To the Graduate Council:

I am submitting herewith a thesis written by Zeni Batte entitled "Automatic Data Acquisition System for an LED Test." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Science.

T. Feagin

Terry Feagin, Major Professor

We have read this thesis and recommend its acceptance:

R. A. B[illegible]

F. W. Donaldson

Accepted for the Council:

L. Evans Roth

Vice Chancellor
Graduate Studies and Research

AUTOMATIC DATA ACQUISITION SYSTEM
FOR AN LED TEST

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Zeni Batte

March 1978

ACKNOWLEDGMENTS

The author wishes to express her appreciation to those who were of great assistance in the preparation of this thesis. A special debt of gratitude is owed to Mr. J. B. Puckett and Mr. B. W. Bomar for their guidance and patience in the development of the software. For his valuable support, encouragement and advice during the preparation of the thesis, the author extends many thanks to Mr. A. E. Lennert, Branch Manager of Advanced Concepts. The author thanks the members of her thesis committee for their helpful suggestions for the improvements of this thesis. The author also wishes to acknowledge Mr. J. M. Mann for the final draft of the drawings for the text.

The work reported herein was supported by Arnold Engineering Development Center, Air Force Systems Command, Arnold Air Force Station, Tennessee, and was sponsored by the Naval Ocean Systems Center under Government Contract Number MIPR N0095376MP3011, ARO, Inc., Project Number B34I-03A, Program Element No. 65807F.

ABSTRACT

The object of the work performed was to develop a computer controlled data acquisition and data analysis capability for an uninterrupted 6,000 hour test to monitor the degradation of approximately 250 infrared light emitting diodes (LED's). The test program comprised subjecting a sampling of 254 diodes to preselected constant currents and temperatures while periodically measuring the voltage drop and the light output of each of the diodes. To perform the data acquisition portion of this effort, an automated data acquisition system was designed around a Hewlett-Packard 2100A computer. This system utilized a 1,000 point random access multiplexer, a 16-bit Relay Output Register and a digital voltmeter for data acquisition and transmission. Special purpose assembly language input/output routines were written for the computer's BASIC Interpreter to make this special equipment accessible to the computer. The Control program handled data storage to meet the requirements specified for the project.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

The objective of the LED test program was to monitor the degradation of infrared light emitting diodes (LED'S) for an uninterrupted operating period of 6,000 hours. To obtain the objective a sampling of 254 diodes were subjected to predetermined constant current and constant temperature conditions while periodically verifying the current and heat sink temperatures and measuring the voltage drop and the light output of each of the LED's. An automatic data acquisition system was needed to obtain and process the resulting large volume of data for the extended length of time involved. The following material describes the design and performance of the system.

CHAPTER I

OVERVIEW OF LED TEST COMPLEX

The test program was designed to monitor 48 each of 5 different types of commercial LED's. The diodes were equally divided among 4 environmental chambers maintained at the following nominal operating temperatures: (a) -65^{o}C, (b) 20^{o}C, and (d) 120^{o}C. Twelve of each of the five types of diodes were contained within one chamber. The LED's were mounted in these chambers on specially designed heat sink racks. The heat sink and electrical connections to each of the diodes in all four chambers were designed to be as nearly identical as possible to minimize any mechanical or electrical bias in the measurements since no direct measurement within the chambers was possible. Each type LED was driven at one of four closely controlled currents by its own current regulator. Three of each type LED in each chamber were operated at the same current level. Precision mercury cells were used as references for the individual current regulators to maintain better than 0.5% current regulation throughout the test.

In order to keep the number of instruments required for acquiring data in the LED test complex to a minimum, a Hewlett-Packard 2100A computer was hard-wired from a 16-bit input/output port through a computer/manual control panel to a random access, reed-relay operated, 1000-point multiplexer (MUX). (Schematic of the test complex is shown in Figure 1.) The MUX was used in selecting the test parameter to be measured. The output of the multiplexer and a measurement-reference

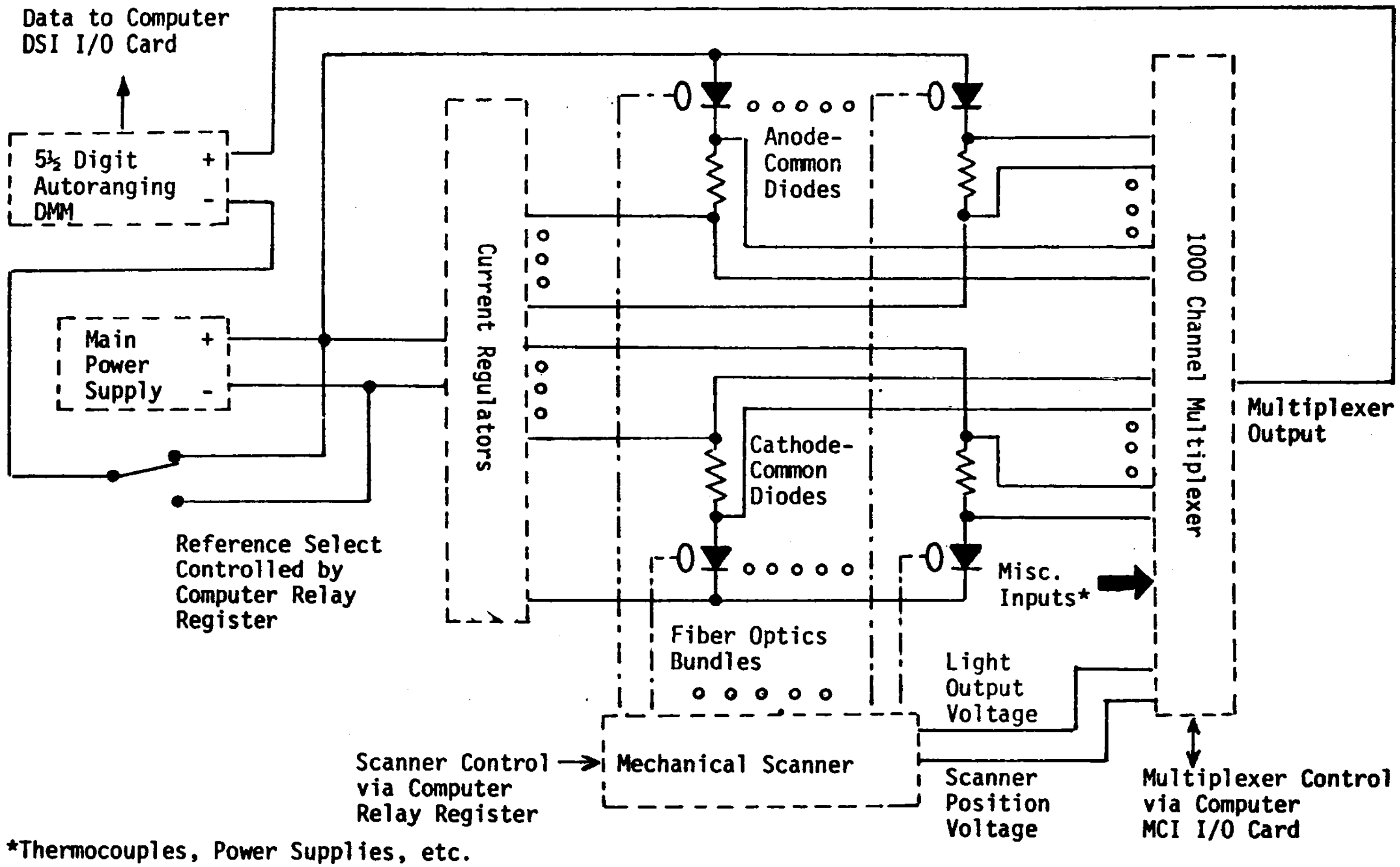


Figure 1. Simplified Schematic of Test Facility.

relay were directly connected to the input of a 5-1/2 digit autoranging Digital Multimeter (DMM). The BCD output of the DMM was hard-wired to the computer through a 32-bit parallel data input port. The light output of each LED was coupled through a fiber optic bundle to a specially designed optical scanner which accurately positioned a calibrated photodetector (PD) in front of each bundle to measure the infrared radiation of each diode. The exact location of the photodetector was monitored by a potentiometer with its output connected to one channel of the multiplexer. The output of the photodetector was amplified by a temperature compensated amplifier connected to another multiplexer channel. A 16-bit Relay Output Register controlled the scanner as well as the measurement-referencing relay.

The entire electrical system was protected against power failure by the automatic starting of a 20KVA diesel auxiliary generator which took approximately 12 seconds to come up to full power. During this transition period, the LED's and their associated control electronics were powered by means of batteries. Once the main power came back on line the generator continued to run for a minimum of 10 minutes to insure a full charge on the 6 volt storage battery used for furnishing transition LED current during power failure, and for charging the diesel start batteries.

An HP2100A computer was used as a controller for automatic data acquisition. A block diagram of the computer system used in the test is shown in Figure 2. This system had a Magnetic Tape Unit, high-speed paper tape reader and punch units, a Tektronix 4010 CRT terminal, and

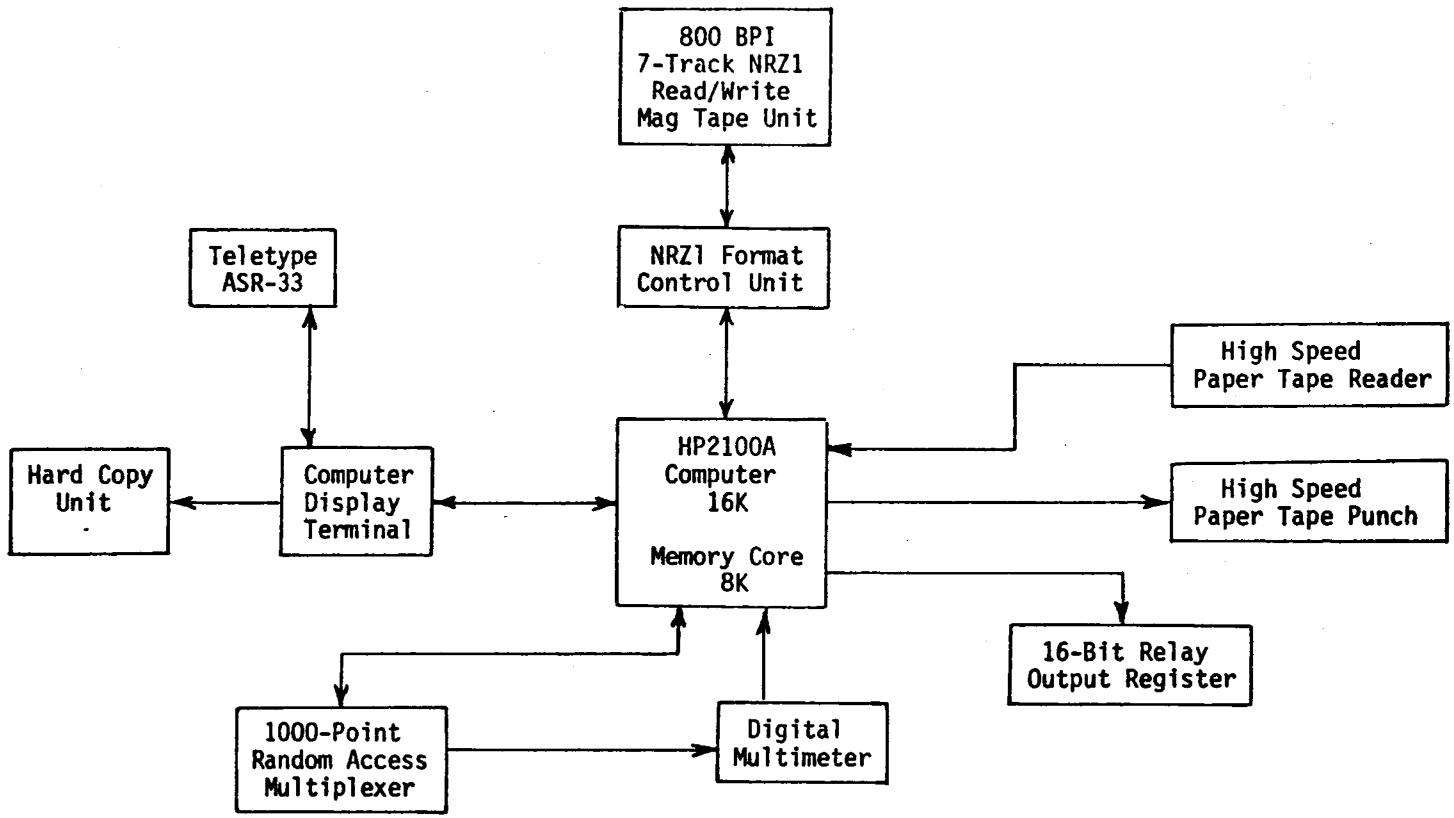


Figure 2. Block Diagram of Computer System.

a teletype. Magnetic tape was used for the main storage of data. It provided fairly rapid access to data for bulk processing. Paper tape was used for the back-up storage of data to safeguard against accidental erasure of the magnetic tape and was used by the client for additional processing of the data for his requirements. Special purpose assembly language Input/Output (I/O) routines were added permitting data to be punched on paper tape in compact binary format. The CRT terminal was used mainly for program development. The teletype was used during the computer runs to relieve the operator from being present at the terminal during the entire run of the program. The teletype printer provided a hard-copy of each run's printout.

CHAPTER II

DESIGN OF COMPUTER SYSTEM

1. Choice of Language for Controlling Program

Software for the HP2100A computer included two high level programming languages: HP FORTRAN and HP BASIC. Relocatable FORTRAN programs were produced using the HP FORTRAN Compiler under the BASIC Control System, whereas all BASIC programming was done without compilation under the BASIC Interpreter. A compiled program was translated into assembly language and then was assembled into machine language. Since this particular system had a two-pass FORTRAN Compiler, the intermediate results were output to paper tape and then reloaded. The BASIC Interpreter interpreted and executed each statement without translation. As a general rule a compiled program can be created to operate within less memory and the final version can be executed more quickly. However, an extra 8K of core memory had been previously added to the system so that the amount of memory was not a critical factor in choosing a programming language. Speed, likewise, was not a critical factor because the devices added to the system to acquire data were relatively slow. For example, the DMM through which all data passed required a quarter of a second to obtain a reading. The advantage of using the BASIC Interpreter was its ease for developing programs and altering these programs as the need arose. Using a compiler, the entire compilation procedure has to be repeated to incorporate changes or corrections. Thus, for this particular system, BASIC was decided to be the better choice of high level language for the controlling program.

2. Framework of Special Purpose Assembly Language I/O Routines

In using BASIC, special purpose subroutines had to be added to the BASIC Interpreter to handle the special devices required for gathering data. Each of the four devices added to the test facility had a specialized purpose. The multiplexer was used to select the channel where data was to be taken. The Digital Multimeter was used for all measurements. The Relay Output Register controlled the referencing of the DMM and the stepping of the Optical Scanner. The Optical Scanner positioned the photodetector which translated light output into a measurable voltage. Of these devices, only the Optical Scanner had no direct connection with the minicomputer and thus did not need a special routine to handle it.

Each special purpose subroutine was designed to handle a particular function of one of the devices. One subroutine input a reading from the DMM to be returned to the controlling program. Three subroutines were used for the Relay Output Register. One output a number between 0 and 65,535 to the Relay Output Register. Another changed the state of a single bit of the Register. The third input the current state of the Relay Output Register. This last routine was used mainly for checking the first two. Finally, two other subroutines were developed to handle the multiplexer. One subroutine output a channel number to the multiplexer while the other input the present channel setting. A complete listing of the subroutines appears in Appendix A.

The controlling program transferred control to an assembly language subroutine with a statement of the form "CALL (subroutine number, parameter list)." The BASIC Interpreter accessed the "called" subroutine through a subroutine table containing linkage information. Entries in the subroutine table, one per subroutine, were two words in length (16 bits per word). Bits 5-0 of the first word contained the number identifying the subroutine. Bits 15-8 contained the number of parameters passed to the subroutine. The second word contained the absolute address of the entry point of the subroutine. All entries in the subroutine table had to be contiguous and, when subroutine entries were added, location 122_8 had to be redefined to contain the address of the last word + 1 of the subroutine linkage table. The subroutines were added below the BASIC Interpreter in memory which is normally free space. To keep this area from being used for other purposes, the address of the last word + 1 of the last subroutine was stored in location 110_8 as the first word of available memory [1][1].

Prior to transferring control to the subroutine, BASIC evaluated the parameters and stacked the address of the results. Upon entering the subroutine, the A-register contained the address of this stack. A subroutine called ".ENTR" had previously been added to the Interpreter to transfer a maximum of four parameter addresses to an

---

[1]Numbers in brackets refer to similarly numbered references in the List of References.

allocated space of memory. By calling ".ENTR" immediately after subroutine entry there was the twofold gain of freeing the A-register and decreasing the depth of indirect addressing. This method was used in all of the subroutines for the special devices.

The special devices are treated individually in the remaining portion of this chapter. This information was required in the development of the assembly language subroutines and the controlling programs.

3. 1000-Point Multiplexer

The random access multiplexer was interfaced to the HP computer via a Microcircuit Interface card (Figures 3 and 4). This interface card provided a 16-bit output register and a 16-bit input register for data transfers. A Device Command signal from the interface card enabled the multiplexer to perform its I/O operation. The interface card accepted a Device Flag signal from the multiplexer indicating a new channel was selected. The "standby" line was pulsed to reset the multiplexer. This line was connected to bit 15 of the output register of the interface card. The channel number (0-999) was output in packed 8421 BCD Format requiring 12 bits. The Control Flip-Flop (FF) was set and the Flag FF was cleared. The reed relay settling time was clocked internally to the multiplexer and a signal was returned which cleared the Control FF and set the Flag FF. After an output operation was completed (Flag FF set), a readback input operation to check the present channel number could follow if required. The input data was also in BCD format [2].

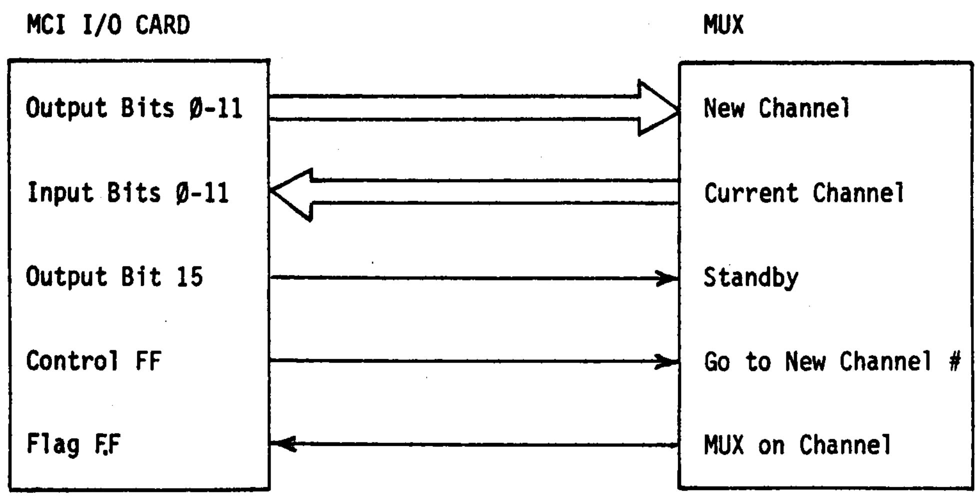


Figure 3. Block Diagram of MUX.

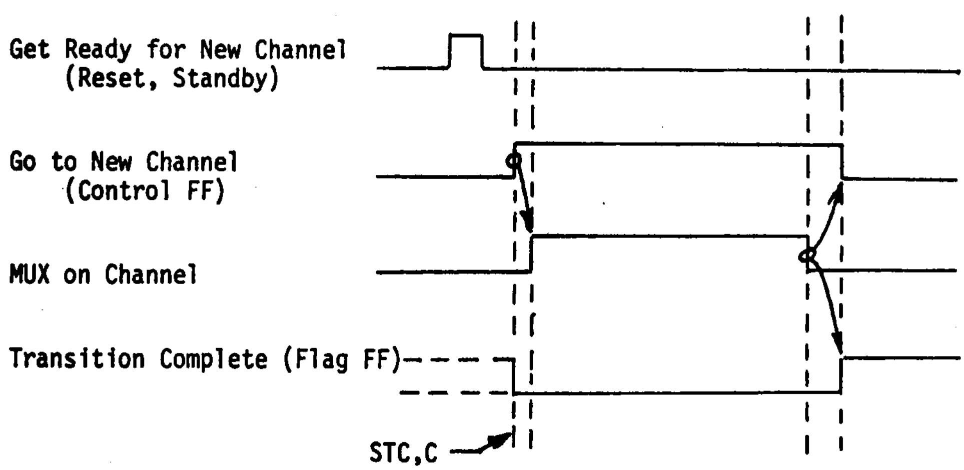


Figure 4. Control Lines for MUX.

The multiplexer was used to specify where a measurement was to be made by the DMM. By outputting a channel number, certain relays were closed to connect an outside point designated by the channel number to one input line of the Digital Multimeter. The other input line of the DMM was connected to +7 volt, or common, reference as determined by the Relay Output Register. The Multiplexer itself could be operated in computer or manual mode.

4. Hewlett-Packard 16-Bit Relay Output Register

The Relay Output Register provided 16 low-current, single-pole, floating contact closures numbered K1 through K16 which could be used in combination or separately to control 1 to 16 devices. The Register had a maximum relay settling time of one millisecond. The 16 relays corresponded to 16 bits of the A- or B-register used for input or output by the computer. Relays were energized by logic "1" bits and de-energized by logic "0" bits output by the computer [3]. The correlation of the bits to the relays and their application in the controlling computer program is shown in Table 1. Bits 0 through 12 were unused.

Relays K16 and K15 were used to control high-current double-pole relays in the LED test complex. By energizing (or de-energizing) the single-pole relays, the circuit between the power supply and the corresponding double pole relays was opened (or closed), thus determining their state. Relay K14 was energized for a certain length of time, controlled by the calling program, in order to activate and send a pulse of current through a high-current single-pole relay to the

TABLE 1

CORRELATION AND FUNCTIONS OF REGISTER BITS

Bit No.	Reference	Purpose	State "0"	State "1"
15	K16	Voltmeter Reference	Common	+7V
14	K15	Direction of Mechanical Scanner Stepping	Forward	Backward
13	K14	Step Scanner	End Pulse	Start Pulse

Optical Scanner stepping motor (forward or reverse motor section determined by K15). This pulse caused the stepping motor to move the scanner one position. Figure 5 is a simplified schematic of the circuitry of the operational relays when in computer mode. Manual operation relays (not shown in Figure 5) parallel the functions of K14, K15, and K16 when the system is in manual mode. When in computer mode, all relays of the LED test complex could be controlled through a single assembly language subroutine.

5. Digital Multimeter

In the LED test complex, the 5-1/2 digit auto-ranging Digital Multimeter was always used as a voltmeter in the auto-ranging mode. The Digital Multimeter was interfaced to the HP2100A through an HP Data Source Interface (DSI) card which transferred up to 32 bits into the HP computer. The data lines from the DMM to the DSI card, as shown in Figure 6, remained unchanged until a request for another reading was made. The Control FF was used to signal a request and the Flag FF was used to signal its completion. Figure 7 shows the states of the control lines during a request for an updated reading. The computer initiated a request by setting the Control FF and clearing the Flag FF which caused the Busy line to be set. The Busy line remained high until a new reading was complete and the data lines had changed accordingly. When the Busy line cleared, the Control FF was cleared and the Flag FF was set signalling to the computer that the updated data was ready to be input. The computer then loaded in 2 16-bit words.

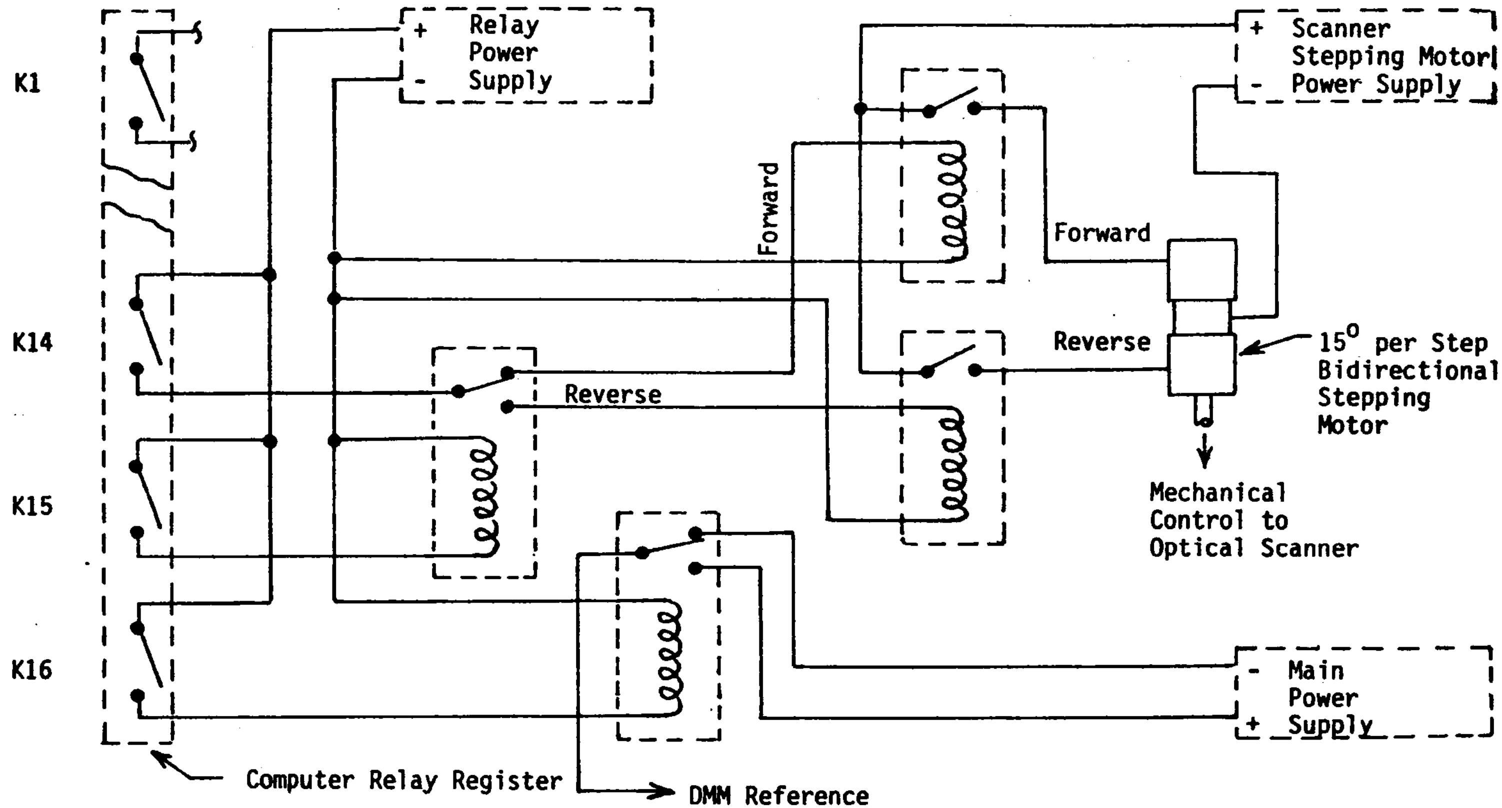


Figure 5. Simplified Schematic of Operational Relays.

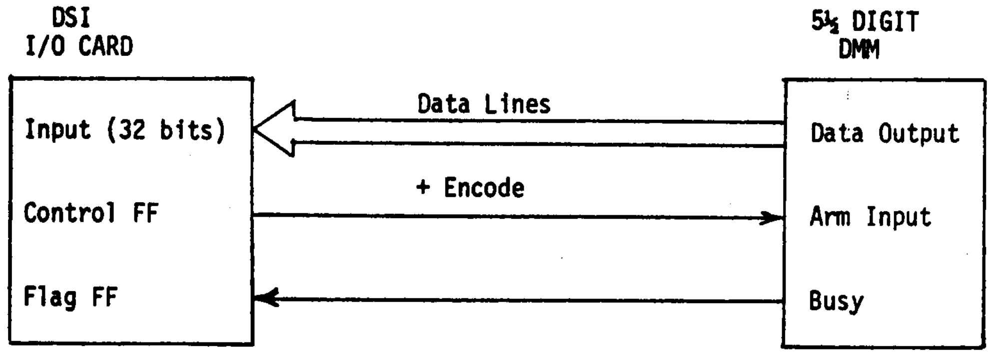


Figure 6. Block Diagram of DMM Interface.

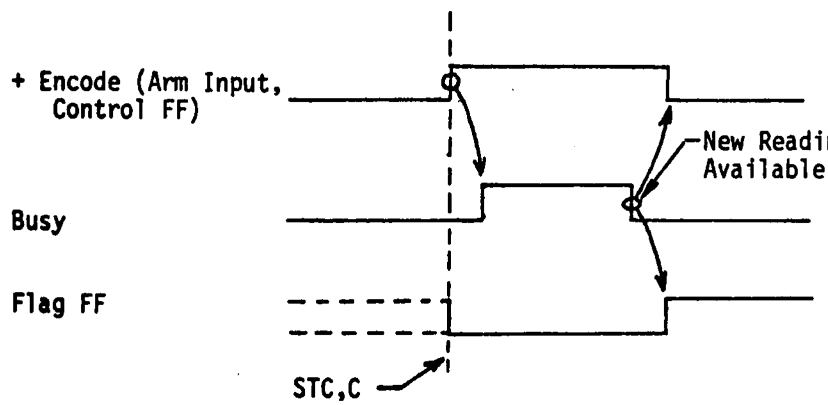


Figure 7. Control Lines for DMM.

The first word transferred contained the least significant bits (0 through 15) and the second word contained the most significant bits (16 through 31) [4].

The DMM presented the magnitude of the measurement in integer BCD format with a sign bit as shown in Figure 8. If the sign bit was logic "1," the reading was negative. A range code was also included in the data to determine the proper placement of the decimal point (see Table 2). Adjustment of the decimal point was done in the assembly language subroutine before returning to the calling program. The overrange bit was set on an overload condition. This was always checked first because if it was logic "1," the range code as well as the sign and magnitude specified a meaningless reading. Software returned a range code of "7" to the controlling program to signal this condition. This occurred whenever a reading required a range larger than the one last used. The DMM automatically stepped to the next range but software had to command that a new reading be taken. To verify that the DMM had completed "auto-ranging," the controlling program required two consecutive readings with the same range code before checking the stability of the readings.

6. Optical Scanner

The Optical Scanner, also called mechanical scanner, operated with a bidirectional stepping motor which positioned a photodetector on the inside of the scanner bonnet to measure light output. Each step moved the photodetector 15^{0} in a spiral fashion. After 24 steps

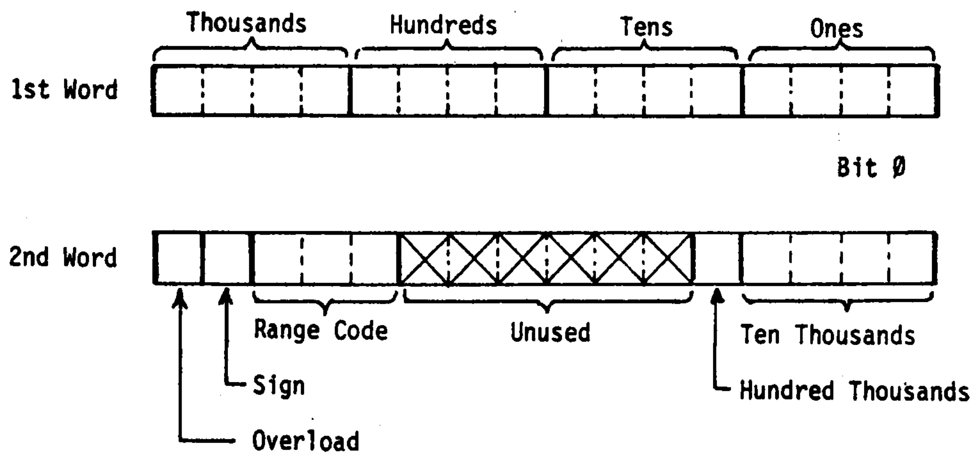


Figure 8. Format of Data from Digital Multimeter.

TABLE 2

RANGE CODES

Range Code	Range	Maximum
1	200MV	.199999V
2	2V	1.99999V
3	20V	19.9999V
4	200V	199.999V
5	1200V	1199.999V

the photodetector had been raised 0.2 inches. The fiber optic bundles were fitted to the fiber optic terminal strips on 0.4 inch centers attached vertically around the scanner bonnet. These terminal strips were aligned in such a way that each level of fiber optic bundles corresponded to one revolution of the photodetector with alternating levels of 24 steps as "blanks" where no measurements were made. The scanner bonnet was sealed so that no outside light could enter and reflect internally to interfere with the measurement by the photodetector. The reading of the photodetector was amplified before being sent through the multiplexer to the digital voltmeter.

The position of the photodetector was measured using a potentiometer. The potentiometer was set so that the position of the first light reading registered zero volts. The span of the potentiometer was set before starting each computer run so that each step of the scanner caused a change in potential of approximately 50 millivolts, normally varying from 45 to 55 millivolts during a single run. This fluctuation created an accumulative error if one tried to define a given position of the scanner as a set voltage. For this reason the diode measurements were taken in the order in which the diodes were fitted around the scanner. After each step only the change of potential was checked to insure that one and only one step had been taken.

CHAPTER III

MEASUREMENT OF DATA

The main purpose of the controlling program was, of course, to automate the acquisition of data; however, for this test the program had a second purpose which is equally important, that is, the systematic storage and retrieval of data. The type of information as well as its source had to be identified and the data treated accordingly. For this purpose, the controlling program was divided into three main sections. The first task of the controlling program, as shown in the generalized flow chart of Figure 9, was to position the Magnetic Tape. The operator was required to specify whether a new reel was to be used and to verify the position of the tape if not. The program then entered the first main section which dealt with the measurement of items that affected all or a large portion of the diodes. Briefly, this section included a preliminary system check and a check on the heat sink temperatures and potentials. Upon completion of this section, the program proceeded to prepare for the measurement of the diodes which was the second main section. The operator was required to enter the time and date at the start of the diode measurements and the reading of the Run Time Meter (RTM). The scanner was positioned to take a background light reading and then was stepped forward to the position of the first diode. All information concerning the individual diodes was gathered and stored in arrays. At the end of this section the operator again entered the RTM reading and the section

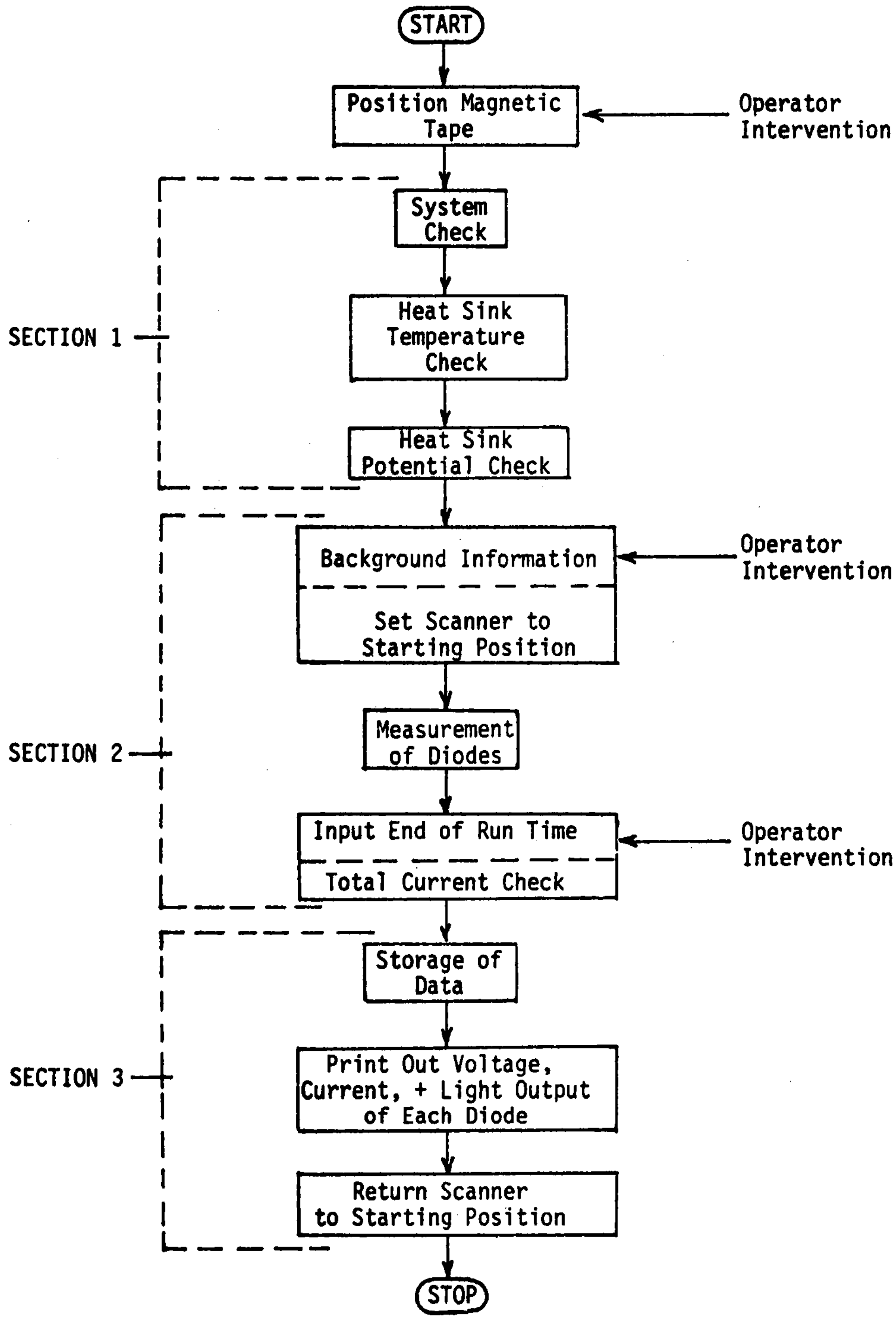


Figure 9. Generalized Flow Chart of Controlling Program.

ends by printing the total time used for the measurement of the diodes, the sum of their currents, and the total current drawn by the system. The third section stored the data on the appropriate devices after all measurements had been completed. All data was first stored on magnetic tape and paper tape, and finally the voltage, current, and light output of each diode was listed on the teletype. The last task of the program was to return the Optical Scanner to its starting position. The three main sections are discussed more fully in the following material.

1. System Check

The first measurements taken by the computer program composed a preliminary system check. Power supplies, back-up power supplies and an auxiliary storage battery were checked. Then the total current drawn by the system, the span setting of the optical scanner and the potential of the optical scanner were checked. A variation of 5% from the expected values or 1 millivolt in the case of the optical scanner potential caused an error message to be printed. The program could then be halted if the situation warranted.

The next step in measurement dealt with the individual heat sinks. The temperature of each heat sink was measured via thermocouples in equivalent volts and then translated into degrees Celsius. Using a $150^{o}F$ thermocouple reference caused an offset of $-2{,}709 \times 10^{-3}$ volts which must be subtracted before a conversion equation based on zero temperature (Celsius) registering zero volts could be applied. The conversion equation used millivolts so that a preliminary conversion of volts to millivolts was necessary. Given V1 as the measured voltage the conversion to degrees Celsius was as follows:

$$V2 = 1000 * (V1 + 2.709 * 10^{-3})$$

$$T = 3.01361 * 10^{-4} * (V2^5) - 5.62523*10^{-3} * (V2^4) + 6.48804*10^{-2} * (V2^3) - 0.805664 * (V2^2) + 25.9697 * (V2) + 2.27051*10^{-2}$$

The latter equation was obtained by a least square curve fit [5] on a millivolt-temperature conversion table. Great accuracy was not needed since the number of operating diodes on a heat sink contributed a varying heat source. The measured temperature was used more as an indicator than as an absolute monitor for adjusting the chamber temperature. The potentials of the heat sinks were then checked. The relay register set the reference according to whether the heat sink was anode or cathode (+7 volt or common) referenced. The purpose of the heat sink potential will be discussed in the next section.

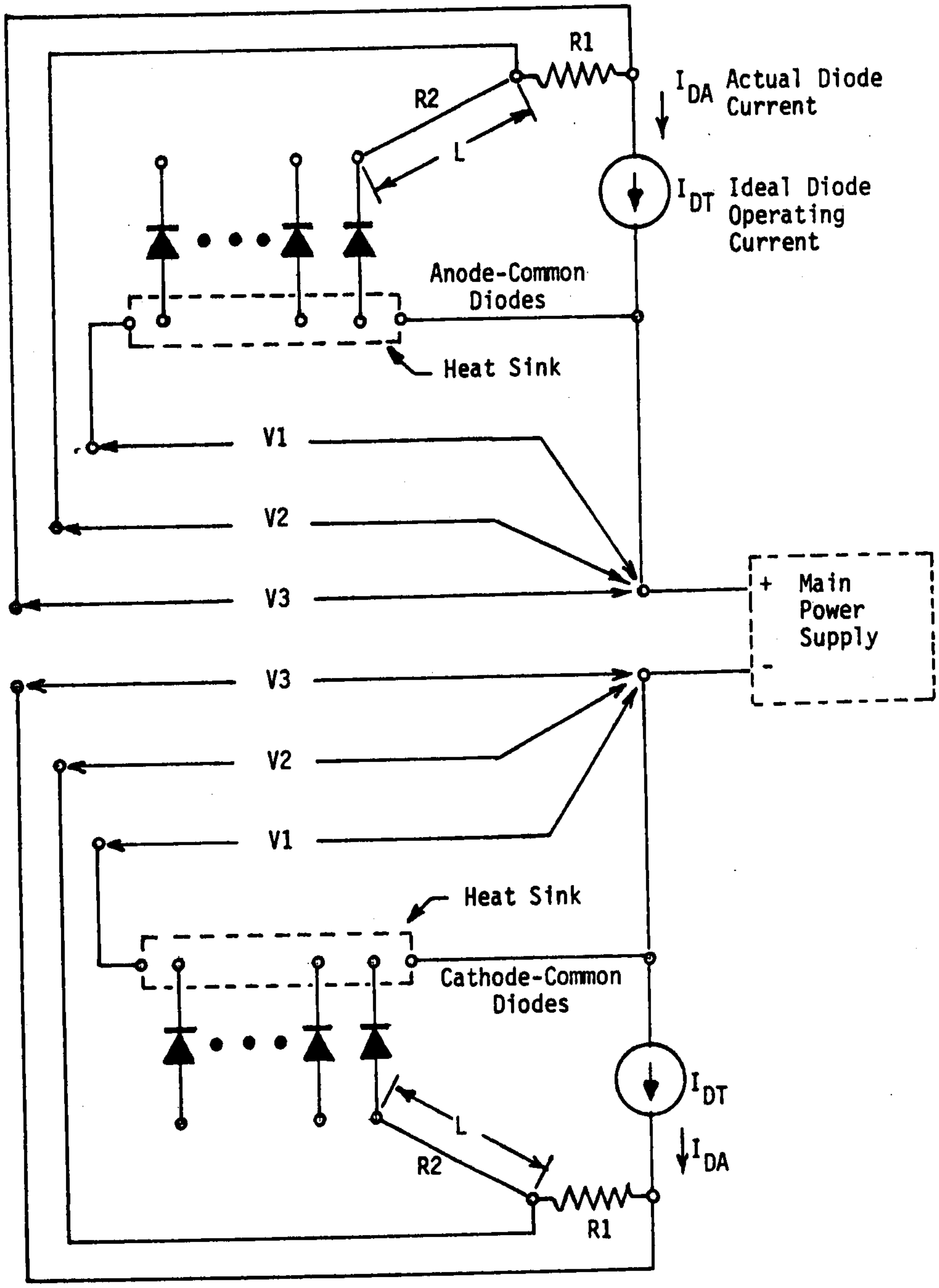
2. Diode Current and Voltage Measurement

Each diode sat in a heat sink and was sealed into one of four environmental chambers. The heat sinks were made of a heavy bar of copper so that all points on the heat sink would have essentially the same potential. Each diode was connected to the outside with equal lengths of 20 gauge stranded wire. The voltage drop due to the wire was used in the final calculation of the voltage measurement of each diode. The current to each diode was measured at the point where it entered the chamber by measuring the voltage drop across a nominal

2 ohm sense resistor in series with the diode. The resistance of each sense resistor was measured to an accuracy of three decimal places prior to the test and stored in the computer program to permit precise current determination.

With this setup, the potential of each heat sink needed only to be measured once and stored in an array indexed to all diodes on that heat sink. These measurements were completed before any measurements for individual diodes were begun. All further measurements for a single diode were made in immediate sequence. This method tied the voltage, current, and light output measurements of each diode closely in time.

Since the setup for each diode was identical, the procedure for taking data and making simple calculations with this data was the same. The potential of the heat sink (V1), as shown in Figure 10, was measured and stored at the start of each run. The potential on the side of the sense resistor nearest the diode (V2) was taken. This potential included the voltage drop due to the connecting wire resistance (R2), the diode, and the heat sink. The potential on the other side of the sense resistor (V3) was then measured. The difference of V3-V2 was the potential drop due only to the resistor. This difference was divided by the resistance of the sense resistor (R1) to give the current passing through this diode. The voltage across the diode could then be calculated by taking V2 and subtracting the potential drop due to the connecting wire ((V3-V2) * (R2/R1) and the



V1, V2, and V3 = Voltages Read Through the Multiplexer

Figure 10. Setup of LED's.

heat sink (V1). The light output was measured, as described in Chapter II, Section 6, and the Optical Scanner was positioned for the next diode.

The only minor complication to the procedure was that all diodes were not common referenced. Thus the relay register had to be set accordingly before proceeding. Since all anode diodes were set in +7 volt referenced heat sinks, the reference for a particular diode could be determined by the heat sink on which it was installed. The heat sinks were numbered in the order in which their voltages were measured. An array whose index corresponded to the diodes in the order they were scanned held the corresponding heat sink number for each diode. This array was used to index the array of heat sink voltages as well as to determine the reference of a particular diode.

To take each voltage measurement, the multiplexer was set to the corresponding channel. As seen before, only two voltage measurements (in addition to the heat sink potential) were needed to determine the voltage and current for each diode. The light outputs of all diodes were measured through the same multiplexer channel with the Optical Scanner providing diode selection. Thus, only the channel numbers for the two voltage measurements were determined for each diode. Although these two channel numbers were consecutive, the first channel number of the pair was not in the same order as the diodes were scanned. For this reason, another array stored the first channel number for each diode to obtain proper ordering.

Overall, for the acquisition of diode data, only three data arrays stored in the program were necessary. In review they contained the heat sink number, the first channel number of the sense resistor, and resistance of the sense resistor for each diode. Another array was added containing the correct current of each diode for comparison with the measured value. As mentioned before each type of diode in each chamber was operated at four different current settings. Each diode was to be maintained within 0.5% of its set-point current which was stored in this array. As each diode was checked, if the current was out of tolerance, the variation was printed on the teletype. In this manner, manual adjustment of the current could take place. Diode currents which had dropped below 50% of the set-point current due to diode failure could not be adjusted back within tolerance. Since the teletype was fairly slow, the messages for these diodes were omitted.

Figure 11 is a flow chart of the section of the controlling program which gathered the data from the individual diodes as explained previously. This section refers to two subroutines, the DMM subroutine and the scanner subroutine, whose flow charts appear in Figures 12 and 13, respectively. Before entering this section of the controlling program, a background reading of light output was taken and the scanner was positioned for the first diode's light output measurement.

As mentioned in the text, the scanner skipped alternate rows of 24 steps. For this purpose, a counter, F3, was used to determine when the end of a row had been reached. R1 was the resistance of the wire

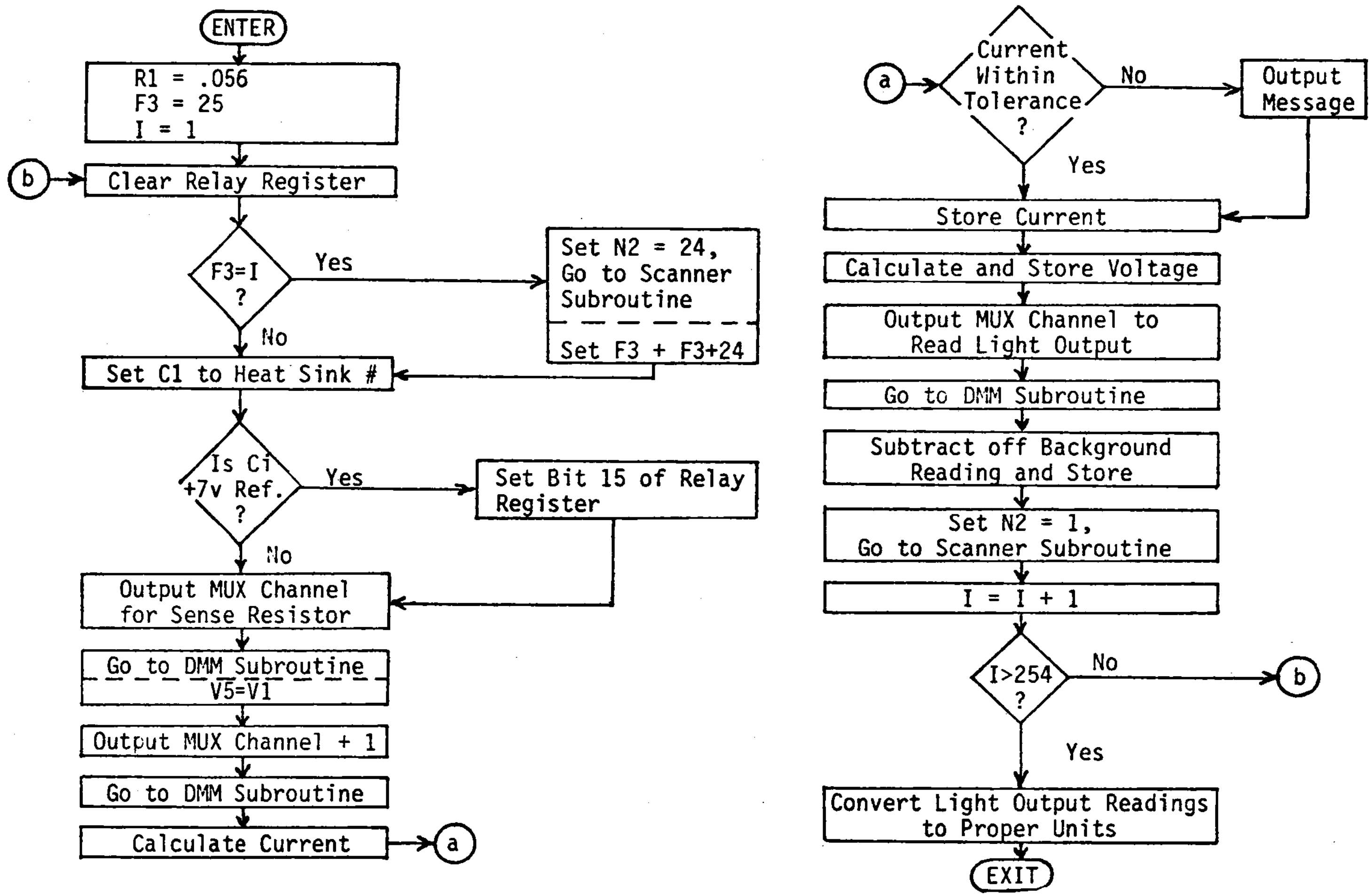


Figure 11. Flow Chart of Diode Measurement Section.

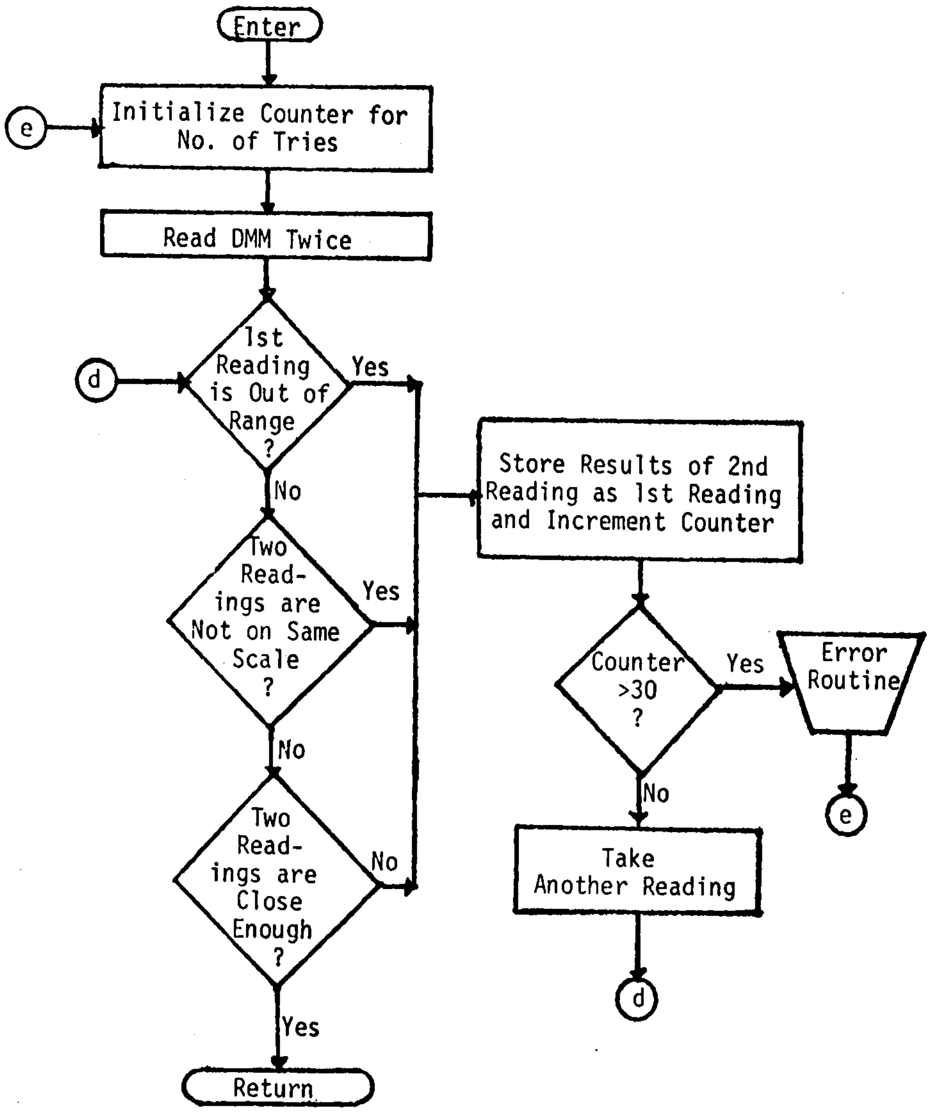


Figure 12. Flow Chart of DMM Subroutine.

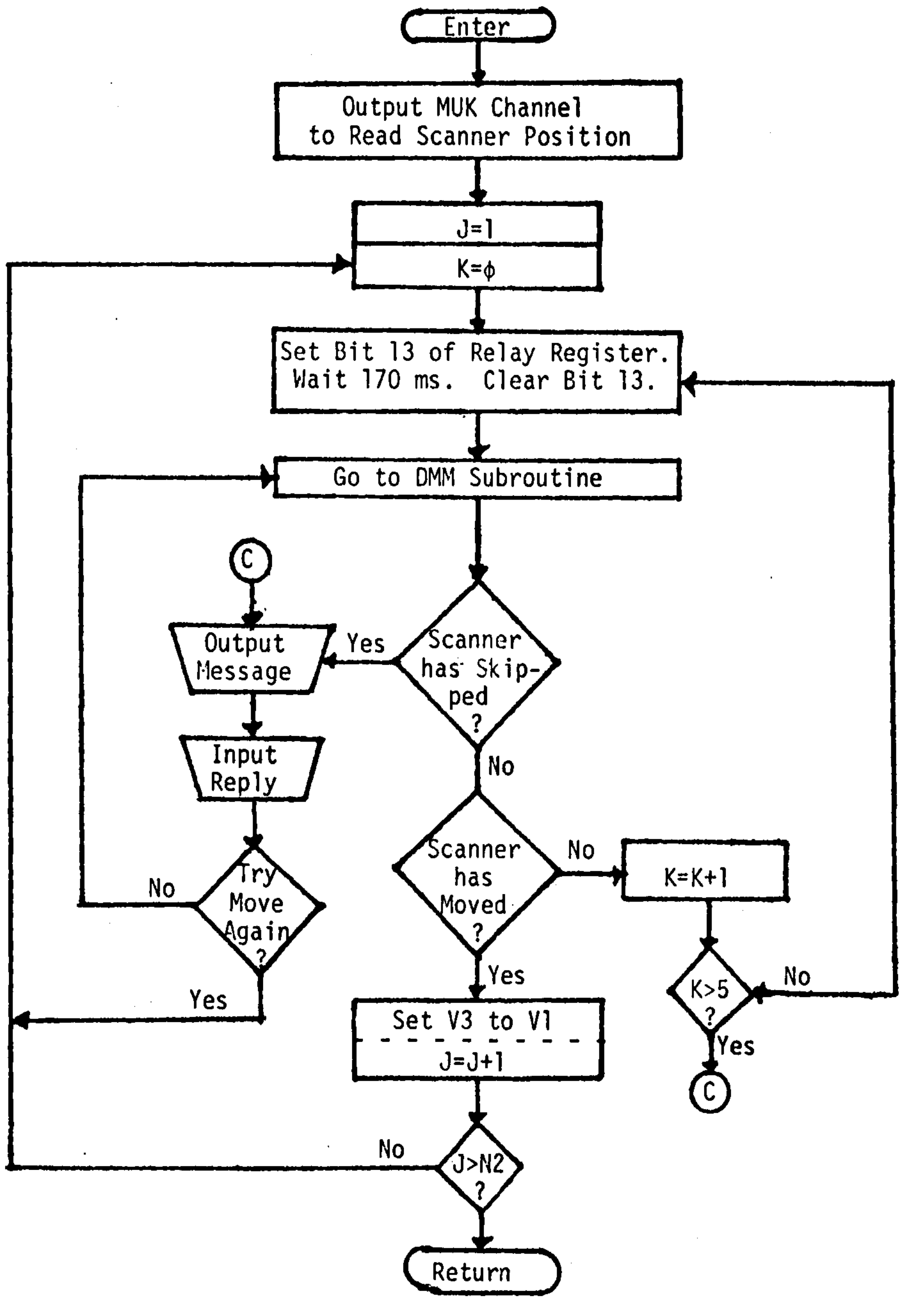


Figure 13. Flow Chart of Scanner Subroutine.

connecting each diode to its respective sense resistor. The variable, I, is a counter for the diodes and was used for indexing the data arrays. By clearing the Relay Register at the beginning of each loop through this section, the DMM was set for negative reference and the scanner was set for forward motion. Alterations to the Relay Register were made one bit at a time when necessary.

The DMM subroutine was used to acquire a stable DMM reading and return the result in V1. Before a measurement was considered acceptable, two consecutive readings were taken on the same scale and had to have a difference of less than 0.1% of the second reading or less than 0.1 millivolt. If neither of these conditions was met, successive measurements were made until one of the conditions was met or until a nominal number of measurements had been made and operator intervention was required.

The scanner subroutine moved the Optical Scanner N2 positions in the direction determined by the main program. Before this subroutine was entered for the first time, the position voltage of the scanner was measured and stored in V3. Therefore, V3 was set to the last measured voltage by this subroutine. V3 was used to determine whether the scanner had moved or had skipped a position. If the scanner failed to move after five attempts or if the scanner skipped a position, an error message was printed noting the type of error, the present voltage, and the last measured voltage of a successful "move" command. With this information and an observation of the equipment, the operator could make the proper adjustments for the program to continue. The scanner could be moved manually to either the position to which it was supposed to be set or to the position at which the last successful "move" was made. The operator then entered a "0" or "1," respectively, indicating which correction had been made and the program continued.

The program variable J was a counter for the number of moves completed and K was a counter for the number of attempts per "move" command. The DMM subroutine returned the position voltage as V1.

3. Storage of Data

All data taken in the computer program was stored on magnetic tape and paper tape. This information was:

1. Time and date at start of run.
2. Results of preliminary system check.
3. Temperature of heat sinks.
4. Voltages of heat sinks.
5. Voltage of each diode.
6. Light output of each diode.
7. Current of each diode.
8. Scanner position voltage for each diode.
9. Fiber optic data.

With the exceptions of 2, 4, and 8, all of this information was printed at some point in the computer run. The printed form provided visual checking as the program ran with a complete printout of voltage, current, and light output for each diode at the end of the run.

The magnetic tape was positioned at the beginning of the run. If the operator designated that a new tape had been loaded, a filemark was placed on the tape and the program proceeded to take data. The data was always written on tape followed by two filemarks. If the program was to find the last file on an existing data tape, it skipped a filemark and read a data point, repeating this process until it encountered a filemark instead of a data point. As a check, the program moved the tape to the last written file, read the first record, and printed the time and date of the last run so that the operator was sure that the last file had been found. The tape was then positioned so that the second of the double filemarks would be written over. Data was written on magnetic tape and paper tape after all measurements were completed. Data was stored in a common block of memory before being stored on tape. If the program halted or an error occurred on both the magnetic tape and paper tape punch unit, data could still be recovered since the location of this memory block was known.

CHAPTER IV

CONCLUSIONS

The LED test successfully completed the 6,000 hour run. The total time required to take each set of measurements on the 254 diodes was 33 minutes. The major portion of this time was consumed in positioning the Optical Scanner. Manual operation of the system provided easy means for checking questionable data and maintaining system reliability. By having all data acquisition under computer control, more information was attained in a given period of time than could ever have been possible by manual means. This was an important asset to this test since LED's exhibited peculiar behavior during the first 200 hours of the test program. Under manual measuring conditions these peculiarities in performance would not have been observed. By having the data stored in a form accessible by the computer, analysis of the data was greatly facilitated.

LIST OF REFERENCES

LIST OF REFERENCES

1. H P BASIC. Third edition. Cupertino, California: Hewlett-Packard, 1970.

2. 500/100 Channel Datareed Scanner: D-A-012501. Rochester, New York: A. D. Systems, Inc.

3. "16-Bit Relay Output Register." Instruction set for 12551B and 12551b-001 Interface Kits. U.S.A.: Hewlett-Packard, 1970.

4. Instruction Manual for Model 8800A Digital Multimeter. Seattle, Washington: John Fluke Mfg. Co., Inc., 1974.

5. FORTRAN IV Language for IBM System 360. Eighth edition. New York, N. Y.: IBM Corp., 1968. Pp. 109-115.

6. Finkel, Jules. Computer/Aided Experimentation: Interfacing to Minicomputers. New York: John Wiley & Sons, Inc., 1975.

APPENDIXES

APPENDIX A

SPECIAL PURPOSE ASSEMBLY LANGUAGE ROUTINES

The following material is a listing of the special purpose assembly language "CALL" routines added to the BASIC Interpreter to handle the Digital Multimeter (voltmeter), the 16-bit Relay Output Register, and the multiplexer. Parameters being sent to the devices were first checked for acceptability and then placed in the format suitable to the device. All parameters transferred from the devices to the calling program were placed in floating-point doubleword format. Each routine has a preface describing the routine's function and its parameters.

```
*
***
*        C A L L ( 6 , V , R )
**
**        SUBROUTINE TO INPUT VOLTMETER READING
**
**        WHERE V IS THE VOLTMETER IN VOLTS WITH SIGN.
**                ( MEANINGLESS IS R = 7 )
**         AND R IS THE RANGE CODE .
**
**             RANGE CODE                    RANGE
**            -------------               ----------
**                  1                        200 MV
**                  2                          2 V
**                  3                         20 V
**                  4                        200 V
**                  5                       1200 V
**                  6                           N/A
**            ------------------------------------------
**                  7                       OVERRANGE
***
*
     14175 000000  CALL6 NOP
     14176 014030        JSB .ENTR
     14177 064432        LDB M2
 PAGE 0021 #01
```

```
14200 076255        STB SIGN        SET SIGN = -2
14201 103711        STC VMIO,C       VOLTMETER I/O
14202 102311        SFS VMIO        DELAY UNTIL
14203 026202        JMP *-1            INPUT COMPLETE
14204 102511        LIA VMIO
14205 014044        JSB CONV        CONVERT BCD TO FLT. PT. BINARY
14206 104400        DST VOLTS
14207 014257

14210 102511        LIA VMIO
14211 002020        SSA
14212 026250        JMP OVRNG       OVERRANGE IF BIT 15 = 1
14213 071734        STA COPY
14214 001200        RAL             CHECK SIGN BIT 14
14215 002021        SSA,RSS         IF SIGN BIT = 0 ,
14216 036255        ISZ SIGN          SET SIGN = -1 .
14217 001700        ALF
14220 010331        AND D7          ISOLATE RANGE BITS
14221 072255        STA RANGE
14222 105120        FLT
14223 104400        DST PARA2,I     STORE FOR RETURN
14224 100374

14225 062255        LDA RANGE
14226 040437        ADA M7          TO BE USED
14227 072255        STA RANGE            AS COUNTER
14230 061734        LDA COPY
14231 010344        AND .31         ISOLATE LAST FIVE BITS
14232 014044        JSB CONV
14233 105040        FMP TNTHD       MULTIPLY BY 10,000 (DEC)
14234 014263

14235 105000        FAD VOLTS
14236 014257

14237 036255        ISZ SIGN        IF SIGN + 1 = 0 , SKIP .
14240 114242        JSB ARINA,I       NEGATE VOLTS
14241 105060        FDV FLT10       DIVIDE BY 10.0 TO
14242 014261
```

```
1006   14243 036253         ISZ RANGE          ADJUST SCALE .
1007   14244 026241         JMP *-3
1008   14245 104400         DST PARA1,I
       14246 100373
1009   14247 126175         JMP CALL6,I
1010   14250 060331   OVRNG LDA D7
1011   14251 105120         FLT
1012   14252 104400         DST PARA2,I
       14253 100374
1013   14254 126175         JMP CALL6,I
1014*
1015   03437          M7    EQU 437B         MINUS 7
1016   03344          .31   EQU 344B         DECIMAL 31
1017   01734          COPY  EQU 1734B        TEMP STORAGE
1018   14255 000000   RANGE BSS 1
1019   14256 000000   SIGN  BSS 1
1020   14257 000000   VOLTS BSS 2
1021   14261 050000   FLT10 DEC 10.
       14262 000010
1022   14263 047040   TNTHD DEC 10000.
       14264 000034
1023   00311          VMIO  EQU 11B          VOLTMETER SELECT CODE
 PAGE 0022 #01

1056*
1057***
1058**                CALL(8,N)
1059**
1060**         ROUTINE FOR RELAY OUTPUT REGISTER
1061**         CAN HANDLE INTEGER FROM ZERO TO 65,535 (DEC.)
1062**         USES REG. A OR B FOR INPUT AND OUTPUT
1063**
1064**         N DESIGNATES DESIRED RELAY OUTPUT
1065***
1066*
```

```
1067    14312 000000   CALL8 NOP
1068    14313 014030         JSB .ENTR
1069    14314 104200         DLD PARA1,I
        14315 100373
1070    14316 002020         SSA                IF PARA1 IS < 0 ,
1071    14317 125752         JMP ERR1,I                 OUT OF RANGE .
1072    14320 105020         FSB MAXLM
        14321 014346
1073    14322 002021         SSA,RSS            IF PARA1 IS > 65,535 ,
1074    14323 125752         JMP ERR1,I                 OUT OF RANGE .
1075    14324 105000         FAD LIMIT
        14325 014350
 PAGE 0023 #01

1076    14326 002020         SSA                IF PARA1 IS < 32,768 ,
1077    14327 026333         JMP AGAIN                  BRING BACK ORIG. QUANTI
1078    14330 105100         FIX
1079    14331 030470         IOR BIT15
1080    14332 026336         JMP OUTIT
1081    14333 105000   AGAIN FAD LIMIT
        14334 014350
1082    14335 105100         FIX
1083    14336 016340   OUTIT JSB RLOUT          RELAY OUTPUT ROUTINE
1084    14337 126312         JMP CALL8,I
1085*
1086    14340 000000   RLOUT NOP                RELAY OUTPUT ROUTINE ENTRY
1087    14341 102621         OTA RELAY
1088    14342 064460         LDB M256            MINUS 256 (DEC)
1089    14343 034001         ISZ 1               DELAY 1.26 MILLISECONDS
1090    14344 026343         JMP *-1                FOR RELAY TO BE COMPLETE
1091    14345 126340         JMP RLOUT,I
1092*
1093    00021          RELAY EQU 21B            RELAY REGISTER SELECT CODE
1094    14346 040000   MAXLM DEC 65536.         = 2 ^ (16)
```

```
      14347 000042
      14350 040000  LIMIT DEC 32768.     = 2 ^ (15)
      14351 000040
*
***
**                C A L L ( 9 , B , V )
**
**          SUBROUTINE TO OPERATE ANY RELAY INDIVIDUALLY
**           WITHOUT ALTERING ANY OTHER RELAY .
**
**          SETS BIT B RELAY TO STATE V ON RELAY OUTPUT CARD.
**
**          IF V = 0 , BIT IS SET TO LOGIC 0 .
**          OTHERWISE , BIT B IS SET TO LOGIC 1 .
**
**          B MUST BE BETWEEN 0 AND 15 INCLUSIVE .
***
*
      14352 000000  CALL9 NOP
      14353 014030        JSB .ENTR
      14354 104200        DLD PARA1,I
      14355 100373
      14356 002020        SSA                 IF B < 0 ,
      14357 125752        JMP ERR1,I                   OUT OF RANGE .
      14360 105100        FIX
      14361 072036        STA TEMP
      14362 040445        ADA M16
      14363 002021        SSA,RSS             IF B > 15 ,
      14364 125752        JMP ERR1,I                   OUT OF RANGE
      14365 062036        LDA TEMP
      14366 003004        CMA,INA             USE AS COUNTER FOR ROTATION
      14367 006404        CLB,INB             SET REG. B = 1 .
      14370 002024        SSA,INA
      14371 005200        RBL
      14372 002024        SSA,INA
```

```
1127  14373 026371          JMP *-2
1128  14374 076036          STB TEMP
 PAGE 0024 #01

1129  14375 104200          DLD PARA2,I
      14376 100374
1130  14377 002020          SSA                    IF V < 0 ,
1131  14400 114242          JSB ARINA,I                 NEGATE .
1132  14401 105100          FIX
1133  14402 066036          LDB TEMP
1134  14403 002002          SZA
1135  14404 026411          JMP OR
1136  14405 037000          CMB
1137  14406 102521          LIA RELAY
1138  14407 010031          AND B
1139  14410 026413          JMP ROUT
1140  14411 102521    OR    LIA RELAY
1141  14412 030031          IOR B
1142  14413 016340    ROUT  JSB RLOUT          JUMP TO RELAY OUTPUT ROUTINE
1143  14414 126352          JMP CALL9,I
1144*
1145  00445           M16   EQU 445B
1146*
1147***
1148**               C A L L ( 10 , N )
1149**
1150**        RETURNS THE CURRENT STATE OF THE RELAY
1151**          OUTPUT CARD IN N .
1152***
1153*
1154  14415 000000    SB10  NOP
1155  14416 014030          JSB .ENTR
1156  14417 005400          CLB
1157  14420 102521          LIA RELAY
```

```
1158  14421 002020             SSA
1159  14422 026427             JMP MASK
1160  14423 105120             FLT
1161  14424 104400             DST PARA1,I
      14425 100073
1162  14426 126415             JMP S310,I
1163  14427 010422  MASK       AND INF         INF = 77777B
1164  14430 105120             FLT
1165  14431 105000             FAD LIMIT
      14432 014350
1166  14433 104400             DST PARA1,I
      14434 100073
1167  14435 126415             JMP S310,I
1168*
1169***
1170*
1171*                MULTIPLEXER  ROUTINES                      *
1172*
1173*      CALL(11,C)  SETS MULTIPLEXER TO CHANNEL C            *
1174*
1175*      CALL(12,C)  RETURNS CURRENT CHANNEL SETTING          *
1176*
1177*
1178  14436 000000  MPXR1      NOP
1179  14437 014030             JSB 303         FETCH CHANNEL NO.
1180  14440 060470             LDA MNEG        SET SIGN BIT
1181  14441 102616             OTA MPXIO         PULSE THE "STANDBY"
 PAGE 0025 #01

1182  14442 060445             LDA DELAY           LINE TO
1183  14443 002006             INA,SZA               RESET THE
1184  14444 026443             JMP *-1                 MULTIPLEXER .
1185  14445 102616             OTA MPXIO       OUTPUT ZERO
1186  14446 104200             DLD PARA1,I
```

```
     14447 100373
1187 14450 015753        JSB 1753B      BINARY(FLT) TO BCD
1188 14451 106516        OTB MPXIO
1189 14452 103716        STC MPXIO,C
1190 14453 102316        SFS MPXIO
1191 14454 026453        JMP *-1
1192 14455 126435        JMP MPXR1,I
1193*
1194 14456 000000  MPXR2 NOP
1195 14457 014030        JSB 30B
1196 14460 102516        LIA MPXIO
1197 14461 014044        JSB 44B        BCD TO BINARY (FLT)
1198 14462 104400        DST 733,I
     14463 100373
1199 14464 126456        JMP MPXR2,I
1200*
1201 00016        MPXIO EQU 16B        MULTIPLEXER SELECT CODE
1202 00470        MNEG  EQU 470B       OCT 100000
1203 00445        DELAY EQU 445B       DEC -16
1204*
1205***
```

APPENDIX B

SAMPLE COMPUTER RUN

The following material is a partial printout from a typical run. This example shows the normal interaction between the operator and the computer. The prompts and error messages shown in this sample typify the completeness of the information which was given to the operator during a run.

```
 * * * *  - - -     RUN # 95     - - -  * * * *
- - - - - - - - - - - - - - - - - - - - - - - - - - -

IF STORING DATA ON NEW REEL OF MAG TAPE, INPUT 1.
OTHERWISE INPUT 0 (ZERO) .
?0

LAST DATE WAS  9    : 18      6    / 20    / 1977
INPUT 0 IF DATE IS CORRECT AND PROGRAM IS TO CONTINUE.
?0

PUNCH FEED FRAMES ON PUNCH UNIT AND CHECK AMOUNT OF TAPE .

VOLTAGE READ ON CHANNEL 806  IS -.25864     VOLTS.

ACTUAL VALUE IS OFF BY  4.13600E-02   VOLTS .

VOLTAGE READ ON CHANNEL 807  IS -.25898     VOLTS.

ACTUAL VALUE IS OFF BY  4.10200E-02   VOLTS .

HEAT SINK #     TEMP. (DEG. C)

    A 1         -56.3317

    A 2         -57.6339

    A 3         -61.0806
```

A 4	-58.2414
A 5	-58.211
B 1	35.8959
B 2	34.9593
B 3	32.5252
B 4	34.2375
B 5	35.3678
C 1	99.6863
C 2	99.5793
C 3	97.3929
C 4	98.7013
C 5	98.0582
D 1	124.398
D 2	124.026
D 3	122.515
D 4	124.315
D 5	124.253
A 5X	-56.7854

```
D 1%          123.798

ENTER HOUR, MINUTES, DAY, MONTH, AND YEAR IN THAT ORDER.
SEPARATE EACH WITH A COMMA.  (ALL ENTRIES NUMERICAL)

?12,48,24,6,1977

 12   : 48      6    / 24    / 1977

ENTER RTM FOR START OF DATA RUN.
?2927.9

TOTAL TEST TIME AT START OF DATA RUN IS 4461.8     HRS.

INITIAL PHOTOCURRENT OFFSET IS 1.13087E-03

INPUT J16 READING
?.095

RATIO OF J16 TO F.O.#1 IS 7.71763E+06

CURRENT DIFFERS FROM SET-POINT ON LED'S AS FOLLOWS:

NUMBER      CURRENT DEVIATION   %ERROR

 303             1.30087          .520349

 356             2.51445          2.01156

TOTAL TEST TIME AT END OF DATA RUN IS 4462.33    HRS.
```

TOTAL RUN TIME IS .533203   HRS.

SUM OF DIODE CURRENTS IS 33.3406   AMPS. THE CURRENT CALCULATED USING THE SHUNT RESISTOR WAS 34.28   AMPS AT THE BEGINNING OF THE RUN AND 34.28   AMPS AT THE END OF THE RUN.

DIFFERENCE BETWEEN TWO MEASUREMENTS FOR SHUNT RESISTOR IS 0   AMPS.

FIRST READING MINUS SUM OF CURRENTS IS .93943   AMPS.

SECOND READING MINUS SUM OF CURRENTS IS .93943   AMPS.

NUMBER	CURRENT	VOLTAGE	LIGHT OUTPUT
1	99.955	1.85554	1.23575E-06
2	99.795	1.81825	1.29049E-06
3	99.9101	1.81154	9.82556E-07
4	199.94	1.91064	1.53988E-06
5	199.6	1.89396	1.68625E-06
6	199.87	1.91585	1.52748E-06
101	99.9095	1.496	4.54350E-07
102	100.04	1.46119	4.53949E-07
103	99.8749	1.4816	5.26555E-07
104	199.519	1.60812	9.24211E-07
105	199.7	1.54451	9.82356E-07
106	200.38	1.57137	8.05328E-07
201	99.97	1.48974	2.82716E-07
202	100.055	1.45815	2.57498E-07
203	100.12	1.4118	3.37560E-07
204	200.28	1.76222	3.63079E-07
205	200.19	1.49693	5.21501E-07
206	5.05445	5.55266	-1.70131E-09

VITA

The author was born in Memphis, Tennessee, on December 5, 1953. She attended Nicholas Blackwell High School in Bartlett, Tennessee, and graduated in 1971 with honors. In 1975 she graduated Summa Cum Laude from Christian Brothers College at Memphis, Tennessee, and received a Bachelor of Science degree in Mathematics. For the past two years, she has been attending The University of Tennessee as a research assistant stationed at the Arnold Engineering Development Center in order to attain a Master of Science degree in Computer Science.