PITTAMPALLI, ESHWAR

INTERACTIVE GRAPHICS CONTROLLED POWER SYSTEM SIMULATION: A DISTRIBUTED MINI/MICRO PROCESSING APPROACH

The University of Oklahoma PH.D. 1979

University
Microfilms
International 300 N. Zeeb Road, Ann Arbor, MI 48106 18 Bedford Row, London WC1R 4EJ, England

THE UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

INTERACTIVE GRAPHICS CONTROLLED POWER SYSTEM SIMULATION: A DISTRIBUTED MINI/MICRO PROCESSING APPROACH.

A DISSERTATION

SUBMITTED TO THE GRADUATE FACULTY

in partial fulfillment of the requirement for the

degree of

DOCTOR OF PHILOSOPHY

BY

ESHWAR PITTAMPALLI

1979

INTERACTIVE GRAPHICS CONTROLLED POWER SYSTEM SIMULATION: A DISTRIBUTED MINI/MICRO PROCESSING APPROACH.

APPROVED BY

John E. Fagan

William L. Kuriger

William N. Huff

Gerald Tuma

DISSERTATION COMMITTEE

ACKNOWLEDGEMENT

While reading the works of many authors I have noted that they have expressed appreciation to their respective supervising professors in a very mandatory style. I used to think that when my turn came it would be one of those. But a sudden turn in my career changed this notion. I have experienced a real feeling of gratitude for all the moral courage given by most of the faculty and staff except one. I do not have words to express all my gratitude for their help in the continuation of my work at the University of Oklahoma. The first and foremost member to whom I am indebted all my life is Professor John E. Fagan, who took personal pains to help in all possible directions. Without his help this work would not have been possible. The next person to whom I wish to express my sincere appreciation is Professor Leon W. Zelby for his encouragement during crucial times.

My appreciation is also due to Professor Seun K. Kahng, who was there to give me support when I badly needed it. I am also grateful to Dr. Clovis R. Haden for his advice at every step I took during those crucial days.

I am thankful to Professor Gerald Tuma for expressing sympathy and support during those days. My gratitude is also due Professor William R. Kuriger and Professor William T. Huff.

I would like to express my sincere appreciation to all those friends who boosted my moral courage during the most unfavorable time of my career at this institution.

My acknowledgments are due to University Research Council for supporting me financially through their fellowship program.

Lastly I would like to thank my wife, Veera and daughter, Geeta for sharing all the agony and for their sacrifices in spending lonesome weekends and sleepless nights.

DEDICATION

I would like to dedicate this work to my late father, Bikshapathi Pittampalli and to my mother Anasuya.

GLOSSARY

Bus

Set of signal carring lines.

Protocol

A set of rules for operating a communication system.

Baud

A unit of signaling speed equal to the number of discrete conditions or signal events per second.

Buffer

A storage device used to compensate for a difference in the rate of data flow when transmitting data from one device to another.

Direct Memory Access

DMA is a facility that permits I/O transfers directly into or out of memory without passing through the processor's general registers; either performed independently of the processor or on a cyclestealing basis.

Link

A communication path between two nodes.

Interrupt

To stop a running program in such a way that it can be resumed at a later time, and in the meanwhile permit some other action to be performed.

Interface

A piece of hardware used between two pieces of equipment.

Assembly Language

A direct symbolic representation of the binary instructions which can be executed by the processor.

Random Access Memory (RAM)

A traditional name for read/write memory.

Read Only Memory (ROM)

A ROM is a memory which, once "programmed", can only be read by the CPU.

Erasable Programmable Read Only Memory (EPROM)

These are read-only memories programmable by the user which can be re-programmed a number of times.

TABLE OF CONTENTS

Chapter Page

LIST OF ILLUSTRATIONS

ABSTRACT

The study of real-time control of a power system simulator using the interactive graphics is investigated. The distributed processing is employed using a minicomputer and a microcomputer for control and data acquisition tasks involved in the research. A real model power system is interfaced to a microcomputer, a minicomputer and a graphics terminal. The computers duo are assigned two different specific basic tasks on a dynamic basis, which in turn are interfaced through a communication network. The microcomputer services the needs of data acquisition of all the necessary state variables of the power system model. The minicomputer acts in a supervisory status in delegating certain scheduled routines of data acquisition and also in operating the status change conditions of all functional and circuit breaker switches of the simulator. In addition to these tasks the minicomputer also updates the graphic displays representing the simulator on the graphics terminal. The result of this research is a functional power system environment with distributed computer network, facilitating real time study and analysis through interacive graphics.

CHAPTER 1

INTRODUCTION

Electrical power generation, transmission and distribution networks coupled by switching and load center substation make up a system that covers a vast geographical area. These power systems are growing larger and required interconnections are more complex to meet increased society energy requirements. This increased growth in size and sophistication has led electric power utilities and research activities to use digital computers for on-line supervisory control of power system networks. Although computers have been used extensively for operation and control in power industry over the past decade, direct digital control is relatively a recent development. [30] The use of digital computers in such fashion has resulted in large degree of improvement and significant contributions to efficiency, reliability and secure operation of the system.

[1,2,5,7,17] Various attempts have been made in the direction of power system monitoring and control. [33] Martin discusses in his work a simple and inexpensive method of monitoring the steady state behaviour of power systems using inexpensive microprocessors. This method assumes that the voltage and current variables are in steady state. [34]

Deliyannides , See discuss the use of microprocessors in distributed processing for power system control applications. In this work the current spectrum of equipment functional sophistication from large energy management centers to simple supervisory master stations and remote data acquisition and control units is examined to identify areas of profitable use of microprocessors. [23,25] Much work has been done in the studies of automated individual machinery. [35] Pullman, Hogg described a laboratory micro-alternator system, which is a scaled model of a large turbogenerator. [36] Scriber in his paper discusses the implementation of a computer driven supervisory control and data acquisition system. Most of the work described so far has been totally theoretical in nature and remains untested in the power system environment. The real time power system control models such as automatic contingency analysis and avoidance, automatic load dispatching, generation control and unit commitment, and automatic security analysis on network scheduling, remain largely untested since a model test environment does not currently exist other than the actual power system. For power system security reasons these tests are not permitted to be run in the United States utility grids. Limited tests have been conducted, however by nationally owned utilities such as Tai Power and National Swiss Grid. However the results of these tests are kept within the testing organizations and are not available outside. These organizations are not using any closed loop control.

Recent visits by Tai Power and National Swiss Grid and their intention to duplicate the OU Simulator and complete computer control network indicate the key role such a model test environment can play in the future evaluations of power system control schemes.

The research conducted in conjunction with this dissertation consists of conceptual development and verification of open/close-loop functional test environment confined to a scaled power system, through interactive graphics. This work studies the feasibility of providing the functional environment for studies towards closed loop control of a power system in real-time.

[4,8,24] Large number of papers have been written indicating various developments in the direction of data acquisition systems. These differ either in the type of device used for digitization or the cost of the system itself. The following research evolved a unique technique of data acquisition. This technique consists of developing predetermined scheduled routines and displaying these routines on the graphics terminal along with approriate line diagram of the power system. Thus these routines are executable at the touch of a light pen. To summarize, this research establishes a proven model of a functional environment for further research in real-time control of a power system. Thus this research advances the present state of art of control

of power systems, through closed-loop real-time interactive computer graphics.

The system proposed consists of mini and microcomputers as processing elements performing independent control and data acquisition tasks.

The system developed consists of a power system simulator by Hampden Engineering Corporation, PDP 11T34, a minicomputer with graphics capability by Digital Equipment Corporation and a SBC 80/10A microcomputer by Intel Corporation.

The concepts investigated in this research include (a) Representation of simulator model in single line graphic displays along with associated state variables. (b) Transformation of absolute values of variables into scaled units compatible with data acquisition system employed. (c) Development of various interface networks. (d) Development of software structure involving various data acquisition algorithms, supervisory, communication and simulator controller interface routines with well defined protocol command/data formats.

The Interactive graphics controlled power system control proposed and demonstrated in this dissertation work advances the state of art to make a future closed loop power system control a reality. The distributed processing outlined in this work illustrates the feasibility of using inexpensive microcomputer systems for the data acquisition and control at various lower levels of a power system. The ultimate result of work is a unique test environment for use in the development of real-time power system control schemes.

Even though a two node distributed processing network has been developed, it is possible to extend the concept to a multi-node network limited by cost only.

CHAPTER 2

BACKGROUND

Contingencies, whether scheduled or emergency, which involve the loss of any essential component of the power system, such as a transmission line or generator or load result in large degree network disturbances eventually leading to a possible failure of a larger portion or the entire network. During unscheduled system contingencies, the automatic controls operate to protect the system from overload and damage. The system dispatcher monitors and makes decisions concerning load shedding, generation scheduling or total system rescheduling, attempting to keep as much of the system supplying power as possible. [8] Attempts have been made to propose schemes that could control primary functions such as voltages, speeds and loads. Such schemes offer the advantages of integration and compatibility with the devices at various levels of system function. This practical device must be made to meet the security and reliability requirements of power utilities. The problem encountered by operations and system planning, is the inability to do system testing and checking of closed-loop algorithms and real-time on-line analysis tools on the actual power systems. A typical line diagram of a power system is shown in figure 1. In the event that these algorithms fail, then often catastrophic system failures occur which are detrimental to the power system.

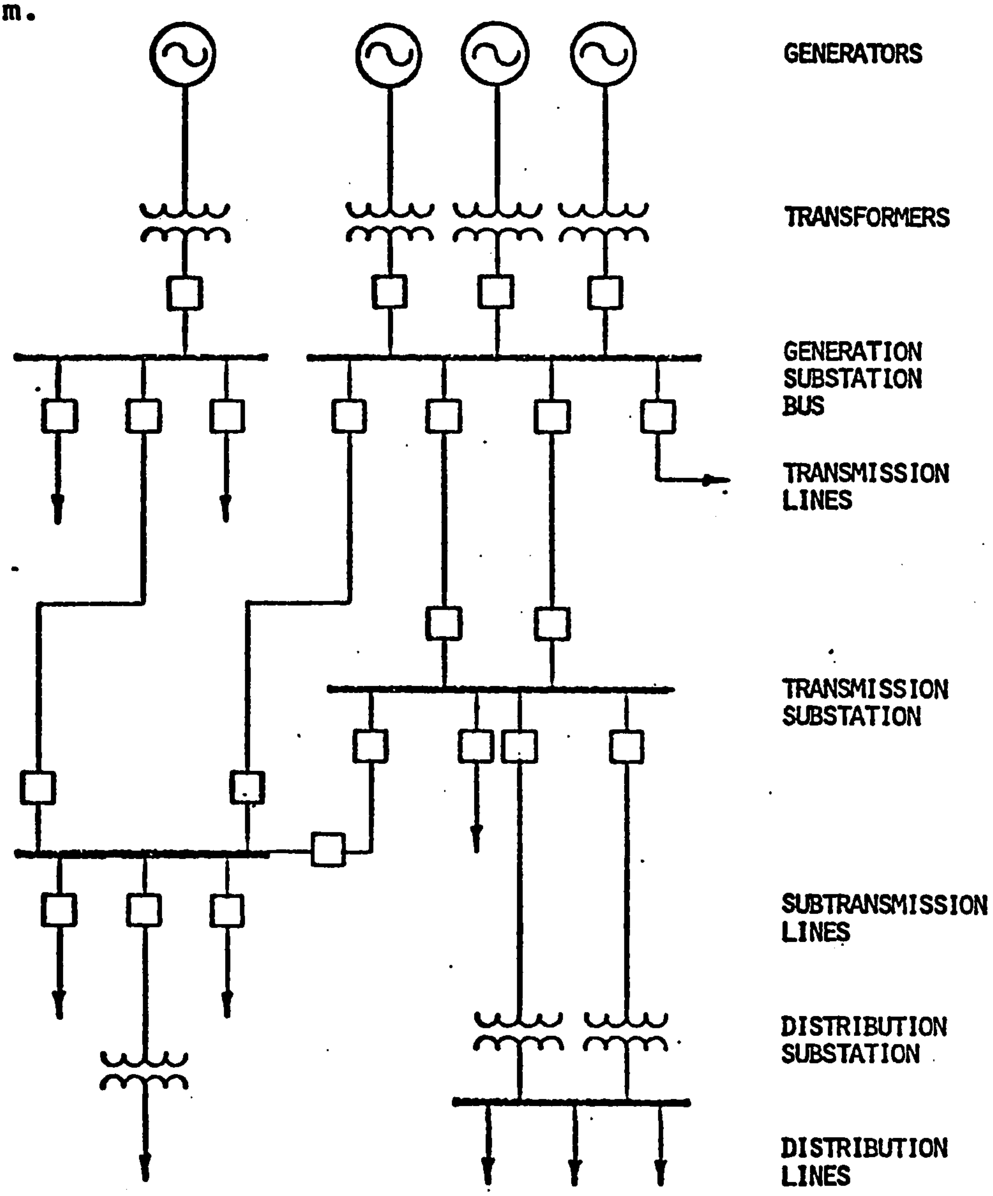


Figure 1. Line Diagram of a Typical Power System.

[6] A study has been done on on-line digital controller

dedicated to the closed loop real-time control of such control functions as terminal voltage, speed and load control. At lowest level of control the controller assumes direct control of main generator functions via a digital-analog interface, passes data to computers at higher levels of control for evaluation of overall system performance and also receives inputs from these levels, such as changes in set points, and interacts directly with plant operator. [9] The real-time control system is built around the computer interrupt system and is made up of a system executive to coordinate the execution of many concurrent priority structured program modules. Economic evaluations must lead to an acceptable cost if acceptance by the utility industry is to be expected. While exact comparisons are impossible, relative cost estimates can be made, and even when one considers the cost of the peripheral equipment, the system cost will fall below the estimated cost per line terminal currently required for protection. [5] Since functions of automation go beyond the point of protection of system, the actual cost per function should prove approximately 20% less than the present systems.

The prime objective of the power system dispatcher is to make electric energy available, in usable form, whenever demanded by the customer. In order to accomplish this task the operator can alter various network elements. These include generation sources, the transmission lines and the distribution of the load. If the power system has been correctly planned, it will be designed to meet the needs of a specific area, since the capital costs of surplus generation equipment is prohibitive. However, a secure power system design requires that the bulk power system function during the scheduled or unscheduled outage of one or more major system components.

The power system operator does have spare equipment that can be used in accordance with his own ideas. Economics of power system operations enters the problem and the system operator is obliged to meet the power system commitments in most economical way. Unfortunately, the cheapest way to operate the power system may not be the most economical, because the probable cost of system outages must be balanced against increased operational cost to make power system more secure against component failure.

[13] The occurrence of abnormal conditions is the most normal state for a large power system. Abnormal means that one or more of power system components are not available for service. The component may not be available because of equipment failure, or equipment out for maintenance, or because of new construction.

Except during extreme emergency conditions, the power system operator will have multiple choices as to how the system can be operated. In such a case, a fast and accurate and efficient decision can be taken by a pre-programmed computer compared to a human operator. Thus there arises a need for the introduction of computer operations in the maintenance of the power systems.

Case study for the necessity of automation of the power system.

NEW YORK BLACKOUT:

[13] On July 13, 1977, an intense storm accompanied by heavy winds and rain moved southeast across Westchester county in the state of New York. At aprroximately 8:37p.m., lighting struck the towers on the section of the right-of-way between two power transmission stations of the Consolidated Edison system, which serves the boroughs of Manhattan, Bronx, Brooklyn, Queens and Staten Island and most of Westchester county. Less than 20 minutes later, other towers were struck. These two lightning strokes initiated a chain of events that led to the shutdown of the electrical supply for the city of New York.

The conditions that led to the eventual blackout of New York city bring to light the inability of the equipment and personnel to respond fully to fast developing emergencies when required to operate under severe generation and transmission constraints. Due to strong interconnected system that is essential to meet the reliability of supply within the city, all of the faults that occurred as a result of first initial lightning strokes led to large surge currents that were sensed by a large number of protective devices. The vast majority of these devices responded to these surges properly. The protection equipment malfunctions that did occur provide a challenge to improve the testing of installed equipment of the system.

Major elements that led to the failure of the system to withstand the impact of the storm were the following:

1. The inability of the system as operated to withstand the loss of all its major transmission ties to the North, a contingency beyond Con Edison's design criterion.

2.The inability of generation equipment in the city to meet maximum capability as reported to the system operator.

3.The absence, at the energy control center, of a clear display of the change in status of key bulk power data to the operator.

The accumulated effect of these factors was such that the system operator, based upon his prior experience and training could not fully respond to an emergency of the proportions associated with the loss of the entire transmission support from the north.

These facts indicate a real time control of a power system with a possible power to an operator is inevitably required. However, the human operator can be made to possess the provision to deactivate any of computer decisions or actions at any time of operation. It is with this in mind that this project was originated. The concept of bringing the entire system to well within reachable limits of operator for control of the entire system is achieved in this work through the representation of the system by line diagrams on the graphics terminal. In this way a close on-line interactive approach is made possible. The computer can check enormous amount of data regarding certain contingency and relay the outcome of laborious computations to the operator, thus relieving him from the studying of this data, as a basis for any of his decisions. A modified form of Hampden's Power Simulator is used as the base of the system.

[12] Interactive computer graphics involves two-way communication between computer and user. The computer, upon receiving signals from the input device, can modify the displayed picture appropriately. The user can give a series of commands, each one generating a graphical response from the computer. In this way he maintains a conversation, or dialogue with the computer. With the ability to interact with the computer the operator can quickly correct a design error and see a revised picture or data. In this a way there is always a faster response possible from the operator for unanticipated result from the system. Thus interactive graphics improves the response time of communication between the user and the computer in both directions.

The task of control and monitoring of an electrical power system through physical components is very extensive in nature. The size of the system prohibits its representation confined to the limits of a small room. The ability of computer to store the relevant data which can be retrieved at any point of time makes it possible for the control of the entire system through a graphics terminal. The graphics terminal provides an instantaneous access for any interaction intended by the operator with the system. The most important aspect of the system control depends on the part of data acquisition system. The data is required by the controller before any action is taken. With human operator gathering data from either a central computer or other means , one can see the correlation of the data for the present contingency is, in fact, a unsurmountable task. In this project the data is acquired from all desired points of interest and presented in the most readable form on the graphics terminal for any decision.

The data acquisition of these state variables of the power system is done by a microcomputer system, with the help of a real-time interface subsystem. The microcomputer has a preassigned task of collecting data from these variables and communicating the system status to the central control computer which takes the supervisory role in the control of the entire system.

[10] Many improved performances are being attributed to distributed processing. These include high system performance, fast response, high throughput, high availibility,

high reliability, ease of modular incremental growth, easy expansion in both capacity and function. The definition of distributed processing include the following: A multiplicity of general-purpose resourse components, including both physical and logical resourses, that can be assigned to specific tasks on a dynamic basis. A high level operating system unifies and integrates the control of distributed components. Individual processes each have their own local level operating system and these may be unique. A physical distribution of these physical and logical components of the system interact through a communication network. The operation of the distributed logical resources is to a greater extent autonomous.

Some speak of a system that has any one of these components distributed as being "Distributed data processing system". A proper definition must cover the concepts under which the distributed components interact. There is certainly no distribution of processing functions if there is no distribution of processing hardware and conversely a system that distributes hardware without distribution processing is difficult to imagine.

CHAPTER 3

HARDWARE DESCRIPTION

This chapter will outline the hardware structure of the research project. The simulator and the computers will be introduced briefly in the beginning of the chapter followed by the introduction on DL-11W and DRS-11 peripheral devices of minicomputer. In addition, this chapter also includes the hardware concepts of author's contribution of the research project.

POWER SYSTEM SIMULATOR:-

The power system simulator is an excellent tool for education as well as research. In reality the power systems are very large and expensive. A physical realization on a small scale enables the user to see the basics and perform experiments that would have been prohibitive on real systems.

The simulator (Figure 2) is a small power system consisting of scaled generators, transmission line system, a subtransmission system, substations, and loads. The generation is in the form of three DC motor/synchronous generator sets of 1KW, 1KW and 3KW capacity. Additional capacity may be supllied to the simulator through the interconnnection.

The high voltage(6ØØV) transmission connècts the generation

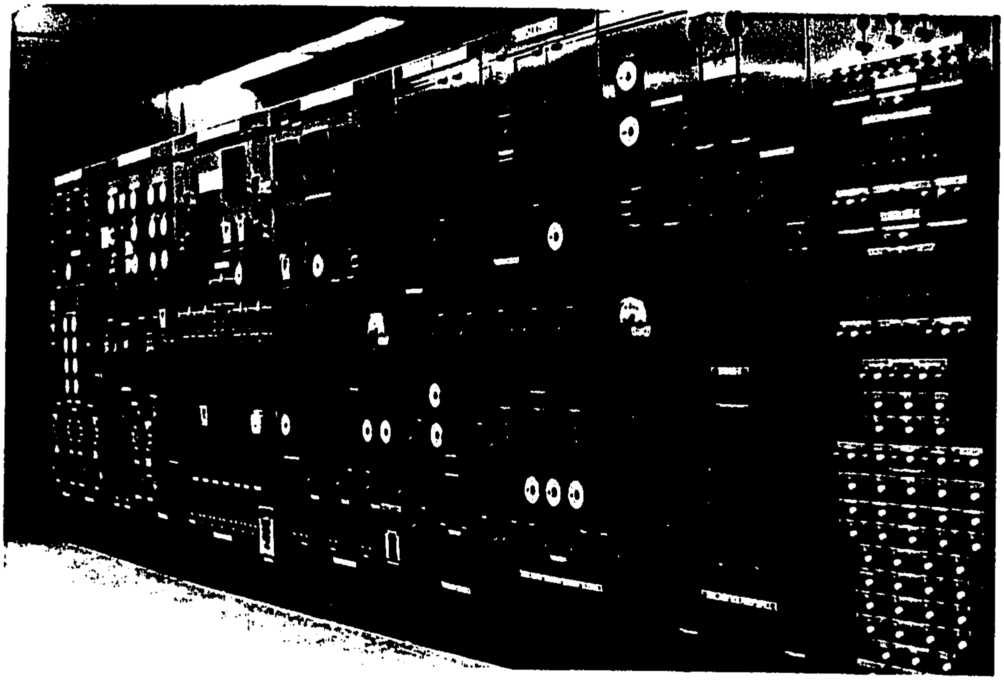


Figure 2. Photograph of Power Simulator

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 circuit controls switches are accompanied by special modifications, thus allowing them to be operated by the computer commands. The relays provided in the system are 12ØVAC operated.

COMPUTER SYSTEMS:-

The system described here, employs two computers for its operation. They are PDP-11T34 Minicomputer and SBC 8Ø/1ØA Microcomputer systems. In addition to these central çomputers, we have associated peripheral devices to both of these systems. The interconnection of peripheral devices with PDP-11 processor are shown in figure 3.

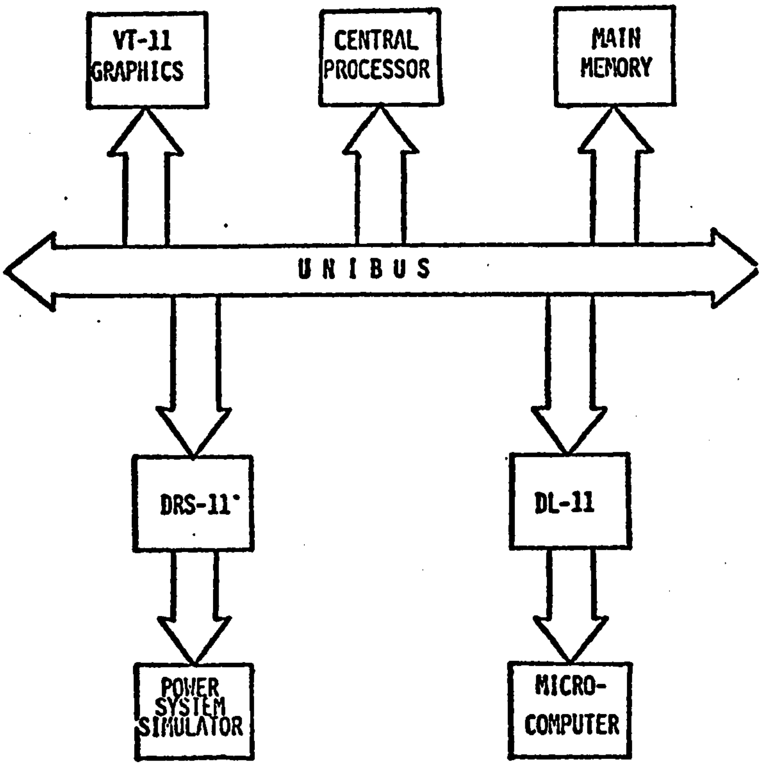


Figure 3. Unibus Interface Structure

MINICOMPUTER SYSTEM:-

The minicomputer employed in this research is a DEC (Digital Equipment Corporation) PDP 11T34 shown in figure 4. The system supports off-line storage in the form of two disk drive units (RKØ5 Disk Cassettes). The CPU is connected to the peripherals by a UNIBUS system. UNIBUS is the name given to the single bus structure of the PDP-11. The processor, memory and all peripheral devices share the same high-speed bus. [38] The maximum transfer rate on the Unibus is one 16-bit word for every 75Ø nanoseconds or 1.3 million 16-bit words per second.

Figure 4. Photograph of Minicomputer System

The Unibus enables the processor to view peripheral devices as active memory locations which perform special functions. Peripherals can thus be addressed as memory. In other words, memory reference instructions can be operated directly on control, status, or data registers in peripheral devices. Data transfers from input to output devices can bypass the processor completely. The memory contains 64K words of semiconductor memory, of which less than 32K is needed to perform the task described in this work.

SERIAL LINE UNIT (DL-11W):-

The Serial line unit (DL-11W) is an asynchronous line interface which can handle full or half duplex communication between a wide variety of serial communication channels and a PDP-11 computer. With a DL11-W interface, a PDP-11 computer can communicate with a local terminal such as a console teleprinter, a remote terminal via data sets or private or public switched telephone facilities, or with another local or remote computer. The interface between DL11-W and the UNIBUS is shown in figure 5.

The DL-11W is a character buffered communications interface designed to assemble or disassemble the serial information required by a communication device for parallel transfer to or from the PDP-11 Unibus. It also provides the logic and buffer registers necessary for program-controlled transfer of data between a PDP-11 system requiring parallel data and an external device requiring serial data. The DL11-W provides the flexibility needed to handle a variety of terminals. For example, the user can use a DL11-W as a Teletype controller, or in conjunction with another serial line interface. It can be also used as a communication link between two processor systems.

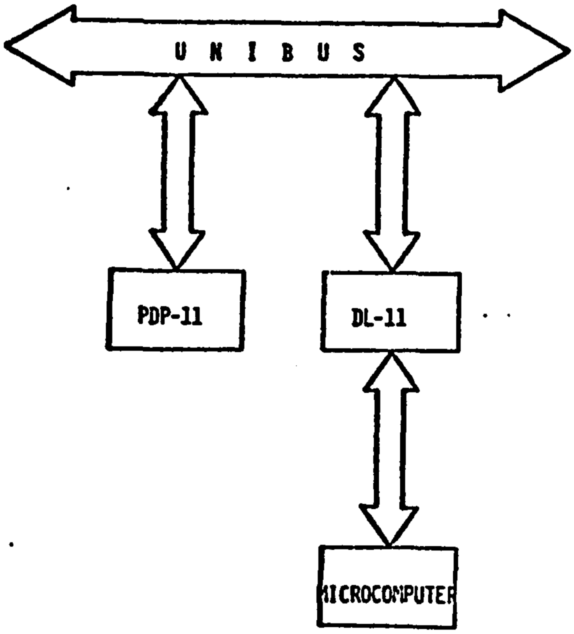


Figure 5. Communication Interface Between Mini and Microcomputers.

INPUT-OUTPUT SYSTEM (DRS-11):-

The DRS-11 is an output system to control up to 48 digital outputs by the computer. Each output module has in addition to 48 buffered output lines, a RC filtered interrupt input with either TTL or open collector characteristics. The interface of DRS-11 with the UNIBUS is shown in figure 6.

Since the utilization of DRS-11 is made through assembly language it is necessary to define device address. The ten position DIP switch provided on the card is used to set the device register address. Positions 1-1Ø correspond to address bits 12-Ø3 on the DRS-11. To select the desired address, the switch is put in the OFF position to signify a ONE in that bit position. The device address in this work is set at 16ØØ3Ø.

The vector address of this module is selected with the eight position DIP switch.

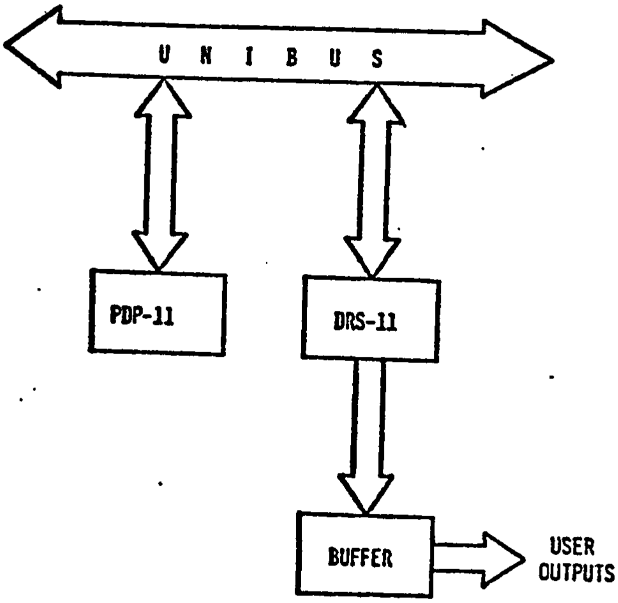


Figure 6. Output System (DRS-11).

To obtain the desired vector address, the switch is put in the ON position to signify a ONE for that bit. The vector address set for this module is 3ØØ. Note that switch positions 2-8 correspond to data bits Ø2-Ø8 respectively. The corresponding device address for second DRS-11 unit is 16ØØ4Ø and vector address is set at 32Ø.

GRAPHICS TERMINAL (VT-11):-

[16] The VT-11 graphics terminal is a single color, variable intensity, random position scan CRT terminal for real time graphics display. The screen is 17" diagonal and defines 1Ø23 coordinate points in both x and y axis. The VT-11 Graphics Display Processor (GDP) is a high performance display processing unit that can operate as a peripheral to any PDP-11 series computer. It "sits" on the Unibus like any other peripheral (Figure 7), and can be addressed by the PDP-11 CPU. The VT-11 GDP is a Direct Memory Accessing (DMA) device, and can, if granted control of the Unibus by the PDP-11, fetch its display program independently of the central processor. This processing unit of the VT-11 reads the display buffer located in the main memory of the PDP-11 and displays an image on the screen in accordance with the instructions in the buffer. In the meantime, new instructions will be placed in the buffer by PDP-11 and on the next read by the VT-11 , a new image will appear on the screen. This process is known as refreshing the screen rather than raster

screen. The more data in the display buffer the longer the collection and transportation time to the screen and this can cause problems of flickering. A light pen is provided with the graphics terminal for user interaction. The

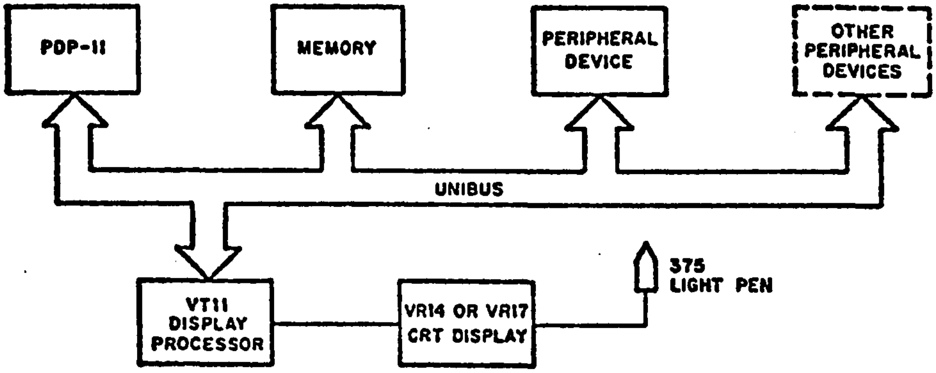


Figure 7. Block Diagram of Graphics Terminal.

display buffer must reside in the lower 28K of memory. The buffer must be set up by software as an initialization process. The VT-11 is capable of issuing interrupts to CPU when it detects the light pen is being pointed to the sensitive area of the screen.

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, an interrupt is communicated to the processor. Then the processor remembers the beam position and subpicture number it was tracing when the light pen hit occurred. Global flags, positional data, and sub-picture number it was tracing are then passed to CPU. This information is used by the program for its changes in the display buffer. Thus the graphics display is changed with the light pen interaction.

MICROCOMPUTER SYSTEM:-

The SBC 8Ø/1ØA is a complete microcomputer system on a single card(Figure 8). The CPU, system clock, read/write memory, non-volatile read-only-memory, I/O ports and drivers, serial communication interface, bus control logic and drivers all reside on the board. Intel's 8-bit n-channel MOS 8Ø8ØA is the central processor for the SBC 8Ø/1ØA microcomputer. This system also contains 1K 8-bit words of read/write memory using Intel's 81Ø2 static RAMs. Up to 4K of read-only-memory(ROM) may be added in 1K byte increments using Intel's 27Ø8 electrically erasable read only memories(EPROM).

A programmable serial communications interface using Intel's 8251 Universal Synchronous/Asynchronous Receiver/Transmitter (USART) is contained on the board. The mode of operation, data format, control character format,

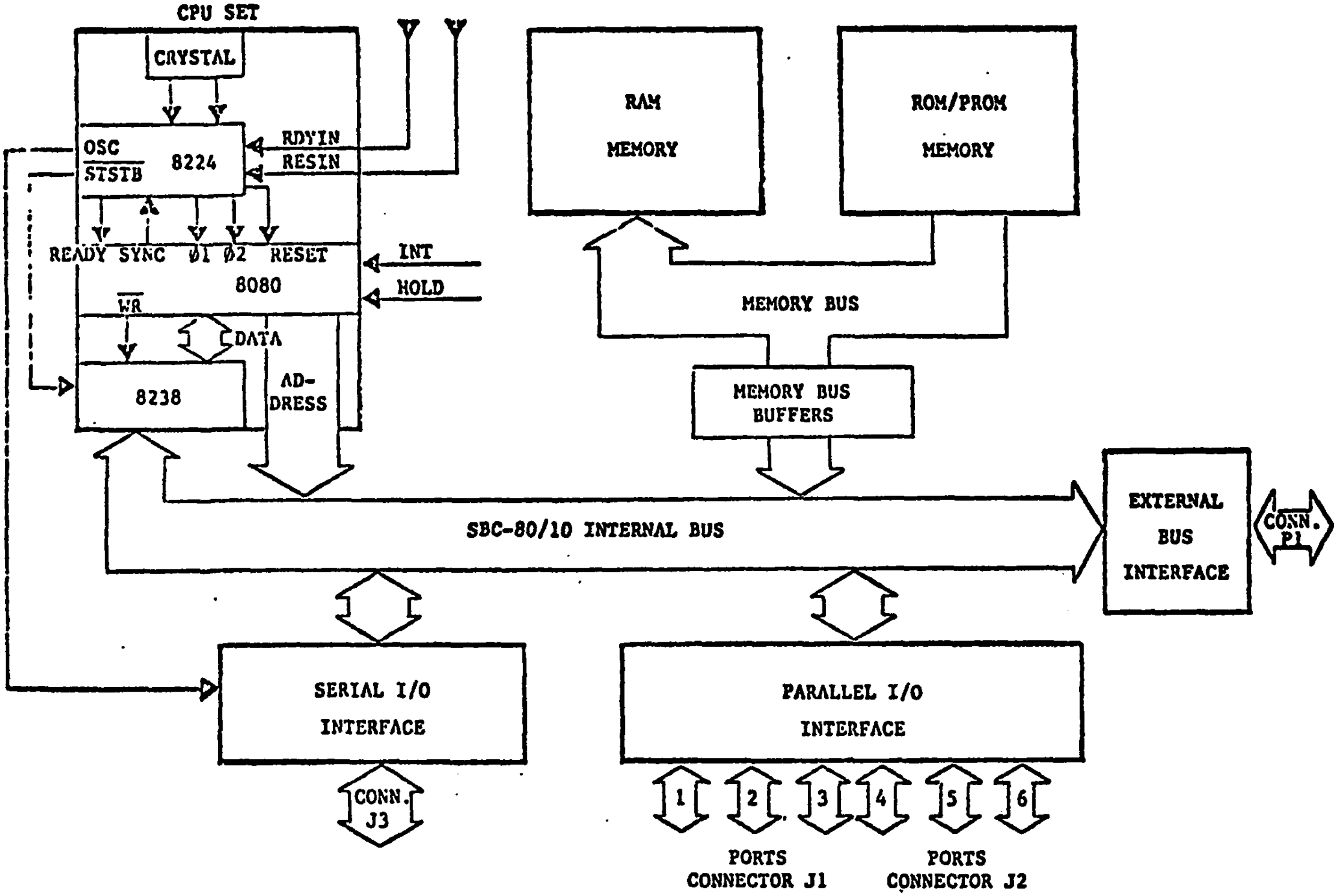


FIGURE 8 FUNCTIONAL BLOCK DIAGRAM

parity, and asynchronous serial transmission rates are all under program control. The 8251 provides full duplex, double buffered transmission and receive capability. Parity, overrun, and framing error detection are all incorporated in the USART. The inclusion of jumper selectable teletype, or RS232C compatible interfaces on the board in conjunction with USART provide a direct interface to a teletype, CRT, RS232-C compatible devices, and asynchronous and synchronous modems. Memory and I/O expansion may be achieved using standard Intel boards. [19] Memory may be expanded to 65,536 bytes by adding user specified combinations of SBC Ø16 16K byte RAM board, SBC4Ø6 6K byte and SBC416 16K byte PROM.

REAL TIME INTERFACE:-

The Real Time Interface (RTI) used in this research is RTI-12ØØ by Analog Devices (Figure 9). This subsystem is compatible for direct interface with SBC 8Ø/1ØA. This can be considered as an integral part of microcomputer. The most basic function of this subsystem is data acquisition. The RTI-12ØØ contains an analog multiplexer that can accept 32 analog channels (single ended), a software programmable gain amplifier, a sample-and-hold amplifier, and a 12 bit analog-to-digital converter. The A/D converter can be preset by the user for any of three voltage input ranges(Ø to +1Ø V, -5 to +5V and -1Ø to +1Ø V). The programmable gain amplifier can be used to increase the input sensitivity of

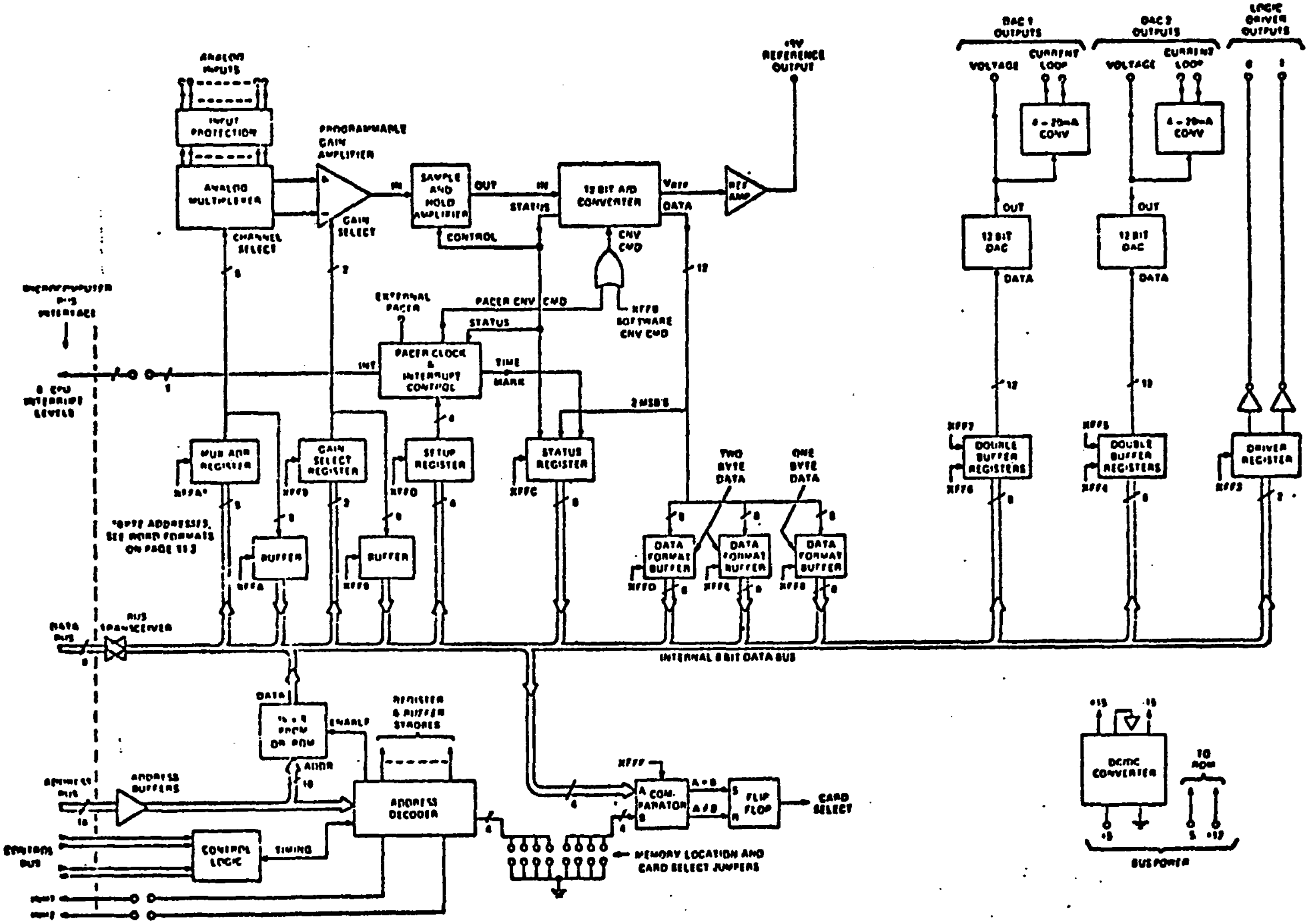


FIGURE 9 RTI-1200 BLOCK DIAGRAM

analog inputs. It also allows the user's program to specify different gains for different channels, or implement automatic gain ranging.

One of the most important features of the RTI-1200 is the way it interfaces to a microcomputer system such as SBC80/10A. This interface is known as memory mapped interface. This is similar to UNIBUS interface mentioned in the minicomputer description. Thus the RTI-1200 appears to a microcomputer as a block of memory locations in the microcomputer's memory address space (Figure 16.1). Data and command information is transmitted to the RTI-1200 via instructions that write into memory, and data and status information is retrieved from the RTI-1200 via instructions that read from the memory.

The RTI-1200 contains a socket which can accommodate a 1024 byte X 8 bit PROM, such as 2708. This could be used as an extension of microcomputer's PROM capability.

In this work the input range of -5 to +5 Volts is selected. Since the analog signals from the simulator are very high compared to the range of voltage set, the following scaling networks are incorporated.

SCALING NETWORKS:-

The scaling networks consist of passive elements including current and voltage transformers. The amplitude measurement is done using a peak detection algorithm. For a better accuracy with nonuniform sinusoidal voltages, the peak detection is done three times in three consecutive cycles. Of these three peak values, the highest number is selected as the desired value. The scaling network used for amplitude measurement is shown in figure 1Ø.

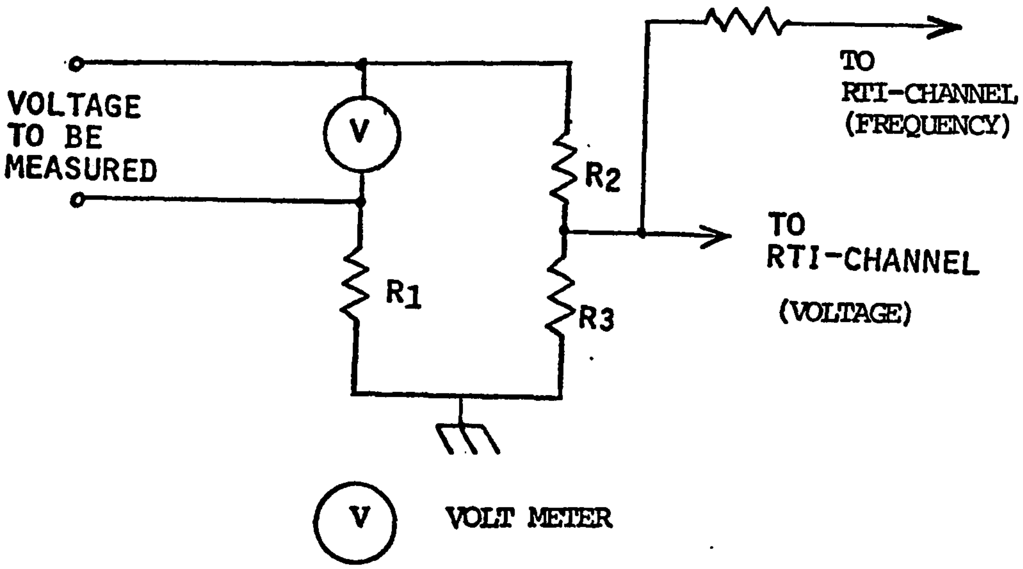
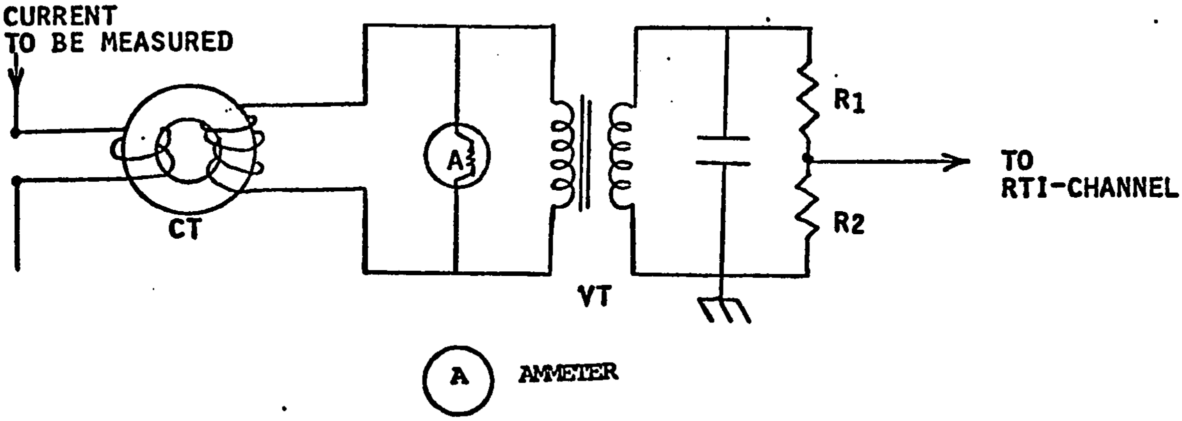


Figure 1Ø. Amplitude Scaling Network

For current measurement the scaling network used is shown in figure 11. This circuit consists of a current transformer(CT) and a voltage transformer(VT) along with resistive and capacitive elements as shown. The current through the secondary of CT, which is proportional to the current to be measured , develops a voltage across the impedance of the ammeter. This voltage is stepped up through a voltage transformer and filtered for high frequency components. A sample of voltage across the capacitor shown is then measured and used to calculate the current through the primary of the current transformer.

In real power systems there may not be a need to keep an analog ammeter as an indicating device, as it is done in this work, while digital reading is measured. In such cases a linear or piece wise linear current transformer is selected for direct conversion of current to voltage and the value of voltage read is interpreted accordingly.



For frequency measurement, the sample voltage is derived from voltage sampling attenuator as shown in figure 1Ø.

THE SIMULATOR INTERFACE CIRCUIT:-

As mentioned earlier the contactors used on this power simulator are all 120VAC operated. Since these contactors are expected to be operated through TTL command signals from computer, there arises a need for an interface network. The Simulator interface circuitry is used to perform level changes from 5 Volt TTL signals available on the DRS-11 board of the PDP-11 to 120 Volts AC, which is required to operate contactors within the simulator. The purpose of the interface , in addition to level conversion , is to provide a high-level of isolation between the system simulator and the computer. In order to accomplish this, a relay/triac arrangement has been utilized. The circuit for one of the channels is shown in figure 12 .

This circuit consists of a transistor switch and a TTL driven relay along with a triac. The transistor switch supplies the sufficient current to turn on the relay, which is not possible through DRS-11, because of the current limit associated with this device. The output from this circuit is connected across the manual switch terminals x,y shown in the figure 12 . While the simulator is being operated in manual mode, the manual switch bypasses the interface circuit completely. A negative going pulse at the base of the transistor switch will turn it on, which in turn, turns the triac on through the TTL driven relay.

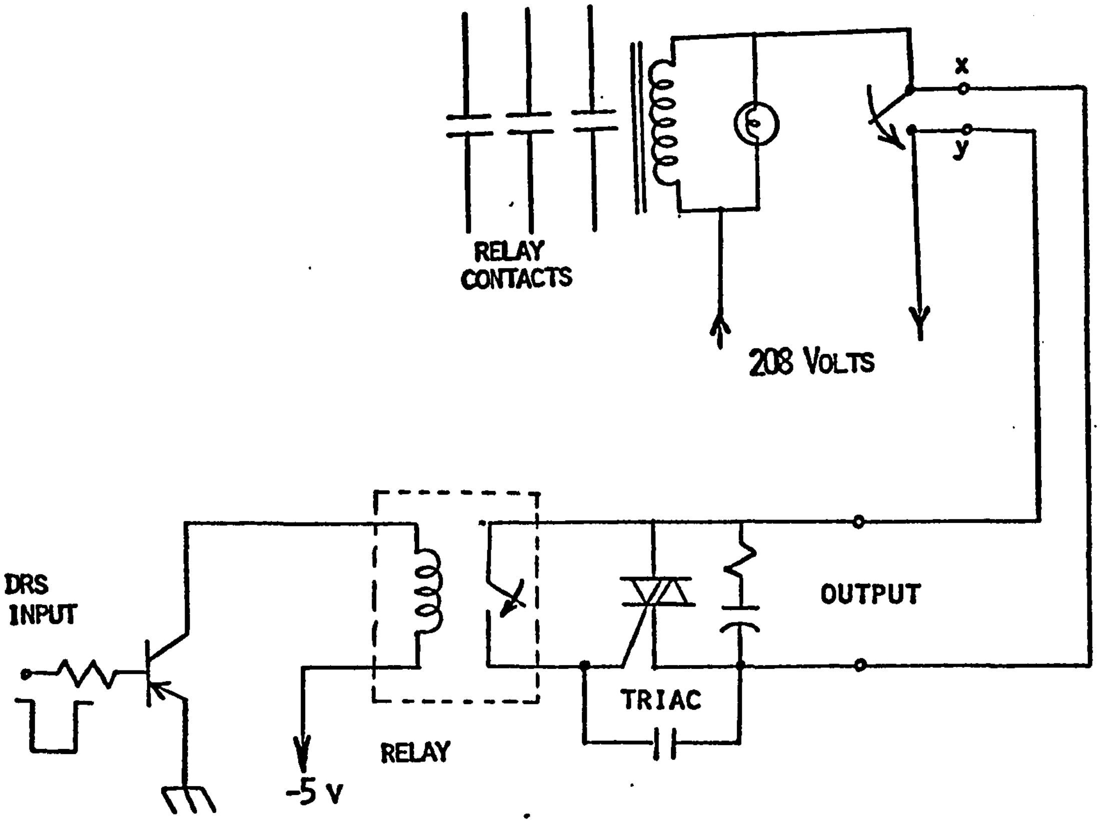


Figure 12. Simulator Interface Circuit.

The contactor relay is then energized through the virtual short circuit of the triac circuit.

COMMUNICATION LINK BETWEEN MINI AND MICROCOMPUTERS:-

The interconnection of the communication link between the minicomputer and the microcomputer is shown in figure 13. The CRT provides a transparent medium to observe the protocol between these two processors. The echo feature of mini provides the information received by it to be displayed on the CRT console while the information transmitted by the micro is displayed since it is connected to the keyboard terminals of the microcomputer system. The numbers indicated in this figure correspond to standard RS-232C pin descriptions.

The interconnection of the entire hardware structure is shown in figure 14.

This block diagram shows details of interconnection of various elements described earlier for a successful operation of a power simulator in a closed loop and open loop conditions. It is seen that a virtual closed loop is formed with the inclusion of the operator in the system. This typical arrangement makes it possible to study the real-time control of the power system simulator through interactive graphics terminal.

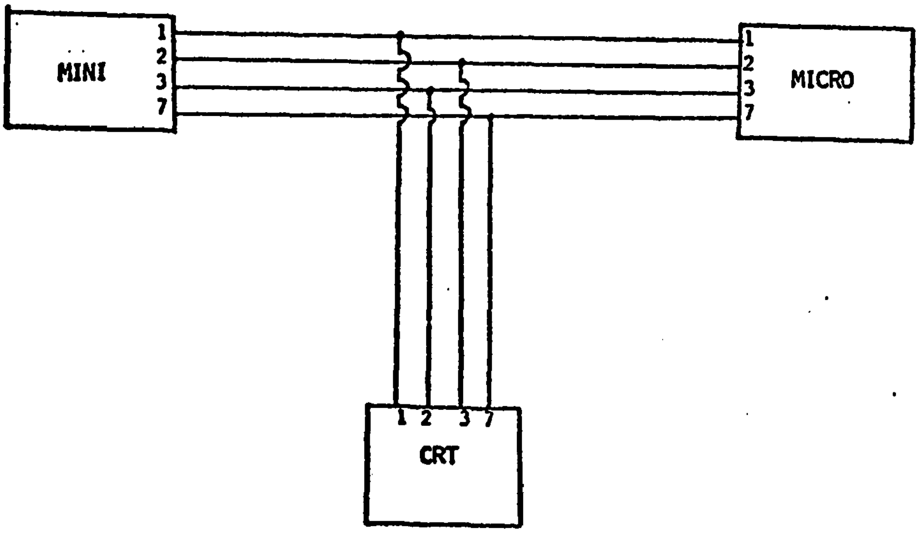


Figure 13. Communication Link Between
Mini and Microcomputers

A special design version of Hampden power simulator is used in this research. To interface this system with SBC 80/10A microcomputer, RTI-1200 Real Time Interface subsystem is used. The compatibility of RTI-1200 with SBC 80/10A is the basic reason for its selection. The input voltage range of

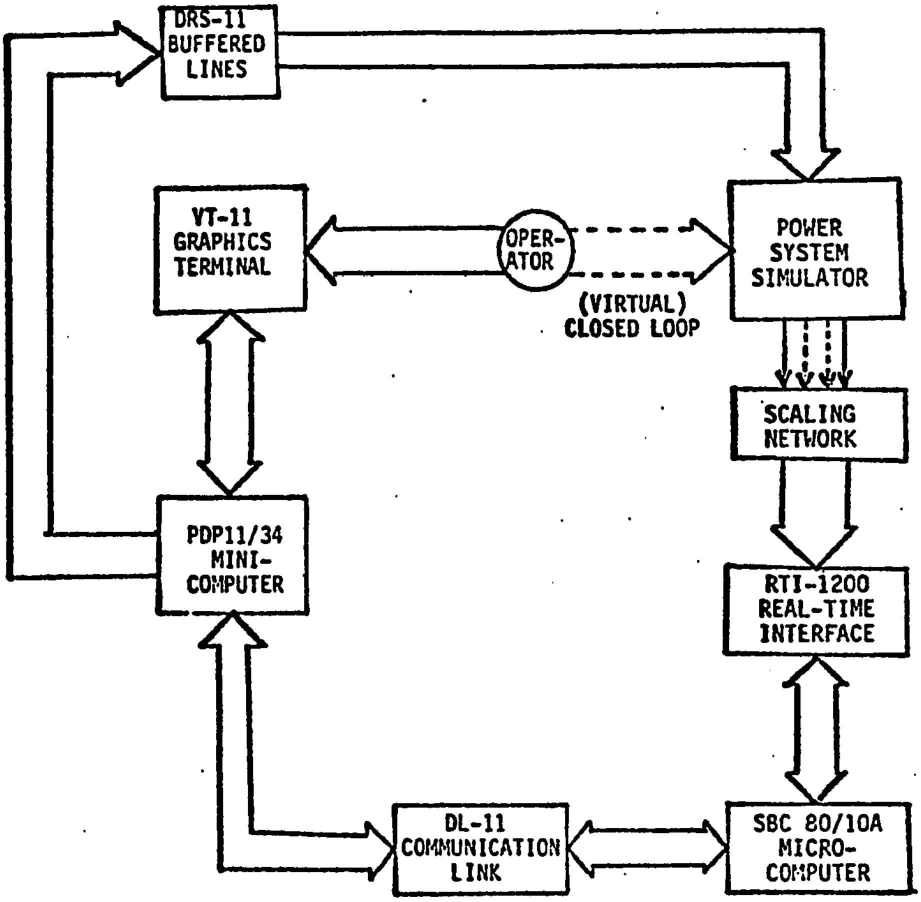


Figure 14. Block Diagram of Hardware Structure.

real time interface module is -5 V to +5 Volts. Since the voltage ranges involved in the power simultor can vary from 0V to 600V rms, an attenuator has been introduced into the circuit. This network is shown as scaling network between the elements Power system simulator and RTI-12ØØ in the block diagram (Figure 15). The microcomputer SBC 8Ø/1ØA is an inexpensive total computer sytem with 1K byte RAM and 4K byte ROM facilities. The availability of MDS-8ØØ, Intel Corporation, development systems prompted the purchase of SBC 8Ø/1ØA , Single Board Computer system, for this project.

The microcomputer SBC 8Ø/1ØA is connected to minicomputer PDP-11 through an asynchronous communication link DL-11W as shown in figure 15. Through this channel the protocol between two processors takes place. This protocol includes, transmission of command from mini to micro and return of data from micro to mini. With the help of software developed the data received by the minicomputer will be processed and converted to actual values existing at the simulator end. These values are then displayed on the graphics terminal indicating the magnitudes of various desired

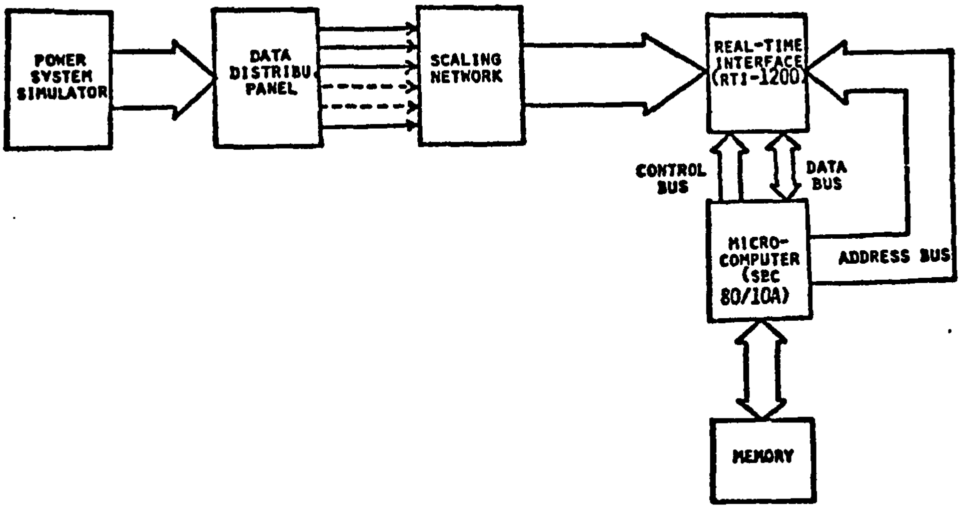


Figure 15. Architecture of Microcomputer Based Data Acquisition System.

variables.

The switching of power simulator contactors is done through a peripheral device called DRS-11. This device is connected to the power simulator through a buffer interface circuit shown in figure 12. This buffer not only serves the purpose of isolating the computer from the simulator, it also converts the TTL level signals from computer to 120VAC signals in order to operate the contactors of the power system. The operator closes the loop of the entire hardware configuration to make it a closed loop system. By interacting through the graphics terminal the user is able to know the status of any desired variable of the power system, and is able to either to delete or include any switching element or load from the system . Thus the user has complete information of the system at any time for decision.

CHAPTER 4

SOFTWARE DESCRIPTION

The chapter will outline the unique software structure of the research project. The operating systems of both mini and micro will be introduced briefly and the relationship of author originated software is discussed. Additionally this chapter outlines the system control program ,display software , mini and micro communication software and protocol details. This chapter also furnishes the discussion on the novel techniques of data acquisition developed in this research.

MINICOMPUTER OPERATING SYSTEM:-

The operating system used with the minicomputer is RSX-11M. [14] RSX-11M is a multiprogramming, real-time operating system. Its fundamental function is to provide the control for sharing system resources among any number of user prepared tasks. Tasks stored on a file-structured volume may be installed into an RSX-11M system and subsequently run by issuing a command to the Monitor Console Routine (MCR). MCR provides the language interface between the operator and the system.

All system tasks in a unmapped system are built to run at specific physical memory address and must be rebuilt if they are to run elsewhere. Whereas in a mapped system, which requires a minimum memory of 24K, the user will not have to rebuild non-privileged tasks if his partition boundaries move. This is true because non-privileged tasks on a mapped system are built to run at a virtual base address of zero, rather than at a physical base address.

Privileged tasks, as with a unmapped system, must always be rebuilt because they are linked to the Executive symbol table file.

This operating system in our case is 64K mapped, which is designed to give PDP-11 the capability of multi-tasking. [15] Multi-tasking can be thought of as a type of time sharing operation. Many tasks can be created and stored on the disk. Each task is assigned a check-pointing priority. Normally one task will have the priority of CPU until the completion of the program. Under a multitasking arrangement the CPU has the power to run one task for a certain period, switch to another task, and return to the first task. This situation might arise when one task is stalled for an I/O operation. The CPU would then switch to another task until some other interrupt occurs.

Of the many features that RSX-11M possesses , the feature of our interest is Shared Region. A Shared Region is a block of data or code that can be shared by any number of tasks.

Shared Regions are useful because 1. They make more efficient use of memory, 2. They provide a way in which two or more tasks can communicate, and 3. They provide a way in which a single copy of a data base or commonly used subroutines can be shared by several tasks.

Consider the case shown in figure 16. Task A and task B communicate through sharing of common memory region for their operation. These two tasks can communicate through this shared region independent of their occurence in time domain.

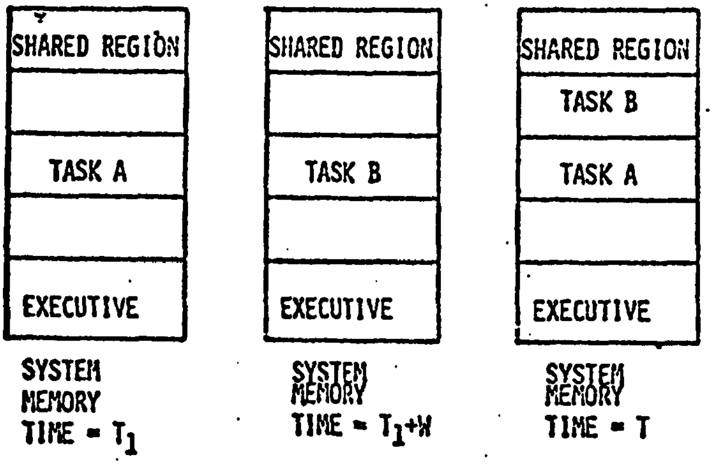


Figure 16. Shared Region

A task can link as many as three shared regions. A shared region has associated with it a task image file and a symbol definition file. When a task links to a shared region the Task Builder uses the symbol definition file of the shared region to establish the linkage between the task and the shared region.

Setting a Common partition/region:-

A common partition is established through a MCR Set command which specifies that the partition type "COM". For example, the following command:

```
>SET /MAIN=DTA:1000:40:COM
```

will set up a main common partition named "DTA", starting at location 100000 that is 4000 bytes long.

Before a task referring a library or common block can be installed, the common block or library must be installed. This is done with the MCR INSTALL command, as shown below:

```
>INS [1,1]DTA/PAR=DTA
```

This command causes the library or common block to be loaded into the partition. Tasks that reference the library will be installed later.

Graphics Terminal:-

The VT-11 display processor can operate as a peripheral on the Unibus just as any other device. It can fetch and act on the data independent of CPU. The memory accessed is called display buffer and must exist within lower 28K memory.

Integral to VT-11 there are special hardware units, like vector generator and special alpha-numeric generator. There are four types of lines which can be drawn on the display. These are, solid, long dash, short dash and dot dash. A ROM is provided which contains digital equivalent data to draw alpha-numeric characters. The text generator has 96 ASCII characters and 31 additional special characters such as Greek letters and math symbols.

The commands that are used to display images on the screen are FORTRAN calls. A graphics library can be built into the operating system, which contains the specific FORTRAN subset dealing with the graphics calls. Thus to draw figures an ordinary FORTRAN program needs to be written. The graphics calls are treated as any other FORTRAN calls and are completely compatible with regular statements.

Some of the graphics calls are listed below:

CALL INIT(X) Informs the CPU the number of words(X) to be used in the display buffer.

CALL APNT(X,Y,l,i,f,t) APNT sets the CRT beam at the point X,Y on th screen, If l,i,f,t are included. They control the light pen (on or off), light intensity, display flash, and line type.

CALL TEXT('xxx') Prints the characters specified,xxx

CALL VECT(X,Y,l,i,f,t) Causes a line to be drawn from present beam position to a point X,Y relative to present position.

CALL RDOT(X,Y) Causes the beam to be displaced X and Y, relative to beam present position.

CALL SUBP(M) Begins the definition of a subpicture identified by a number M. A subpicture is a group of graphic calls forming a user controlled image.

This image can be redrawn in its entirety without continually having to recreate it.

CALL OFF(M) Removes the subpicture M image from the screen.

CALL ESUB Identifies the end of a subpicture definition.

CALL ON(M) Restores the image created in subpicture M.

CALL INIT[(n)] The INIT subroutine sets up a display file for use in performing graphics operations or initializes the file for subsequent use.

CALL NMBR(m,var[,n,format])

The NMBR subroutine creates a special numeric subpicture that can be displayed on the screen in any FORTRAN format and can be updated in "odometer" fashion. The m parameter identifies the tag to be assigned to the subpicture, and var is the FORTRAN variable containing the numeric data to be output on the display screen.

CALL LPEN (IH,IT[,X,Y,IP,IA,IM])

Indicates whether or not a light pen hit has taken place, and returns the following variables:

IH: nonzero if light pen hit has occurred; always Ø or 1 for VT-11.

IT: tag of the subpicture in which the hit occurred.

X,Y: coordinates of the hit.

IP: number of the primitive within the subpicture at which the hit occurred.

IA: array in which the precedents or ancestors of subpicture IT are stored.

IM: always 1 for light pen hit

MICROCOMPUTER OPERATING SYSTEM:-

[19] The microcomputer system does not employ a highly developed software for its operation. It is a command controlled operations supervisor for the single board computer. It provides input and output facilities in the form of I/O drivers for user console devices. The user may access two of I/O system routines from his program by calling them as normal subroutines. The following paragraphs describe the routines available and their respective functions.

CI - Console Input

This routine returns an 8 bit character received from the console device to the caller in the A- Register. The A-Register and the CPU condition codes are affected by this operation. The entry point of this routine is 3FDH.

Example:

```
CI      EQU     3FDH

        ...
        CALL    CI
        STA     DATA
```

...

CO - Console Output

This routine transmits an 8 bit character, passed from the caller in the C- Register, to the console device. The A and C registers, and the CPU condition code registers, are affected by this operation. The entry point of this routine is 3FAH.

Example:

```
                CO      EQU     3FAH
                        ...
                        MVI     C,'*'
                        CALL    CO
                        ...
```

The monitor provides following functional facilities.

A.Display register/memory contents

B.Substitute/change the contents of memory locations

C.Insert the data/program in hex starting at a certain location.

D.Go to start a loaded program from a defined starting address.

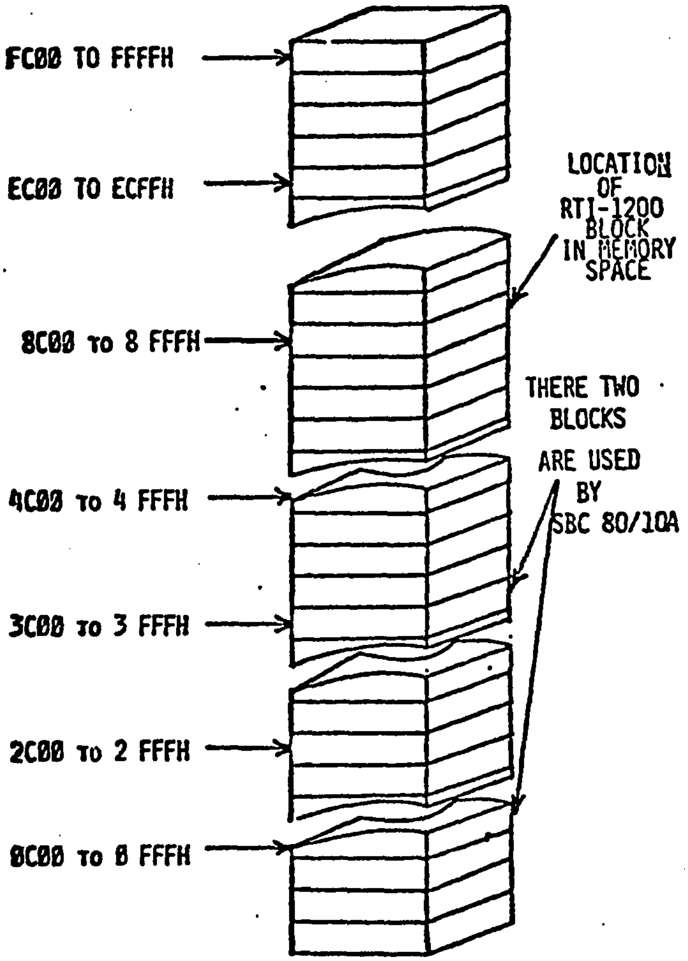
The SBC 8Ø/1ØA monitor is a command controlled operations supervisor. These basic functions are necessary for changing contents of some locations in order to either increase or decrease values of some constants for satisfactory operation of the program. For example, the scan interval, which will be introduced later in this chapter, can be adjusted by changing the content value at the location 3CA1H location. It is also possible to delete this monitor from the system for the basic operation of the entire system but does not provide any additional facilities other than providing a socket for an additional 1K of EPROM in case of future need.

DATA ACQUISITION PROGRAM:-

This section discusses conceptual development of unique sampling schedules of data acquisition. This program serves the basic functions of data acquisition and communication with the minicomputer. These two functions are interwoven in this program. The occurrence of the functional activity depends on the timeframe of the state of the system operation. The Real Time Interface subsystem employed has an analog input multiplexer, a programmable gain amplifier, a sample and hold amplifier, and a 12-bit A/D converter. The RTI-1200 provides 32 single ended or 16 differential ended inputs. The most important feature of this subsystem is that it employs memory mapped interface with the SBC80/10A microcomputer. The SBC 80/10A can address 65,536 bytes of memory, which can be viewed as 64 blocks of 1024 bytes of each. The RTI-1200 can be inserted to occupy one of 14 selected blocks. The possible 14 blocks are spread throughout the memory space ,so that user can select any one of these 14 blocks. In this work the RTI-1200 is inserted in block 8, indicating the starting address of the card is 8FF0H. These top 16 addresses (ie.the highest numbered) in the 1K block occupied by the RTI are devoted to the data and control functions of the RTI-1200. These addresses containing RTI-1200 control, data, and status information are referenced in assembly language via labels. The labels and their corresponding addresses are shown below:

LABEL	EQUATE	ADDRESS
-----	------	-------
CRDSEL	EQU	8FFF H
ADCHI	EQU	8FFE H
ADCLO	EQU	8FFD H
STATUS	EQU	8FFC H
CNVCMD	EQU	8FFB H
MUXADR	EQU	8FFA H
GNSEL	EQU	8FF9 H
ADC8	EQU	8FF8 H
SETUP	EQU	8FF0 H

The bottom 1008 addresses are available for the use of the operator. The memory map of RTI-12ØØ is shown below (Figure 16.1).



NOTE: EACH BLOCK CONTAINING 1024 BYTES.

Figure 16.1 Memory Map of RTI-12ØØ

Since the RTI interfaces as the memory, any of the 8Ø8Ø's memory reference instructions can be used. When acquiring data, the desired channel is written into address 8FFAH. The number of selected channel can also be read back, allowing the use of an increment instruction to advance the input multiplexer to the next channel. The desired gain of the programmable gain amplifier is written into address 8FF9H. A convertion is commanded either by a pulse from one of the pacer clocks, or by writing convert command into address 8FFBH. The end of conversion can be determined by checking the End Of Conversion(EOC) bit in the status word, or by directing the A/D conveter's EOC signal to trigger an interrupt. The A/D converter's output data can be read as 12 bit data in two byte format at addresses 8FFD and 8FFEH. This can be performed by a single LHLD instruction. If only 8 bit data is required, as it is in the present case, the 8 most significant bits can be read as a single byte at the location 8FF8H. The uniques sampling schedules are outlined below:

COMMAND	ANALOG CHANNELS (CHANNEL #S)
A	Gen.1 Voltage (1)
B	Gen.2 Voltage (3)
C	Continue previous command
D	Gen.3 Voltage (5)
E	Inter Tie Voltage (7)
F	Gen.1 Frequency (17)

G	Gen.2 Frequency (18)
H	Gen.3 Frequency (19)
I	Inter Tie Frequency (20)
J	Gen.1 Phase w.r.t Int. Tie (17,20)
K	Gen.2 Phase w.r.t Int. Tie (18,20)
l	Gen.3 Phase w.r.t Int. Tie (19,20)
M	Gen.1 Current (2)
N	Gen.2 Current (4)
O	Gen.3 Current (6)
P	Gen.1 Voltage (1), Gen.1 Current (2) Gen.1 Frequency (17) and Gen.1 Phase w.r.t Inter Tie (17,20).
Q	Gen.2 Voltage (3), Gen.2 Current (4) Gen.2 Frequency (18) and Gen.2 Phase w.r.t Inter Tie (18,20)
R	Gen.3 Voltage (5), Gen.3 Current(6) Gen.3 Frequency (19) and Gen.3 Phase w.r.t Inter Tie (19,20)
S	Gen.1 Freq. (17), Gen.2 Freq.(18) Gen.3 Frequency (19) and Inter Tie Frequency (20)
T	Gen.1 Voltage (1), Gen.2 Voltage (3) Gen.3 Voltage (5) and Inter Tie Voltage (7)
U	Sub-station 4 Right Bus (9), Station 1 Voltage (10),

	Industrial Regulator Voltage (11), Line B1-3 Voltage (13), Substation-6 Bus Voltage (14), Network system Voltage L-N1 (15)
V	Inter Tie Current (8), Industrial Load current (12), Network system Ccurrent line 1 (16)
W	Same as T and U
X	Same as M, N, O and V
Y	For future expansion
Z	For future expansion

The various stages involved in this program can be explained in detail by the flowchart shown in figure 17.

The execution of sampling program begins with the command from the minicomputer. The command structure consists of a certain defined format and a sequence of characters. The command format is shown in figure 18. This format consists of a leader , the command and a terminator. The character corresponding to the desired command is duplicated in the command structure for verification purposes. Eventhough this seems to be redundant it is important to realize that if the simulator(in other words the power system) and the computer were to be quite apart the command format serves a great deal of verification purpose. The leading character 'p' can be considered as part of leader with only difference that this character follows a time delay during which the

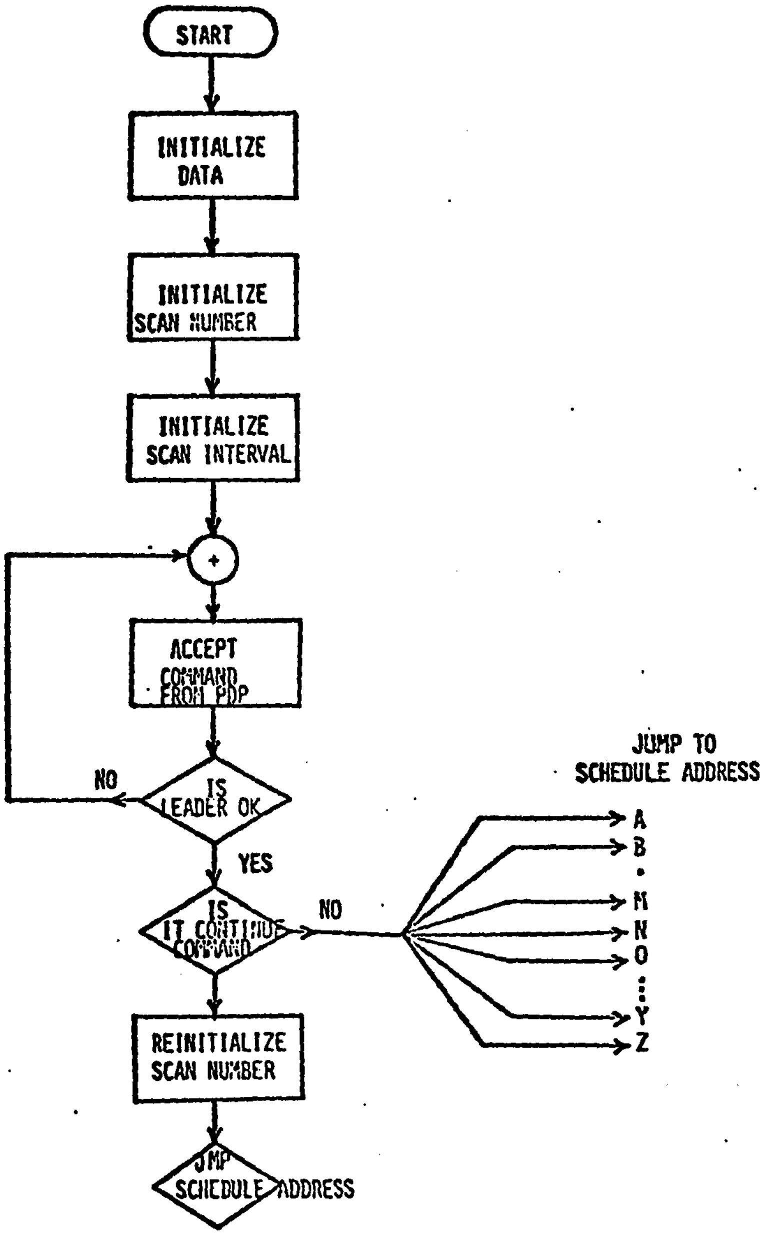


Figure 17. Schedule Branching Routine.

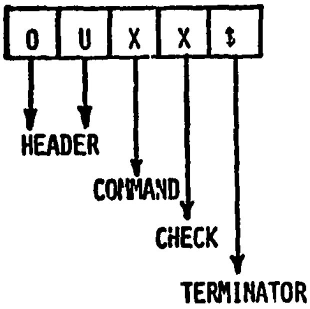
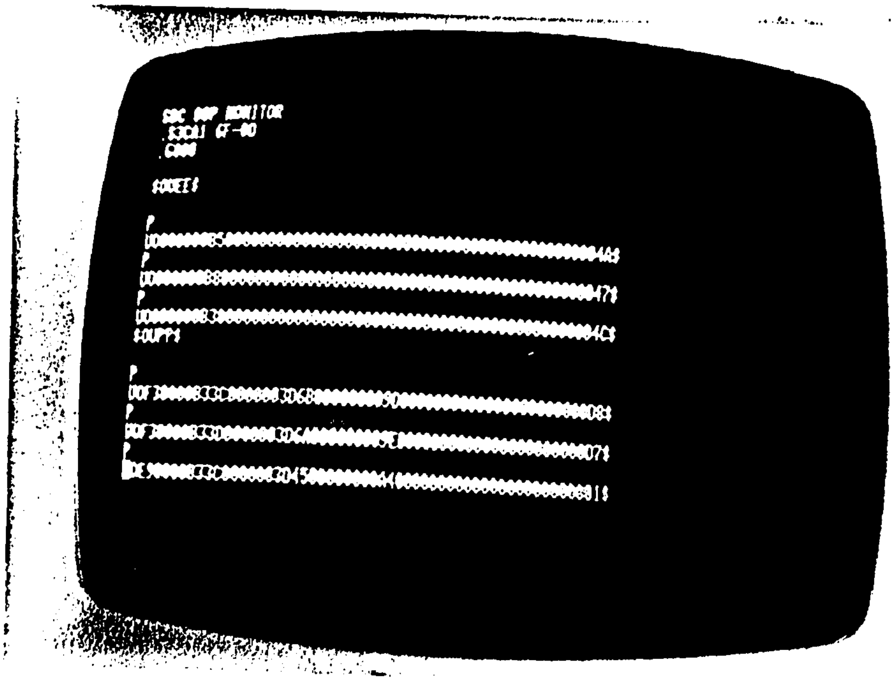


Figure 18. Command Format

minicomputer will be alerted to anticipate the arrival of actual data string from the microcomputer. This leading character is clearly seen in photograph shown in figure 19.



```
[illegible] MONITOR
[illegible]
[illegible]

$OOEE$

P
00[illegible]4A$
P
00[illegible]47$
P
00[illegible]4C$
$OUPP$

P
00F[illegible]08$
P
00F[illegible]07$
P
DE9[illegible]81$
```

Figure 19. Photograph of Protocol.

DATA ACQUSITION ROUTINE:-

The Data acquisition program can be explained in more detail by referring to the flowchart shown in figure 2Ø The program begins with the initialization of the memory locations correspondintg to the power system chosen variables. Then follows the initialization of Scan Number. This scan number determines the number of times a particular schedule is to be repeated before requesting another schedule. At present this number is fixed at 3. The next step is initialization of scan interval. This is a crucial parameter for real time operation of the system. Since the time required by other tasks is definite, the selection of this parameter is started with trial and error method. This value greatly depends on the number of I/O operations the proceesor is requested to do. Lesser the I/O operations smaller the value of this variable is. The scan interval can be changed any time by addressing the location 3CA1H in the random access memory of SBC 8Ø/1ØA.

After selection of required constants and initialization of desired variables the program waits for the command/schedule from the minicomputer. Once the command is received the validity of this command is checked. If it is found to be valid, program branches to corresponding location for the continuation of the service . If the command happens to be a continue command, the initialization process preceeds the execution of prior command again.

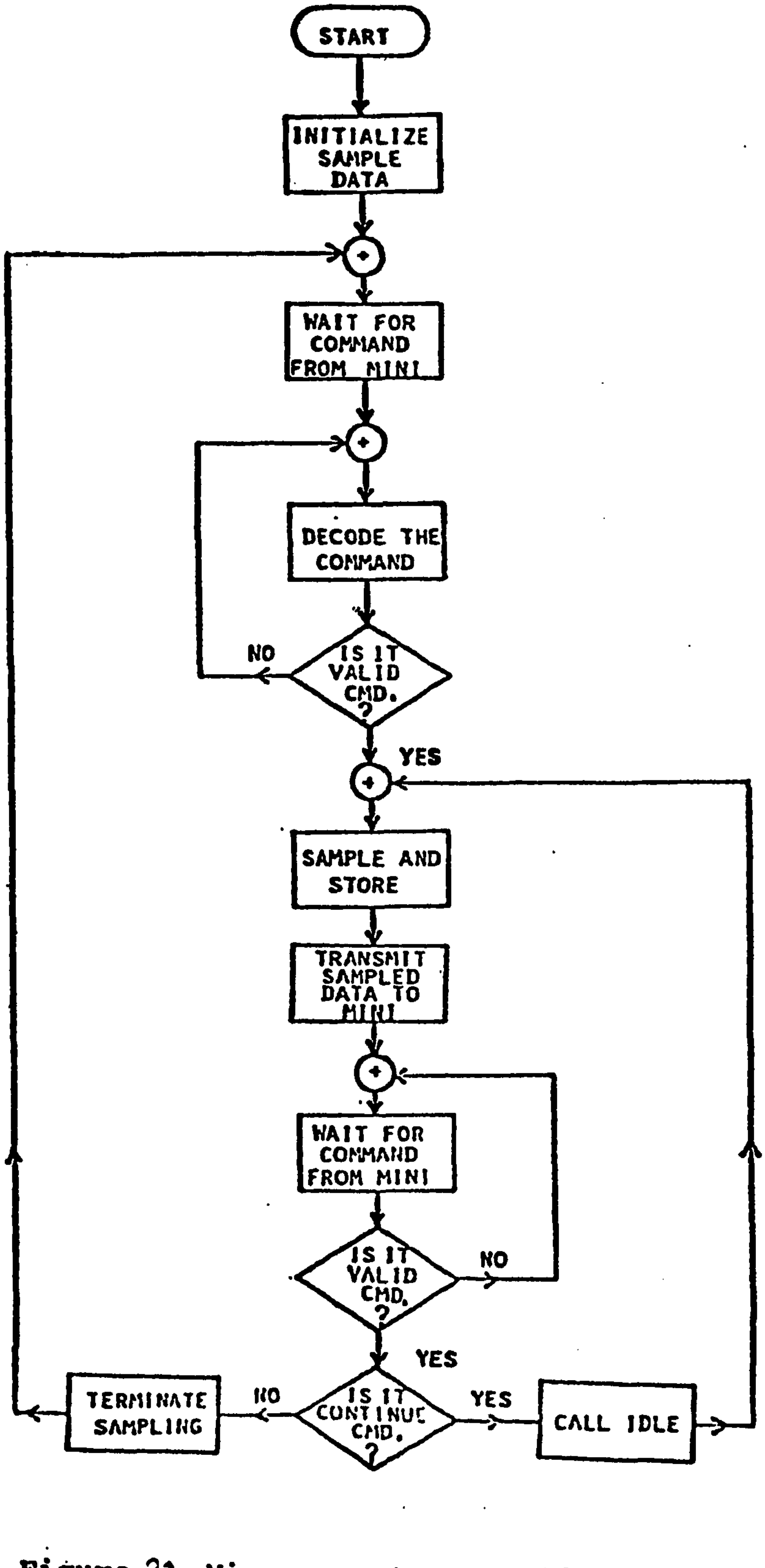


Figure 2^. Microcomputer Sampling routine

The service of a particular routine can be explained with the flowchart shown in figure 21. The variable under

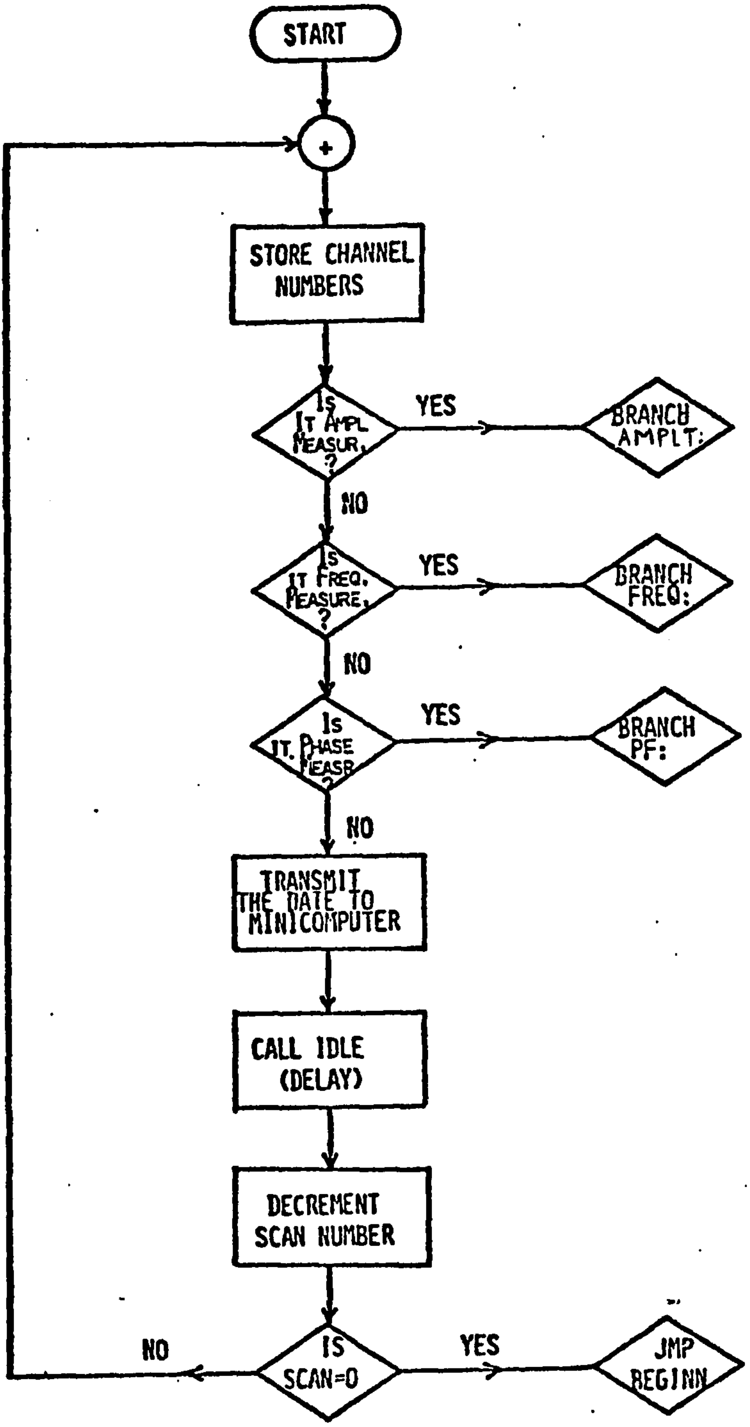


Figure 21. Service Recognition Routine.

request for display is reached through its corresponding channel number. This number is stored at a location from where the sampling routine requests the channel number to be sampled. In order to determine the type of sampling demanded some constant values are stored in intermediate locations, facilitating the necessary action to be taken. In this program hexadecimal number 31 is stored to indicate that the following request is for phase measurement and 32H is stored to indicate that the request is for frequency measurement. Finally a constant number 30H indicates the termination of the process of sampling process, or in other words 30H implies that the request has been serviced.

AMPLITUDE MEASUREMENT:-

Figure 22 shows various steps involved in digitizing an analog signal through RTI-1200 subsytem. This subprogram starts with the selection of the amplifier gain constant followed by turning off the pacer. This pacer controls a real time clock which is an optional package on the real time interface card. Then follows the reading of the channel number from the stored location so that the multiplexer can decide the channel to look for the signal. Now the analog to digital conversion is triggered. It is not necessary to load the accumulator with any particular data prior to executing the STA CNVCMD (triggering the conversion instruction). This byte, when addressed, uses the memory

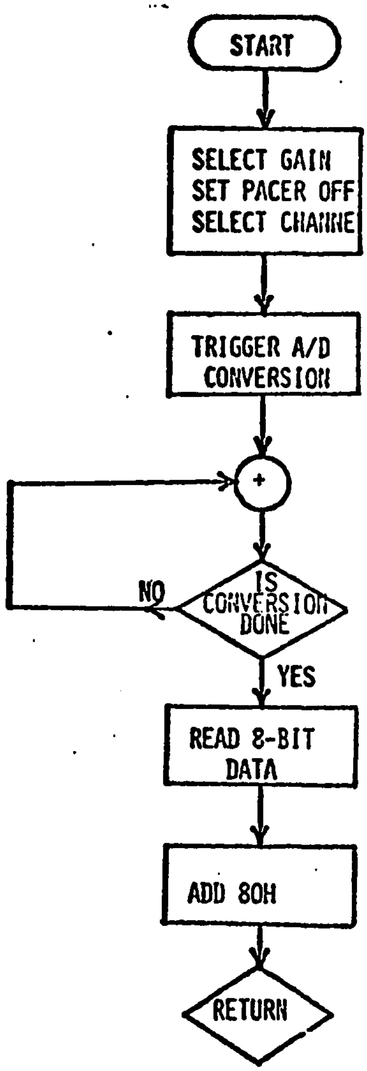


Figure 22. A/D Conversion.

write pulse as a convert command signal, and the actual value of the data has no effect. Note that the card select code is the first parameter to be initialized. All the initializaion refers to a card which has been already selected.

Therefore, in a multicard system, typically same initialization sequence is followed for each card.

[18] Next, although a minimum of 1Ø microseconds must be allowed to elapse between the addressing of a channel and the start of an A/D conversion, the user does not have to be concerned about it. If a convert command is issued within 1Ø microseconds of addressing either the MUXADR or GNSEL bytes, the beginning of an A/D conversion will automatically be delayed until a 1Ø microseconds time period has elapsed. Thus this program is perfectly valid even though less than 1Ø microseconds may have elapsed between the multiplexer channel select instruction and the convert command instruction.

Before reading the ADC data it must be ascertained that the conversion process is over. This is accomplished by checking the End of Conversion (EOC) bit in the status byte. The RLC and JNC instructions are used in a loop to determine when the EOC bit changes from Ø to 1, indicating that the conversion is over and the data is ready. Eventhough 12 bits of data is available as a result of ADC, we are required only 8 bits for the 8 bit microprocessor employed. This 8 bit data is read at the address ADC8.

[18] Note that the time for the instructions between the convert command and data read instructions must be at least 25 microseconds unless the convert command is immediately preceeded by a gain select or multiplexer channel

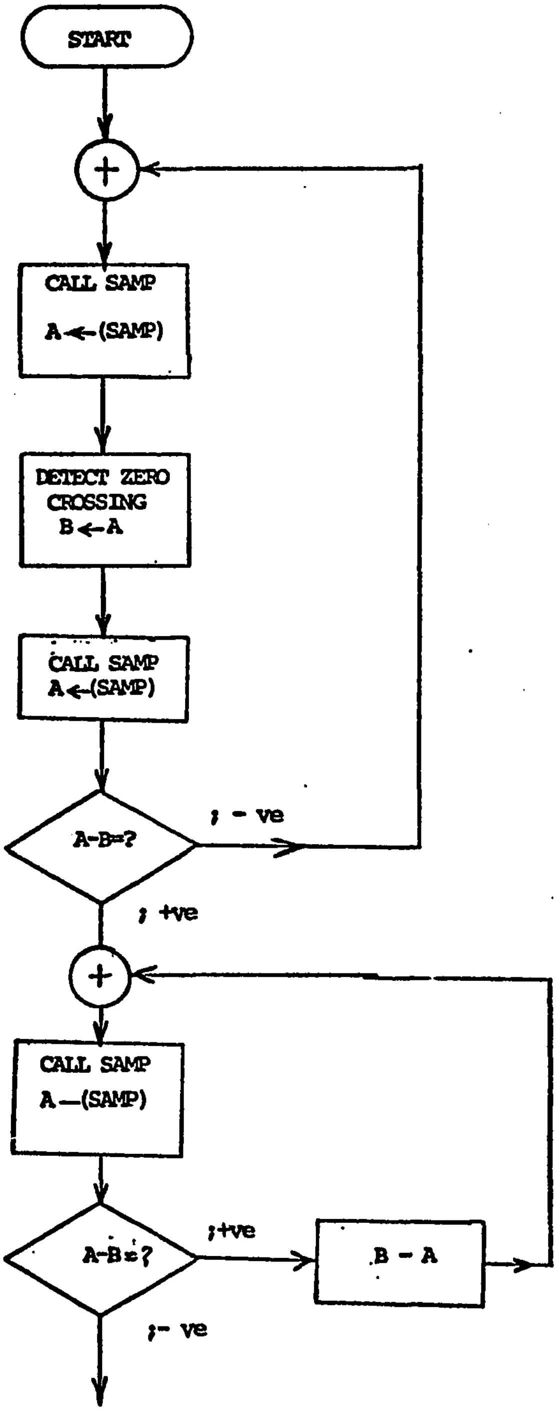


Figure 23. Amplitude Measurement

select instruction. In the later case at least 35 microseconds must be allowed to pass before reading the data.

In the beginning the amplitude measurement was done by converting the signal under observation into DC value by rectification. This DC value was then sampled and later converted into its actual value by taking into account the error involved because of non-linear element like diode. For more accurate measurement this method of measurement is switched to peak detection technique. This technique does not involve any rectification process, thus allowing more precise measurements. Various steps involved in this algorithm are shown in figure 23. First, the positive slope of of the waveform is deteced and then followed by the detection of the peak of the signal. During this peak detection a constant search for negative slope is made. The detection of negative slope terminates the search for the peak of the signal under test.

FREQUENCY MEASUREMENT:-

The principle involved in this measurement is to count the the number of eqully time spaced pulses spanned between two zero crossing of the waveform. The count thus obtained is directly the measure of half the period of the unknown signal. While counting the pulses it is made sure that the first zero crossing lies in the positive slope of the waveform and the second zero crossing lies in the negative slope of the waveform as shown in the figure 24. The time interval between each count is adjusted to a equal value by introducing NOP instructions between each sampling. The resolution of the zero crossing is limited to Ø3H of the count. The sampled data is read from location ADC8 is in two's complement form. In order to linearize the analog voltage values, 8ØH is added so that the range now lies between ØØH to FFH. The zero crossing then corresponds to 8ØH.

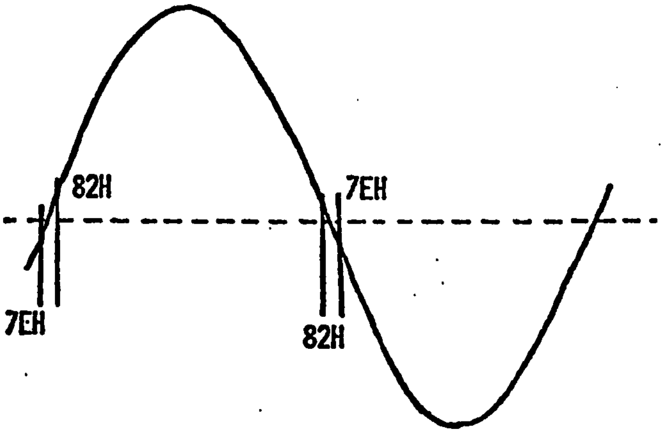


Figure 24. Zero Crossing Limits.

The assumption made for determining the zero crossing point: Any sample lying between the range 7EH and 82H is

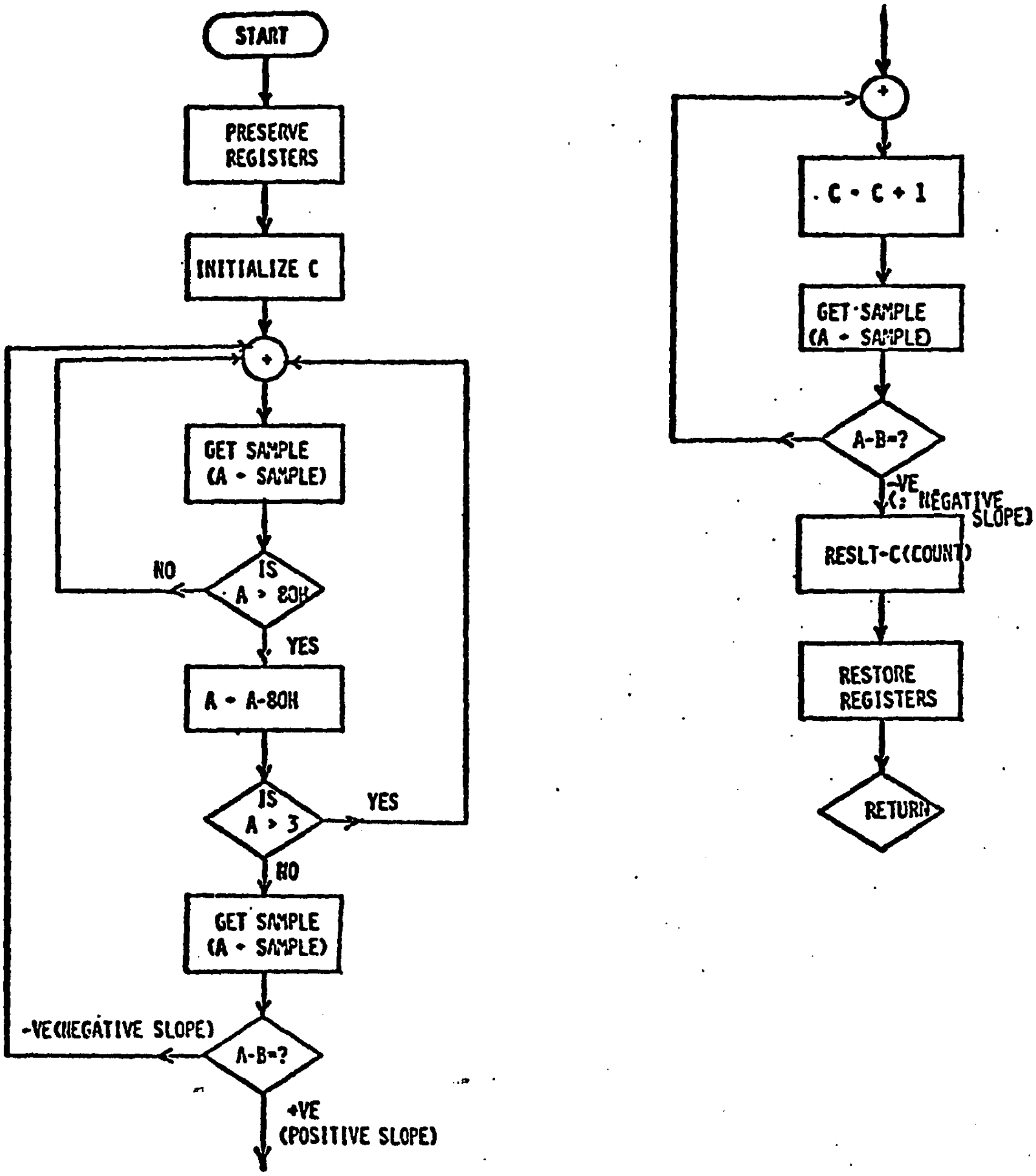


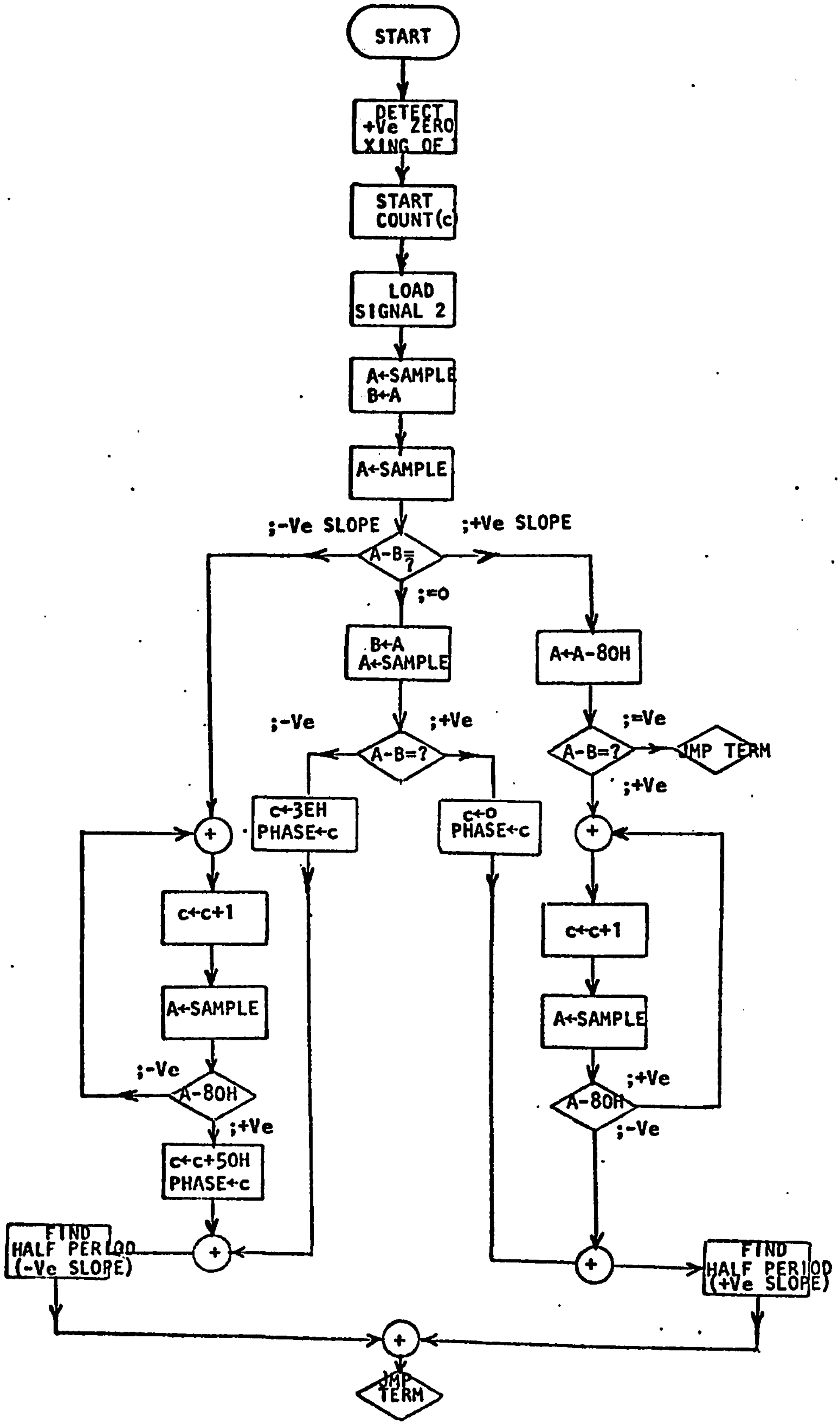
Figure 25. Frequency Measurement Flowchart

considered to be zero crossing point (Figure 24). Since this error exists in all measurements the error in the net result is not very significant. The various program steps involved in this frequency measuring algorithm are illustrated in the flowchart shown in figure 25.

To begin, the count is initialized and various registers are preserved. Then follows the search for positive slope of the waveform. Once the positive slope is detected the count is started. This count is incremented for every sample taken until negative slope zero crossing is found. The count thus obtained is the measure of half the period of the waveform.

PHASE MEASUREMENT:-

The principle utilized in this measurement is to count number of equally time spaced pulses between two positive zero crossings of two different signals. This is followed by the count corresponding to the half period of the second waveform . These two measures will be used to calculate the power factor of a signal. The various steps involved in this measurement are vividly shown in the flowchart given in the figure 26. The register values are preserved before entering this routine followed by the initialization of count. The search will be made for a sample in the neighborhood of 8ØH value. Once a sample around 8ØH is found the decision is made regarding the slope of the waveform on which it lies.


START
DETECT +Ve ZERO XING OF 1
START COUNT(c
LOAD SIGNAL 2
A←SAMPLE B←A
A←SAMPLE
;-Ve SLOPE
A-B=?
;+Ve SLOPE
;=o
B←A A←SAMPLE
A←A-80H
;-Ve
;+Ve
A-B=?
;=Ve
A-B=?
JMP TERM
;+Ve
c←3EH PHASE←c
c←o PHASE←c
+
+
c←c+1
c←c+1
A←SAMPLE
A←SAMPLE
;-Ve
A-80H
;+Ve
A-80H
;+Ve
;-Ve
c←c+50H PHASE←c
FIND HALF PERIOD (-Ve SLOPE)
+
+
FIND HALF PERIOD (+Ve SLOPE)
+
JMP TERM

If it is found to lie on the positive slope, the program is continued, otherwise it is looped back to the beginning, where it starts the search for the positive slope. If the positive slope zero crossing is detected the count will be incremented for every sample taken of the second channel, till the positive zero crossing of the second waveform is detected. As mentioned earlier the count interval is adjusted to be equal during any stage of the program with the introduction of NOP instructions wherever required. When the positive zero crossing of second waveform is found the count value is stored at location PHASE, since this gives a number directly proportional to the phase difference between two signals (Figure 27). After initializing the count second time the sampling is continued till the negative zero crossing of second signal is found. The count obtained thus will give a measure of the frequency of the waveform.

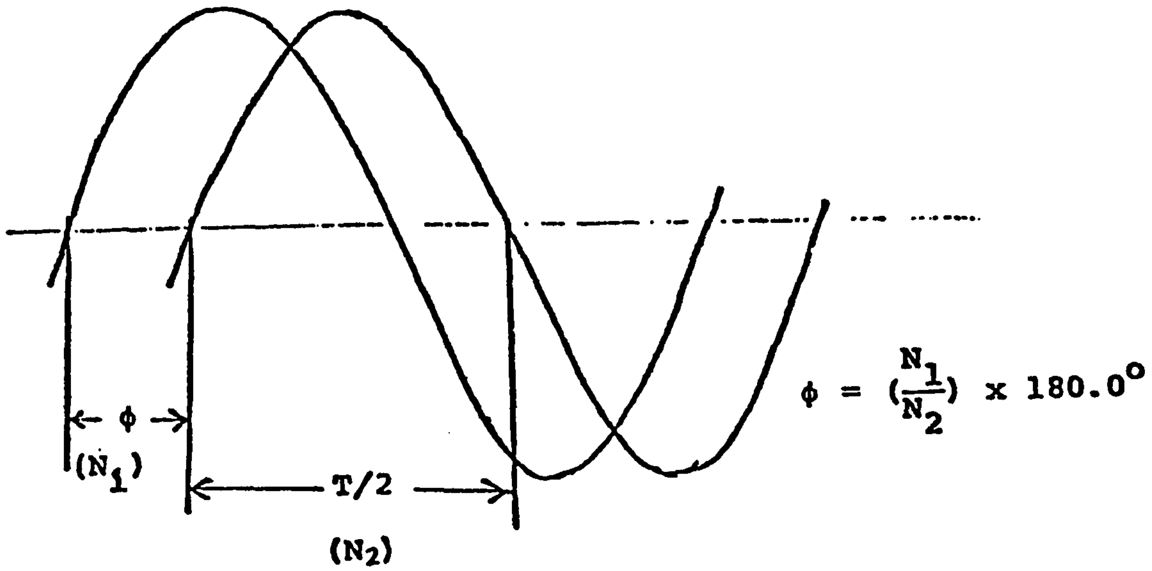


Figure 27. Phase and Frequency Relationship.

In all above subroutines if the desired channel is under critical observation the sampling is done at twice the normal frequency. The normal frequency is that which is used to display some variable information for routine check purposes only. This criterion is embedded in the software so the user does not have to change anything for such operation.

SOFTWARE STRUCTURE OF THE SYSTEM:-

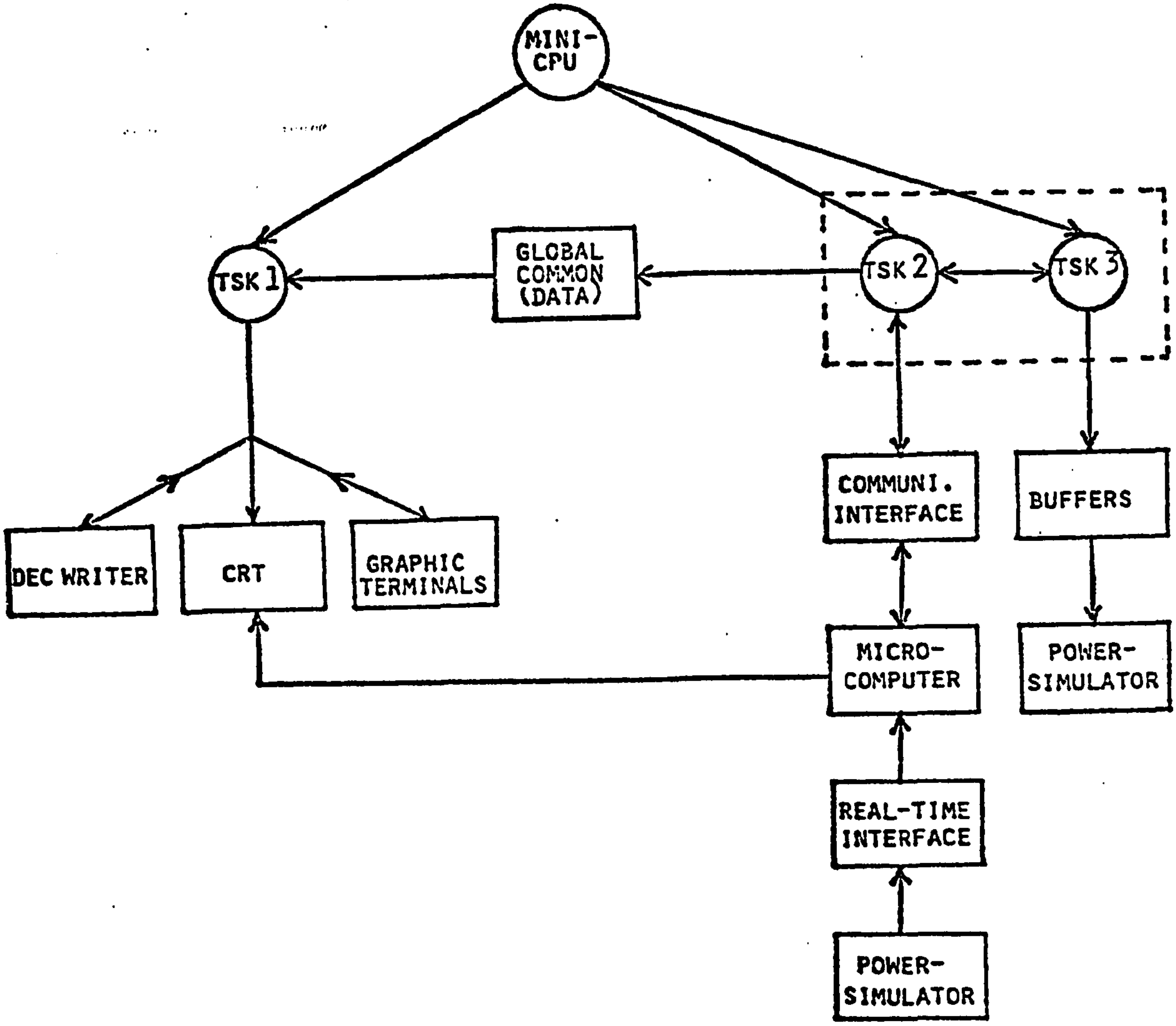


Figure 28. Software Structure of the System.

A block diagram of the entire software structure is shown in figure 28. The software consists of three basic components, which are Supervisory routine (TSK 1), Command and Communication Routine (TSK 2) and Simulator Controller Routine (TSK 3). TSK1 performs the major role in the operation of the entire structure of the software. TSK2 program establishes the communication link with the microcomputer through DL-11 W interface. TSK1 and TSK2 are connected with global common memory shared region called Data(DTA). Functions of TSK1 include support of graphics routines and the operation of updating the system variable values on the display. TSK3 controls the status of the functional switches of the power simulator, through DRS-11 interface. The CRT display provides transparent medium to view the communication protocol between the mini and micro. The connection details of CRT with mini and micro are shown in figure 13.

SUPERVISORY ROUTINE (TSK 1):-

This is the main control program for the interface between the graphics terminal and the simulator. It coordinates commands issued from the graphics terminal with the simulator and keeps the displays updated with the current status of the switches and variable data. The operation of this program is explained by referring to the flowchart shown in figure 29. The first step of this program is to display the operating instructions of the entire operation of the graphics interaction and selection of desired display (Figure 30). After updating the selected display the loop

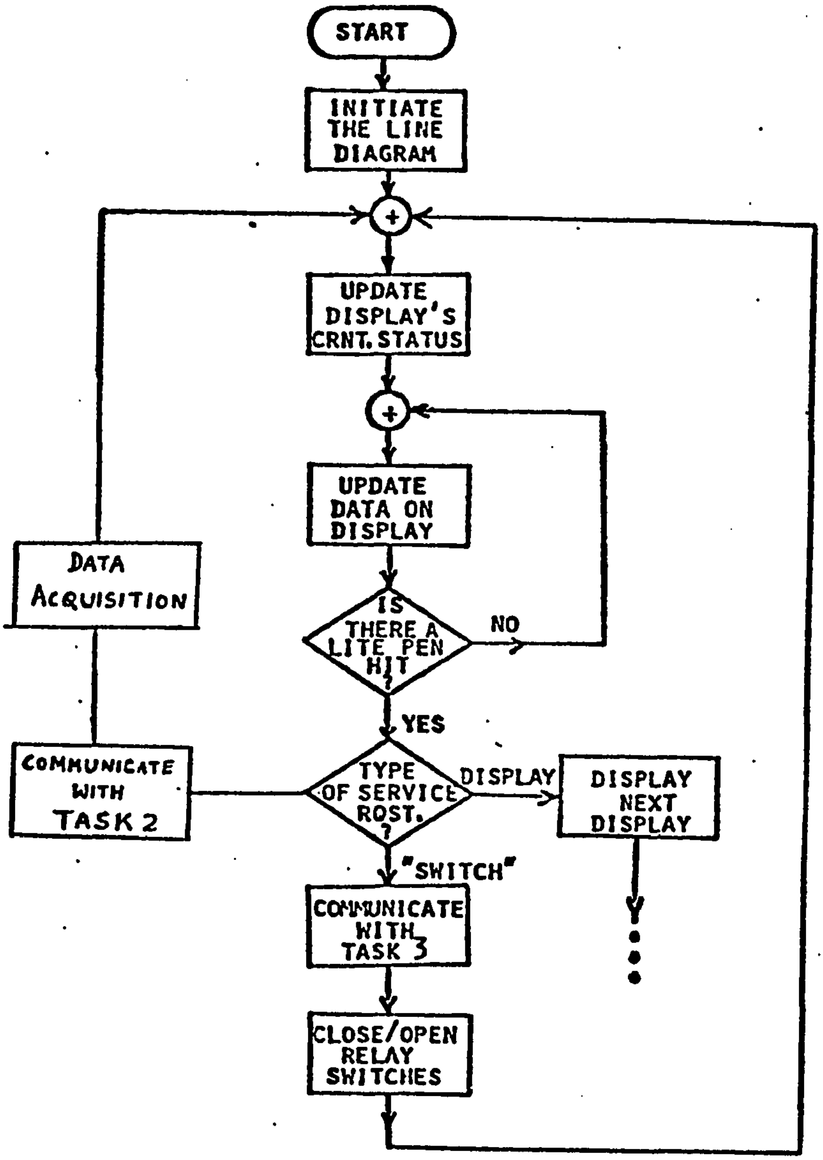
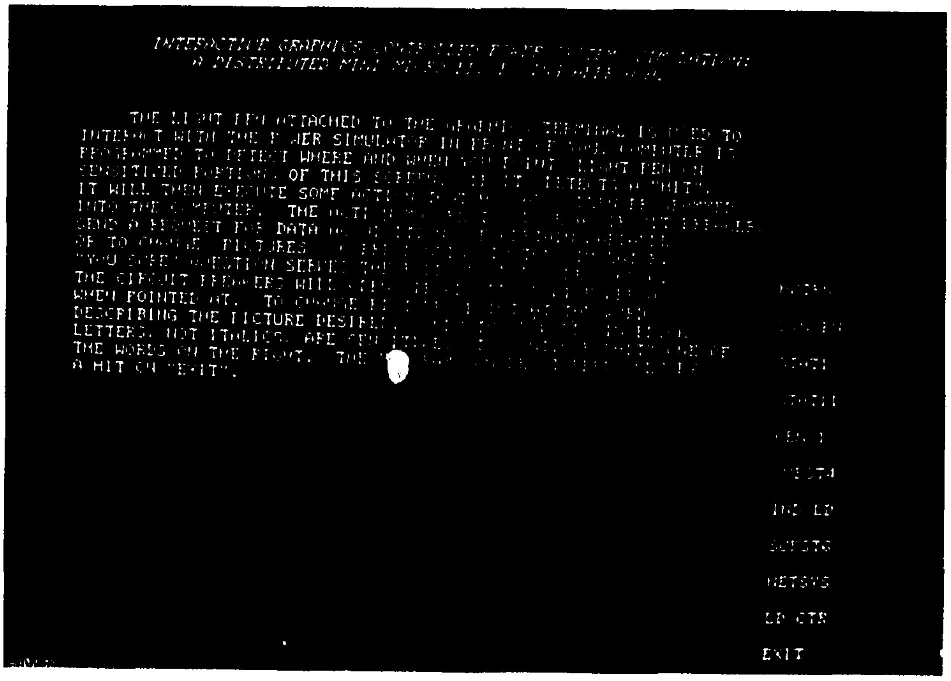


Figure 29. Control Program (TSK 1)

for monitoring data and waiting for light pen hits is entered. If there is a light pen, it is ascertained whether the hit corresponds to a request which results in displaying already stored display or a request to change the status of certain switch, which controls opening or closing of relay/switch on the power simulator panel. If the request is



INTERACTIVE GRAPHICS [illegible]
A DISTRIBUTED MINI [illegible]

THE LIGHT PEN ATTACHED TO THE GRAPHI[illegible] TERMINAL IS USED TO INTERACT WITH THE POWER SIMULATOR IN [illegible] COMPUTER IS PROGRAMMED TO DETECT WHERE AND WHEN [illegible] LIGHT PEN ON SENSITIZED PORTIONS OF THIS SCREEN [illegible] "HITS". IT WILL THEN EXECUTE SOME ACTI[illegible] INTO THE COMPUTER. THE [illegible] SEND A REQUEST FOR DATA [illegible] OR TO CHANGE PICTURES [illegible] "YOU SEE" QUESTION SERIES [illegible] THE CIRCUIT BREAKERS WILL [illegible] WHEN POINTED AT. TO CHANGE [illegible] DESCRIBING THE PICTURE DESIRED [illegible] LETTERS, NOT ITALICS, ARE [illegible] THE WORDS ON THE RIGHT. THE [illegible] A HIT ON "EXIT".

Figure 30. Photograph of Initial Display

for the later, then it communicates with TSK3 through common block of data called SW(96) and updates the requested relay/switch status. If the hit corresponds to a sampling schedule, then the necessary command in predetermined format is transmitted to the microcomputer through DL-11 W communication link. This communication is displayed on CRT as shown in figure 31.

```
$OUII$
P
U0EDC20A833E3E3C3C30 [illegible] 483858330$
P
U0EDC20A833E3E3C3C3094309C [illegible] 284E5F483858330$
P
U0EDC20A833E3E3C3C3094309C3CC2989792EFD08284E5F483858330$
$OURR$

P
U0EDC208833E3E303C3094309C3D449897 86EFD08284E5F4838583C4$
P
U0EDC208833E3E3C3C3094309C3EC2989795EFD08284E5F483858337$
P
U0EDC2E0833E3E303C3094309C3D55989787EFD08284E5F4838583AD$
$OURR$

P
U0EDC207833E3E303C3094309C3DA7989784EFD08284E5F483858367$
P
U0EDC208833E3E3C3C3094309C3E4A989786EFD08284E5F48385838E$
P
█0EDC2E2833E3E303C3094309C3C06989793EFD08284E5F4838583EF$
```

Figure 31. Photograph of Commuñication Between Mini and Micro.

If not, program jumps back to the stage of updating the display's current status and waits for the light pen hit to occur, as described earlier.

COMMAND AND COMMUNICATION ROUTINE (TSK 2):-

The flowchart of this program is shown in figure 32. The program starts with the initialization of system variables. After reading the command from graphics terminal because of user interaction ,the validity of the command is checked against predefined format, which is shown in figure 18. If it is a valid command, it is transmitted to the microcomputer.

The program then waits for the arrival of digital data from the microcomputer corresponding to the state variables under request. This data is checked against the predetermined data format shown in figure 33. If the received data is found to be valid it is normalized.

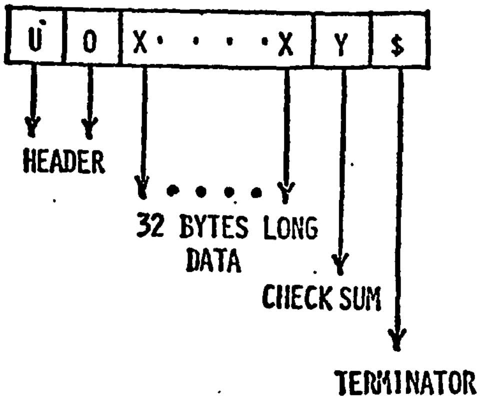


Figure 31. Data Format.

The following table 1 provides the relationship between the elments of Common Array A(174) with its corresponding analog

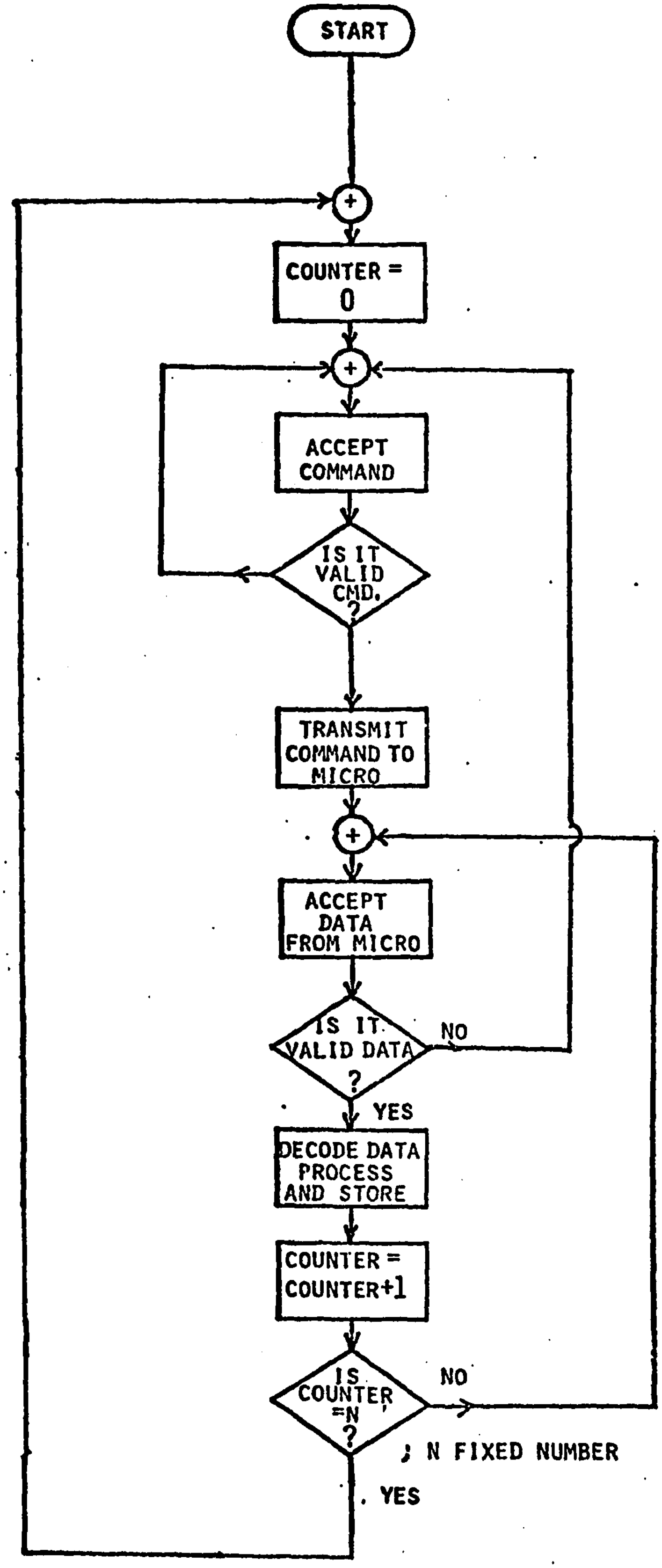


Figure 32. Command and Communication Flowchart.

channels of Power Simulator.

ARRAY A(164) VARIABLE	SIMULATOR VARIABLE
-----------	---------
A(131)	Gen.1 Voltage
A(132)	Gen.1 Frequency
A(133)	Gen.1 Current
A(134)	Gen.1 Power factor
A(135)	Gen.1 KW
A(136)	Gen.2 Voltage
A(137)	Gen.2 Frequency
A(138)	Gen.2 Current
A(139)	Gen.2 Power factor
A(14Ø)	Gen.2 KW
A(141)	Gen.3 Voltage
A(142)	Gen.3 Frequency
A(143)	Gen.3 Current
A(144)	Gen.3 Power factor
A(145)	Gen.3 KW
A(146)	Inter Tie Voltage
A(147)	Inter Tie Frequency
A(148)	Inter Tie Current
A(15Ø)	Substation 4 Right Bus
A(151)	Industrial Load Voltage

A(152)	Industrial Load Powerfactor
A(153)	Industrial Load Current
A(154)	Load Center Right current
A(155)	Load Center left Current
A(156)	Load center left Voltage
A(157)	Load center right Voltage
A(158)	Substation 6 Bus Voltage
A(159)	Network System L-N1 Voltage
A(160)	Network System L-N2 Voltage
A(161)	Network System L-n3 Voltage
A(162)	Net. Sys. Line-1 Current
A(163)	Net. Sys. Line-2 Current
A(164)	Net. Sys. Line-3 Current

Table 1

The following expressions are used to normalize the state variables of power simulator.

A(131) = (Y1(1)-128) * 5.045

A(132) = (62.0 * 60.0) / Y1(5)

A(133) = (Y1(15)-126) * 0.0167

A(134) = (((Y1(10) / Y1(9)) * 180.0)

or

A(134) = -((Y1(10)/Y1(9)) * 180.0)

```
A(135) = A(131) * A(133) / 1732.Ø

A(136) = (Y1(2)-128) * 4.96

A(137) = (62.Ø * 6Ø.Ø) / Y1(6)

A(138) = (Y1(16)-128) * Ø.Ø165

A(139) = ((Y1(12) / Y1(11)) * 18Ø.Ø)
        or
A(139) = -(Y1(12)/Y1(11)) * 18Ø.Ø)

A(14Ø) = (A(136) * A(138)) / 1732.Ø

A(141) = (Y1(3)-124) * 5.35

A(142) = (62.Ø * 6Ø.Ø) / Y1(7)

A(143) = (Y1(17) - 128) * Ø.Ø1936

A(144) = ((Y1(14) / Y1(13)) * 18Ø.Ø)

A(145) = (A(141) * A(143)) / 1732.Ø

A(146) = (Y1(4)-128) * 4.Ø

A(148) = (Y1(24) - 128) * Ø.2238
A(147) = (62.Ø * 6Ø.Ø) / Y1(8)

A(15Ø) = (Y1(18)-128) * 4.47

A(151) = (Y1(2Ø) - 128) * 4.273
```

```
A(152) = 1.Ø

A(153) = (Y1(25) - 128) * Ø.23178

A(155) = A(146) * 2.8846

A(156) = (Y1(19) - 122) * 5.1Ø9

A(158) = (Y1(22)-128) * 4.765

A(159) = (Y1(23)-128) * 1.Ø98

A(162) = (Y1(26) - 128) * Ø.1Ø556
```

Thus normalized data is stored in a global common data which is a common memory region shared by TSK2 and TSK3. This data is used to update the displays on the graphics console as described before. This is looped for a fixed number of times(N=3), which is a software variable parameter, before a decision is taken whether to continue the same parameters sampling or not. If it is desired to continue the same for another set of N times, the process continues, otherwise counter is reset to zero and the command for a change of variables is accepted.

SIMULATOR CONTROLLER INTERFACE:-

This program controls the status of various switches involved in the power simulator. The status of a particular switch is derived from the global common array SW(96). This array maintains current status of all functional controlling switches of the simulator. The opening or closing of a certain switch is decided by this program. The operator's interaction is transferred as a change in the value of one of the elements of the common array. If the switch's current status is Ø, after user interaction it changes to 1, and vice versa. TSK3 senses this change and acts accordingly in updating the status of a particular switch of the power simulator.

INTERDEPENDENCY OF ALL TASKS:-

The interdependency of various tasks can be explained referring to the timing diagrams shown in figure 34.

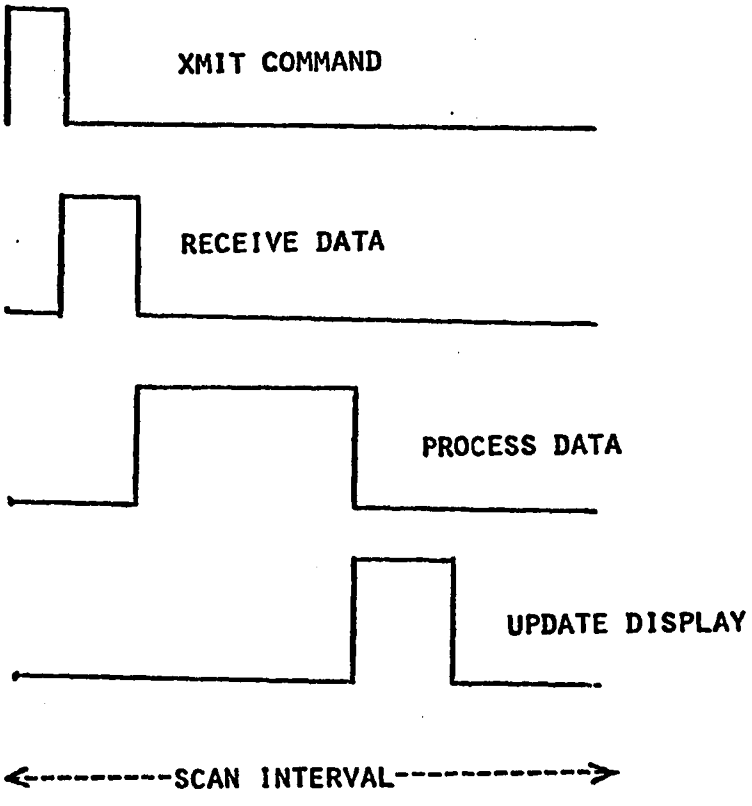


Figure 34. Interdependency of all Tasks.

There are two flag variables A(1Ø1) and A(1Ø2), which are used as indirect software interrupts among all the programs. To begin, these two flags are reset { A(1Ø1)=Ø.Ø and A(1Ø2)=Ø.Ø }. In the main control program the user interaction with the graphics display results in a light pen hit. If this hit implies a request for sampling schedule the flag A(1Ø1) is set to a numerical value which directly corresponds the name of the schedule requested. For example if the schedule requested is E then the values of this variable is equalled to 5.Ø, or if the schedule requested is H, the variable A(1Ø1) is set equal to 8.Ø. These numerical values of A(1Ø1) indicate the position of the Alphabetical command in the 26 alphabetical list. During this time Communication program is waiting for the variable A(1Ø1) to be set to a non-zero value. Once the variable A(1Ø1) is set to Non-Zero value, the communication routine is reactivated. This program sends properly formatted command to microcomputer and waits for the data from the microcomputer. Meanwhile, the display updating routine of main control program, MOD is waiting for the variable A(1Ø2) to be set. When communication routine receives data relating all the 26 variables of the simulator, it will process the data and normalize using the equations given in Table 1 . After completing the processing of received data A(1Ø2) is set, indicating that the data is ready for display. At this time the MOD routine takes over and updates the display of simulator section on the graphics terminal. Before this routine loops back to testing of A(1Ø2), it will reset A(1Ø2). Bythis time the scan interval is over resulting second transmission of data from microcomputer to the minicomputer communication program. Thus the process of updating the display for a period of N times is follwed. After completion of predetermined number of scans, the communication program loops back to the stage of checking the non-zero value for A(1Ø1) which was reset by communication program before it transmits the command to micro.

Following are some of the graphical displays corresponding to the power simulator. These line diagrams show the node voltages and associated branch currents and frequencies.

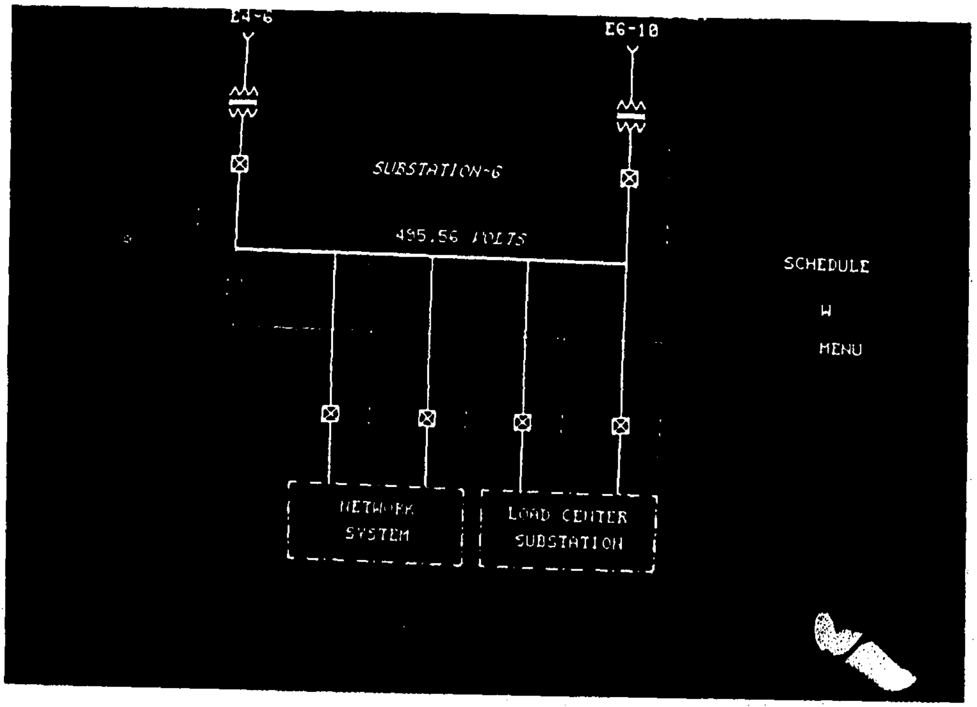


Figure 35. Substation-6 Display.

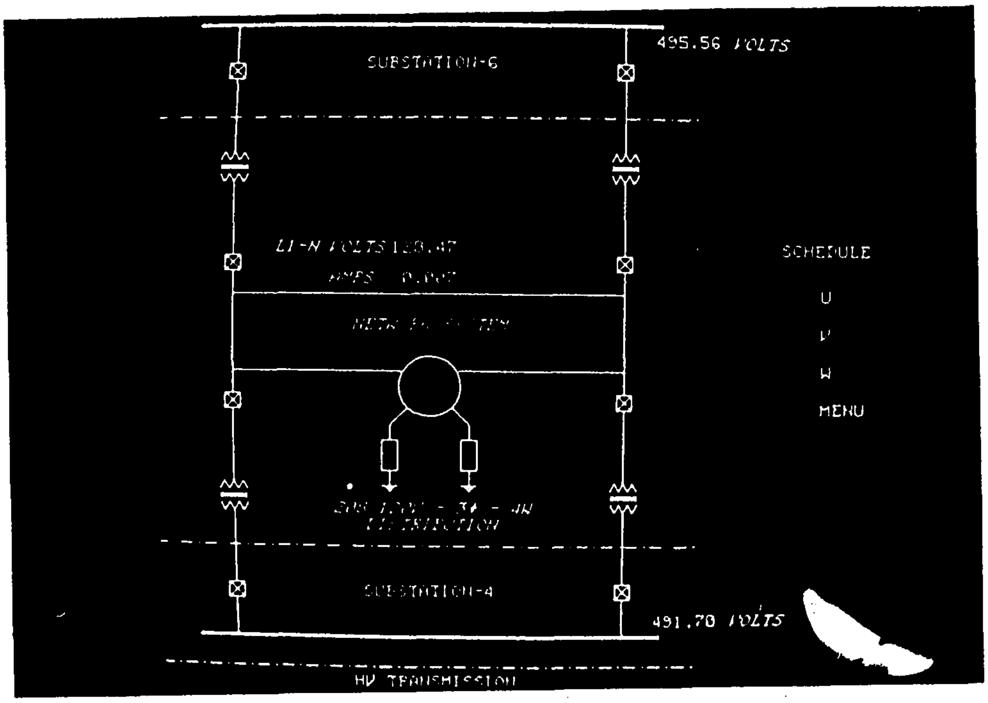


Figure 36. Network System

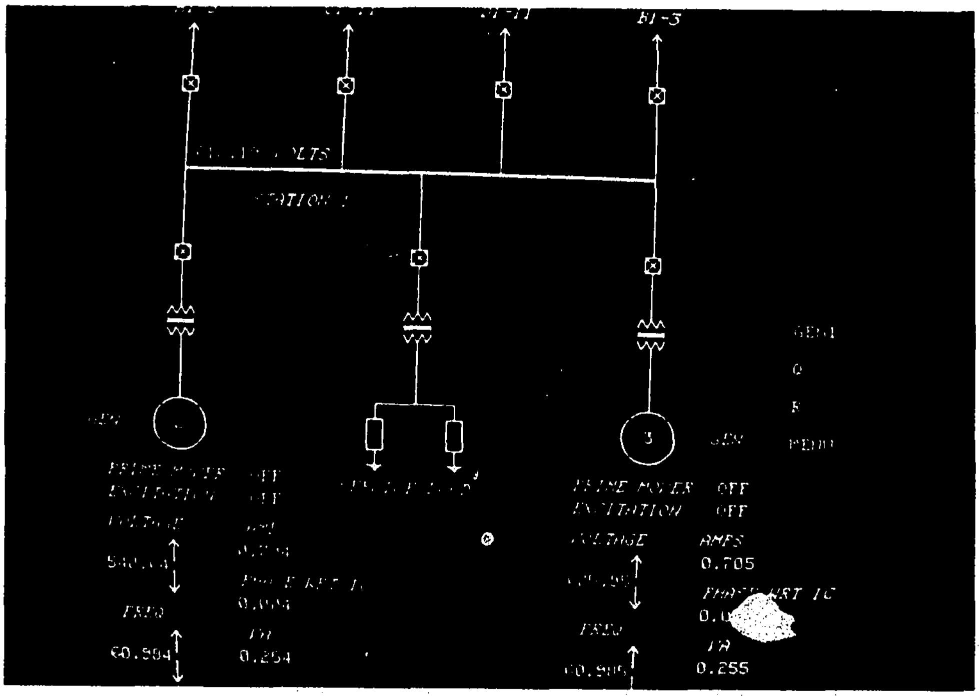


Figure 37. Station-1

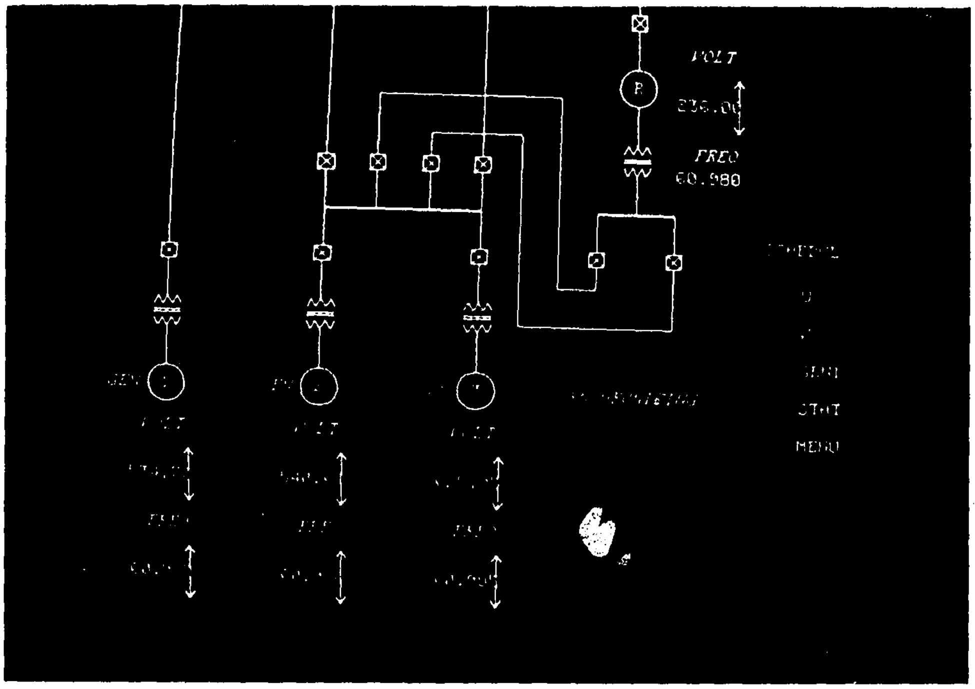


Figure 38. Synchronization Display

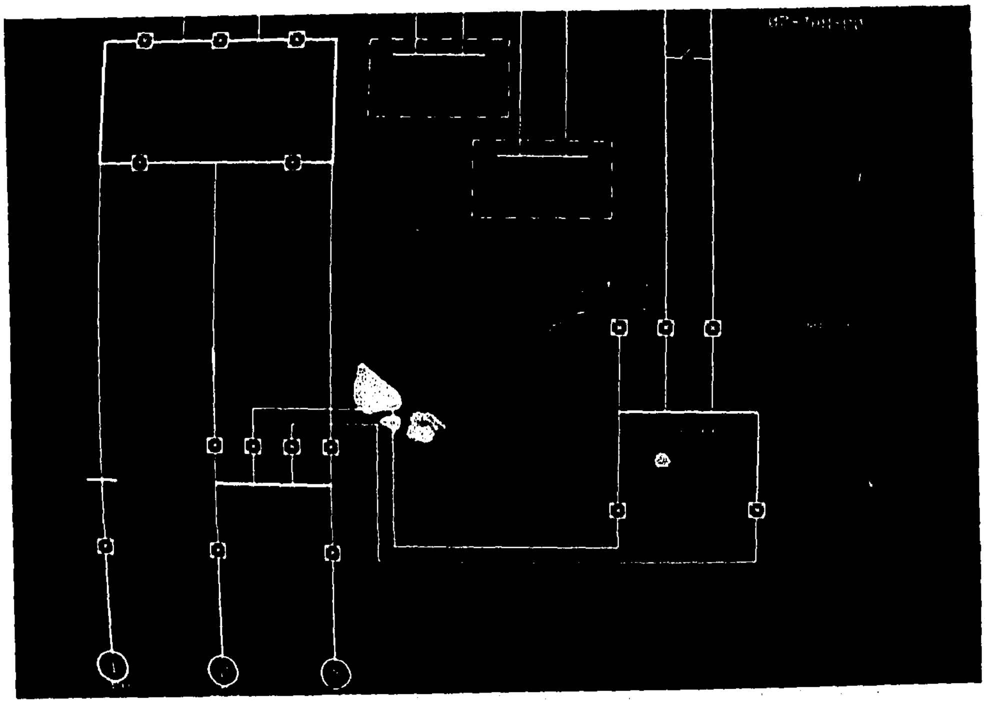


Figure 39. HV Transmission Display.

To summarize, this chapter discusses the operating systems of both computer systems followed by the software introduction of the real time interface module. The sampling schedules for data acquisition are introduced and their operational details discussed. This followed by the discussion on software structure of system with reference to the operation of the entire configuration.

CHAPTER 5

SUMMARY AND CONCLUSIONS

Interactive graphics controlled power system control demonstrated in this dissertation advances the state of art to make a future closed loop power system control a reality. The distributed processing outlined herein demonstrates the use of an inexpensive microcomputer systems for the data acquisition and control at various lower levels of a power system. The contributions of this research are outlined as follows:

1. The conceptual development and verification of a closed loop real-time controlled power system with a multi-node computer network having interactive graphics as the medium of interaction with the system.

2. Conceptual development of communication protocol between mini and microcomputers with RSX-11M as the operating system of the minicomputer.

3. Development of interface between mini and micro, mini and simulator and micro and simulator.

4. Conceptual development of graphics oriented interactive data acquisition schemes. These schemes serve the

measurement of phase, frequency and amplitude measurements.

5. The concept of integrated power system with distributed processing computer systems for real-time monitoring and control.

The result of all above, is a one of a kind research and educational tool in the area of power engineering, which provides the functional environment for studies towards closed loop control of a power system. Although hardware and software are in a continuous state of evolution, the tool serves as an operational device for education as well as research. As a testbed for research it can be used for testing contingency analysis algorithms or forced fault analysis which cannot be possible on a real life power system, because of fear of disrupting the power supplies to the community.

In the problems involving the monitoring of data for a decision, the proposed technique of on-line graphics displays makes it easy for quick review of various state variables of the system. The interactive model can also be used by distribution planners to model, study, and test present and future distribution systems. The system also can be used to as an operating tool for quick evaluation of alternatives when abnormal circuit conditions arise due to planned outages or when problems arise because of the delays involved in the construction or repairs .

In addition to above the result of this research is a "one of a kind" simulator of power systems which will be used to improve real-time understanding and operation of the power system.

CONCLUSION:

Although hardware and software are in a continuous state of evolution, the tool serves as an operational device for education as well as research. The system developed in this work forms a unique testbed for future research in the direction of real time control of power systems. This device can also be used to train the operators and students for understanding the system.

The employment of inexpensive microcomputer system for successful operation in the lower levels of control of power system indicate that a multiple number of microcomputer systems can be employed to share the processing and acquisition of data. The future development lies in writing various contingency algorithms either to control generation or fault analysis of the power systems.

Appendix I

Microcomputer based Data Acquisition Algorithms.

```
ASM80 JAN7.SRC MACROFILE PAGELENGTH(58) PAGEUIDTH(90)


ISIS-II 8080/8085 MACRO ASSEMBLER, V3.0        MODULE   PAGE    1


  LOC  OBJ         LINE        SOURCE STATEMENT

  0800               1          ORG     800H
                     2 ;
                     3 ;THIS PROGRAM RUNS ON SBC 80/10A AND ITS
                     4 ;FAMILY RELATED SINGLE BOARD COMPUTERS. THE USER
                     5 ;IS REQUIRED TO HAVE MINIMAL MONITOR FUNCTIONS
                     6 ;UITH THE SYSTEM IN ORDER TO EXECUTE THIS PROGRAM.
                     7 ;EXECUTABLE INSTRUCTIONS ARE STORED IN EPROMS AND
                     8 ;UPPER PORTION OF THE SCRATCHPAD MEMORY IS USED
                     9 ;FOR STACKING AND OTHER TEMPORARY DATA STORAGE
                    10 ;REQUIREMENTS.
                    11 ;THIS ASSEMBLY LISTING FACILITATES THE USER TO
                    12 ;TO USE ANY SUBROUTINE FOR THE DEVELOPMENT OF
                    13 ;HIS/HER OUN PROGRAMSFOR FUTURE ADDITIONS
                    14 ;TO THIS PROGRAM.
                    15 ;FOLLOUING EQUATE COMMANDS CAN BE SEEN FOR MORE
                    16 ;EXPLANATIONS IN THE DOCUMENTS LISTED BELOU:
                    17 ;
                    18 ;
                    19 ;1...RTI-1200,ANALOG DEVICES, USERS MANUAL
                    20 ;2...SBC 80/10A,INTEL CORP.,USERS GUIDE
                    21 ;
                    22 ;THIS PROGRAM SATISFIES ALL THE REQUIREMENTS OF DATA
                    23 ;ACQUISITION OF MOST DESIRED VARIABLES OF POUER
                    24 ;SYSTEM MODEL FOR DATA UPDATE AND DECISION
                    25 ;REQUIREMENTS BY THE MINICOMPUTER.
                    26 ;
                    27 ;
  03FD              28 CI       EQU     03FDH
  03FA              29 CO       EQU     03FAH
  8FFE              30 ADCHI    EQU     8FFEH
  8FFD              31 ADLCO    EQU     8FFDH
  8FFC              32 STATUS   EQU     8FFCH
  8FFB              33 CNVCMD   EQU     8FFBH
  8FFA              34 MUXADR   EQU     8FFAH
  8FF9              35 GNSEL    EQU     8FF9H
  8FF8              36 ADC8     EQU     8FF8H
  8FF3              37 DRIVE    EQU     8FF3H
  8FF0              38 SETUP    EQU     8FF0H
  8FFF              39 CRDSEL   EQU     8FFFH
  001B              40 ESC      EQU     1BH
  0024              41 DOL      EQU     24H
  02DF              42 REGDS    EQU     02DFH
  002C              43 GETCH    EQU     2CH
                    44 ;
                    45 ;        INITIALIZE THE STACKPOINTER
                    46 ;
```

```
 LOC  OBJ         LINE        SOURCE STATEMENT

 0800 21F03F        47         LXI     H,3FF0H
 0803 F9            48         SPHL
                    49 ;
                    50 ;
                    51 ;         INITIALIZE THE CONSTANTS TO FIX
                    52 ;         THE DELAY REQUIRED FOR DL-11 TO RESPOND
                    53 ;
 0804 3E0F          54         MVI     A,0FH
 0806 329F3C        55         STA     MARCH
 0809 3EFF          56         MVI     A,0FFH
 080B 32A03C        57         STA     MARCH+1
                    58 ;
                    59 ;
                    60 ;         INITIALIZE THE DATA STORAGE LOCATIONS
                    61 ;
 080E 21673C        62         LXI     H,DATA
 0811 061A          63         MVI     B,26
 0813 3E00          64         MVI     A,0
 0815 77            65 BEG:    MOV     M,A
 0816 23            66         INX     H
 0817 05            67         DCR     B
 0818 C21508        68         JNZ     BEG
 081B 32403C        69         STA     TEST     ;INTIALIZE CONTINUE CONND. FLAG
 081E 3E03          70         MVI     A,03     ;INITIALIZE THE NUMBER OF
                    71 ;                        ;TIMES THE SAMPLING REQUIRED WITHOUT OPERA
                       TOR
                    72 ;                        ;INPUT
 0820 328C3C        73         STA     NUMB
 0823 C32608        74         JMP     START
                    75 ;
                    76 ;
                    77 ;         MACRO DEFINITION FOR ONE CHANNNEL SAMPLING
                    78 ;
                    79 ;
                    80 ONE     MACRO   N1
 -                  81         MVI     A,N1
 -                  82         MOV     M,A
 -                  83         INX     H
                    84         ENDM
                    85 ;
                    86 ;         MACRO DEFINITION FOR TWO CHANNEL SAMPLING
                    87 ;
                    88 ;
                    89 TWO     MACRO   N2,N3
 -                  90         INX     H
 -                  91         MVI     A,N2
 -                  92         MOV     M,A
 LOC  OBJ         LINE        SOURCE STATEMENT

 -                  93         INX     M
 -                  94         MVI     A,N3
 -                  95         MOV     M,A
                    96         ENDM
                    97 ;
                    98 ;
                    99 ;       MACRO FOR SAMPLING TERMINATOR CHARACTER
                   100 ;
                   101 STP     MACRO
 -                 102         MVI     A,30H
 -                 103         MOV     M,A
                   104         ENDM
                   105 ;
                   106 ;       MACRO DEFINITION TO SET THE IDLE-TIME FOR
                   107 ;       FAST SCHEDULES
                   108 ;
                   109 FAST    MACRO
 -                 110         MVI     A,1
 -                 111         STA     TIME
                   112         ENDM
                   113 ;
                   114 ;       RECEIVE AND STORE THE COMMAND(SCHEDULE)
                   115 ;       TRANSMITTED BY THE MINICOMPUTER
                   116 ;
                   117 ;
 0826 218D3C       118 START:  LXI     H,CMND
 0829 060E         119         MVI     B,14
 082B CDFD03       120 L1:     CALL    CI
 082E E67F         121         ANI     7FH
 0830 77           122         MOV     M,A
 0831 23           123         INX     H
 0832 05           124         DCR     B
 0833 C22B08       125         JNZ     L1
 0836 00           126         NOP
                   127 ;
                   128 ;
                   129 ;
                   130 ;       CHECK THE VALIDITY OF THE COMMAND
                   131 ;
                   132 ;
 0837 218E3C       133         LXI     H,CMND+1
 083A 7E           134         MOV     A,M
 083B FE24         135         CPI     DOL
 083D C2A10F       136         JNZ     ERROR
 0840 23           137 L2:     INX     H
 0841 7E           138         MOV     A,M
 0842 FE4F         139         CPI     'O'
 LOC  OBJ         LINE        SOURCE STATEMENT

 0844 C2A10F       140          JNZ     ERROR
 0847 23           141          INX     H
 0848 7E           142          MOV     A,M
 0849 FE55         143          CPI     'U'
 084B C2A10F       144          JNZ     ERROR
 084E 23           145          INX     H
 084F 7E           146          MOV     A,M
 0850 23           147          INX     H
 0851 46           148          MOV     B,M
 0852 B8           149          CMP     B
 0853 C2A10F       150          JNZ     ERROR
 0856 FE43         151          CPI     'C'       ;IS IT CONTINUE COMMAND ?
 0858 CA8008       152          JZ      CONT      ;YES
 085B 3E01         153          MVI     A,1
 085D 32403C       154          STA     TEST      ;SET CONTINUE FLAG IF CCMD IS NOT FIRST CO
                        MMAND
 0860 78           155          MOV     A,B
                   156 ;
                   157 ;
                   158 ;        FIND THE CORRESPONDING SERVICE ROUTINE ADDRESS
                   159 ;
 0861 011A00       160          LXI     B,26
 0864 21D90E       161          LXI     H,SCHED
 0867 BE           162 LOC1:    CMP     M
 0868 CA7308       163          JZ      LOC2
 086B 23           164          INX     H
 086C 0D           165          DCR     C
 086D C26708       166          JNZ     LOC1
 0870 C3A10F       167          JMP     ERROR
 0873 21F30E       168 LOC2:    LXI     H,ADRES
 0876 09           169          DAD     B
 0877 09           170          DAD     B
 0878 7E           171          MOV     A,M
 0879 23           172          INX     H
 087A 66           173          MOV     H,M
 087B 6F           174          MOV     L,A
 087C 229D3C       175          SHLD    SLEP
 087F E9           176          PCHL
 0880 3A403C       177 CONT:    LDA     TEST
 0883 FE01         178          CPI     1         ;IS THIS FIRST COMMAND ?
 0885 CA430C       179          JZ      CCMD1     ;NO
 0888 C33F0C       180          JMP     CCMD      ;YES
                   181 ;
                   182 ;        'A' SCHEDULE(COMMAND)
                   183 ;        GENERATOR 1 VOLTAGE (CHANNEL 1) DC
                   184 ;
                   185 ACMD:
```

```
 LOC  OBJ         LINE        SOURCE STATEMENT

                   186          FAST
 088B 3E01         187+         MVI     A,1
 088D 329C3C       188+         STA     TIME
 0890 21513C       189          LXI     H,TEMP
                   190          ONE     1
 0893 3E01         191+         MVI     A,1
 0895 77           192+         MOV     M,A
 0896 23           193+         INX     H
                   194          STP
 0897 3E30         195+         MVI     A,30H
 0899 77           196+         MOV     M,A
 089A CD290F       197          CALL    PREP
 089D 3A813C       198          LDA     QUASI
 08A0 32673C       199          STA     DATA
 08A3 C3480C       200          JMP     LOC4
                   201 ;
                   202 ;        'B'COMMAND(SCHEDULE)
                   203 ;        GENERATOR 2 VOLTAGE (CHANNEL 2) DC
                   204 ;
                   205 ;
                   206 BCMD:
                   207          FAST
 08A6 3E01         208+         MVI     A,1
 08A8 329C3C       209+         STA     TIME
 08AB 21513C       210          LXI     H,TEMP
                   211          ONE     3
 08AE 3E03         212+         MVI     A,3
 08B0 77           213+         MOV     M,A
 08B1 23           214+         INX     H
                   215          STP
 08B2 3E30         216+         MVI     A,30H
 08B4 77           217+         MOV     M,A
 08B5 CD290F       218          CALL    PREP
 08B8 3A813C       219          LDA     QUASI   ;STORE THE SAMPLE VALUE
 08BB 32683C       220          STA     DATA+1  ;STORE AT CORRESPONDING LOCATION
 08BE C3480C       221          JMP     LOC4
                   222 ;
                   223 ;        'D'COMMAND(SCHEDULE)
                   224 ;        GENERATOR 3VOLTAGE(CHANNEL 5) DC
                   225 ;
                   226 DCMD:
                   227          FAST
 08C1 3E01         228+         MVI     A,1
 08C3 329C3C       229+         STA     TIME
 08C6 21513C       230          LXI     H,TEMP
                   231          ONE     5
 08C9 3E05         232+         MVI     A,5
```

```
LOC  OBJ          LINE        SOURCE STATEMENT

08CB 77            233+        MOV     M,A
08CC 23            234+        INX     H
                   235         STP
08CD 3E30          236+        MVI     A,30H
08CF 77            237+        MOV     M,A
08D0 CD290F        238         CALL    PREP
08D3 3A813C        239         LDA     QUASI   ;LOAD THE SAMPLE VALUE
08D6 32693C        240         STA     DATA+2  ;STORE AT CORRESPONDING LOCATION
08D9 C3480C        241         JMP     LOC4
                   242 ;
                   243 ;
                   244 ;       'E'COMMAND(SCHEDULE)
                   245 ;       INTERCONNECTION VOLTAGE (CHANNEL 7) DC
                   246 ;
                   247 ECMD:
                   248         FAST
08DC 3E01          249+        MVI     A,1
08DE 329C3C        250+        STA     TIME
08E1 21513C        251         LXI     H,TEMP
                   252         ONE     7
08E4 3E07          253+        MVI     A,7
08E6 77            254+        MOV     M,A
08E7 23            255+        INX     H
                   256         STP
08E8 3E30          257+        MVI     A,30H
08EA 77            258+        MOV     M,A
08EB CD290F        259         CALL    PREP
08EE 3A813C        260         LDA     QUASI
08F1 326A3C        261         STA     DATA+3
08F4 C3480C        262         JMP     LOC4
                   263 ;
                   264 ;       'F'COMMAND(SCHEDULE)
                   265 ;       GENERATOR 1 FREQUENCY (CHANNEL 17) AC
                   266 ;
                   267 FCMD:
                   268         FAST
08F7 3E01          269+        MVI     A,1
08F9 329C3C        270+        STA     TIME
08FC 21513C        271         LXI     H,TEMP
08FF 3E32          272         MVI     A,32H
0901 77            273         MOV     M,A
0902 23            274         INX     H
                   275         ONE     17
0903 3E11          276+        MVI     A,17
0905 77            277+        MOV     M,A
0906 23            278+        INX     H
                   279         STP
```

```
 LOC  OBJ         LINE        SOURCE STATEMENT

 0907 3E30         280+        MVI     A,30H
 0909 77           281+        MOV     M,A
 090A CD290F       282         CALL    PREP
 090D 3A813C       283         LDA     QUASI
 0910 326B3C       284         STA     DATA+4
 0913 C3480C       285         JMP     LOC4
                   286 ;
                   287 ;       'G'COMMAND(SCHEDULE)
                   288 ;       GENERATOR 2 FREQUENCY (CHANNEL 18) AC
                   289 ;
                   290 GCMD:
                   291         FAST
 0916 3E01         292+        MVI     A,1
 0918 329C3C       293+        STA     TIME
 091B 21513C       294         LXI     H,TEMP
 091E 3E32         295         MVI     A,32H
 0920 77           296         MOV     M,A
 0921 23           297         INX     H
                   298         ONE     18
 0922 3E12         299+        MVI     A,18
 0924 77           300+        MOV     M,A
 0925 23           301+        INX     H
                   302         STP
 0926 3E30         303+        MVI     A,30H
 0928 77           304+        MOV     M,A
 0929 CD290F       305         CALL    PREP
 092C 3A813C       306         LDA     QUASI
 092F 326C3C       307         STA     DATA+5
 0932 C3480C       308         JMP     LOC4
                   309 ;
                   310 ;       'H'COMMAND(SCHEDULE)
                   311 ;       GENERATOR 3 FREQUENCY (CHANNEL 19) AC
                   312 ;
                   313 HCMD:
                   314         FAST
 0935 3E01         315+        MVI     A,1
 0937 329C3C       316+        STA     TIME
 093A 21513C       317         LXI     H,TEMP
 093D 3E32         318         MVI     A,32H
 093F 77           319         MOV     M,A
 0940 23           320         INX     H
                   321         ONE     19
 0941 3E13         322+        MVI     A,19
 0943 77           323+        MOV     M,A
 0944 23           324+        INX     H
                   325         STP
 0945 3E30         326+        MVI     A,30H
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  0947 77           327+        MOV     M,A
  0948 CD290F       328         CALL    PREP
  094B 3A813C       329         LDA     QUASI
  094E 326D3C       330         STA     DATA+6
  0951 C3480C       331         JMP     LOC4
                    332 ;
                    333 ;
                    334 ;       'I'COMMAND(SCHEDULE)
                    335 ;       INTERCONNECTION FREQUENCY (CHANNEL 20) AC
                    336 ;
                    337 ICMD:
                    338         FAST
  0954 3E01         339+        MVI     A,1
  0956 329C3C       340+        STA     TIME
  0959 21513C       341         LXI     H,TEMP
  095C 3E32         342         MVI     A,32H
  095E 77           343         MOV     M,A
  095F 23           344         INX     H
                    345         ONE     20
  0960 3E14         346+        MVI     A,20
  0962 77           347+        MOV     M,A
  0963 23           348+        INX     H
                    349         STP
  0964 3E30         350+        MVI     A,30H
  0966 77           351+        MOV     M,A
  0967 CD290F       352         CALL    PREP
  096A 3A813C       353         LDA     QUASI
  096D 326E3C       354         STA     DATA+7
  0970 C3480C       355         JMP     LOC4
                    356 ;
                    357 ;       'J'COMMAND(SCHEDULE)
                    358 ;       GEN. 1 ?HASE U.R.T INTERCONNECTION (CHANNELS 17,20)AC,AC
                    359 ;
                    360 JCMD:
                    361         FAST
  0973 3E01         362+        MVI     A,1
  0975 329C3C       363+        STA     TIME
  0978 21513C       364         LXI     H,TEMP
  097B 3E31         365         MVI     A,31H
  097D 77           366         MOV     M,A
                    367         TWO     17,20
  097E 23           368+        INX     H
  097F 3E11         369+        MVI     A,17
  0981 77           370+        MOV     M,A
  0982 23           371+        INX     H
  0983 3E14         372+        MVI     A,20
  0985 77           373+        MOV     M,A
  LOC  OBJ         LINE        SOURCE STATEMENT

  0986 23           374         INX     H
                    375         STP
  0987 3E30         376+        MVI     A,30H
  0989 77           377+        MOV     M,A
  098A CD290F       378         CALL    PREP
  098D 21813C       379         LXI     H,QUASI
  0990 7E           380         MOV     A,M
  0991 326F3C       381         STA     DATA+8
  0994 23           382         INX     H
  0995 7E           383         MOV     A,M
  0996 32703C       384         STA     DATA+9
  0999 C3480C       385         JMP     LOC4
                    386 ;
                    387 ;       'K'COMMAND(SCHEDULE)
                    388 ;       GEN.2 PHASE W.R.T INTERCONNECTION (CHANNELS 18,20) AC,AC
                    389 ;
                    390 KCMD:
                    391         FAST
  099C 3E01         392+        MVI     A,1
  099E 329C3C       393+        STA     TIME
  09A1 21513C       394         LXI     H,TEMP
  09A4 3E31         395         MVI     A,31H
  09A6 77           396         MOV     M,A
                    397         TWO     18,20
  09A7 23           398+        INX     H
  09A8 3E12         399+        MVI     A,18
  09AA 77           400+        MOV     M,A
  09AB 23           401+        INX     H
  09AC 3E14         402+        MVI     A,20
  09AE 77           403+        MOV     M,A
  09AF 23           404         INX     H
                    405         STP
  09B0 3E30         406+        MVI     A,30H
  09B2 77           407+        MOV     M,A
  09B3 CD290F       408         CALL    PREP
  09B6 21813C       409         LXI     H,QUASI
  09B9 7E           410         MOV     A,M
  09BA 32713C       411         STA     DATA+10
  09BD 23           412         INX     H
  09BE 7E           413         MOV     A,M
  09BF 32723C       414         STA     DATA+11
  09C2 C3480C       415         JMP     LOC4
                    416 ;
                    417 ;       'L'COMMAND(SCHEDULE)
                    418 ;       GEN.3 PHASE W.R.T INTERCONNECTION (CHANNELS 19,20) AC,AC
                    419 ;
                    420 LCMD:
```

```
 LOC  OBJ          LINE        SOURCE STATEMENT

 09C5 21513C        421         LXI     H,TEMP
 09C8 3E31          422         MVI     A,31H
 09CA 77            423         MOV     M,A
                    424         TWO     19,20
 09CB 23            425+        INX     H
 09CC 3E13          426+        MVI     A,19
 09CE 77            427+        MOV     M,A
 09CF 23            428+        INX     H
 09D0 3E14          429+        MVI     A,20
 09D2 77            430+        MOV     M,A
 09D3 23            431         INX     H
                    432         STP
 09D4 3E30          433+        MVI     A,30H
 09D6 77            434+        MOV     M,A
 09D7 CD290F        435         CALL    PREP
 09DA 21813C        436         LXI     H,QUASI
 09DD 7E            437         MOV     A,M
 09DE 32733C        438         STA     DATA+12
 09E1 23            439         INX     H
 09E2 7E            440         MOV     A,M
 09E3 32743C        441         STA     DATA+13
 09E6 C3480C        442         JMP     LOC4
                    443 ;
                    444 ;       'H'COMMAND(SCHEDULE)
                    445 ;       GENERATOR 1 CURRENT (CHANNEL 2) DC
                    446 ;
                    447 HCMD:
                    448         FAST
 09E9 3E01          449+        MVI     A,1
 09EB 329C3C        450+        STA     TIME
 09EE 21513C        451         LXI     H,TEMP
                    452         ONE     2
 09F1 3E02          453+        MVI     A,2
 09F3 77            454+        MOV     M,A
 09F4 23            455+        INX     H
                    456         STP
 09F5 3E30          457+        MVI     A,30H
 09F7 77            458+        MOV     M,A
 09F8 CD290F        459         CALL    PREP
 09FB 3A813C        460         LDA     QUASI
 09FE 32753C        461         STA     DATA+14
 0A01 C3480C        462         JMP     LOC4
                    463 ;
                    464 ;       'N'COMMAND(SCHEDULE)
                    465 ;       GENERATOR 2 CURRENT(CHANNEL 4) DC
                    466 ;
                    467 NCMD:
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

                    468         FAST
 0A04 3E01          469+        MVI     A,1
 0A06 329C3C        470+        STA     TIME
 0A09 21513C        471         LXI     H,TEMP
                    472         ONE     4
 0A0C 3E04          473+        MVI     A,4
 0A0E 77            474+        MOV     M,A
 0A0F 23            475+        INX     H
                    476         STP
 0A10 3E30          477+        MVI     A,30H
 0A12 77            478+        MOV     M,A
 0A13 CD290F        479         CALL    PREP
 0A16 3A813C        480         LDA     QUASI
 0A19 32763C        481         STA     DATA+15
 0A1C C3480C        482         JMP     LOC4
                    483 ;
                    484 ;       'O'COMMAND(SCHEDULE)
                    485 ;       GENERATOR 3 CURRENT (CHANNEL 6) DC
                    486 ;
                    487 OCMD:
                    488         FAST
 0A1F 3E01          489+        MVI     A,1
 0A21 329C3C        490+        STA     TIME
 0A24 21513C        491         LXI     H,TEMP
                    492         ONE     6
 0A27 3E06          493+        MVI     A,6
 0A29 77            494+        MOV     M,A
 0A2A 23            495+        INX     H
                    496         STP
 0A2B 3E30          497+        MVI     A,30H
 0A2D 77            498+        MOV     M,A
 0A2E CD290F        499         CALL    PREP
 0A31 3A813C        500         LDA     QUASI
 0A34 32773C        501         STA     DATA+16
 0A37 C3480C        502         JMP     LOC4
                    503 ;
                    504 ;       'P'COMMAND(SCHEDULE)
                    505 ;       GEN.1 VOLTAGE(1),GEN.1 CURRENT(2),FREQUENCY(17)
                    506 ;       AND PHASE W.R.T INTERCONNECTION (17,20)
                    507 ;
                    508 PCMD:
 0A3A 21513C        509         LXI     H,TEMP
 0A3D 3E01          510         MVI     A,1
 0A3F 77            511         MOV     M,A
 0A40 23            512         INX     H
 0A41 3E02          513         MVI     A,2
 0A43 77            514         MOV     M,A
```

```
LOC  OBJ         LINE        SOURCE STATEMENT

0A44 23           515         INX     H
0A45 3E32         516         MVI     A,32H
0A47 77           517         MOV     M,A
0A48 23           518         INX     H
0A49 3E11         519         MVI     A,17
0A4B 77           520         MOV     M,A
0A4C 23           521         INX     H
0A4D 3E31         522         MVI     A,31H
0A4F 77           523         MOV     M,A
                  524         TWO     17,20
0A50 23           525+        INX     H
0A51 3E11         526+        MVI     A,17
0A53 77           527+        MOV     M,A
0A54 23           528+        INX     H
0A55 3E14         529+        MVI     A,20
0A57 77           530+        MOV     M,A
0A58 23           531         INX     H
                  532         STP
0A59 3E30         533+        MVI     A,30H
0A5B 77           534+        MOV     M,A
0A5C CD290F       535         CALL    PREP
0A5F 21813C       536         LXI     H,QUASI
0A62 7E           537         MOV     A,M
0A63 32673C       538         STA     DATA
0A66 23           539         INX     H
0A67 7E           540         MOV     A,M
0A68 32753C       541         STA     DATA+14
0A6B 23           542         INX     H
0A6C 7E           543         MOV     A,M
0A6D 326B3C       544         STA     DATA+4
0A70 23           545         INX     H
0A71 7E           546         MOV     A,M
0A72 326F3C       547         STA     DATA+8
0A75 23           548         INX     H
0A76 7E           549         MOV     A,M
0A77 32703C       550         STA     DATA+9
0A7A C3480C       551         JMP     LOC4
                  552 ;
                  553 ;
                  554 ;       'Q' COMMAND (SCHEDULE)
                  555 ;       GEN2 VOLTAGE(3),CURRENT(4), FREQUENCY(18)
                  556 ;       AND PHASE W.R.T INTERCONNECTION (18,20)
                  557 ;
                  558 QCMD:
0A7D 21513C       559         LXI     H,TEMP
0A80 3E03         560         MVI     A,3
0A82 77           561         MOV     M,A
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  0A83 23           562          INX     H
  0A84 3E04         563          MVI     A,4
  0A86 77           564          MOV     M,A
  0A87 23           565          INX     H
  0A88 3E32         566          MVI     A,32H
  0A8A 77           567          MOV     M,A
  0A8B 23           568          INX     H
  0A8C 3E12         569          MVI     A,18
  0A8E 77           570          MOV     M,A
  0A8F 23           571          INX     H
  0A90 3E31         572          MVI     A,31H
  0A92 77           573          MOV     M,A
                    574          TWO     18,20
  0A93 23           575+         INX     H
  0A94 3E12         576+         MVI     A,18
  0A96 77           577+         MOV     M,A
  0A97 23           578+         INX     H
  0A98 3E14         579+         MVI     A,20
  0A9A 77           580+         MOV     M,A
  0A9B 23           581          INX     H
                    582          STP
  0A9C 3E30         583+         MVI     A,30H
  0A9E 77           584+         MOV     M,A
  0A9F CD290F       585          CALL    PREP
  0AA2 21813C       586          LXI     H,QUASI
  0AA5 7E           587          MOV     A,M
  0AA6 32683C       588          STA     DATA+1
  0AA9 23           589          INX     H
  0AAA 7E           590          MOV     A,M
  0AAB 32763C       591          STA     DATA+15
  0AAE 23           592          INX     H
  0AAF 7E           593          MOV     A,M
  0AB0 326C3C       594          STA     DATA+5
  0AB3 23           595          INX     H
  0AB4 7E           596          MOV     A,M
  0AB5 32713C       597          STA     DATA+10
  0AB8 23           598          INX     H
  0AB9 7E           599          MOV     A,M
  0ABA 32723C       600          STA     DATA+11
  0ABD C3480C       601          JMP     LOC4
                    602 ;
                    603 ;        'R'COMMAND(SCHEDULE)
                    604 ;        GEN3 VOLTAGE(5),CURRENT(6),FREQUENCY(19)
                    605 ;        AND PHASE W.R.T INTERCONNECTION (19,20)
                    606 ;
                    607 RCMD:
  0AC0 21513C       608          LXI     H,TEMP
```

```
LOC  OBJ         LINE        SOURCE STATEMENT

0AC3 3E05         609         MVI     A,5
0AC5 77           610         MOV     M,A
0AC6 23           611         INX     H
0AC7 3E06         612         MVI     A,6
0AC9 77           613         MOV     M,A
0ACA 23           614         INX     H
0ACB 3E32         615         MVI     A,32H
0ACD 77           616         MOV     M,A
0ACE 23           617         INX     H
0ACF 3E13         618         MVI     A,19
0AD1 77           619         MOV     M,A
0AD2 23           620         INX     H
0AD3 3E31         621         MVI     A,31H
0AD5 77           622         MOV     M,A
                  623         TWO     19,20
0AD6 23           624+        INX     H
0AD7 3E13         625+        MVI     A,19
0AD9 77           626+        MOV     M,A
0ADA 23           627+        INX     H
0ADB 3E14         628+        MVI     A,20
0ADD 77           629+        MOV     M,A
0ADE 23           630         INX     H
                  631         STP
0ADF 3E30         632+        MVI     A,30H
0AE1 77           633+        MOV     M,A
0AE2 CD290F       634         CALL    PREP
0AE5 21813C       635         LXI     H,QUASI
0AE8 7E           636         MOV     A,M
0AE9 32693C       637         STA     DATA+2
0AEC 23           638         INX     H
0AED 7E           639         MOV     A,M
0AEE 32773C       640         STA     DATA+16
0AF1 23           641         INX     H
0AF2 7E           642         MOV     A,M
0AF3 326D3C       643         STA     DATA+6
0AF6 23           644         INX     H
0AF7 7E           645         MOV     A,M
0AF8 32733C       646         STA     DATA+12
0AFB 23           647         INX     H
0AFC 7E           648         MOV     A,M
0AFD 32743C       649         STA     DATA+13
0B00 C3480C       650         JMP     LOC4
                  651 ;
                  652 ;       'S'COMMAND(SCHEDULE)
                  653 ;       GEN1 FREQUENCY(17),GEN2 FREQUENCY(18),GEN3 FREQUENCY(19)
                  654 ;       AND INTER CONNECTION FREQUENCY(20)
                  655 ;
```

```
LOC  OBJ         LINE        SOURCE STATEMENT

                  656 SCMD:
0B03 21513C       657         LXI      H,TEMP
0B06 3E32         658         MVI      A,32H
0B08 77           659         MOV      M,A
0B09 23           660         INX      H
0B0A 3E11         661         MVI      A,17
0B0C 77           662         MOV      M,A
0B0D 23           663         INX      H
0B0E 3E32         664         MVI      A,32H
0B10 77           665         MOV      M,A
0B11 23           666         INX      H
0B12 3E12         667         MVI      A,18
0B14 77           668         MOV      M,A
0B15 23           669         INX      H
0B16 3E32         670         MVI      A,32H
0B18 77           671         MOV      M,A
0B19 23           672         INX      H
0B1A 3E13         673         MVI      A,19
0B1C 77           674         MOV      M,A
0B1D 23           675         INX      H
0B1E 3E32         676         MVI      A,32H
0B20 77           677         MOV      M,A
0B21 23           678         INX      H
0B22 3E14         679         MVI      A,20
0B24 77           680         MOV      M,A
0B25 23           681         INX      H
0B26 3E30         682         MVI      A,30H
0B28 77           683         MOV      M,A
0B29 CD290F       684         CALL     PREP
0B2C 0604         685         MVI      B,4
0B2E 21813C       686         LXI      H,QUASI
0B31 116B3C       687         LXI      D,DATA+4
0B34 CD810F       688         CALL     FILL
0B37 C3480C       689         JMP      LOC4
                  690 ;
                  691 ;       'T'COMMAND(SCHEDULE)
                  692 ;       GEN1 VOLTAGE(1),GEN2 VOLTAGE(3),GEN3 VOLTAGE(5)
                  693 ;       AND INTER CONNECTION VOLTAGE(7)
                  694 ;
                  695 TCMD:
0B3A 21503C       696         LXI      H,TEMP-1
                  697         TWO      1,3
0B3D 23           698+        INX      H
0B3E 3E01         699+        MVI      A,1
0B40 77           700+        MOV      M,A
0B41 23           701+        INX      H
0B42 3E03         702+        MVI      A,3
```

```
LOC  OBJ         LINE        SOURCE STATEMENT

0B44 77           703+        MOV     M,A
0B45 2B           704         DCX     H
                  705         TWO     5,7
0B46 23           706+        INX     H
0B47 3E05         707+        MVI     A,5
0B49 77           708+        MOV     M,A
0B4A 23           709+        INX     H
0B4B 3E07         710+        MVI     A,7
0B4D 77           711+        MOV     M,A
0B4E 23           712         INX     H
                  713         STP
0B4F 3E30         714+        MVI     A,30H
0B51 77           715+        MOV     M,A
0B52 CD290F       716         CALL    PREP
0B55 0604         717         MVI     B,4
0B57 21813C       718         LXI     H,QUASI
0B5A 11673C       719         LXI     D,DATA
0B5B CD810F       720         CALL    FILL
0B60 C3480C       721         JMP     LOC4
                  722 ;
                  723 ;       'U'COMMAND(SCHEDULE)
                  724 ;       SUBSTATION 4 RIGHT BUS(9),STATION-1 BUS(10)
                  725 ;       INDUSTRIAL LOAD VOLTAGE(11),LINE B1-3 VOLTAGE(13)
                  726 ;       SUBSTATION 6 BUS(14),NETWORK SYSYTEM L-N1(15)
                  727 ;
                  728 UCMD:
0B63 21503C       729         LXI     H,TEMP-1
                  730         TWO     9,10
0B66 23           731+        INX     H
0B67 3E09         732+        MVI     A,9
0B69 77           733+        MOV     M,A
0B6A 23           734+        INX     H
0B6B 3E0A         735+        MVI     A,10
0B6D 77           736+        MOV     M,A
                  737         TWO     11,13
0B6E 23           738+        INX     H
0B6F 3E0B         739+        MVI     A,11
0B71 77           740+        MOV     M,A
0B72 23           741+        INX     H
0B73 3E0B         742+        MVI     A,13
0B75 77           743+        MOV     M,A
                  744         TWO     14,15
0B76 23           745+        INX     H
0B77 3E0E         746+        MVI     A,14
0B79 77           747+        MOV     M,A
0B7A 23           748+        INX     H
0B7B 3E0F         749+        MVI     A,15
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  0B7D 77           750+        MOV     M,A
  0B7E 23           751         INX     H
                    752         STP
  0B7F 3E30         753+        MVI     A,30H
  0B81 77           754+        MOV     M,A
  0B82 CD290F       755         CALL    PREP
  0B85 0606         756         MVI     B,6
  0B87 21813C       757         LXI     H,QUASI
  0B8A 11783C       758         LXI     D,DATA+17
  0B8D CD810F       759         CALL    FILL
  0B90 C3480C       760         JMP     LOC4
                    761 ;
                    762 ;       'V'COMMAND(SCHEDULE)
                    763 ;       INTER CONNECTION CURRENT(8), INDUSTRIAL LOAD
                    764 ;       CURRENT (12) AND NETWORK SYSTEM CURRENT LINE1 (16)
                    765 ;
                    766 VCMD:
  0B93 21503C       767         LXI     H,TEMP-1
                    768         TWO     8,12
  0B96 23           769+        INX     H
  0B97 3E08         770+        MVI     A,8
  0B99 77           771+        MOV     M,A
  0B9A 23           772+        INX     H
  0B9B 3E0C         773+        MVI     A,12
  0B9D 77           774+        MOV     M,A
  0B9E 23           775         INX     H
                    776         ONE     16
  0B9F 3E10         777+        MVI     A,16
  0BA1 77           778+        MOV     M,A
  0BA2 23           779+        INX     H
                    780         STP
  0BA3 3E30         781+        MVI     A,30H
  0BA5 77           782+        MOV     M,A
  0BA6 CD290F       783         CALL    PREP
  0BA9 0603         784         MVI     B,3
  0BAB 21813C       785         LXI     H,QUASI
  0BAE 117E3C       786         LXI     D,DATA+23
  0BB1 CD810F       787         CALL    FILL
  0BB4 C3480C       788         JMP     LOC4
                    789 ;
                    790 ;       'U'COMMAND(SCHEDULE)
                    791 ;       ALL OTHER VOLTAGES INDICATED IN SCHEDULES T AND U
                    792 ;
                    793 UCMD:
  0BB7 21503C       794         LXI     H,TEMP-1
                    795         TWO     1,3
  0BBA 23           796+        INX     H
  LOC  OBJ          LINE        SOURCE STATEMENT

  0BBB 3E01          797+        MVI     A,1
  0BBD 77            798+        MOV     M,A
  0BBE 23            799+        INX     H
  0BBF 3E03          800+        MVI     A,3
  0BC1 77            801+        MOV     M,A
                     802         TWO     5,7
  0BC2 23            803+        INX     H
  0BC3 3E05          804+        MVI     A,5
  0BC5 77            805+        MOV     M,A
  0BC6 23            806+        INX     H
  0BC7 3E07          807+        MVI     A,7
  0BC9 77            808+        MOV     M,A
                     809         TWO     9,10
  0BCA 23            810+        INX     H
  0BCB 3E09          811+        MVI     A,9
  0BCD 77            812+        MOV     M,A
  0BCE 23            813+        INX     H
  0BCF 3E0A          814+        MVI     A,10
  0BD1 77            815+        MOV     M,A
                     816         TWO     11,13
  0BD2 23            817+        INX     H
  0BD3 3E0B          818+        MVI     A,11
  0BD5 77            819+        MOV     M,A
  0BD6 23            820+        INX     H
  0BD7 3E0D          821+        MVI     A,13
  0BD9 77            822+        MOV     M,A
                     823         TWO     14,15
  0BDA 23            824+        INX     H
  0BDB 3E0E          825+        MVI     A,14
  0BDD 77            826+        MOV     M,A
  0BDE 23            827+        INX     H
  0BDF 3E0F          828+        MVI     A,15
  0BE1 77            829+        MOV     M,A
  0BE2 23            830         INX     H
                     831         STP
  0BE3 3E30          832+        MVI     A,30H
  0BE5 77            833+        MOV     M,A
  0BE6 CD290F        834         CALL    PREP
  0BE9 0604          835         MVI     B,4
  0BEB 21813C        836         LXI     H,QUASI
  0BEE 11673C        837         LXI     D,DATA
  0BF1 CD810F        838         CALL    FILL
  0BF4 0606          839         MVI     B,6
  0BF6 11783C        840         LXI     D,DATA+17
  0BF9 CD810F        841         CALL    FILL
  0BFC C3480C        842         JMP     LOC4
                     843 ;
  LOC  OBJ         LINE        SOURCE STATEMENT

                    844 ;       'X'COMMAND(SCHEDULE)
                    845 ;       ALL OTHER CURRENTS INDICATED IN SCHEDULES M,N,O AND V
                    846 ;
                    847 XCMD:
 0BFF 21503C        848         LXI     H,TEMP-1
                    849         TWO     2,4
 0C02 23            850+        INX     H
 0C03 3E02          851+        MVI     A,2
 0C05 77            852+        MOV     M,A
 0C06 23            853+        INX     H
 0C07 3E04          854+        MVI     A,4
 0C09 77            855+        MOV     M,A
                    856         TWO     6,8
 0C0A 23            857+        INX     H
 0C0B 3E06          858+        MVI     A,6
 0C0D 77            859+        MOV     M,A
 0C0E 23            860+        INX     H
 0C0F 3E08          861+        MVI     A,8
 0C11 77            862+        MOV     M,A
                    863         TWO     12,16
 0C12 23            864+        INX     H
 0C13 3E0C          865+        MVI     A,12
 0C15 77            866+        MOV     M,A
 0C16 23            867+        INX     H
 0C17 3E10          868+        MVI     A,16
 0C19 77            869+        MOV     M,A
 0C1A 23            870         INX     H
                    871         STP
 0C1B 3E30          872+        MVI     A,30H
 0C1D 77            873+        MOV     M,A
 0C1E CD290F        874         CALL    PREP
 0C21 0603          875         MVI     B,3
 0C23 21813C        876         LXI     H,QUASI
 0C26 11753C        877         LXI     D,DATA+14
 0C29 CD810F        878         CALL    FILL
 0C2C 0603          879         MVI     B,3
 0C2E 117E3C        880         LXI     D,DATA+23
 0C31 CD810F        881         CALL    FILL
 0C34 C3480C        882         JMP     LOC4
                    883 ;
                    884 ;       FOR FUTURE EXPANSION
                    885 ;
                    886 YCMD:
 0C37 00            887         NOP
 0C38 C3A10F        888         JMP     ERROR
                    889 ;
                    890 ;       FOR FUTURE EXPANSION
 LOC  OBJ          LINE        SOURCE STATEMENT

                    891 ;
                    892 ZCMD:
 0C3B 00            893         NOP
 0C3C C3A10F        894         JMP     ERROR
                    895 ;
                    896 ;       'C'COMMAND(SCHEDULE)
                    897 ;       CONTINUE COMMAND.. SAMPLES PREVIOUSLY
                    898 ;       DETERMINED SCHEDULE FOR  3 TIMES
                    899 ;
 0C3F 00            900 CCMD:   NOP
 0C40 C3A10F        901         JMP     ERROR   ;FIRST COMMAND CANNOT BE A C COMMAND
                    902 CCMD1:
 0C43 2A9D3C        903         LHLD    SLEP
 0C46 E9            904         PCHL
 0C47 00            905 LOC3:   NOP
                    906 LOC4:
 0C48 CD640C        907         CALL    XMIT
 0C4B CD060D        908         CALL    IDLE
 0C4E 3A8C3C        909         LDA     NUMB
 0C51 3D            910         DCR     A
 0C52 CA5B0C        911         JZ      LOC6
 0C55 328C3C        912         STA     NUMD
 0C58 C3430C        913         JMP     CCMD1
 0C5D 3E03          914 LOC6:   MVI     A,3
 0C5D 328C3C        915         STA     NUMB
                    916 LOC5:
 0C60 00            917         NOP
 0C61 C3A70F        918         JMP     DONE
                    919 ;
                    920 ;
                    921 ;       SUBROUTINE;..XMIT
                    922 ;
                    923 XMIT:
 0C64 CDB10C        924         CALL    DELAY
 0C67 D5            925         PUSH    D
 0C68 E5            926         PUSH    H
 0C69 0E50          927 L5:     MVI     C,'P'
 0C6B CDFA03        928         CALL    CO
 0C6E 0E0D          929         MVI     C,0DH
 0C70 CDFA03        930         CALL    CO
 0C73 CDB10C        931         CALL    DELAY
 0C76 CDB10C        932         CALL    DELAY
 0C79 CDB10C        933         CALL    DELAY
 0C7C CDB10C        934         CALL    DELAY
 0C7F CDB10C        935         CALL    DELAY
 0C82 0E55          936         MVI     C,'U'
 0C84 CDFA03        937         CALL    CO
 LOC  OBJ         LINE        SOURCE STATEMENT

 0C87 0E4F         938          MVI     C,'O'
 0C89 CDFA03       939          CALL    CO
 0C8C 21673C       940          LXI     H,DATA
 0C8F 0600         941          MVI     B,0     ;INITIALIZE B
 0C91 161A         942          MVI     D,26
 0C93 7E           943 L6:      MOV     A,M
 0C94 80           944          ADD     B
 0C95 47           945          MOV     B,A
 0C96 7E           946          MOV     A,M
 0C97 CDC50C       947          CALL    CONV
 0C9A 23           948          INX     H
 0C9B 15           949          DCR     D
 0C9C C2930C       950          JNZ     L6
 0C9F 78           951          MOV     A,B
 0CA0 2F           952          CMA             ;COMPLEMENT THE SUM OF 35 BYTES OF DATA
 0CA1 CDC50C       953          CALL    CONV
 0CA4 0E24         954          MVI     C,'$'
 0CA6 CDFA03       955          CALL    CO
 0CA9 0E0D         956          MVI     C,0DH   ;CARRIAGE RETURN
 0CAB CDFA03       957          CALL    CO
 0CAE E1           958          POP     H
 0CAF D1           959          POP     D
 0CB0 C9           960          RET
                   961 ;
                   962 ;
                   963 ;        SUBROUTINE..DELAY
                   964 ;
                   965 ;
                   966 DELAY:
 0CB1 F5           967          PUSH    PSW
 0CB2 C5           968          PUSH    B
 0CB3 3A9F3C       969          LDA     MARCH
 0CB6 47           970          MOV     B,A
 0CB7 3AA03C       971 APR:     LDA     MARCH+1
 0CBA 3D           972 MAY:     DCR     A
 0CBB C2BA0C       973          JNZ     MAY
 0CBE 05           974          DCR     B
 0CBF C2B70C       975          JNZ     APR
 0CC2 C1           976          POP     B
 0CC3 F1           977          POP     PSW
 0CC4 C9           978          RET
                   979 ;
                   980 ;
                   981 ;        SUBROUTINE..CONV
                   982 ;        OUTPUTS ASCII EQUIVALENTS OF BOTH HIGHER AND LOWER 4BITS
                   983 ;
                   984 CONV:
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  0CC5 C5           985         PUSH    B
  0CC6 47           986         MOV     B,A
  0CC7 E6F0         987         ANI     0F0H
  0CC9 0F           988         RRC
  0CCA 0F           989         RRC
  0CCB 0F           990         RRC
  0CCC 0F           991         RRC
  0CCD FE0A         992         CPI     0AH
  0CCF FAE90C       993         JM      L10
  0CD2 C637         994         ADI     37H
                    995 L11:
  0CD4 4F           996         MOV     C,A
  0CD5 CDFA03       997         CALL    CO
  0CD8 78           998         MOV     A,B
  0CD9 E60F         999         ANI     0FH
  0CDB FE0A        1000         CPI     0AH
  0CDD FAEE0C      1001         JM      L12
  0CE0 C637        1002         ADI     37H
                   1003 L13:
  0CE2 4F          1004         MOV     C,A
  0CE3 CDFA03      1005         CALL    CO
  0CE6 C3F30C      1006         JMP     L14
  0CE9 C630        1007 L10:    ADI     30H
  0CEB C3D40C      1008         JMP     L11
  0CEE C630        1009 L12:    ADI     30H
  0CF0 C3E20C      1010         JMP     L13
  0CF3 C1          1011 L14:    POP     B
  0CF4 C9          1012         RET
                   1013 ;
                   1014 ;
                   1015 ;        SUBROUTINE..PRNT
                   1016 ;
                   1017 PRNT:
  0CF5 C5          1018         PUSH    B
  0CF6 7E          1019 L15:    MOV     A,M
  0CF7 FE00        1020         CPI     0
  0CF9 CA040D      1021         JZ      L16
  0CFC 4F          1022         MOV     C,A
  0CFD CDFA03      1023         CALL    CO
  0D00 23          1024         INX     H
  0D01 C3F60C      1025         JMP     L15
  0D04 C1          1026 L16:    POP     B
  0D05 C9          1027         RET
                   1028 ;
                   1029 ;
                   1030 ;        SUBROUTINE..IDLE
                   1031 ;        THE IDLE TIME CAN BE CHANGED BY CHANGING THE
  LOC  OBJ         LINE        SOURCE STATEMENT

                   1032 ;       DATA STORED AT LOCATION AVG
                   1033 ;       UNIT INCREMENT OF AVG LOCATION CONTENT
                   1034 ;       WILL DELAY BY APPROXIMATELY BY 0.25 SECS.
                   1035 ;
                   1036 IDLE:
  0D06 F5          1037         PUSH    PSW
  0D07 C5          1038         PUSH    B
  0D08 D5          1039         PUSH    D
  0D09 3AA13C      1040 L24:    LDA     AVG
  0D0C 57          1041         MOV     D,A
  0D0D 0640        1042 L22:    MVI     B,40H   ;DELAY APPROX. 0.5 SECS
  0D0F 0EFF        1043 L21:    MVI     C,0FFH
  0D11 0D          1044 L20:    DCR     C
  0D12 C2110D      1045         JNZ     L20
  0D15 05          1046         DCR     B
  0D16 C20F0D      1047         JNZ     L21
  0D19 15          1048         DCR     D
  0D1A C20D0D      1049         JNZ     L22
  0D1D 3A9C3C      1050         LDA     TIME
  0D20 3D          1051         DCR     A
  0D21 CA2F0D      1052         JZ      L23
  0D24 329C3C      1053         STA     TIME
  0D27 C3090D      1054         JMP     L24
  0D2A 3E02        1055         MVI     A,2
  0D2C 329C3C      1056         STA     TIME
  0D2F D1          1057 L23:    POP     D
  0D30 C1          1058         POP     B
  0D31 F1          1059         POP     PSW
  0D32 C9          1060         RET
                   1061 ;
                   1062 ;
                   1063 ;
                   1064 ;        SUBROUTINE..SAMP3
                   1065 SAMP3:
  0D33 11673C      1066         LXI     D,DATA
  0D36 21FA8F      1067         LXI     H,MUXADR
  0D39 3E00        1068         MVI     A,0
  0D3B 77          1069         MOV     M,A
  0D3C 32FB8F      1070 NEXT:   STA     CNVCMD
  0D3F 34          1071         INR     M
  0D40 3AFC8F      1072 LOOP:   LDA     STATUS
  0D43 07          1073         RLC
  0D44 D2400D      1074         JNC     LOOP
  0D47 3AF88F      1075         LDA     ADC8
  0D4A 12          1076         STAX    D
  0D4B 13          1077         INX     D
  0D4C 7E          1078         MOV     A,M
```

```
 LOC  OBJ         LINE        SOURCE STATEMENT

 0D4B FE20        1079          CPI     20H
 0D4F C23C0D      1080          JNZ     NEXT
 0D52 C9          1081          RET
                  1082 ;
                  1083 ;
                  1084 ;         SUBROUTINE .. FREQ
                  1085 ;
                  1086 FREQ:
 0D53 F5          1087          PUSH    PSW
 0D54 C5          1088          PUSH    B
 0D55 D5          1089          PUSH    D
 0D56 E5          1090          PUSH    H
 0D57 0E00        1091          MVI     C,0
 0D59 CD980D      1092 FREQ1:   CALL    SAMP1
 0D5C FE80        1093          CPI     80H
 0D5E FA590D      1094          JM      FREQ1
 0D61 47          1095          MOV     B,A
 0D62 D680        1096          SUI     80H
 0D64 FE03        1097          CPI     3
 0D66 F2590D      1098          JP      FREQ1
 0D69 CD980D      1099          CALL    SAMP1
 0D6C B8          1100          CMP     B
 0D6D FA590D      1101          JM      FREQ1
 0D70 CA590D      1102          JZ      FREQ1
 0D73 C3810D      1103          JMP     PP1
 0D76 0C          1104 FREQ2:   INR     C
 0D77 CD980D      1105          CALL    SAMP1
 0D7A B8          1106          CMP     B
 0D7B FA8F0D      1107          JM      P4
 0D7E C3860D      1108          JMP     PP3
 0D81 00          1109 PP1:     NOP
 0B82 00          1110          NOP
 0D83 C3760D      1111          JMP     FREQ2
 0D86 00          1112 PP3:     NOP
 0D87 00          1113          NOP
 0D88 C38B0D      1114          JMP     PP4
 0D8B 00          1115 PP4:     NOP
 0D8C C3760D      1116          JMP     FREQ2
 0D8F 79          1117 P4:      MOV     A,C
 0D90 32493C      1118          STA     RESLT
 0D93 E1          1119          POP     H
 0D94 D1          1120          POP     D
 0D95 C1          1121          POP     B
 0B96 F1          1122          POP     PSW
 0D97 C9          1123          RET
                  1124 ;
                  1125 ;
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

                   1126 ;       SUBROUTINE..SAMP1
                   1127 ;
                   1128 SAMP1:
  0D98 3E00        1129         MVI     A,0
  0D9A 32F98F      1130         STA     GNSEL   ;SET GAIN TO 1
  0D9D 32F08F      1131         STA     SETUP   ;SET PACER OFF
  0DA0 3A4F3C      1132         LDA     CHAN    ;SELECT THE CHANNEL
  0DA3 32FA8F      1133         STA     MUXADR
  0DA6 32FB8F      1134         STA     CNVCMD  ;START A/D CONVERSION
  0DA9 3AFC8F      1135 KAT:    LDA     STATUS
  0DAC 07          1136         RLC
  0DAD D2A90D      1137         JNC     KAT     ;IS SAMPLING DONE?
  0DB0 3AF88F      1138         LDA     ADC8    ;READ ADC DATA
  0DB3 C680        1139         ADI     80H
  0DB5 C9          1140         RET
                   1141 ;
                   1142 ;
                   1143 ;       SUBROUTINE..AMPLT
                   1144 ;
                   1145 ;
                   1146 AMPLT:
  0DB6 C5          1147         PUSH    B
  0DB7 E5          1148         PUSH    H
  0DB8 21A23C      1149         LXI     H,3CA2H
  0DBB 0E06        1150         MVI     C,6
  0DBD CD980D      1151 AMPL1:  CALL    SAMP1
  0DC0 FE80        1152         CPI     80H
  0DC2 FABD0D      1153         JM      AMPL1
  0DC5 47          1154         MOV     B,A
  0DC6 D680        1155         SUI     80H
  0DC8 FE03        1156         CPI     3
  0DCA F2BD0D      1157         JP      AMPL1
  0DCD CD980D      1158 AMPL2:  CALL    SAMP1
  0DD0 B8          1159         CMP     B
  0DD1 FABD0D      1160         JM      AMPL1
  0DD4 CABD0D      1161         JZ      AMPL1
  0DD7 47          1162         MOV     B,A
  0DD8 CD980D      1163 AMPL3:  CALL    SAMP1
  0DDB B8          1164         CMP     B
  0DDC FAE30D      1165         JM      AMPL4
  0DDF 47          1166         MOV     B,A
  0DE0 C3D80D      1167         JMP     AMPL3
  0DE3 78          1168 AMPL4:  MOV     A,B
  0DE4 77          1169         MOV     M,A
  0DE5 23          1170         INX     H
  0DE6 0D          1171         DCR     C
  0DE7 C2BD0D      1172         JNZ     AMPL1
```

```
 LOC  OBJ          LINE        SOURCE STATEMENT

 0DEA 21A23C       1173         LXI     H,3CA2H
 0DED 0E06         1174         MVI     C,6
 0DEF 7E           1175         MOV     A,M
 0DF0 23           1176         INX     H
 0DF1 0D           1177         DCR     C
 0DF2 46           1178 AMPL7:  MOV     B,M
 0DF3 B8           1179         CMP     B
 0DF4 FA020E       1180         JM      AMPL5   ;B>A
 0DF7 F20B0E       1181         JP      AMPL6   ;A>B
 0DFA 23           1182         INX     H       ;A=B
 0DFB 0D           1183         DCR     C
 0DFC C2F20D       1184         JNZ     AMPL7
 0DFF C3100E       1185         JMP     AMPL8
 0E02 78           1186 AMPL5:  MOV     A,B     ;B>A
 0E03 23           1187         INX     H
 0E04 0D           1188         DCR     C
 0E05 C2F20D       1189         JNZ     AMPL7
 0E08 C3100E       1190         JMP     AMPL8
 0E0B 23           1191 AMPL6:  INX     H
 0E0C 0D           1192         DCR     C
 0E0D C2F20D       1193         JNZ     AMPL7
 0E10 E1           1194 AMPL8:  POP     H
 0E11 C1           1195         POP     B
 0E12 C9           1196         RET
                   1197 ;
                   1198 ;
                   1199 ;        SUBROUTINE..PF
                   1200 ;
                   1201 ;
                   1202 PF:
 0E13 F5           1203         PUSH    PSW
 0E14 C5           1204         PUSH    B
 0E15 D5           1205         PUSH    D
 0E16 E5           1206         PUSH    H
 0E17 0E00         1207         MVI     C,0
 0E19 CD980D       1208 FIRST:  CALL    SAMP1
 0E1C FE80         1209         CPI     80H
 0E1E FA190E       1210         JM      FIRST
 0E21 47           1211         MOV     B,A     ;POSITIVE AND EQUAL TO 80H
 0E22 D680         1212         SUI     80H
 0E24 FE03         1213         CPI     3
 0E26 F2190E       1214         JP      FIRST   ;IF THE SAMPLE IS GREATER THAN 80H
                   1215 ;                        AND SMALLER THAN 83H
 0E29 CD980D       1216 P37:    CALL    SAMP1
 0E2C B8           1217         CMP     B
 0E2D FA190E       1218         JM      FIRST
 0E30 CA190E       1219         JZ      FIRST
 LOC  OBJ         LINE        SOURCE STATEMENT

 0E33 0C          1220          INR     C         ;COUNT STARTS HERE
 0E34 3A503C      1221          LDA     CHAN+1
 0E37 324F3C      1222          STA     CHAN
 0E3A CD980D      1223 P31:     CALL    SAMP1
 0E3D FE80        1224          CPI     80H
 0E3F CA8F0E      1225          JZ      P32
 0E42 FA480E      1226          JM      LAG
 0E45 F26C0E      1227          JP      LEAD
                  1228 LAG:
 0E48 0C          1229          INR     C
 0E49 CD980D      1230          CALL    SAMP1
 0E4C FE80        1231          CPI     80H
 0E4E FAAA0E      1232          JM      LAG1
 0E51 47          1233          MOV     B,A
 0E52 79          1234          MOV     A,C
 0E53 C650        1235          ADI     50H
 0E55 32483C      1236          STA     PHASE
 0E58 0E00        1237          MVI     C,0
                  1238 LAG5:
 0E5A 0C          1239          INR     C
 0E5B CD980D      1240          CALL    SAMP1
 0E5E B8          1241          CMP     B
 0E5F FA650E      1242          JM      P33
 0E62 C3B60E      1243          JMP     LAG6
                  1244 P33:
 0E65 79          1245          MOV     A,C
 0E66 32473C      1246          STA     HAFP
 0E69 C3D40E      1247          JMP     TERM
 0E6C 0C          1248 LEAD:    INR     C
 0E6D D680        1249          SUI     80H
 0E6F FE08        1250          CPI     8
 0E71 FA8F0E      1251          JM      P32
 0E74 0C          1252 LEAD1:   INR     C
 0E75 CD980D      1253          CALL    SAMP1
 0E78 FE80        1254          CPI     80H
 0E7A F2BF0E      1255          JP      LEAD2
 0E7D 47          1256          MOV     B,A
 0E7E 79          1257          MOV     A,C
 0E7F 32483C      1258          STA     PHASE
 0E82 0E00        1259          MVI     C,0
                  1260 LEAD5:
 0E84 0C          1261          INR     C
 0E85 CD980D      1262          CALL    SAMP1
 0E88 B8          1263          CMP     B
 0E89 F2650E      1264          JP      P33
 0E8C C3CB0E      1265          JMP     LEAD6
                  1266 P32:
```

```
  LOC  OBJ         LINE        SOURCE STATEMENT

  0E8F 47          1267         MOV     B,A
  0E90 CD980B      1268         CALL    SAMP1
  0E93 B8          1269         CMP     B
  0E94 FAA00E      1270         JM      P36
  0E97 3E00        1271         MVI     A,0
  0E99 32483C      1272         STA     PHASE
  0E9C 4F          1273         MOV     C,A
  0E9D C35A0E      1274         JMP     LAG5
                   1275 P36:
  0EA0 3E3E        1276         MVI     A,3EH
  0EA2 32483C      1277         STA     PHASE
  0EA5 0E00        1278         MVI     C,0
  0EA7 C3840E      1279         JMP     LEAD5
                   1280 LAG1:
  0EAA 00          1281         NOP
  0EAB C3AE0E      1282         JMP     LAG2
                   1283 LAG2:
  0EAE 00          1284         NOP
  0EAF C3B20E      1285         JMP     LAG3
                   1286 LAG3:
  0EB2 00          1287         NOP
  0EB3 C3480E      1288         JMP     LA8
                   1289 LAG6:
  0EB6 C3B90E      1290         JMP     LAG7
                   1291 LAG7:
  0EB9 C3BC0E      1292         JMP     LAG8
                   1293 LAG8:
  0EBC C35A0E      1294         JMP     LAG5
                   1295 LEAD2:
  0EBF 00          1296         NOP
  0EC0 C3C30E      1297         JMP     LEAD3
                   1298 LEAD3:
  0EC3 00          1299         NOP
  0EC4 C3C70E      1300         JMP     LEAD4
                   1301 LEAD4:
  0EC7 00          1302         NOP
  0EC8 C3740E      1303         JMP     LEAD1
                   1304 LEAD6:
  0ECB C3CE0E      1305         JMP     LEAD7
                   1306 LEAD7:
  0ECE C3D10E      1307         JMP     LEAD8
                   1308 LEAD8:
  0ED1 C3840E      1309         JMP     LEAD5
  0ED4 E1          1310 TERM:   POP     H
  0ED5 D1          1311         POP     D
  0ED6 C1          1312         POP     B
  0ED7 F1          1313         POP     PSW
```

```
LOC  OBJ         LINE        SOURCE STATEMENT

0ED8 C9          1314         RET
                 1315 ;
                 1316 ;
                 1317 ;       SCHEDULE TABLE
                 1318 ;
                 1319 SCHED:
0ED9 41          1320         DB      'A'
0EDA 42          1321         DB      'B'
0EDB 44          1322         DB      'D'
0EDC 45          1323         DB      'E'
0EDD 46          1324         DB      'F'
0EDE 47          1325         DB      'G'
0EDF 48          1326         DB      'H'
0EE0 49          1327         DB      'I'
0EE1 4A          1328         DB      'J'
0EE2 4B          1329         DB      'K'
0EE3 4C          1330         DB      'L'
0EE4 4D          1331         DB      'M'
0EE5 4E          1332         DB      'N'
0EE6 4F          1333         DB      'O'
0EE7 50          1334         DB      'P'
0EE8 51          1335         DB      'Q'
0EE9 52          1336         DB      'R'
0EEA 53          1337         DB      'S'
0EEB 54          1338         DB      'T'
0EEC 55          1339         DB      'U'
0EED 56          1340         DB      'V'
0EEE 57          1341         DB      'W'
0EEF 58          1342         DB      'X'
0EF0 59          1343         DB      'Y'
0EF1 5A          1344         DB      'Z'
0EF2 43          1345         DB      'C'
                 1346 ;
                 1347 ;       SCHEDULE ADDRESS TABLE
                 1348 ;
                 1349 ADRES:
0EF3 0000        1350         DW      0
0EF5 3F0C        1351         DW      CCMD
0EF7 3B0C        1352         DW      ZCMD
0EF9 370C        1353         DW      YCMD
0EFB FF0B        1354         DW      XCMD
0EFD B70B        1355         DW      WCMD
0EFF 930B        1356         DW      VCMD
0F01 630B        1357         DW      UCMD
0F03 3A0B        1358         DW      TCMD
0F05 030B        1359         DW      SCMD
0F07 C00A        1360         DW      RCMD
```

```
  LOC  OBJ          LINE        SOURCE STATEMENT

  0F09 7D0A         1361         DW      QCMD
  0F0B 3A0A         1362         DW      PCMD
  0F0D 1F0A         1363         DW      OCMD
  0F0F 040A         1364         DW      NCMD
  0F11 E909         1365         DW      MCMD
  0F13 C509         1366         DW      LCMD
  0F15 9C09         1367         DW      KCMD
  0F17 7309         1368         DW      JCMD
  0F19 5409         1369         DW      ICMD
  0F1B 3509         1370         DW      HCMD
  0F1D 1609         1371         DW      GCMD
  0F1F F708         1372         DW      FCMD
  0F21 DC08         1373         DW      ECMD
  0F23 C108         1374         DW      DCMD
  0F25 A608         1375         DW      BCMD
  0F27 8B08         1376         DW      ACMD
                    1377 ;
                    1378 ;       SUBROUTINE..PREP
                    1379 ;
                    1380 PREP:
  0F29 C5           1381         PUSH    B
  0F2A D5           1382         PUSH    D
  0F2B E5           1383         PUSH    H
  0F2C 21513C       1384         LXI     H,TEMP
  0F2F 11813C       1385         LXI     D,QUASI
                    1386 CAPS:
  0F32 7E           1387         MOV     A,M
  0F33 FE30         1388         CPI     30H      ;IS IT SAMPLING TERMINATOR?
  0F35 CA7D0F       1389         JZ      LOCK
  0F38 FE31         1390         CPI     31H      ;IS IT PHASE MEASURE REQUEST?
  0F3A CA5E0F       1391         JZ      SPECL
  0F3D FE32         1392         CPI     32H      ;IS IT FREQ. MEASURE REQUEST?
  0F3F CA500F       1393         JZ      PERD
  0F42 324F3C       1394         STA     CHAN     ;AMPLITUDE
  0F45 CDB60D       1395         CALL    AMPLT
  0F48 EB           1396 INTER:  XCHG             ;AMPLITUDE MEASUREMENT
  0F49 77           1397         MOV     M,A
  0F4A 23           1398         INX     H
  0F4B ED           1399         XCHG
  0F4C 23           1400         INX     H
  0F4D C3320F       1401         JMP     CAPS
  0F50 23           1402 PERD:   INX     H        ;FREQUENCY MEASUREMENT
  0F51 7E           1403         MOV     A,M
  0F52 324F3C       1404         STA     CHAN
  0F55 CD530D       1405         CALL    FREQ
  0F58 3A493C       1406         LDA     RESLT
  0F5B C3480F       1407         JMP     INTER
  LOC  OBJ         LINE        SOURCE STATEMENT

  0F5E 23          1408 SPECL:  INX     H        ;PHASE MEASUREMENT
  0F5F 7E          1409         MOV     A,M
  0F60 324F3C      1410         STA     CHAN
  0F63 23          1411         INX     H
  0F64 7E          1412         MOV     A,M
  0F65 32503C      1413         STA     CHAN+1
  0F68 CD130E      1414         CALL    PF
  0F6B 3A473C      1415         LDA     HAFP
  0F6E EB          1416         XCHG
  0F6F 77          1417         MOV     M,A
  0F70 23          1418         INX     H
  0F71 EB          1419         XCHG
  0F72 23          1420         INX     H
  0F73 3A483C      1421         LDA     PHASE
  0F76 EB          1422         XCHG
  0F77 77          1423         MOV     M,A
  0F78 23          1424         INX     H
  0F79 EB          1425         XCHG
  0F7A C3320F      1426         JMP     CAPS
  0F7D E1          1427 LOCK:   POP     H
  0F7E D1          1428         POP     D
  0F7F C1          1429         POP     B
  0F80 C9          1430         RET
                   1431 ;
                   1432 ;        SUBROUTINE..FILL
                   1433 ;TRNSFERS DATA FROM TEMPORARY LOCS. TO
                   1434 ;ALLOCATED MEMORY LOCATIONS.
                   1435 ;
                   1436 FILL:
  0F81 7E          1437         MOV     A,M
  0F82 EB          1438         XCHG
  0F83 77          1439         MOV     M,A
  0F84 23          1440         INX     H
  0F85 EB          1441         XCHG
  0F86 23          1442         INX     H
  0F87 05          1443         DCR     B
  0F88 C2810F      1444         JNZ     FILL
  0F8B C9          1445         RET
                   1446 ;
                   1447 ;
                   1448 MESG1:
  0F8C 494E5641    1449         DB      'INVALID COMMAND'
  0F90 4C494420
  0F94 434F4D4D
  0F98 414E44
  0F9B 0D          1450         DB      0DH
  0F9C 00          1451         DB      0
  LOC  OBJ         LINE        SOURCE STATEMENT

                   1452 ;
                   1453 EROR1:
  0F9D 00          1454         NOP
  0F9E C3600C      1455         JMP     LOC5
                   1456 ;
                   1457 ;
                   1458 ;
                   1459 ERROR:
  0FA1 218C0F      1460         LXI     H,MESG1
  0FA4 CDF50C      1461         CALL    PRNT
  0FA7 CDB10C      1462 DONE:   CALL    DELAY
  0FAA CDFD03      1463         CALL    CI
  0FAD C32608      1464         JMP     START
                   1465 ;
                   1466 ;THIS SECTION OF THE PROGRAM RESIDES
                   1467 ;IN THE RANDOM ACCESS MEMORY.
                   1468 ;
  3C40             1469         ORG     3C40H
  3C40             1470 TEST:   DS      1
  3C41             1471 PF1:    DS      2
  3C43             1472 PF2:    DS      2
  3C45             1473 PF3:    DS      2
  3C47             1474 HAFP:   DS      1
  3C48             1475 PHASE:  DS      1
  3C49             1476 RESLT:  DS      2
  3C4B             1477 COUNT:  DS      1
  3C4C             1478 FLAG1:  DS      1
  3C4D             1479 FLAG2:  DS      1
  3C4E             1480 FLAG3:  DS      1
  3C4F             1481 CHAN:   DS      2
  3C51             1482 TEMP:   DS      22
  3C67             1483 DATA:   DS      26
  3C81             1484 QUASI:  DS      11
  3C8C             1485 NUMB:   DS      1
  3C8D             1486 CMND:   DS      14
  3C9B             1487 FLAG:   DS      1
  3C9C             1488 TIME:   DS      1
  3C9D             1489 SLEP:   DS      2
  3C9F             1490 MARCH:  DS      2
  3CA1             1491 AUG:    DS      1
  0800             1492         END     800H

PUBLIC SYMBOLS

EXTERNAL SYMBOLS
```

```
USER SYMBOLS
ACMD   A 088B   ADC8   A 8FF8   ADCHI  A 8FFE   ADLCO  A 8FFD   ADRES  A 0EF3
AMPL1  A 0DBD   AMPL2  A 0DCD   AMPL3  A 0DD8   AMPL4  A 0DE3   AMPL5  A 0E02
AMPL6  A 0E0B   AMPL7  A 0DF2   AMPL8  A 0E10   AMPLT  A 0DB6   APR    A 0CB7
AVG    A 3CA1   BCMD   A 08A6   BEG    A 0815   CAPS   A 0F32   CCMD   A 0C3F
CCMD1  A 0C43   CHAN   A 3C4F   CI     A 03FD   CMND   A 3C8D   CNVCMD A 8FFB
CO     A 03FA   CONT   A 0880   CONV   A 0CC5   COUNT  A 3C4B   CRDSEL A 8FFF
DATA   A 3C67   DCMD   A 08C1   DELAY  A 0CB1   DOL    A 0024   DONE   A 0FA7
DRIVE  A 8FF3   ECMD   A 08DC   EROR1  A 0F9D   ERROR  A 0FA1   ESC    A 001B
FAST   + 0003   FCMD   A 08F7   FILL   A 0F81   FIRST  A 0E19   FLAG   A 3C9B
FLAG1  A 3C4C   FLAG2  A 3C4D   FLAG3  A 3C4E   FREQ   A 0D53   FREQ1  A 0D59
FREQ2  A 0D76   GCMD   A 0916   GETCH  A 002C   GNSEL  A 8FF9   HAFP   A 3C47
HCMD   A 0935   ICMD   A 0954   IDLE   A 0D06   INTER  A 0F48   JCMD   A 0973
KAT    A 0DA9   KCMD   A 099C   L1     A 082B   L10    A 0CE9   L11    A 0CD4
L12    A 0CEE   L13    A 0CE2   L14    A 0CF3   L15    A 0CF6   L16    A 0D04
L2     A 0840   L20    A 0D11   L21    A 0D0F   L22    A 0D0D   L23    A 0D2F
L24    A 0D09   L5     A 0C69   L6     A 0C93   LAG    A 0E48   LAG1   A 0EAA
LAG2   A 0EAE   LAG3   A 0EB2   LAG5   A 0E5A   LAG6   A 0EB6   LAG7   A 0EB9
LAG8   A 0EBC   LCMD   A 09C5   LEAD   A 0E6C   LEAD1  A 0E74   LEAD2  A 0EBF
LEAD3  A 0EC3   LEAD4  A 0EC7   LEAD5  A 0E84   LEAD6  A 0ECB   LEAD7  A 0ECE
LEAD8  A 0ED1   LOC1   A 0867   LOC2   A 0873   LOC3   A 0C47   LOC4   A 0C48
LOC5   A 0C60   LOC6   A 0C5B   LOCK   A 0F7D   LOOP   A 0D40   MARCH  A 3C9F
MAY    A 0CBA   MCMD   A 09E9   MESG1  A 0F8C   MUXADR A 8FFA   NCMD   A 0A04
NEXT   A 0D3C   NUMB   A 3C8C   OCMD   A 0A1F   ONE    + 0000   P31    A 0E3A
P32    A 0E8F   P33    A 0E65   P36    A 0EA0   P37    A 0E29   P4     A 0D8F
PCMD   A 0A3A   PERD   A 0F50   PF     A 0E13   PF1    A 3C41   PF2    A 3C43
PF3    A 3C45   PHASE  A 3C48   PP1    A 0D81   PP3    A 0D86   PP4    A 0D8B
PREP   A 0F29   PRNT   A 0CF5   QCMD   A 0A7D   QUASI  A 3C81   RCMD   A 0AC0
REGDS  A 02DF   RESLT  A 3C49   SAMP1  A 0D98   SAMP3  A 0D33   SCHED  A 0ED9
SCMD   A 0B03   SETUP  A 8FF0   SLEP   A 3C9D   SPECL  A 0F5E   START  A 0826
STATUS A 8FFC   STP    + 0002   TCMD   A 0B3A   TEMP   A 3C51   TERM   A 0ED4
TEST   A 3C40   TIME   A 3C9C   TWO    + 0001   UCMD   A 0B63   VCMD   A 0B93
WCMD   A 0BB7   XCMD   A 0BFF   XMIT   A 0C64   YCMD   A 0C37   ZCMD   A 0C3B

ASSEMBLY COMPLETE,   NO ERRORS
COPIED :F0:JAN7.LST TO :CO:
-
```

Appendix II

Control Routine (TSK 1) Listing.

```
C
C       THE MAIN ROUTINE (TSK 1) FOR THE SIMULATOR.
C
C       THE COMMON BLOCK DTA/A(164)/ IS THE MEMORY REGION
C       SHARED BY MAIN CONTROL TASK (TSK 1) AND
C       COMMUNICATION TASK (TSK 2) OF THE SOFTWARE STRUCTURE
C
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 SWITCH OPEN POSITION TO BEGIN THE PROGRAM.
C
        LOGICAL*1 SW
        COMMON /DTA/ A(164),I
        COMMON/DFILE/IBUF(1100)/SUB/SW(100)
        DIMENSION TIM(2),DAT(3)
        DATA SW/100*.FALSE./
        DATA COMMN /6RCOMMN /
C
C
C       INTRODUCTORY DISPLAY WITH OPERATING INFORMATION.
C
        CALL REQUES(COMMN)
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)
        CALL CLREF(11)                      ! CLR LFT PEN EVENT FLAG
        NZ=0
130     CALL LPEN(MZ,NZ)                    ! CK IF HIT IS SURE
        IF(NZ.NE.91 .AND. NZ.NE.92)GOTO 130
        CALL OFF(90)
        IF(NZ.EQ.92)GOTO 140
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,8,'(2A4)')
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 ON A SWITCH OR CIRCUIT BREAKER GO TO 250
C
          IF(N.GT.200)GOTO 260
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         THE SUBSTATION 4 PICTURE AND DATA.
C
300       CALL INIT(1100)
          CALL RSTR('SUBST4.DPY')
335       CALL CHECK(21,26,27,27,950,950,950,950,950,950,950,950,N)
          NB=N-300
          WRITE(6,336)NB
336       FORMAT(1X,I3)
          GOTO(200,600,700,331,100),NB
331       A(101)=21.0
          A(102)=0.0
```

```
      WRITE(6,332)A(101),A(102)
332   FORMAT(1X,2F8.2)
      CALL MOD(950,950,950,950,950,950,950,950)
      GOTO 335
C
C
C     THE SUBSTATION 6 PICTURE AND DATA.
C     AT PRESENT THERE IS NO CHANNEL ASSIGNED TO THIS VARIABLE.
C
400   CALL INIT(1100)
      CALL RSTR('SUBST6.DPY')
435   CALL CHECK(32,38,71,76,958,958,958,958,958,958,958,958,N)
      NB=N-400
      WRITE(6,436)NB
436   FORMAT(1X,I3)
      GOTO(200,500,600,431,100),NB
431   A(101)=23.0
      A(102)=0.0
      WRITE(6,432)A(101),A(102)
432   FORMAT(1X,2F8.2)
      CALL MOD(958,958,958,958,958,958,958,958)
      GOTO 435
C
C
C     THE LOAD CENTER PICTURE AND DATA PLACEMENT.
C
500   CALL INIT(1100)
      CALL RSTR('LDCTR.DPY')
535   CALL CHECK(37,44,45,45,958,958,958,958,958,958,958,958,N)
      NB=N-549
      WRITE(6,536)
536   FORMAT(1X,I3)
      GOTO (200,400,531,100),NB
531   A(101)=21.0
      A(102)=0.0
      WRITE(6,532)A(101),A(102)
532   FORMAT(1X,2F8.2)
      CALL MOD(958,958,958,958,958,958,958,958)
      GOTO 535
C
C
C     THE NETWORK SYSTEM PICTURE AND DATA PLACEMENT.
C
600   CALL INIT(1100)
      CALL RSTR('NETSYS.DPY')
636   CALL CHECK(26,32,33,33,950,950,958,959,962,962,962,962,N)
      NB=N-600
      WRITE(6,637)NB
637   FORMAT(1X,I3)
      GOTO(200,400,300,200,631,632,633,100),NB
631   A(101)=21.0
      GOTO 634
632   A(101)=22.0
      GOTO 634
633   A(101)=23.0
634   A(102)=0.0
      WRITE(6,635)A(101),A(102)
635   FORMAT(1X,2F8.2)
      CALL MOD(950,950,958,959,960,960,960,960)
      GOTO 636
```

```
C
C
C       THE INDUSTRIAL LOAD PICTURE AND DATA PLACEMENT.
C
700     CALL INIT(1100)
        CALL RSTR('INDLD.DPY')
735     CALL CHECK(53,58,23,23,950,953,953,953,953,953,953,953,N)
        NB=N-700
        WRITE(6,736)NB
736     FORMAT(1X,I3)
        GOTO(200,300,700,700,731,732,900,1000,800,100),NB
731     A(101)=21.0
        GOTO 733
732     A(101)=22.0
733     A(102)=0.0
        WRITE(6,734)A(101),A(102)
734     FORMAT(1X,2F8.2)
        CALL MOD(950,953,953,953,953,953,953,953)
        GOTO 735
C
C
C       THE STATION 11 PICTURE AND DATA PLACEMENT.
C
800     CALL INIT(1100)
        CALL RSTR('STAT11.DPY')
837     CALL CHECK(13,17,65,70,946,948,965,965,965,965,965,965,N)
        NB=N-800
        WRITE(6,877)NB
877     FORMAT(1X,I3)
        GOTO(200,200,800,800,831,832,833,834,900,1000,100),NB
831     A(101)=5.0
        GOTO 835
832     A(101)=22.0
        GOTO 835
833     A(101)=9.0
        GOTO 835
834     A(101)=21.0
835     A(102)=0.0
        WRITE(6,839)A(101),A(102)
839     FORMAT(30X,2F8.2)
        CALL MOD(946,948,965,965,965,965,965,965)
        GOTO 837
C
C
C
C       THE STATION 1 PICTURE AND DATA PLACEMENT.
C
900     CALL INIT(1100)
        CALL RSTR('STAT1.DPY')
995     CALL CHECK(7,12,46,50,936,945,966,966,966,966,966,966,N)
        IF(A(1).EQ.1.0)GOTO 99
        NB=N-900
        WRITE(6,996)NB
996     FORMAT(1X,I3)
        GOTO(200,900,900,900,900,900,900,900,900,1000,991,992,100),NB
991     A(101)=17.0
        GOTO 993
992     A(101)=18.0
993     A(102)=0.0
        WRITE(6,994)A(101),A(102)
```

```
994     FORMAT(1X,2F8.2)
        CALL MOD(936,945,966,966,966,966,966,966)
        GOTO 995
99      CALL DRSOUT
        A(101)=0.0
        GOTO 995
C
C
C       THE GENERATOR 1 PICTURE AND DATA PLACEMENT.
C
1000    CALL INIT(1100)
        CALL RSTR('GEN1.DPY')
1038    CALL CHECK(6,6,51,52,931,937,940,942,945,947,967,967,N)
        NB=N-1000
        GOTO(200,1031,1032,1033,1034,1035,1100,100,
     1 1000,1000,1000,1000),NB
1031    A(101)=16.0
        GOTO 1036
1032    A(101)=17.0
        GOTO 1036
1033    A(101)=18.0
        GOTO 1036
1034    A(101)=5.0
        GOTO 1036
1035    A(101)=9.0
1036    A(102)=0.0
        WRITE(6,1037)A(101),A(102)
1037    FORMAT(1X,2F8.2)
        CALL MOD(931,937,940,942,945,947,967,967)
        GOTO 1038
C
C
C       THE SYNCRONIZING PICTURE AND DATA PLACEMENT.
C
1100    CALL INIT(1100)
        CALL RSTR('SYNCRN.DPY')
1135    CALL CHECK(4,14,15,15,931,932,936,937,941,942,946,947,N)
        NB=N-1100
        WRITE(6,1136)NB
1136    FORMAT(1X,I3)
        IF(NB.GT.49)GOTO 1130
        GOTO (1100,1100,1100,1100,1100,1100,1100,1100,1100,1100,1100,
     1 1100,1100,1100),NB
1130    NB=NB-49
        GOTO(1131,1132,1000,900,100),NB
1131    A(101)=21.0
        GOTO 1133
1132    A(101)=22.0
1133    A(102)=0.0
        WRITE(6,1134)A(101),A(102)
1134    FORMAT(1X,2F8.2)
        CALL MOD(931,932,936,937,941,942,946,947)
        GOTO 1135
C
C
9999    STOP
        END
```

```
C
      SUBROUTINE UPDATE(J1,J2)
      LOGICAL*1 SW
      COMMON/SUB/SW(100)
      COMMON /DTA/ A(164),I
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 CHECK(L1,L2,L3,L4,L5,L6,L7,L8,L9,L10,L11,L12,N)
      LOGICAL*1 SW
      COMMON/SUB/SW(100)
      COMMON /DTA/ A(164),I
      DATA COMMN /6RCOMMN /
100   CALL UPDATE(L1,L2)
      CALL UPDATE(L3,L4)
120   CALL CLREF(11)
130   DO 131 J=L5,L6
      I=J-800
      VOLTS=A(I)
131   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 331 J=L7,L8
      I=J-800
      VOLTS=A(I)
331   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 332 J=L9,L10
      I=J-800
      VOLTS=A(I)
332   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 333 J=L11,L12
      I=J-800
      VOLTS=A(I)
333   CALL NMBR(J,VOLTS,5,'(F5.1)')
      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)
      WRITE(6,141)NZ
141   FORMAT(1X,I3)
      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 152
151   SW(N)=.FALSE.
152   N=1
      A(101)=1.0
150   RETURN
      END
C
C     SUBROUTINE TO UPDATE THE DISPLAY FOR
C     3 TIMES IRRESPECTIVE OF ANYTHING
C
      SUBROUTINE MOD(M1,M2,M3,M4,M5,M6,M7,M8)
      LOGICAL*1 SW
      COMMON/SUB/SW(100)
      COMMON /DTA/ A(164),I
      DATA COMMN /6RCOMMN /
      NUM=0
      ITEMP=0
161   IF(A(102).EQ.1.0)GOTO 162
      WRITE(6,166)
166   FORMAT(1X,'TEST')
      GOTO 161
162   DO 164 J=M1,M2
      I=J-800
      VOLTS=A(I)
      CALL IDLE
164   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 334 J=M3,M4
      I=J-800
      VOLTS=A(I)
334   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 335 J=M5,M6
      I=J-800
      VOLTS=A(I)
335   CALL NMBR(J,VOLTS,5,'(F5.1)')
      DO 336 J=M7,M8
      I=J-800
      VOLTS=A(I)
336   CALL NMBR(J,VOLTS,5,'(F5.1)')
      A(102)=0.0
      NUM=NUM+1
      IF(NUM.EQ.3)GOTO 163
      GOTO 161
163   RETURN
      END
```

```
C
        SUBROUTINE IDLE
        COMMON /DTA/ A(164),I
        ICLK=1
10      ICLK=ICLK+1
        IF(ICLK.EQ.100)GOTO 20
        GOTO 10
20      RETURN
        END
```

Appendix III

Command and Communication Routine (TSK 2)

Listing.

```
C
C       THIS COMMUNICATION ROUTINE (TSK 2) PERFORMS
C       THE TRANSMISSION OF EITHER COMMANDS TO MICRO.
C       FOR DATA ACQUISITION OR TO SIMULATOR CONTROLLER
C       (TSK 3) ROUTINE FOR UPDATE OF FUNCTIONAL SWITCHES
C       AND RELAY CONTACTORS. IN ADDITION THIS PROGRAM
C       RECEIVES THE DATA FROM MICRO AND NORMALIZES TO REAL
C       VARIABLE UNITS AND VALUES.
C
C
        INTEGER HD,P
        INTEGER SCAN
        INTEGER S(26),Z(16)
        INTEGER Y1(26),Y2(26)
        INTEGER X2(58)
        COMMON /DTA/A(174),I
        DATA S/'A','B','C','D','E','F','G','H','I','J','K','L','M',
     1 'N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/
        DATA Z/'A','B','C','D','E','F','0','1','2','3','4','5',
     1 '6','7','8','9'/
        DATA X,Y,T/'0','U','$'/
C
C       THE DATA FROM MICRO IS RECEIVED THREE TIMES FOR
C       EVERY SCHEDULE REQUESTED.
C       INITIALIZES THE SCAN NUMBER
C
        SCAN=0
C
C       INITIALIZE THE THE ARRAY A(174) WHICH CORRESPONDS
C       TO THE VARIABLES OF POWER SYSTEM SIMULATOR.
C
        DO 10 I=1,174
        A(I)=0.0
10      CONTINUE
C
        CALL DELAY
25      IF(A(101).NE.0.0)GOTO 30
        CALL DELAY
        GOTO 25
C
C       DECODE THE USER INPUT THRU THE LIGHT PEN
C       AND SEND PROPER FORMATTED COMMAND TO MICRO.
C
30      K=A(101)/1
        P=S(K)
        A(101)=0.0
        WRITE(4,40)T,X,Y,P,P,T
40      FORMAT(1X,6A1,/)
C
C       RECEIVES THE HEXADECIMAL DATA FROM MICRO AND
C       PROCESSES TO REAL MAGNITUDES AND UNITS.
C
201     READ(4,45)X2(1)
45      FORMAT(A1)
        IF(X2(1).EQ.S(16))GOTO 231
        GOTO 201
231     READ(4,46)(X2(I),I=1,58)
46      FORMAT(58A1)
        WRITE(6,47)
```

```
47      FORMAT(1X,' ')
        IF(X2(1).NE.S(9))GOTO 50
        A(102)=0.0
        SCAN=0
        GOTO 30
C
C       CONVERSION OF HEX NUMBERS INTO DECIMAL NUMBERS
C
50      HD=16
        DO 52 I=3,53,2
        J=((I+1)/2)-1
        IF(X2(I).EQ.Z(1))GOTO 60
        IF(X2(I).EQ.Z(2))GOTO 61
        IF(X2(I).EQ.Z(3))GOTO 62
        IF(X2(I).EQ.Z(4))GOTO 63
        IF(X2(I).EQ.Z(5))GOTO 64
        IF(X2(I).EQ.Z(6))GOTO 65
        IF(X2(I).EQ.Z(7))GOTO 160
        IF(X2(I).EQ.Z(8))GOTO 161
        IF(X2(I).EQ.Z(9))GOTO 162
        IF(X2(I).EQ.Z(10))GOTO 163
        IF(X2(I).EQ.Z(11))GOTO 164
        IF(X2(I).EQ.Z(12))GOTO 165
        IF(X2(I).EQ.Z(13))GOTO 166
        IF(X2(I).EQ.Z(14))GOTO 167
        IF(X2(I).EQ.Z(15))GOTO 168
        IF(X2(I).EQ.Z(16))GOTO 169
60      Y2(I)=HD*10
        GOTO 70
61      Y2(I)=HD*11
        GOTO 70
62      Y2(I)=HD*12
        GOTO 70
63      Y2(I)=HD*13
        GOTO 70
64      Y2(I)=HD*14
        GOTO 70
65      Y2(I)=HD*15
        GOTO 70
160     Y2(I)=0
        GOTO 70
161     Y2(I)=HD*1
        GOTO 70
162     Y2(I)=HD*2
        GOTO 70
163     Y2(I)=HD*3
        GOTO 70
164     Y2(I)=HD*4
        GOTO 70
165     Y2(I)=HD*5
        GOTO 70
166     Y2(I)=HD*6
        GOTO 70
167     Y2(I)=HD*7
        GOTO 70
168     Y2(I)=HD*8
        GOTO 70
169     Y2(I)=HD*9
70      IF(X2(I+1).EQ.Z(1))GOTO 80
```

```
        IF(X2(I+1).EQ.Z(3))GOTO 82
        IF(X2(I+1).EQ.Z(4))GOTO 83
        IF(X2(I+1).EQ.Z(5))GOTO 84
        IF(X2(I+1).EQ.Z(6))GOTO 85
        IF(X2(I+1).EQ.Z(7))GOTO 180
        IF(X2(I+1).EQ.Z(8))GOTO 181
        IF(X2(I+1).EQ.Z(9))GOTO 182
        IF(X2(I+1).EQ.Z(10))GOTO 183
        IF(X2(I+1).EQ.Z(11))GOTO 184
        IF(X2(I+1).EQ.Z(12))GOTO 185
        IF(X2(I+1).EQ.Z(13))GOTO 186
        IF(X2(I+1).EQ.Z(14))GOTO 187
        IF(X2(I+1).EQ.Z(15))GOTO 188
        IF(X2(I+1).EQ.Z(16))GOTO 189
C
80      Y1(J)=Y2(I)+10
        GOTO 52
81      Y1(J)=Y2(I)+11
        GOTO 52
82      Y1(J)=Y2(I)+12
        GOTO 52
83      Y1(J)=Y2(I)+13
        GOTO 52
84      Y1(J)=Y2(I)+14
        GOTO 52
85      Y1(J)=Y2(I)+15
        GOTO 52
180     Y1(J)=Y2(I)
        GOTO 52
181     Y1(J)=Y2(I)+1
        GOTO 52
182     Y1(J)=Y2(I)+2
        GOTO 52
183     Y1(J)=Y2(I)+3
        GOTO 52
184     Y1(J)=Y2(I)+4
        GOTO 52
185     Y1(J)=Y2(I)+5
        GOTO 52
186     Y1(J)=Y2(I)+6
        GOTO 52
187     Y1(J)=Y2(I)+7
        GOTO 52
188     Y1(J)=Y2(I)+8
        GOTO 52
189     Y1(J)=Y2(I)+9
52      CONTINUE
C
C
C       CONVERSION OF SCALED DECIMAL DATA INTO REAL VALUES
C
        IF(Y1(1).EQ.0)GOTO 146
C
C       GENERATOR 1  VOLTAGE
C
        A(131)=(Y1(1)-122)*5.045
        GOTO 147
146     A(131)=0.0
C
```

```
C       GENERATOR 1 FREQUENCY
C
147     IF(Y1(5).EQ.0)GOTO 141
        A(132)=(62.0*60.0)/Y1(5)
        GOTO 142
141     A(132)=0.0
C
C       GENERATOR 1 CURRENT
C
142     IF(Y1(15).EQ.0)GOTO 148
        A(133)=(Y1(15)-126)*0.0167
        GOTO 149
148     A(133)=0.0
C
C       GENERATOR 1 PHASE WRT TO INTER TIE
C
149     IF(Y1(9).EQ.0)GOTO 101
        IF(Y1(10).GT.80)GOTO 500
        A(134)=((Y1(10)/Y1(9))*180.0)
        GOTO 91
500     Y1(10)=Y1(10)-80
        GOTO 90
101     A(134)=0.0
        GOTO 91
90      A(134)=-((Y1(10)/Y1(9))*180.0)
C
C       GENERATOR 1 VOLTAMPERES
C
91      A(135)=A(131)*A(133)/1732.0
C
C       GENERATOR 2 VOLTAGE
C
C
        IF(Y1(2).EQ.0)GOTO 171
        A(136)=(Y1(2)-128)*4.96
C
C       GENERATOR 2 FREQUENCY
C
        GOTO 172
171     A(136)=0.0
172     IF(Y1(6).EQ.0)GOTO 102
        A(137)=(62.0*60.0)/Y1(6)
        GOTO 103
102     A(137)=0.0
C
C       GENERATOR 2 CURRENT
C
103     IF(Y1(16).EQ.0)GOTO 173
        A(138)=(Y1(16)-128)*0.0165
        GOTO 174
173     A(138)=0.0
C
C       GENERATOR 2 PHASE WRT INTER TIE
C
174     IF(Y1(11).EQ.0)GOTO 104
        IF(Y1(12).GT.80)GOTO 501
        A(139)=((Y1(12)/Y1(11))*180.0)
        GOTO 93
501     Y1(12)=Y1(12)-80
        GOTO 92
```

```
104     A(139)=0.0
        GOTO 93
92      A(139)=-((Y1(12)/Y1(11))*180.0)
C
C       GENERATOR 2 VOLTAMPERES
C
93      A(140)=(A(136)*A(138))/1732.0
C
C       GENERATOR 3 VOLTAGE
C
        IF(Y1(3).EQ.0)GOTO 175
        A(141)=(Y1(3)-124)*5.35
        GOTO 176
175     A(141)=0.0
C
C       GENERATOR 3 FREQUENCY
C
176     IF(Y1(7).EQ.0)GOTO 105
        A(142)=(62.0*60.0)/Y1(7)
        GOTO 106
105     A(142)=0.0
C
C       GENERATOR 3 CURRENT
C
106     IF(Y1(17).EQ.0)GOTO 177
        A(143)=(Y1(17)-128)*0.01936
        GOTO 178
177     A(143)=0.0
C
C       GENERATOR 3 PHASE WRT INTER TIE
C
178     IF(Y1(13).EQ.0)GOTO 107
        IF(Y1(14).GT.80)GOTO 503
        A(144)=((Y1(14)/Y1(13))*180.0)
        GOTO 95
503     Y1(14)=Y1(14)-80
        GOTO 94
107     A(144)=0.0
        GOTO 95
94      A(144)=-((Y1(14)/Y1(13))*180.0)
C
C       GENERATOR 3 VOLTAMPERES
C
95      A(145)=(A(141)*A(143))/1732.0
C
C       INTERTIE VOLTAGE
C
        IF(Y1(4).EQ.0)GOTO 191
        A(146)=(Y1(4)-128)*4.0
        GOTO 192
191     A(146)=0.0
C
C       INTERTIE CURRENT
C       FOR 5 AMPS. RANGE ONLY.
C
192     IF(Y1(24).EQ.0)GOTO 193
        A(148)=(Y1(24)-128)*0.2238
        GOTO 194
193     A(148)=0.0
C
```

```
C       INTERTIE FREQUENCY
C
194     IF(Y1(8).EQ.0)GOTO 195
        A(147)=(62.0*60.0)/Y1(8)
        GOTO 196
195     A(147)=0.0
C
C
C       SUBSTATION-4 RIGHT BUS VOLTAGE
C
196     IF(Y1(18).EQ.0)GOTO 211
        A(150)=(Y1(18)-128)*4.47
        GOTO 212
211     A(150)=0.0
C
C       INDUSTRIAL LOAD VOLTAGE
C
212     IF(Y1(20).EQ.0)GOTO 301
        A(151)=(Y1(20)-128)*4.273
        GOTO 302
301     A(151)=0.0
C
C
C       NETWORK SYSTEM VOLTAGE L1-N
C
302     IF(Y1(23).EQ.0)GOTO 307
        A(159)=(Y1(23)-128)*1.098
        GOTO 308
307     A(159)=0.0
C
C       INDUSTRIAL LOAD CURRENT
C
308     IF(Y1(25).EQ.0)GOTO 309
        A(153)=(Y1(25)-128)*0.23178
        GOTO 311
309     A(153)=0.0
C
C       NETWORK SYSTEM LOAD CURRENT
C
311     GOTO 312
        A(162)=(Y1(26)-128)*0.10556
        GOTO 313
312     A(162)=0.0
C
C       SUBSTATION-6 BUS VOLTAGE
C
313     IF(Y1(22).EQ.0)GOTO 315
        A(158)=(Y1(22)-128)*4.765
        GOTO 316
315     A(158)=0.0
316     A(152)=1.0
C
C       STATION-1 BUS VOLTAGE
C
        IF(Y1(19).EQ.0)GOTO 445
        A(156)=(Y1(19)-122)*5.109
        GOTO 446
445     A(156)=0.0
C
C       STATION-11 BUS VOLTAGE
C
446     A(155)=A(146)*2.8846
        WRITE(6,444)
444     FORMAT(1X,'  ')
        A(102)=1.0
        SCAN=SCAN+1
        IF(SCAN.EQ.3)GOTO 225
        GOTO 201
225     SCAN=0
        GOTO 25
        STOP
        END
C
        SUBROUTINE DELAY
        COMMON /DTA/ A(164),I
        ITEST=1
10      ITEST=ITEST+1
        IF(ITEST.EQ.50)GOTO 20
        GOTO 10
20      RETURN
        END
PIP>
```

Appendix IV

DRS - Output Driver Routine (TSK 3)

Listing.

(Simulator Controller Interface Routine)

```
        .NLIST    TTM,ME
        .GLOBL     DRSOUT
        ;THIS PROGRAM (TSK 3) CONTROLS THE
        ;FUNCTIONAL SWITCHES AND CIRCUIT
        ;BREAKERS  OF THE POWER SIMULATOR
        ;THROUGH RELAY/TRIAC ARRANGEMENT
        .TITLE DRSOUT - SET DRS-11 OUTPUTS
        ; ROUTINE TO SET DRS-11 OUTPUT DEVICES. IT
        ; WILL READ THE COMMON INTEGER ARRAY SW
        ; (100 LOGICAL*1) AND SET THE 48 LINES ON
        ; EACH DRS-11 ACCORDING TO THE SETTINGS OF
        ; THE LOGICALS IN SW.  THE SUBROUTINE ALSO
        ; FEATURES A SOFTWARE PATCH BOARD (BYTE
        ; ARRAY PATCH) WHICH ALLOWS A SOFTWARE RE-
        ; WIRING BETWEEN THE ENTRY NUMBER IN SW
        ; AND ANY PIN NUMBER IN EITHER DRS-11.
        ; UPON EACH CALL, THE ENTIRE OUTPUT FROM
        ; EACH DRS-11 IS REGENERATED FROM SW.
        ;
        PSW=177776
       .PSECT DRSOUT
       .RADIX 10
DRSOUT: JSR       PC,SAVREG    ; SAVE REGS
        MOV       #DRSREG,R5
        MOV      #16,R1
        MOV      #PATCH,R2
AGAIN:  CLR      R3
        MOVB     (R2)+,R3          ;  R3 IS SW ENTRY# FOR NEXT PIN
        DEC      R3      ;  ALLOW FOR FORTRAN '1' INDEX
        ADD      #SUB,R3     ;  R3 HAS ADDR OF SW BYTE FOR PIN
        MOVB     (R3),R4;  R4IS SW VALUE FOR PIN 0
        ROR      R4      ;  SET CARRY
        ROR      R0      ; MOVE INTO R0
        DEC      R1      ;  TIME TO WRITE R0?
        BNE      TEST    ;  IF NOT, SKIP
        MOV      R0,@(R5)+       ; MOVE TO DRSREG VALUE
        MOV      #16,R1          ;  RESET COUNT
TEST:   CMP      R2,#PATCH+96  ;  TIME TO STOP?
        BNE      AGAIN
        JSR      PC,RESREG
        RTS       PC
DRSREG: DRS0R0                     ;  BUS ADDRESS OF DRS-11
        DRS0R1
        DRS0R2
        DRS1R0
        DRS1R1
        DRS1R2
PATCH:  .BYTE  1                   ; PIN 0 = SW( 1)
        .BYTE  2                   ; PIN 1 = SW( 2)
        .BYTE  3       ; PIN 2 = SW( 3)
        .BYTE  4       ; PIN 3 = SW(4)
        .BYTE  5       ; PIN 4 = SW( 5)
        .BYTE  6       ; PI5 = SW( 6)
        .BYTE  7       ; PIN 6 = SW( 7)
        .BYTE  8       ; PIN 7 = SW( 8)
        .BYTE  9       ; PIN 8 = SW( 9)
        .BYTE 10       ; PIN 9 = SW(10)
```

```
  .BYTE 11      ; PIN 10= SW(11)
  .BYTE 12      ; PIN 11= SW(12)
  .BYTE 13      ; PIN 12= SW(13)
  .BYTE 14      ; PIN 13= SW(13)
  .BYTE 15      ; PIN 14= SW(15)
  .BYTE 16      ; PIN 15= SW(16)
  .BYTE 17      ; PIN 16= SW(17)
  .BYTE 18      ; PIN 17= SW(18)
  .BYTE 19      ; PIN 18= SW(19)
  .BYTE 20      ; PIN 19= SW(20)
  .BYTE 21      ; PIN 20= SW(21)
  .BYTE 22      ; PIN 21= SW(22)
  .BYTE 23      ; PIN 22= SW(23)
  .BYTE 24      ; PIN 23=SW(24)
  .BYTE 25      ; PIN 24= SW(25)
  .BYTE 26      ; PIN 25= SW(26)
  .BYTE 27      ; PIN 26= SW(27)
  .BYTE 28      ; PIN 27= SW(28)
  .BYTE 29      ; PIN 28= SW(29)
  .BYTE 30      ; PIN 29= SW(30)
  .BYTE 31      ; PIN 30= SW(31)
  .BYTE 32      ; PIN 31= SW(32)
  .BYTE 33      ; PIN 32= SW(33)
  .BYTE 34      ; PIN 33= SW(34)
  .BYTE 35      ; PIN 34= SW(35)
  .BYTE 36      ; PIN 35= SW(36)
  .BYTE 37      ; PIN 36= SW(37)
  .BYTE 38      ; PIN 37= SW(38)
  .BYTE 39      ; PIN 38= SW(39)
  .BYTE 40      ; PIN 39= SW(40)
  .BYTE 41      ; PIN 40= SW(41)
  .BYTE 42      ; PIN 41= SW(42)
  .BYTE 43      ; PIN 42= SW(43)
  .BYTE 44      ; PIN 43= SW(44)
  .BYTE 45      ; PIN 44 = SW(45)
  .BYTE 46      ; PIN 45= SW(46)
  .BYTE 47      ; PIN 46= SW(47)
  .BYTE 48      ; PIN 47= SW(48)
  .BYTE 49      ; PIN 0 = SW(49)   *DRS#2 STARTS
  .BYTE 50      ; PIN 1 = SW(50)
  .BYTE 51      ; PIN 2 = SW(51)
  .BYTE 52      ; PIN 3 = SW(52)
  .BYTE 53      ; PIN 4 = SW(53)
  .BYTE 54      ; PIN 5 = SW(54)
  .BYTE 55      ; PIN 6 = SW(55)
  .BYTE 56      ; PIN 7 = SW(56)
  .BYTE 57     ; PIN 8 = SW(57)
  .BYTE 58      ; PIN 9 = SW(58)
  .BYTE 59      ; PIN 10= SW(59)
 .BYTE 60      ; PIN 11= SW(60)
  .BYTE 61      ; PIN 12= SW(61)
  .BYTE 62      ; PIN 13= SW(62)
  .BYTE 63      ; PIN 14= SW(63)
  .BYTE 64      ; PIN 15= SW(64)
  .BYTE 65      ; PIN 16= SW(~%)
  .BYTE 66      ; PIN 16= SW(66)
  .BYTE 67      ; PIN 18= SW(67)
  .BYTE 68      ; PIN 19= SW(68)
  .BYTE 69      ; PIN 20= SW(69)
  .BYTE 70      ; PIN 21= SW(70)
```

```
         .BYTE 71       ; PIN 22= SW(71)
         .BYTE 72       ; PIN 23= SW(72)
         .BYTE 73       ; PIN 24= SW(73)
         .BYTE 74       ; PIN 25= SW(74)
         .BYTE 75       ; PIN 26= SW(75)
         .BYTE 76       ; PIN 27= SW(75)
         .BYTE 77       ; PIN 28= SW(77)
         .BYTE 78       ; PIN 29= SW(78)
         .BYTE 79       ; PIN 30= SW(79)
         .BYTE 80       ; PIN 31= SW(80)
         .BYTE 81       ; PIN 32= SW(81)
         .BYTE 82       ; PIN 33=SW(82)
        .BYTE 83       ; PIN 34= SW(83)
         .BYTE 84       ; PIN35= SW(85)
         .BYTE 85       ; PIN 36= SW(86)
         .BYTE 86       ; PIN 37= SW(86)
         .BYTE 87       ; PIN 38= SW(87)
         .BYTE 88       ; PIN 39= SW(88)
         .BYTE 89       ; PIN 40= SW(89)
         .BYTE 90       ; PIN 41= SW(90)
         .BYTE 91       ; PIN 42= SW(91)
         .BYTE 92       ; PIN 43= SW(92)
         .BYTE 93       ; PIN 44= SW(93)
         .BYTE 94       ; PIN 45= SW(94)
         .BYTE 95       ; PIN 46= SW(95)
         .BYTE 96       ; PIN 47= SW(96)
SAVREG:  MOV        (SP)+,SAVPCR       ; SAVE REGS
         MOV        #PSW,-(SP)
         MOV        R5,-(SP)
         MOV        R4,-(SP)
         MOV        R3,-(SP)
         MOV        R2,-(SP)
         MOV        R1,-(SP)
         MOV        R0,-(SP)
         MOV        SAVPCR,-(SP)
         RTS        PC
RESREG:  MOV        (SP)+,RSTPCR       ; RESTORE REGS
         MOV        (SP)+,R0
         MOV        (SP)+,R1
         MOV        (SP)+,R2
         MOV        (SP)+,R3
         MOV        (SP)+,R4
         MOV        (SP)+,R5
         MOV        (SP)+,#PSW
         MOV        RSTPCR,-(SP)
         RTS        PC
SAVPCR:  0
RSTPCR:  0
          .PSECT SUB,OVR
SUB:  .BLKB 100
          .END
```

Appendix V

Indirect Command Files and Startup File.

```
CONTROL/-CP,TI:/SH=CONTROL,CNTLSB,DRSOUT,GLIB/LB
/
PAR=GEN
COMMON=DTA:RW
COMMON=DEVPAR:RW
ASG=GR0:1
//
```

```
COMMN=COMMN
/
PAR=GEN
COMMON=DTA:RW
//
```

```
[1,1]DTA/PI,TI:/SH,SY0:[1,1]DTA/-HD=DTA
/
STACK=0
PAR=DTA
//
```

```
 IP>TI:=[1,2]STARTUP.CMD


SET /NOPRIV=TT1:
SET /BUF=CL:132.
INS [1,1]DTA/PAR=DTA
INS [200,200]COMMN
REA COMMN 4 TT1:
INS [200,200]CONTROL
;*****GIVE ME THE TIME AND DATE*****
;*****FOR INSTRUCTIONS TYPE @PREP*****
```

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

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

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

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

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

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

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

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

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

Appendix VI

Graphic Display Routines (line diagrams)

```
C
C     PROGRAM TO WRITE THE INTRODUCTION AND ENTRY MENU FOR THE
C     MAIN SUPERVISORY ROUTINE (TSK 1) PROGRAM.
C
      COMMON/DFILE/IBUF(1000)
      CALL INIT(1000)
C
C     WRITE THE TEXT FOR EXPLANATION STARTING WITH TITLE.
C
      CALL APNT(85.,850.,-1,-5)
      CALL STAT(-1)
      CALL TEXT('INTERACTIVE GRAPHICS CONTROLLED POWER SYSTEM
     1 SIMULATION:',1,'          A DISTRIBUTED MINI/MICRO PROCESSING
     2 APPROACH.',1)
      CALL STAT(1)
C
C     WRITE TEXT.
C
      CALL APNT(50.,750.,-1,-5)
      CALL TEXT(' THE LIGHT PEN ATTACHED TO THE GRAPHICS TERMINAL
     1 IS USED TO',1,'INTERACT WITH THE POWER SIMULATOR IN FRONT OF YOU.
     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 BREAKER,',1,'SEND A REQUEST FOR DATA ACQUISITION
     3 OF CERTAIN VARIABLES',1,'OR TO CHANGE  PICTURES
     4   CORRESPONDING TO THE SIMULATOR.',1)
      CALL TEXT('"YOU SURE" QUESTION SERVES THE PURPOSE OF DOUBLE
     1 CHECK. ',1,'THE CIRCUIT BREAKERS WILL OPEN
     2 IF CLOSED OR 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     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(900.,800.,-1,-5,1)
      CALL TEXT('YOU SURE?')
C
C     DRAW THE "YES" SUBPICTURE.
C
      CALL SUBP(91)                             !SUBP: CK IF YES
      CALL APNT(915.,750.,1,-5,-1)
      CALL TEXT('YES')
      CALL ESUB
```

```
C
C       DRAW THE 'NO' SUBPICTURE.
C
        CALL SUBP(92)
        CALL APNT(965.,750.,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
```

```
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(1009)                              !SUBP: RAISE VOLT GEN 1
      CALL VECT(0.,30.)
      CALL ARROWU(1,4,-1)
      CALL ESUB
      CALL APNT(290.,170.,1,-4)
      CALL SUBP(1010)                              !SUBP: LWR VOLT GEN 1
      CALL VECT(0.,-30.)
      CALL ARROWD(1,4,-1)
      CALL ESUB
      CALL APNT(375.,150.,-1,-5)
      CALL TEXT('PHASE WRT IC')
      CALL APNT(215.,100.,-1,-5)
      CALL TEXT('FREQ')
      CALL APNT(290.,60.,1,-4)
      CALL SUBP(1011,1009)                         !SUBP: RAISE FREQ GEN 1
      CALL APNT(290.,50.,1,-5)
      CALL SUBP(1012,1010)                         !SUBP: LWR FREQ GEN 1
      CALL APNT(385.,80.,-1,-5)
      CALL TEXT('VA')
      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 EXCIT ON
      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('  VA')
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('  VA')
C
C     WRITE THE DATA FOR INTERCONNECTION.
C
      CALL APNT(600.,260.,-1,-5)
      CALL TEXT('INT CON(IC)')
      CALL APNT(650.,220.,-1,-5)
      CALL STAT(1)
      CALL NMDR(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 APNT(900.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(950.,540.,-50.,1002,'P','Q','R',
     1 'E','I','SYNCRN','MENU')
C
C     SAVE THE PICTURE IN A FILE CALLED 'GEN1.DPY'.
C
      CALL SAVE('GEN1.DPY')
C
      STOP
      END
```

```
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)
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')
      CALL TEXT('A1-3')
      CALL APNT(314.,700.,-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     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('PHASE WRT IC')
      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('VA')
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('PHASE WRT IC')
        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('VA')
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
```

```
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,'GEN1','Q','R','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(750.,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
```

```
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)               ! PB 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.,830.,-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
      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
      CALL APNT(915.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(965.,540.,-50.,805,'E','V','I',
     1 'U','STAT 1','GEN 1','MENU')
C
C     SAVE THE PICTURE ON A FILE CALLED "STAT11.DPY".
C
      CALL SAVE('STAT11.DPY')
C
      STOP
      END
```

```
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(550.,620.,-1,-5)
      CALL NMBR(950,VOLTS,5,'(F5.2)')
      CALL STAT(-1)
      CALL TEXT(' VOLTS')
      CALL STAT(1)
C
      CALL APNT(900.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(950.,540.,-50.,304,'U','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 'SUBST4.DPY'.
C
      CALL SAVE('SUBST4.DPY')
C
      STOP
      END
```

```
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
      CALL APNT(900.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(950.,540.,-50.,404,'W','MENU')
C
C
C     ADD THE BUS VOLTAGE.
C
      CALL APNT(400.,615.,-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
```

```
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 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.,30.,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 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 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       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.,205.,-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.,235.,-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.
        CALL APNT(915.,600.,-1,-5)
        CALL TEXT('SCHEDULES')
        CALL MENU(965.,540.,-50.,1150,'U','V','GEN1',
     1 'STAT1','MENU')
C
C
C       SAVE THE PICTURE IN A FILE CALLED 'SYNCRN.DPY'.
C
        CALL SAVE('SYNCRN.DPY')
C
C
        RETURN
        END
```

```
C
C     PROGRAM TO DRAW THE INDUSTRIAL LOADING PICTURE.
C     THIS IS NOT COMPLETE REPRESENTATION OF THE SUB-SECTION
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: DP SW LTG LD
      CALL APNT(100.,230.,-1,-4)
      CALL VECT(0.,-30.)
      CALL SUBP(720)                            !SUPD: 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: DP 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 CHFCK
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 APNT(895.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(950.,540.,-50.,705,'U','V','STAT1',
     1 'GEN 1','STAT 11','MENU')
C
C     SAVE THE PICTURE IN A FILE CALLED 'INDLD.DPY'.
C
      CALL SAVE('INDLD.DPY')
C
      STOP
      END
```

```
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(550)                          !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(551)                          !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
        CALL STAT(1)
        CALL APNT(900.,600.,-1,-5)
        CALL TEXT('SCHEDULES')
        CALL MENU(950.,540.,-50.,552,'U','MENU')
C
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.,870.,-1,-5)
        CALL NMBR(958,VOLTS,5,'(F5.2)')
        CALL STAT(-1)
        CALL TEXT(' VOLTS')
C
C
C       SAVE THE PICTURE IN A FILE CALLED 'LDCTR.DPY'.
C
        CALL SAVE('LDCTR.DPY')
C
        STOP
        END
PIP>
```

```
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
      CALL APNT(900.,600.,-1,-5)
      CALL TEXT('SCHEDULES')
      CALL MENU(950.,540.,-50.,605,'U','V','W','MENU')
C
```

```
C     PLACE THE DATA TO BE MONITORED.
C
      CALL STAT(-1)
      CALL APNT(250.,600.,-1,-5)
      CALL TEXT('L1-N 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(400.,560.,-1,-5)
      CALL NMBR(962,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
```

Appendix VII

Publications resulted from this research.

Interactive Graphics Controlled Power System Simulation: A Distributed Mini/Micro Processing Approach. ESHWAR PITTAMPALLI, JOHN E. FAGAN, University of Oklahoma--An interactive electric power system model using distributed processing employing a graphics terminal as an input/output medium is investigated. A real model power system is interfaced to a microcomputer, a minicomputer and a graphics terminal. The computers duo are assigned two different specific tasks on a dynamic basis, which in turn are interfaced through a communication network. The microcomputer services the needs of data acquisition of all the state variables of the power system model. The minicomputer acts as a master computer in delegating certain scheduled routines of data acquisition and control to the microcomputer and in operating necessary control functions of the power system model, It also updates a graphic display of power system through manual interaction with the graphics terminal. The control software of minicomputer uses a real time representation of the power system configuration, to do such tasks as synchronization and automatic scheduling. This system design and research provides a very useful contribution to the control system design of operation of an electric power system both in education and research.

ACM Computing Reviews
Category number(s)

3.24
3.89
8.2
4.32

List major key words
in order of importance

Interactive graphics
Distributed Processing
Mini/Micro computers
Data Acquisition
Control System

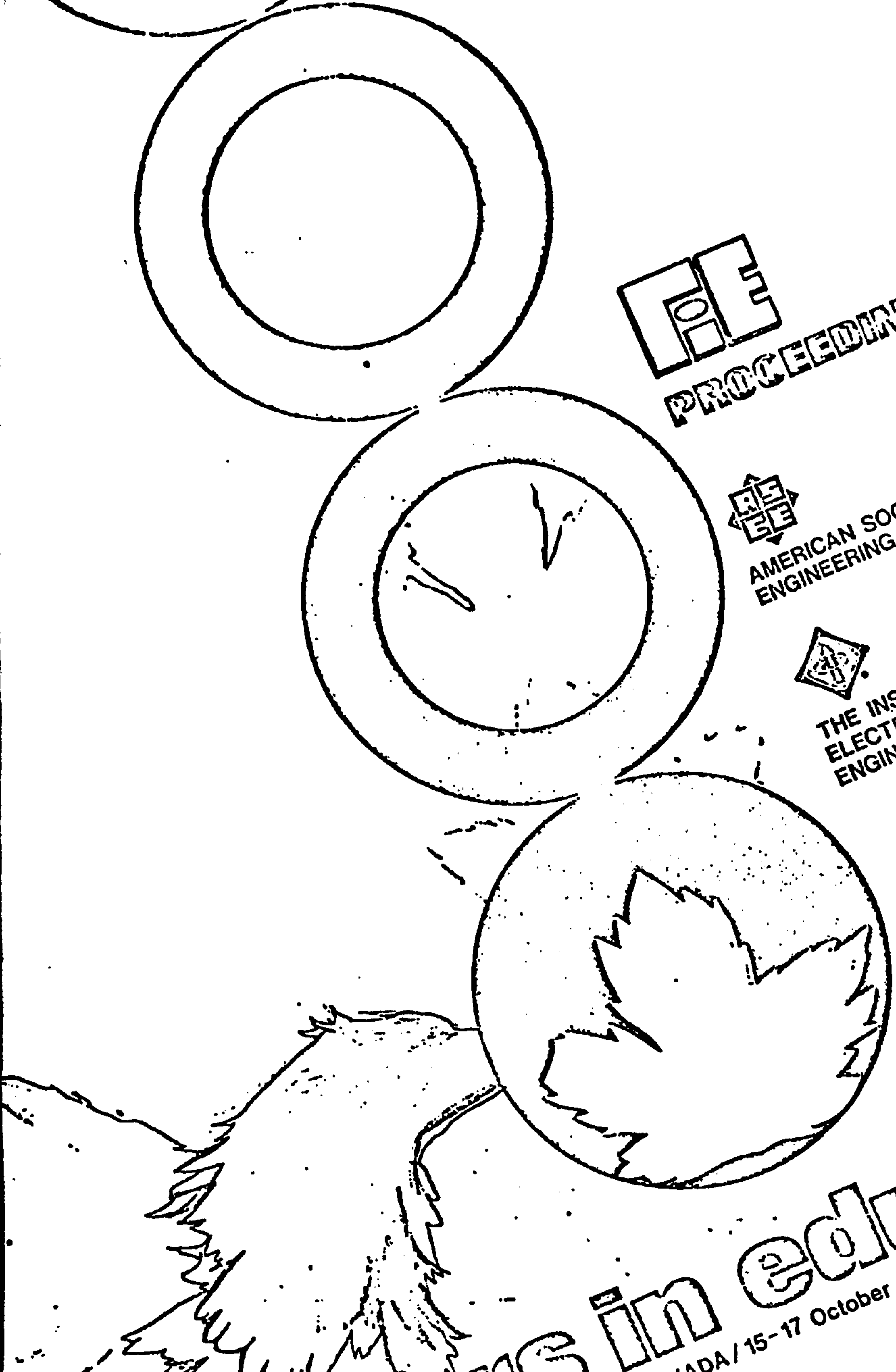
Eshwar Pittampalli or John E. Fagan
Name of Person Presenting Paper

Eshwar Pittampalli
John E. Fagan
Authors Signature

Eshwar Pittampalli & John E. Fagan
Author's Name

Overhead Projector needed

School of Electrical Engineering
and Computing Sciences
University of Oklahoma
202 West Boyd, Room 219
Norman, Oklahoma 73019
Author's Complete Mailing Address



FIE PROCEEDINGS

AMERICAN SOCIETY FOR ENGINEERING EDUCATION

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.

frontiers in education

NINTH ANNUAL CONFERENCE Niagara Falls, CANADA / 15-17 October 1979

Lawrence P. Grayson
Joseph M. Biedenbac
editors

DIGITAL COMPUTER CONTROLLED POWER SYSTEM SIMULATOR USING DISTRIBUTED MINI/MICRO PROCESSING

Eshwar Pittampalli
School of Electrical Engineering
and Computer Sciences
University of Oklahoma
Norman, Oklahoma

Dean John E. Fagan
Dean of College of Engineering
College of Engineering
University of Oklahoma
Norman, Oklahoma

ABSTRACT

An interactive electric power system model utilized in the power systems program at the University of Oklahoma using distributed processing employing a graphic terminal as an input/output medium is investigated. A real model power system is interfaced to a microcomputer, a minicomputer and a graphics terminal. The computer duo are assigned two different specific tasks on a dynamic basis, which in turn are interfaced through a communication network. The microcomputer services the needs of data acquisition of all the necessary state variables of the power system model. The minicomputer acts as a supervisory system in delegating certain scheduled routines of data acquisition and the control to the microcomputer and also in operating certain necessary control functions of the power system model. In addition, it also updates a graphic displays. The control software of minicomputer uses a real time representation of the power system configuration, to do such tasks as synchronization and automatic network, load and generation scheduling.

BACKGROUND

Power system generation, transmission and distribution networks coupled by switching and load center substations, make up a system that covers a vast geographic area. Contingencies, whether scheduled or emergency, which involve the removal of any essential component of the power system, such as transmission line or generation can cause large disturbances in the network, and eventually lead to failure of a portion or the entire network. During these contingencies, the system's automatic controls operate to protect the system from overload or damage. The human controller monitors and makes the decisions as to load shedding, generation scheduling, or total system rescheduling, attempting to keep the system supplying power to the user. In an effort to enhance the control of the electric power system, the digital computer and related devices are being used extensively to better perform data acquisition and supervisory control activities.

Investigations were made to define present substation's automatic requirements. This included a complete evaluation of substation functional and control requirements. Models to dynamically simulate and predict power system's present and future states have been incorporated to implement real-time control. In this application the tool of an interactive graphics has been selected to provide near real-time communication between a human operator, the computer and the power system being controlled. A single line representation of real power system model along with line diagrams of various power system components of entire real life power system are depicted on interactive terminal with purpose of coordinating the control of displays on graphical representation to the actual system.

INTRODUCTION

The mini/micro computer based system is designed to control the operations of a model power system simulator. The minicomputer supports graphics display and two other tasks, one of which establishes communication link with microcomputer system. The microcomputer with a real time interface system scans 32 state variables of the model power system providing the digital equivalent data to the minicomputer through a common block of data, which resides a shared memory region. This enables the minicomputer to update the graphic displays of power simulator. The multi-tasking capability of minicomputer facilitates the simultaneous run of two tasks independent of each other through its zero power condition to steady state operation.

The sequence of operation starts with disabling of all the manual controlled switches of the simulator. These are replaced by automatic switches which can be controlled by the minicomputer through D.M.A. interface circuitry. The operator either closes or opens a particular switch of interest through the help of a light pen attached to the graphics terminal. The photographs of the minicomputer system/simulator are shown in Figures 1a and 1b.

SYSTEM HARDWARE

A block diagram of the system interconnection is shown in Figure 2. The minicomputer employed is a PDP 11T34 Digital Equipment corporation system which runs with VT-11 graphic display unit. It includes a LA36 DEC typewriter, an input/output terminal, and two RK05 magnetic disk cassette drives. The 11T34 is connected to the peripherals by unibus system. The data transfer takes place over this bus. The memory contains 64k words of semiconductor memory, of which less than 32k is

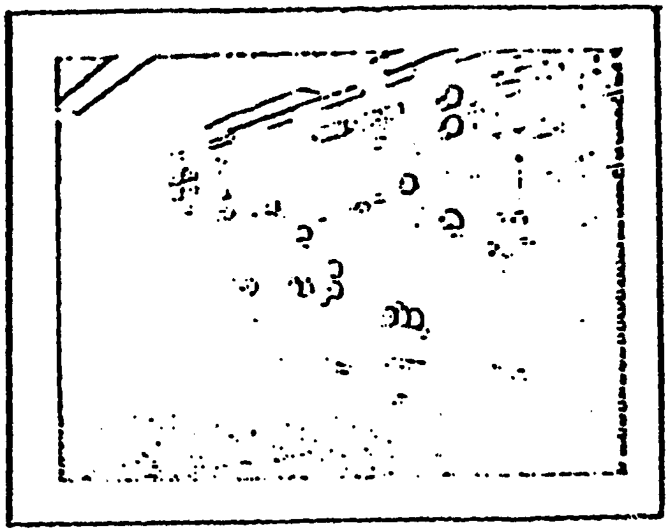


FIGURE 1a

POWER SYSTEM SIMULATOR

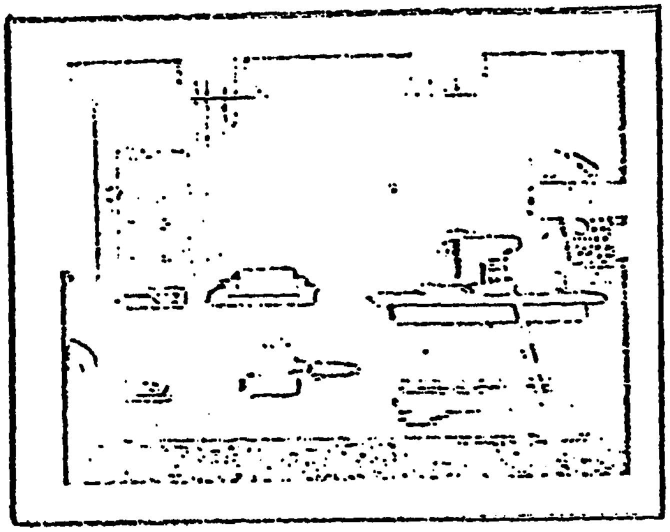


FIGURE 1b

DEC 11T34 SYSTEM

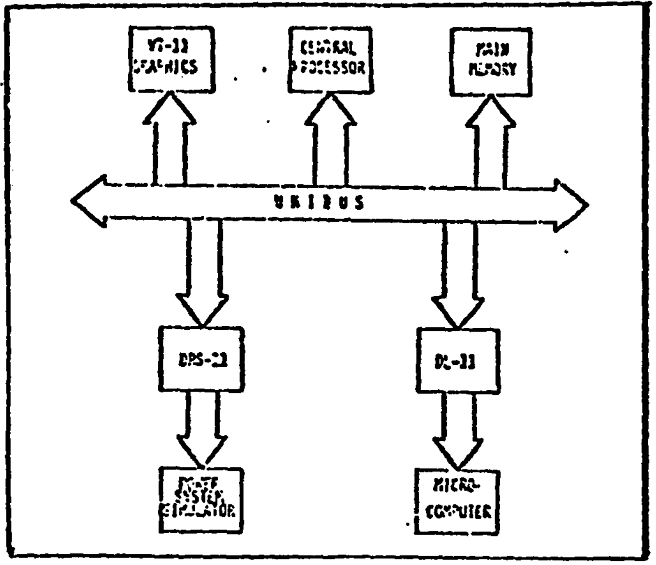


FIGURE 2

HARDWARE INTERCONNECTION MAP

needed to perform the task described in this paper. The operating system is the RSX-11M with multi-tasking capability.

The VT11 graphics terminal is a single color, variable intensity random position scan CRT terminal for real time graphics display. The screen is 17" diagonal and defines 1,023 x 1,023 coordinate points in both X- and Y-directions. The terminal contains a display processor which acts as any other peripheral device on the UNIBUS. The VT11 will also issue interupts to the CPU when it detects that the light pen has been pointed in the sensitive area of the screen.

The data acquisition system is based on SBC 80/10A single card computer system. This microcomputer system is interfaced to the power simulator through a real time interface sub-system Analog Devices RTI-1200. This interface sub-system uses memory mapped interface thus appearing to the microcomputer as a block memory addresses in the memory space. Data and command information is transmitted to the RTI-1200 via instructions that write into memory, and data and status information is retrieved from RTI-1200 via instructions that read from memory.

Simulator is a small power system model consisting of scaled generators, transmission systems, substations, distributed systems, and loads. The generation is three dc motor/synchronous generator sets of 1Kw, 1Kw and 3Kw capacity. Additional capacity may be supplied through the simulator system interconnection to the local utility. A high voltage transmission system connects the generation to two types of distribution substations where varied forms of residential and industrial loads are simulated. There is also a network system fed from both substations and a load center substations as well. The circuit breakers and switches are accompanied by special modifications. The breakers have a remotely controlled contactor in series, thus allowing breaker and circuit status to be altered under program or manual control. The remote switching is routed to a data level conversion panel where special buffer circuits change the 5 volt TTL signals to 115 volt signals for switching. This is accomplished with the help of an interface circuitry which performs the voltage level changes from that of 5 volt TTL signals available on the DRS-11 to 115 volts AC commands needed to operate controls within the simulator. The purpose of interface, in addition to level change, is to offer a high level of isolation between the system simulator and the computer. In order to accomplish this a relay/triac arrangement has been designed. The circuit of one of the 96 channel arrangements is shown in Figure 3.

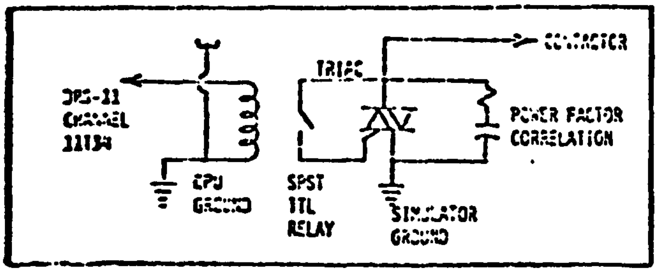


FIGURE 3

SAMPLE INTERFACE CIRCUIT

The microcomputer used is Intel Corporation SBC 80/10A. The microcomputer is interfaced to 32 state variables of the power system simulator through a real time interface RTI-1200. This analog to digital converter has capacity of sampling 32 channels, single ended or 16 channels differential. The distribution of hardware processing systems interactively communicate through the asynchronous DL-11A interface. The control software of the minicomputer unifies and integrates the distributed components. The microcomputer services the needs of the data acquisition of all the state variables of the power system model. The interconnection of the total system is shown in Figure 4.

SYSTEM SOFTWARE

A block diagram of the software structure is shown in Figure 5. The software consists of three basic components, which are Supervisory Routine (TSK1), Command and Communication Routine (TSK2) and Simulator Controller Routine (TSK3). TSK1 is the main routine of the entire software structure. TSK2 is the program which establishes the communication link with the microcomputer through DL-11A interface. TSK1 and TSK2 are connected with the common global data called DATA. Functions of TSK1 software include support of the graphics routine and the operation of the updating of the system variable values in the display. The status of the 32 variables is stored after sampling, normalizing in the common global area by the TSK2. TSK3 controls the status of the functional switches of the power simulator, through DRS-11 interface. The CRT display provides transparent medium to view the communication protocol between the mini and micro. The decwriter provides the medium of transmitting the necessary commands to the minicomputer.

A. Supervisory Routine (TSK1)

This is the main control program for the interface between the graphics terminal and the simulator, ref. Fig.7. It coordinates commands issued from the graphics terminal with the simulator commands and keeps the displays updated with current switch positions and data. This program is on-line to the simulator. The first step of this program is to display the menu of the entire displays. After updating the selected display the loop for monitoring data and waiting for light pen hits is entered. If there is a light pen hit, it is ascertained whether the hit corresponds to a request which results in displaying already stored display or a request to change the status of certain switch, which controls opening or closing of relay/switch of the power simulator. If the request is for the latter, then it communicates with TSK3 through common block of data called SW(96) and updates the requested relay/switch status. This terminates the loop and program jumps back to

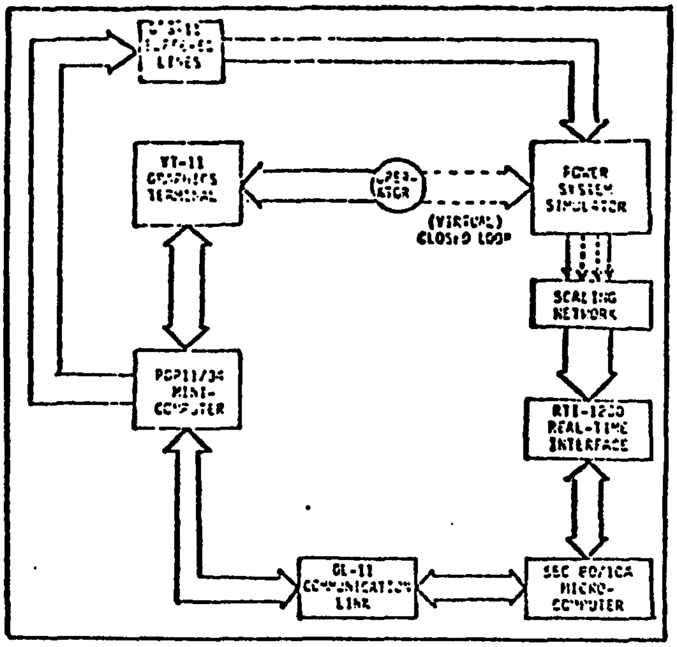


FIGURE 4

TOTAL SYSTEM INTERCONNECTION MAP

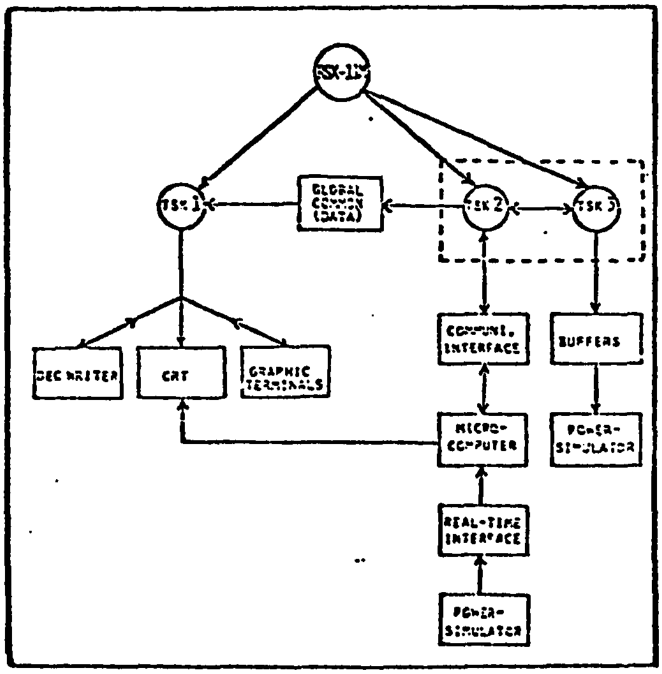


FIGURE 5

SOFTWARE STRUCTURE

the stage of updating the display's current status and waits for the light pen hit to occur, as described earlier.

B. Command and Communication Routine (TSK2)

The flow chart of TSK2 is shown in Figure 6. The program begins with the introductory message printed on the console (DEC writer) explaining the details of the interactive nature of the program. After accepting the command regarding the schedule of sampling, it tests the validity of the command, if it passes, command is transmitted to the microcomputer system. Then the program waits for the arrival of digital data corresponding to the variables requested. This data is checked against transmission errors with a simple checksum byte of data. If the data received is found to be correct then it is normalized to unit scale. Normalized data is stored in a global common data which is a common memory region shared by TSK2 and TSK3. This data is used to update the displays on the graphics console as described before. This activity is looped for a fixed number of times (N), which is a software variable parameter, before a decision is taken whether to continue the same parameters sampling or not. If it is desired to continue the same for another set of N times, the process continues, otherwise counter is reset to zero and the command for a change of variables is accepted.

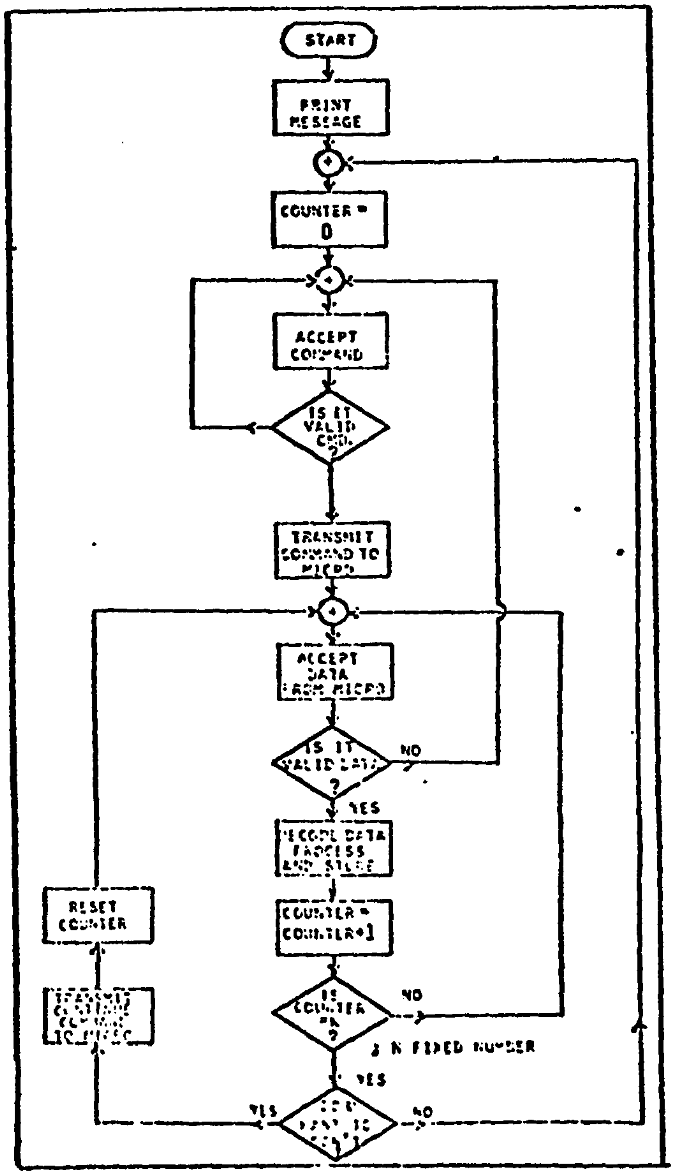


FIGURE 6

COMMUNICATION ROUTINE FLOW CHART

C. Simulator Controller Routine (TSK3)

This program controls the status of various switches involved in the power simulator. The status of a particular switch is derived from the global common array SW(96). This array maintains the current status of all the controlling switches of the simulator. The opening or closing of a

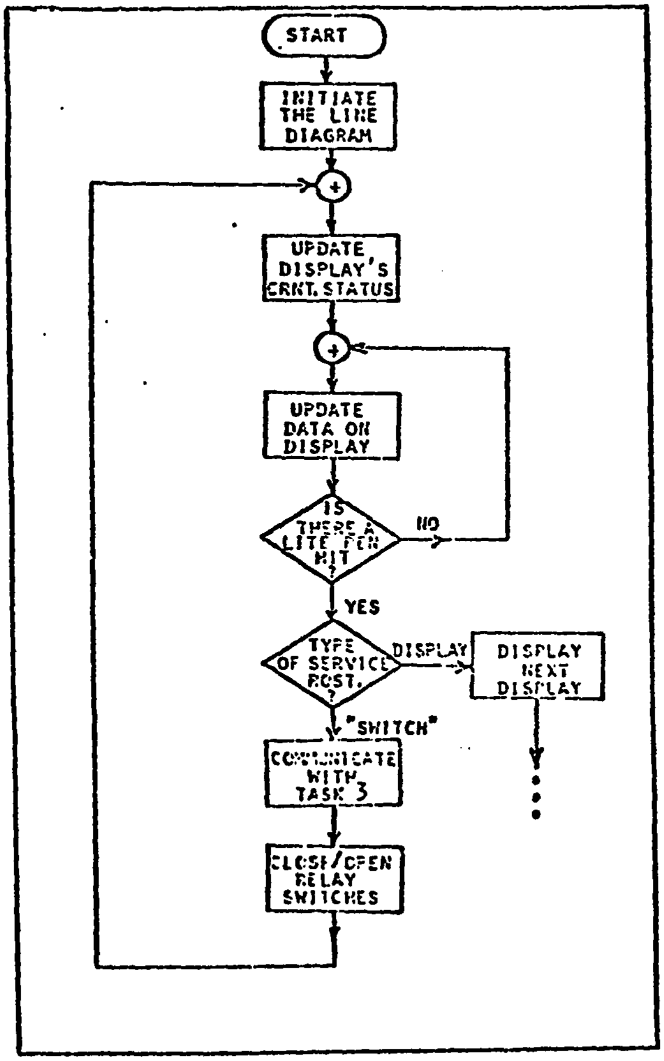


FIGURE 7

SUPERVISORY ROUTINE FLOW CHART

certain switch is decided by this program. The operator's interaction is transferred as a change in the value of one of the elements of the common array SW(96). TSK3 senses this change and acts accordingly in updating the status of a certain switch of the power simulator.

D. Microcomputer Sampling Routine

A simplified flowchart of microcomputer sampling routine is shown in Figure 9. Sequence begins by initialization of the global common data. Next the program waits for the command from the console for the required sampling schedule. The command follows a definite structure as shown in Figure 8a.

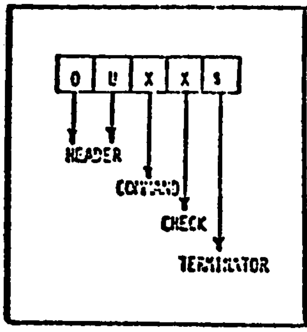


FIGURE 8a

COMMAND FORMAT

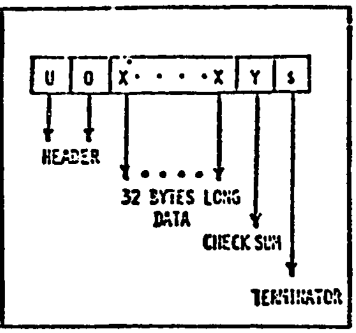


FIGURE 8b

DATA FORMAT

The command header consists of two letter OU followed by the command itself. The command letter is repeated twice in order to make sure the operator's decision and this results in a simple verification process at the receiving end. The command is terminated by S term. If the received command at the microcomputer differs in any respect from that of command structure format shown in Figure 8a, it will be rejected and denial of the execution of the command is reflected on the console. If the command is decoded as a valid command, based on the predetermined schedule routines, Table 1, Appx. A, the sampling process is carried out by the micro. The sampled variable information is transmitted to the mini in the format shown in Figure 8b. The data word is 33 bytes long with 32 bytes of the sampled data followed by the check sum byte, which is the complemented sum of the 32 bytes of data. The data word is terminated by the terminator S. At this time the micro is in receiving mode for either the continuation command or the termination command. If it is a continue command the program idles for some time before it resumes its sampling process. This allows the operator to follow closely the parameter value changes on the graphic display. If it were to be a terminate command the mini goes to the beginning of the program waiting fot the input from the console.

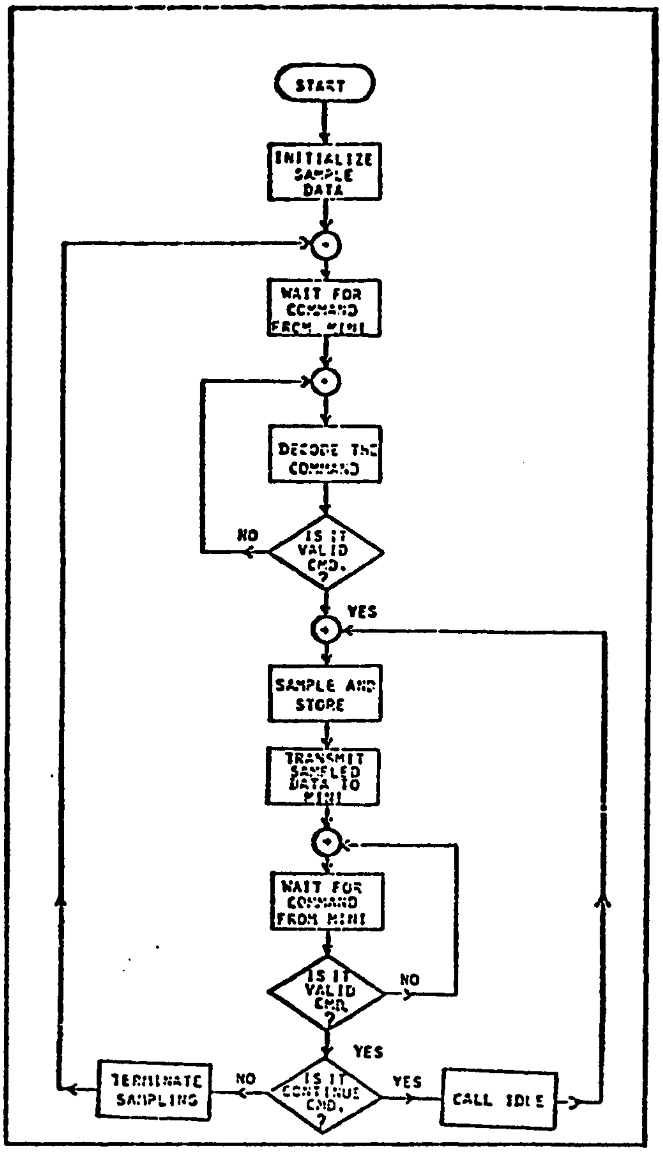


FIGURE 9

MICROCOMPUTER SAMPLING ROUTINE
FLOW CHART

SYSTEM PERFORMANCE AND UTILIZATION

The system hardware and software described herein has been operational very satisfactorily here at the University of Oklahoma. The major obstacles have occured in the system interface in the riding of the triac circuits of interchannel cross talk. Regarding operating system of the minicomputer is concerned the RSX-11M has been operationally good, but bit manipulations in TSK3 have necessitated the use of Assembly language routines. It has been also decided in this course of trials that the use of the Bell's UNIX operating system is the next upgrade for the total system, UNIX provided easier inter task communication capability and is much better multi tasking development system.

The total system is utilized extensively in the power systems program at the University of Oklahoma. The system is utilized in research as well as undergraduate and graduate level teaching. Built around the system simulator, and the system simulator with the attached computer, a group of programmed undergraduate level experiments which are performed by the student electing to pursue courses in energy conversion and power systems area. System simulator can be utilized in the manual mode as well as the automatic mode, and the group of experiments taught utilizing the simulator and the controllor range from the study of synchronous machines, in the steady state as well as in the transient state, the study of the power system network utilizing load flow techniques and transient stability techniques, the study of the characteristic of loads and how they affect the power system operation. Additionally, experiments include those in automatic generation control, power system emergency operation, system relaying and protection, and total systems security techniques. The system has been an invaluable teaching and research tool utilized in combination with the Faraday machines in the laboratory.

CONCLUSIONS

Although hardware and software are in a continual state of evolution, the tool serves as an operational tool for instruction and research. Further work which is progress include the linking of 11T34 with 11/70, The College of Engineering computer system. This addition of computing capabilities will hopefully result in a system in which the on-line simulators will be accomplished in the 11/70 on request from 11T34.

Future work includes an auxiliary software package for the 11/70 providing more complex system simulation where 11T34 system turn in to a purely controlling role for the purpose of operator training for normal and emergency network modes.

REFERENCES

1. Davis, L.R. and Fagan, J.E., "Digital Control of Power System Simulator," Page 48-52, Control of Power Systems Conference, Record No. 78CH1282-3, Reg. 5.

2. Russel, B.D., "A Microcomputer-Based Substation Control System," Page 141-145, Control of Power Systems Conference, Record No. 76, CH1057-9, Reg. 5.

3. "Central Computer Control and Protection Functions," IEEE Computer Relaying Subcommittee, 1976.

4. Jones, J.L., "The Application of Microprocessors to Power System Data Acquisition and Control," Control of Power Systems Conference. Pages 149-153, Record No. 77CH1168-4, Reg. 5.

5. Farag, A.S.T., "Real-Time Digital Control Scheme," Pages 87-93, Control of Power Systems Conference, Record No. 76CH1057-9, Reg. 5.

6. Harley, R.G., et. al., "Real-Time Computer Control of a Model Turbo Alternator," Pages 5-9, Control of Power Systems Conference, Record No. 77CH1168-4, Reg. 5.

7. Bjork, D.R., et. al., "Digital Synchronizer for Computer Controlled Generating Stations," Pages 1-4, Control of Power Systems Conference, Record No. 77CH1168-4, Reg. 5.

8. Abegglen, P.C., et. al., "Design of a Real-Time Control Data Acquisition and Analysis System," Pages 38-48, Proc. IEEE, Vol. 58, January 1970.

9. Shapiro, M. and Schultz, A., "A Laboratory-Based Computer System," Pages 398-405, Anal. Chem., Vol. 43, March 1971.

10. Enslow, Jr., P.H., "What is a 'Distributed' Data Processing System?" Pages 13-21, Computer, January 1978.

11. Machover, C., et. al., "Computer Graphics Displays," IEEE Spectrum, August 1977.

12. Walker, B.S., et. al., "Interactive Computer Graphics," New York: Sane, Russack & Company, Inc., 1975.

13. Machover, C., et. al., "Graphics Displays: Factors in System Design," IEEE Spectrum, October 1977.

14. Introduction to RSX-11M, DEC, Maynard, March.

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

16. VT-11 Graphic Display Processor, DEC, Maynard, March.

17. Runtz, K.J., et. al., "Digital Control Scheme for a Generating Unit," IEEE Trans. Power Apparatus and Systems, Vol. PAS-92, No. 2, March/April, 1973, Pages 478-483.

APPENDIX A - TABLE 1

SCHEDULE EXAMPLES

Schedule	Variables
A	All channels
B	Gen 1 V, Gen 1 f, T.L.V.*
E	Gen 2 V, Gen 2 f, T.L.V.
F	Gen 3 V, Gen 3 f, T.L.V.
H	Gen 1 V, Gen 2 V, Gen 3 V
J	Gen 1 I, Gen 2 I, Gen 3 I
K	Gen 1 p.f., Gen 2 p.f., Gen 3 p.f.
P	Dist. V, Gen 2 V, Gen 3 V

*Tie Line Voltage

INTERACTIVE GRAPHICS CONTROLLED POWER SYSTEM SIMULATION: A DISTRIBUTED MINI-MICRO PROCESSING APPROACH

Eshwar Pittampalli
School of Electrical Engineering
and Computing Sciences
University of Oklahoma
Norman, OK 73019

John E. Fagan
School of Electrical Engineering
and Computing Sciences
University of Oklahoma
Norman, OK 73019

ABSTRACT

The purpose of this paper is to present one aspect of continuing research program at the University of Oklahoma for the study of real-time power system control. The paper presents the result of the integration of a digital equipment PDP 11T34 computer system with full graphic support, and an Intel SBC 80/10A microcomputer based data acquisition system to a specially designed power system simulator.

The power system simulator, DEC PDP 11T34 mini-computer, and Intel SBC 80/10A microcomputer combine to form a unique research tool for studying power system static and dynamic behavior as well as a test bed for real-time control scheme evaluations. The power system simulator is composed of actual power system components: relaying, switching, metering, and remote data acquisition equipment. In addition, it is also equipped with an SBC 80/10A microcomputer based data acquisition system, remotely programmable switching, and system protection components. The PDP 11T34 system by means of TTL discrete output and with the aid of the microcomputer, can perform real-time supervisory control and data acquisition, either in a closed-loop or open-loop on the power system simulator. This paper will detail the hardware and software aspects of the system simulator and its control, and relate the teaching success at the graduate/undergraduate level during the development of this teaching and research tool. It will also present additional research aspects of the entire system in such areas as AGC on-line load flow and stability studies, automatic contingency analysis and avoidance, and real-time scheduling of generation, loading, and transmission.

BACKGROUND

Power system generation, transmission, and distribution networks coupled by switching and load-center substations, make up a system that covers a vast geographic area. Contingencies, whether scheduled or emergency, which involve the removal of any essential component of the power system, such as a transmission line or generation can cause large disturbances in the network, and eventually lead to a failure of a portion or the entire network. During these contingencies, the system's automatic controls operate to protect the system from such overload and damage. The human controller monitors and makes decisions as to load-shedding, generation scheduling, or total system rescheduling, attempting to keep the system supplying the power to the user. In an effort to enhance the control of the electric power system, the digital computer and related devices are being used extensively to better perform data acquisition and control activities. Utilization of central computing facilities or distributed computing facilities in power system on-line dispatch and control, are utilized by numerous utilities. Within the United States, the power system, with the on-line controllers and data acquisition systems are seldom used in close-loop control, even though smaller closed-loop operations are being accomplished, such as automatic generation control.

The problem encountered by operations and system planning, is the inability to do system testing and checking of closed-loop algorithms and real-time on-line analysis tools on actual power systems. In the event that these algorithms fail, then often catastrophic system failures occur which are detrimental to the system's performance and customer service. It is with this in mind that this project was originated. A modified form of a Hampton power system simulator was built (Figure 1). This power system simulator was equipped with remotely programmable switching, relaying, and generator control. Additionally, critical state variables within the power system simulator were brought out to a central point for easy access by data acquisition arrangements. A small minicomputer, PDP 11T34, (Figures 2a and 2b),

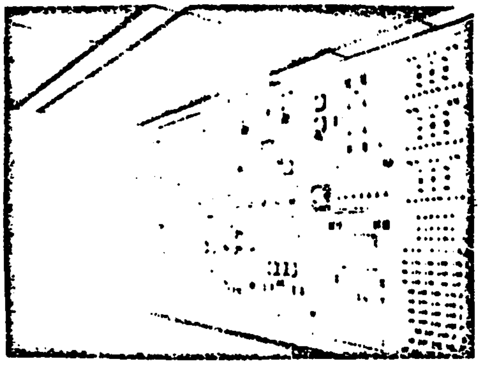


FIGURE 1

POWER SYSTEM SIMULATOR

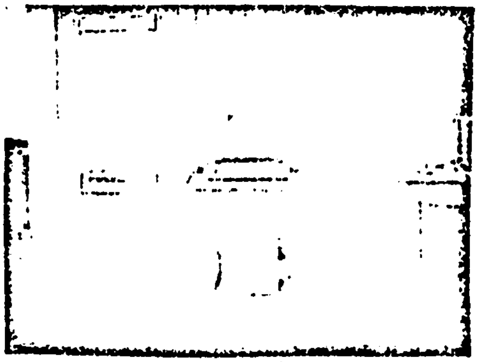


FIGURE 2a

DEC 11T34 MINICOMPUTER

This paper was also accepted for presentation at Computers in Education, Düsseldorf, West Germany.

capable of handling the on-line control problem and graphic system data display, plus real-time system analysiu interfaced to the power system simulator.

FIGURE 2b

VT-11 GRAPHICS DISPLAY

A microcomputer was added to the system to perform the data acquisition. The SBC 80/10A microcomputer system performs scheduled data acquisition tasks based upon the status of the system and the current control task being performed. The 11T34 system is coupled to the SBC 80/10A through a DL-11 asyncnronous interface. The 11T34 system performs the data display and acquisition control. On-line system simulation is accomplished as a separate task in the PDP 11T34.

POWER SYSTEM SIMULATOR

The simulator (Figure 1), is a small power system consisting of scaled generators, a transmission system, a subtransmission system, substations, and loads. The generation is in the form of three DC motor/synchronous generator sets of 1 KW, 1 KW, and 3 KW capacity. Additional tie capacity may be supplied to the simulator through a system interconnection. The high voltage (600 volts) transmission 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 special modifications allowing them to be operational by the computer on command.

COMPUTER

The Digital Equipment Corporation PDP 11T34 system (Figure 2), includes a DEC writer typewriter, input/output terminal, and two RK05 magnetic disk cassettes of 1.5 megabytes each. The CPU is equipped with 64K words of core storage, two DRS-11 TTL output boards, and three DL-11 asynchronous line interfaces. The system is also equipped with a VT-11 point vector, graphics display terminal. The operating system is Digital Equipment Corporation's RSX-11M which has a realtime multi-tasking capability.

SIMULATOR INTERFACE

The simulator interface circuitry is used to perform level changes between the 5 volt TTL signals available on the DRS-11 board of the 11T34 system and the 115 volt AC commands needed to operate controls within the simulator. The purpose of the interface, in addition to level converstion from the DRS-11 TTL signals to the 115 volt control signals, is to offer a high-level of isolation between the system simulator and the computer. In order to accomplish this, a relay/triac arrangement has been utilized. The circuit for one of the 96 channels is shown in Figure 3.

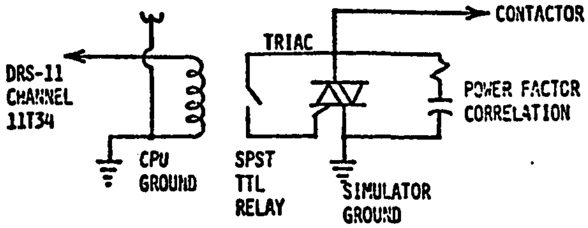


FIGURE 3

SIMULATOR INTERFACE CIRCUIT

MICROCOMPUTER DATA ACQUISITION SYSTEM

Figure 4 shows a schematic representation of the microcomputer-based data acquisition system used. The SBC 80/10A is a complete computer system on a single printed circuit board. The CPU, system clock, read/write memory, non-volatile read-only-memory, I/O ports and drivers all reside on the board. The Real-Time Interface (RTI-1200) is a complete analog I/O subsystem, on a single printed circuit board. This uses memory mapped interface with the microcomputer, thus appearing to the microcomputer as a block of memory locations in the microcomputer's memory address space. Data and command information is transmitted to the RTI-1200 via instructions that write into memory, and data and status information is retrieved from the RTI-1200 via instructions that read from memory.

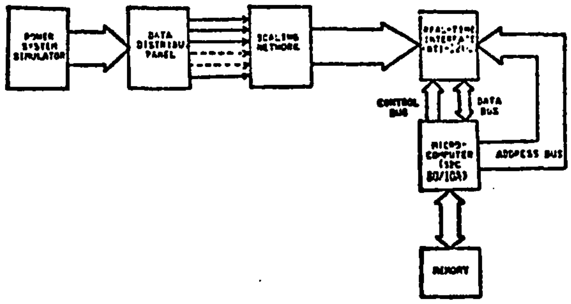


FIGURE 4

ARCHITECTURE OF MICROCOMPUTER-BASED DATA ACQUISITION SYSTEM

As discussed previously, a total of 32 state variables of the power systemsimulator are interfaced to the microcomputer through scaling network and RTI-1200. In general, the data acquisition has two major functions. Normally, data is acquired on all variables of the system and digitally transmitted to the minicomputer, in those special cases, when a particular variable is of utmost importance, the minicomputer directs the microcomputer to sample the variable at a different sampling rate (usually at twice the normal rate of sampling) which has been preset by a prior scheduling decision.

The data on these 32 variables is transferred to common memory block reserved in the minicomputer memory through the DL-11 interface.

SYSTEM DESCRIPTORS

The distribution of processing is done between mini- and microcomputers. This distribution of the physical and logical resources of the system interactively communicate through the asynchronous DL-11 interface. The control software of PDP 11T34 unifies and integrates the control of the distribution components. The microcomputer services the needs of data acquisition of all the state variables of the power system model. The interconnection components of the total system is shown in Figure 5.

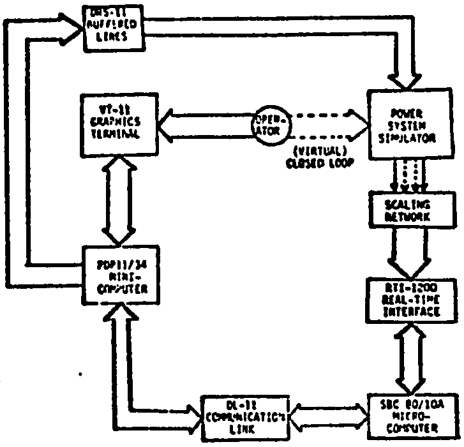


FIGURE 5

SCHEMATIC REPRESENTATION OF THE COMPLETE SYSTEM

SOFTWARE

Graphics

The I/O driver software for the VT11 display operates under the RSX-11M operating system. It forms a set of Fortran callable subroutines that can build display processor buffer. The display processor in the VT11 utilizes this display buffer which resides as a dimensioned common array in the main executive program. The dimension size of the array is limited only by the requirement that it must reside in the lower 28 K words of memory.

All of the displays are built in the form of subpictures and stored on disks. Disk capacity allows a rapid building of a single display picture using pre-drawn components. A subpicture is some part of an overall picture or nested part of another subpicture. The use of subpictures saves construction time of the individual portions of the display speeding display figure deployment.

Executive

The executive control program has two functions (Task 1) (Figure 6): it provides menu selection of the display, system status display, and power systems simulator state variable display. The purpose of this program is to gather commands, from the display device in the form of light pen hits and provide these to selected areas of global switch variables, which maintain a current record of all system switching status, and update this record of switching status based upon interaction of the operator with the display device. The switching status record is maintained as an area in global common access by the controller task software. The executive is entirely written in Fortran under the RSX-11M Fortran four compiler.

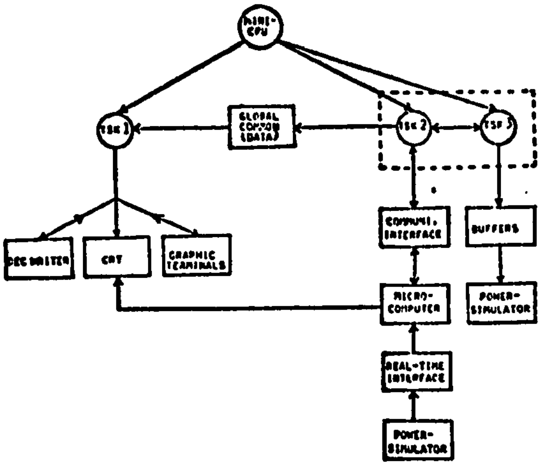


FIGURE 6

FUNCTIONAL OPERATING SYSTEM STRUCTURE

Controller

The second task but, running under the executive, is the DRS-11 controller software (Task 3) (Figure 6). This program's function is to strobe that area of global common, which represents the current switch status established by the executive program, and make available this current switch status on 96 lines of two DRS-11 TTL output boards. The DRS-11 software runs a strobe at once a second unless an update has occurred from the master executive program, then runs immediately upon a flag of an update. The DRS-11 controller program is written entirely in macro-11 assembler.

Micro Communication

The third and final task that runs in the memory is the data acquisition communications interface (Task 2) (Figure 6). This program receives commands through the global common from the executive program as to what type of task is being performed, and thus what type of data acquisition scheme is needed to perform the task. It then selects the appropriate scheduling command for the microcomputer, formats it, and transmits to the microcomputer via the DL-11 and awaits a reply.

The program then halts pending an I/O request from the microcomputer until such scheduled data acquisition can be made. It then receives the updated state variable data, formats it for display purposes into per unit reading, and places them in global common variables area for use by the executive display program.

MICROCOMPUTER DATA ACQUISITION

This software serves two purposes: 1) performs basic protocol operations between the minicomputer and the microcomputer; and 2) based on the command from the mini, it engages in data sampling and data transfer. The microcomputer data acquisitions software consists of a communication loop which strobes the microcomputer command live from the DL-11 interface for a microcomputer command word. After detection of such a command word, checks the validity of the command and then decodes it and takes the necessary action of selecting the variables for sampling. The analog to digital converter employed, Real-Time Interface RTI-1200, uses a memory mapped interface, thus facilitating the command information transmitted via instructions that write into memory, and data and status information is retrieved from it via instructions that read from memory. The sampled state variables data is transmitted to executive software through DL-11 interface using simple data error checking byte attached to the data word.

SYSTEM PERFORMANCE AND UTILIZATION

The system hardware and software described herein has proven reliable in the sense that there has been only one major system failure in the past year of operation. The major obstacles have occured in the system interface in the riding the triac circuits of interchannel cross task. The system is utilized in a teaching and research role involving upper level undergraduate and graduate students.

The microcomputèr data acquisition system has functioned quite well with reliability of both the software and hardware proving no significant problem in the operation of the mini/micro duo.

The RSX-11M operating system, although has been operationally good, has proven to be marginal, in that DRS-11 software support had to be developed for the system. Also, system software development is somewhat crude due to the task builder of the operating system. A possible change of operating system to UNIX is being considered to simplify inter-task communication and scheduling of the executive.

The distributed processing mini/micro control power systems simulator, is used for a variety of purposes at the University of Oklahoma. One of its main purposes is to serve the faculty in research, with the interactive graphics utilized as an I/O tool. The computer controlled power system simulator is additionally used in instruction of the power engineering students and is part of a structural as well as discovery laboratory form. A typical experiment for the student would be that all manual switches are placed in the "On" position and the student is situated at the power system control center. He then merely selects the interconnect and appropriate closing switches from the interconnect, thus bringing the interconnect to the system. The student can schedule or direct by means of the light pen which one of three generation systems is brought on-line. Once the operator selects the generation system to be brought on-line and supply power, the 11T34 directs the microcomputer to a prescribed sampling algorithm and in a closed-loop fashion the microcomputer together with the mini, bring up the speed and put on the line the selected generation units. The student or the operator can then select specific loads to be brought on-line, whether they be distribution, light industrial, or commercial loads. The software will readjust generation to account for varflow, voltage regulation, and automatic frequency control. The student then can select any one of a group of unscheduled contingencies and 1) manually interact with the system to reschedule a system avoidance, and 2) allow the computer to reschedule around the system contingencies. The student is then capable of observing the power system simulator both in a close-loop automatic mode and an open-loop manual control mode. The power system simulator is very successful, both from a manual and an automatic standpoint, and is utilized at the University of Oklahoma in the power system education program. The power system simulator can be utilized by both industry and academic units in both training and research.

CONCLUSIONS AND FURTHER WORK

The combination of the simulator, PDP 11T34, and the SBC 80/10A systems has proven to produce a very unique research and teaching tool for control of power systems. Although hardware and software are in a continual state of evolution, the tool serves as an operational teaching and research tool.

Further development of on-line simulations are needed to bring the package to a complete system. Currently, a second DL-11 is being implemented to link the 11T34 system to the College 11/70 interactive system. This new invention will hopefully produce a system in which the on-line simulators will be accomplished in the 11/70 on request of the 11T34. This is desirable since the 11T34 system does not have the horsepower to do control, graphics display, and more detailed simulators. Both hardware and software aspects to this problem are currently being examined and proves to be a useful extension of the work already accomplished.

Further work includes an auxillary software package for the 11/70 providing more complex system simulation and the 11T34 system cast into a purely control role for the purpose of operator training for normal and emergency network modes.

REFERENCES

1. Davis, L.R. and Fagan, J.E., "Digital Control of Power System Simulator," Page 48-52, Control of Power Systems Conference, Record No. 78CH1282-3, Reg. 5.

2. Russel, B.D., "A Microcomputer-Based substation Control System," Page 141-145, Control of Power Systems Conference, Record No. 76, CH1057-9, Reg. 5.

3. "Central Computer Control and Protection Functions," IEEE Computer Relaying Subcommittee, 1976.

4. Jones, J.L., "The Application of Microprocessors to Power System Data Acquisition and Control," Control of Power Systems Conference, Pages 149-153, Record No. 77CH1168-4, Reg. 5.

5. Farag, A.S.T., "Real-Time Digital Control of Power Systems Conference, Record No. 76CH1057-9, Reg. 5.

6. Harley, R.G., et al., "Real Time Computer Control of a Model Turbo Alternator," Pages 5-9, Control of Power Systems Conference, Record No. 77CH1168-4, Reg. 5.

7. Bjork, D.R., et al., "Digital Synchronizer for Computer Controlled Generating Stations," Pages 1-4, Control of Power Systems Conference, Record No. 77CH1168-4, Reg. 5.

8. Abegglen, P.C., et al., "Design of a Real-Time Control Data Acquisition and Analysis System," Pages 38-48, Proc. IEEE, Vol. 58, Jan. 1970.

9. Shapiro, M. and Schultz, A., "A Laboratory-Based Computer System," Pages 398-405, Anal. Chem., Vol. 43, March 1971.

10. Enslow, Jr., P.H., "What is a 'Distributed' Data Processing System?" Pages 13-21, Computer, Jan. 1978.

11. Machover, C., et al., "Computer Graphics Displays," IEEE Spectrum, Aug. 1977.

12. Walker, B.S., et al., "Interactive Computer Graphics," New York: Sane, Russack & Company, Inc., 1975.

13. Machover, C., et al., "Graphics Displays: Factors in System cesign," IEEE Spectrum, Oct. 1977.

14. Introduction to RSX-11M, DEC, Maynard, March.

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

16. VT-11 Graphic Display Processor, DEC, Maynard, March.

17. Runtz, K.J., et al., "Digital Control Scheme for a Generating Unit," IEEE Trans. Power Apparatus and Systems, Vol. PAS-92, No. 2, Pages 478-483, March/April, 1973.

BIBLIOGRAPHY

1. Davis, L.R. and Fagan, J.E., "Digital Control of Power System Simulator," Pages 48-52, Control of Power Systems Conference, Record No. 78CH1282-3, Reg. 5.

2. Russel, B.D., "A Micro-computer-based Substation Control System," Pages 141-145, Control of Power Systems Conference, Record No. 76, CH1057-9, Reg. 5.

3. "Central Computer Control and Protection Functions," IEEE Computer Relaying Subcommittee, 1976.

4. Jones, J.L., "The Application of Microprocessors to Power System Data Acquisition and Control," Control of Power Systems Conference, Pages 149-153, Record No. 77CH1168-4, Reg. 5.

5. Farag, A.S.T., "Real-Time Digital Control Scheme," Page 87-93, Control of Power Systems Conference, Record No. 76CH1057-9, Reg. 5.

6. Harley, R.G., et. al., "Real-time Computer Control of a Model Turbo Alternator," Pages 5-9, Control of Power Systems Conference, Record, No. 77CH1168-4, Reg. 5.

7. Bjork, D.R., et. al., "Digital Synchronizer for Computer Controlled Generating Stations," Pages 1-4, Control of Power Systems Conference, Record No. 77CH1168-4, Reg. 5.

8. Abegglen, P.C., et. al., "Design of a Real-time Control Data Acquisition and Analysis System," Pages 38-48, Proc. IEEE, Vol. 58, January 1970.

9. Shapiro, M. and Schultz, A., "A Laboratory-Based Computer System," Pages 398-405, Anal. Chem., Vol. 43, March 1971.

10. Enslow, Jr., P.H., "What is a 'Distributed' Data Processing System?" Pages 13-21, Computer, January 1978.

11. Machover, C., et.al., "Computer Graphics Displays," IEEE Spectrum, August 1977.

12. Walker, B.S., et.al., "Interactive Computer Graphics,"

New York: Sane, Russack & Company, Inc., 1975.

13. Machover, C., et.al., "Graphics Displays: Factors in System Design," IEEE Spectrum, October 1977.

14. Introduction to RSX-11M, DEC, Maynard; March, 1975.

15. RSX-11M Task Builder Reference Manual, DEC, Maynard, Mar.75.

16. VT-11 Graphic Display Processor, DEC, Maynard, March. 1975.

17. Runtz, K.J., et.al., "Digital Control Scheme for a Generating Unit," IEEE Trans. Power Apparatus and Systems, Vol. PAS-92, No. 2, March/April, 1973, Pages, 478-483.

18. User's Guide for the RTI-1200, Analog Devices, P.O. Box 280, Norwood, Mass 02062.

19. SBC80P Prototyping Package User's Guide, Intel Corporation, Santa Clara, CA 95051.

20. Brazzell, F.S., Balu, N.J., "An Algorithm for Dispatching Generation in Contingency Load Flows used in Power System Planning Studies", PICA Conference Toronto, Canada 24-27, May 1977.

21. Bollinger, K.E., Gilchrist, R. and Norum, E., "A Digital Controller for Testing Control Strategy at power Plants", Power Ind. Computer Applications, Cleveland, Ohio, 15-18, May 1979.

22. Ciesielski, E.E., Friend, D.G., and Rooney, K.J., "Experience in Maintaining and Using a Large On-Line Power Flow Model", Proc. of Amer. Power Conf., Vol. 39, Chicago, IL, 18-20 April 1977.

23. Gorski, Y.M., Ushakov, V.A., Hope, G.S., and Malik, O.P., "Digital and Adaptive Control of Synchronous Machine Excitation and Speed" Communications and Power Conference, Montreal, Canada, 18-20 October 1978.

24. Ryusawa, S., Tohru, K., and Toda, T., "Multi-Microprocessor Applied SCADA System for Electric Power System", 5th Annual Conference Ind. and Cont. Applic. of Microprocessors, Philadelphia, PA, 19-21 March 1979.

25. Lowe, B., Selmayr, F., and Crewe, A., "Software Techniques for HV Network Control in Bavaria", Proceedings of Second International Conference on Centralized Controlled Systems, London, England, 20-23 March 1978.

26. Kowarl, K., "Hierarchial Structure of the Supervisory and Control Systems of EDF Power System, IEEE, London, England, pp. 86-89, 1978.

27. Green, D.G., and Perry, E., "Microprocessor Control of Standardized Power Control System", Southeastern, 1978 Region 3, Atlanta, GA, 10-12 April 1978.

28. Shimor, A., and Wallach, Y., "A Multions Oriented Parallel Processor System", IEEE Trans. Elec. and Control Instr., pp. 137-141, May 1978.

29. Keyhain, A., and Moayer, B., "Simulation Study in Computerized Load Management System for Scheduled Load Shedding Operation by Ripple Control System", IEEE, 1978 Region V Annual Conference, pp. 228-232, 1978.

30. Eklund, S., and Hermansson, U., "The Supervision and Control of Electric Power System"., ASEA Journal (Swedan), (ASEJAS), Vol. 51, No. 3, pp. 59-64, 1978.

31. Castro, C.H., and Soltovia, P.R., IEEE 1024 pp., pp. 205-7/ 1-7, 1978, 4 Ref.

32. Marks, G.E., "A Method of Comburing High-Speed Contingency Load Flow Analysis with Stochastic Probability Methods to Calculate a Quantitative Measure of Overall Power System Reliability", IEEE PES, Winter Meeting 29 January-3 February 1978.

33. Martin, D.G., Wilkinson, A.J., and Bealtie, W.C., Power System Monitoring", Microprocessor (GB), Vol. 1, No. 8, pp. 463-465, December 1977.

34. Deliyannides,J.S.,See,R.E., "The Use of Microprocessor in Distributed Processing for Power System Control Applications." , Automatic Control Conference, San Francisco, CA, 22-24 Jun 1977

35. Pullman,R.T.,Hogg,B.W.,"Automated Measurements for Computer Control of a Micro-Machine System.", European Conference on PreciseElectrical Measurement, Brighton, Sussex, England, 5-9 Sept. 1977.

36. Scriber, M.V., "Implementation of Computer-Driven Supervisory Control and DAS." Control of Power System Conference, Oklahoma City, OK, 1-3 March 1978.