A MICROPROCESSOR BASED CONTROL SYSTEM FOR A SOLAR HEATED GREENHOUSE

By

SIMON TIMOTHY MURRAY

Bachelor of Engineering (Agricultural)

Darling Downs Institute of

Advanced Education

Toowoomba, Queensland

Australia

1980

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
Master of Science
July, 1982

A MICROPROCESSOR BASED CONTROL SYSTEM

FOR A SOLAR HEATED GREENHOUSE

Thesis Approved:

Thesis Adviser

Dean of the Graduate College

PREFACE

The purpose of this study was to design and build a control system for a solar heated greenhouse. Funding of this study was from the Oklahoma Experiment Station Project 1690 "Solar Heating and Energy Conservation/Greenhouses".

Gratitude is extended to the Agricultural Engineering Department for their financial assistantship throughout the term of study.

A special thanks is extended to the major advisor Dr. George W. A. Mahoney for his guidance and understanding throughout the study period. Appreciation is extended to Dr. Peter Bloome and Dr. Dave Seldom for their assistance in time of need. To Bruce Lambert and Randy Cochran of the Agricultural Engineering Laboratory staff, for their help and guidance in electronics and computer knowledge.

To the other Agricultural Engineering staff, Norvil Cole, Robert Harrington, and Clifford Riley for their help and guidance.

For her many hours spent typing this thesis, a deep appreciation is extended to Fran Holbrook.

TABLE OF CONTENTS

Chapter		Page
I.	INTRODUCTION	1
II.	OBJECTIVES	8
III.	LITERATURE REVIEW	9
IV.	SYSTEM DESCRIPTION	16
	Hardware Description	16
	Operation Description	22
V.	METHODS AND MATERIALS	25
	Temperature Transducers	26
	Multiplexer	27
	Analog to Digital Converter	29
	Light Transducer	32
	Floor Water Set Temperature Switches . . .	36
	LED's	36
	Controller Relays	38
	Memory	40
VI.	PROGRAMS	43
	SYM-1 BASIC Language Program	43
	SYM-1 ASSEMBLY Language Subroutines	45
	AIM-65 to Tektronix Commmunication Program.	47
	Tektronix BASIC Program	48
	Statistical Analysis System	50
VII.	RESULTS AND DISCUSSION	52
VIII.	CONCLUSION AND RECOMMENDATIONS	56
	Recommendations for Future Work	57
BIBLIOGRAPHY		58
APPENDIXES		60

Chapter | Page

APPENDIX A - BASIC PROGRAM FLOWCHART FOR THE SYM-1 61

APPENDIX B - BASIC PROGRAM FOR THE SYM-1 65

APPENDIX C - ASSEMBLY LANGUAGE SUBROUTINES FOR THE SYM-1 72

APPENDIX D - BASIC DRIVER PROGRAM 81

APPENDIX E - ZERO PAGE RELOCATOR FOR RUNNING BASIC 83

APPENDIX F - AIM-65 TO TEKTRONIX DATA TRANSFER PROGRAM 85

APPENDIX G - TEKTRONIX BASIC PROGRAM 88

APPENDIX H - STATISTICAL ANALYSIS SYSTEM PROGRAM USED TO ANALYZE GREEN-HOUSE DATA 96

APPENDIX I - GREENHOUSE DATA 99

APPENDIX J - GREENHOUSE DATA 103

LIST OF TABLES

Table Page

I. Example Thermistor Temperature Resistance Response 28

II. Current to Frequency, Frequency Resistance Response 35

III. Greenhouse Data 100

IV. Greenhouse Data 104

LIST OF FIGURES

Figure Page

1. Greenhouse Construction 2
2. Pit Temperature Estimation Chart 5
3. System Flowchart 16
4. Control System 17
5. Control Case Face View 17
6. Control Case Rear View 20
7. Control Case Side View 20
8. Control Interior View 21
9. Relay Case 23
10. Current to Frequency Converter Circuitry 30
11. Multiplexor and Current to Frequency Converter Circuit 33
12. Photo Resistor Circuitry 35
13. LED Circuitry 37
14. Controller Relay Circuitry 39
15. Memory Board Circuitry 41

CHAPTER I

INTRODUCTION

A greenhouse design has been developed by Dr. C. Whitcomb of the Oklahoma State University's Horticultural Department which harvests excess solar heat. The excess heat in the greenhouse is collected and stored as heated water in a bed beneath the greenhouse. The Agricultural Engineering Department at Oklahoma State University has been cooperating with the Horticulture Department in experiments with this type of a greenhouse system (Paine, 1978, 1979). A slightly modified version of the greenhouse was built in 1980. A controller is required to control the harvesting of the excess solar energy and to control air temperatures within the greenhouse.

The greenhouse is a Quonset type pipe framework construction, with two layers of plastic, sealed at the edges and inflated by air pressure, as illustrated in Figure 1. Below the inner layer of roofing plastic is a third layer of plastic on a semi-circular pipe framework. An overhead spray line is installed on either side the length of the greenhouse between the middle and bottom layers of plastic. Beneath the greenhouse is a pit approximately thirty centimeters deep and filled with gravel-water combination. A

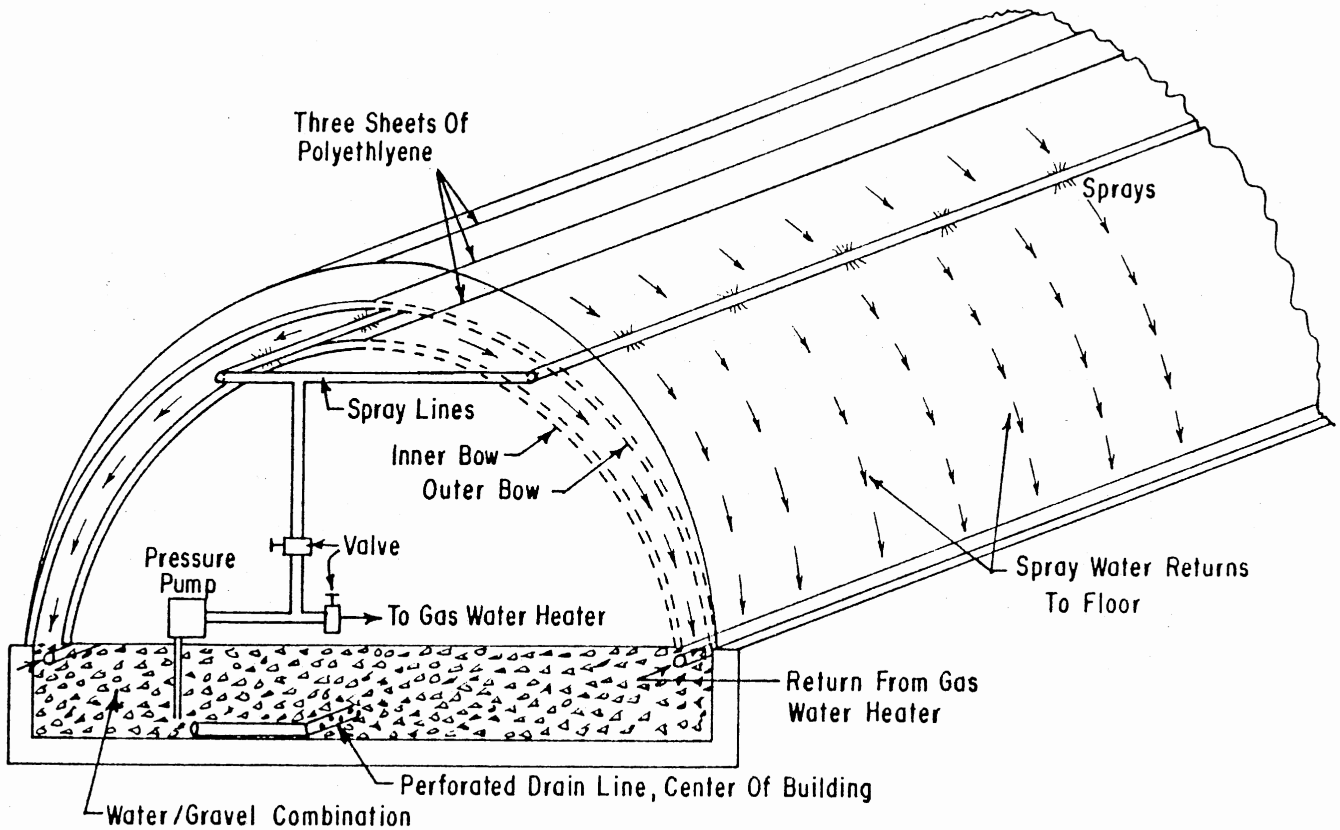


Figure 1. Greenhouse Construction.

layer of plastic serves as a floor barrier between the inside of the greenhouse and the pit. Water is pumped from the pit and sprayed into the space above the lower airspace. The water then falls on the inner plastic cover and, in running down the surface, extracts surplus heat from the greenhouse by conduction through the 6 mil plastic. The water then returns to the pit, transferring surplus heat from the greenhouse and airspace to the water/gravel mass. Harvesting of excess solar energy in this manner utilizes two important factors. Firstly, most greenhouses receive too much solar energy on sunny days and so must vent the greenhouse to prevent excessive high temperatures. By harvesting excess solar energy with spray water, requirements of venting are reduced. Secondly, stored solar energy may be used at a later time reducing the need of fossil fuels.

This type of greenhouse, which stores solar energy in a pit beneath the floor, is a warm floor greenhouse. The large surface area of the floor becomes a good radiator for the low temperature differentials characteristic of solar collectors of this type. Plants are positioned directly on the floor allowing heat from the pit to be conducted directly to plant roots. Maintenance of warm plant roots allows air temperatures surrounding the plant to fall much lower than those allowed by most greenhouses. By allowing temperature fluctuating from low temperatures at night to high temperatures during the day, energy input to the greenhouse by fossil fuels has been reduced markedly. Fall and spring plants have produced good physical and aesthetic characteristics.

A controller for this type of greenhouse has many control decisions. The major requirement is to maintain water in the pit at such a temperature that the floor will conduct and radiate sufficient energy to the plants to maintain them at a suitable temperature. Paine (1979) was able to develop a heat flow equation for this particular type of greenhouse. He was then able to calculate the required pit temperature to maintain greenhouse temperatures above a specified minimum temperature if the expected outside minimum temperature can be predicted. From Paine's equation, a chart (Figure 2) was developed for this greenhouse, allowing estimation of minimum required pit temperature. The bottom axis is the expected low outside air temperature, the vertical axis is the pit temperature and the three sloping lines in the chart are for 3 minimum required air temperature in the greenhouse. If the expected minimum outdoor temperature were too low to maintain the desired temperature in the greenhouse, the pit temperature could be increased to the level needed, as given by the graph. Two methods are used to raise pit temperature. Excess solar energy is used to raise the pit water temperature whenever possible otherwise, a backup water heating source is required.

There are a number of circumstances which may eventuate when the warm pit has insufficient energy transfer to the

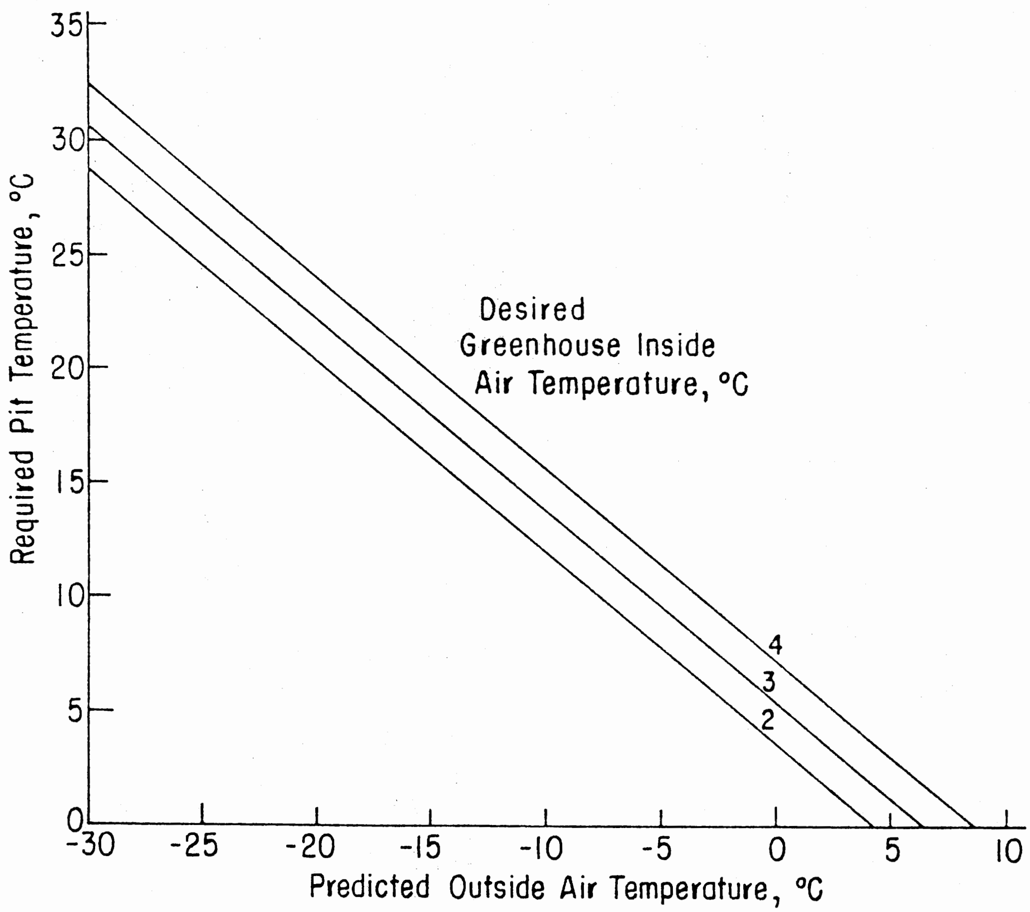


Figure 2. Pit Temperature Estimation Chart

inside of the greenhouse. This could be due to the operator not selecting the correct pit temperature, a malfunction in the backup water heater, or excess ventilation from a door being ajar. Whatever the reason, the controller must be able to accommodate the situation. An emergency auxiliary heater is installed in the greenhouse for this purpose. Should temperatures in the greenhouse still remain in a state detrimental to the plants after the auxiliary heater is activated, the controller may spray warm pit water between the middle and bottom plastic sheets, raising house temperatures and forming a heat loss source between the house and outside cold air. This would be an emergency situation, as energy would be rapidly lost from the pit. If both methods fail to raise or maintain the temperature, an alarm must be activated to indicate critical situation.

For all of these control functions, a 'smart' controller is required. In most situations, control automation in greenhouses has been achieved using solid state relays. This type of hardware system has proven to be reliable for most situations. However, they are applicable only to simple strategies of greenhouse operation.

With complex control algorithms required for the abovementioned greenhouse, microprocessors are more applicable. Their price has reduced markedly over the past few years, bringing them below the cost of an "equivalent capability" solid state relay system. Accuracy of the control operation depends upon the control strategy programmed into software memory and the accuracy of the measurement transducers. Control strategies are easily changed since they are software programs. In addition to their control ability, microprocessors have the ability to store and/or manipulate data received. Willits (1979) explained his use of microprocessors as:

> ideally suited to greenhouse research. The ability to precisely control multiple operation, while at the same time collecting and processing data, enable the research to conduct full-scale tests under near-laboratory conditions (p. 688).

Mitchell (1980) found that "the microcomputer system advantages far outweigh those of the conventional solid-state controls" (p. 9).

CHAPTER II

OBJECTIVES

This research is applicable to control of the greenhouse developed by the Agricultural Engineering Department of Oklahoma State University. The following requirements were viewed as the specific objectives:

1. Define and flowchart the functional requirements of the controller.

2. Design the control system hardware for the experimental greenhouse based on a microprocessor.

3. Write the software required for the correct operation of the controller.

4. Test the controller.

5. Determine the merits of extended capabilities for the controller to acquire greenhouse and controller data.

CHAPTER III

LITERATURE REVIEW

When this project was initiated, it was determined that a small, cost competitive control system was required for solar heated greenhouses. We directed our efforts towards using microprocessors as the control device, believing that the majority of control applications in the future will use these devices. To be cost competitive, the final device had to be able to compete in performance and price with hardware-type controllers.

The majority of research into control of greenhouses was achieved using 8-bit microprocessors, with the memory capabilities of the machine depending on the complexity of control required. Willits (1979) used a control Logic M-Series microcomputer which had separate memory and input/output control. His system had a total program memory of 6.5K of Erasable Programmable Read Only Memory and a workspace of 3K of Random Access Memory. Although he discussed the merits of the different programming languages which can be used, he chose a BASIC language which could interact with ASSEMBLY language. The M-series microcomputer was also chosen by Bowers (1978), utilizing BASIC as the control language and 6K of RAM memory. McClure (1977) in his work with tobacco curing used a M-series microcomputer, but used ASSEMBLY language rather than BASIC. For large control algorithms, he did advise the use of a higher level language.

A microprocessor enjoying increased popularity is the SYM-1 from Synertek. Mitchell (1980) chose the SYM-1 for his comparison of microprocessor control to solid state control. He decided to use the on-board 4K of RAM for his BASIC program, rather than EPROM's or PROM's. Walker (1981) also used the SYM-1, based on the 6502 language system, for control of Alcohol Fuel Fumigation. As with most researchers, he used BASIC language, which is slower to run but easier for operator programming.

Advantages of using microprocessors for control application are described in Mitchell's (1980) conclusions.

1. Microprocessor control strategy, although more complex than the strategy required for solid state systems, is implemented in considerably less time than with conventional systems.

2. Control changes are easier to make using software with microprocessors.

3. The hardware hookup for the microprocessor system is usually simpler than for the conventional system.

4. Easier temperature set points using microprocessors, rather than conventional systems.

5. If terminals are connected to the microprocessor, monitoring of the microprocessor control activities can be performed under program control.

Environmental control is basically temperature dependent, and the majority of temperature transducers used are linearized thermistors or thermocouples. Willits (1979) found that by using thermocouples with his M-series microprocessor and using one thermocouple as a reference temperature in an ice bath, he could achieve ±0.25 degrees centigrade accuracy. This type of accuracy is obtainable, however only if the ice reference junction is maintained accurately. Bowers (1978) also used type T thermocouples with an ice bath reference, but experienced thermal drift problems in their non-temperature compensated reed relays.

A different method of using thermocouples is to use a reference junction with a floating temperature. This method requires accurate temperature measurement of the reference junction by some means other than thermocouples. It also require the microprocessor be able to calculate equivalent reference units compatible to thermocouples.

Mitchell (1980) used linearized-thermistors. Accuracy of control to ±1 degree centigrade was maintained. His temperature measurement accuracy was limited by the resolution of the analogue to digital converter. However, measurements to ±0.5 degrees centigrade were obtained.

Most research using controllers utilize several temperature sensors. Individual temperature transducers must be selected for temperature readings. Willits (1979) used type T thermocouples connected to an analog to digital converter through relays. Due to a thermocouple's low voltage signals, three-wire, guarded, low thermal drift reed relays, Coto-coil model CR-3350-5, were chosen. Switching was achieved by decoding the upper four lines of the address buss to select 1 of 16 relay groups and the lower four lines of the address buss to select 1 of 16 relays in a particular group. Bowers (1978) also used a similar arrangement to Willits (1979), a series of latches, 1 out of 16 decoders and reed relays, feeding the output of the multiplexing networth to an analog to digital converter. Later research conducted by Mitchell (1980) used an analog to digital converter which had a built in 8-channel multiplexor.

In earlier research, Willits (1979) used a DVM, Newport Model 2003 converter, for advantages of stability and lack of need for amplification. This converter had a conversion rate of 4 conversions per second with a resolution of 10μv or 0.3 degrees centigrade using type T thermocouples. Due to a thermocouple's low signal output voltage, Bowers (1978) required to amplify the analog signal to the 0-10 volt input required for an analog digital converter. McClure (1977) also used a gain amplifier for the thermocouple analog signal, before the 8-bit analog to digital converter, model ADC-85. This A/D unit had a resolution of 208/256 or ±0.81 degrees centigrade.

Mitchell (1980), as mentioned before, used an analog to digital converter. He amplified analog signals before the A/D converter, which had an 8 bit resolution, allowing an accuracy of ±0.5 degrees centigrade. For greater accuracy, an A/D converter with 12-bit resolution could be used. The majority of A/D converters used are 8-bit. Mitchell, (1981) in the description of analog to digital converters, confirmed this view of what size of resolution should be used. Higher resolution generally increases the cost of the A/D. Conversion speeds depend on the speed of data process required, and again, cost of conversion is related to the conversion speed.

In some circumstances, analog to digital converters may not be the only method of signal conversion. Walker (1981) in his work with control systems for fuel fumigation converted an analog signal to a frequency using a 9400 integrated circuit Voltage-to-Frequency Converter (VFC). With the SYM-1 microprocessor, the frequency was an input to a frequency counter on board the SYM-1. Although much slower than A/D converters, V/F converters do have one significant advantage for any microprocessor. The frequency of two converters can be measured simultaneously. Cost of the V/F converter is usually below that of the A/D converter.

All low voltage signals are susceptible to interference from outside signals, commonly known as noise. Removal of noise from a signal may be achieved using two methods. Mitchell (1981) presents a paper with several ideas to accomplish filtering for microprocessors, both hardware and software. He presented a method of hardware filtering for noise on the input signal to the analog to digital converter, by inserting a capacitor across the feedback resistor of a gain amplifier. If no amplification was required, a simple resistance capacitance filter is useful.

Sampling rates varied according to researchers. Bowers (1978) used two rates in his solar energy drying. One rate of 15 minute scans was used for temperature and other data measurements, and the second rate was a control rate of 2 minute scans. Willits (1980) used a time scan interval of 15 seconds, which he theorized would be a trade off between a continuous sampling and control. He theorized continuous sampling is more advantageous for control, but scan intervals greater than 15 seconds are more beneficial for data acquisition. Mitchell (1980), using the SYM-1 microprocessor chose a sampling rate of 10 minutes and in his later paper (1981) gave a rule of thumb approach for estimating the sampling rate.

> Sample each channel at a frequency of at least five times the frequency of the signal to be measured if you want to accurately reproduce the input wave shape. For averaging, a slower rate would frequently be adequate (p. 6).

On the control side of the system, the microprocessor needs a buffer between it and the mechanism which it is controlling. Buffers are used to transfer microprocessor low DC voltage signals to a suitable signal applicable for controllers. Buffers serve a second purpose, that of isolating the microprocessor from the controller in order that no voltage spikes from the controller side can be transmitted through to the microprocessor. Bowers (1978) transferred control commands through latches to optic couplers. Optic couplers are ideal for purely resistive loads, but for inductive loads such as electric motors, a relay is required. Bowers (1978) had the output of the optic couplers fed into relays for control of inductive loads. As he mentioned, since optic couplers allow the microprocessor complete isolation from controllers, noise and feedback loops are eliminated. Willits (1979) and Mitchell (1980) used solid state relays which provided isolation for the microprocessor as well as being able to handle inductive loads.

CHAPTER IV

SYSTEM DESCRIPTION

Hardware Description

The operational requirements of the microprocessor controller chosen for control of the greenhouse previously mentioned can best be described in the block flowchart shown in Figure 3.

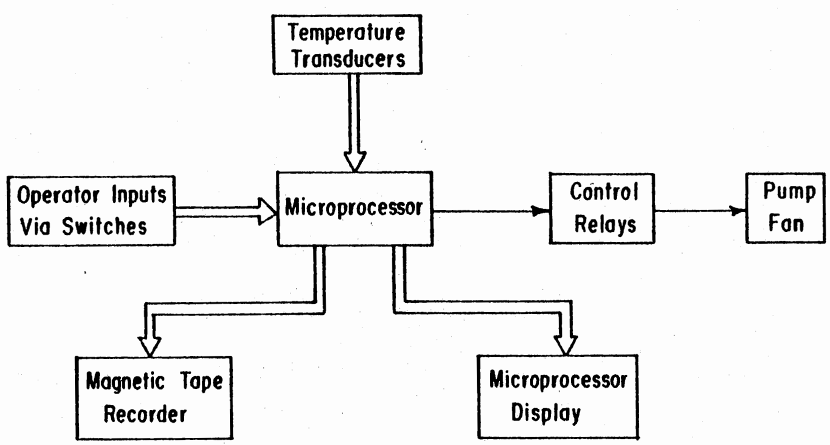


Figure 3. System Flowchart.

Programs and subroutines for temperature sensing, operation control and data transfer is described in Chapter V.

Figure 4 is a photograph of the microprocessor in its

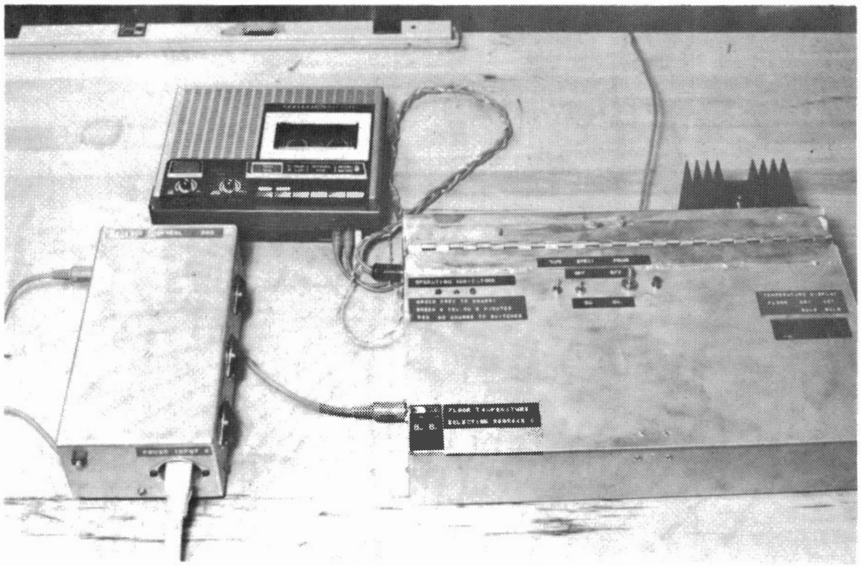


Figure 4. Control System

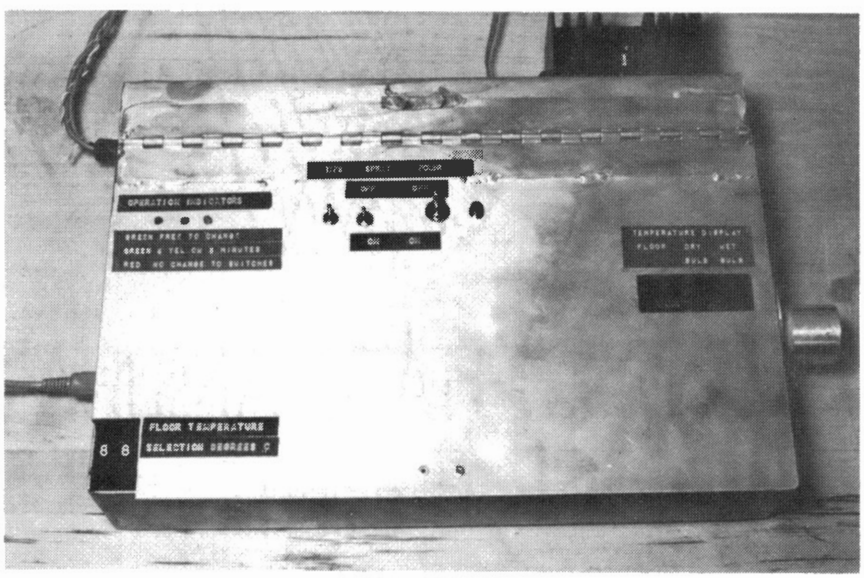


Figure 5. Control Case Face View

weather proof case, the two auxiliary connections, a relay case, and the magnetic tape recorder. Along the top of the box (Figure 5) are three on-off switches. They are labeled, power, night spray, and tape removal. There are two BCD thumb switches on the bottom left corner for setting pit water temperature in degrees C. On the center right hand side is a plastic window to 6 seven segment displays on the microprocessor, for displaying measured floor temperature, wet bulb temperature and dry bulb temperature. Another display is three colored LED's on the top left corner of the case. These are colored red, amber and green. They are indicators of the state of the microprocessor program operation. If the green LED is on, the microprocessor is only displaying the three temperatures. In this state, the operator can change the state of either of the two on/off switches, or the pit water temperature setting, without fear of the microprocessor reading them incorrectly. If the red LED is on, it signifies the microprocessor is in a state of calculations, control, or outputting the data to tape. In this state if one of the switches states were changed exactly when the microprocessor was looking at it, extraneous results might occur due to switch bounce. To overcome the problem of the operator not knowing the state of program execution, the amber LED was installed. It is lit simultaneously with the green, indicating to the operator that the microprocessor program execution is less than five minutes before the calculation mode.

To the rear of the case, Figure 6, there are phone receptacle sockets for sixteen temperature transducers. The transducers are individually numbered and must be inserted into the appropriate receptacles.

Along the left hand side of the case, Figure 7, there are two 5-pin dip sockets. The dip socket to the rear side of the case is connected to a magnetic tape recorder, while the second dip socket is connected to the relay box. This lead carries five control signals and one common ground.

Inside the microprocessor case, Figure 8, is the SYM-1 microprocessor and a custom made interface for the temperature transducer and the three colored LED's. Each temperature transducer has one input to the interface, the second line being connected to a common ground. These lines are connected to the inputs of a 16 to 1 multiplexor. The multiplexor has four select lines and one inhibit line. The output of the multiplexor is connected to a voltage to frequency converter, with output in turn, connected to an input/output line of the microprocessor. Another circuit on-board the interface is for a light detection transducer interface to the microprocessor.

On the interface board, there are two different power supplies for various functions. An 8V supply is regulated from the 12V incoming power supply for the multiplexor, voltage to frequency converter, and the light transducer interface circuit. The 5V regulated power for the microprocessor

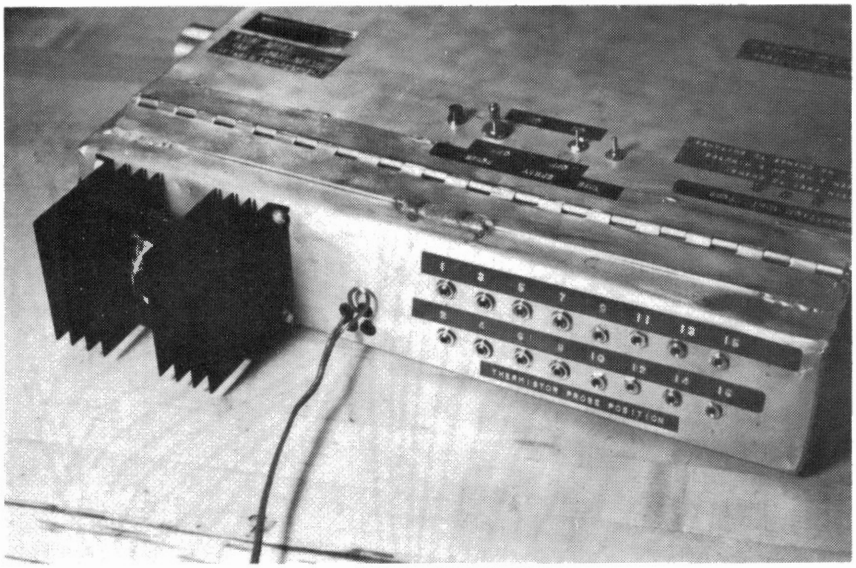


Figure 6. Control Case Rear View

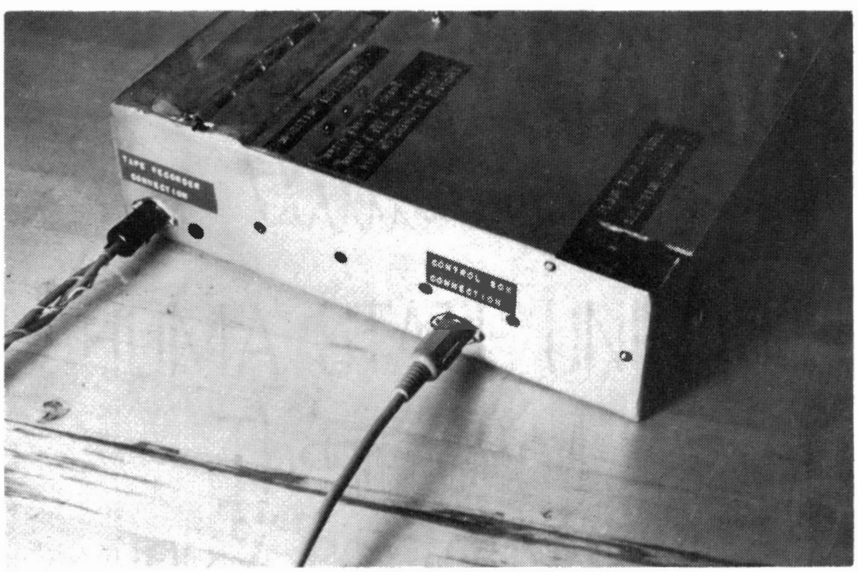


Figure 7. Control Case Side View

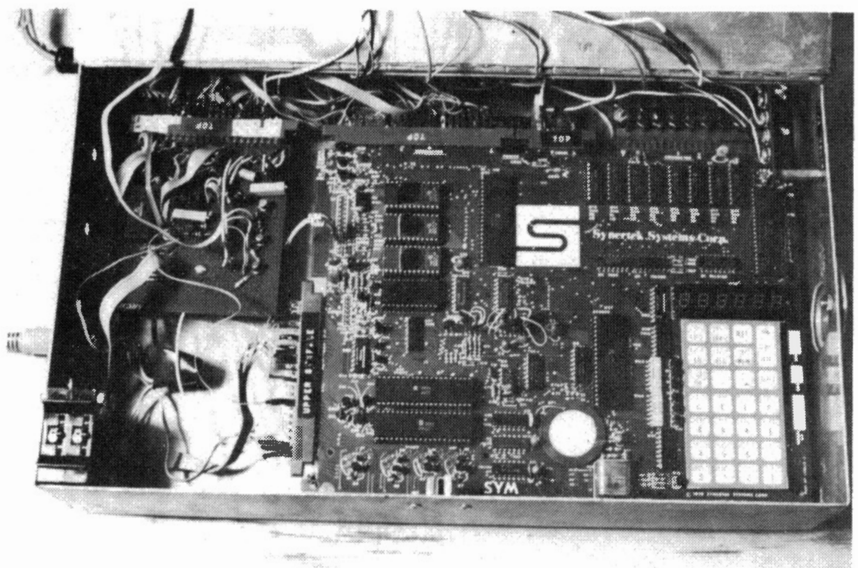


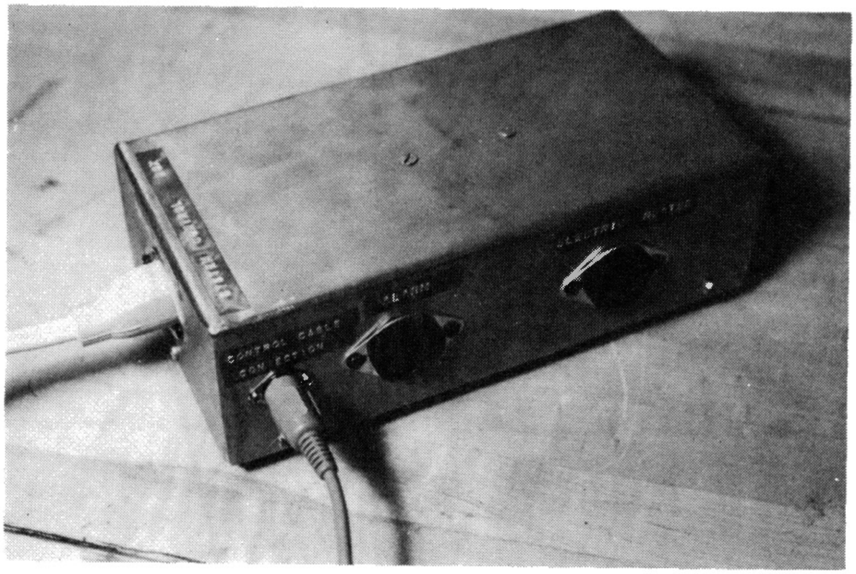
Figure 8. Control Interior View

is also connected to the interface board to provide power to the BCD switches, colored LED's and frequency reference voltage.

Adjoining the microprocessor case is the box housing the switching relays. Distance from the relay case to the microprocessor case reduces noise from A.C. switching. Connection is by a five wire cable. (Figure 9). The microprocessor has five different units to control. A solenoid valve is used to circulate pit water from a pressure pump through the gas water heater. Another solenoid valve is used to direct the pumped water through sprays in the space between the two plastic skins which serve as the roof. Movement of air into the greenhouse for purposes of cooling is accomplished by the use of an electric fan. The two controlled devices remaining are an electric heater and an alarm. The controller sockets are arranged for convenience, connected the side of the control case by using standard three pin power sockets.

Operation Description

The operator selects the starting address of the program and presses the GO button. The program enters a five minute display subroutine, displaying the digit 2 on the 6 seven segment displays. This number was selected purely arbitrarily, to display something before the first temperature measurements. After 5 minutes, a software interrupt occurs, which updates a software clock calendar. After the


CONTROL CABLE

first interrupt, the program jumps to temperature measurement subroutine, then to a control routine to set the controllers, based on the gathered temperature data. If in the control routine the air temperature within the greenhouse is below 4 degrees centigrade, the processor checks to see if the operator has selected the spray switch reflecting the desire to spray water between the plastic sheets to prevent the interior temperatures from falling below freezing. If so, the spray solenoid is switched on and the control routine jumps to a routine to dump the gathered data to tape. If the spray switch is off, the control routine must switch on an electric heater, then jump to a routine to dump data to tape.

The data dump to tape routine is also designed for operator input. If the tape is to be removed, the operator must switch on the "tape" switch and wait for the next interrupt. The microprocessor, therefore, checks the status of the switch to see if it is set after the data is put onto tape and if so, dumps to tape some extra information indicating the end of the tape data.

After the data dump routine, the microprocessor program waits for 5 minutes before again executing the temperature measurement subroutine. While waiting for the next software interrupt, the average pit temperature, the wet bulb and dry bulb temperatures are displayed on the microprocessor.

CHAPTER V

METHODS AND MATERIALS

A 6502 CPU microprocessor, the SYM-1 from SYNERTEK was chosen. The four major microprocessors that use the 6502 are the KIM-1, the SYM-1, the AIM-65 and the APPLE.

The SYM-1 is a 8 bit microprocessor which comes with 1K RAM which was expanded to 4K. Address and data lines have pinouts for an expansion memory of up to 64K bytes. The processor has a 4K monitor (SUPERMON), a 28 key dual function keyboard with a 6 digit LED display. The system has the capability to be connected to a teletype terminal via an RS232 connection, with the required software in SUPERMON. BASIC language has plug-in positions on board and/or an Editor with an Assembler. Jumper selection determines which chips are active on power up and if any are write-protected.

Connection to external devices is accomplished through 51 active Input/Output (I/O) lines which, by the addition of an extra ROM chip can be expanded to 71 I/O lines. The extra ROM chip brought the total number of timers to six, four of which are available for immediate use. Magnetic tape interface with full remote control was also available.

The large number of Input/Output lines and the availability of additional lines made the SYM-1 a more attractive microcomputer than the AIM-65 or the KIM-1 for this application.

For BASIC programming, a terminal with full-ASCII keyboard had to be connected to the SYM-1. Programs written for the SYM-1 are permanently stored in an EPROM. They can be comprised of BASIC programs, ASSEMBLY language programs, or a combination of the two. There is only one plug-in socket remaining for an EPROM if the BASIC chips are installed. SYNERTEK also produces a single board controller called the Super Jolt which incorporates the 6502 processor, a ROM, some RAM and eight Input/Output lines, plus a plug-in position for an EPROM. If programs could be designed on the SYM-1 and then placed on the Super Jolt, a good controller system could be developed. Also, with the versatility of EPROM's, the programs could be customized for a particular application.

Temperature Transducers

Thermistors, although non-linear, were chosen as the temperature transducers. The thermistors are model GA52P2 from Fenwall.

These thermistors were not interchangeable since they require individual calibration. Calibration was achieved by measuring the resistance of each thermistor at various temperatures over the range expected in reality. Water was used as the medium for calibration where temperatures were above freezing and alcohol was substituted for water for below zero temperatures. Temperature measurements of the medium were conducted using a Hewlett Packard 85 microcomputer connected to a 3497A Data Logger unit.

Experimental resistance temperature response of the thermistors are linearized to a higher accuracy if the temperature range is reduced from the large range expected at the greenhouse. Each thermistor is chosen for a particular location in the greenhouse and calibrated for the expected temperature range in that region (Table I). For positions where a large temperature range exists, two thermistors are used. One measures a high temperature range and the other thermistor measures the low temperature range.

Log of the resistance with respect to temperature yielded better linearization than a polonynomial curve fitting routine. The equation of the algorithm is then:

$$\text{Ln } R = C_1 + C_2{*}T \quad (1)$$

where

R = Resistance of a thermistor

C_1 & C_2 = Constants of linerization

T = Temperature of a thermistor.

Multiplexor

Selection of each of the sixteen thermistors in turn requires an efficient and reliable switch. Reed relays have been common in the past. However, selection is simplified by the use of a multiplexor.

TABLE I

AN EXAMPLE THERMISTOR TEMPERATURE RESISTANCE RESPONSE

Temperature °C	Resistance OHMS
16.4	305080
22.9	222480
26.5	187450
27.7	177580
29.1	165710
31.5	148600
33.3	136820
35.1	126090
38.1	110580

The multiplexor chosen was a 4067B from Motorola. This chip requires between plus 5V DC and 36V DC ground, sixteen analog inputs, four select lines, and an enable line. For thermistor selection, the four select line voltages are set to correspond to the input line desired by putting some high, i.e. plus 5 volts and other low i.e. ground. Selection was on hexadecimal notation basis. Enabling the chip was achieved by pulling the chip enable line low and holding it low throughout the sixteen selections. After the thermistors were selected, the chip enable line was pulled high to switch it off.

Analog to Digital Converter

The two main converters available convert either a voltage to a binary number, or convert a voltage to a frequency.

The voltage to frequency converter used in this research was an AD 537. Instead of using this chip to convert a voltage to frequency, it was adjusted to convert a current to frequency. The log resistance of the thermistors is inversely proportional to the temperature, and for a constant voltage across the thermistor, the current through the thermistor is inversely proportional to the resistance. Thus by measuring the current for a constant voltage, the current is proportional to the temperature.

The current to frequency converter used is shown in Figure 10. Input is through pin one and output is through pin nine. A pull up resistor to five volts on pin nine provides a zero to five volt frequency. The pull-up resistance is to give 5 volt frequency to the input/output pin at the microprocessor. Pin eight is the power supply to the chip. Between pins six and seven is the full scale

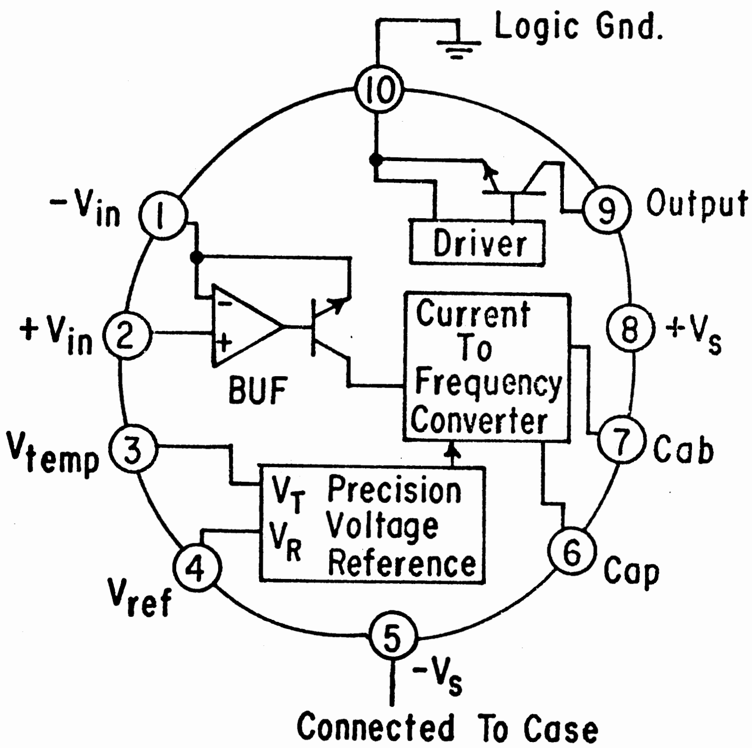


Figure 10. Current to Frequency Converter Circuitry

frequency adjustment. The frequency range output of the voltage to frequency converter can be adjusted using suitable equations, which select a capacitor to be positioned between pins 6 and 7. A 200 pF capacitor was chosen, giving an approximate full scale frequency of 20,000 Hz.

The reference voltage required for the thermistors is pin four, which is then joined to pin two of the input to the current to frequency converter (C/F).

With all low voltage signals noise is one of the major pitfalls. Signals of 1 volt or less are susceptible to noise, and suitable signal conditioning is provided. Thermistor and multiplexor grounding was isolated from the microprocessor. This was accomplished by channelling all of the grounds of the thermistors to pin five of the current to frequency converter. A heavy ground lead was provided from pin five to the ground terminal of the microprocessor. Two capacitors were bridged across the C/F converter input and ground (0.02μF and 1μF).

Calibration of the current to frequency converter was achieved by using a circuit in which the thermistor was replaced by a variable resistor. At each resistor setting, resistance was measured accurately on a digital multimeter and output frequency from the C/F converter was read on a digital oscilloscope as well as a frequency counter. Resistance to frequency values measured are shown in Table II. Resistance was related to the inverse of the frequency. Constants in equation 1 were determined from a regression analysis on the thermistor calibration data. Constants in equation 2 was determined by regression analysis from voltage to frequency converter calibration data:

$$\text{Ln } R = C_1 + C_2 * T \tag{1}$$

$$R = C_3 + \frac{C_4}{F} \tag{2}$$

where C_1, C_2, C_3, and C_4 are constants F = frequency from converter.

Combining equations 1 and 2.

$$T = \frac{\text{Ln} \dfrac{C_4}{F - C_3} - C_1}{C_2} \tag{3}$$

An overall circuit of the thermistors connected to the multiplexor which in turn is connected to the voltage to frequency converter is shown in Figure 11.

Light Transducer

Controllers often have different control algorithm for night and day. A light transducer was required for this distinction. Willets (1979) used a pyranometer as his light transducer. A device known as a photocell or photo resistor was chosen for this application. The photocell is a light variable resistor and the model used is a cadmium sulphide

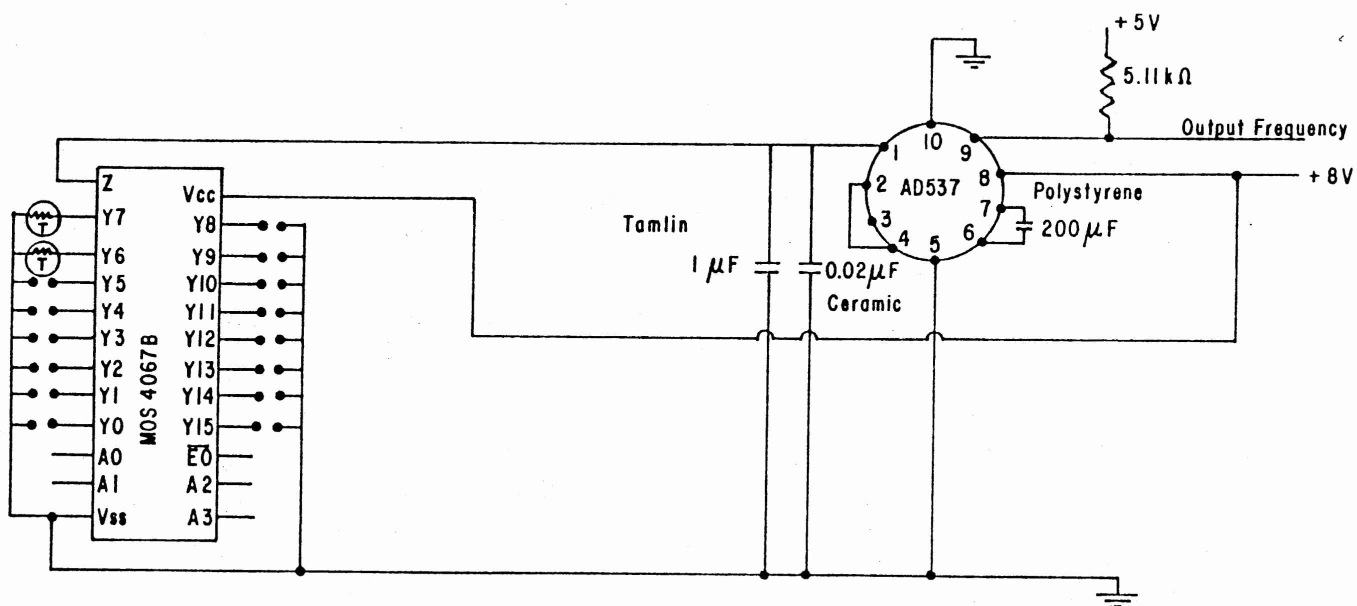


Figure 11. Multiplexor and Current to Frequency Converter Circuit

photocell 276-116.

The photocell was connected in a circuit as shown in Figure 12. An LM339 inverting comparator was used to compare the two inputs. By connecting the negative input of the comparator to a voltage divider which has a fixed and variable resistor and the positive input to another voltage divider which has a fixed resistor and the photocell, the output of the LM 339 remains high until the positive input voltage rises higher than the negative input voltage. Calibration was achieved by adjustment of the variable resistor until the comparator output fell low for the required light intensity.

TABLE II

CURRENT TO FREQUENCY, FREQUENCY

RESISTANCE RESPONSE

Frequency (Hz)	Resistance (Ohms)
564	831530
684	690550
788	600780
958	497730
1192	400270
1598	300850
2384	202250
4694	103260
9824	49280
17486	27570
27998	17190

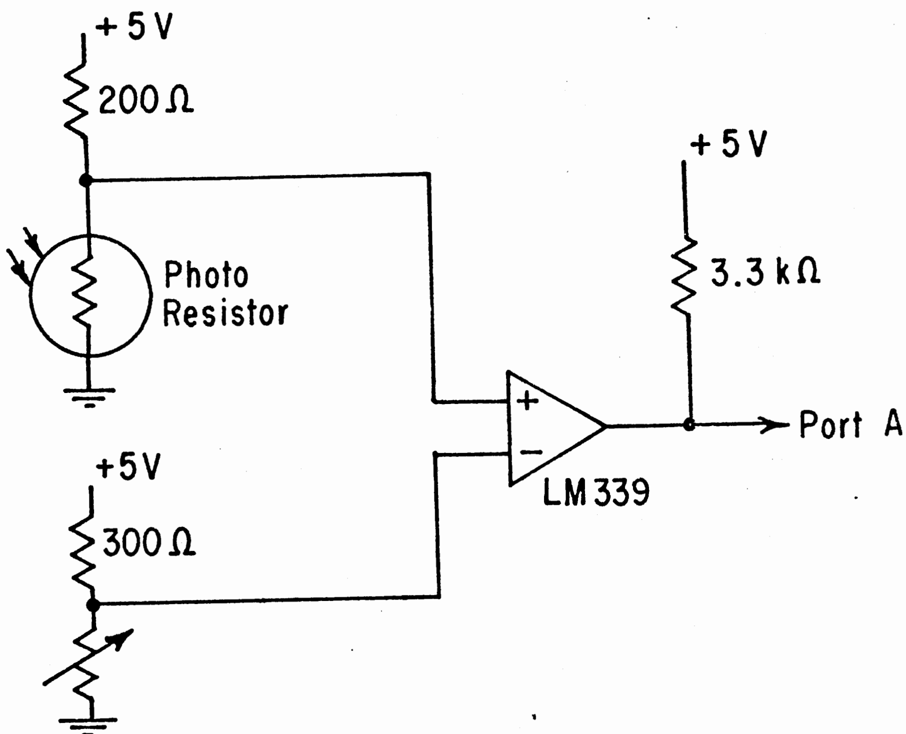


Figure 12. Photo Resistor Circuitry

The output of the comparator was fed directly into one of the input lines of the microprocessor where its status was monitored by the SYM-1 BASIC program.

Floor Water Set Temperature Switches

The operator must be able to enter the pit water to the microprocessor temperature that is required to maintain inside air temperature above 4 degrees centigrade. The minimum water temperature was entered by two BCD thumb switches, each having an output range from 0-9. With two switches combined a total range of 0-99 degrees centigrade is obtainable.

The switches have four active lines, each switch being connected to half of an 8 line Input/Output port of the microprocessor. As the temperature was selected, the active low lines of the switches was monitored in the BASIC program.

LED's

Program execution state was indicated by three LED's, red, amber and green. Figure 13 illustrates the LED's circuitry. LED's were selected due to their low current requirements and relative ease of connection. A pnp transistor was used as the switch.

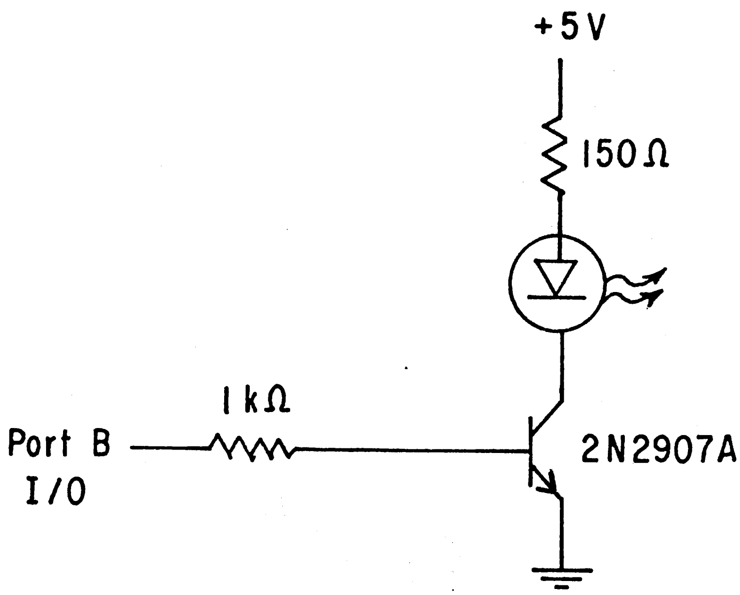


Figure 13. LED Circuitry

Controller Relays

One of the primary requirements of relays connected to microprocessors is to isolate the driven load from the microprocessor. This reduces problems of voltage spikes and of noise filtering back to the microprocessor. Relays must be correctly matched to the microprocessor. Without output buffering, an input/output line can usually drive a single TTL load of approximately 2.35 mA. Many control relays require a higher coil current than the I/O lines can produce, limiting the relays suitable to microprocessors to solid state relays or optical relays.

Optical relays were initially chosen for all control relays. The AC current to the controllers through these relays were suitable for all of the controllers except for the electric heater. To obtain the current required for the electric heater from the relay, the output from the optical relay was passed to the base of a TRIAC. A separate AC power line was used for the electric heater. All of the other relays received power from one AC power line (Figure 14).

All control lines from the microprocessor were connected directly to the relays, with all having a common ground line which was grounded at the microprocessor. AC grounds were common to all controllers except for the electric heater which had a separate ground.

The relay type chosen was the optical relay, which

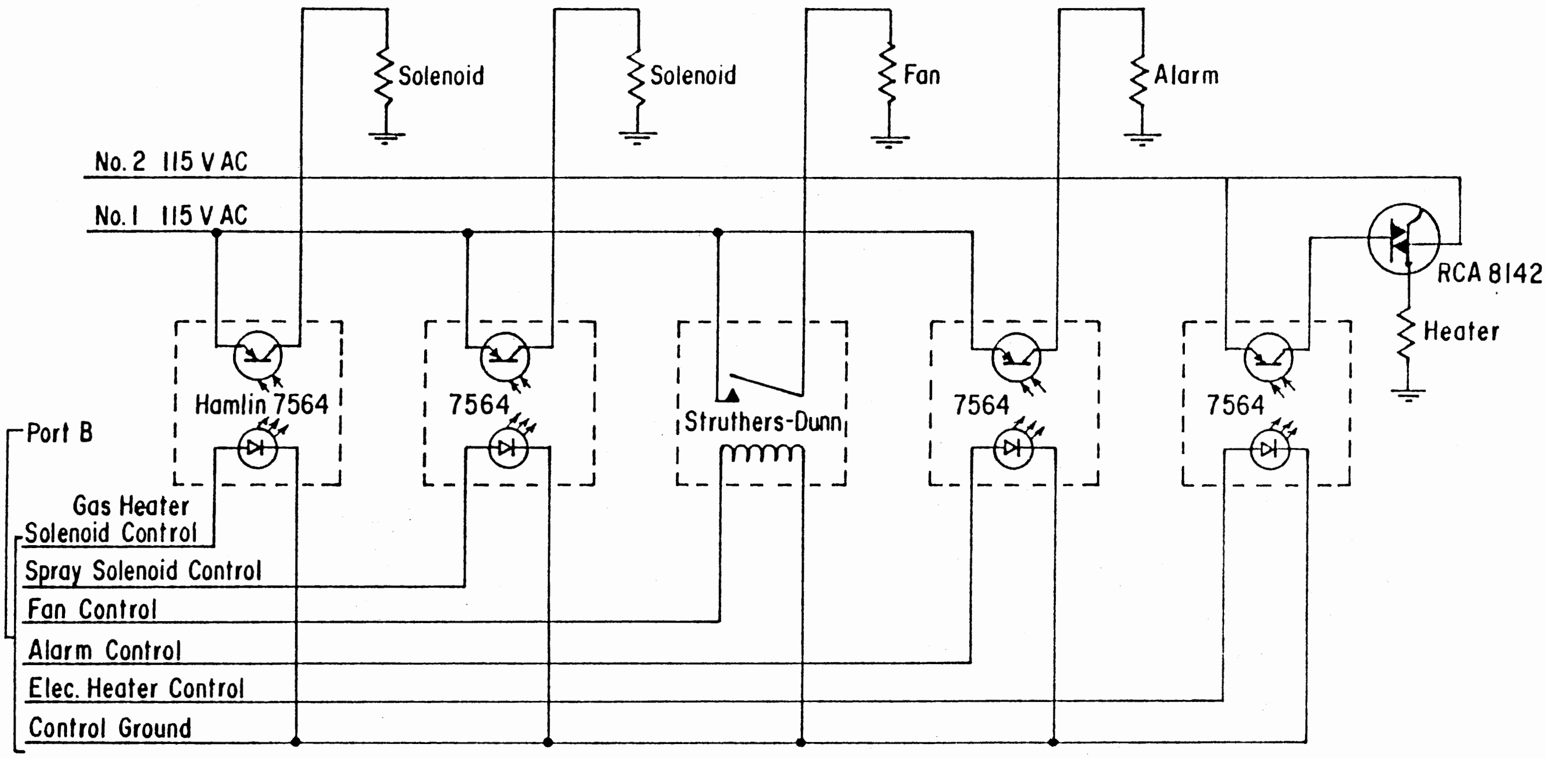


Figure 14. Controller Relay Circuitry

worked well for purely resistive loads. However, when connected to an inductive load the relay burned out. A solid state Struthers-Dunn relay, was used in place of the Hamlin 7564 optical relay.

Memory

The AIM-65 development system was used to program the EPROM for the SYM-1. The memory page B000 was found to be the only compatible 4K block which did not interfere with either the SYM-1 or AIM-65 monitors.

To achieve the selection of memory, the memory is divided into four 1K blocks. The top three address lines of the microprocessor address 16 4K blocks of memory. Address lines 10, 11, 12 can be used to select eight 1K blocks of memory. By passing the two groups of three address lines through two 3 to 8 line active low output decoders 74LS138, the desired 4K of memory may be selected. An OR gate 74LS32N was used to correctly select the 1K block of memory required (Figure 15).

Address lines are buffered by 74LS241's and the data lines are buffered by a bidirectional buffer, a 74LS245. The 74LS245 was only activated when the memory board was selected, preventing unwanted data from being on the data bus.

The memory board was connected directly to the expansion connector of the SYM-1. A 4-K memory board had to be designed and built for the SYM-1. The memory chips used

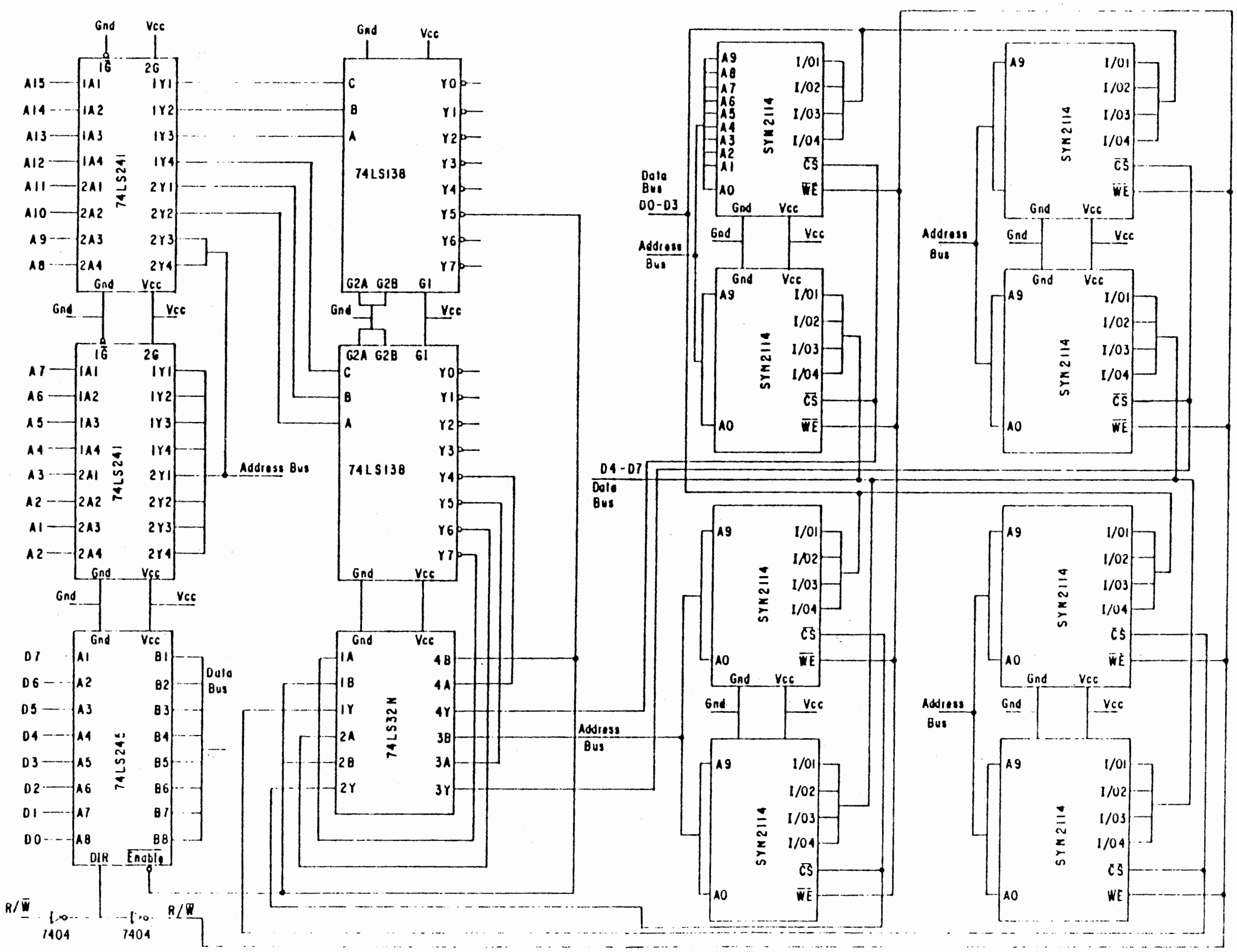


Figure 15. Memory Board Circuitry

were 2114's, which are a 4096 bit static random access memory. Memory of the 2114 is arranged as 1024 words of 4-bits, thus for an 8-bit word of 1024 words, two 2114's are required.

CHAPTER VI

PROGRAMS

Programs used for this study were broken into four groups. Those associated with the SYM-1 microprocessor, a program for the AIM-65, a program for a Tektronix microcomputer and a program for an IBM 370 main frame.

SYM-1 BASIC Language Program

The SYM-1 BASIC program is structured to interact with ASSEMBLY language subroutines for information gathering to and distributed from the SYM-1. Data manipulation and control is accomplished using BASIC.

A flowchart of the program is shown in Appendix A and the program in Appendix B. Organization of the program was centered around a repetitive 5 minute software interrupt caused by the timing-out of one of the on-board timers. After the BASIC program is started, two ASSEMBLY language subroutines is accessed before control status is changed.

One is for memory initialization and starting timers. The second subroutine accessed is a routine to convert hexadecimal temperatures to decimal temperatures and display these temperatures while waiting for the software interrupt.

Hexadecimal data gathered by the ASSEMBLY language subroutine is transferred by BASIC to an array by PEEKing to a memory location. The PEEKing step automatically converts the hexadecimal number to decimal. From the data in the array, average floor temperature and the average inside air temperature were computed.

There were three control algorithms used in the BASIC program. One algorithm was used when the inside air temperature at plant level fell less than 4 degrees centigrade. Two control subalgorithms may be used. Warm floor water may be sprayed between the plastic sheets, or an auxiliary electric heater may be used to supply extra energy to prevent freezing. The second control algorithm is a night mode control of repetitively comparing the averaged pit water temperature to the operator set point temperature and adjusting by use of a gas water heater. The final algorithm for control is for daylight conditions. Solar energy harvesting is initiated if the temperature between the plastic skins is 6 degrees centigrade above the average pit temperature. A pump sprays water between the plastic sheets to collect excess energy. It is stopped whenever the return water form the sprays is less than the average pit water temperature.

After a control algorithm has been run, the tempe atures of the greenhouse and operation of the controllers is dumped to a magnetic tape recorder by use of an ASSEMBLY language subroutine. Inside high temperatures that may cause reduced plant growth are then controlled on a 5 minute time interval by the switching of a fan. Once two of these software interrupts have occurred, the programs returns to the second ASSEMBLY language subroutine encountered when the program was first run.

SYM-1 ASSEMBLY Language Subroutines

There are three subroutines accessed by the SYM-1 BASIC program, some containing more than one individual routine (Appendix C). The initialization subroutine brings the output lines connected to the controller and other devices to a known state and determines which input/output lines are to be for data input and which for data output. Two timers on-board are set for a 5 minute interrupt and the interrupt routine vectors are set. The last step of the Initialization routine is to switch on the green LED before returning to the SYM-1 BASIC program.

The purpose of the interrupt routine is to reset the timers used to cause the software interrupt, update a clock calendar and switch off the green LED and switch on the red LED. The clock calendar is based on months, days (30 per month), hours and minutes, updated by 5 minutes every software interrupt. An interrupt flag is also set to indicate that an interrupt has occurred.

The second subroutine accessed by the SYM-1 BASIC program is comprised of three routines. Three bytes of data in three specific memory locations are converted from hexadecimal to decimal, and stored in a new memory location (DeJong, 1980). The second routine displays the three numbers just converted to decimal (DeJong, 1980). Displaying of the three temperatures is in a loop, escaped only if an interrupt has occurred. If the interrupt flag is set the third routine of this subroutine is run. This is the counting of all of the frequencies associated with the thermistor outputs. Counting is based on half-second time spans for each thermistor. Frequencies counted by this routine are stored in two data banks, one bank for the low byte of data and second bank for the high byte.

The third ASSEMBLY language subroutine accessed by the SYM-1 BASIC program, is a routine to put the various temperatures and controller operation data onto magnetic tape. Reading of the data from off the magnetic tape is achieved by use of the AIM-65 in the laboratory. The AIM-65 and the SYM-1 each have two rates of sending data to the magnetic tape, of which the KIM-1 speed is compatible to both. The SYM-1 monitor has the actual routine for putting the data onto tape but requires the starting and ending address of the memory block, the tape dump rate (i.e. KIM-1) and the data block identification. After the data is dumped to magnetic tape, a check is made to see if the 'end of tape' switch is on. If on, a specified number is put into a memory location and the data re-dumped to indicate the end of the data block.

An ASSEMBLY language routine which is not accessed by SYM-1 BASIC is the BASIC driver program (Appendix D). This is the very first program run in the EPROM. This program sets up zero page which had previously been stored in a high portion of the EPROM using the Zero Page Locator program using BASIC (Appendix E). After zero page is initialized for the SYM-1 BASIC program, the BASIC program automatically starts.

AIM-65 to Tektronix Communication Program

Communication between the AIM-65 development system and the Tektronix is achieved by use of the program in Appendix F. The program works in the following manner. Data existing in memory in the AIM-65 is sent across on an R$S 232 calbe to the Tektronix in a hexadecimal format. The Tektronix receives the data and converts it to decimal.

Each byte of data is separated by a carriage return. On completion of sending the data the AIM-65 switches on the tape recorder and loads a new block of data.

Hexadecimal numbers can be sent from the AIM-65 to the Tektronix, however the Tektronix can only convert ASCII to decimal, not hexadecimal. To ensure problems do not occur when the Tektronix tries to convert the hexadecimal data to decimal, the decimal 33 is added to each number sent from the AIM-65. Data was put in the RS 232 port to the Tektronix using a routine in the AIM-65 monitor.

To get data from the tape by AIM-65, bytes of data had to be put into the AIM's memory locations for initialization of the tape load routine. Such data as the tape data format, input data device, and data identification had to be stored into memory. A subroutine in the monitor was accessed, setting up the data load routine based on data previously put into memory. A second subroutine is accessed to reset input/output devices which were changed in the dump subroutine.

The last step in the program is to switch off the tape recorder. When the subroutine to load data from magnetic tape is accessed, the tape recorder is switched on, but not switched off after the data is loaded.

Tektronix BASIC Program

The purpose of the Tektronix is to provide sufficient memory and capabilities to manipulate and store the data coming from the AIM-65 development system, print the data in column form and transfer the data to the University's main computer (Appendix G).

The AIM-65 development system and the Tektronix are connected via a cable on input/output Port 40 of the Tektronix. This port has a memory buffer to hold up to 255 bytes of data plus software capabilities to clear this buffer for input.

The Tektronix has software routines to establish the communications between other computers using baud rates accessed by a "call" statement with the various parameters included. For Tektronix controlled communication between the two computers, a baud rate of 1200 operates satisfactorily. This only requires the Tektronix program to send to the AIM-65 the Rubout character (decimal 127) before the two computers are communicating. A data file on the Tektronix has the ASSEMBLY language (Appendix F) program which is sent across the RS 232 cable to the AIM-65. Each mnemonic program instruction is broken into single characters to be sent out. A delay has to be incorporated on the Tektronix to allow the AIM-65 to accept instructions.

Thirteen columns of data will be printed out by the Tektronix only after all the data has been received from the AIM-65.

After printing the headings for the column and establishing communication between the two computers, the Tektronix begins sending instructions to the AIM-65 on where to run the ASSEMBLY language routine it has stored in memory.

As each block of data is sent from the AIM-65 to the Tektronix, the constant 33 decimal is subtracted before the data is manipulated and stored in an array. Manipulation involves updating a real time clock calendar, determining if temperatures are positive or negative and setting an indicator in the array for the operation of the controllers. If the controller was ON, a 1 is put in the array. If the controller was OFF, a 0 is put in the array.

After a data tape has been processed, the operator can elect to have additional data tapes processed before the data is printed.

The relative humidity is now calculated from an algorithm, in chapter 5 of ASHRAE Handbook of Fundamentals, 1977, which requires the wet bulb temperature, dry bulb temperature and total atmospheric pressure. Atmospheric pressure in atmospheres was chosen according to the height of Stillwater above sea level.

The method by which this algorithm operates, is that the saturation water vapor pressure over a free water is calculated using a method developed by Keenan, Keyes, Hill and Moore. The humidity ratio of moist air is then calculated knowing the saturation water vapor pressure and the total atmospheric pressure. Calculation of the humidity ratio is accomplished knowing the wet bulb temperature. The degree of saturation is then calculated knowing the two humidity ratios calculated, which then allows the calculation of the relative humidity.

After computation of the relative humidity, the Tektronix may be connected to the University's main frame by telephone and the data transferred.

Statistical Analysis System

The program on the University's main frame computer (IBM 370) is used as an indication of the ability of being able to manipulate and present in various ways, the data received from the Tektronix microprocessor. The data received from the Tektronix is stored in a cumulative array which can be analyzed at a later data (Appendix G).

CHAPTER VII

RESULTS AND DISCUSSION

Some problems arose when the controller was first operated. As each was detected it was solved, however it was difficult to obtain continuous operation of some time length with these problems. The period concided with the cold January weather and for this short time, the SYM-1 was able to perform the control functions within the greenhouse acceptably. Data was collected about the temperatures within the greenhouse and on the operation of various controllers. The ability of the SYM-1 to put the data onto magnetic tape for further evaluation by other computers allows the performance of the control systems to be monitored.

Two day's data is shown in Table III (Appendix I) and Table IV (Appendix J). The first day's data is shown in Table III (Appendix I). Operation of the gas and electric heater and the sprays are indicated as either ON or OFF. The greenhouse pit has a leak which requires water to be added. This refilling is usually done in late afternoon. Refill water is usually about 10 degrees centigrade lower in temperature than the pit water temperature. Refilling, then, usually requires the SYM-1 to switch on the gas heater to raise the water temperature to that set by the operator. Setting of the desired floor water temperature is usually done in late afternoon as well.

For the data in Table III (Appendix I) for the first day, the expected overnight low temperature allowed the operator to set the pit water temperature at 15 degrees centigrade. The gas heater was not required to be switched on since the average pit water temperature after refill remained higher than that 15 degrees. Because the pit water temperature was so high, the inside air temperature did not fall below 4 degrees centigrade which would cause the SYM-1 BASIC to go into the critical mode. Overall, the SYM-1 did no controlling as none was required.

Twice the sprays were switched on during the day routine. However, for both times, they were operated for only 15 minutes. This was due to the fact that the return water temperature was being compared to the average pit water temperature. In fact it should be compared to the temperature of water entering the pump the coldest water in the reservior. A slight programming change would overcome this situation.

The wet bulb and dry bulb temperatures are tabulated as integer values and are truncated by the SYM-1 when BASIC POKE's the data into memory. This situation has been rectified by breaking the wet bulb and dry bulb temperatures into integer and decimal values.

Data from the following day, is shown in Table IV (Appendix J). The set temperatures had been raised from the previous setting of 15 to 18 degrees centigrade, resulting in the gas heater being operated from 1720 hours to 1835 hours. Night spraying to warm the greenhouse was allowed. Between the hours of 2250 and 2320 the temperature within the greehouse had fallen below 4 degrees centigrade, so the sprays were activated. Use of the sprays at night not only prevented excessively low temperatures, but raised the air temperature in the greenhouse.

Three times during the day the sprays were activated to collect solar heat but on all three occasions the spraying did not last for more than 15 minutes.

There were four problem areas experienced in the initial testing of the SYM-1 controller. Firstly, there were SYM-1 software BASIC problems. One of these was when the inside air temperature thermistor near the top of the greenhouse experienced temperatures below zero. The BASIC program cannot POKE into memory a negative number, so went into an error state. The other software problem was that of the spray return water temperature being compared to the average pit water temperature. The pit water temperature may average higher then the temperature of the water going to the sprays so solar energy may have been harvested.

Secondly, a bad choice of one particular relay led to one of the optical relays burning out. This was for the control of the fan where the inductive load had not been previously considered. The solution was to use solid state relays.

Thirdly, the AIM-65 development system used to burn and reburn EPROM's failed repeatedly. This prevented the program modification from being done.

Fourthly, data transfer from the magnetic tape to the AIM-65 development system had problems. The non-linear response of the tape recorder prevented the operator from estimating the end of the data for the particular run. Blank tapes had been used, but proved time consuming to earase the data before reuse. A switch was installed on the control box so that the operator could indicate to the SYM-1 when he was ready to remove the tape from the recorder. If the "tape" switch is on, the SYM-1 dumps an end of block indicator which the Tektronix can recognize.

CHAPTER VIII

CONCLUSIONS AND RECOMMENDATIONS

The control system was designed according to the original objectives.

1. The requirements of the controller were to control the temperatures, harvest excess solar energy and collect data within the greenhouse built by the Agricultural Engineering Department at Oklahoma State University.

2. A SYM-1 microprocessor was chosen as the controller.

3. BASIC language was chosen with ASSEMBLY language subroutines, for the control software.

4. The controller performed to the design requirements for the limited time span of the test.

5. Data acquisition proved to be very helpful and efficient, allowing rapid data transfer from magnetic tape to the University's computer.

The system is relatively low in total cost and the data acquisition could be expanded easily for more data collection for research.

Recommendations for Future Work

1. Replace the present thermistors with an interchangeable type which does not require individual calibration. Make frequent checks on the operation of the controllers.

2. For research applications, increase the number of temperature transducers to use the data acquisition capabilities and increase the variety of operations to control.

3. If recommendation 2 is to be used, reprogramming for more temperature transducers and additional control algorithms are recommended.

BIBLIOGRAPHY

Bowers, C. G., Huang, B. K., Nassar, A. H. 1978. Solar Energy Drying and Process Control with Microcomputer. The American Society of Agricultural Engineers, Paper No. 78-3537.

De Jong, M. L., 1980. Programming and Interfacing the 6502, with Experiments. Howard W. Sams & Co., Inc., Indianapolis, ID.

Doebelin, E. O. 1966. Measurement Systems Application and Design. McGraw-Hill Book Company, New York, NY.

Handbook of Fundamentals. American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc. 1977. Sec. 5.0-5.5.

McClure, W. F. 1977. Design and Development of Two Microprocessor Based Data Acquisition Systems for Tobacco Curing and Greenhouse Operation. The American Society of Agricultural Engineers, Paper No. 77-5036.

Mitchell, B. W., Drury, L. N. 1980. Microcomputer and Conventional Solid-State Controls for A Solar Heating System. The American Society of Agricultural Engineers, Paper No. 80-1561.

Mitchell, B. W. 1981. Signal Conditioning for Analog Inputs to Microcomputers. The American Society of Agricultural Engineers, Paper No. 81-1611.

Paine, M. D., Mahoney, G. W. A., Whitcomb, C. E. 1978. Self-Sufficient Solar Greenhouse. The American Society of Agricultural Engineers, Paper No. 78-489.

Paine, M. D., Mahoney, G. W. A., Whitcomb, C. E. 1979. Predicting Minimum Overnight Temperatures in Solar Greenhouses. The American Society of Agricultural Engineers, Paper No. 79-696.

Walker, J. T. 1981. Microprocessor Control System for Alcohol Fuel Fumigation. The American Society of Agricultural Engineers, Paper No. 81-1610.

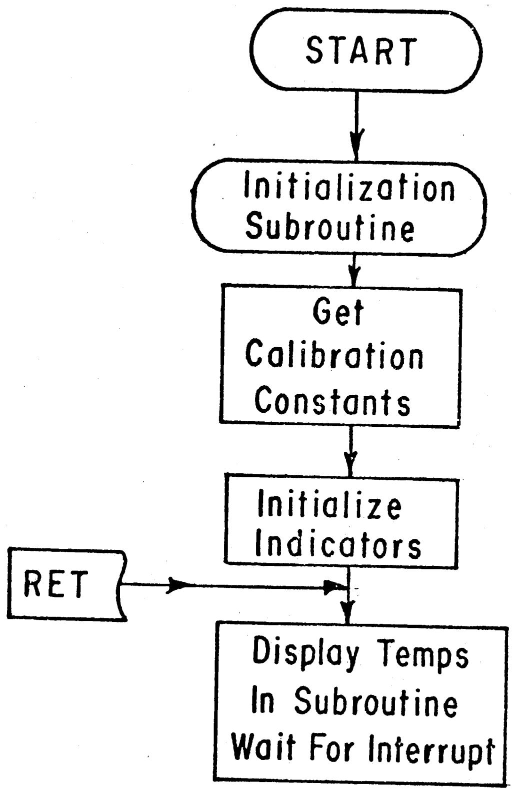
Willits, D. H., Karnoski, T. K., McClure, W. F. 1980. A Microprocessor-Based Control System for Greenhouse Research: Part I Hardware. Transactions of the American Society of Agricultural Engineers, Vol. 23, No. 3, pp. 693 - 698.

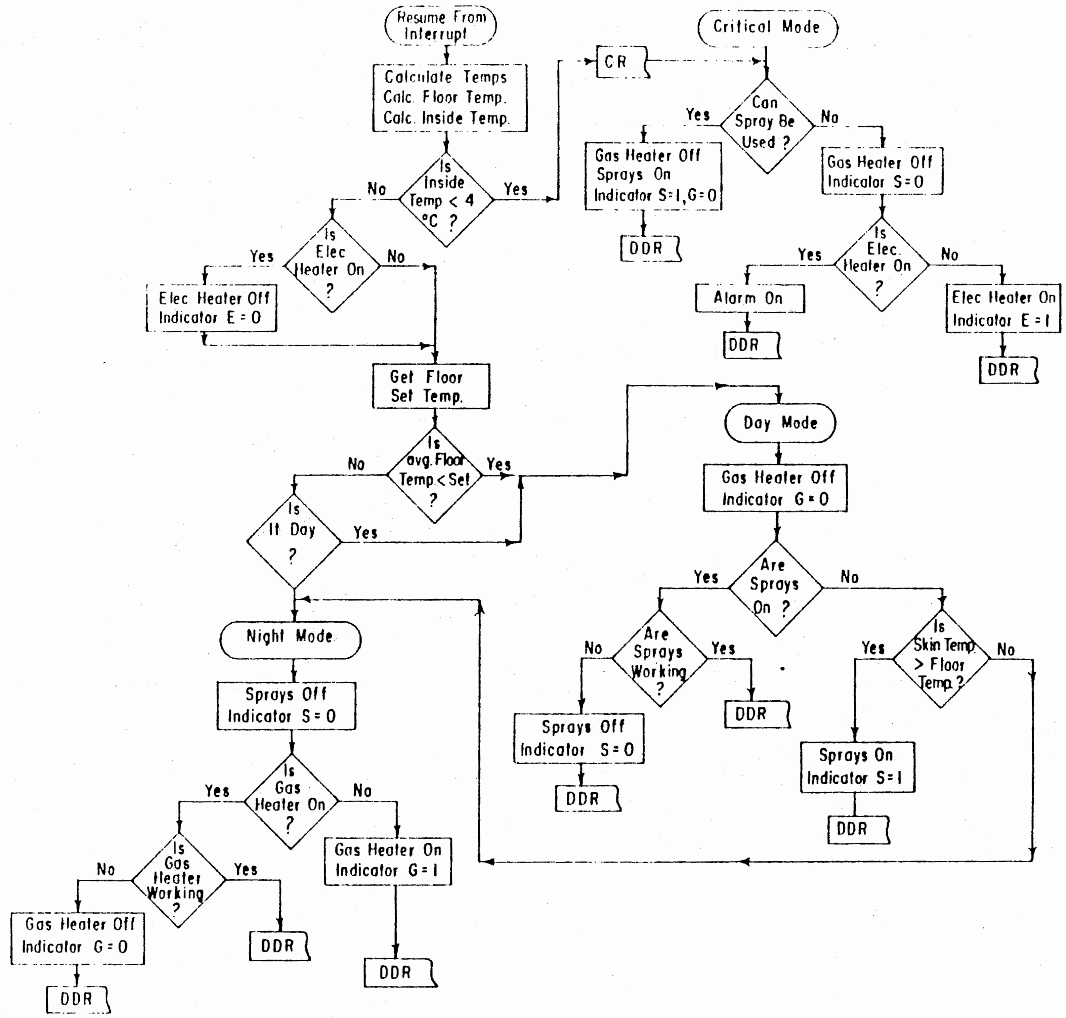
Willits, D. H., Karnoski, T. K., Wiser, E. H. 1980. A Microprocessor-Based Control System for Greenhouse Research: Part II Software. Transactions of the American Society of Agricultural Engineers, Vol. 23, No. 3. pp. 693 - 698.

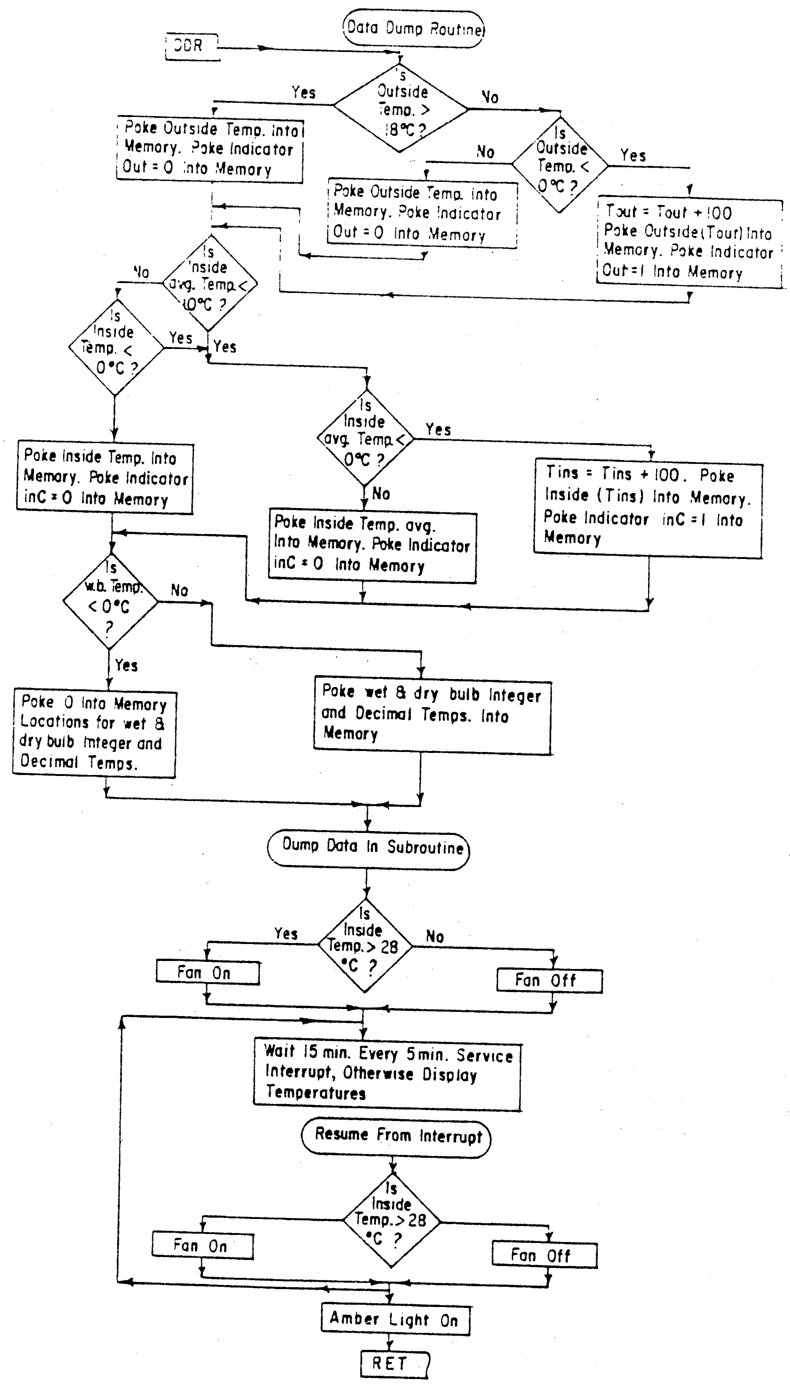
APPENDIXES

APPENDIX A

BASIC PROGRAM FLOWCHART FOR THE SYM-1


START
Initialization Subroutine
Get Calibration Constants
Initialize Indicators
RET
Display Temps In Subroutine Wait For Interrupt


Resume From Interrupt
Calculate Temps
Calc. Floor Temp.
Calc. Inside Temp.
Is Inside Temp < 4 °C ?
No
Yes
Is Elec Heater On ?
Yes
No
Elec Heater Off
Indicator E = 0
Get Floor Set Temp.
Is avg. Floor Temp < Set ?
No
Yes
Is It Day ?
Yes
Night Mode
Sprays Off
Indicator S = 0
Is Gas Heater On ?
Yes
No
Gas Heater On
Indicator G = 1
DDR
Is Gas Heater Working ?
No
Yes
Gas Heater Off
Indicator G = 0
DDR
DDR
Critical Mode
CR
Can Spray Be Used ?
Yes
No
Gas Heater Off
Sprays On
Indicator S=1, G=0
DDR
Gas Heater Off
Indicator S = 0
Is Elec. Heater On ?
Yes
No
Alarm On
DDR
Elec Heater On
Indicator E = 1
DDR
Day Mode
Gas Heater Off
Indicator G = 0
Are Sprays On ?
Yes
No
Are Sprays Working ?
No
Yes
Sprays Off
Indicator S = 0
DDR
DDR
Is Skin Temp > Floor Temp ?
Yes
No
Sprays On
Indicator S = 1
DDR



Data Dump Routine
DDR
Is Outside Temp. > 18°C ?
Yes
No
Poke Outside Temp. Into Memory. Poke Indicator Out = 0 Into Memory
Is Outside Temp. < 0°C ?
No
Yes
Poke Outside Temp. into Memory. Poke Indicator Out = 0 Into Memory
Tout = Tout + 100 Poke Outside(Tout) Into Memory. Poke Indicator Out=1 Into Memory
Is Inside avg. Temp.< 10°C ?
No
Yes
Is Inside Temp. < 0°C ?
Yes
Is Inside avg. Temp.< 0°C ?
Yes
No
Poke Inside Temp. Into Memory. Poke Indicator inC = 0 Into Memory
Poke Inside Temp. avg. Into Memory. Poke Indicator inC = 0 Into Memory
Tins = Tins + 100. Poke Inside (Tins) Into Memory. Poke Indicator inC = 1 Into Memory
Is w.b. Temp. < 0°C ?
No
Yes
Poke 0 Into Memory Locations for wet & dry bulb Integer and Decimal Temps.
Poke wet & dry bulb Integer and Decimal Temps. Into Memory
Dump Data In Subroutine
Is Inside Temp.> 28 °C ?
Yes
No
Fan On
Fan Off
Wait 15 min. Every 5 min. Service Interrupt, Otherwise Display Temperatures
Resume From Interrupt
Is Inside Temp.> 28 °C ?
Fan On
Fan Off
Amber Light On
RET

APPENDIX B

BASIC PROGRAM FOR THE SYM-1

```
NEW
REM *****************************************************
REM BASIC PROGRAM TO CONTROL THE OPERATION OF A GREENHOUSE
REM *****************************************************
REM
REM SIXTEEN THERMISTORES ARE POSITIONED
REM IN THE GREENHOUSE TO MONITOR TEMPERATURES.
REM THE THERMISTORS RESISTANCE IS RELATED
REM TO TEMPERATURE.
REM
REM THE FOLLOWING THERMISTORS ARE INSTALLED.
REM T1 IS THE OUTSIDE HIGH TEMPERATURE 15,45 C
REM T2 IS THE BETWEEN SKIN TEMPERATURE 10,50 C.
REM T3 IS THE SPRAY RETURN TEMPERATURE 10,50 C.
REM T4 IS A FLOOR TEMPERATURE 10,21 C.
REM T5 IS A FLOOR TEMPERATURE 10,21 C.
REM T6 FLOOR TEMPERATURE 10,21 C
REM T7 FLOOR TEMPERATURE 10,21 C.
REM T8 DEEP FLOOR TEMPERATURE 10,21 C.
REM T9 FLOOR TEMPERATURE 10,21 C.
REM T10 INSIDE LOW TEMPERATURE 1: 1,7 C.
REM T11 INSIDE LOW TEMPERATURE 2: 1,7 C.
REM T12 INSIDE HIGH TEMPERATURE 15,50 C.
REM T13 WET BULB TEMPERATURE 15,35 C.
REM T14 DRY BULB TEMPERATURE 15,35 C.
REM T15 GAS HEATER RETURN WATER TEMPERATURE 10,38 C.
REM T16 OUTSIDE LOW TEMPERATURE -10,20 C.
REM
REM ALL TEMPERATURES ARE IN DEGREES C.
REM ***************************************
REM      BASIC PROGRAM
REM ***************************************
REM
REM INITIALIZE INPUT AND OUTPUT PORTS.
REM SWITCH OFF ALL OPERATIONS.
REM START REAL TIME CLOCK
REM GO TO ASSEMBLY LANGUAGE SUBROUTINE.
50 X=USR(&"8000",0)
REM
REM INPUT THE START TIME OF OPERATION.
60 POKE 1039,0
70 POKE 1040,0
80 POKE 1041,0
90 POKE 1042,0
REM
REM DIMENSION THREE ARRAYS FOR VARIABLES.
100 DIM D(16),R(16),T(16)
REM
REM READ FROM MEMORY THE FREQUENCY CONVERSION PARAMETERS.
110 FOR I=1 TO 16
120 READ D(I),R(I)
130 NEXT I
REM
REM INITIALIZE PROGRESS INDICATORS .
140 E=0:H=0:S=0
REM
REM VOLTAGE TO FREQUENCY CALIBRATION COEFFICIENTS.
150 S1=4.817685369E+8
160 I1=0.553901115442
REM
REM ROUTINE TO DISPLAY FLOOR TEMPERATURE,WETBULB TEM,DRYBULB
```

```
REM GO TO ASSEMBLY LANGUAGE SUBROUTINE.
170 X=USR(&"BCC1",0)
REM
REM WAIT IN ASSEMBLY LANGUAGE SUBROUTINE FOR SOFTWARE
REM INTERRUPT.
REM GET FREQUENCIES AND PUT INTO ARRAY.
REM CONVERT FREQUENCIES TO TEMPERATURES.
180 FOR I=1 TO 16
190 A=PEEK(1042+I):B=PEEK(1058+I)
200 T(I)=(D(I)-LOG(S1/(2*(B*16*16+A)+I1)))/R(I)
210 NEXT I
REM
REM CALCULATE AVERAGE FLOOR TEMPERATURE.
220 A=(T(4)+T(5)+T(6)+T(7)+T(8)+T(9))/6:B=A+6
REM
REM CALCULATE MINIMUM INSIDE TEMPERATURE.
230 C=(T(10)+T(11))/2
REM
REM HAVE SET THE MINIMUM INSIDE TEMPERATURE TO 4 DEG C.
240 IF C<4 THEN 550
REM
REM CHECK TO SEE IF ALARM ON.
REM IF ON SWITCH OFF ALARM AND ELECTRIC HEATER.
250 IF NOT PEEK(44032)AND 16 THEN 280
260 X=PEEK(44032)AND 231
270 POKE 44032,X:E=0
REM
REM GET MINIMUM TEMPERATURE SET BY SWITCH.
280 P=(15 AND NOT PEEK(43009))*10+(15 AND NOT PEEK(43008))
REM
REM ********************
REM NIGHT MODE OF OPERATION.
REM ********************
REM
REM IS FLOOR TEMPERATURE GREATER THAN SET?.
290 IF A>P THEN 420
REM
REM IS IT THE DAY MODE?.
300 IF NOT PEEK(40960) AND 8 THEN 420
REM
REM NO SO SWITCH OFF SPRAYS.
310 X=PEEK(44032)AND 127
320 POKE 44032,X:S=0
REM
REM IS THE HEATER ON?.
330 IF PEEK(44032)AND 32 THEN 370
REM
REM NO SO SWITCH ON.
340 X=PEEK(44032)OR 32
350 POKE 44032,X:H=1
REM
REM GO TO ROUTINE TO DUMP DATA.
360 GO TO 700
REM
REM IS THE HEATER WORKING?.
370 IF(T(15)-T(3))<3 THEN 390
REM
REM HEATER WORKING SO DUMP DATA.
380 GO TO 700
REM
REM HEATER NOT WORKING SO SWITCH OFF.
390 X=PEEK(44032)AND 223
400 POKE 44032,X:H=0
```

```
REM
REM GO TO DUMP DATA.
410 GO TO 700
REM
REM**************************************************
REM   DAY MODE
REM**************************************************
REM
REM SWITCH OFF HEATER .
420 X=PEEK(44032)AND 223
430 POKE 44032,X:H=0
REM
REM ARE THE SPRAYS ON?.
440 IF NOT PEEK(44032)AND 128 THEN 490
REM
REM SPRAYS ON, SO IS THE RETURN WATER TO THE
REM FLOOR GREATER THAN THE FLOOR AVERAGE?.
450 IF T(3)+2>A THEN 700
REM
REM RETURN TEMPERATURE LESS THAN FLOOR
REM SO SWITCH OFF SPRAYS.
460 X=PEEK(44032)AND 127
470 POKE 44032,X:S=0
REM
REM GO TO DUMP DATA.
480 GO TO 700
REM
REM SPRAYS NOT ON, IS SKIN TEMPERATURE
REM GREATER THAN AVERAGE FLOOR TEMPERATURE?.
490 IF T(2)>B THEN 520
REM
REM NO, SO IS FLOOR TEMPERATURE LESS THAN MINIMUM?.
REM IF SO RETURN TO NIGHT MODE.
500 IF A<P THEN 310
REM
510 GO TO 700
REM
REM SKIN TEMPERATURE IS GREATER THAN FLOOR
REM TEMPERATURE, SO SWITCH ON SPRAYS.
520 X=PEEK(44032)OR 128
530 POKE 44032,X:S=1
REM
REM GO TO DUMP DATA.
540 GO TO 700
REM
REM**************************************************
REM CRITICAL TEMPERATURE OPERATION.
REM**************************************************
REM
REM ARE THE SPRAYS ABLE TO BE USED
REM TO RAISE THE TEMPERATURE IN THE HOUSE?.
REM
550 IF NOT PEEK(40960)AND 16 THEN 620
REM
REM SWITCH OFF GAS HEATER.
560 IF PEEK(44032)AND 128 THEN 640
570 X=PEEK(44032)AND 223
580 POKE 44032,X:H=0
REM
REM SWITCH ON SPRAYS.
590 X=PEEK(44032)OR 128
600 POKE 44032,X:S=1
REM
```

```
REM GO TO DUMP DATA.
610 GO TO 700
REM
REM IN CASE SPRAYS ARE ON, SWITCH OFF
REM SINCE SPECIFIED BY OPERATOR.
620 X=PEEK(44032)AND 127
630 POKE 44032,X:S=0
REM
REM IS THE ELECTRIC HEATER ON?.
640 IF PEEK(44032)AND 16 THEN 680
REM
REM NO SO SWITCH ON.
650 X=PEEK(44032)OR 16
660 POKE 44032,X:E=1
REM
REM GO TO DUMP DATA.
670 GO TO 700
REM
REM CRITICAL CONDITIONS SO SWITCH ON ALARM.
680 X=PEEK(44032)OR 8
690 POKE 44032,X
REM
REM WHEN BASIC PUTS NUMBERS INTO MEMORY FOR
REM ASSEMBLY SUBROUTINES, IT DOES SO IN HEX
REM SO NEGATIVE NUMBERS MUST BE INDICATED.
REM POKE TO MEMORY AN ARRAY OF TEMPERATURES
REM AS WELL AS THE OPERATIONAL INDICATORS,E,H,S
REM CHECK IF OUTSIDE TEMPERATURE GREATER THAN 18 DEGREES C.
700 IF T(1)>18 THEN 730
REM CHECK IF THE OUTSIDE TEMPERATURE LESS THAN ZERO.
705 IF T(16)<0 THEN 800
REM PUT POSITIVE OUTSIDE TEMPERATURES INTO MEMORY.
710 POKE 1043,0:POKE 1044,T(16)
720 GO TO 740
PUT OUTSIDE TEMPERATURE THAT IS GREATER THAN 18 C INTO MEMOR
730 POKE 1043,0:POKE 1044,T(1)
REM CHECK IF INSIDE TEMPERATURES ABOVE ZERO.
740 IF C<10 THEN 840
REM CHECK IF INSIDE HIGH TEMPERATURE THERMISTOR IS BELOW O.
745 IF T(12)<0 THEN 840
REM PUT INTO MEMORY POSITIVE INSIDE HIGH TEMPERATURES.
750 POKE 1045,0:POKE 1046,T(12)
REM CHECK IF WET BULB TEMPERATURE IS BELOW O.
760 IF T(13)<0 THEN 920
REM BREAK THE WET AND DRY BULB TEMPERATURES INTO DOUBLE BYTE
770 A1=INT(T(13))
771 B1=(T(13)-A1)*100
772 POKE 1047,A1
773 POKE 1048,B1
780 A1=INT(T(14))
781 B1=(T(14)-A1)*100
782 POKE 1049,A1
783 POKE 1050,B1
790 GO TO 940
REM ADD A CONSTANT TO MAKE OUTSIDE TEMPERATURE POSITIVE.
800 T(16)=T(16)+100
810 POKE 1043,1
820 POKE 1044,T(16)
830 GO TO 740
REM AGAIN CHECK IF INSIDE MINIMUM TEMPERATURE BELOW O.
840 IF C<0 THEN 880
REM PUT INTO MEMORY POSITIVE INSIDE MINIMUM TEMPERATURES.
850 POKE 1045,0
```

```
860 POKE 1046,C
870 GO TO 760
REM ADD A CONSTANT TO MAKE THE INSIDE MINIMUM POSITIVE.
880 C=C+100
890 POKE 1045,1
900 POKE 1046,C
910 GO TO 760
REM SINCE WET BULB BELOW 0 PUT ZERO INTO MEMORIES.
920 POKE 1047,0
930 POKE 1048,0
941 POKE 1049,0
942 POKE 1050,0
REM PUT CONTROLLER OPERATIONS INTO MEMORY.
940 POKE 1051,H
950 POKE 1052,S
960 POKE 1053,E
970 POKE 1054,P
REM PUT INTO MEMORY THE AVERAGE FLOOR TEMPERATURE.
975 POKE 1055,A
REM
REM GO TO DUMP TO TAPE ROUTINE.
980 X=USR(&"BDC9",0)
REM
REM IS THE INSIDE TEMPERATURE TOO HIGH?.
990 IF T(12)>28 THEN 1030
REM
REM SWITCH OFF THE FAN IF ON
1000 X=PEEK(44032)AND 251
1010 POKE 44032,X
1020 GO TO 1050
REM
REM SWITCH ON THE FAN.
1030 X=PEEK(44032)OR 4
1040 POKE 44032,X
REM
REM EVERY FIVE MINUTES CHECK THE TEMPERATURE
REM INSIDE AND THE OPERATION OF THE FAN.
1050 FOR K=1 TO 2
REM
REM DISPLAY FLOOR TEMPERATURE.
REM DISPLAY WETBULB TEMPERATURE.
REM DISPLAY DRY BULB TEMPERATURE.
1060 X=USR(%"BCC1",0)
REM
REM CALCULATE THE INSIDE HIGH TEMPERATURE.
1070 A=PEEK(1042+12):B=PEEK(1058+12)
1080 T(12)=(D(12)-LOG(S1/(2*(B*16*16+A)+I1)))/R(12)
REM
REM IS THE INSIDE HIGH TEMPERATURE
REM TOO HIGH?
1090 IF T(12)>28 THEN 1130
REM
REM SWITCH OFF FAN.
1100 X=PEEK(44032)AND 251
1110 POKE 44032,X
1120 GO TO 1150
REM
REM SWITCH ON FAN.
1130 X=PEEK(44032)OR 4
1140 POKE 44032,X
REM ON THE SECOND INTERRUPT
REM SWITCH ON THE AMBER LIGHT.
REM THIS INDICATES 5 MINUTES BEFORE
```

```
REM CALCULATIONS BEGIN.
1150 IF K=1 THEN 1180
1160 X=PEEK(40960) AND 249
1170 POKE 40960,X
1180 NEXT K
1190 GO TO 170
REM
REM DATA OF THE CALIBRATION OF THE THERMISTORS.
1200 DATA 13.386899,0.04631907
1210 DATA 13.412297,0.04600793
1220 DATA 13.386431,0.04700407
1230 DATA 13.567614,0.05025128
1240 DATA 13.412648,0.05012332
1250 DATA 13.473366,0.04995454
1260 DATA 13.376838,0.04986896
1270 DATA 13.312866,0.04974919
1280 DATA 13.603675,0.05035371
1290 DATA 13.383542,0.05291910
1300 DATA 13.402487,0.05392877
1310 DATA 13.472727,0.04464357
1320 DATA 13.333096,0.04663342
1330 DATA 13.512867,0.04689527
1340 DATA 13.445212,0.04732307
1350 DATA 13.578326,0.05060959
1360 END
```

APPENDIX C

ASSEMBLY LANGUAGE SUBROUTINE

FOR THE SYM-1

```
PASS 1

PASS 2

------------------------------------------------------------------------------
               ASSEMBLY LANGUAGE PROGRAMS FOR GREENHOUSE OPERATION.

------------------------------------------------------------------------------

0000           ;PROGRAMS WRITTEN BY SIMON MURRAY
0000           ;OF THE OKLAHOMA STATE UNIVERSITY

------------------------------------------------------------------------------

0000           ;SUPERMON SUBROUTINES AND LOCATIONS.

0000           TAB    =$8C29          ;SEVEN SEGMENT CODE
0000           DISBUF =$A640          ;DISPLAY BUFFER
0000           SCAND  =$8906          ;ROUTINE TO DISPLAY BUFFER
0000           ACCESS =$8B86          ;ROUTINE TO ACCESS RAM
0000           NACC   =$8B9C          ;WRITE PROTECT SYM RAM.
0000           DUMPT  =$8E87          ;ROUTINE TO DUMP DATA TO TAPE
0000           IRQL   =$FFFE          ;INTERRUPT LOW BYTE VECTOR
0000           IRQH   =$FFFF          ;INTERRUPT HIGH BYTE VECTOR

------------------------------------------------------------------------------

0000           ;SYSTEM LOCATIONS

0000           ;DATADL AND DATADH ARE THE STARTING AND
0000           ;ENDING ADDRESS OF DATA TO TAPE.
0000           ;CHANGE DATADL AND DATADH DEPENDING
0000           ;UPON YOUR PARTICULAR APPLICATION.

0000           DATADL =$040F          ;LOW MEMORY DATA FOR TAPE.
0000           DATADH =$041F          ;HIGH MEMORY DATA TO TAPE.
0000           PBD    =$A000          ;PORT B DATA
0000           PAD    =$A001          ;PORT A DATA
0000           PBDD   =$A002          ;PORT B DATA DIRECTION
0000           PADD   =$A003          ;PORT A DATA DIRECTION
0000           TPOUT  =$A402          ;TAPE DIRECTION OUT
0000           TAPDEL =$A630          ;HS TAPE DELAY
0000           EAL    =$A64A          ;ENDING ADDRESS LOW FOR TAPE
0000           EAH    =$A64B          ;ENDING ADDRESS HIGH FOR TAPE
0000           SAL    =$A64C          ;STARTING ADDRESS LOW FOR TAPE
0000           SAH    =$A64D          ;STARTING ADDRESS HIGH FOR TAPE
0000           ID     =$A64E          ;TAPE IDENTIFICATION

0000           PBADD  =$A802          ;AUXILIARY PORT B DATA DIRECTION
0000           PAADD  =$A803          ;AUXILIARY PORT A DATA DIRECTION
0000           RT1CH  =$A805          ;REAL TIMER ONE HIGH COUNTER
0000           RT1CL  =$A806          ;REAL TIMER ONE LOW COUNTER
0000           RT2CL  =$A808          ;REAL TIMER TWO LOW COUNTER
0000           RT2CH  =$A809          ;REAL TIMER TWO HIGH COUNTER
0000           RACR   =$A80B          ;REAL TIME AUXILIARY CONTROL REG.
0000           RIER   =$A80E          ;REAL TIME INTERRUPT ENABLE REG.
```

```
0000               APBD    =$AC00         :AUXILIARY PORT B DATA
0000               APBDD   =$AC02         :AUXILIARY PORT B DATA
0000               ;DIRECTION
0000               FT1L    =$AC04         :FREQUENCY TIMER LATCH LOW
0000               FT1CH   =$AC05         :FREQUENCY TIMER HIGH COUNTER
0000               FT1CL   =$AC06         :FREQUENCY TIMER ONE LOW COUNTER
0000               FT2CL   =$AC08         :FREQUENCY TIMER LOW COUNTER
0000               FT2CH   =$AC09         :FREQUENCY TIMER TWO HIGH COUNTER
0000               FACR    =$AC0B         :AUXILIARY CONTROL
0000               ;REGISTER
0000               PCR     =$AC0C         :TAPE CONTROL SWITCH
0000               FIFR    =$AC0D         :FREQUENCY INTERRUPT FLAG REG.
0000               FIER    =$AC0E         :FREQUENCY ENABLE REGISTER
0000               MODE    =$00FD         :TAPE DATA MODE, SYM OR KIM.

------------------------------------------------------------------------------

0000               ;SYSTEM VARIABLES

0000                       *=$0400

0400               MS      *=*+1          :MULTIPLEXER SELECTION.
0401               FLAG    *=*+1          :INTERRUPT OCCURANCE INDICATION.
0402               COUNTL  *=*+1          :FREQUENCY COUNT LOW.
0403               COUNTH  *=*+1          :FREQUENCY COUNT HIGH.
0404               SECONT  *=*+1          :0.5 SECONDS FREQUENCY COUNTING.
0405               TEMPT   *=*+1          :TEMPORARY MEMORY FOR HEX/DEC.
0406               FLORAV  *=*+3          :TEMPERATURES DISPLAYED.
0409               DEC1    *=*+3          :DECIMAL MEMORY FOR DISPLAY.
040C               TDEC    *=*+3          :TEMPORARY MEMORY DEC. DISPLAY.
040F               MIN     *=*+1          :MINUTES OF OPERATION.
0410               HRS     *=*+1          :HOURS OF OPERATION.
0411               DAY     *=*+1          :DAYS OF OPERATION.
0412               MON     *=*+1          :MONTHS OF OPERATION.
0413               TEMPL   *=*+16         :LOW FREQUENCY COUNTS.
0423               TEMPH   *=*+16         :HIGH FREQUENCY COUNTS.

------------------------------------------------------------------------------

0433                       *=$8C00

------------------------------------------------------------------------------
                   INITIALIZATION PROGRAM

------------------------------------------------------------------------------

8C00               ;THE FIRST PROGRAM INITIALIZES
8C00               :THE INPUT OUTPUT PORTS AND STARTS THE TIMERS
8C00               ;LED'S ARE CONNECTED TO PORT B
8C00               :MULTIPLEXER IS CONNECTED TO PORT A
8C00               ;MINIMUM TEMPERATURE SWITCHES ARE CONNECTED TO
8C00               ;AUXILIARY PORTS A AND B

8C00   A9 BF                 LDA #$BF
8C02   8D 02 AC              STA APBDD     :ININIALIZE CONTROL PORTS
8C05   8D 03 A0              STA PADD      :INITIALIZE MULTIPLEXER SELECT
8C08   8D 01 A0              STA PAD

8C0B   A9 00                 LDA #$00
8C0D   8D 00 AC              STA APBD      :SWITCH OFF THE CONTROLLERS.
```

```
BC10   8D 02 A8         STA PBADD        ;TEMPERATURE SWITCH LOW DECIMAL.
BC13   8D 03 A8         STA PAADD        ;TEMPERATURE SWITCH HIGH DECIMAL.
BC16   8D 00 A0         STA PBD          ;INITIALIZE THE DATA TO THE LED'S
BC19   8D 01 04         STA FLAG         ;INDICATE NO INTERRUPTS

BC1C   A9 07            LDA #$07
BC1E   8D 02 A0         STA PBDD         ;HALF OF PORT B IS FOR LED'S.

BC21                 ;FOR THE FIRST DISPLAY, DISPLAY THESE VALUES
BC21                 ;FOR FLOOR WETBULB AND DRYBULB TEMPERATURES

BC21   A9 16            LDA #$16         ;FOR THE FIRST 5 MINUTES DISPLAY
BC23   8D 06 04         STA FLORAV       ;AVERAGE FLOOR TEMPERATURE.
BC26   8D 07 04         STA FLORAV+1     ;WET BULB TEMPERATURE.
BC29   8D 08 04         STA FLORAV+2     ;DRY BULB TEMPERATURE.

BC2C                 ;SET UP TIMER FOR INTERRUPT EVERY FIVE MINUTES
BC2C                 ;TIMER T2 IS DECRIMENTED EVERY SECOND TIME OUT
BC2C                 ;OF TIMER T1

BC2C   A9 E0            LDA #$E0         ;TIMER #1 PRODUCE SQUARE WAVE.
BC2E   8D 0B A8         STA RACR         ;TIMER #2 DECRIMENTED BY TIMER #1
BC31   A9 A0            LDA #$A0         ;ONLY TIMER #2 TO CAUSE INTERRUPT
BC33   8D 0E A8         STA RIER         ;INTERRUPT OCCURS IF T2 OUT.
BC36   A9 C4            LDA #$C4         ;SET TIMER #2.
BC38   8D 08 A8         STA RT2CL        ;LOW BYTE HAS 196 DECIMAL.
BC3B   A9 09            LDA #$09
BC3D   8D 09 A8         STA RT2CH        ;HIGH BYTE HAS 9 DECIMAL.

BC40   A9 6A            LDA #$6A         ;INITIALIZE TIMER #1.
BC42   8D 06 A8         STA RT1CL        ;LOW BYTE HAS 106 DECIMAL.
BC45   A9 EA            LDA #$EA
BC47   8D 05 A8         STA RT1CH        ;HIGH BYTE HAS 234 DECIMAL.

BC4A                 ;SET UP INTERRUPT VECTORS

BC4A   20 86 8B         JSR ACCESS       ;ACCESS SYM RAM.
BC4D   A9 61            LDA #<IRQ
BC4F   8D FE FF         STA IRQL         ;SET INTERRUPT VECTOR LOW.
BC52   A9 BC            LDA #>IRQ
BC54   8D FF FF         STA IRQH         ;SET INTERRUPT VECTOR HIGH.
BC57   20 9C 8B         JSR NACC         ;WRITE PROTECT RAM.
BC5A   58               CLI              ;CLEAR INTERRUPT FLAG IF SET.

BC5B                 ;FINISHED INITIALIZATION SO SWITCH ON
BC5B                 ;GREEN LED AND GO TO BASIC ROUTINE

BC5B   A9 03            LDA #$03         ;LIGHTS ON, PULL I/O LINE LOW.
BC5D   8D 00 A0         STA PBD
BC60   60               RTS              ;RETURN TO BASIC.

---------------------------------------------------------------------------
                     INTERRUPT ROUTINE

---------------------------------------------------------------------------

BC61                 ;SAVE THE REGISTERS ONTO THE STACK.
BC61   48            IRQ   PHA           ;SAVE THE ACCUMULATOR.
BC62   8A                  TXA           ;SAVE X REGISTER.
BC63   48                  PHA
BC64   98                  TYA           ;SAVE THE Y REGISTER.
BC65   48                  PHA
```

```
BC66                    ;SWITCH ON THE RED LED TO INDICATE CALCULATIONS IN
BC66                    ;OPERATION

BC66   A9 06                  LDA #$06
BC68   8D 00 A0               STA PBD

BC6B                    ;RESET TIMER 2 SINCE TIMER 1 IS IN CONTINUOUS MODE.

BC6B   A9 C4                  LDA #$C4
BC6D   8D 08 A8               STA RT2CL         :RESET TIMER TWO
BC70   A9 09                  LDA #$09
BC72   8D 09 A8               STA RT2CH
BC75   18                     CLC               :CLEAR CARRY FLAG.
BC76   D8                     CLD               :HEXADECIMAL MODE ONLY.

BC77                    ;UPDATE THE REAL TIME CLOCK

BC77   AD 0F 04               LDA MIN           :LOAD MINUTES.
BC7A   69 05                  ADC #$05          :ADD 5 MINUTES.
BC7C   8D 0F 04               STA MIN           :STORE IN MINUTES.
BC7F   C9 3C                  CMP #$3C          :IS HOUR UP?
BC81   D0 22                  BNE TIMET         :IF NOT GO AROUND.

BC83   A9 00                  LDA #$00          :RESET MINUTES TO ZERO.
BC85   8D 0F 04               STA MIN
BC88   18                     CLC
BC89   AD 10 04               LDA HRS           :LOAD HOURS.
BC8C   69 01                  ADC #$01          :ADD ONE HOUR.
BC8E   8D 10 04               STA HRS           :STORE HOURS.
BC91   C9 18                  CMP #$18          :IS DAY UP?
BC93   D0 20                  BNE TIMET         :IF NOT GO AROUND.

BC95   A9 00                  LDA #00           :RESET DAYS.
BC97   8D 10 04               STA HRS
BC9A   18                     CLC
BC9B   AD 11 04               LDA DAY           :LOAD DAYS.
BC9E   69 01                  ADC #$01          :ADD ONE DAY TO TOTAL.
BCA0   8D 11 04               STA DAY           :STORE IN DAYS.
BCA3   C9 1E                  CMP #$1E          :IS MONTH OF 30 DAYS UP?
BCA5   D0 0E                  BNE TIMET -       :IF NOT GO AROUND.

BCA7   A9 00                  LDA #$00          :RESET DAYS.
BCA9   8D 11 04               STA DAY
BCAC   18                     CLC
BCAD   AD 12 04               LDA MON           :LOAD MONTHS.
BCB0   69 01                  ADC #$01          :ADD ONE MONTH.
BCB2   8D 12 04               STA MON           :STORE IN MONTH.
BCB5   58               TIMET CLI               :CLEAR INTERRUPT FLAG.

BCB6                    :SET UP A SOFTWARE FLAG TO INDICATE INTERRUPT OCCURANCE.

BCB6   A9 01                  LDA #$01
BCB8   8D 01 04               STA FLAG          :SET THE USER INTERRUPT FLAG.

BCBB                    :RESTORE ALL REGISTERS.

BCBB   68                     PLA               :RESTORE THE Y REGISTER.
BCBC   A8                     TAY
BCBD   68                     PLA               :RESTORE THE X REGISTER.
BCBE   AA                     TAX
BCBF   68                     PLA               :RESTORE ACCUMULATOR.
BCC0   40                     RTI               :RETURN FROM INTERRUPT.
```

HEXADECIMAL TO DECIMAL CONVERSION

```
BCC1                    ;SUBPROGRAM TO CONVERT THE AVERAGE FLOOR TEMPERATURE
BCC1                    ;THE WET AND DRY BULB TEMPERATURES TO DECIMAL TO DISPLAY

BCC1   A9 00     HEXDEC LDA #$00        ;INITIALIZE COUNTER.
BCC3   A8        MORE   TAY
BCC4   B9 06 04         LDA FLORAV,Y    ;LOAD VALUES TO BE CONVERTED.
BCC7   8D 05 04         STA TEMFT       ;STORE INTEMPORARY STORAGE.
BCCA   A9 00            LDA #$00
BCCC   99 09 04         STA DEC1,Y      ;CLEAR DECIMAL LOCATION.

BCCF   AD 05 04         LDA TEMFT       ;GET HEXIDECIMAL VALUE.
BCD2   29 0F            AND #$0F        ;MASK HIGH NIBBLE LEAVING LOW.
BCD4   F0 0E            BEQ OVER        ;BRANCH TO 16'S IF ZERO.
BCD6   AA               TAX             ;TRANSFER LOW NIBBLE TO COUNTER.
BCD7   18               CLC
BCD8   F8               SED             ;SET DECIMAL MODE.

BCD9   B9 09 04  RPT1   LDA DEC1,Y      ;GET DECIMAL CONTENTS.
BCDC   69 01            ADC #01         ;ADD ONE.
BCDE   99 09 04         STA DEC1,Y      ;RESULT IN DECIMAL.
BCE1   CA               DEX
BCE2   D0 F5            BNE RPT1

BCE4   AD 05 04  OVER   LDA TEMFT       ;GET HEXIDECIMAL VALUE.
BCE7   4A               LSR A           ;SHIFT RIGHT FOUR TIMES
BCE8   4A               LSR A           ;TO GET HIGH NIBBLE IN
BCE9   4A               LSR A           ;LOW ORDER NIBBLE.
BCEA   4A               LSR A
BCEB   F0 0D            BEQ FINISH      ;STOP CONVERSION IF ZERO.

BCED   AA               TAX
BCEE   18               CLC
BCEF   B9 09 04  RPT2   LDA DEC1,Y      ;GET DECIMAL VALUE AND ADD 16
BCF2   69 16            ADC #$16        ;ADD 16 TO THE MEMORY.
BCF4   99 09 04         STA DEC1,Y      ;STORE THE DECIMAL RESULT.
BCF7   CA               DEX
BCF8   D0 F5            BNE RPT2

BCFA   18        FINISH CLC             ;FINISHED CONVERSION.
BCFB   98               TYA
BCFC   69 01            ADC #01         ;COMPLETE THREE CONVERSIONS.
BCFE   C9 03            CMP #03         ;OF FLOOR TEMPERATURE,WETBULB
BD00   D0 C1            BNE MORE        ;AND DRYBULB TEMPERATURE.

BD02   D8               CLD
BD03   A9 03            LDA #$03        ;SWITCH ON THE GREEN LIGHT.
BD05   8D 00 A0         STA PBD
```

DISPLAY ROUTINE

```
BD08                    ;ROUTINE TO DISPLAY THE DECIMAL VALUES JUST CALCULATED
```

```
BD08   AD 09 04    DIS    LDA DEC1        :TEMPORARY STORE DECIMAL.
BD0B   8D 0C 04           STA TDEC
BD0E   AD 0A 04           LDA DEC1+1
BD11   8D 0D 04           STA TDEC+1
BD14   AD 0B 04           LDA DEC1+2
BD17   8D 0E 04           STA TDEC+2
BD1A   78                 SEI             :PREVENT INTERRUPT FROM HAPPENING

BD1B   20 86 8B           JSR ACCESS      :ACCESS RAM.
BD1E   A2 05              LDX #$05        :INITIALIZE X INDEX.
BD20   AD 0C 04    PRR    LDA TDEC        :GET FIRST VALUE.
BD23   29 0F              AND #$0F        :MASK HIGH NIBBLE.
BD25   A8                 TAY

BD26   B9 29 8C           LDA TAB,Y       :GET SEVEN SEGMENT DISPLAY.
BD29   9D 40 A6           STA DISBUF,X    :STORE IN DISPLAY BUFFER.
BD2C   A0 04              LDY #$04

BD2E   4E 0E 04    DGL    LSR TDEC+2      :SHIFT LAST VALUE INTO CARRY.
BD31   6E 0D 04           ROR TDEC+1      :CARRY INTO SECOND VALUE
BD34   6E 0C 04           ROR TDEC        :CARRY INTO FIRST VALUE.
BD37   88                 DEY             :SHIFT UNTIL ONE NIBBLE SHIFTED.
BD38   D0 F4              BNE DGL

BD3A   CA                 DEX
BD3B   10 E3              BPL PRR
BD3D   A2 FF              LDX #$FF        :INITIALIZE X FOR TIMING LOOP.

BD3F   8A          JAT    TXA             :SAVE X
BD40   48                 PHA
BD41   20 06 89           JSR SCAND       :JUMP TO SCAN SUBROUTINE OF SYM
BD44   68                 PLA             :GET X BACK.
BD45   AA                 TAX
BD46   CA                 DEX             :DECRIMENT X FOR TIMING
BD47   D0 F6              BNE JAT
BD49   58                 CLI             :CLEAR THE INTERRUPT FLAG.

BD4A   AD 01 04           LDA FLAG        :BRANCH ONLY IF FLAG CLEAR.
BD4D   F0 B9              BEQ DIS
BD4F   A9 00              LDA #$00
BD51   8D 01 04           STA FLAG        :CLEAR THE INTERRUPT FLAG.

--------------------------------------------------------------------------
                   FREQUENCY COUNTING OF THERMISTORS.

--------------------------------------------------------------------------

BD54                      :SWITCH ON THE MULTIPLEXER.

BD54   A9 1F              LDA #$1F
BD56   8D 03 A0           STA PADD
BD59   A9 00              LDA #$00        :INITIALIZE THE MULTIPLEXER.
BD5B   8D 00 04           STA MS
BD5E   A2 60              LDX #$60        :SET UP A DELAY TO ALLOW THE
BD60   CA          DELAY  DEX             :MULTIPLEXER TO STAPILIZE.
BD61   D0 FD              BNE DELAY

BD63                      :SELECT ONE THERMISTOR AT A TIME UNTIL ALL DONE.

BD63   AD 00 04    NEXT   LDA MS          :SELECT THE THERMISTORS
BD66   8D 01 A0           STA PAD
BD69   A9 60              LDA #$60        :TIMER #2 DECRIMENT FROM FFFF.
```

```
BD6B   8D 0B AC          STA FACR
BD6E   A9 0A             LDA #$0A        :TIMER #1 RUNS FOR 0.05 SECONDS.
BD70   8D 04 04          STA SECONT

BD73   A9 80             LDA #$80
BD75   8D 0E AC          STA FIER        :CLEAR THE INTERRUPT ENABLE
BD78   A9 58             LDA #$58
BD7A   8D 06 AC          STA FT1CL       :INITIALIZE T1 TIMER LOW COUNTER
BD7D   A9 FF             LDA #$FF
BD7F   8D 08 AC          STA FT2CL       :INITIALIZE THE T2 TIMER
BD82   8D 09 AC          STA FT2CH

BD85   A9 C3             LDA #$C3        :LOAD HIGH BYTE OF TIMER #1.
BD87   8D 05 AC          STA FT1CH
BD8A   A9 40      DOWN   LDA #$40
BD8C   2C 0D AC   WAIT   BIT FIFR        :WAIT UNTIL T1 TIMED OUT.
BD8F   F0 FB             BEQ WAIT        :WAIT FOR INTERRUPT FLAG SET.

BD91   AD 04 AC          LDA FT1L        :CLEAR INTERRUPT FLAG RESET T1.
BD94   CE 04 04          DEC SECONT      :DECRIMENT TIME COUNTER
BD97   D0 F1             BNE DOWN        :WAIT FOR 0.5 SECONDS.

BD99   AD 08 AC          LDA FT2CL       :GET THE DECRIMENTED VALUES AND
BD9C   49 FF             EOR #$FF        :CONVERT TO COUNT
BD9E   8D 02 04          STA COUNTL      :STORE IN MEMORY.
BDA1   AD 09 AC          LDA FT2CH       :REPEAT FOR HIGH BYTE.
BDA4   49 FF             EOR #$FF
BDA6   8D 03 04          STA COUNTH

BDA9   AD 00 04          LDA MS          :INTITIALIZE TABLE INDEX.
BDAC   AA                TAX

BDAD               :ROUTINE TO STORE INDIVIDUAL FREQUENCIES IN
BDAD               :A MEMORY STACK FOR FURTHER ANALYSIS

BDAD   AD 02 04          LDA COUNTL      :STORE LOW FREQUENCIES IN MEMORY.
BDB0   9D 13 04          STA TEMPL,X
BDB3   AD 03 04          LDA COUNTH      :STORE HIGH FREQUENCIES IN MEMORY
BDB6   9D 23 04          STA TEMPH,X

BDB9   EE 00 04          INC MS          :INCRIMENT THERMISTOR SELECTION.
BDBC   AD 00 04          LDA MS          :LOAD THE THERMISTOR SELECTION.
BDBF   C9 10             CMP #$10        :COMPARE TO 16 IN DECIMAL.
BDC1   D0 A0             BNE NEXT        :GET NEXT IF NOT ALL DONE.

BDC3               :SWITCH OFF MULTIPLEXER AFTER USE

BDC3   A9 10             LDA #$10
BDC5   8D 01 A0          STA PAD
BDC8   60                RTS             :RETURN TO BASIC.

------------------------------------------------------------------------------
                   ROUTINE TO DUMP DATA TO TAPE

------------------------------------------------------------------------------

BDC9               :ROUTINE TO DUMP DATA TO TAPE
BDC9               :USING KIM SPEED COMPATABLE TO SYM AND AIM

BDC9   20 86 8B   TAPED  JSR ACCESS
BDCC   A9 80             LDA #$80
BDCE   85 FD             STA MODE        :SET KIM SPEED
```

```
BDD0    A9 04               LDA #$04
BDD2    8D 30 A6            STA TAPDEL       ;SET THE TAPE GAP
BDD5    A9 48               LDA #$48
BDD7    8D 02 A4            STA TPOUT        ;DIRECT THE DATA TO TAPE
BDD9    A9 01               LDA #$01
BDDC    8D 4E A6            STA ID           ;IDENTIFICATION IS SET TO ONE

BDDF    A9 0F               LDA #<DATADL     ;LOAD LOW BYTE OF DATA
BDE1    8D 4C A6            STA SAL          ;TO BE PUT ON TAPE.
BDE4    A9 04               LDA #>DATADL     ;LOAD THE HIGH BYTE OF DATA
BDE6    8D 4D A6            STA SAH          ;TO BE PUT ON TAPE.
BDE9    A9 20               LDA #<DATADH+1   ;LOAD BYTE OF TOP OF DATA.
BDEB    8D 4A A6            STA EAL
BDEE    A9 04               LDA #>DATADH     ;LOAD HIGH BYTE OF TOP
BDF0    8D 4B A6            STA EAH          ;DATA TO BE DUMPED.

BDF3    20 37 BE            JSR DUMPT        ;JUMP TO THE DUMP ROUTINE
BDF6    A9 CC               LDA #$CC
BDF8    8D 0C AC            STA PCR          ;STOP THE TAPE RECORDER.

BDFB    AD 17 04            LDA DATADL+8     ;LOAD WET BULB TEMPERATURE.
BDFE    8D 06 04            STA FLORAV       ;STORE IN DISPLAY MEMORY.
BE01    AD 19 04            LDA DATADL+10    ;LOAD DRY BULB TEMPERATURE.
BE04    8D 07 04            STA FLORAV+1
BE07    AD 1E 04            LDA DATADL+15    ;LOAD AVERAGE FLOOR TEMPERATURE.
BE0A    8D 08 04            STA FLORAV+2

BE0D                 ;CHECK IF TAPE TO BE REMOVED.

BE0D    A9 20               LDA #$20         ;CHECK TO SEE IF PIN NO 5.
BE0F    2C 00 A0            BIT PBD          ;ON PORT B IS ACTIVE.
BE12    F0 0D               BEQ ROUND        ;IF NOT BRANCH AROUND.

BE14                 ;TAPE TO BE REMOVED SO PUT INDICATION ON END OF TAPE.

BE14    A9 11               LDA #$11         ;LOAD THE DECIMAL 17.
BE16    8D 0F 04            STA DATADL       ;STORE IN START OF DUMP MEMORY.

BE19    20 37 BE            JSR DUMPT        ;DUMP TO TAPE AGAIN.
BE1C    A9 CC               LDA #$CC
BE1E    8D 0C AC            STA PCR          ;STOP TAPE RECORDER.
BE21    60           ROUND  RTS              ;RETURN TO BASIC.
------------------------------------------------------------------------------

BE22                        .END
        ERRORS= 0000
```

APPENDIX D

BASIC DRIVER PROGRAM

```
PASS 1

PASS 2

--------------------------------------------------------------------------
                  BASIC DRIVER PROGRAM FOR THE SYM-1
--------------------------------------------------------------------------

0000                 ;WRITTEN BY SIMON MURRAY FOR THE SYM-1

0000                 ;FOR THE OKLAHOMA STATE UNIVERSITY.
--------------------------------------------------------------------------

0000                 ;SYSTEM LOCATIONS

0000                 MEMORY =$BF00         ;POSITION WHERE ZERO PAGE RESIDES
0000                 STORE  =$0000         ;STORAGE SPACE.
0000                 ASSEMB =$C5A0         ;LOCATION IN BASIC.
0000                 RUNC   =$C46D         ;RUN THE BASIC PROGRAM.
--------------------------------------------------------------------------

0000                 ;RELOCATION PROGRAM.
--------------------------------------------------------------------------

0000                        *=$B000

B000   A2 00                LDX #$00       ;CLEAR COUNTER.
B002   BD 00 BF     NEXT    LDA MEMORY,X   ;LOAD MEMORY LOCATION.
B005   95 00                STA STORE,X    ;STORE IN ZERO PAGE.
B007   E8                   INX            ;INCREMENT COUNTER.
B008   E0 F0                CPX #$F0       ;HAS BLOCK BEEN MOVED?
B00A   D0 F6                BNE NEXT       ;WAIT UNTIL ALL HAS BEEN MOVED.
B00C   A9 C5                LDA #>ASSEMB   ;GET HIGH BYTE OF ASSEMB.
B00E   48                   PHA            ;PUSH ONTO STACK.
B00F   A9 A0                LDA #<ASSEMB   ;GET LOW BYTE OF ASSEMB.
B011   48                   PHA            ;PUSH ONTO STACK.
B012   4C 6D C4             JMP RUNC       ;START RUNNING BASIC.
--------------------------------------------------------------------------

B015                        .END
       ERRORS= 0000
```

APPENDIX E

ZERO PAGE RELOCATOR FOR RUNNING BASIC

```
PASS 1

PASS 2

---------------------------------------------------------------------------
                    PROGRAM TO PLACE ZERO PAGE INTO MEMORY FOR BASIC.

0000                ;WRITTEN BY SIMON MURRAY
0000                ;FOR THE OKLAHOMA STATE UNIVERSITY.

---------------------------------------------------------------------------

0000                ;SYSTEM LOCATIONS

0000                MEMORY =$BF00           ;MEMORY FOR ZERO PAGE.
0000                START  =$0000           ;STARTING ADDRESS OF ZERO PAGE.
0000                MONIT  =$8003           ;SYM MONITOR ROUTINE.

0000                       *=$0200

0200  A2 00                LDX #00          ;INITIALIZE COUNTER.
0202  B5 00         NEXT   LDA START,X      ;LOAD INDIVIDUAL DATA.
0204  9D 00 BF             STA MEMORY,X     ;STORE IN MEMORY.
0207  E8                   INX              ;INCREMENT COUNTER.
0208  E0 F0                CPX #$F0         ;DATA ALL MOVED?
020A  D0 F6                BNE NEXT         ;IF NOT GO AROUND.
020C  4C 03 80             JMP MONIT        ;JUMP TO MONITOR ROUTINE.

---------------------------------------------------------------------------

020F                       .END
      ERRORS= 0000
```

APPENDIX F

AIM-65 TO TEKTRONIX DATA TRANSFER PROGRAM

```
PASS 1

PASS 2

-----------------------------------------------------------------------------
                    TRANSLOCATION PROGRAM.
-----------------------------------------------------------------------------

0000                ;WRITTEN BY SIMON MURRAY

-----------------------------------------------------------------------------

0000                ;SYSTEM LOCATIONS

0000                MEMORY =$0400          ;DATA STORAGE LOCATION.
0000                TSPEED =$A408          ;TAPE SPEED (C7,5B,5A)
0000                INFLG  =$A412          ;INPUT DEVICE.
0000                NAME   =$A432          ;FILE NAME.
0000                TAPIN  =$A434          ;IN FLAG (TAPE 1 OR 2)
0000                DRB    =$A800          ;DATA REG. B

0000                ;MONITOR ROUTINES.

0000                LOADK1 =$E3A4          ;LOAD ROUTINE WITH KIM FORMAT.
0000                LL     =$E3FE          ;SET I,O TO TERMINAL (KB,D,P,T)
0000                OUTPUT =$E97A          ;OUTPUTS CHARACTER TO TTY.
0000                TAISET =$EDEA          ;SET TAPE (1 OR 2) FOR INPUT.

-----------------------------------------------------------------------------

0000                ;PROGRAM RESIDES ON THE TEKTRONIX TAPE.
0000                ;IT IS OPERATED UNDER TEKTRONIX CONTROL.

-----------------------------------------------------------------------------

0000                        *=$0200

-----------------------------------------------------------------------------
                    OUTPUT DATA FROM AIM TO TTY.
-----------------------------------------------------------------------------

0200   D8           START   CLD            ;HEXIDECIMAL MODE.
0201   18                   CLC            ;CLEAR CARRY FLAG.
0202   A2 00                LDX #$00
0204   BD 00 04     REST    LDA MEMORY,X   ;LOAD DATA FROM MEMORY.
0207   69 21                ADC #$21
0209                ;HAVE TO ADD A CONSTANT TO ALL VALUES
0209                ;SO THAT THE TEKTRONIX DOESN'T BECOME TIED UP
0209                ;WITH CONTROL CHARACTERS.

0209   20 7A E9             JSR OUTPUT     ;OUTPUT THE CHARACTERTO TTY.
020C   A9 0D                LDA #$0D       ;LOAD A CARRIAGE RETURN.
020E   20 7A E9             JSR OUTPUT     ;OUTPUT THE CARRIAGE RETURN.

0211   18                   CLC            ;RECLEAR THE CARRY FLAG.
```

```
0212   E8                 INX             :INCREMENT THE COUNTER.
0213   E0 11              CPX #$11        :HAVE 17 CHARACTERS.
0215   D0 ED              BNE REST        :COMPLETE THE TRANSFER.

----------------------------------------------------------------------------
                     DATA FROM TAPE IN KIM FORMAT.
----------------------------------------------------------------------------

0217   A9 02              LDA #>BACK      :HIGH BYTE OF RETURN.
0219   48                 PHA             :ONTO STACK.
021A   A9 39              LDA #<BACK      :LOW BYTE OF RETURN.
021C   48                 PHA             :ONTO STACK.

021D   A9 BC              LDA #$BC        :START THE TAPE RECORDER.
021F   8D 00 A8           STA DRB
0222   A9 5A              LDA #$5A        :LOAD THE LETTER 'K'
0224   8D 08 A4           STA TSPEED      :STORE IN TAPE SPEED.

0227   A9 00              LDA #$00        :INPUT DEVICE.
0229   8D 34 A4           STA TAPIN       :TAPE IS THE DEVICE.
022C   A9 4B              LDA #$4B        :DEVICE I,O DIRECTION.
022E   8D 12 A4           STA INFLG       :SET FLAG FOR INPUT.

0231   20 EA ED           JSR TAISET      :SET UP FOR INPUT.
0234   A9 01              LDA #$01        :01 IS THE FILE ID.
0236   8D 32 A4           STA NAME
0239   20 A4 E3    BACK   JSR LOADK1      :GET A BLOCK OF DATA.
023C   20 FE E3           JSR LL          :RESET I,O DEVICE.

023F   A9 AC              LDA #$AC        :SWITCH OFF THE TAPE.
0241   8D 00 A8           STA DRB
0244   00                 BRK             :STOP THE PROGRAM

----------------------------------------------------------------------------

0245                      .PAGE
       ERRORS= 0000
```

APPENDIX G

TEKTRONIX BASIC PROGRAM

```
100 PAGE
110 PRI "***************************************************************"
120 PRINT "____"
130 PRINT "                 PROGRAM WRITTEN BY                              "
140 PRINT "__"
150 PRI "                    SIMON MURRAY                                   "
160 PRINT "__"
170 PRINT "                         FOR THE AGRICULTURAL ENGINEERING DEPT."
180 PRINT "_"
190 PRINT "                            OKLAHOMA STATE UNIVERSITY"
200 PRINT "___"
210 PRI "***************************************************************"
220 CALL "WAIT",4
230 REM*************************************************************
240 REM THIS PROGRAM TRANSFERS DATA FROM THE AIM TO THE TEKTRONIX.
250 PRINT @32,26:2
260 PAGE
270 PRINT ""
280 PRINT "___"
290 PRINT "_"
300 PRI "WHAT WAS THE PROCESSOR STARTING TIME IN MINUTES,HOUR,DAY,MONTH"
310 INPUT M,H,D,M1
320 PRINT ""
330 REM*************************************************************
340 REM TITLES FOR THE ARRAYS
350 A$="MONTH"
360 B$="DAY"
370 C$="HOUR"
380 D$="MINUTES"
390 F$="OUTSIDE"
400 G$="INSIDE"
410 H$="WET BULB"
420 I$="DRY BULB"
430 J$="GAS HEATER"
440 L$="SPRAYS"
450 N$="ELECTRIC HEATER"
460 O$="TEMP"
470 P$=" ON"
480 Q$="OFF"
490 R$="SET TEMP"
500 T$="FLOOR TEMP"
510 N=200
520 DELETE T0,T1,T2,T3,T4,T5,T6,T7,T8,T9,O1,O2,O3,R1,F3
530 DIM T0(N),T1(N),T2(N),T3(N),T4(N),T5(N),T6(N),T7(N),T8(N),T9(N)
540 DIM O1(N),O2(N),O3(N),R1(N),F3(8)
550 REM INITIALIZE PRINTER AND PRINT COLUMN HEADINGS.
560 PRINT @41.11:1
570 PRINT @41: USING 590:A$,B$,C$,D$,F$,G$,R$,T$,H$
580 PRINT @41: USING 600:I$,J$,L$,N$
590 IMAGE(5A,2X,3A,2X,4A,2X,7A,2X,7A,2X,6A,2X,8A,2X,10A,2X,8A,2X),S
600 IMAGE(8A,2X,10A,2X,8A,2X,15A)
610 PRINT @41: USING 620:O$,O$,O$,O$
620 IMAGE(29X,4A,4X,4A,27X,4A,6X,4A)
630 PRINT @41:"_"
640 PAGE
650 REM*************************************************************
660 REM INSTRUCTION SET FOR CONNECTION OF AIM-65 TO THE TEKTRONIX.
670 PRINT "_"
680 PRINT "TURN ON THE AIM AND INSERT THE DATA TAPE INTO THE RECORDER"
690 PRINT ""
700 PRINT "PRESS RESET ON THE AIM"
710 PRINT ""
720 PRINT "SWITCH THE TTY KEY ON THE AIM TO TTY"
```

```
730 PRINT ""
740 PRINT "AGAIN PRESS RESET ON THE AIM"
750 PRINT ""
760 PRINT "PRESS RETURN ON TEKTRONIX WHEN COMPLETE"
770 INPUT W$
780 PAGE
790 PRINT "TRANSFERRING PROGRAM TO THE AIM TO GET THE DATA FROM TAPE"
800 REM CHR(127) IS THE RUBOUT KEY TO ESTABLISH BAUD RATE.
810 REM CHR(13) IS RETURN
820 REM FILE 3 ON THE TEKTRONIX HAS THE ASSEMBLY PROGRAM
830 FIND 3
840 W$=CHR(127)
850 S$=CHR(13)
860 CALL "RATE",1200,2,2
870 PRINT @40:W$
880 CALL "WAIT",0.5
890 REM*****************************************************************
900 REM STARTING POSITION FOR AIM PROGRAM
910 M$="0200"
920 PRINT @40:"*"
930 CALL "WAIT",0.2
940 FOR I=1 TO LEN(M$)
950 K$=SEG(M$,I,1)
960 PRINT @40:K$
970 CALL "WAIT",0.2
980 NEXT I
990 PRINT @40:S$
1000 CALL "WAIT",0.5
1010 REM*************************************************************
1020 REM START TRANSFERRING THE PROGRAM ACROSS.
1030 PRINT @40:"I"
1040 CALL "WAIT",0.5
1050 READ @33:M$
1060 ON EOF (0) THEN 1200
1070 FOR I=1 TO LEN(M$)
1080 K$=SEG(M$,I,1)
1090 IF I=4 THEN 1110
1100 PRINT @40:K$
1110 CALL "WAIT",0.4
1120 NEXT I
1130 REM**************************************************************
1140 REM CHECK FOR AUTOMATIC COMMANDS SUCH AS INCREMENT X COUNTER, INX
1150 IF LEN(M$)=3 THEN 1050
1160 S$=CHR(13)
1170 PRINT @40:S$
1180 CALL "WAIT",0.5
1190 GO TO 1050
1200 E$=CHR(27)
1210 PRINT @40:E$
1220 CALL "WAIT",0.4
1230 REM**************************************************************
1240 REM READY TO GET FIRST BLOCK OF DATA OFF TAPE.
1250 PAGE
1260 PRINT "_"
1270 PRINT "_"
1280 PRINT "PRESS PLAY ON THE TAPE RECORDER"
1290 PRINT ""
1300 PRINT "PRESS RETURN ON THE TEKTRONIX"
1310 INPUT W$
1320 PRINT @40:"*"
1330 CALL "WAIT",0.2
1340 REM**************************************************************
1350 REM ADDRESS OF THE PROGRAM TO GET DATA OFF TAPE
```

```
1360 M$="0217"
1370 FOR I=1 TO LEN(M$)
1380 K$=SEG(M$,I,1)
1390 PRINT @40:K$
1400 CALL "WAIT",0.2
1410 NEXT I
1420 PRINT @40:S$
1430 CALL "WAIT",0.2
1440 PRINT @40:"G"
1450 PRINT @40,30:
1460 CALL "WAIT",0.2
1470 PRINT @40:S$
1480 REM*************************************************************
1490 REM WAIT UNTIL BLOCK HAS BEEN PUT INTO MEMORY.
1500 FOR I=1 TO 3
1510 INPUT @40:M$
1520 PRINT M$
1530 NEXT I
1540 REM*************************************************************
1550 REM THE FIRST BLOCK OF DATA HAS BEEN TRANSFERRED FROM TAPE TO AIM.
1560 CALL "WAIT",0.5
1570 T0(1)=M
1580 T1(1)=H
1590 T2(1)=D
1600 T3(1)=M1
1610 PAGE
1620 REM*************************************************************
1630 REM THIS IS THE BEGINNING OF THE TRANSFERRAL AND PROCESSING.
1640 REM SETTING LOOP TO 200 AS SPECIAL INDICATORS ON THE TAPE WILL
1650 REM STOP THE DATA TRANSFER.
1660 B5=200
1670 B7=0
1680 FOR J=1 TO B5
1690 PRINT @40:"*"
1700 CALL "WAIT",0.2
1710 M$="0200"
1720 FOR I=1 TO LEN(M$)
1730 K$=SEG(M$,I,1)
1740 PRINT @40:K$
1750 CALL "WAIT",0.2
1760 NEXT I
1770 PRINT @40:S$
1780 CALL "WAIT",0.2
1790 PRINT @40:"G"
1800 PRINT @40,30:
1810 CALL "WAIT",0.2
1820 PRINT @40:S$
1830 REM*************************************************************
1840 REM INPUT THE DATA FROM THE INPUT OUTPUT PORT
1850 FOR I=1 TO 18
1860 INPUT @40:M$
1870 IF I<12 THEN 2010
1880 IF I>2 THEN 1910
1890 K$=SEG(M$,I,1)
1900 M$=K$
1910 A=ASC(M$)
1920 A=A-33
1930 GO TO I OF 2010,2630,2720,2840,2960,3300,3380,3450,3520,3590,3610
1940 IF I=12 THEN 3670
1950 IF I=13 THEN 3690
1960 IF I=14 THEN 3750
1970 IF I=15 THEN 3790
1980 IF I=16 THEN 3830
```

```
1990 IF I=17 THEN 3870
2000 IF I=18 THEN 3910
2010 NEXT I
2020 REM***************************************************************
2030 REM WAIT UNTIL THE AIM HAS ACQUIRED ANOTHER BLOCK OF DATA
2040 FOR I=1 TO 2
2050 INPUT @40:M$
2060 CALL "WAIT",0.2
2070 NEXT I
2080 PRINT @40,30:
2090 NEXT J
2100 REM***************************************************************
2110 B7=B7+J
2120 PRINT "DO YOU HAVE ANOTHER DATA TAPE TO PROCESS (1=YES 0=NO)"
2130 INPUT B6
2140 IF B6=1 THEN 2170
2150 B5=200
2160 GO TO 1680
2170 N=B7
2180 REM***************************************************************
2190 REM PRINT OUT THE DATA IN COLUMNS
2200 FOR J=1 TO N
2210 GOSUB 3940
2220 PRINT @41: USING 2240:T3(J),T2(J),T1(J),T0(J),T4(J),T5(J),T8(J)
2230 PRINT @41: USING 2250:T9(J),T6(J),T7(J),U$,V$,Z$
2240 IMAGE(2X,2D,4X,2D,4X,2D,7X,2D,5X,4D,5X,3D,5X,3D),S
2250 IMAGE(7X,3D,8X,3D.2D,6X,3D.2D,6X,3A,3X,3A,9X,3A)
2260 NEXT J
2270 GOSUB 4210
2280 PRINT "___"
2290 PAGE
2300 PRINT "PLEASE PRESS FORM_FEED ON THE PRINTER"
2310 PRINT "PRESS RETURN ON THE TEKTRONIX WHEN COMPLETE"
2320 INPUT W$
2330 PRINT "YOU ARE NOW ABOUT TO TRANSFER DATA FROM THE TEKTRONIX"
2340 PRINT "TO THE IBM"
2350 PRINT "THIS WILL NEARLY ALWAYS BE REQUIRED TO BE DONE IN THE"
2360 PRINT "LATE AFTERNOON WHEN THE COMPUTER USE IS LESS"
2370 PRINT "_"
2380 PRINT "REMOVE THE RS-232 CABLE FROM THE AIM AND CONNECT IT TO THE"
2390 PRINT "PLOTTER BACK, PRESS RETURN ON THE TEKTRONIX WHEN FINISHED"
2400 INPUT W$
2410 PRINT "SWITCH ON THE POWER AT THE BACK OF THE MODEM  RACAL-VADIC"
2420 PRINT "PRESS RETURN ON THE TEKTRONIX WHEN COMPLETE"
2430 INPUT W$
2440 PAGE
2450 PRINT "___"
2460 PRINT "WHEN THE I/O BUSY LIGHTS STOP FLASHING,TYPE IN LOGON"
2470 PRINT @41:"DIAL 7600 ON THE TELEPHONE AND INSTALL IN MODEM & WHEN";
2480 PRINT @41:" YOU HEAR THE BEEP"
2490 PRI @41:"TYPE IN LOGON AND THE IBM WILL PROMPT YOU WITH QUESTIONS";
2500 PRINT @41:" OPEN UP A DATA FILE CALLED GREEN.DATA"
2510 PRINT @41:"WHEN THE IBM COMES BACK WITH 0010 PRESS THE FOLLOWING "
2520 PRINT @41:"RETURN TO BASIC";
2530 PRINT @41:"  THEN RUN 4080"
2540 PRINT @41:"___"
2550 CALL "RATE",1200,2,2
2560 REM***************************************************************
2570 REM UPDATE THE FIRST OCCURANCE OF MINUTES FROM DATA.
2580 CALL "TERMIN"
2590 REM***************************************************************
2600 REM T0,T1,T2,T3 ARE THE MINUTES, HOURS, DAY, MONTH RESPESTIVELY
2610 REM THIS SECTION UPDATES A CLOCK BASED ON 15 MINUTE INTERRUPTS
```

```
2620 REM FOR J>1 THEN UPDATE DEPENDS UPON THE 15 MINUTES.
2630 IF J>1 THEN 2010
2640 REM******************************************************************
2650 REM UPDATE THE FIRST OCCURANCE OF HOURS FROM DATA.
2660 E=0
2670 T0(1)=M+A
2680 IF T0(1)<60 THEN 2010
2690 T0(1)=T0(1)-60
2700 E=1
2710 GO TO 2010
2720 IF J>1 THEN 2010
2730 IF E=1 THEN 2780
2740 T1(1)=A+H
2750 GO TO 2800
2760 REM******************************************************************
2770 REM UPDATE THE FIRST OCCURANCE OF DAY FROM DATA.
2780 T1(1)=H+A+E
2790 E=0
2800 IF T1(1)<24 THEN 2010
2810 T1(1)=T1(1)-24
2820 E=1
2830 GO TO 2010
2840 IF J>1 THEN 2010
2850 IF E=1 THEN 2900
2860 T2(1)=D+A
2870 GO TO 2920
2880 REM******************************************************************
2890 REM UPDATE THE FIRST OCCURANCE OF MONTH FROM DATA.
2900 T2(1)=D+A+E
2910 E=0
2920 IF T2(1)<31 THEN 2010
2930 T2(1)=T2(1)-30
2940 E=1
2950 GO TO 2010
2960 IF J>1 THEN 3070
2970 IF E=1 THEN 3020
2980 T3(1)=A+M1
2990 REM******************************************************************
3000 REM CONSECUTATIVELY UPDATE THE SOFTWARE CLOCK ON THE TEKTRONIX.
3010 GO TO 3040
3020 T3(1)=A+M+E
3030 E=0
3040 IF T3(1)<13 THEN 2010
3050 T3(1)=T3(1)-12
3060 GO TO 2010
3070 T0(J)=T0(J-1)+15
3080 E=0
3090 IF T0(J)<60 THEN 3120
3100 T0(J)=T0(J)-60
3110 E=1
3120 T1(J)=T1(J-1)+E
3130 E=0
3140 IF T1(J)<24 THEN 3170
3150 T1(J)=T1(J)-24
3160 E=1
3170 T2(J)=T2(J-1)+E
3180 E=0
3190 IF T2(J)<31 THEN 3230
3200 REM******************************************************************
3210 T2(J)=T2(J)-30
3220 E=1
3230 T3(J)=T3(J-1)+E
3240 IF T3(J)<13 THEN 2010
```

```
3250 T3(J)=T3(J)-12
3260 GO TO 2010
3270 REM**************************************************************
3280 REM INDICATOR FOR OUTSIDE TEMPERATURES EITHER BEING POSITIVE OR NOT
3290 REM IF A=17 THEN THE END OF THE DATA TAPE HAS BEEN REACHED.
3300 IF A=17 THEN 2120
3310 IF A=1 THEN 3340
3320 C=0
3330 GO TO 2010
3340 C=A
3350 GO TO 2010
3360 REM**************************************************************
3370 REM OUTSIDE TEMPERATURE
3380 IF C=0 THEN 3410
3390 T4(J)=A-100
3400 GO TO 2010
3410 T4(J)=A
3420 GO TO 2010
3430 REM**************************************************************
3440 REM INDICATOR FOR INSIDE TEMPERATURES BEING EITHER POSITIVE OR NOT
3450 IF A=1 THEN 3480
3460 C=0
3470 GO TO 2010
3480 C=A
3490 GO TO 2010
3500 REM**************************************************************
3510 REM INSIDE TEMPERATURE
3520 IF C=0 THEN 3550
3530 T5(J)=100-A
3540 GO TO 2010
3550 T5(J)=A
3560 GO TO 2010
3570 REM**************************************************************
3580 REM WET BULB TEMPERATURE
3590 T6(J)=A
3600 GO TO 2010
3610 REM**************************************************************
3620 REM FRACTION PART OF WET BULB TEMPERATURE.
3630 T6(J)=T6(J)+A/100
3640 GO TO 2010
3650 REM**************************************************************
3660 REM DRY BULB TEMPERATURE
3670 T7(J)=A
3680 GO TO 2010
3690 REM**************************************************************
3700 REM FRACTION PART OF DRY BULB TEMPERATURE.
3710 T7(J)=T7(J)+A/100
3720 GO TO 2010
3730 REM**************************************************************
3740 REM GAS HEATER OPERATION
3750 O1(J)=A
3760 GO TO 2010
3770 REM**************************************************************
3780 REM SPRAY OPERATION
3790 O2(J)=A
3800 GO TO 2010
3810 REM**************************************************************
3820 REM ELECTRIC HEATER OPERATION
3830 O3(J)=A
3840 GO TO 2010
3850 REM**************************************************************
3860 REM SET POINT TEMPERATURE
3870 T8(J)=A
```

```
3880 GO TO 2010
3890 REM*****************************************************************
3900 REM AVERAGE FLOOR TEMPERATURE
3910 T9(J)=A
3920 GO TO 2010
3930 T6(J)=A
3940 IF O1(J)=1 THEN 3970
3950 U$=O$
3960 GO TO 3980
3970 U$=P$
3980 IF O2(J)=1 THEN 4020
3990 V$=O$
4000 GO TO 4030
4010 REM*****************************************************************
4020 V$=P$
4030 IF O3(J)=1 THEN 4060
4040 Z$=O$
4050 GO TO 4070
4060 Z$=P$
4070 RETURN
4080 REM*****************************************************************
4090 REM TRANSFER THE DATA TO THE IBM MAINFRAME
4100 FOR I=1 TO N
4110 PRINT @40: USING 4130:T3(I),T2(I),T1(I),T0(I),T4(I),T5(I),R1(I)
4120 PRINT @40: USING 4140:T8(I),T9(I),O1(I),O2(I),O3(I)
4130 IMAGE(2X,2D,2X,2D,2X,2D,2X,2D,2X,4D,2X,4D,2X,3D.1D),S
4140 IMAGE(2X,3D,2X,3D,2X,1D,2X,1D,2X,1D)
4150 CALL "WAIT",0.4
4160 NEXT I
4170 PRINT "AFTER THIS MESSAGE PLEASE PRESS.RETURN TO GET BACK TO IBM"
4180 CALL "TERMIN"
4190 STOP
4200 REM*****************************************************************
4210 REM CALCULATION OF RELATIVE HUMIDITY
4220 FOR I=1 TO 8
4230 READ F3(I)
4240 NEXT I
4250 DATA -741.9242,-29.721,-11.55286,-0.8685635,0.1094098,0.439993
4260 DATA 0.2520658,0.05218684
4270 P2=0.96454
4280 REM*****************************************************************
4290 REM P2 IS THE ATMOSPHERIC PRESSURE
4300 FOR J=1 TO N
4310 P1=0
4320 X=0
4330 F1=T7(J)+273.15
4340 FOR I=1 TO 8
4350 X=F3(I)*(0.65-0.01*T7(J))^(I-1)
4360 P1=P1+X
4370 NEXT I
4380 Y=P1*(273.16-T7(J))*(0.01/F1)
4390 P1=EXP(Y)*217.99
4400 W2=0.62198*(P1/(P2-P1))
4410 W1=((1093-0.556*T6(J))*W2-0.24*(T7(J)-T6(J))
4420 W1=W1/(1093+0.444*T7(J)-T6(J))
4430 U=W1/W2
4440 R1(J)=U/(1-(1-U)*(P1/P2))*100
4450 NEXT J
4460 DELETE T6,T7
4470 RETURN
```

APPENDIX H

STATISTICAL ANALYSIS SYSTEM PROGRAM USED TO ANALYZE GREENHOUSE DATA

```
1                                         S T A T I S T I C A L   A N A L Y S I S   S Y S T E
NOTE: THE JOB MURRAY HAS BEEN RUN UNDER RELEASE 79.5 OF SAS AT OKLAHOMA STATE UNIVERSITY (004

NOTE: SAS OPTIONS SPECIFIED ARE:
       SORT=4

1          DATA NEW;                                                                     00000110
2          INFILE USE;                                                                   00000120
3          INPUT MONTH DAY HOUR MINUTES OUTSIDE INSIDE HUMIDITY SET FLOOR                00000130
4          HEATER SPRAYS ELECTRIC;                                                       00000140

NOTE: INFILE USE IS:
      DSNAME=U11557A.GREEN.DATA,
      UNIT=SYSDA,VOL=SER=DISD80,DISP=SHR,
      DCB=(BLKSIZE=6160,LRECL=80,RECFM=FB)

NOTE: 92 LINES WERE READ FROM INFILE USE.
NOTE: DATA SET WORK.NEW HAS 92 OBSERVATIONS AND 12 VARIABLES. 190 OBS/TRK.
NOTE: THE DATA STATEMENT USED 0.24 SECONDS AND 172K.

5          DATA NEW1;SET NEW;                                                            00000150
6          JULIAN=MONTH*30+DAY+HOUR/24+MINUTES/(60*24);                                  00000160
7          DROP MONTH DAY HOUR MINUTES;                                                  00000170

NOTE: DATA SET WORK.NEW1 HAS 92 OBSERVATIONS AND 9 VARIABLES. 250 OBS/TRK.
NOTE: THE DATA STATEMENT USED 0.16 SECONDS AND 180K.

8          DATA NEW2;SET NEW;                                                            00000180
9          TIME=MINUTES/60+HOUR;                                                         00000190
10         DROP MONTH DAY HOUR MINUTES;                                                  00000200

NOTE: DATA SET WORK.NEW2 HAS 92 OBSERVATIONS AND 9 VARIABLES. 250 OBS/TRK.
NOTE: THE DATA STATEMENT USED 0.15 SECONDS AND 180K.

11         PROC MEANS;                                                                   00000210
12         VAR SET FLOOR;                                                                00000220
13         TITLE COMPARISON OF THE FLOOR TEMPERATURE TO SET TEMPERATURE;                 00000230

NOTE: THE PROCEDURE MEANS USED 0.22 SECONDS AND 172K AND PRINTED PAGE 1.

14         PROC PLOT;                                                                    00000240
15         PLOT HUMIDITY*TIME='H';                                                       00000250
16         LABEL HUMIDITY=PERCENTAGE;                                                    00000260
17         LABEL TIME=TIME IN HOURS FOR ONE DAY PERIOD;                                  00000270
18         TITLE RELATIVE HUMIDITY IN PERCENT.;                                          00000280

NOTE: THE PROCEDURE PLOT USED 0.27 SECONDS AND 180K AND PRINTED PAGE 2.

19         PROC PLOT;                                                                    00000290
20         PLOT SET*TIME='S';                                                            00000300
21         LABEL SET=TEMPERATURE IN DEGREES C;                                           00000310
22         LABEL TIME=TIME IN HOURS FOR ONE DAY PERIOD;                                  00000320
23         TITLE FLOOR SET TEMPERATURE.;                                                 00000330

NOTE: THE PROCEDURE PLOT USED 0.26 SECONDS AND 180K AND PRINTED PAGE 3.
```

```
24          PROC PLOT;                                                          00000340
25          PLOT FLOOR*TIME='F';                                                00000350
26          TITLE FLOOR TEMPERATURES.;                                          00000360

NOTE: THE PROCEDURE PLOT USED 0.26 SECONDS AND 180K AND PRINTED PAGE 4.

27          PROC PLOT;                                                          00000370
28          PLOT HEATER*TIME='H';                                               00000380
29          LABEL HEATER=CONTROLLER OPERATION ;                                 00000390
30          LABEL TIME=EXPERIMENT DURATION;                                     00000400
31          TITLE HEATER OPERATION FOR ONE DAY PERIOD.;                         00000410

NOTE: THE PROCEDURE PLOT USED 0.28 SECONDS AND 180K AND PRINTED PAGE 5.

32          PROC PLOT;                                                          00000420
33          PLOT SPRAY*TIME='S';                                                00000430
34          TITLE SPRAY OPERATION FOR A ONE DAY PERIOD;                         00000440

NOTE: THE PROCEDURE PLOT USED 0.27 SECONDS AND 180K AND PRINTED PAGE 6.

35          PROC PLOT;                                                          00000450
36          PLOT ELECTRIC*TIME='E';                                             00000460
37          TITLE ELECTRIC HEATER OPERATION FOR A ONE DAY PERIOD.;              00000470

NOTE: THE PROCEDURE PLOT USED 0.27 SECONDS AND 180K AND PRINTED PAGE 7.

38          DATA _NULL_;SET NEW1;                                               00000480
39          FILE PERM;                                                          00000490
40          PUT JULIAN OUTSIDE INSIDE HUMIDITY SET FLOOR HEATER SPRAYS ELECTRIC;  00000500

NOTE: FILE PERM IS:
      DSNAME=U11557A.PERMANET.DATA,
      UNIT=SYSDA,VOL=SER=DASD40,DISP=NEW,
      DCB=(BLKSIZE=3200,LRECL=80,RECFM=FB)

NOTE: THE VARIABLE JULIAN IS UNINITIALIZED.
NOTE: 92 LINES WERE WRITTEN TO FILE PERM.
NOTE: THE DATA STATEMENT USED 0.29 SECONDS AND 180K.

41          PROC PRINT DATA=NEW;                                                00000510

NOTE: THE PROCEDURE PRINT USED 0.43 SECONDS AND 172K AND PRINTED PAGES 8 TO 9.

NOTE: SAS USED 180K MEMORY.

NOTE: SAS INSTITUTE INC.
      SAS CIRCLE
      BOX 8000
      CARY, N.C. 27511
```

APPENDIX I

GREENHOUSE DATA

TABLE III

GREENHOUSE DATA

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	22	17	20	8	18	15	19	11	14	OFF	OFF	OFF
1	22	17	35	10	13	15	20	9	12	OFF	OFF	OFF
1	22	17	50	9	11	15	20	9	11	OFF	OFF	OFF
1	22	18	5	10	10	15	20	8	11	OFF	OFF	OFF
1	22	18	20	9	10	15	20	8	10	OFF	OFF	OFF
1	22	18	35	9	9	15	20	8	10	OFF	OFF	OFF
1	22	18	50	9	9	15	20	8	10	OFF	OFF	OFF
1	22	19	5	9	9	15	20	7	9	OFF	OFF	OFF
1	22	19	20	7	9	15	20	6	8	OFF	OFF	OFF
1	22	19	35	7	9	15	20	6	8	OFF	OFF	OFF
1	22	19	50	7	9	15	20	6	8	OFF	OFF	OFF
1	22	20	5	7	8	15	20	6	8	OFF	OFF	OFF
1	22	20	20	7	8	15	20	6	8	OFF	OFF	OFF
1	22	20	35	7	9	15	19	6	8	OFF	OFF	OFF
1	22	20	50	6	8	15	19	6	7	OFF	OFF	OFF
1	22	21	5	5	8	15	19	5	7	OFF	OFF	OFF
1	22	21	20	5	8	15	19	5	7	OFF	OFF	OFF
1	22	21	35	6	8	15	19	5	7	OFF	OFF	OFF
1	22	21	50	5	7	15	19	4	5	OFF	OFF	OFF
1	22	22	5	4	7	15	19	4	5	OFF	OFF	OFF
1	22	22	20	4	7	15	19	4	5	OFF	OFF	OFF
1	22	22	35	4	7	15	19	4	6	OFF	OFF	OFF
1	22	22	50	4	7	15	18	4	6	OFF	OFF	OFF
1	22	23	5	2	6	15	18	3	4	OFF	OFF	OFF
1	22	23	20	3	6	15	18	3	5	OFF	OFF	OFF
1	22	23	35	2	6	15	18	2	4	OFF	OFF	OFF
1	22	23	50	2	6	15	18	2	4	OFF	OFF	OFF
1	23	0	5	3	6	15	18	3	4	OFF	OFF	OFF
1	23	0	20	3	6	15	18	3	4	OFF	OFF	OFF
1	23	0	35	2	6	15	18	2	4	OFF	OFF	OFF
1	23	0	50	2	5	15	18	2	3	OFF	OFF	OFF
1	23	1	5	2	5	15	18	2	4	OFF	OFF	OFF
1	23	1	20	2	5	15	17	2	3	OFF	OFF	OFF
1	23	1	35	2	5	15	17	2	3	OFF	OFF	OFF
1	23	1	50	1	5	15	17	2	4	OFF	OFF	OFF
1	23	2	5	2	5	15	17	3	4	OFF	OFF	OFF

TABLE III (Continued)

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	23	2	20	2	5	15	17	2	4	OFF	OFF	OFF
1	23	2	35	1	5	15	17	2	3	OFF	OFF	OFF
1	23	2	50	1	5	15	17	2	3	OFF	OFF	OFF
1	23	3	5	1	5	15	17	2	3	OFF	OFF	OFF
1	23	3	20	0	5	15	17	2	4	OFF	OFF	OFF
1	23	3	35	1	5	15	17	2	4	OFF	OFF	OFF
1	23	3	50	1	5	15	16	2	4	OFF	OFF	OFF
1	23	4	5	1	5	15	16	3	4	OFF	OFF	OFF
1	23	4	20	1	5	15	16	3	5	OFF	OFF	OFF
1	23	4	35	1	5	15	16	3	4	OFF	OFF	OFF
1	23	4	50	1	4	15	16	2	4	OFF	OFF	OFF
1	23	5	5	1	4	15	16	2	3	OFF	OFF	OFF
1	23	5	20	1	4	15	16	1	1	OFF	OFF	OFF
1	23	5	35	0	4	15	16	1	3	OFF	OFF	OFF
1	23	5	50	0	4	15	16	2	4	OFF	OFF	OFF
1	23	6	5	0	4	15	16	1	3	OFF	OFF	OFF
1	23	6	20	1	4	15	16	2	4	OFF	OFF	OFF
1	23	6	35	1	4	15	15	2	3	OFF	OFF	OFF
1	23	6	50	1	4	15	15	2	3	OFF	OFF	OFF
1	23	7	5	2	4	15	15	2	4	OFF	OFF	OFF
1	23	7	20	2	4	15	15	2	4	OFF	OFF	OFF
1	23	7	35	2	5	15	15	2	4	OFF	OFF	OFF
1	23	7	50	2	5	15	15	3	4	OFF	OFF	OFF
1	23	8	5	2	6	15	15	3	4	OFF	OFF	OFF
1	23	8	20	1	6	15	15	2	4	OFF	OFF	OFF
1	23	8	35	1	9	15	15	4	6	OFF	OFF	OFF
1	23	8	50	2	9	15	15	6	7	OFF	OFF	OFF
1	23	9	5	3	11	15	15	7	10	OFF	OFF	OFF
1	23	9	20	4	12	15	15	8	10	OFF	OFF	OFF
1	23	9	35	4	14	15	15	9	13	OFF	OFF	OFF
1	23	9	50	6	15	15	15	10	14	OFF	OFF	OFF
1	23	10	5	6	17	15	15	11	15	OFF	OFF	OFF
1	23	10	20	4	17	15	15	11	14	OFF	OFF	OFF
1	23	10	35	6	19	15	15	12	17	OFF	OFF	OFF
1	23	10	50	5	20	15	15	12	17	OFF	OFF	OFF
1	23	11	5	6	21	15	15	13	18	OFF	OFF	OFF
1	23	11	20	6	21	15	15	13	18	OFF	OFF	OFF

TABLE III (Continued)

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	23	11	35	7	22	15	15	14	18	OFF	OFF	OFF
1	23	11	50	8	22	15	15	15	19	OFF	ON	OFF
1	23	12	5	9	20	15	15	15	20	OFF	OFF	OFF
1	23	12	20	10	21	15	16	16	20	OFF	OFF	OFF
1	23	12	35	11	22	15	16	16	21	OFF	OFF	OFF
1	23	12	50	12	22	15	16	16	21	OFF	OFF	OFF
1	23	13	5	13	22	15	16	16	21	OFF	OFF	OFF
1	23	20	13	23	15	15	16	17	22	OFF	OFF	OFF
1	23	13	35	14	24	15	16	18	22	OFF	OFF	OFF
1	23	13	50	12	24	15	16	18	23	OFF	OFF	OFF
1	23	14	5	14	24	15	16	18	23	OFF	ON	OFF
1	23	14	20	14	23	15	16	18	22	OFF	OFF	OFF
1	23	14	35	12	24	15	16	17	23	OFF	OFF	OFF
1	23	14	50	14	24	15	16	18	23	OFF	OFF	OFF
1	23	15	5	14	24	15	16	19	24	OFF	OFF	OFF
1	23	15	20	13	24	15	16	19	24	OFF	OFF	OFF
1	23	15	35	15	24	15	16	19	23	OFF	OFF	OFF
1	23	15	50	16	24	15	17	19	23	OFF	OFF	OFF
1	23	16	5	16	23	15	17	18	22	OFF	OFF	OFF
1	23	16	20	16	21	15	17	17	21	OFF	OFF	OFF

APPENDIX J

GREENHOUSE DATA

TABLE IV

GREENHOUSE DATA

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	23	17	20	17	12	18	16	12	13	ON	OFF	OFF
1	23	17	35	16	12	18	16	12	12	ON	OFF	OFF
1	23	17	50	16	10	18	17	11	11	ON	OFF	OFF
1	23	18	5	16	11	18	17	11	11	ON	OFF	OFF
1	23	18	20	15	10	18	17	10	10	ON	OFF	OFF
1	23	18	35	14	9	18	18	10	9	OFF	OFF	OFF
1	23	18	50	14	9	18	18	9	9	OFF	OFF	OFF
1	23	19	5	13	8	18	18	8	8	OFF	OFF	OFF
1	23	19	20	13	7	18	18	8	8	OFF	OFF	OFF
1	23	19	35	13	7	18	18	7	7	OFF	OFF	OFF
1	23	19	50	12	6	18	18	7	7	OFF	OFF	OFF
1	23	20	5	12	6	18	18	6	6	OFF	OFF	OFF
1	23	20	20	12	6	18	18	6	6	OFF	OFF	OFF
1	23	20	35	11	5	18	18	6	6	OFF	OFF	OFF
1	23	20	50	11	5	18	18	5	5	OFF	OFF	OFF
1	23	21	5	10	5	18	18	5	5	OFF	OFF	OFF
1	23	21	20	10	5	18	18	5	5	OFF	OFF	OFF
1	23	21	35	9	4	18	18	4	4	OFF	OFF	OFF
1	23	21	50	9	4	18	18	4	4	OFF	OFF	OFF
1	23	22	5	9	4	18	18	4	4	OFF	OFF	OFF
1	23	22	20	8	4	18	18	4	3	OFF	OFF	OFF
1	23	22	35	8	4	18	18	4	4	OFF	OFF	OFF
1	23	22	50	7	3	18	18	4	3	OFF	ON	OFF
1	23	23	5	7	3	18	18	3	3	OFF	ON	OFF
1	23	23	20	7	5	18	18	5	5	OFF	OFF	OFF
1	23	23	35	7	5	18	18	5	5	OFF	OFF	OFF
1	23	23	50	7	5	18	18	5	5	OFF	OFF	OFF
1	24	0	5	6	5	18	18	5	5	OFF	OFF	OFF
1	24	0	20	6	5	18	18	4	5	OFF	OFF	OFF
1	24	0	35	6	5	18	18	4	5	OFF	OFF	OFF
1	24	0	50	6	5	18	18	4	5	OFF	OFF	OFF
1	24	1	5	6	5	18	18	4	4	OFF	OFF	OFF
1	24	1	20	6	5	18	18	4	4	OFF	OFF	OFF
1	24	1	35	6	5	18	18	4	4	OFF	OFF	OFF
1	24	1	50	6	5	18	18	4	4	OFF	OFF	OFF

TABLE IV (Continued)

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	24	2	5	6	4	18	18	4	4	OFF	OFF	OFF
1	24	2	20	6	4	18	18	4	4	OFF	OFF	OFF
1	24	2	35	6	4	18	18	4	4	OFF	OFF	OFF
1	24	2	50	5	4	18	18	4	4	OFF	OFF	OFF
1	24	3	5	4	4	18	18	4	4	OFF	OFF	OFF
1	24	3	20	5	4	18	18	4	4	OFF	OFF	OFF
1	24	3	35	5	4	18	18	3	4	OFF	OFF	OFF
1	24	3	50	5	4	18	18	3	3	OFF	OFF	OFF
1	24	4	5	5	4	18	18	3	3	OFF	OFF	OFF
1	24	4	20	4	4	18	18	3	4	OFF	OFF	OFF
1	24	4	35	4	4	18	18	3	3	OFF	OFF	OFF
1	24	4	50	5	4	18	18	3	3	OFF	OFF	OFF
1	24	5	5	5	4	18	18	3	3	OFF	OFF	OFF
1	24	5	20	5	4	18	18	3	3	OFF	OFF	OFF
1	24	5	35	5	4	18	18	3	3	OFF	OFF	OFF
1	24	5	50	5	4	18	18	3	3	OFF	OFF	OFF
1	24	6	5	5	4	18	18	3	3	OFF	OFF	OFF
1	24	6	20	5	4	18	18	3	3	OFF	OFF	OFF
1	24	6	35	5	3	18	18	3	3	OFF	ON	OFF
1	24	6	50	4	5	18	18	3	3	OFF	OFF	OFF
1	24	7	5	4	4	18	18	3	3	OFF	OFF	OFF
1	24	7	20	4	4	18	18	3	3	OFF	OFF	OFF
1	24	7	35	4	4	18	18	3	3	OFF	OFF	OFF
1	24	7	50	4	4	18	18	3	3	OFF	OFF	OFF
1	24	8	5	5	4	18	18	4	4	OFF	OFF	OFF
1	24	8	20	4	6	18	18	4	4	OFF	OFF	OFF
1	24	8	35	4	8	18	18	5	6	OFF	OFF	OFF
1	24	8	50	4	10	18	18	8	8	OFF	OFF	OFF
1	24	9	5	5	13	18	18	10	11	ON	OFF	OFF
1	24	9	20	6	16	18	18	13	14	OFF	OFF	OFF
1	24	9	35	8	19	18	18	15	17	OFF	OFF	OFF
1	24	9	50	9	20	18	18	16	18	OFF	OFF	OFF
1	24	10	5	11	22	18	18	17	19	OFF	OFF	OFF
1	24	10	20	12	22	18	18	18	20	OFF	OFF	OFF
1	24	10	35	14	25	18	18	19	22	OFF	OFF	OFF

TABLE IV (Continued)

MONTH	DAY	HOUR	MINUTES	OUTSIDE TEMP	INSIDE TEMP	SET TEMP	FLOOR TEMP	WET BULB TEMP	DRY BULB TEMP	GAS HEATER	SPRAYS	ELECTRIC HEATER
1	24	10	50	15	26	18	18	20	23	OFF	OFF	OFF
1	24	11	5	17	28	18	18	21	24	OFF	ON	OFF
1	24	11	20	17	26	18	18	21	24	OFF	OFF	OFF
1	24	11	35	18	28	18	18	22	25	OFF	OFF	OFF
1	24	11	50	20	28	18	18	22	25	OFF	OFF	OFF
1	24	12	5	20	28	18	18	22	25	OFF	OFF	OFF
1	24	12	20	21	28	18	18	22	25	OFF	OFF	OFF
1	24	12	35	22	29	18	18	23	25	OFF	OFF	OFF
1	24	12	50	22	27	18	18	22	24	OFF	OFF	OFF
1	24	13	5	23	27	18	19	22	24	OFF	OFF	OFF
1	24	13	20	23	27	18	18	22	24	OFF	ON	OFF
1	24	13	35	23	25	18	19	21	23	OFF	OFF	OFF
1	24	13	50	23	27	18	19	22	24	OFF	OFF	OFF
1	24	14	5	23	26	18	19	21	24	OFF	OFF	OFF
1	24	14	20	23	26	18	19	21	24	OFF	OFF	OFF
1	24	14	35	22	27	18	19	22	24	OFF	OFF	OFF
1	24	14	50	22	29	18	19	23	26	OFF	OFF	OFF
1	24	15	5	22	31	18	19	24	27	OFF	ON	OFF
1	24	15	20	23	26	18	19	22	24	OFF	OFF	OFF
1	24	15	35	23	29	18	19	23	16	OFF	OFF	OFF
1	24	15	50	23	32	18	19	24	28	OFF	ON	OFF
1	24	16	5	23	24	18	19	21	23	OFF	OFF	OFF

VITA

Simon Timothy Murray

Candidate for the Degree of

Master of Science

Thesis: A MICROPROCESSOR BASED CONTROL SYSTEM FOR A SOLAR HEATED GREENHOUSE

Major Field: Agricultural Engineering

Biographical:

Personal Data: Born in 1957 at Hughenden, Queensland, Australia, the son of Greg and Denise Murray

Education: Graduated from high school in Brisbane, Queensland in 1974; attended college at the Darling Down Institute of Advanced Education in Toowoomba, Queensland from January, 1976 to December, 1979; received a Bachelor of Engineering (Agricultural) in May, 1980; attended Oklahoma State University in Stillwater, Oklahoma, U.S.A. from August, 1980 to July, 1982 to complete the requirements for a Master of Science degree.

Professional Experience: Was Employed as an assistant to an agricultural engineer-livestock housing in Australia for two months. Served as a Graduate Research Assistant from August, 1980 to December, 1981, and a Graduate Teaching Assistant from January, 1982 to May, 1982, for the Agricultural Engineering Department of Oklahoma State University.

Professional Organizations: Student member of the American Society of Agricultural Engineers; and the National Society of Professional Engineers.