALL APNT(850.,940.,-1,-4,-1,4)
          CALL VECT(-800.,0.)
C
C         WRITE TEXT ON THE OUTGOING LINES.
C
          CALL STAT(-1)
          CALL APNT(120.,900.,-1,-5,-1,1)
          CALL TEXT('A1-3')
          CALL APNT(314.,900.,-1,-5)
          CALL TEXT('C1-11')
          CALL APNT(514.,900.,-1,-5)
          CALL TEXT('D1-11')
          CALL APNT(720.,900.,-1,-5)
          CALL TEXT('B1-3')
C
C         DRAW LINE A1-3.
C
          CALL APNT(150.,600.,-1,-4)
          CALL VECT(0.,200.)
          CALL RDOT(-10.,0.,-1,-4)
          CALL SUBP(9)                                 !SUBP: CL CB A1-3
          CALL CBCLD(4)
          CALL ESUB
          CALL SUBP(109)                               !SUBP: OP CB A1-3
          CALL CBOPN(4)
          CALL ESUB
          CALL APNT(150.,820.,-1,-4)
          CALL VECT(0.,70.)
          CALL ARROWU(-1,4,-1)
C
C         DRAW LINE C1-11.
C
          CALL APNT(350.,700.,-1,-4)
          CALL VECT(0.,100.)
          CALL RDOT(-10.,0.,-1,-4)
          CALL SUBP(11,9)                              !SUBP: CL CB C1-11
          CALL SUBP(111,109)                           !SUBP: OP CB C1-11
          CALL APNT(350.,820.,-1,-4)
          CALL VECT(0.,70.)
          CALL ARROWU(-1,4,-1)
C
C         DRAW LINE D1-11.
C
          CALL APNT(550.,700.,-1,-4)
          CALL VECT(0.,100.)
          CALL RDOT(-10.,0.,-1,-4)
          CALL SUBP(12,9)                              !SUBP: CL CB D1-11
          CALL SUBP(112,109)                           !SUBP: OP CB D1-11
          CALL APNT(550.,820.,-1,-4)
          CALL VECT(0.,70.)
```

```
      CALL ARROWU(-1,4,-1)
C
C     DRAW LINE B1-3.
C
      CALL APNT(750.,600.,-1,-4)
      CALL VECT(0.,200.)
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(10,9)                          !SUBP: CL CB B1-3
      CALL SUBP(110,109)                       !SUBP: OP CB B1-3
      CALL APNT(750.,820.,-1,-4)
      CALL VECT(0.,70.)
      CALL ARROWU(1,4,-1)
C
C     DRAW THE BUS.
C
      CALL APNT(750.,700.,-1,-4)
      CALL HBUS(-600.,5)
C
C     DRAW CONNECTION TO GENERATOR 2.
C
      CALL APNT(140.,580.,-1,-4)
      CALL SUBP(7,9)                           !SUBP: CL CB GEN 2
      CALL SUBP(107,109)                       !SUBP: OP CB GEN 2
      CALL APNT(116.,365.,1,-4)
      CALL SUBP (920)                          !SUBP: GEN 2 TRANS
      CALL CIRCLE(35.,-1,4,-1)
      CALL RDOT(34.,35.,-1,-4)
      CALL SUBP(921)                           !SUBP: GEN 2 TRANS ONLY
      CALL VECT(0.,80.)
      CALL TRANSF
      CALL VECT(0.,60.)
      CALL ESUB
      CALL ESUB
C
C     WRITE THE TEXT FOR THE GENERATOR.
C
      CALL APNT(25.,355.,-1,-5)
      CALL STAT(-1)
      CALL SUBP(922)                           !SUBP: 'GEN' TEST
      CALL TEXT('GEN')
      CALL ESUB
      CALL STAT(1)
      CALL APNT(143.,355.,-1,-5)
      CALL TEXT('2')
C
C     DRAW THE SERVICE LOADS.
C
      CALL APNT(400.,400.,-1,-4)
      CALL SUBP(923)                           !SUBP: ST SERV LD
      CALL LOAD
      CALL ESUB
      CALL APNT(400.,400.,-1,-4)
      CALL VECT(100.,0.)
      CALL SUBP(924,923)                       !SUBP: AG SERV LD
      CALL STAT(-1)
      CALL APNT(350.,280.,-1,-5)
      CALL TEXT('SERVICE LOADS')
      CALL APNT(450.,400.,-1,-4)
      CALL SUBP(925,921)                       !SUBP: ST SERV TRANS
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(46,9)                          !SUBP: CL CB ST SERV
      CALL SUBP(146,109)                       !SUBP: OP CB ST SERV
      CALL APNT(450.,600.,-1,-4)
      CALL VECT(0.,100.)
C
C     DRAW GENERATOR 3.
```

```
C
      CALL APNT(715.,365.,-1,-4)
      CALL SUBP(926,920)                      !SUBP: GEN 3 TRANS
      CALL RDOT(-10.,0.,-1,-4)
      CALL SUBP(8,9)                          !SUBP: CL CB GEN 3
      CALL SUBP(108,109)                      !SUBP: OP CB GEN 3
      CALL APNT(825.,355.,-1,-5)
      CALL SUBP(927,922)                      !SUBP: GEN TEXT
      CALL APNT(743.,355.,-1,-5)
      CALL STAT(1)
      CALL TEXT('3')
C
C     WRITE ALL THE TEXT FOR THE GENERATORS AS A SUBPICTURE TO BE
C     COPIED FOR THE OTHER GENERATOR.
C
      CALL APNT(650.,290.,-1,-5)
      CALL STAT(-1)
      CALL TEXT('PRIME MOVER')
      CALL APNT(650.,260.,-1,-5)
      CALL TEXT('EXCITATION')
      CALL APNT(650.,220.,-1,-5)
      CALL TEXT('VOLTAGE      AMPS')
      CALL APNT(740.,180.,1,-4)
      CALL SUBP(902)                          !SUBP: RAISE VOLT GEN 3
      CALL VECT(0.,30.)
      CALL ARROWU(1,4,-1)
      CALL ESUB
      CALL APNT(740.,170.,1,-4)
      CALL SUBP(903)                          !SUBP: LWR VOLT GEN 3
      CALL VECT(0.,-30.)
      CALL ARROWD(1,4,-1)
      CALL ESUB
      CALL APNT(825.,150.,-1,-5)
      CALL TEXT('P.F.')
      CALL APNT(665.,100.,-1,-5)
      CALL TEXT('FREQ')
      CALL APNT(740.,60.,1,-4)
      CALL SUBP(904,902)                      !SUBP: RAISE VOLT GEN 3
      CALL APNT(740.,50.,1,-4)
      CALL SUBP(905,903)                      !SUBP: LWR FREQ GEN 3
      CALL APNT(835.,80.,-1,-5)
      CALL TEXT('KW')
C
C     DRAW THE PRIME MOVER AND EXCITATION SWITCHES FOR GENERATOR 3.
C
      CALL STAT(1)
      CALL APNT(840.,290.,1,-5)
      CALL SUBP(149)                          !SUBP: GEN 3 DC SUP ON
      CALL TEXT('ON')
      CALL ESUB
      CALL OFF(149)
      CALL SUBP(49)                           !SUBP: GEN 3 DC SUP OFF
      CALL TEXT('OFF')
      CALL ESUB
      CALL APNT(840.,260.,1,-5)
      CALL SUBP(150,149)                      !SUBP: GEN 3 EXCIT ON
      CALL OFF(150)
      CALL SUBP(50,49)                        !SUBP: GEN 3 EXCIT OFF
C
C     PLACE THE DATA TO BE MONITORED IN CORRECT POSITION.
C
      CALL APNT(650.,165.,-1,-5)
      CALL NMBR(941,VOLTS,5,'(F5.2)')
      CALL APNT(650.,45.,-1,-5)
      CALL NMBR(942,VOLTS,5,'(F5.2)')
      CALL APNT(810.,190.,-1,-5)
```

```
      CALL NMBR(943,VOLTS,5,'(F5.2)')
      CALL APNT(810.,120.,-1,-5)
      CALL NMBR(944,VOLTS,5,'(F5.2)')
      CALL APNT(810.,50.,-1,-5)
      CALL NMBR(945,VOLTS,5,'(F5.2)')
C     WRITE ALL THE TEXT FOR THE GENERATORS AS A SUBPICTURE TO BE
C     COPIED FOR THE OTHER GENERATOR.
C
      CALL APNT(50.,290.,-1,-5)
      CALL STAT(-1)
      CALL TEXT('PRIME MOVER')
      CALL APNT(50.,260.,-1,-5)
      CALL TEXT('EXCITATION')
      CALL APNT(50.,220.,-1,-5)
      CALL TEXT('VOLTAGE       AMPS')
      CALL APNT(140.,180.,1,-4)
      CALL SUBP(906,902)                    !SUBP: RAISE VOLT GEN 2
      CALL APNT(140.,170.,1,-4)
      CALL SUBP(907,903)                    !SUBP: LWR VOLT GEN 2
      CALL APNT(225.,150.,-1,-5)
      CALL TEXT('P.F.')
      CALL APNT(65.,100.,-1,-5)
      CALL TEXT('FREQ')
      CALL APNT(140.,60.,1,-4)
      CALL SUBP(908,902)                    !SUBP: RAISE FREQ GEN 2
      CALL APNT(140.,50.,1,-4)
      CALL SUBP(909,903)                    !SUBP: LWR FREQ GEN 2
      CALL APNT(235.,80.,-1,-5)
      CALL TEXT('KW')
C
C     DRAW THE PRIME MOVER AND EXCITATION SWITCHES FOR GENERATOR 2.
C
      CALL STAT(1)
      CALL APNT(240.,290.,1,-5)
      CALL SUBP(147,149)                    !SUBP: GEN 2 DC SUP ON
      CALL OFF(147)
      CALL SUBP(47,49)                      !SUBP: GEN 2 DC SUP OFF
      CALL APNT(240.,260.,1,-5)
      CALL SUBP(148,149)                    !SUBP: GEN 2 EXCIT ON
      CALL OFF(148)
      CALL SUBP(48,49)                      !SUBP: GEN 2 EXCIT OFF
C
C     CALL THE REST OF THE PROGRAM.
C
      CALL FINISH
      STOP
      END
```

THE FOLLOWING IS CONTAINED IN THE FILE, STAT1.CMD:

```
STAT1/-CP,TI:/SH=STAT1,ST1FIN,DSPSUB,GLIB/LB
/
ASG=GRO:1
//
```

```
C
C       SUBROUTINE TO FINISH THE STATION 1 PICTURE.
C
        SUBROUTINE FINISH
        CALL OFF(155)
C
C       PLACE THE DATA TO BE MONITORED IN CORRECT POSITION.
C
        CALL APNT(50.,165.,-1,-5)
        CALL NMBR(936,VOLTS,5,'(F5.2)')
        CALL APNT(50.,45.,-1,-5)
        CALL NMBR(937,VOLTS,5,'(F5.2)')
        CALL APNT(210.,190.,-1,-5)
        CALL NMBR(938,VOLTS,5,'(F5.2)')
        CALL APNT(210.,120.,-1,-5)
        CALL NMBR(939,VOLTS,5,'(F5.2)')
        CALL APNT(210.,50.,-1,-5)
        CALL NMBR(940,VOLTS,5,'(F5.2)')
        CALL APNT(162.,710.,-1,-5)
        CALL NMBR(966.,VOLTS,5,'(F5.2)')
        CALL STAT(-1)
        CALL TEXT(' VOLTS')
        CALL APNT(233.,650.,-1,-5)
        CALL TEXT('STATION 1')
        CALL STAT(1)
C
C       ADD THE MENU FOR THE SWITCHES FOR THE OTHER PICTURES NEEDED
C       ON THIS PICTURE.
C
        CALL MENU(930.,500.,-50.,910,'SYNCRN','GEN 1','STAT11','MENU')
C
C       WRITE THE TEXT FOR THE 'ARE YOU SURE' DOUBLE CHECK
C       FOR LIGHT PEN HITS DURING THE PROGRAM RUN.
        CALL SUBP(90)                               !SUBP: R U SURE
        CALL APNT(870.,750.,-1,-5,1)
        CALL TEXT('YOU SURE?')
C
C       DRAW THE 'YES' SUBPICTURE.
C
        CALL SUBP(91)                               !SUBP: CK IF YES
        CALL APNT(880.,700.,1,-5,-1)
        CALL TEXT('YES')
        CALL ESUB
C
C       DRAW THE 'NO' SUBPICTURE.
C
        CALL SUBP(92)
        CALL APNT(950.,700.,1,-5,-1)
        CALL TEXT('NO')
        CALL ESUB
C
        CALL ESUB
C
        CALL OFF(90)
C
C       SAVE THE PICTURE IN A FILE CALLED 'STAT1.DPY'.
C
        CALL SAVE('STAT1.DPY')
C
        RETURN
        END
```

The Contents of the file, DSPSUB.FTN;

```
C
C       SUBROUTINE TO DRAW A CIRCLE OF RADIUS R.  BEGINS AT THE
C       LEFT SIDE AND ENDS AT THE LEFT SIDE.
C
C       PASS THE DESIRED PARAMETERS IN THE ARGUMENTS IN THE
C       SAME ORDER AS L,I,F IN A NORMAL GRAPHICS CALL.
C
        SUBROUTINE CIRCLE(R,LP,INT,LF)
        CALL RDOT(0.,0.,LP,INT,LF)
        XOLD=-R
        YOLD=0.
        TH=3.14159
        DTH=15./57.2958
        DO 1 I=1,24
        TH=TH+DTH
        XNEW=R*COS(TH)
        YNEW=R*SIN(TH)
        XP=XNEW-XOLD
        YP=YNEW-YOLD
        IF(XP .LT. 0.)GOTO 2
        XP=XP+.5
        GOTO 3
2       XP=XP-.5
3       CONTINUE
        IF(YP .LT. 0.)GOTO 4
        YP=YP+.5
        GOTO 5
4       YP=YP-.5
5       CONTINUE
        CALL VECT(XP,YP)
        XOLD=XNEW
        YOLD=YNEW
1       CONTINUE
        RETURN
        END
C
C       SUBROUTINE TO DRAW A HORIZONTAL BUS OF VARIABLE LENGTH
C       STARTING FROM THE LEFT CENTER.  DIMENSIONS ARE H,3.
C       THE END IS THE RIGHT CENTER.
C
        SUBROUTINE HBUS(H,INT)
        CALL RDOT(0.,1.,-1,-INT)
        CALL VECT(H,0.)
        CALL RDOT(0.,-2.,,-INT)
        CALL VECT(-H,0.)
        CALL RDOT(H,1.,,-INT)
        RETURN
        END
C
C
C       SUBROUTINE TO DRAW A VERTICAL BUS OF LENGTH (V). STARTING
C       FROM THE BOTTOM CENTER AND ENDING AT THE TOP CENTER.
C       DIMENSIONS ARE 3 BY V.
C
        SUBROUTINE VBUS(V,INT)
        CALL RDOT(-1.,0.,-1,-INT)
        CALL VECT(0.,V)
        CALL RDOT(2.,0.,,-INT)
        CALL VECT(0.,-V)
        CALL RDOT(-1.,V,,-INT)
        RETURN
        END
```

```
C
C       SUBROUTINE TO DRAW A TRANSFORMER.
C       IT  STARTS ON THE BOTTOM CENTER AND ENDS ON THE
C       TOP CENTER OF THE TRANSFORMER. POSITION THE BEAM
C       ACCORDINGLY.  SIZE IS 36 BY 40.
C
        SUBROUTINE TRANSF
        DIMENSION TRANS(12)
        DATA TRANS/6.,12.,6.,-12.,6.,12.,6.,-12.,
     1 6.,12.,6.,-12./
        CALL RDOT(-18.,28.,0,-4)                     ! R; -18,28
        CALL FIGR(TRANS,12,101)                      !SUBP: TOP  OF TRANSFORMER
        CALL RDOT(0.,-6.,0,-4)                       ! R; 18,22
        CALL VECT(-36.,0.,0,4)
        CALL RDOT(0.,-4.,0,-4)                       ! R; -18,18
        CALL VECT(36.,0.,0,4)
        CALL RDOT(0.,-6.,0,-4)                       ! R; 18,12
        DO 2 I=1,12
2       TRANS(I)=-TRANS(I)
        CALL FIGR(TRANS,12,102)                      !SUBP: BOTTOM OF TRANS
        CALL RDOT(18.,28.,0,4)                       ! R; 0,40
        RETURN
        END
C
C       SUBROUTINE TO DRAW A CLOSED CIRCUIT BREAKER.
C       START AT THE LOWER LEFT AND FINISH THERE. SIZE IS 20 BY 20.
C
        SUBROUTINE CBCLD(INT)
        CALL VECT(0.,20.,1,INT)
        CALL VECT(20.,0.)
        CALL VECT(0.,-20.)
        CALL VECT(-20.,0.)
        RETURN
        END
C
C       SUBROUTINE TO DRAW A OPEN CIRCUIT BREAKER.
C       START AT THE LOWER LEFT AND FINISH AT THE LOWER RIGHT.
C       SIZE IS 20 BY 20.
C
        SUBROUTINE CBOPN(INT)
        CALL VECT(20.,20.,-1,INT)
        CALL RDOT(-20.,0.,,-INT)
        CALL VECT(20.,-20.)
        RETURN
        END
C
C       SUBROUTINE TO CREATE A HORIZONTAL RESISTOR, 20 BY 45.
C       STARTING AT TOP.  ENDING AT THE BOTTOM.
C       THE RESISTOR HAS A GROUND SYMBOL AT THE BOTTOM.
C
        SUBROUTINE RESIST
        DIMENSION RESIS(14)
        DATA RESIS/0.,-20.,-10.,-5.,20.,-10.,-20.,-10.,20.,
     1 -10.,-10.,-5.,0.,-5./
        CALL FIGR(RESIS,14,114,,4)                   !SUBP: HORIZ RES
        CALL RDOT(-10.,0.,,-4)                       ! PB FOR GRND
        CALL VECT(20.,0.,,4)
        CALL RDOT(-4.,-3.,,-4)
        CALL VECT(-12.,0.,,4)
        CALL RDOT(4.,-3.,,-4)
        CALL VECT(4.,0.)
        RETURN
        END
```

```
C
C       SUBROUTINE TO CREATE A VERTICAL RECTANGULAR BOX
C       REPRESENTATION OF A LOAD.
C       DIMENSIONS ARE 20 BY 86, STARTING AT TOP AND ENDING AT BOTTOM.
C
        SUBROUTINE LOAD
        CALL VECT(0.,-20.,-1,4,-1,1)              ! R;  0,-20
        CALL RDOT(10.,0.,,-4)                     ! R;  10,-10
        CALL VECT(0.,-40.)                        ! R;  10,-60
        CALL VECT(-20.,0.)                        ! R; -10,-60
        CALL VECT(0.,40.)                         ! R; -10,-20
        CALL VECT(20.,0.)                         ! R;  10,-20
        CALL RDOT(-10.,-40.,,-4)                  ! R;  0,-60
        CALL VECT(0.,-20.)                        ! R;  0,-80
        CALL GND
        RETURN
        END
C
C       SUBROUTINE TO DRAW A GROUND SYMBOL HORIZONTALLY.
C       ITS DIMENSIONS ARE 20 BY 6.
C
        SUBROUTINE GND
        CALL RDOT(-10.,0.,,-4)                    ! R;-10,-80  PB FOR GND
        CALL VECT(20.,0.)
        CALL RDOT(-4.,-3.,,-4)                    ! R; 16,-83
        CALL VECT(-12.,0.)                        ! R; -8,-83
        CALL RDOT(4.,-3.,,-4)
        CALL VECT(4.,0.)
        RETURN
        END
C
C       SUBROUTINES TO
C       DRAW AN OPEN AND CLSD MANUAL SWITCH VERTICALLY, 20 UNITS LONG
C       HINGE IT ON THE TOP AND SWING IN TO THE RIGHT.  STARTS AT THE
C       TOP AND ENDS ON THE BOTTOM CONTACT.
C
C       SUBROUTINE TO DRAW THE  OPEN MANUAL SWITCH.
C
        SUBROUTINE SWOPV(INT)
        CALL RDOT(0.,-1.,1,INT+1)
        CALL RDOT(1.,-1.,1,-INT)                  ! R; 1,-2
        CALL VECT(12.,-12.,1,INT)
        CALL RDOT(-13.,-5.,1,INT+1)               ! R; 0,-19
        RETURN
        END
C
C       SUBROUTINE TO DRAW THE CLOSED MANUAL SWITCH.
C
        SUBROUTINE SWCLV
        CALL RDOT(0.,-1.,1,6)
        CALL RDOT(0.,-1.,1,-4)                    ! R; 0,-2
        CALL VECT(0.,-16.,1,4)
        CALL RDOT(0.,-1.,1,6)                     ! R; 0,-19
        RETURN
        END
C
C       DRAW AN OPEN AND CLOSED MANUAL SWITCH HORIZONTALLY.
C       20 UNITS LONG.  HINGE IT ON THE LEFT AND SWING IT UP.
C       STARTS ON THE LEFT AND ENDS ON THE RIGHT.
C
C       SUBROUTINE TO DRAW OPEN SWITCH HORIZONTALLY.
C
        SUBROUTINE SWOPH(INT)
        CALL RDOT(1.,0.,,INT+1)
        CALL RDOT(1.,1.,,-INT)                    ! R; 2,1
        CALL VECT(12.,12.,1)                      ! R; 14,13
        CALL RDOT(5.,-13.,-1,INT+1)               ! R; 19,0
        RETURN
        END
```

```
C
C       SUBROUTINE TO DRAW CLOSED SWITCH HORIZONTALLY.
C
        SUBROUTINE SWCLH(INT)
        CALL RDOT(1.,0.,,INT+1)                     ! R; 1,0
        CALL RDOT(1.,0.,,-4)                        ! R; 2,0
        CALL VECT(16.,0.,1)                         ! R; 18,0
        CALL RDOT(1.,0.,-1,INT+1)                   ! R; 19,0
        RETURN
        END
C
C       SUBROUTINE TO DRAW A DOWNWARD POINTING ARROW.
C       STARTS AT THE POINT AND ENDS THERE.
C
        SUBROUTINE ARROWD(LP,INT,LF)
        CALL VECT(-7.,10.,LP,INT,LF)
        CALL RDOT(14.,0.,,-INT)
        CALL VECT(-7.,-10.)
        RETURN
        END
C
C       SUBROUTINE TO DRAW A UPWARD POINTING ARROW.
C       STARTS AT THE POINT AND ENDS THERE.
C
        SUBROUTINE ARROWU(LP,INT,LF)
        CALL VECT(-7.,-10.,LP,INT,LF)
        CALL RDOT(14.,0.,,-INT)
        CALL VECT(-7.,10.)
        RETURN
        END
C
C       SUBROUTINE TO DRAW A CAPACITOR HORIZONTALLY.  DIMENSIONS ARE
C       30 BY 10.
C
        SUBROUTINE CAPCTR
        CALL RDOT(-15.,0.,-1,-4)
        CALL VECT(30.,0.)
        CALL RDOT(0.,-9.,-1,-4)
        CALL VECT(-2.,1.)
        CALL VECT(-3.,1.)
        CALL VECT(-2.,1.)
        CALL VECT(-6.,1.)
        CALL VECT(-4.,0.)
        CALL VECT(-6.,-1.)
        CALL VECT(-2.,-1.)
        CALL VECT(-3.,-1.)
        CALL VECT(-2.,-1.)
        CALL RDOT(15.,4.)
        RETURN
        END
```

THE 'CONTROL.FTN' FILE CONTAINS THE FORTRAN SOURCE FOR THE MAIN CONTROL PROGRAM. IT USES SUBROUTINES CONTAINED IN A FILE CALLED 'CNTLSB.FTN'. EDIT THE FILES USING THE EDIT UTILITY.

TO COMPILE WITHOUT A SOURCE LISTING, TYPE:

```
                >FOR CONTROL=CONTROL
AND             >FOR CNTLSB=CNTLSB
```

TO COMPILE WITH A NUMBERED SOURCE LISTING, TYPE:

```
                >FOR CONTROL=CONTROL/LI:1
AND             >FOR CNTLSB=CNTLSB/LI:1
```

TO TASK BUILD USING THE INDIRECT COMMAND FILE, TYPE:

```
                >TKB @CONTROL
```

THIS CAUSES FILES, 'CONTROL.OBJ' AND 'CNTLSB.OBJ', TO BE CREATED BY THE 'FOR' COMMAND AND A FILE, 'CONTROL.TSK', TO BE CREATED BY THE 'TKB' COMMAND.
THE '@' SYMBOL INDICATES THE USE OF AN INDIRECT COMMAND FILE. THIS IS FILE, 'CONTROL.CMD', AND CONTAINS THE TASK BUILDER COMMANDS NECESSARY TO BUILD THIS TASK.

TO CONTROL THE SIMULATOR WITH THE CONTROL TASK, TURN ON THE VT-11 AND TYPE:

```
                >RUN CONTROL
```

THE LIST OF SUBPICTURES ACTIVE AT ANY ONE TIME WILL DEPEND UPON WHICH DISPLAY IS ACTIVE AT THE TIME.

```
C
C       THE MAIN CONTROL PROGRAM FOR THE SIMULATOR.
C
C       DETAILED COMMENT APPLIES ONLY TO THE FIRST TWO PICTURES.
C       THE REMAINDER ARE IDENTICAL AND USE A SUBROUTINE TO
C       ACCOMPLISH THE TRICKERY.
C
C       SET UP THE COMMON BLOCKS REQUIRED FOR THE GRAPHICS BUFFER
C       AND ALSO THE BLOCK THAT STORES THE ARRAY SW(100).
C       THIS ARRAY IS A 'LOGICAL*1' ARRAY WHICH MEANS IT USES ONLY
C       ONE BYTE PER SUBSCRIPT.  THE ARRAY'S NAME IS SHORT FOR SWITCH,
C       AND THE ARRAY CONTAINS DATA TO CHANGE 100 SWITCHES ON THE DRS-11/
C       SIMULATOR INTERFACE.  THE ARRAY IS ALSO INITIALIZED TO ALL 'FALSE'
C       OR INITIAL  SWITCH  POSITION TO BEGIN THE PROGRAM.
C
        LOGICAL*1 SW
        COMMON/DFILE/IBUF(1100)/SUB/SW(100)
        DIMENSION TIM(2),DAT(3)
        DATA SW/100*.FALSE./
        DATA TIM/2*0./
C
C       THE ONLY AUTOMATIC PICTURE ON THE PROGRAM IS THE INTRODUCTION
C       PICTURE AND HERE IT IS.
C
100     CALL INIT(1100)
        CALL RSTR('INTRO.DPY')             ! RESTORE PICT FROM DISK
140     CALL CLREF(11)                     ! CLR THE LGT PEN EVENT FLAG
120     CALL LPEN(M,N,XX,YY)               ! RET THE SUBP WHERE LGT PEN HIT
        IF(M.EQ.0)GOTO 120                 ! LOOP TO WAIT FOR LGT PEN HIT
        CALL ON(90)                        ! TURN ON YOU SURE?
        CALL CLREF(11)                     ! CLR LPT PEN EVENT FLAG
        NZ=0                               ! RESET NZ TO ZERO
130     CALL LPEN(MZ,NZ)                   ! CK IF HIT IS SURE
        IF(NZ.NE.91 .AND. NZ.NE.92)GOTO 130       ! CHECK YES OR NO
        CALL OFF(90)                       ! TURN OFF YOU SURE?
        IF(NZ.EQ.92)GOTO 140               ! IF NO GO TO START
C
C       COMPUTED GO TO FOR ROUTING OF MENU HITS.
C
        GOTO(200,1100,900,800,1000,300,700,400,600,500,9999),N
C
C       INITIATE THE HV TRANSMISSION PICTURE WITH ASSOCIATED DATA.
C
200     CALL INIT(1100)
        CALL RSTR('HVTRNS.DPY')            ! RESTORE PICTURE
C
C       PLACE THE DATE ON THE PICTURE.
C
        CALL DATE(DAT)
        CALL APNT(885.,900.,-1,-5)
        CALL TEXT(DAT)
C
C       UPDATE THE PICTURE'S SWITCHES AND CIRCUIT BREAKERS.
C
210     CALL UPDATE(1,20)
        CALL UPDATE(60,64)
C
C       SET UP LOOP TO DISPLAY DATA AND DETECT LIGHT PEN HITS.
C
220     CALL CLREF(11)                     ! CLR LPEN HIT
230     CALL TIME(TIM)
        CALL NMBR(239,TIM(1),4,'(A4)')
```

```
      CALL NMBR(240,TIM(2),4,'(A4)')
C
C     CHECK FOR LIGHT PEN HIT.
C
      CALL LPEN(M,N,XX,YY)
      IF(M.EQ.0)GOTO 230
C
C     CLEAR EVENT FLAG FOR THE LIGHT PEN SO NO FURTHER ACTION WILL
C     TAKE PLACE UNTIL THE DOUBLE CHECK LIGHT PEN HIT IS MADE.
C     FIRST TURN ON THE 'YOU SURE?' SWITCH.
C
      CALL ON(90)
      CALL CLREF(11)
C
C     SET UP LOOP FOR LIGHT PEN HIT ON 'ARE YOU SURE?' ANSWER.
C
      NZ=0
240   CALL LPEN(MZ,NZ)
      IF(NZ.NE.91 .AND. NZ.NE.92)GO TO 240
C
C     TURN OFF 'YOU SURE?' PICTURE.
C
      CALL OFF(90)
C
      IF(NZ.EQ.92)GOTO 220              ! IF 'NO' GO TO START
C
C     IF THE HIT WAS NOT ON A SWITCH OR CIRCUIT BREAKER GO TO 250.
C
      IF(N.GT.200)GOTO 260
      IF(N.GT.90)N=N-100
C
C     ROUTINE TO CHANGE CIRCUIT BREAKER AND SWITCHES ARRAY.
C
250   IF(SW(N).EQ..TRUE.)GOTO 251
      SW(N)=.TRUE.
      GOTO 210
251   SW(N)=.FALSE.
      GOTO 210
C
C
C     CALCULATE THE NUMBERS FOR THE COMPUTED GO TO STATEMENTS.
C
260   NB=N-200
C
      GOTO(1000,900,100,300,400,800),NB
C
C
C***********************************************************************
C
C     THE SUBSTATION 4 PICTURE AND DATA.
C
300   CALL INIT(1100)
      CALL RSTR('SUBST4.DPY')
      CALL CHECK(21,26,27,27,949,950,N)
      NB=N-300
      GOTO(200,600,700,100),NB
C
C***********************************************************************
C
C     THE SUBSTATION 6 PICTURE AND DATA.
C
400   CALL INIT(1100)
      CALL RSTR('SUBST6.DPY')
      CALL CHECK(32,38,71,76,958,958,N)
      NB=N-400
      GOTO(200,500,600,100),NB
```

```
C
C*********************************************************************
C
C        THE LOAD CENTER PICTURE AND DATA PLACEMENT.
C
500      CALL INIT(1100)
         CALL RSTR('LDCTR.DPY')
         CALL CHECK(37,44,45,45,954,958,N)
         NB=N-500
         GOTO (200,400,100),NB
C
C*********************************************************************
C
C        THE NETWORK SYSTEM PICTURE AND DATA PLACEMENT.
C
600      CALL INIT(1100)
         CALL RSTR('NETSYS.DPY')
         CALL CHECK(26,32,33,33,959,964,N)
         NB=N-600
         GOTO(200,400,300,200,100),NB
C
C*********************************************************************
C
C        THE INDUSTRIAL LOAD PICTURE AND DATA PLACEMENT.
C
700      CALL INIT(1100)
         CALL RSTR('INDLD.DPY')
         CALL CHECK(53,58,23,23,951,953,N)
         NB=N-700
         GOTO(200,300,700,700,900,1000,800,100),NB
C
C*********************************************************************
C
C        THE STATION 11 PICTURE AND DATA PLACEMENT.
C
800      CALL INIT(1100)
         CALL RSTR('STAT11.DPY')
         CALL CHECK(13,17,65,70,946,948,N)
         NB=N-800
         GOTO(200,200,800,800,900,1000,100),NB
C
C
C*********************************************************************
C
C        THE STATION 1 PICTURE AND DATA PLACEMENT.
C
900      CALL INIT(1100)
         CALL RSTR('STAT1.DPY')
         CALL CHECK(7,12,46,50,936,945,N)
         NB=N-900
         GOTO(200,900,900,900,900,900,900,900,900,1100,1000,800,100),NB
C
C*********************************************************************
C
C        THE GENERATOR 1 PICTURE AND DATA PLACEMENT.
C
1000     CALL INIT(1100)
         CALL RSTR('GEN1.DPY')
         CALL CHECK(6,6,51,52,931,935,N)
         NB=N-1000
         GOTO(200,1100,900,800,100,1000,1000,1000,1000),NB
C
C*********************************************************************
C
C        THE SYNCRONIZING PICTURE AND DATA PLACEMENT.
C
```

```
1100    CALL INIT(1100)
        CALL RSTR('SYNCRN.DPY')
        CALL CHECK(4,14,15,15,931,932,N)
        NR=N-1100
        GOTO (1100,1100,1100,1100,1100,1100,1100,1100,1100,1100,1100,
     1 1100,1100,1100,200,900,1000,800,100),NR
C
C
9999    STOP
        END
```

THE FOLLOWING IS CONTAINED IN THE FILE, CONTROL.CMD:

```
CONTROL/-CP,TI:/SH=CONTROL,CNTLSB,GLIB/LB
/
ASG=GR0:1
//
```

The Contents of the file, CNTLSB.FTN;

```
C        SUBROUTINE TO CHOOSE A CHANNEL OF THE A-D CONVERTERS FOR
C        DISPLAY ON THE SCREEN.
C
         SUBROUTINE SINCON(ICHAN,VOLTS)
         DIMENSION IERR(2),VAR(1)
C
C        SINGLE CHANNEL A/D CONVERSION PROGRAM
C        PARAMETER (ICHAN) CONTAINS THE CHANNEL NO. (1-32)
C
10       ICHAL=ICHAN-931
         IF(ICHAL.GE.17)GO TO 30
         CALL ASARLN(3,IERR,0)
         GO TO 50
30       ICHAL=ICHAL-17
35       CALL ASARLN(3,IERR,1)
50       CONV=ADC(ICHAL,VAR,1,IERR)/64.
55       VOLTS=(CONV*5.0)/1023.0
85       RETURN
         END
C
C        SUBROUTINE TO UPDATE THE PICTURE EVERY TIME IT IS CALLED.
C        THE ARGUMENTS ARE THE BOUNDS OF THE SUBPICTURE'S NUMBERS
C        THAT BELONG TO THE PICTURE BEING DISPLAYED AT THE TIME.
C
         SUBROUTINE UPDATE(J1,J2)
         LOGICAL*1 SW
         COMMON/SUB/SW(100)
C
         DO 250 I=J1,J2
C
C        CHECK IF UPDATE IS FOR FUNCTIONAL SWITCH OF CIRCUIT BREAKER.
C
         IF(I.GT.46)GOTO 440
         II=I+100
C
C        UPDATE CIRCUIT BREAKER SUBPICTURE.
C
         IF(SW(I).EQ..FALSE.)GOTO 251
         CALL OFF(II)
         GOTO 250
251      CALL ON(II)
         GOTO 250
C
C        UPDATE FUNCTIONAL SWITCH PICTURE.
C
440      IF(SW(I).EQ..FALSE.)GOTO 451
         CALL OFF(I)
         CALL ON(I+100)
         GOTO 250
451      CALL OFF(I+100)
         CALL ON(I)
250      CONTINUE
         RETURN
         END
C
C
C        SUBROUTINE TO UPDATE THE DATA AND CHANGE THE SWITCHES IN THE
C        MAIN PROGRAM.  L1,L2,L3, AND L4 ARE ARGUMENTS FOR TWO UPDATE
C        CALLS.  L5 AND L6 ARE ARGUMENTS FOR DO LOOP TO POSITION DATA
C        ON THE SCREEN.  N IS SUBPICTURE NUMBER WHERE LIGHT PEN HIT OCCURS
C        FOR PASSING BACK TO MAIN PROGRAM.  THE ROUTINE WORKS AS THE LOOPS
C        IN THE MAIN PROGRAM.
```

```
C
      SUBROUTINE CHECK(L1,L2,L3,L4,L5,L6,N)
      LOGICAL*1 SW
      COMMON/SUB/SW(100)
100   CALL UPDATE(L1,L2)
      CALL UPDATE(L3,L4)
120   CALL CLREF(11)
130   DO 131 I=L5,L6
      CALL SINCON(I,VOLTS)
131   CALL NMBR(I,VOLTS,5,'(F5.2)')
      CALL LPEN(M,N,XX,YY)
      IF(M.EQ.0)GOTO 130
      CALL ON(90)
      CALL CLREF(11)
      NZ=0
140   CALL LPEN(MZ,NZ)
      IF(NZ.NE.91 .AND. NZ.NE.92)GOTO 140
      CALL OFF(90)
      IF(NZ.EQ.92)GOTO 120
      IF(N.GT.200)GOTO 150
      IF(N.GT.90)N=N-100
      IF(SW(N).EQ..TRUE.)GOTO 151
      SW(N)=.TRUE.
      GOTO 100
151   SW(N)=.FALSE.
      GOTO 100
150   RETURN
      END
```

The Contents of the file, ADCHNL.TXT;

```
        THIS FILE CONTAINS THE INFORMATION FOR THE ANALOG TO DIGITAL
CHANNELS.  THE INFORMATION NECESSARY TO USE THE AR-11'S SO THAT THEY
MONITOR THE CHANNELS THAT THE CONTROL PROGRAM DISPLAYS, FOLLOWS.
THERE ARE PRESENTLY 32 CHANNELS AVAILABLE NUMBERED FROM ONE TO 32.
THE PROGRAM CHANGES THE CHANNELS TO BE NUMBERED FROM ZERO TO 31.
FOR THE AR-11'S.  THIS LISTING IS NUMBERED FROM ONE TO 32 SO THAT
ANYONE WORKING WITH THE SOFTWARE WILL NOT HAVE TO CHANGE THE NUMBERS.
ANYONE WORKING WITH THE HARDWARE WILL HAVE TO SUBTRACT ONE FROM THE
CHANNEL NUMBERS GIVEN HERE.
        THERE ARE SOME ADDITIONAL CHANNELS NEEDED.  THEY HAVE BEEN GIVEN
PICTURE NUMBERS FROM 33 UP.  ADDITIONAL CIRCUITRY WILL HAVE TO BE BUILT
TO HANDLE THESE.  IT SHOULD BE POSSIBLE TO OPERATE WITHOUT THE EXTRA
CHANNELS ANYWAY.

CHANNEL NUMBER        VARIABLE          FROM              SUBPICTURE NUMBER

       1               VOLTS            GEN 1                   931
       2               FREQ             GEN 1                   932
       3               AMPS             GEN 1                   933
       4               PF               GEN 1                   934
       5               KW               GEN 1                   935
       6               VOLTS            GEN 2                   936
       7               FREQ             GEN 2                   937
       8               AMPS             GEN 2                   938
       9               PF               GEN 2                   939
      10               KW               GEN 2                   940
      11               VOLTS            GEN 3                   941
      12               FREQ             GEN 3                   942
      13               AMPS             GEN 3                   943
      14               PF               GEN 3                   944
      15               KW               GEN 3                   945
      16               VOLTS            INT CON L-L             946
      17               FREQ             INT CON                 947
      18               AMPS             INT CON                 948
      19               VOLTS            SUB-ST 4 LEFT BUS       949
      20               VOLTS            SUB-ST 4 RIGHT BUS      950
      21               VOLTS            INDUST LD               951
      22               PF               INDUST LD               952
      23               AMPS             INDUST LD               953
      24               AMPS             LD CTR RT               954
      25               AMPS             LD CTR LT               955
      26               VOLTS            LD CTR LT               956
      27               VOLTS            LD CTR RT               957
      28               VOLTS            SUB-ST 6 BUS            958
      29               VOLTS L-N 1      NETWORK SYS             959
      30               VOLTS L-N 2      NETWORK SYS             960
      31               VOLTS L-N 3      NETWORK SYS             961
      32               AMPS LINE 1      NETWORK SYS             962
       *       *        *       *        *       *       *        *
      33               AMPS LINE 2      NETWORK SYS             963
      34               AMPS LINE 3      NETWORK SYS             964
      35               VOLTS            ST 11 BUS               965
      36               VOLTS            ST 1 BUS                966
      37               VOLTS            GEN 1 ABOVE CKT BKR     967
```

The Contents of the file, SUBNUM.TXT; a list of the common subpictures.

THIS FILE CONTAINS THE LISTINGS OF THE SUBPICTURE NUMBERS AND THEIR CORRESPONDING SIMULATOR PART OR DEFINITION.

THE FIRST SECTION CONTAINS THE SUBPICTURE NUMBERS ASSOCIATED WITH ALL THE PICTURES. THE FIRST SUBPICTURE NUMBER IS THE INITIAL CONDITION POSITION OF THE ASSOCIATED SWITCH AND THE SECOND NUMBER IS THE FIRST PLUS 100. THE SECOND SUBPICTURE IS THE OPPOSITE POSITION OF THE SWITCH FROM ITS INITIAL CONDITION. THE INITIAL CONDITION OF ALL CIRCUIT BREAKERS AND SWITCHES IS OPEN EXCEPT FOR THOSE MARKED CLOSED.

SUBPICTURE NUMBER	SWITCH	COMPANION SUBPICTURE
1	CB RING BUS #1	101
2	CB RING BUS #2	102
3	CB RING BUS #3	103
4	CB RING BUS #4	104
5	CB RING BUS #5	105
6	CB GEN 1	106
7	CB GEN 2	107
8	CB GEN 3	108
9	CB LINE A1-3	109
10	CB LINE B1-3	110
11	CB LINE C1-11 STAT 1	111
12	CB LINE D1-11 STAT 1	112
13	CB INT CON	113
14	CB LINE C1-11 STAT 11	114
15	CB LINE D1-11 STAT 11	115
16	CB LINE F10-11	116
17	CB LINE E10-11	117
18	CB LINE E4-6	118
19	CB SUB ST 4 #1	119
20	CB SUB ST 4 #2	120
21	CB SUB ST 4 #3	121
22	CB SUB ST 4 #6	122
23	CB SUB ST 4 #9	123
24	CB SUB ST 4 #4	124
25	CB SUB ST 4 #5	125
26	CB SUB ST 4 #8	126
27	CB SUB ST 4 #7	127
28	CB NET SYS #4	128
29	CB NET SYS #2	129
30	CB NET SYS #3	130
31	CB NET SYS #1	131
32	CB SUB ST 6 #4	132

SUBPICTURE NUMBER	SWITCH	COMPANION SUBPICTURE
33	CB SUB ST 6 #5	133
34	CB SUB ST 6 #1	134
35	CB SUB ST 6 #2	135
36	CB SUB ST 6 #3	136
37	CB SUB ST 6 #7	137
38	CB SUB ST 6 #6	138
39	CB LD CTR #SS-1	139
40	CB LD CTR #SS-3	140
41	CB LD CTR #1	141
42	CB LD CTR #2	142
43	CB LD CTR #3	143
44	CB LD CTR #4	144
45	CB LD CTR #SS-2	145
46	CB ST 1 SERV LD	146
47	SW GEN 2 DC SUP-PRIME MOVER	147
48	SW GEN 2 EXCITATION	148
49	SW GEN 3 DC SUP-PRIME MOVER	149
50	SW GEN 3 EXCITATION	150
51	SW GEN 1 DC SUP-PRIME MOVER	151
52	SW GEN 1 EXCITATION	152
53	SW INDUST LGT LD	153
54	SW INDUST HEAT LD	154
55	SW INDUST POWER LD #1	155
56	SW INDUST POWER LD #2	156
57	SW INDUST REACTIVE LD	157
58	SW INDUST CAPACITOR	158
59	NOT IN USE	159
60	SW SECT ST #602 (CLOSED)	160
61	SW SECT ST #605 (CLOSED)	161
62	SW SECT ST #601 (CLOSED)	162
63	SW SECT ST #604 (CLOSED)	163
64	SW SECT ST #603	164
65	SW STATION 11 EMER GEN	165
66	SW STATION 11 D1-11	166
67	SW STATION 11 F10-11	167
68	SW STATION 11 E10-11	168
69	SW STATION 11 INT CON	169
70	SW STATION 11 C1-11	170
71	SW SUB ST 6 #407	171
72	SW SUB ST 6 #509	172
73	SW SUB ST 6 #521	173
74	SW SUB ST 6 #514	174
75	SW SUB ST 6 #517	175
76	SW SUB ST 6 #414	176

REFERENCES

AR-11 User's Guide. Digital Equipment Corporation. Maynard, Mass.

IAS/RSX-11 FORTRAN IV User's Guide. DEC. Maynard, Mass.

Introduction to RSX-11M. DEC. Maynard, Mass.

Machover, C., Neighbors, M., Stuart, C. "Computer Graphics Displays," IEEE Spectrum, August, 1977.

Machover, C., Neighbors, M., Stuart, C. "Graphics displays: Factors in Systems Design," IEEE Spectrum, October, 1977.

PDP-11 FORTRAN Language Reference Manual. DEC. Maynard, Mass.

RSX-11M Executive Reference Manual. DEC. Maynard, Mass.

RSX-11M Operator's Procedures Manual. DEC. Maynard, Mass.

RSX-11M Utilities Procedures Manual. DEC. Maynard, Mass.

RSX-11M Task Builder Reference Manual. DEC. Maynard, Mass.

VT-11 Graphic Display Processor. DEC. Maynard, Mass.

Walker, B.S., Gurd, J.R., Drawneek, E.A. Interactive Computer Graphics. New York: Crane, Russak & Company, Inc., 1975.

VITA

Lawrence Ray Davis was born in Oklahoma City, Oklahoma on 1 August, 1948 to Lawrence A. and Rosemary Davis. Upon graduation from Southeast High School in Oklahoma City, he attended the University of Oklahoma, receiving a BSEE degree in December of 1970. Since then he has been a member of the United States Air Force; serving as a pilot and currently as a graduate student at the University of Oklahoma under the Air Force Institute of Technology's Civilian Institution program. After receiving his MSEE, he will be assigned as a member of the faculty at the Air Force Academy.