To the Graduate Council:

I am submitting herewith a thesis written by Richard Lee Kissell entitled "Digital Control for an Existing Variable-Air-Volume Air Handling System." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Electrical Engineering.

Walter L. Green

Walter L. Green, Major Professor

We have read this dissertation and recommend its acceptance:

Robert E. Bodenheimer

Donald W. Bouldin

Accepted for the Council:

L. Evans Roth

Vice Chancellor
Graduate Studies and Research

DIGITAL CONTROL FOR AN EXISTING VARIABLE-AIR-VOLUME AIR HANDLING SYSTEM

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Richard Lee Kissell

June 1980

ACKNOWLEDGMENTS

The author would like to express his sincere appreciation to Dr. Walter L. Green for his friendship, support and encouragement throughout the graduate program.

This thesis is dedicated to the author's wife, Nancy, and children, Jennifer, Kevin, and Bethany. Without their love, patience, faith, and understanding, this thesis would not have been written.

The author would like to thank Morgan Monks for his work on the data acquisition portion of this project. His friendship, discussions, and assistance were both enjoyable and enlightening.

Thanks also go to Paul Kelley and Mike Day for their work in making the system hardware a reality.

The author would like to thank Ann Lacava from the Manuscript Shop for all her help and especially B. J. Klepser for typing this manuscript.

A special thanks also to Knoxville Utilities Board for supporting this project, and to Bob "Poncho" Headrick and Joe Lawson of the Properties Department for their indispensable assistance.

ABSTRACT

This thesis presents the modifications to a variable-air-volume (VAV) system designed to improve the system efficiency. An improved control strategy is applied using direct digital control. The system operates in a medium-size office building (85,000 $ft.^2$) and is used only for cooling. The original VAV system experienced a problem (following new government regulations) in providing adequate air circulation for occupant comfort. The system was also incapable of using outside air for supplemental cooling.

Modifications focus upon three areas. First, 135 temperature sensors are installed in the various zones. These sensors are connected to two distributed microprocessors which are in turn connected to a host computer. Actual zone temperatures are used to adjust the set-point of the air leaving the cooling coils. Second, the capability for maintaining compliance with government regulations and providing documentation is provided. Third, the capability to bring in outside air for supplemental cooling is designed. Use of outside air is based on enthalpy calculations.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE PAGE

CHAPTER 1

INTRODUCTION

Knoxville Utilities Board's engineering and operations office building at the Middlebrook Pike Power Operations Center employs a variable-air-volume (VAV) system for cooling. This system controls the amount of conditioned air entering a space (zone) in response to the cooling demand. The system consists of two air handling units, each rated at 35,000 cubic feet of air per minute (CFM). Chilled water supplied to the office building from a separate mechanical building provides the cooling medium.

The VAV system operation was affected by recent government regulations[1] requiring the implementation of energy conservation measures. As zone temperature set-points were increased to comply with the regulations, air circulation in the building decreased due to the nature of the VAV system operation. This caused occupant discomfort resulting in the tampering of zone register settings. Actual zone temperatures were not monitored by the system controls. Verifying compliance with the new regulations required measuring and recording zone temperatures and manually readjusting the zone register settings.

Another deficiency identified with the original VAV system was the inability to use outside air to replace or supplement the chilled water cooling. The controlled use of outside air for cooling offers the potential for 25% savings in energy consumption.[2]

Having identified these problems three criteria were selected upon which system modifications were focused. First, zone temperature feedback was to be included into the system control strategy. Second, the VAV system control was to automatically maintain compliance with the new regulations, and capability for documentation verifying compliance was to be provided. Third, the capability for using outside air for supplemental cooling was to be implemented in order to maximize energy conservation.

The implementation of the VAV system modifications is divided into two phases. Phase I includes the installation of an improved control system design incorporating zone temperature feedback. This phase included making the proper interfaces from the sensors to the digital controller and from the controller to the chilled water valves. Phase I of the implementation should be completed during June 1980. Phase II includes the mechanical modifications necessary for outside air usage and the associated enthalpy control provisions. The implementation of phase II will be completed at a later date to be determined by Knoxville Utilities Board.

A distributed processing computer system was designed to implement the improved control system strategy. Two major categories of the system are (1) a remote data acquisition system and (2) a host computer. The development of the host computer system was the responsibility of the author. The control system strategy utilizing the host computer, and the outside air modification are discussed in detail in this thesis.

Chapter 2 describes the original VAV system operation. Chapter 3 presents the control system modification including the laboratory simulation of the cooling system. Chapter 4 explains the outside air modification. Chapter 5 presents the conclusions drawn from the modifications to the VAV air handling system.

CHAPTER 2

THE ORIGINAL VAV AIR HANDLING SYSTEM

The two air handling units comprising the VAV system are housed in a mechanical room located on top of the building. Each cooling unit provides conditioned air for approximately half of each floor in the four story building.

Two automatic control features were originally included on the air handling units. First, the temperature of the discharge air was controlled by throttling a chilled water valve. Second, the pressure in the supply air duct was controlled by modulating an inlet vane on the supply fan. Both of these controls were implemented with pneumatic proportional controllers.

A perimeter heating system and both cooling units share a common return air plenum. Figure 1 illustrates the cooling system configuration. Return air is drawn through return air vents along fluorescent lights into the ceiling space for each floor. From there each supply fan draws the air through the return air plenum back into the mechanical room (which is actually part of the plenum). From the mechanical room the air re-enters the intakes of the cooling units. For each air handler, an exhaust fan operates to balance the building pressure when outside air is used.

The supply air is delivered to each zone through a duct system and a set of ceiling registers or terminals. For each zone there is one self-contained control unit in one of the terminals. The remaining terminals operate as slaves to the master control terminal. Each terminal has a bellows-type damper assembly which controls the air

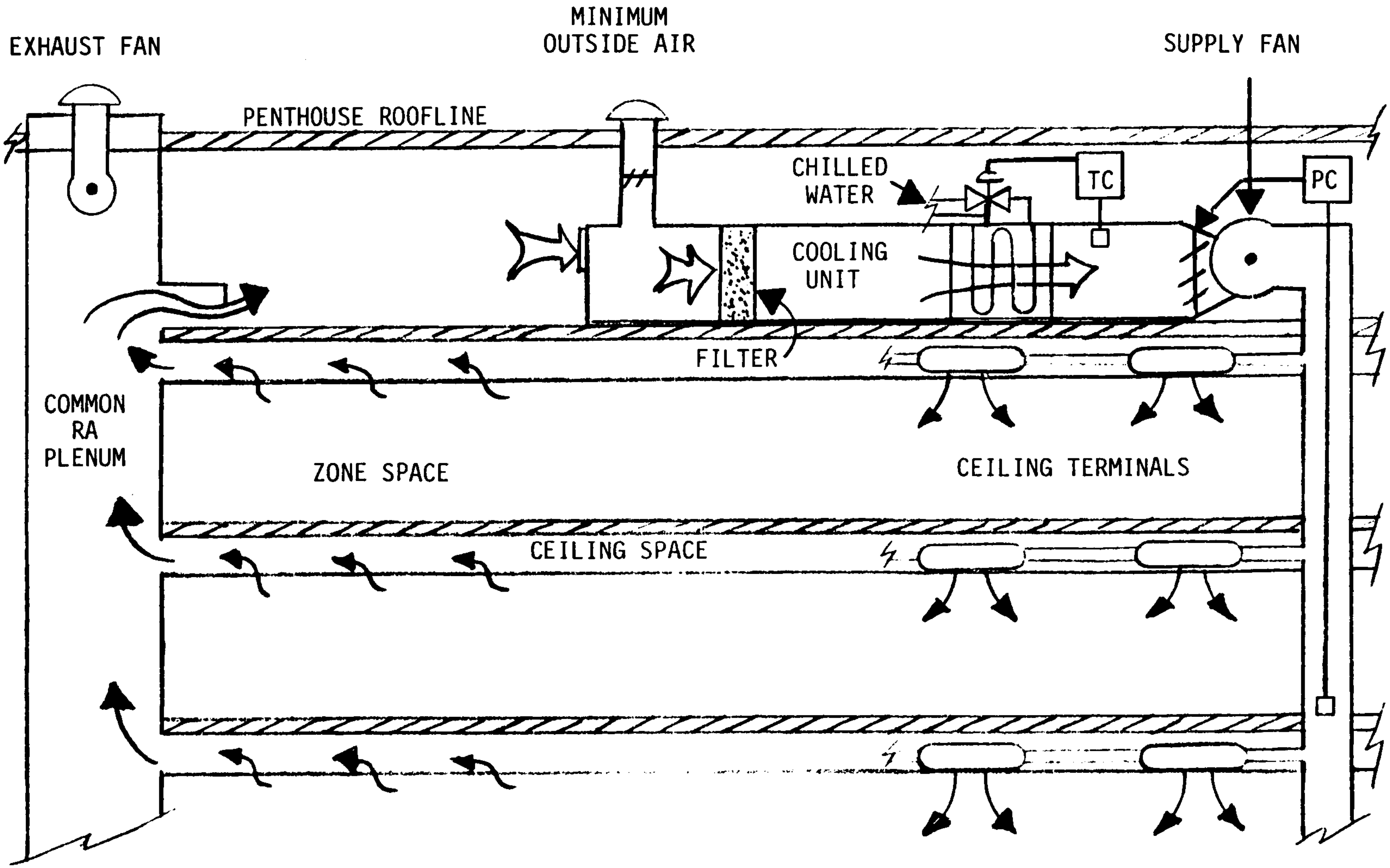


Figure 1. Cooling System Configuration.

delivery. The control system schematic for the air terminal is shown in Figure 2. Supply air used as control air passes through a filter into a volume regulator. The regulator is adjusted to allow a precise amount of control air pressurize the bellows. Thus, the terminal supply air delivery will match the regulator setting. Excess air is bled off through a fixed orifice in the regulator body. When the supply air pressure changes, the regulator proportionally inflates or deflates the bellows to maintain constant unit air flow. To provide variable-air-volume control, room air sampled by an aspirator is sensed by the unit's thermostat. On temperature change, the bimetallic element in the thermostat adjusts an air orifice which changes the bleed rate from the regulator. When the temperature rises above the thermostat setting, the thermostat opens the bleed orifice. The bellows deflate and unit air delivery increases to the maximum limit set by the regulator. As room temperature drops below the thermostat setting, the thermostat closes the bleed orifice. The bellows then inflate and unit air delivery decreases. The aspirator assures that room air is continuously drawn over the thermostat bimetallic element. The thermostat is adjustable to the desired room temperature setting by positioning a lever which extends into the room.

The raising of zone temperature set-points reduced the amount of air supplied to the zone. This decrease in air circulation created stale air conditions in some areas of the building. The discomfort experienced by some of the occupants prompted attempts to adjust terminal thermostat settings. The air circulation could be improved in this manner, but the resulting zone temperatures often no longer complied with the regulations. When temperature compliance checks were

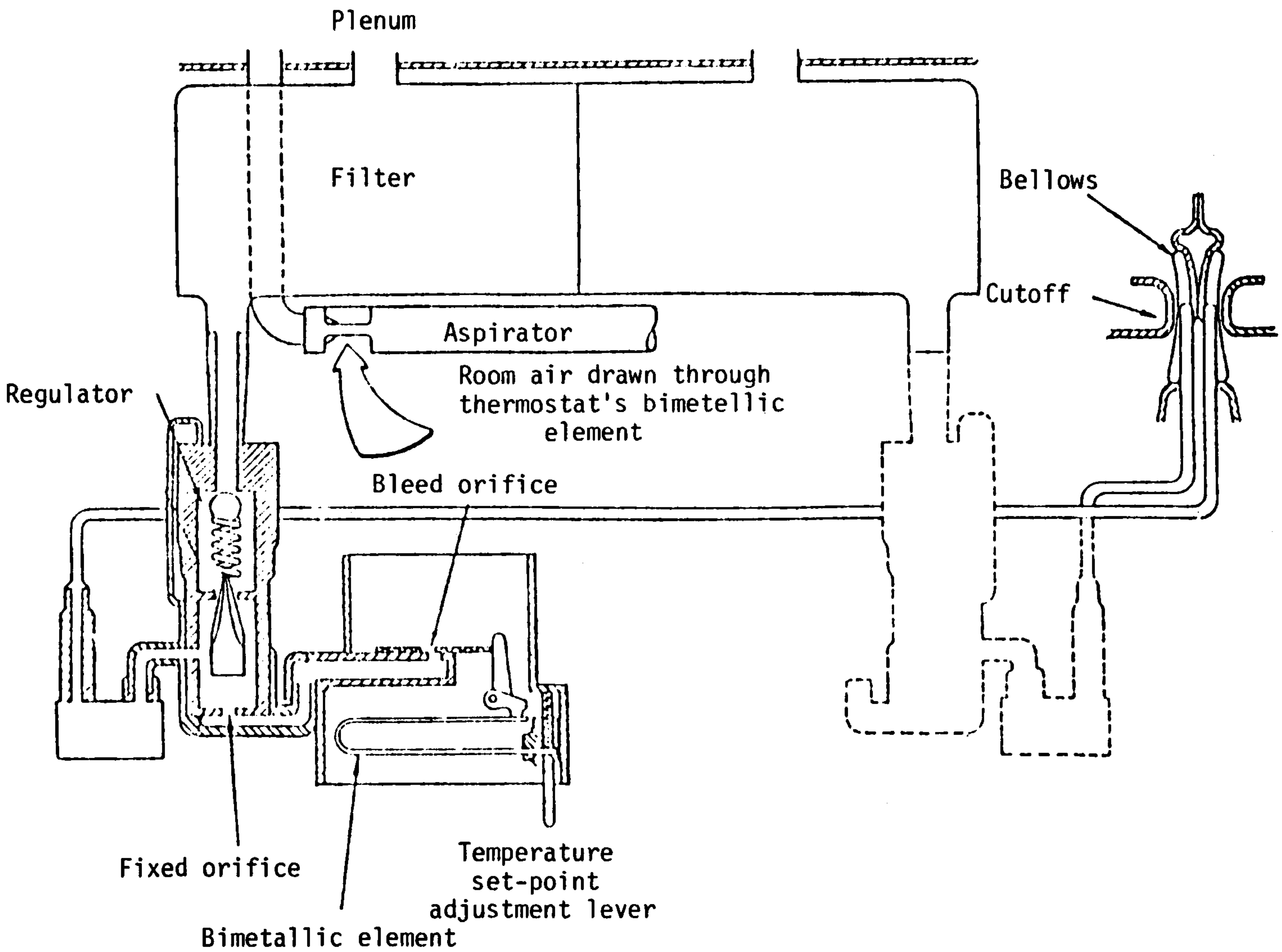


Figure 2. VAV Air Terminal Control System.

made the thermostats would be readjusted to the required temperature settings again causing the circulation problem.

The original VAV system and associated problems have been described. The following chapter presents the control system modification. The control strategy and the computer system hardware and software are discussed in detail.

CHAPTER 3

CONTROL SYSTEM MODIFICATION

The control system modification was designed using a distributed processing computer system comprised of a remote data acquisition system and a host computer. Two Motorola 6802 based microprocessor systems handle the remote data acquisition responsibilities. The host computer is a MACSYM 2 by Analog Devices, Incorporated.

The computer control system was configured and tested in a laboratory in Ferris Hall at The University of Tennessee, Knoxville. The laboratory system included an analog simulated cooling unit system, the operator control station without the pneumatic interface, and one of the microprocessor systems, all interfaced to the MACSYM 2.

The control system modification is presented in three sections. First, the control system strategy, the laboratory simulation, and system testing are described. Second, the hardware selected and designed to implement the control system is presented. Third, the software developed to execute the control strategy is discussed.

Control System Strategy and Laboratory Simulation

The control system strategy incorporates several distinct functions which are performed by the computer system. These functions include:

1. Controlling discharge air and zone temperature for each of the cooling units.
2. Providing the interfaces for the operator and the system control station.

3. Monitoring the system operation and providing alarms.
4. Verifying and documenting compliance with government regulations.

Discharge air and zone temperature control. The discharge air temperature and those zone temperatures affected by each cooling unit are controlled by a digital proportional-integral (PI) cascade controller regulating the chilled water valve as shown in Figure 3. The inner control loop consists of a digital proportional controller applied to the chilled water valve using the discharge air temperature as feedback. The set-point for the proportional controller is provided in software by the PI controller.

The PI control is implemented using the position form of the discrete PI algorithm[3] defined as:

$$m_n = K_c \left[e_n + \frac{T_s}{T_i} \sum_{k=0}^{n} e_k \right] + M_R \qquad (1)$$

where e_n = process variable error

K_c = proportional gain

T_i = reset time

T_s = sampling time

M_R = initial valve position

The initial valve position is preset manually prior to switching to computer control.

A first-order-lag-plus-dead-time model was obtained for the cooling unit system in the form

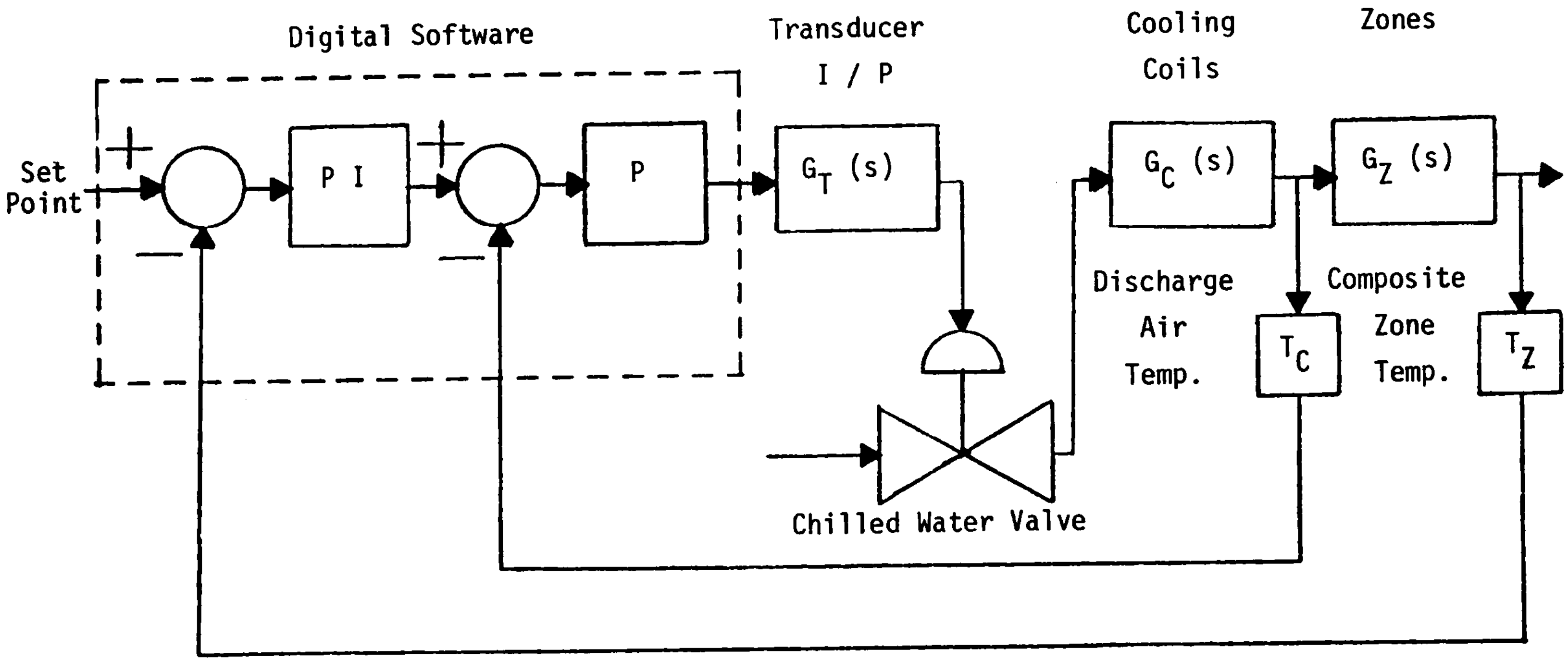


Figure 3. Cascade Control Loop.

$$G(s) = \frac{K e^{-\theta s}}{1 + \tau s} \tag{2}$$

where θ = dead time

K = steady-state gain

τ = time constant

A simulation of a cooling unit system was built on an analog manifold using operational amplifiers and passive circuit elements. The MACSYM 2 was interfaced to the manifold and the control loop software was developed and debugged using the simulated system. Initial tuning of the PI controller was performed using coefficients associated with the integral of the absolute value of the error (IAE) criteria.[3] In addition to the analog manifold simulation utilizing the host computer, the cooling system PI control was simulated using the Continuous System Modeling Program (CSMP)[4] on an IBM 360 computer. The design of the cooling system model, the simulation using the analog manifold, and the CSMP simulation are described in detail in Appendix A.

Zone temperature feedback is provided to the PI controller by the remote data acquisition system. Each microprocessor system collects all of the zone temperatures from two of the four floors. Data are kept separated by floor number and cooling unit number. The temperatures are rounded off to the nearest degree Celsius, and then the highest, lowest and average temperatures are computed for each floor and for each cooling unit. Again, zone data for the first cooling unit are kept separated from the data for the second cooling unit. Next the microprocessors exchange their data with each other over a serial communications link. This results in both microprocessors having all of the measured and computed zone temperatures for both of the cooling units. At this point the overall highest, lowest, and average zone temperatures are computed for each cooling unit in both microprocessors.

Each microprocessor is connected to the host computer across a separate serial communications link. The host communicates with only one microprocessor until a problem occurs; then the host begins communication with the second microprocessor. The data are redundant in the two micros providing backup communication capability. Thus the host computer can continue operation of the VAV system if one microprocessor system or communication link is temporarily out of service.

Two data sets are required by the host computer. First, all zone temperatures are requested from the data acquisition system every five minutes. These data are stored in the host for documentation purposes and operator information. Second, for each cooling unit, the highest, lowest, and average zone temperatures overall and by floor, are requested from the data acquisition system once each minute. These data are used to compute a composite zone temperature for each cooling unit PI controller. Initially the composite zone temperature will be the overall zone average for each cooling unit. After performance data has been accumulated for the modified system operation, the composite number may be weighted differently from the different floors to improve the representation of the variance from the average temperature.

The laboratory system incorporated one microprocessor with the capability for collecting 16 temperatures. This facilitated developing and debugging the communications software and the remote data acquisition sensor configuration.

System control panel and operator interfaces. A panel housing the host computer and the operator control station will be located in the mechanical equipment room. A selector switch on the operator control panel indicates to the host computer which of three modes of system operation has been selected. The three modes are (a) automatic computer control, (b) manual control, and (c) back up pneumatic control as shown in Figure 4. In automatic computer mode each control loop's software is interfaced to a current-to-pressure (I/P) transducer which in turn is connected to the chilled water valve. Manual mode allows the connection of manual adjustment controls directly to the I/P transducer disconnecting the control loop software. The back up pneumatic control activates the original pneumatic controller. This allows continued operation of the VAV system if a host computer problem occurs. Loss of computer power automatically activates the back up pneumatic control.

During initial start up from a computer system shutdown, the mode switch is kept in the back up pneumatic position until a message on the computer console instructs the operator to reposition the mode switch. The operator is instructed to select manual control mode and adjust the level of each control function to its initial setting. Once this has been completed the control selector is switched into automatic-computer mode allowing the control loop software to become activated. The normal daily start up and shut down of the VAV system is performed independently under computer control.

Two digital L.E.D. output displays on the control panel enable the operator to examine desired temperatures and relative humidities (see Figure 5). The display functions which may be selected include

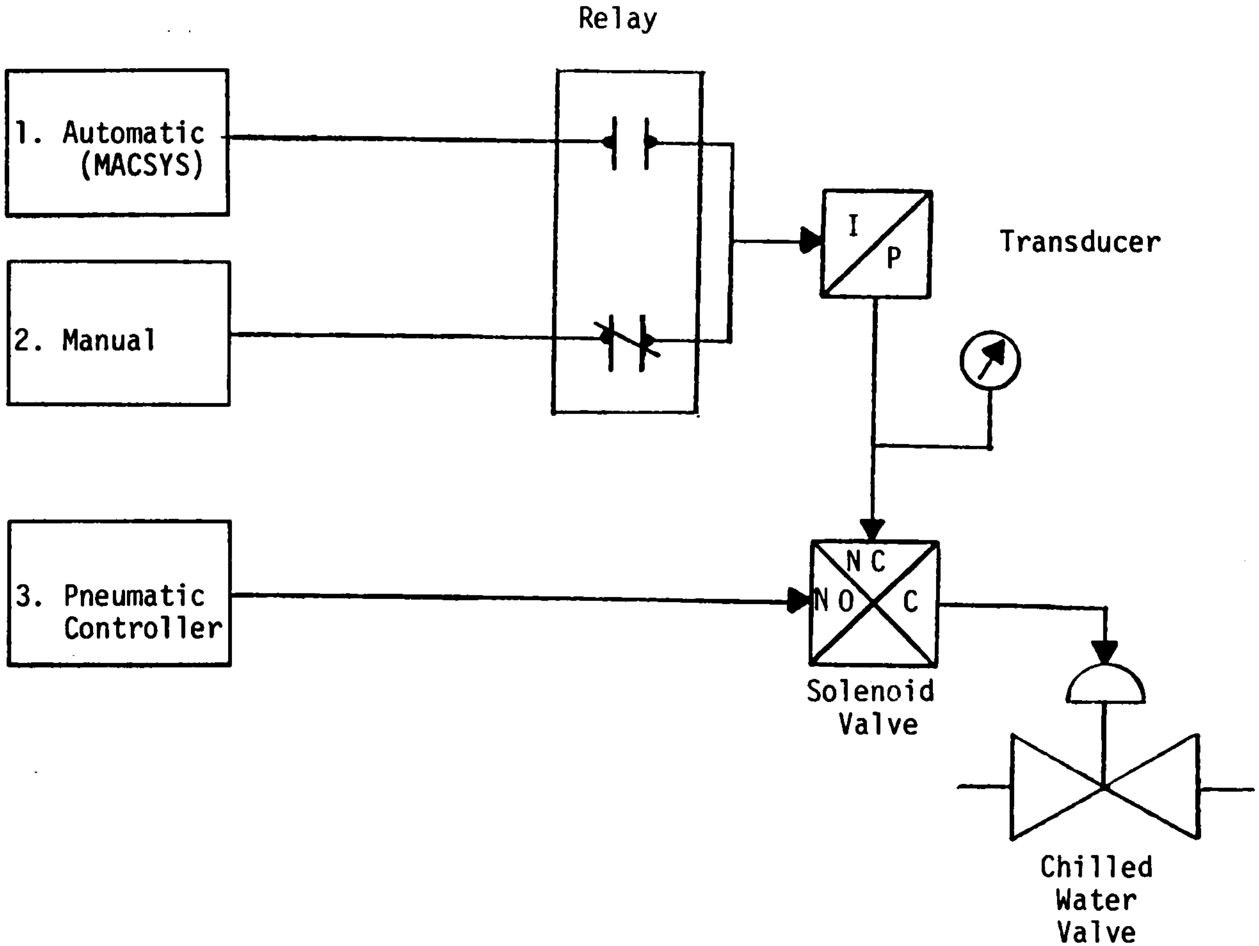


Figure 4. Control System Operation Modes.

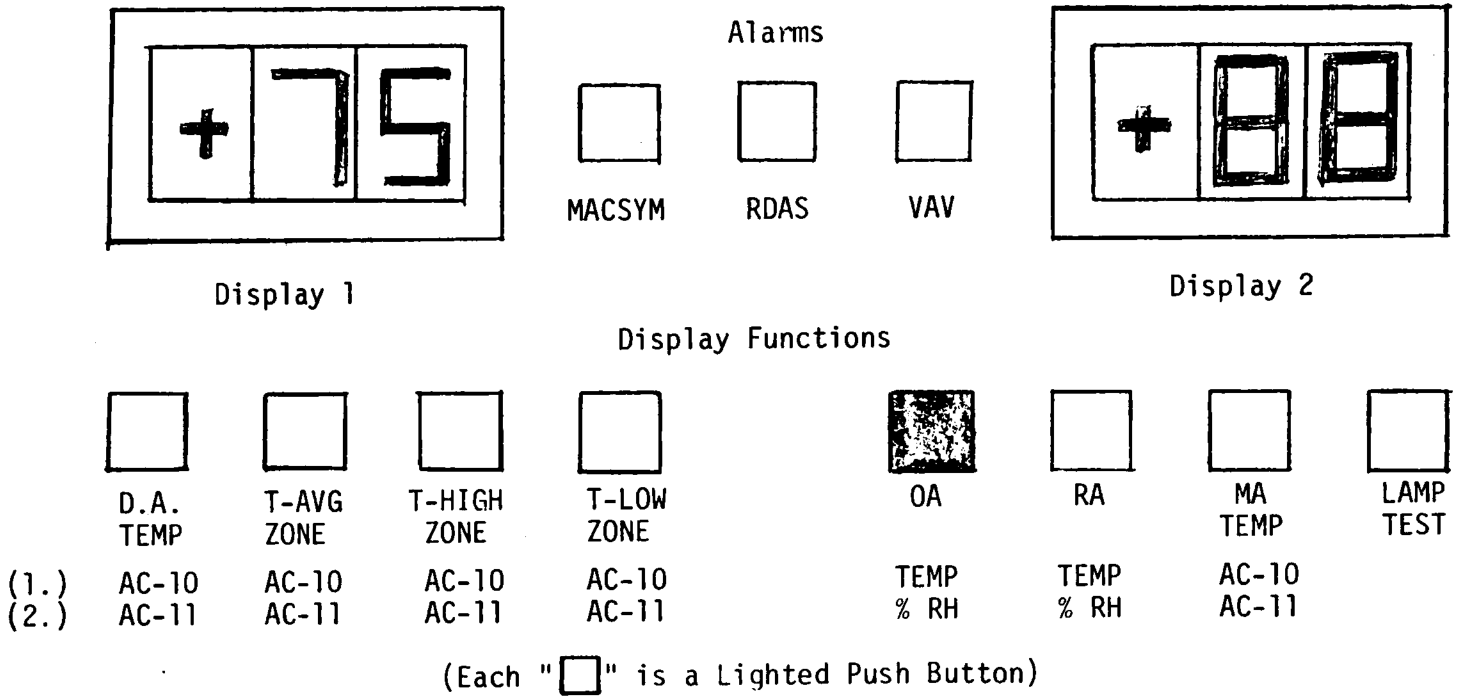


Figure 5. Control Station Display Panel.

outside and return air temperatures and relative humidities (humidity sensors will be installed during Phase II of the implementation). In addition, for each cooling unit the mixed and discharge air and highest, lowest and average zone temperatures can be displayed. To complete the operator interface, keyboard functions enable any particular zone temperature to be displayed on the computer screen. Also zone temperatures and other system functions may be displayed on a computer terminal at a remote centralized operator station in the mechanical building.

The laboratory control station included the display panel with temperature display capability, the mode selector along with a simulated manual control capability, and a computer terminal. This facilitated the development and testing of the software interfaces.

System operation monitoring and alarms. While each computer function is executed, checks are made for process variable limits being executed, failures of data acquisition sensors, and microprocessors or communications link problems. The protocol for data communication between the host computer and the microprocessors automatically includes failed sensor information. When a problem occurs a system alarm is activated and a corresponding message is provided through the computer console screen. Alarms at the remote centralized operator station are also activated for all VAV system problems.

Verifying and documenting compliance with government regulations. The computer will control the zone conditions to maintain the desired set-point. In this manner the compliance of government regulations will be automatically and continuously verified. The documentation of the compliance is provided by having the computer log the zone temperatures with the date and time of data acquisition for establishing a record of the actual system operation.

The laboratory system provided the capability for performing the monitoring and alarm annunciation. Documentation capability was also developed and tested using the MACSYM 2 cartridge tape unit and a remote computer terminal for printing. A detailed description of the control system hardware is presented next.

Control System Hardware

The control system hardware includes two systems: (1) the MACSYM 2 host computer system with process sensors and the associated system control station, and (2) the remote data acquisition system.

Host computer system. The MACSYM 2 computer system organization[5] is illustrated in Figure 6. The system consists of a 16-bit central processing unit (CPU), 32K bytes of random access memory (RAM), an interactive CRT screen/keyboard console terminal, a cartridge tape mass storage facility, and a variety of analog and digital input/output (I/O) cards. The I/O configuration of the MACSYM 2 reduced the hardware interface synthesis required to perform measurement and control functions. An I/O controller provides an interface between the CPU and the I/O signal conditioning cards; this controller handles tasks common to several I/O cards such as analog to digital conversion. The I/O controller is implemented as a memory mapped device that appears to the CPU as a block of memory; this permits a flexible means for software access to the I/O. In addition to analog and digital I/O functions, externally driven interrupts activate software tasks to handle event dependent functions.

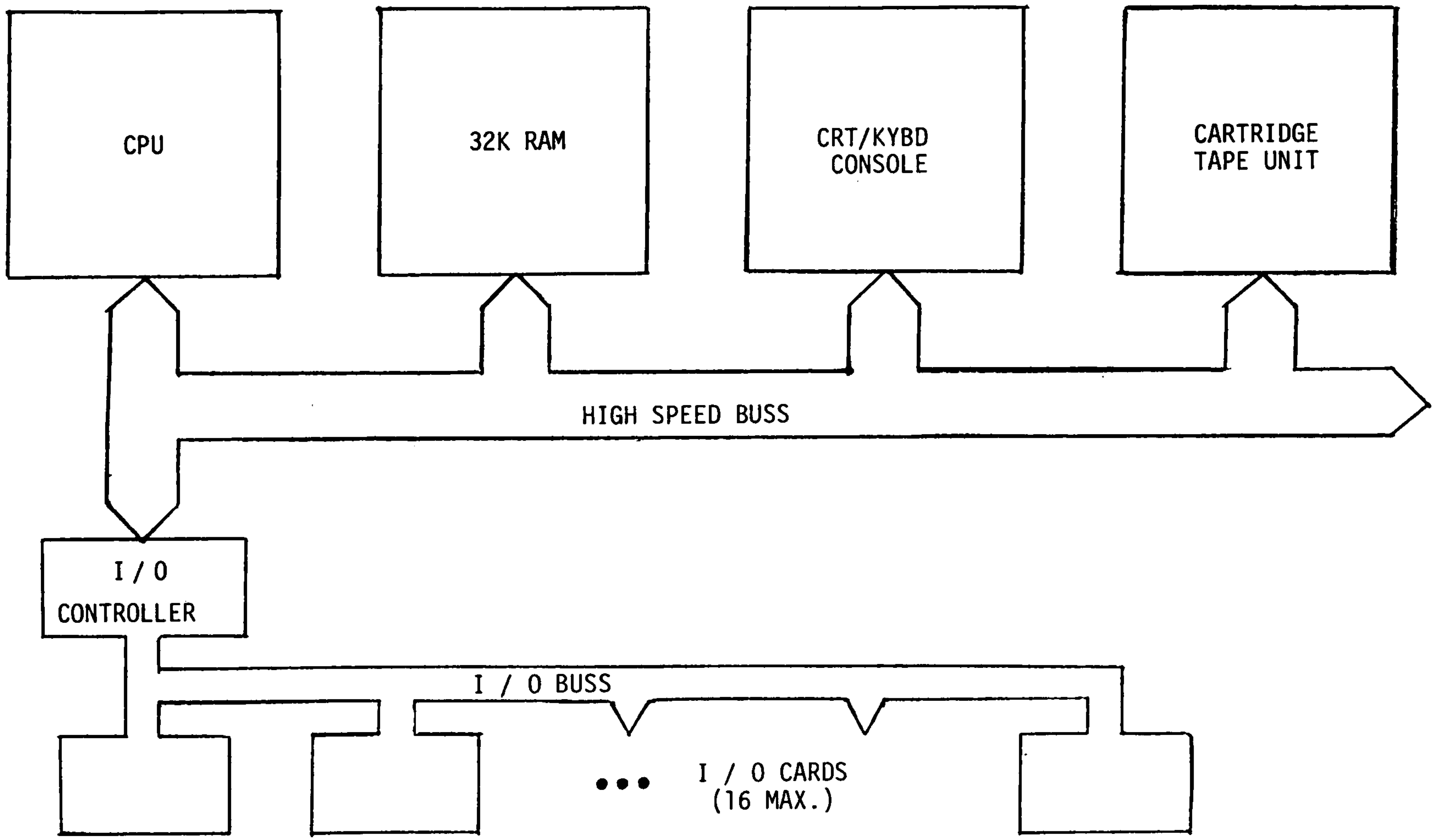


Figure 6. MACSYM 2 Computer Organization.

The host computer performs the switching requirements for the digital display panel and the manual control mode. The interface between the computer and the operator control station consists of solid-state relays. These relays provide for conversion of digital signal levels to standard mechanical equipment control voltage levels for both inputs and outputs. The pneumatic interface consists of the current-to-pressure transducer for each chilled water valve and a sole-noid valve for switching the control air lines. Manual control consists of a circuit with a potentiometer which can output a 4-20 mA. current along with a pressure gauge on the I/P output for visual feedback. Dry bulb temperature is measured by the host computer using a two terminal integrated circuit temperature transducer.[6] Figure 7 illustrates the signal conditioning and the calibration circuit for the transducer. The transducer produces an output current proportional to absolute temperature. Potentiometer R1 adjusts the calibration for low temperatures. High temperature calibration is adjusted with potentiometer R2.

Remote data acquisition system. The microprocessor organization is shown in Figure 8. The system consists of three major parts: (a) a single board microcomputer, (b) a 12-bit analog to digital converter (ADC) card, and (c) signal conditioning cards. All of the cards are connected through a common communications buss.

Zone temperatures are measured by each microprocessor using the same transducer used by the host computer. Each zone sensor will be installed into the VAV ceiling terminal control unit adjacent to the thermostat bimetallic element. The control unit aspirator will assure

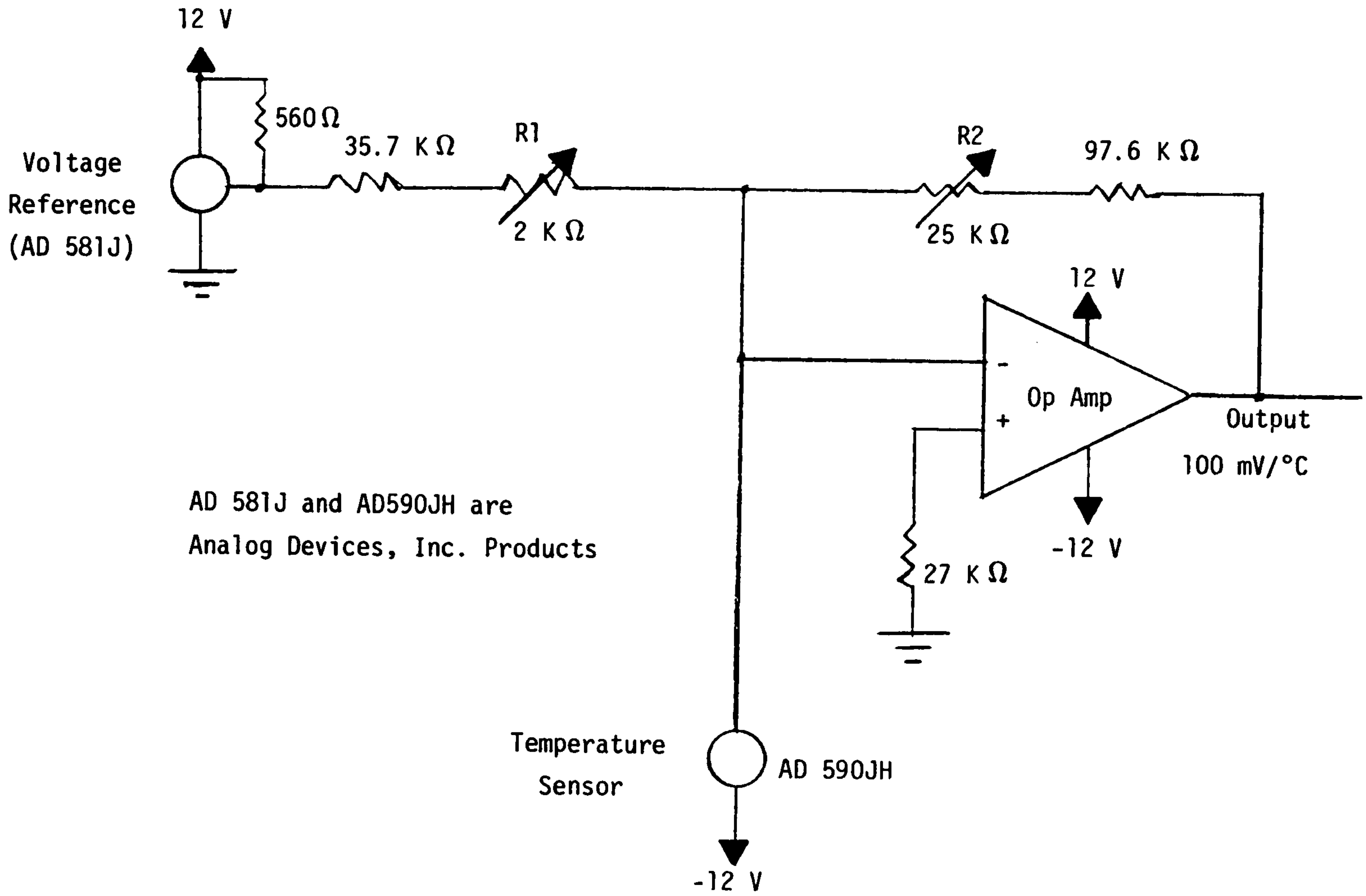


Figure 7. Temperature Transducer Circuit.

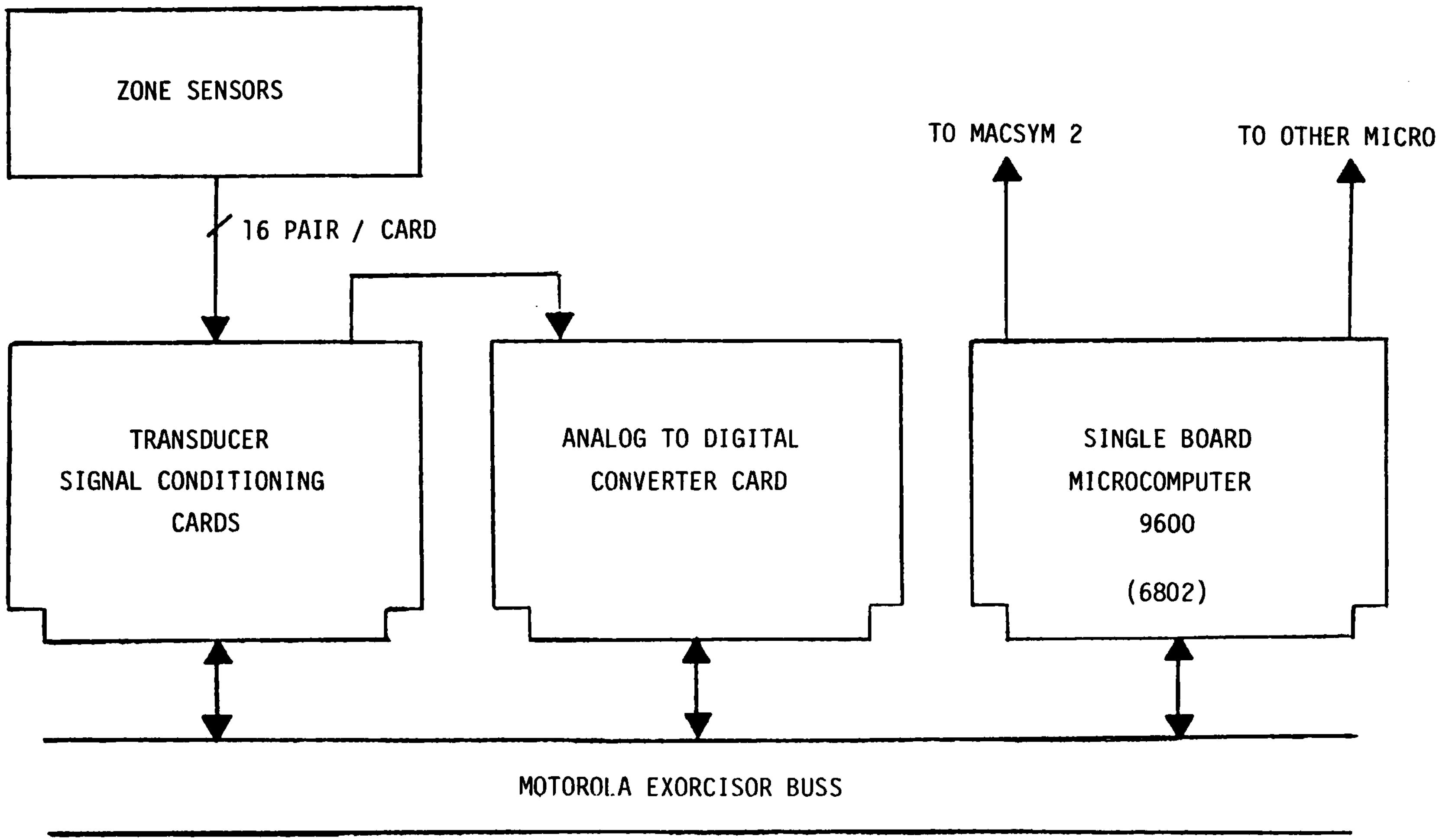


Figure 8. Microprocessor System.

that room air is continuously drawn across the transducer. In this manner the zone temperature transducer and the control unit thermostat will be sensing the same zone condition.

The signal conditioning cards multiplex 16 transducers and calibration circuits into a single operational amplifier for output. Thus, the ADC input channel requirement and the amount of signal conditioning electronics are both reduced.

Communication between the microprocessors and the host computer is performed using an asynchronous serial communications scheme. The data format is comprised of a start bit, seven data bits including an even party bit, and one stop bit. The transmission rate is 4800 BAUD.

The description of the control system software follows.

Control System Software

The MACSYM 2 software language is MACBASIC,[5] an enhanced version of Dartmouth BASIC. This language has a number of facilities that support measurement and control functions. In addition, multitasking capability allows the execution of simultaneous asynchronous events or operations in a simple orderly manner.

The software program which executes the control strategy was developed around the multitasking capability of the MACSYM 2. Each distinct function in the control strategy was divided into one or more dependent or independent tasks and given a priority. Task priorities establish the order of execution. The priority number can range from 1 to 255; 1 is the lowest priority and 255 is the highest. Active tasks with the same priority execute concurrently on a round-robin basis: one line of a task will execute, then one line of the next task will execute, and so forth. When a higher priority task is activated, it will interrupt the execution of any lower priority tasks and then execute to completion or suspension. Tasks are suspended during large data transfers or program delays such as the execution of a WAIT instruction. Tasks can be activated unconditionally, by other tasks through software, or on the arrival of an external event, a time of day, or the conclusion of a periodic interval.

A listing of the laboratory system software program which has been divided into separate tasks is given in Appendix B. Each task is documented with an explanation of the task function, the method of activation, and the definition of program variables. The final software for the VAV control system will be altered slightly from the laboratory system software. The changes will provide the correct I/O references for the I/O connections used with the actual hardware.

Figure 9 illustrates the software program organization. Upon computer system start up (from power down), the main program executes once identifying all tasks and assigning priorities. All arrays and strings used in the entire program are then dimensioned. The first task to be activated is system initialization. System initialization also executes only once and sets up all of the necessary starting values for the computer control to begin execution. Instructions are given through the console to the operator to switch the system control mode to manual and initialize the control loops. Next, the event dependent tasks have their interrupts enabled. Each periodic task is then activated in the proper sequence completing the automatic computer control start up.

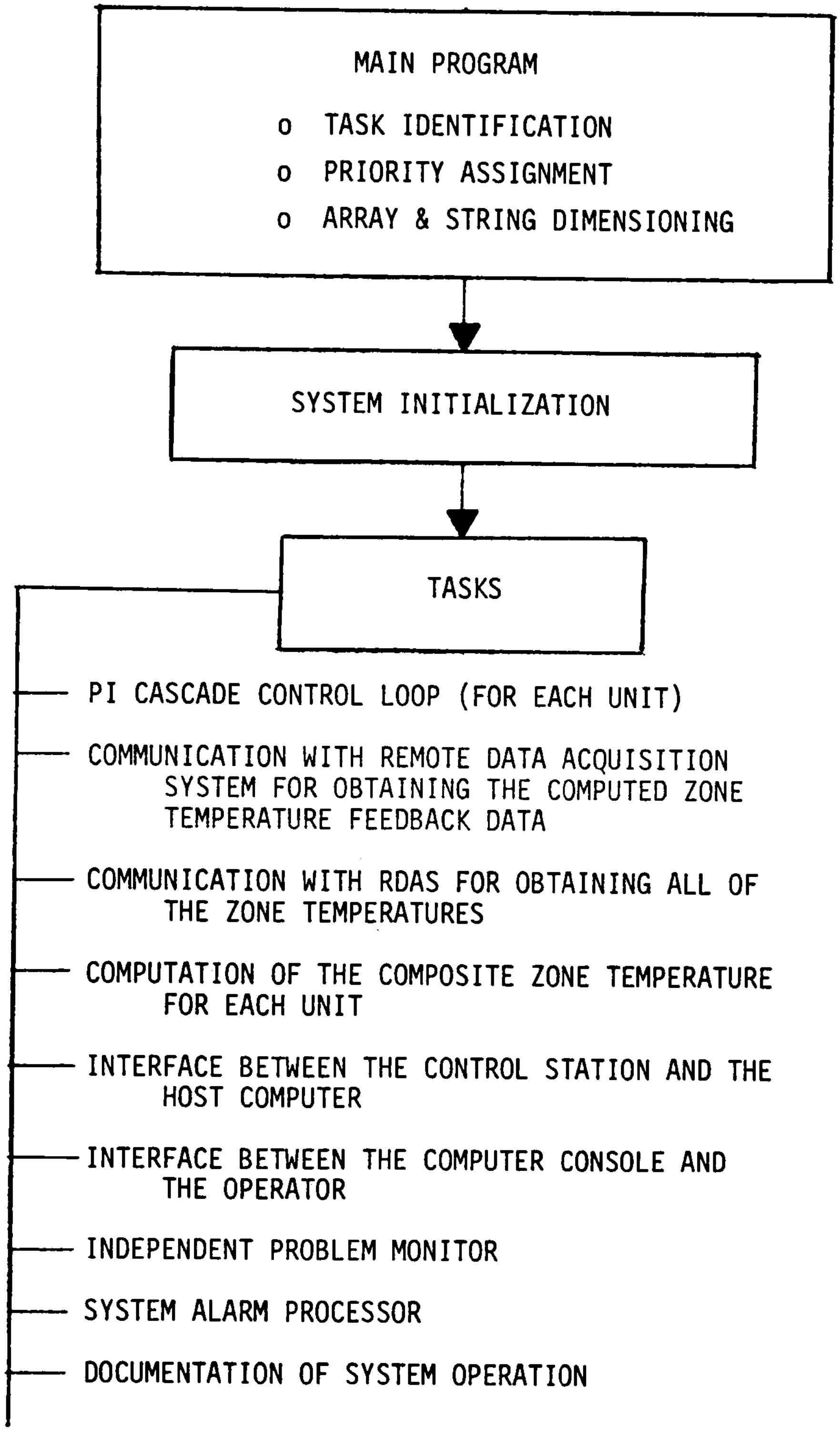


Figure 9. Software Organization.

The above sequence is executed only when the computer is brought on line. For normal operation the computer handles the daily start up and shut down of the VAV system without operator intervention. The repetitive tasks performed by the host computer comprising the normal system operation were divided into nine areas and are listed in Figure 9.

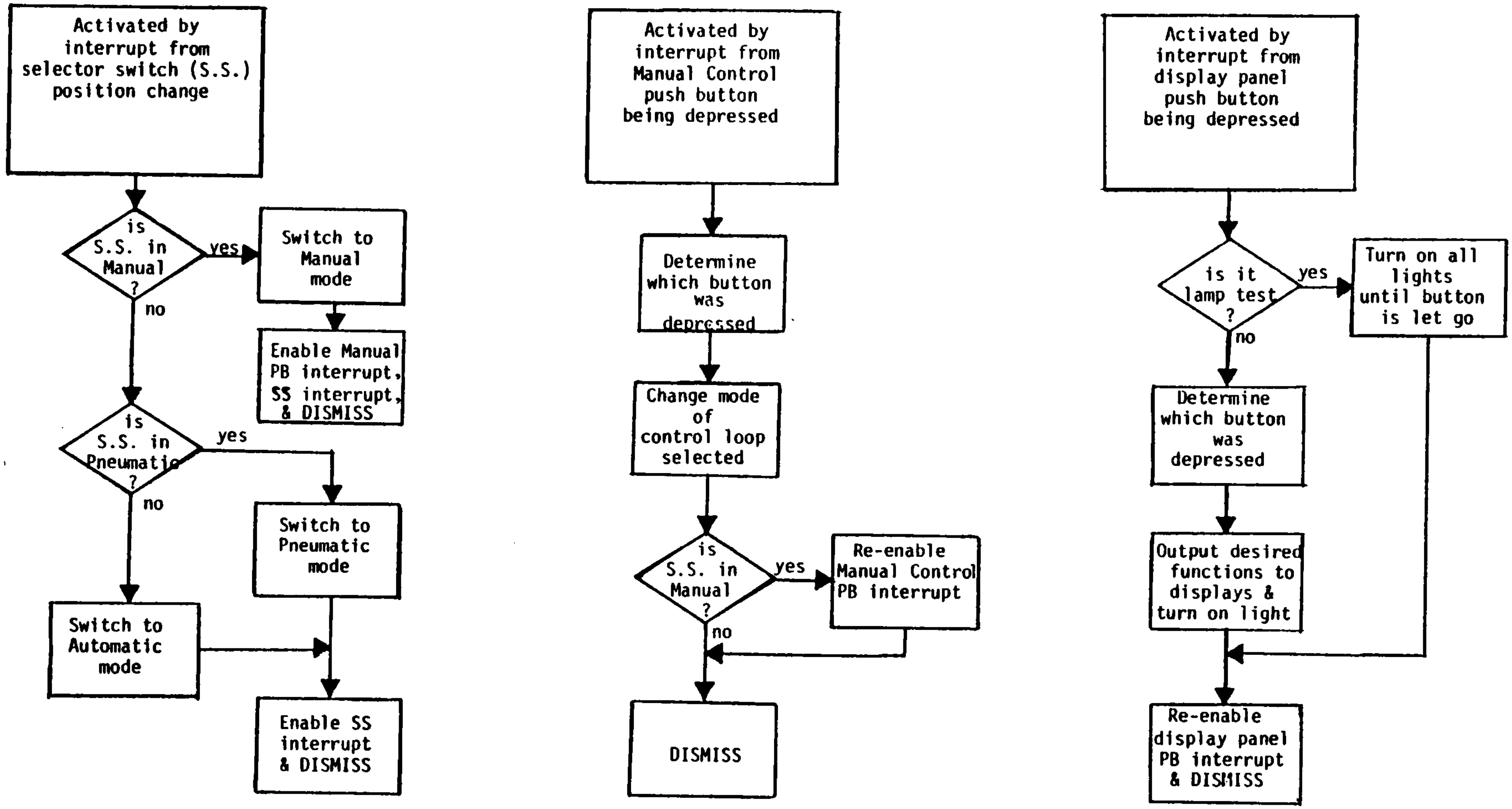
PI cascade control loop. The digital PI algorithm from equation 1 was implemented for the outside loop of the cascade control in Figure 3, page 11. A digital proportional controller is implemented for the inner loop. The outer loop task is repeated once each 30 seconds. The inner loop task is repeated once each second. Both tasks are activated periodically.

Communication with the RDAS for obtaining the computed zone temperature feedback data. This periodic task is executed once each minute. A beginning of transmission control character followed by a code letter is sent to the RDAS activating an interrupt in the microprocessor receiving the call. The micro responds by sending the computed zone temperature data to the host computer followed by a carriage return character. These data are stored in the host computer to be used by the task which computes the composite zone temperature for each cooling unit. The last function this communications task performs is to activate the task which computes the composite zone temperatures.

Communication with the RDAS for obtaining all of the zone temperatures. This task is also activated periodically and executes once every five minutes. The same communications protocol is used as in the preceding task. A different code letter prompts the sending of all zone temperatures from the microprocessor. The transmission of the data is divided into two segments with a delay in between to allow other periodic tasks the necessary time to complete their execution. The zone temperatures are stored in the host to be used by the operator interface and documentation tasks.

Computation of composite zone temperatures for each cooling unit. This task is activated immediately after receiving the computed zone temperature feedback data from the RDAS. Initially the overall average zone temperatures will be used as the composite temperature by the PI controllers. A more sophisticated method for computing the composite numbers may be implemented after data on the control system performance has been evaluated. The composite numbers are stored and later used as the feedback signal for the outer loop of the cascade control system.

Interface between the control station and the host computer. The functions of this interface were divided into four tasks. Three of the four tasks are activated by interrupts (see Figure 10). The control system mode selector switch is connected to an interrupt line. A change in the selector position generates an interrupt activating the first task which determines which operational mode was selected. The corresponding solid-state relays are then switched, and the mode selected is initiated. An L.E.D. indicator gives the particular system operational mode. The final function of the task is to enable a second interrupt line connected to the manual control buttons when the system is switched to manual mode. The second task is activated by the



A. Task 1 B. Task 2 C. Task 3

Figure 10. Control Station Interrupt Activated Tasks.

interrupt generated when one of the manual control buttons is depressed. This task determines which button was pushed and independently switches the appropriate control loop to manual control.

The display panel select buttons are connected to a third interrupt line. Whenever a display button is pushed a third task is activated and determines which display function was selected. This task then switches the selected outputs to the digital displays and turns on the appropriate indicating light.

A fourth task activated periodically executes once every three seconds updating the information to the displays.

Interface between the computer console and the operator. The functions performed by this interface are handled by two tasks. In the first task the MACSYM keyboard is monitored for an input requesting a particular zone temperature value. The zone temperature is printed on the console screen when a valid input has been received. In the second task the remote computer terminal is monitored for inputs requesting various system functions. The corresponding information is printed to the remote terminal when a valid request is received. Both of these tasks operate as background programs executing only when requests are received.

Independent problem monitor. Detection of malfunctions in system operation is performed within most tasks. Detection of some problems require checks to be made external to a task. The detection of a RDAS communication fault or microprocessor failure are two examples. This task performs the independent diagnostics needed to assure system operation.

System alarm processor. When a system problem occurs an alarm identification array is encoded by the task detecting the problem. The alarm processing task is then activated. This task centralizes the functions needed to be performed for all problems. The identification array is first checked to determine what problem occurred and which alarm needs to be activated. The proper alarm indicator on the operator control station is flashed until it has been acknowledged by an operator pushing its reset button (See Figure 5, page 16). A corresponding message explaining the details of the problem is displayed on the console screen when the alarm is acknowledged. The indication of all system problems at the remote centralized operator station in the mechanical building is also performed by this task. A separate interrupt driven task handles the switching for the push-buttons and indicators.

Documentation of system operation. Documentation of the system is performed in one task. This task is activated periodically and executes once each 30 minutes. The zone temperatures and the computed zone temperature feedback data are recorded and stored in the memory along with the time of data acquisition. At the end of the day's operation the data from memory is stored on tape and is available for printout at operator discretion.

The control system modification has been described. The modification for usage of outside air for supplemental cooling is presented in the next chapter. The necessary mechanical modifications and the enthalpy control provisions are discussed in detail.

CHAPTER 4

OUTSIDE AIR MODIFICATION

The outside air modification is divided into two parts: (1) the design of the mechanical changes required to use outside air for supplemental cooling, and (2) the enthalpy control provisions.

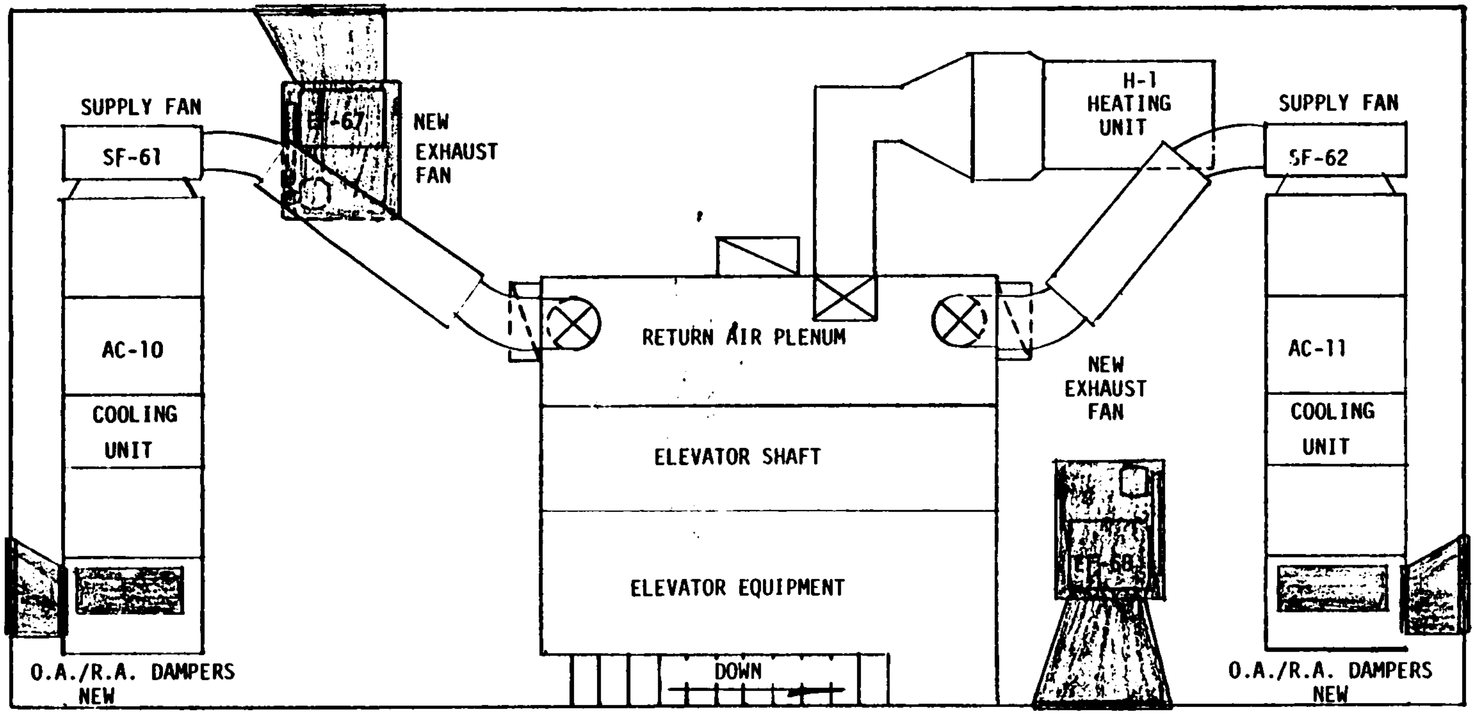
Design of the Outside Air Mechanical Changes

The VAV cooling system was analyzed to determine the necessary steps to provide each air handling unit the capability to bring in outside air at full capacity. Preliminary analysis indicated the outside air usage could probably be implemented by simply providing a duct arrangement with dampers controlling the outside and return air flows. A more in-depth study of the VAV system proved the earlier analysis to be invalid. In order to implement the initial idea the VAV system supply fans would be required to develop a positive pressure differential between the return air and the outside atmosphere. This requirement was determined not to be feasible since a negative pressure was already being developed between the mechanical room and the return air side of the system to draw the return air back into the cooling units. In addition, the volume output of each supply fan was controlled using supply air pressure as feedback. An increase in the supply air pressure to provide the needed differential pressure between the return air and the outside atmosphere would result in the supply fan inlet vane being throttled back. Any increase in the supply air pressure set-point was not possible since the resulting pressures could exceed the burst strength of the flexible ductwork in the system.

At this point, after realizing the scope of the required modification, additional engineering expertise in the heating, ventilation and air conditioning field was sought. Richard M. Kelso of Kelso and Associates Engineers in Knoxville was secured for consulting on the design for implementing the mechanical outside air modifications.

Figure 11 shows a layout of the mechanical equipment room with the modifications indicated. First, each cooling unit required the addition of return air dampers and outside air louvers, duct, and dampers. This equipment provided for the outside air to be brought into the cooling units. Next, in order to balance the building air pressure, two exhaust fans were required to expel an amount of air equal to that brought in by the air handling units. The exhaust fans which were specified are double-width-double-inlet (DWDI) centrifugal fans with variable inlet vanes to change their output. The control required for fan operation is provided by a pneumatic differential pressure controller. The controller senses outside atmospheric pressure with respect to building air pressure. The controller then sequences the fan operation and adjusts the fan inlet vanes so the differential pressure is maintained within limits such that no detrimental effects are caused from out of range building pressure.

Detailed specifications and drawings for completing the outside air equipment modification were prepared and delivered to Knoxville Utilities Board (KUB). The specifications included the requirements for all equipment purchased, the manner in which the work was to be performed, and the minimum acceptable standards for materials and workmanship. The detailed mechanical and electrical drawings specified



(changes are shaded)

Figure 11. Mechanical Equipment Room Modifications.

the installation of the equipment. Bids for the completion of the modification were obtained by KUB, and the decision was made to complete this phase of the project at a later date due to the economic situation at that time.

Enthalpy Control Provisions

The control for bringing in outside air will be performed by the MACSYM 2. Outside air (OA) and return air (RA) dry bulb temperatures and relative humidities will be sensed directly by the host computer. From these readings the total heat content, or enthalpy (h), will be computed for OA and RA.[7] The computer then mixes the OA and RA streams to maximize energy conservation for the cooling process.[8]

A flow chart summarizing the enthalpy control logic execution in software is shown in Figure 12. Figure 13 illustrates the psychrometrics of the enthalpy control logic. The constant enthalpy line crossing through the RA condition divides the chart into two significant regions. Region 1 is above the constant enthalpy line for the RA condition where the enthalpy of RA (h_{RA}) is always less than the enthalpy of OA (h_{OA}). Region 2 is below the constant enthalpy line for the RA condition where $h_{OA} < h_{RA}$. In region 1 the enthalpy of any mixture of OA and RA is always greater than h_{RA}. Therefore, maximum RA is used for OA conditions in region 1. For OA in region 2 the enthalpy of any mixture of OA and RA is always less than h_{RA}; however, region 2 is subdivided further to determine the control action. For OA conditions in region 2a since the OA dry bulb temperature is less than the desired discharge air dry bulb temperature, the OA is mixed with the RA to maintain the desired discharge air temperature. For OA

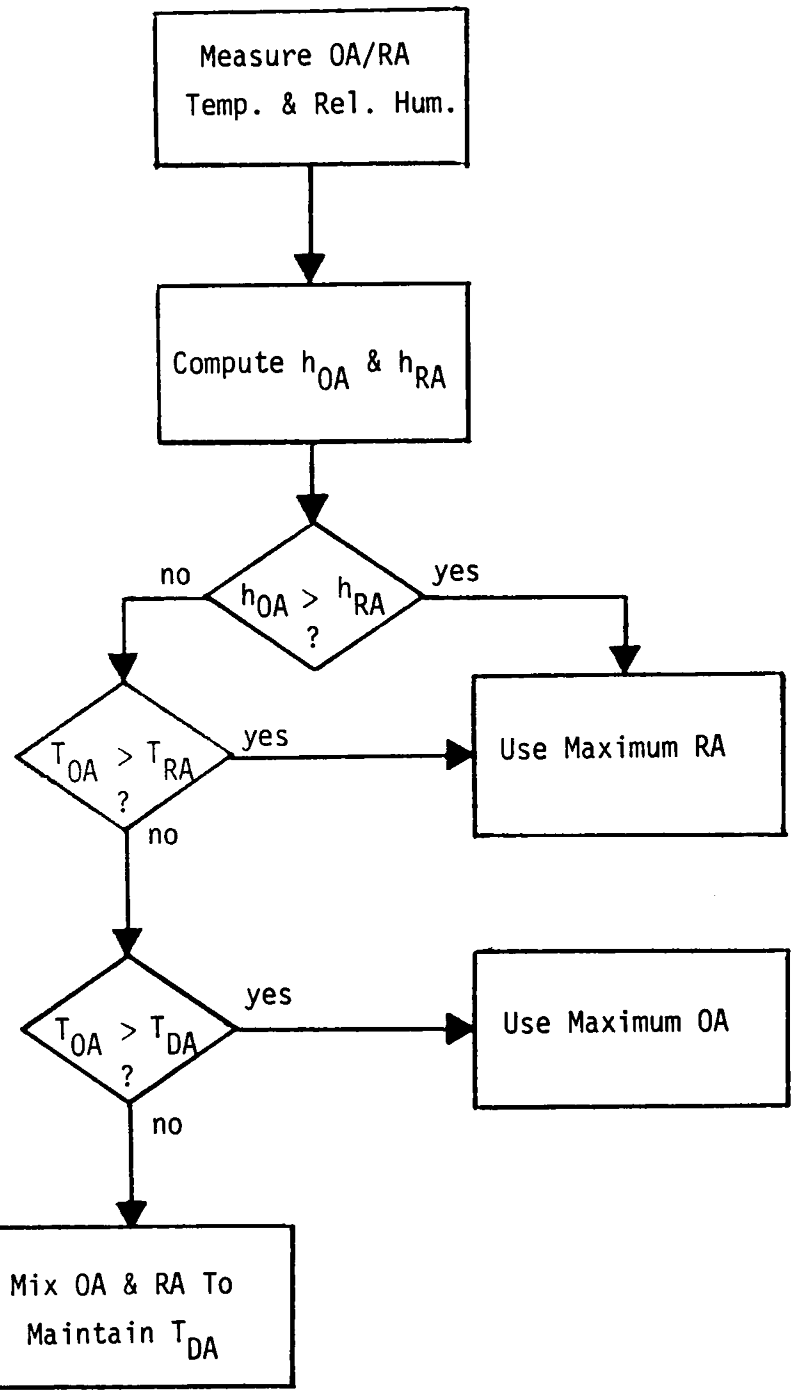


Figure 12. Flowchart for Enthalpy Control Logic.[7]

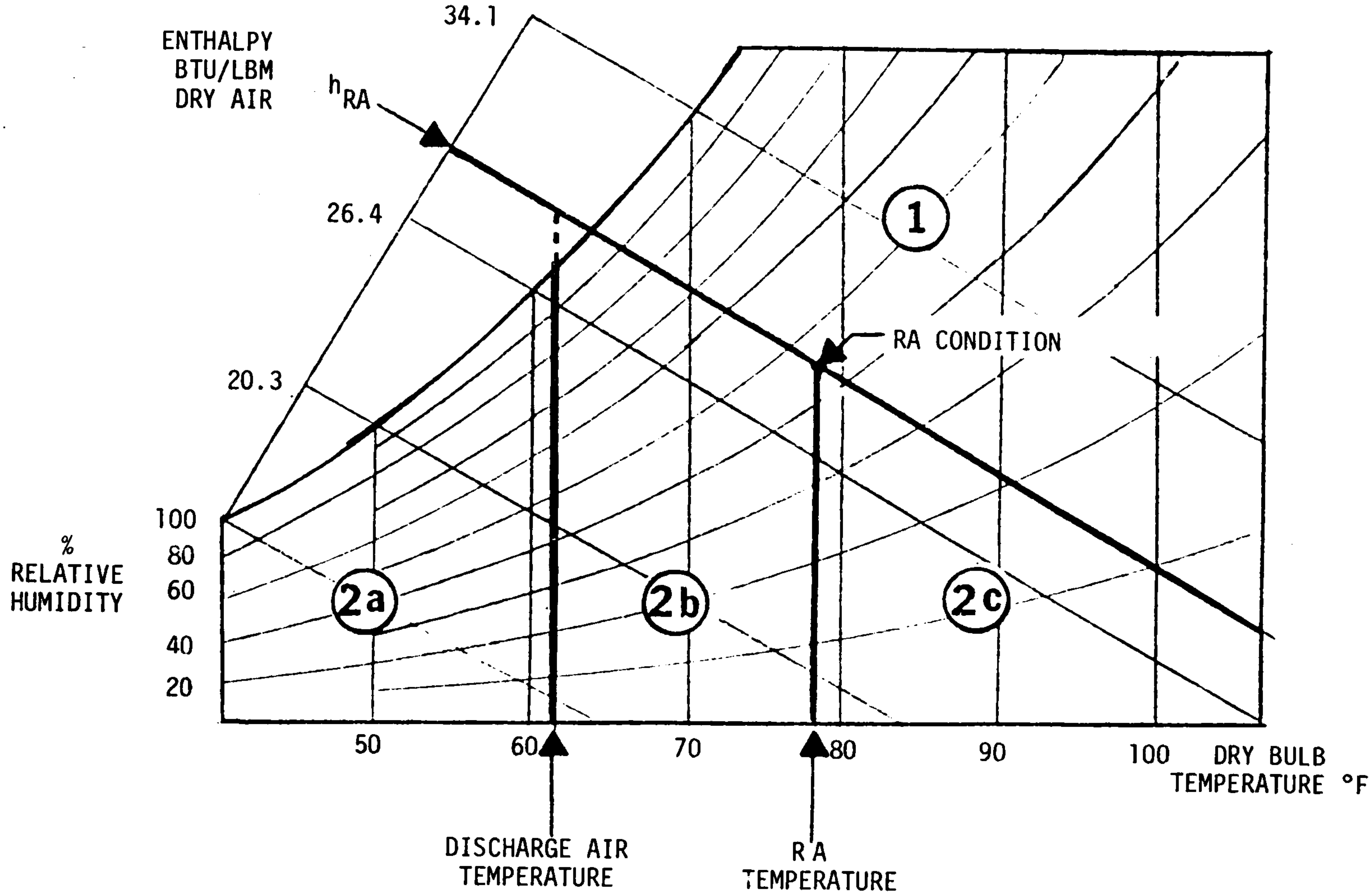


Figure 13. Psychrometrics of Enthalpy Logic.

conditions in region 2b, maximum OA is used since the OA dry bulb temperature is greater than the desired discharge air dry bulb temperature. For OA conditions in region 2c maximum RA should be used because the cooling unit employs a dry coil. The actual energy consumption for operation in this region depends upon the nature of the air conditioning process which takes place at the heat exchanger. Based on engineering weather data for Knoxville, Tennessee[9] the conditions for operation in region 2c rarely occur during the cooling season. Thus the control action specified for OA conditions in region 2c has little impact upon the overall operation in this application.

A subroutine was written for calculating enthalpy having been given the dry bulb temperature and relative humidity. A task performing the enthalpy control using the subroutine was developed and debugged using the laboratory system. The relative humidity values were simulated in the host computer. A listing of the task and subroutine is given in Appendix B.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

Modifications to a VAV system applying an improved digital control strategy have been presented. The VAV system operation and existing problems have been explained in detail.

The design and operation of the improved control strategy incorporating zone temperature feedback was discussed. The configuration and testing of the digital control system in the laboratory has been described. A cooling system model was built and two different simulation results have been presented. The control loop tuning method was included. A method for verifying and documenting regulation compliance automatically and continuously has been given.

A detailed discussion of the hardware used to implement the control strategy was provided. Advantages in selecting the MACSYM 2 computer and several of its important features have been summarized. A remote data acquisition system configuration was presented along with a method for sensing temperatures using integrated circuit transducers.

The organization of the software which executes the control strategy has been described. The tasks comprising the software have been discussed in detail. Control diagnostics were included in the execution of the system software. As operating experience is gained, the diagnostics should be extended to include all new problems encountered.

The design has been summarized for the mechanical changes in the VAV system necessary for outside air usage. A strategy for enthalpy control logic was presented. The outside air modification using enthalpy control will provide a dynamic and optimum method for OA/RA damper control. A conservative estimate of energy which will be saved from the outside air modification is 1855 million B.T.U.'s annually.[10] This savings is based on ten years of United States Weather Bureau data for Knoxville, Tennessee.

Further analysis should be made into the correlation between the method for computing the composite zone temperature feedback and the resulting cooling system response. An optimum representation of the zone conditions needs to be developed. Criteria including regulation compliance, occupant comfort, and system response needs to be integrated for this analysis.

Finally, the entire KUB Middlebrook Pike Facility needs to be studied to develop an improved centralized digital control strategy for all of the heating, ventilation, and air conditioning systems.

LIST OF REFERENCES

LIST OF REFERENCES

1. "Emergency Building Temperature Restrictions," Federal Register. Vol. 44, No. 130, July 5, 1979, pp. 39354-39369.

2. Roose, Robert W. "133 Ways to Save Energy in New and Existing Buildings," Handbook of Energy Conservation for Mechanical Systems in Buildings. Van Nostrand Reinhold Company, New York, NY, 1978, pp. 581-584.

3. Smith, Cecil L. "Control Algorithms," Digital Computer Process Control. Intext Educational Publishers, New York, NY, 1972, pp. 136-183.

4. "System/360 Continuous System Modeling Program," User's Manual GH 20-0367-4, Program Number 360A-CX-16X. IBM Corporation, Technical Publications, White Plains, NY.

5. MACSYM 2 System Manual. Analog Devices, Inc., Norwood, MA, January 1980.

6. Data Acquisition Products Catalog. Analog Devices, Inc., Norwood, MA, 1978, pp. 251-258.

7. Shavit, Gideon. "Enthalpy Control Systems: Increased Energy Conservation," Handbook of Energy Conservation for Mechanical Systems in Buildings. Roose, Robert W., Editor. Van Nostrand Reinhold Company, New York, NY, 1978, pp. 449-460.

8. Flanagan, Robert W. "Application and Adjustment of Controls for Energy Economy," Handbook of Energy Conservation for Mechanical Systems in Buildings. Roose, Robert W., Editor. Van Nostrand Reinhold Company, New York, NY, 1978, pp. 243-245.

9. Facility Design and Planning Engineering Weather Data. Departments of the Air Force, the Army, and the Navy, July 1, 1978.

10. Energy Conservation Control. Johnson Controls, Inc., Milwaukee, WI, June 1977, p. IX-31.

11. Olgesby, David R. "A Digital Simulation of a Dual-Duct Air Handling System," M.S. Thesis, The University of Tennessee, Knoxville, Tennessee, August 1977.

12. Speckhart, Frank H. and Green, Walter L. A Guide to Using CSMP - The Continuous System Modeling Program. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1976.

APPENDIXES

APPENDIX A

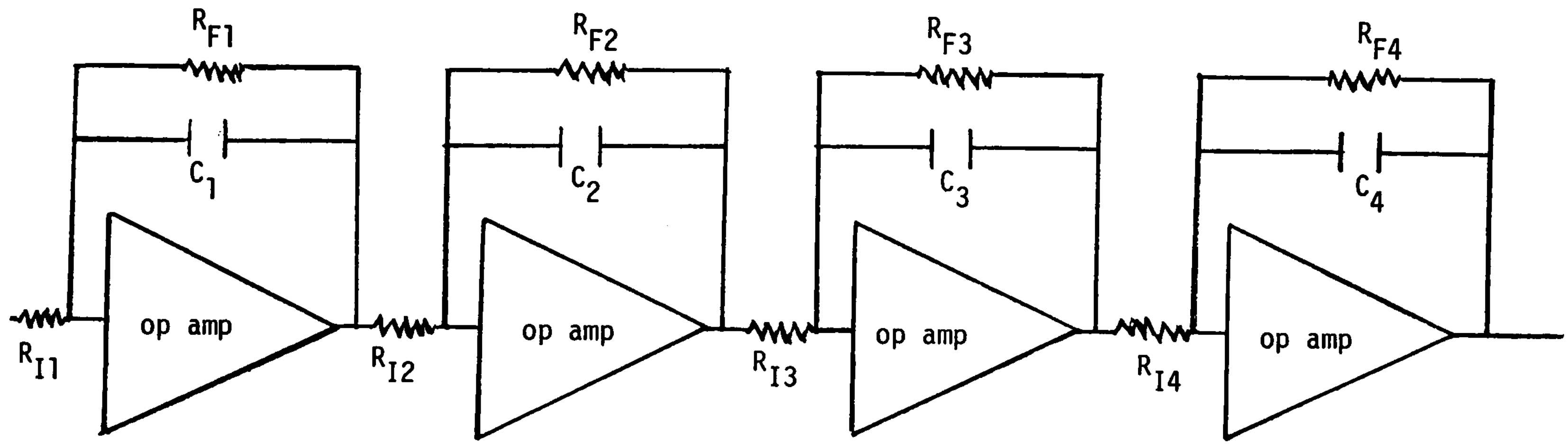
COOLING SYSTEM SIMULATION

The cooling system model was configured by combining two models. First, a cooling coil was simulated with a first-order-lap-plus-dead-time model from equation 2. The model was built on an analog manifold using operational amplifiers. The values for the gain, dead-time, and first order lag time constant for the model were selected based on data from a previous cooling coil simulation by Olgesby.[11] The model paramaters used and operational amplifier circuit is shown in Figure 14. The open loop unit step response was used to graphically determine the model parameters as described by Smith.[3] The circuit component values were adjusted until the desired model parameters were achieved.

The second model simulated the cooling system zone temperature response. Again a first-order-lag-plus-dead-time model was obtained in a manner similar to the first model. The resulting operational amplifier circuit and model parameters are shown in Figure 15.

The cooling coil and zone models were then combined as shown in Figure 16 to simulate the complete system. The MACSYM 2 was interfaced to the manifolds to perform the proportional control for the cooling coil loop, and the open loop unit step response for the complete system model was then obtained. The parameters for an overall first-order-lag-plus-dead-time model were determined graphically as illustrated in Figure 17. The parameters obtained were: θ = 60 seconds, τ = 191 seconds, and K = -0.8167.

Next, for tuning purposes the PI controller parameters were calculated using IAE coefficients for set-point changes as described by

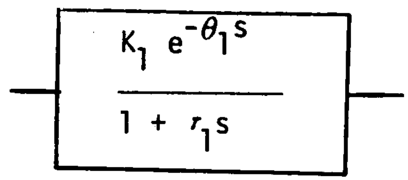


$R_{I1} = 1\ M\Omega$
$R_{F1} = 1\ M\Omega$
$C_1 = 8.54\ \mu F$

$R_{I2} = 1\ M\Omega$
$R_{F2} = 1\ M\Omega$
$C_2 = 5.32\ \mu F$

$R_{I3} = 1\ M\Omega$
$R_{F3} = 1\ M\Omega$
$C_3 = 2\ \mu F$

$R_{I4} = 1\ M\Omega$
$R_{F4} = 1\ M\Omega$
$C_4 = 2\ \mu F$



$K_1 = 1.05$ (neg. sign is taken care of in zone model)
$\theta_1 = 5.3$ seconds
$\tau_1 = 14.6$ seconds

Figure 14. Cooling Coil Model.

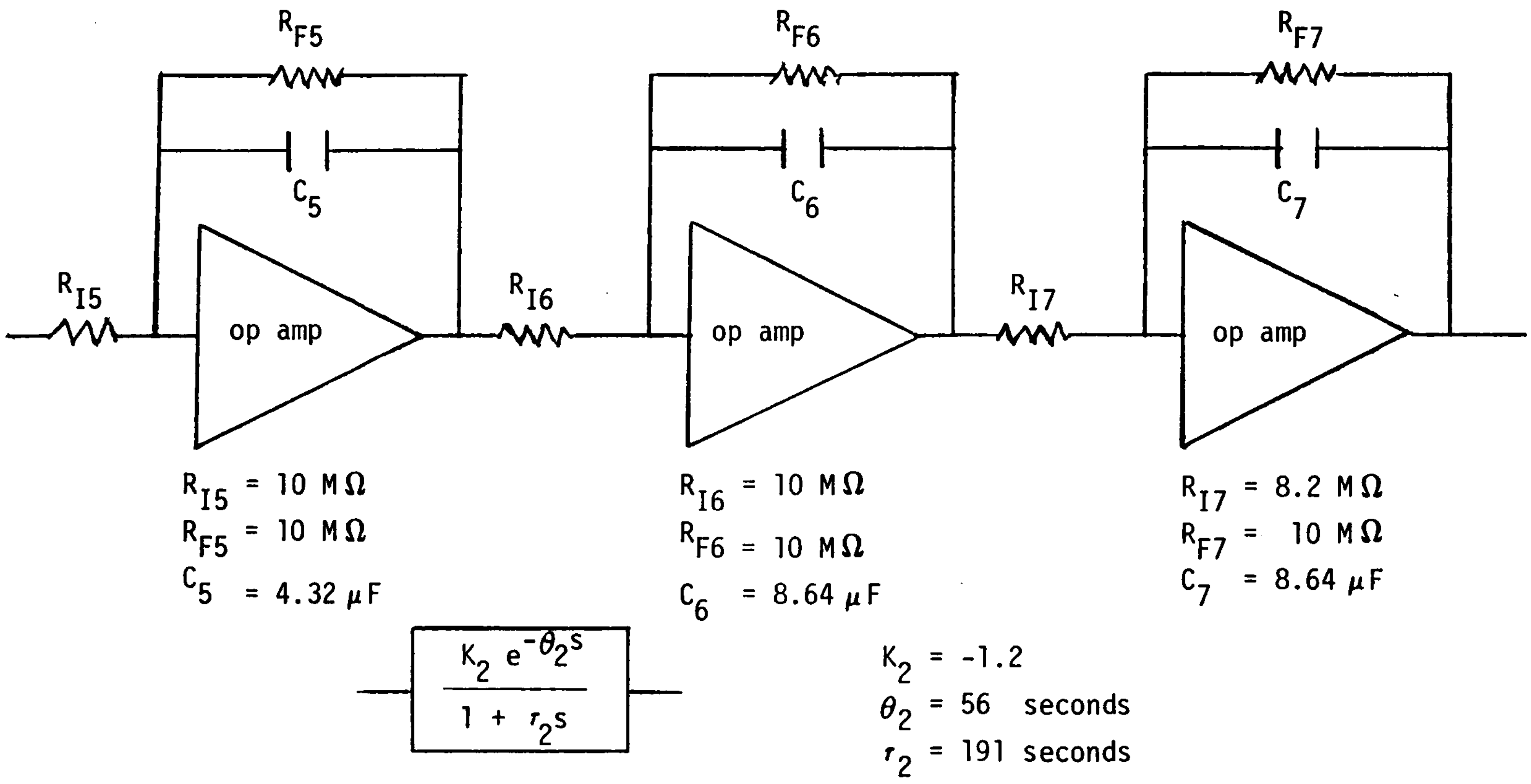
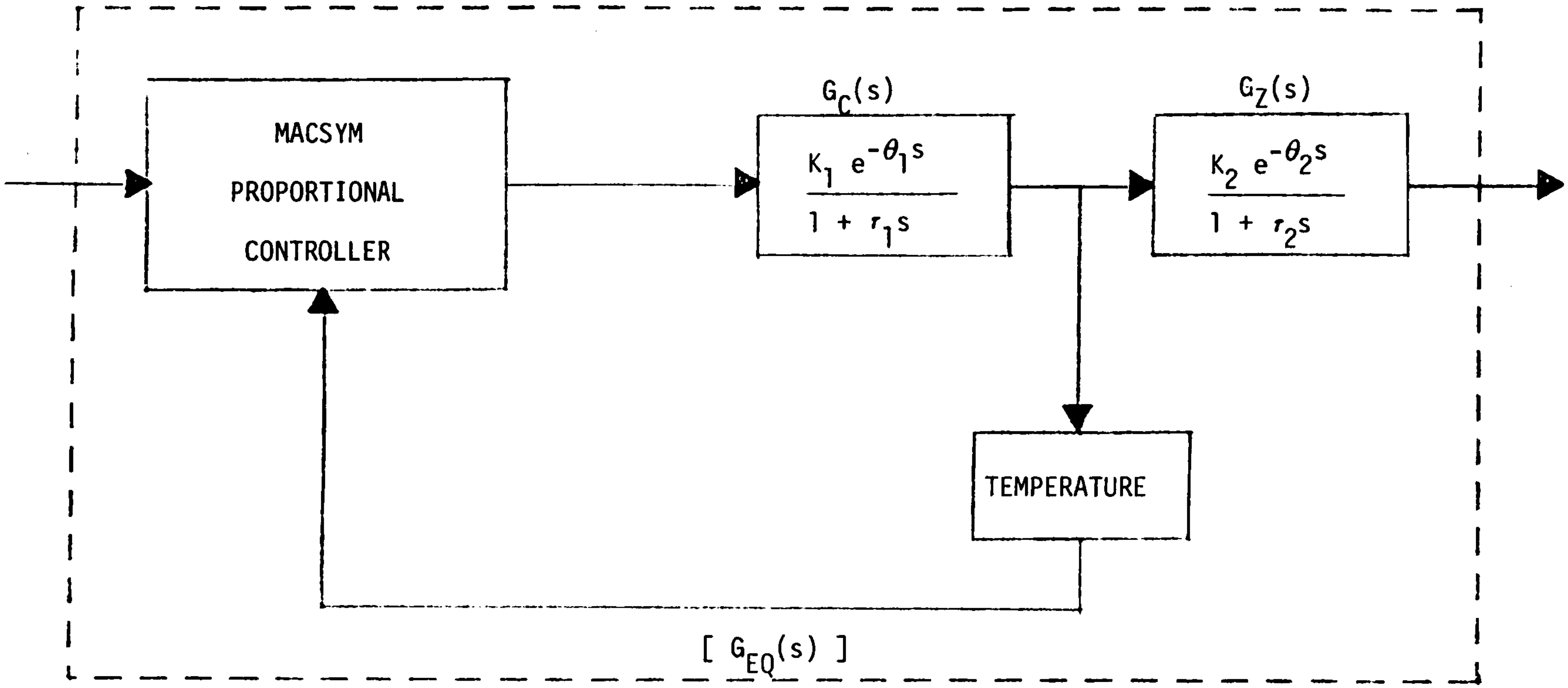


Figure 15. Zone Response Model.



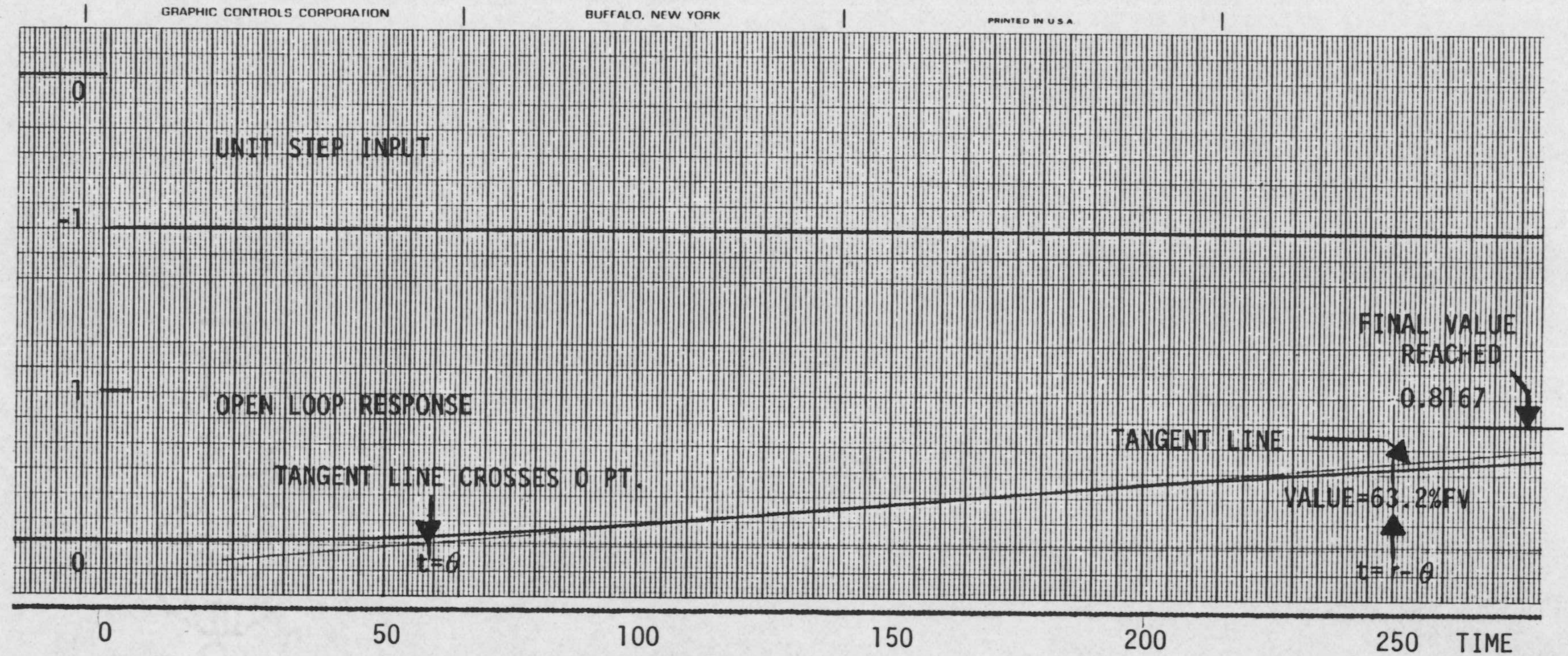
$$G_{EQ}(s) = \frac{K_{EQ}\, e^{-\theta_{EQ} s}}{1 + \tau_{EQ} s}$$

$K_{EQ} = -.8167$

$\theta_{EQ} = 60$ seconds

$\tau_{EQ} = 191$ seconds

Figure 16. Complete System Model.



θ = 60 seconds τ = 191 seconds K = -0.8167

Figure 17. Graphically Determining Model Parameters.

Smith.[3] The controller parameters include the proportional gain, K_c, and the reset time T_i. The coefficients for the proportional mode are A = 0.758 and B = -0.861. The proportional gain is determined from

$$Y = A\,(\frac{\theta}{\tau})^B \qquad (3)$$

and

$$Y = K\,K_c \qquad (4)$$

where each term is defined as previously mentioned. The coefficients for the reset mode are A = 1.02 and B = -0.323. The reset time is determined from two additional relationships.

$$Y = A + B\,(\frac{\theta}{\tau}) \qquad (5)$$

and

$$Y = \frac{\tau}{T_i}$$

Inserting the model parameters into equations 3-6, the controller gain and reset time were found to be K_c = 2.504 and T_i = 206.97 respectively.

At this point the cooling system model was simulated on an IBM 360 computer using CSMP.[12] The CSMP simulation verified the correct operation of the PI algorithm used. The relatively quick turn around of the IBM 360 runs enabled the tuning of the controller to be performed in a short period of time. The PI controller applied to the system model used the calculated controller parameters for initial values. The system to be simulated was written in block diagram form using a separate block for each term as required for CSMP simulation (see Figure 18). The CSMP program used is listed in Figure 19. The controller parameters were adjusted in successive runs of the simulation until the

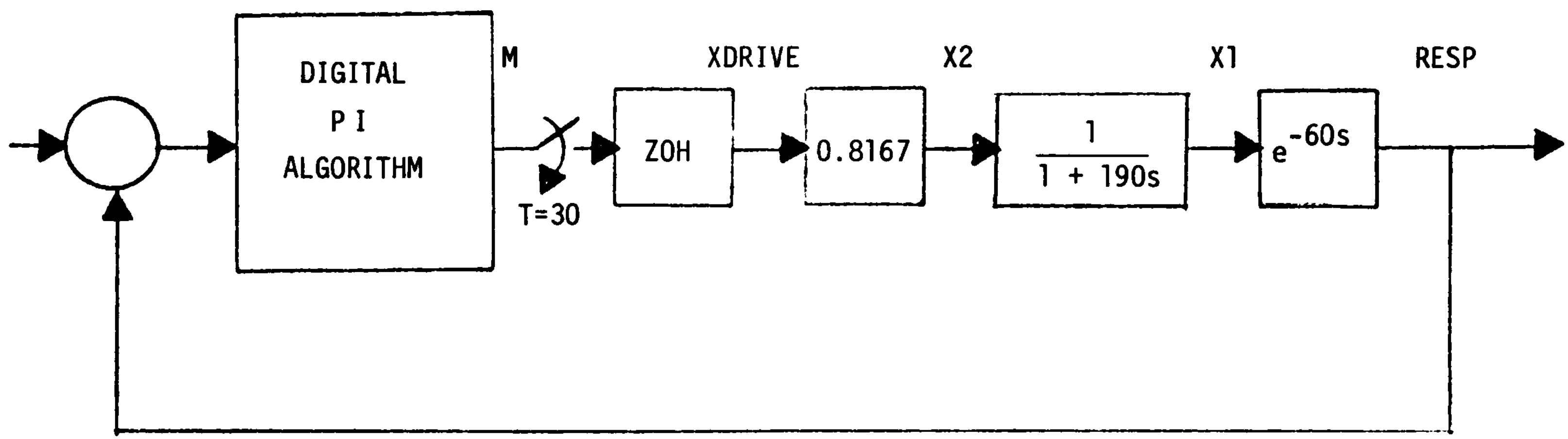


Figure 18. CSMP System Block Diagram.

```
*  SIMULATION OF PI CONTROLLER FOR COOLING COIL-ZONE SYSTEM

INITIAL

PARAM  KC=2.05,  TI=195.0
PARAM  K=0.8167, TAU=190.0, THETA=60.0, T=30.0
INCON  ERROR1=0.0, MR=0.0

DYNAMIC

INPUT=STEP(0.0)
ERROR=INPUT-RESP
RESP=DELAY(20, THETA, X1)
X1=REALPL(0.0, TAU, X2)
X2=K*XDRIVE
XDRIVE=ZHOLD(SWITCH, M)
SWITCH=IMPULS(0.0, T)

PROCEDURE  M , ERROR1 = PI(ERROR,SWITCH)

IF(SWITCH.NE.1.0) GO TO 10
IF(KEEP.NE.1) GO TO 10
ERROR1=ERROR1+ERROR

M = KC * (ERROR + T/TI * (ERROR1)) + MR

  10     CONTINUE

ENDPRO
METHOD  RKSFX

TIMER  FINTIM=750.0, OUTDEL=15.0, DELT=0.1
LABEL  UNIT STEP RESPONSE FOR PI CONTROLLER ...
APPLIED TO THE COOLING COIL SYSTEM
PRTPLT  RESP
END
STOP
```

Figure 19. CSMP Program Listing.

desired unit step response was obtained. The final parameters selected were K_c = 2.05 and T_i = 195.0. Figure 19 shows the unit step response obtained from the simulation with the final parameters. The responses for several different values of sampling time were evaluated using system response and the required computer time as criteria. The sampling time selected was 30 seconds.

The final simulation step was completed next. The MACSYM 2 was used to apply the cascade PI control to the combined system model on the analog manifolds. Figure 20 illustrates the unit step response obtained using the MACSYM 2 and the final controller parameters used. Figure 21 illustrates the laboratory model unit step response.

UNIT STEP RESPONSE FOR PI CONTROLLER APPLIED TO THE COOLING COIL SYSTEM

TIME	RESP
.0	.0
15.000	.0
30.000	.0
45.000	.0
60.000	.0
75.000	.14664
90.000	.28215
105.00	.42693
120.00	.56072
135.00	.70390
150.00	.83621
165.00	.93666
180.00	1.0295
195.00	1.0884
210.00	1.1428
225.00	1.1613
240.00	1.1783
255.00	1.1690
270.00	1.1604
285.00	1.1352
300.00	1.1119
315.00	1.0824
330.00	1.0552
345.00	1.0291
360.00	1.0051
375.00	.98682
390.00	.96998
405.00	.96057
420.00	.95186
435.00	.95009
450.00	.94845
465.00	.95198
480.00	.95524
495.00	.96150
510.00	.96728
525.00	.97406
540.00	.98034
555.00	.98614
570.00	.99151
585.00	.99558
600.00	.99933
615.00	1.0015
630.00	1.0035
645.00	1.0040
660.00	1.0046
675.00	1.0041
690.00	1.0037
705.00	1.0027
720.00	1.0018
735.00	1.0006
750.00	.99966

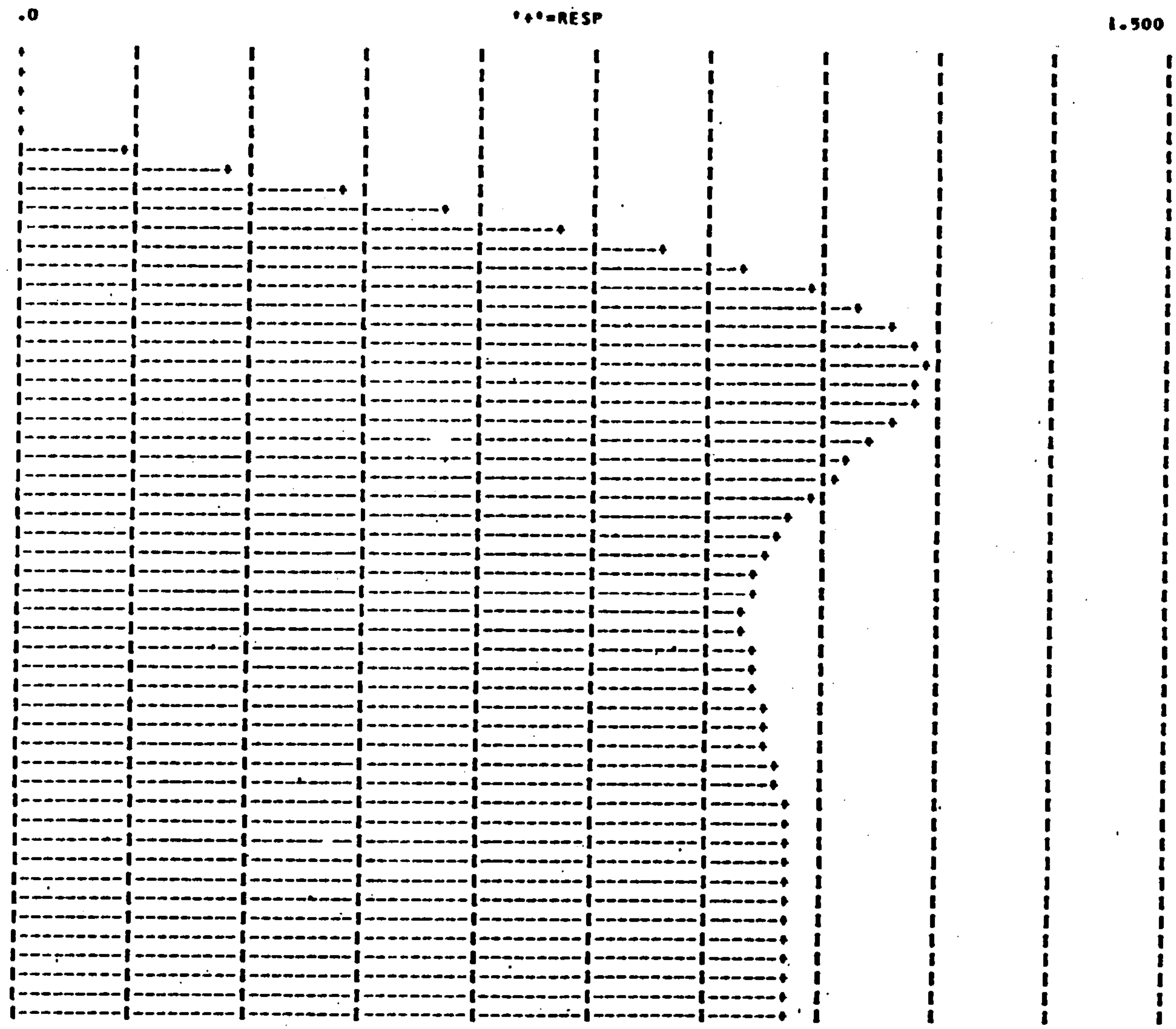


Figure 20. CSMP Simulation Unit Step Response.

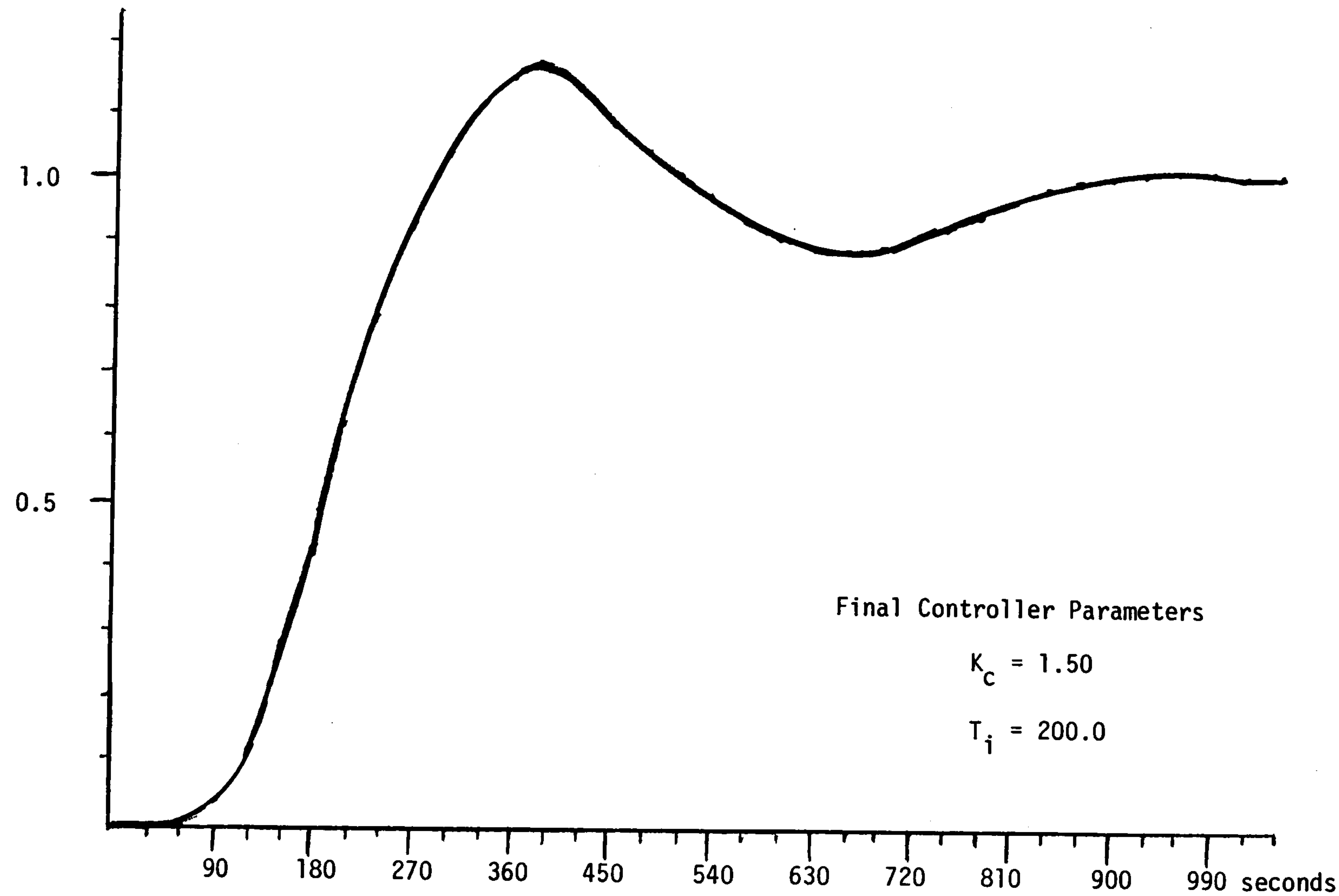


Figure 21. Laboratory Model Unit Step Response.

APPENDIX B

SOFTWARE LISTINGS

The MACBASIC software language incorporates several extensions to the Dartmouth BASIC for measurement and control functions. These extensions are summarized in Figure 22.

The software program for the laboratory system simulation was divided into separate tasks for the various functions performed. Figure 23 provides the listings of the software tasks.

MACBASIC Extensions

OPENR:n s$	Opens device or file specified by s$ for reading on channel n.
OPENW:n s$	Opens device or file specified by s$ for writing on channel n.
OPENA:n s$	Opens device or file specified by s$ for appending on channel n.
CLOSE [:n]	Closes channel n or all channels.
SAVE ARRAY v(1) s$	Saves the data from array v in file or device specified by s$.
LOAD ARRAY v(1) s$	Loads data from file or device specified by s$ into array v.
GOSUB $\ell(v_1[,v_2 \ldots v_n])$	Goes to subroutine at statement number ℓ with formal parameters v_1-v_n available for use by the subroutine.
DECLARE $(v_1[,v_2 \ldots v_n])$	Declares the variables v_1-v_n of this subroutine to be substituted by variables from the calling GOSUB statement.
ON ERROR $v_1,\ell[,v_2']$	If a runtime error occurs, place the error code in v_1, the statement number which caused the error in v_2' and go to ℓ.
INPUT[:n] $v[\$]_1[,v[\$]_2 \ldots v[\$]_n]$	Inputs from the console or channel n the values of variables v_1 to v_n or strings $v\$_1$ to $v\$_n$.
PRINT[:n] $v[\$]_1[,v[\$]_2 \ldots v[\$n]$	Prints on the console or channel n, the value of variables v_1 to v_n or the strings $v\$_1$ to $v\$_n$.
PRINT USING[:n] $A\$,v_1[,v_2 \ldots v_n]$	Prints on the console or channel n, the string A$ with the values of variables v_1 to v_n in the spaces indicated by the format groups in A$.
PNT[:n] v	Prints the 8-bit byte in v.
INP [:n] v	Inputs an 8-bit byte to v from the console or channel n.
PTIME[:n]	Prints the date and time on the console or channel n.
STIME v_1,v_2,v_3	Sets the time to v_1 hours, v_2 minutes, and v_3 seconds.
GTIME v_1,v_2,v_3	Gets the current time: hours into v_1, minutes into v_2, and seconds into v_3.
SDATE v_1,v_2,v_3	Sets the date to v_1 month, v_2 day, and v_3 year.
GDATE v_1,v_2,v_3	Gets the current date: month in v_1, day into v_2, and year into v_3.

NOTE: [] *Indicates optional parameter or type delimiter.*

Variable Types

v	numeric
v'	integer
v(n)	array
v$	string

Figure 22. MACBASIC Extensions.[5]

Process I/O Statements

AIN(s,c[,m,g])	Accepts an analog input from channel c of the analog card located in slot s in the mode (differential or single ended) as specified by m and the gain indicated by g.
AOT(s,c)	Sends an analog signal to channel c of the analog output card located in slot s.
DIN(s,fb[,lb,p])	Accept 1 to 16 digital inputs as defined by fb (first bit) and lb (last bit) from the digital input card located in slot s with a polarity p.
DOT(s,fb[,lb,p])	Send digital outputs as defined by fb (first bit) and lb (last bit) to the digital output card located in slot s with a polarity p.
SCAN (a(i),n[,t,m,g]) INTO x(j)[ON TRIGGER trigger[,timeout]]	Stores n analog readings, spaced t seconds apart, from the channels specified by a (i) into the one dimensional array specified by x(j). m and g are the same as in AIN above.
COLLECT (s,c,n[,t,m,g]) INTO x(j)]ON TRIGGER trigger[,timeout]]	Stores n analog readings, spaced t seconds apart, from channel c of the analog input card located in slot s into the one dimensional array x(j).
AVG (s,c,n[,t,m,g])	Returns the average of n analog readings, spaced t seconds, from channel c of the analog input card located in slot s.

Multitasking Statements

TASK n, ℓ[,p]	Defines task number n, starting at line ℓ, with priority p.
ACTIVATE n	Starts execution of task n.
ACTIVATE n PERIOD v	Activates task n every v seconds.
ACTIVATE n ON EVENT (v_1,v_2,v_3)	Activates n when the occurence of event v has taken place on channel v_2 of Process Interrupt card in slot v_1.
DISMISS	When completed, places the task executing this statement on alert status for next activation.
KILL n / SELF / ALL	Removes the definition of task n (or all tasks) from the system.
SUSPEND n / SELF	Places task n or all tasks in a suspended state.
ALERT n ALL / SELF / ALL	Places specified task n or all tasks in an alert state, ready for execution.
PRIORITY n	Allows a task to change its priority to n.
RESERVE n	Reserves the I/O channel number n for the calling task.
RELEASE n	Releases any reservation on channel n.
BREAK	Suspends execution of a task and prints a message on the console to allow debugging. Task can be continued with a CON command.

Figure 22. (Continued).

```
100.REM ** TASK 1 ** MAIN PROGRAM **
102 REM ** THE SOFTWARE PROGRAM FOR THE LABORATORY SIMULATION FOLLOWS
104 REM ** ALL INITIALIZATION & ACTIVATION WAS PERFORMED INTERACTIVELY FOR THE SIMULATION
106 REM **
110 TASK 2,2020,23
120 TASK 3,2320,24
130 TASK 4,5030,25
140 TASK 5,5530,40
150 TASK 7,6020,44
160 TASK 8,6420,43
170 TASK 9,6630,42
180 TASK 10,6780,22
190 TASK 16,3600,20
200 TASK 17,4100,30
210 DIM D0(4),F1(2),F2(31,2),F3(160,2),F4(160,2),F9(3),S1(2),T(4),X(10)
220 END
```

Figure 23. Software Listing.

```
2000 REM ** TASK 2 ** RDAS COMMUNICATIONS TO GET ZONE FEEDBACK **
2002 REM ** PROIRITY = 23
2004 REM ** ACTIVATED PERIODICALLY WITH A PERIOD OF 60 SECONDS
2006 REM ** FUNCTION :TALK TO RDAS & OBTAIN ZONE FEEDBACK DATA
2008 REM ** VARIABLES : F2(X,Y)= ARRAY FOR FEEDBACK DATA
2010 REM **                F9(F)= ERROR FLAG - ARRAY OVERFLOW
2020 F9(2)=0.0
2030 ON ERROR 101,2180
2040 OPENR :2 "$QTI:2"
2050 OPENR :4 "$QTI:3"
2060 ECHO OFF :2 ECHO OFF :4
2070 IF F9(1)=1.0 THEN  GOTO 2130 REM ** USE B-MICRO
2080 REM ** USE A-MICRO
2090 PNT :2 2 PNT :2 65  REM ** CALL FOR DATA
2100 INPUT :3 F2(31,2)
2110 GOTO 2150
2120 REM ** USE B-MICRO
2130 PNT :4 2 PNT :4 65
2140 INPUT :5 F2(31,2)
2144 ECHO ON :2 ECHO ON :4
2146 CLOSE :2 CLOSE :3 CLOSE :4 CLOSE :5
2150 DISMISS
2160 GOTO 2020
2170 REM ** ARRAY OVERFLOW ERROR
2180 ON ERROR 101,2180
2190 F9(2)=F9(2) + 1  REM ** SET CTR
2200 IF F9(2)>1 THEN  A0=1 M9'=4 A9=1.0 SUSPEND SELF REM ** SET MODE TO PNEUMATIC & ALARM
2210 GOTO 2070
```

Figure 23. (Continued).

```
2300 REM ** TASK 3 ** RDAS COMMUNICATION TO GET ALL ZONE TEMPERATURES
2302 REM ** PRIORITY = 15
2304 REM ** ACTIVATED PERIODICALLY WITH A PERIOD OF 5 MINUTES
2306 REM ** FUNCTION : TELK TO RDAS & OBTAIN ALL ZONE TEMPERATURES
2308 REM ** F3,4(X,Y) = ARRAY FOR ALL ZONE TEMP'S
2310 REM ** F9(F) = ERROR FLAGS
2320 PRIORITY 24
2330 F9(3)=0
2340 ON ERROR 101,2550
2350 OPENR :6 "$QTI:2"
2360 OPENR :8 "$QTI:3"
2370 ECHO OFF :6 ECHO OFF :8
2380 IF F9(1)=1 THEN  GOTO 2450 REM ** USE B-MICRO
2390 REM ** USE A-MICRO
2400 PNT :6 2 PNT :6 66 REM ** CALL FOR 1ST HALF OF DATA
2410 INPUT :7 F3(160,2)
2420 PNT :6 2 PNT :6 67 REM ** CALL FOR 2ND HALF OF DATA
2430 INPUT :7 F4(160,2)
2440 GOTO 2490
2445 REM ** USE B-MICRO
2450 PNT :8 2 PNT :8 66
2460 INPUT :9 F3(160,2)
2470 PNT :8 2 PNT :8 67
2480 INPUT :9 F4(160,2)
2490 ECHO ON :6 ECHO ON :8
2500 CLOSE :6 CLOSE :7 CLOSE :8 CLOSE :9
2510 OFF ERROR
2520 PRIORITY 15
2530 DISMISS
2540 GOTO 2320
2545 REM ** ARRAY OVERFLOW ERROR
2550 F9(3)=F9(3)+1
2560 IF F9(3)>1 THEN  A0=1 M9'=5 A9=1 SUSPEND SELF
2570 GOTO 2380
```

Figure 23. (Continued)

```
5000 REM ** TASK 4 : CHILLED WATER VALVE PI CONTROL (OUTER LOOP)
5002 REM ** PRIORITY = 25
5004 REM ** ACTIVATED BY SYSTEM INITIALIZATION FOR PERIODIC EXECUTION WITH A PERIOD OF 30 SECONDS.
5006 REM ** FUNCTION : PERFORM PI CONTROL FOR OUTER CASCADE LOOP FOR AC-10 & AC-11 CHILLEDWATER VALVES (CWV)
5008 REM ** CALLS 2 SUBROUTINES FOR LOOP CONSTANTS & VARIABLE INITIALIZATION
5010 REM ** PRIOR TO ACTIVATING THIS TASK CWV'S MUST BE MANUALLY ADJUSTED TO INITIAL POSITION
5012 REM ** VARIABLES : X(#) = MANUAL MODE CONDITION ARRAY
5014 REM **              F1(1),F1(2) = COMPOSITE ZONE FEEDBACK TEMPERATURES FOR AC-10 & AC-11.
5016 REM **              T(3),T(4) = SET POINTS FOR AC-10 & AC-11 ZONE TEMPERATURES
5018 REM **
5020 REM ** AC-10 PI LOOP
5022 REM **
5030 IF X(1)=0.0 THEN  GOTO 5120
5040 IF X(5)=0.0 THEN  GOSUB 5300 (T3,L3,K3,T2,E2) REM ** INITIALIZE LOOP
5050 R3= AVG(0,1,0,3) REM ** ZONE SET POINT ... (WILL BE R3=T(3) FOR ACTUAL SYSTEM)
5060 C3= AVG(2,0,0,3) REM ** COMPOSITE ZONE FEEDBACK ... (WILL BE C3=F1(1) FROM RDAS)
5070 E3=R3+C3 REM ** C3 FROM MANIFOLD IS NEG.
5080 E2=E2+E3 REM ** CUM. ERROR
5090 IF ABS(E2) > L3 THEN  A2=1.0 M9'=1 E2=SGN(E2)*L3  REM ** WINDUP LIMIT & ALARM
5100 M3= K3 * (E3 + T3/T2 * E2)  REM ** PI CONTROLLER OUTPUT USED BY INNER P LOOP (AC-10)
5108 REM **
5110 REM ** AC-11 PI LOOP
5112 REM **
5120 IF X(2)=0.0 THEN  GOTO 5200  REM ** LOOP IN MANUAL MODE
5130 IF X(6)=0.0 THEN  GOSUB 5400 (T4,L4,K4,T5,E5)  REM ** INITIALIZE PI LOOP
5140 R4= AVG(0,1,0,3)  REM ** ZONE SET POINT ... (WILL BE R4=T(4) ...)
5150 C4= AVG(2,0,0,3)' REM ** COMPOSITE ZONE FEEDBACK ... (WILL BE C4=F1(2) FROM RDAS)
5160 E4=R4 + C4  REM ** C4 FROM MANIFOLD IS NEG.
5170 E5=E5 + E4  REM ** CUM. ERROR
5180 IF ABS(E5) > L4 THEN  A2=1.0 M9'=2 E5=SGN(E5) * L4  REM ** WINDUP LIMIT & ALARM
5190 M4= K4 * ( E4 + T4/T5 * E5)  REM ** PI CONTROLLER OUTPUT USED BY INNER P LOOP (AC-11)
5200 DISMISS
5210 GOTO 5030
```

Figure 23. (Continued)

```
5284 REM ** SUBROUTINES FOR TASK 4 **
5285 REM **
5286 REM ** SUBROUTINE TO INITIALIZE PI LOOP PARAMETERS (AC-10)
5288 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 5300 (T3,L3,K3,T2,E2)
5290 REM ** T3=SAMPLE TIME, L3=WINDUP LIMIT, K3=GAIN, T2=RESET TIME, E2=CUM. ERROR
5292 REM **
5300 DECLARE (T3,L3,K3,T2,E2)
5310 T3=30.0
5320 L3=100.0
5330 K3=1.5
5340 T2=200.0
5350 E2=0.0
5360 X(5)=1.0  REM ** RESET THE MODE MARKER
5370 RETURN
5386 REM **
5388 REM ** SUBROUTINE TO INITIALIZE PI LOOP PARAMETERS FOR AC-11 CWV
5390 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 5400 (T4,L4,K4,T5,E5)
5392 REM ** T4=SAMPLE TIME, L4=WINDUP LIMIT, K4=GAIN, T5=RESET TIME, E5=CUM. ERROR
5394 REM **
5400 DECLARE (T4,L4,K4,T5,E5)
5410 T4=30.0
5420 L4=100.0
5430 K4=1.5
5440 T5=200.0
5450 E5=0.0
5460 X(6)=1.0  REM ** RESET THE MODE MARKER
5470 RETURN
```

Figure 23. (Continued)

```
5500 REM ** TASK 5 : CHILLED WATER VALVE (CWV) P CONTROL - INNER LOOP
5502 REM ** PRIORITY = 40
5505 REM ** ACTIVATED BY SYSTEM INITIALIZATION FOR PERIODIC EXECUTION WITH A 1 SECOND PERIOD
5506 REM ** FUNCTION : PERFORM P LOOP CONTROL FOR INNER CASCADE LOOPS FOR AC-10 & 11 CWV'S
5507 REM ** TASK 4 MUST EXECUTE AT LEAST ONCE PRIOR TO ACTIVATING THIS TASK
5508 REM ** VARIABLES :  K6,K7 = GAIN
5510 REM **              C6,C7=DISCHARGE AIR TEMP'S
5514 REM **              E6,E7 = ERROR'S
5516 REM **
5518 REM ** AC-10
5520 REM **
5530 IF X(9)=0.0 THEN  GOTO 5600  REM ** IN MANUAL MODE
5540 K6=1.0
5550 C6= AVG(0,0,0,3)  REM ** READ IN DISCHARGE AIR TEMP.
5560 E6=M3-C6  REM ** ERROR
5570 AOT (1,0)=K6 * E6  REM ** P-OUTPUT FOR AC-10
5580 REM **
5590 REM ** AC-11
5596 REM **
5600 IF X(10)=0.0 THEN  GOTO 5650  REM ** IN MANUAL MODE
5610 K7=1.0
5620 C7= AVG(0,0,0,3)  REM ** READ IN DISCHARGE AIR TEMP
5630 E7=M4-C7  REM ** ERROR
5640 AOT(1,0)=K7 * E7  REM ** P-OUTPUT FOR AC-11
5650 DISMISS
5660 GOTO 5530
```

Figure 23. (Continued)

```
6000 REM ** TASK 7 ** CONTROL PANEL SYSTEM OPERATION SELECTION
6002 REM ** PRIORITY = 44
6004 REM ** ACTIVATED BY INTERRUPT FROM CHANGING POSITION OF SELECTOR SWITCH FOR OPERATION MODE
6006 REM ** FUNCTION : DETERMINE WHAT MODE WAS SELECTED & PERFORM THE NECESSARY SWITCHING
6008 REM **            THIS TASK ENABLES THE INTERRUPT FOR MANUAL CONTROL BUTTONS WHEN MANUAL MODE IS SELECTED
6010 REM ** VARIABLES : S1(1)= SWITCH POSITION
6012 REM **             S1(2)= MODE POSITION
6020 IF A9=1 THEN  GOTO 6170  REM ** KEEP IN PNEUMATIC MODE - SYS. FAULT
6030 CO=0
6040 S1(1)=DIN(6,0,2,0)
6050 IF S1(1)=2 THEN  GOTO 6120 REM ** MANUAL MODE
6060 IF S1(1)= 4 THEN  GOTO 6170 REM ** PNEUM. MODE
6070 IF S1(1)=1 THEN  GOTO 6220 REM ** AUTO MODE
6080 REM ** IF S1 IS NOT 1,2,OR 4 THEN INPUT FROM S.S. IS INVALID - INCREMENT CTR. & READ AGAIN
6090 CO=CO+1
6100 IF CO>1 THEN  AO=1 M9'=1 GOTO 6170 REM ** SWITCH TO PNEUM. & ALARM FOR S.S. MALFUNCTION
6110 GOTO 6040
6120 DOT(9,0,2,0)=2  REM ** TURN ON MANUAL LED
6130 DOT(9,3,3,0)=1  REM ** SWITCH SOL. VALVE TO COMPUTER-MANUAL POSITION
6140 S1(2)=2
6150 ACTIVATE 8 ON EVENT  (8,1,1,1)  REM ** ENABLE MANUAL PB INTERRUPT
6160 GOTO 6350
6170 DOT(9,3,3,0)=0  REM ** SWITCH SOL. VALVE TO PNEUM. POSITION
6180 DOT(9,0,2,0)=4  REM ** TURN ON PNEUM. LED
6190 SUSPEND 8  REM ** DISABLE MANUAL PB'S
6200 S1(2)=4
6210 GOTO 6360
6215 REM ** CHECK TO SEE IF MODE WAS JUST IN PNEUM.
6220 IF S1(2)=4 THEN  AO=1 M9'=3 GOTO 6360
6225 REM ** CHECK FOR UNRESET MANUAL CONTROLS
6230 IF X(1)=0 THEN  GOTO 6320
6240 IF X(2)=0 THEN  GOTO 6320
6250 IF X(3)=0 THEN  GOTO 6320
6260 IF X(4)=0 THEN  GOTO 6320
6270 DOT(9,0,2,0)=1  REM ** TURN ON AUTO LED
6280 DOT(9,3,3,0) = 1  REM ** SWITCH VALVE TO COMP.-MAN. POSITION
6290 SUSPEND 8  REM ** DISABLE MAN. PB'S
6300 S1(2)=1
6310 GOTO 6360
6320 A2=1.0 M9'=2 REM ** ALARM FOR TRYING TO SWITCH TO AUTO FROM PNEUM.
6330 ACTIVATE 8 ON EVENT  (8,1,1,1)  REM ** ENABLE MAN. PB'S
6340 DISMISS
6350 GOTO 6220
6360 ACTIVATE 7 ON EVENT  (8,0,1,1)  REM ** RE-ENABLE TASK 7 INTERRUPT
6370 DISMISS
6380 GOTO 6020
```

Figure 23. (Continued)

```
6400 REM ** TASK 8 ** SWITCHING FOR MANUAL ADJUSTMENT CONTROLS
6402 REM ** PRIORITY = 43
6404 REM ** ACTIVATED BY INTERRUPT FROM MANUAL PB'S BEING DEPRESSED
6406 REM ** FUNCTION : DETERMINE WHICH PB WAS DEPRESSED & SWITCH CORRESPONDING CONTROL LOOP
6408 REM ** VARIABLES : S1(2) = OPERATION MODE ACTIVATED
6410 REM **                DIN(6,6,9,0)=THE 4 CONTROL LOOP MANUAL PB'S
6412 REM **                DOT(9,4,7,0)= THE 4 CONTROL LOOP MANUAL PB LIGHTS
6420 IF DIN(6,4,4,0)=1 THEN  GOTO 6470
6430 IF DIN(6,5,5,0)=1 THEN  GOTO 6510
6440 IF DIN(6,6,6,0)=1 THEN  GOTO 6530
6450 IF DIN(6,7,7,0)=1 THEN  GOTO 6560
6460 GOTO 6580
6470 REM
6480 IF X(1)=1.0 THEN  X(1)=0.0 X(5)=0.0 DOT(9,4,4,0)=1 GOTO 6580
6490 IF X(1)= 0.0 THEN  X(1)=1.0 DOT(9,4,4,0)=0 GOTO 6580
6500 REM **
6510 IF X(2)=1.0 THEN  X(2)=0.0 X(6)=0.0 DOT(9,5,5,0)=1 GOTO 6580
6520 IF X(2)=0.0 THEN  X(2)=1.0 DOT(9,5,5,0)=0 GOTO 6580
6525 REM
6530 IF X(3)=1.0 THEN  X(3)=0.0 X(7)=0.0 DOT(9,6,6,0)=1 GOTO 6580
6540 IF X(3)=0.0 THEN  X(3)=1.0 DOT(9,6,6,0)=0 GOTO 6580
6550 REM
6560 IF X(4)=1.0 THEN  X(4)=0.0 X(8)=0.0 DOT(9,7,7,0)=1 GOTO 6580
6570 IF X(4)=0.0 THEN  X(4)=1.0 DOT(9,7,7,0)=0
6580 IF S1(2)=2 THEN  ACTIVATE 8 ON EVENT  (8,1,1,1)
6590 DISMISS
6600 GOTO 6420
```

Figure 23. (Continued)

```
6610 REM ** TASK 9 ** DISPLAY SELECT SWITCHING
6612 REM ** PRIORITY = 42
6614 REM ** ACTIVATED BY INTERRUPT FROM PUSH BUTTONS BEING DEPRESSED
6616 REM ** FUNCTION : DETERMINE WHICH BUTTON WAS DEPRESSED & SWITCH THAT FUNCTION
6618 REM ** VARIABLES : DO(1)= # OF PB FUNCTION SELECTED
6620 REM **              DIN(5,0,7,0) & DOT(4,0,7,0) ARE PB'S AND LIGHTS FOR DISPLAY
6630 IF DIN(5,0,0,0)=1.0 THEN  DOT(4,0,7,0)=255 WAIT 2 DOT(4,0,7,0)=DO(1) GOTO 6730
6640 IF DIN(5,1,1,0)=1 THEN  DO(1)=1.0 DOT(4,0,7,0)=1 GOTO 6720
6650 IF DIN(5,2,2,0)=1.0 THEN  DO(1)=2 DOT(4,0,7,0)=2 GOTO 6720
6660 IF DIN(5,3,3,0)=1.0 THEN  DO(1)=3 DOT(4,0,7,0)=4 GOTO 6720
6670 IF DIN(5,4,4,0)=1.0 THEN  DO(1)=4 DOT (4,0,7,0)=8 GOTO 6720
6680 IF DIN(5,5,5,0)=1.0 THEN  DO(1)=5 DOT (4,0,7,0)=16 GOTO 6720
6690 IF DIN(5,6,6,0)=1.0 THEN  DO(1)=6 DOT(4,0,7,0)=32 GOTO 6720
6700 IF DIN(5,7,7,0)=1.0 THEN  DO(1)=7 DOT(4,0,7,0)=64 GOTO 6720
6710 GOTO 6730
6720 ACTIVATE 10 PER 3
6730 ACTIVATE 9 ON EVENT (8,2,1,1)
6740 DISMISS
6750 GOTO 6630
```

Figure 23. (Continued)

```
6760 REM ** TASK 10 ** DISPLAY UPDATE TASK
6762 REM ** PRIORITY = 24
6764 REM ** ACTIVATED PERIODICALLY WITH A PERIOD OF 3 SECONDS
6766 REM ** FUNCTION : UPDATE THE INFORMATION BEING DISPLAYED
6768 REM ** VARIABLES : AOT(3,0 & 1) = DISPLAY OUTPUTS
6770 REM **                DO(1) = DISPLAY FUNCTION PB SELECTED
6772 REM **                DO(2-5)= TEMPORARY STORAGE FOR DISPLAY FUNCTIONS BEING COMPUTED
6780 IF DO(1)=1 THEN  GOTO 6850
6790 IF DO(1)=2 THEN  GOTO 6880
6800 IF DO(1)=3 THEN  GOTO 6910
6810 IF DO(1)=4 THEN  GOTO 6940
6820 IF DO(1)=5 THEN  GOTO 6970
6830 IF DO(1)=6 THEN  GOTO 7000
6840 IF DO(1)=7 THEN  GOTO 7030
6850 DO(2)=AVG(0,2,0,3) * 1.8 + 32.0
6860 DO(3)=(AVG(0,3,0,3) * 1.8 ) + 32.0
6870 GOTO 7050
6880 DO(2)=(AVG (0,6,0,3) * 1.8) + 32.0
6890 DO(3)=AVG(0,7,0,3) * 46.78
6900 GOTO 7050
6910 DO(2)=(AVG(0,4,0,3) * 1.8) + 32.0
6920 DO(3)=AVG(0,5,0,3) * 46.78
6930 GOTO 7050
6940 DO(2)=(F2(2,1)*10 + F2(2,2)) * 1.8 +32.0
6950 DO(3)=(F2(17,1) * 1.8) + 32
6960 GOTO 7050
6970 DO(2)=(F2(1,1)*10+F2(1,2))*1.8 +32.0
6980 DO(3)=(F2(16,1) * 10 + F2(16,2)) * 1.8 +32.0
6990 GOTO 7050
7000 DO(2)=(F2(3,1) * 10 + F2(3,2)) * 1.8 + 32.0
7010 DO(3)=(F2(18,1) * 10 + F2(18,2)) * 1.8 +32.0
7020 GOTO 7050
7030 DO(2)=(AVG(0,0,0,3) * 1.8) + 32.0
7040 DO(3)=(AVG(0,1,0,3) * 1.8) + 32.0
7050 DO(4)= (DO(2) - 20) * .2
7060 AOT(3,0)= (DO(4) - 4) / 1.6
7070 DO(5)=(DO(3) - 20) *.2
7080 AOT(3,1)=(DO(5) - 4) / 1.6
7090 DISMISS
7100 GOTO 6780
```

Figure 23. (Continued)

```
3500 REM ** TASK 16 : OUTSIDE AIR ENTHALPY LOGIC
3510 REM ** PRIORITY = 20
3520 REM ** ACTIVATED BY SYSTEM INITIALIZATION FOR PERIODIC EXECUTION WITH 5 MINUTE PERIOD
3530 REM ** FUNCTION : EXECUTE OA/RA DAMPER CONTROL BASED ON ENTHALPY LOGIC FOR AC-10 & AC-11 UNITS
3540 REM ** IF OPERATION IS IN REGION 2A ACTIVATE TASK 17 FOR 1 SECOND PERIODIC EXECUTION
3550 REM ** VARIABLES:
3552 REM                  H=ENTHALPY
3554 REM                  T=TEMPERATURES
3556 REM                  R=RELATIVE HUMIDITY
3558 REM                  X1,X2=START(0)/RUN(1) MARKERS FOR TASK 17 FOR EACH AC UNIT
3560 REM                   T(1)=DISCHARGE AIR TEMPERATURE SETPOINT
3570 REM ** OUTPUT TO DAMPER MOTORS : 0.0=100% RA ; 10.0=100% OA
3580 REM
3600 X1=0 X2=0
3610 REM ** GET OA TEMP, RH, & H.
3620 T8= AVG(0,4,0,3)
3630 R8= AVG(0,5,0,3)
3640 GOSUB 1500 (H8,T8,R8) REM ** GET OA ENTHALPY
3650 REM ** GET RA TEMP,RH, & H.
3660 T9= AVG(0,6,0,3)
3670 R9= AVG(0,7,0,3)
3680 GOSUB 1500 (H9,T9,R9) REM ** GET RA ENTHALPY
3690 REM ** DETERMINE CONTROL REGION
3700 IF H8>H9 THEN  GOTO 3750
3710 IF T8>T9 THEN  GOTO 3750
3720 IF T8>T(1) THEN  GOTO 3800 REM ** T(1)=DISCHARGE AIR TEMP. SET POINT
3730 ALERT 17 REM ** START TASK 17 FOR REGION 2A OPERATION
3740 GOTO 3840
3750 IF X1=1 THEN  SUSPEND 17 REM ** STOP REGION 2A OPERATION
3760 X1 = 0 X2 = 0
3770 AOT(1,0)= 0.0 REM ** AC-10 ON 100% RA
3780 AOT(1,1)= 0.0 REM ** AC-11 ON 100% RA
3790 GOTO 3840
3800 IF X1=1 THEN  SUSPEND 17 REM ** STOP REGION 2A OPERATION
3810 AOT(1,0)= 10.0 REM ** AC-10 ON 100% OA
3820 AOT(1,1)= 10.0 REM ** AC-11 ON 100% OA
3830 DISMISS
3840 GOTO 3610
```

Figure 23. (Continued)

```
1400 REM ** SUBROUTINE FOR COMPUTING THE ENTHALPY OF AIR
1402 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 1500 (HO,TO,RO)
1404 REM ** HO=ENTHALPY ; TO=DRY BULB TEMPERATURE ; RO=RELATIVE HUMIDITY
1406 REM **
1500 DECLARE (H1,T,R)
1510 REM ** CONSTANTS FOR CALCULATIONS **
1520 A=-7.90298 B=5.02808 C=-1.3816E-7 D=11.344
1530 E=8.1328E-3 F=-3.49149 G=-9.09718 H=-3.56654
1540 P=0.876793 Q=6.0273E-3 PO=29.921 FO=1.0045
1550 REM ** CALCULATE SATURATED VAPOR PRESSURE (VO) **
1560 T1=(T+459.688)/1.8
1570 IF T1<273.16 THEN  GOTO 1640 REM ** IF TEMP. IS LESS THAN FREEZING USE OTHER EQUATION
1580 Z=373.16/T1
1590 S=A*(Z-1.0)+B*LOG(Z)/LOG(10.0)+C*(10.0^(D*(1.0-1.0/Z))-1.0)
1600 S=S+E*(10.0^(F*(Z-1.0))-1.0)
1610 VO=PO*10.0^S
1620 GOTO 1670
1630 REM ** EQUATION FOR TEMP. < FREEZING **
1640 Z=273.16/T1
1650 S=G*(Z-1.0)+H*LOG(Z)/LOG(10.0)+P*(1.0-1.0/Z)+LOG(Q)/LOG(10.0)
1660 VO=PO*10.0^S
1670 REM ** CALCULATE HUMIDITY RATIO (WO) **
1680 W1=0.622*FO*VO/(PO-FO*VO)
1690 R1=R*0.01
1700 DO=R1*(PO-FO*VO)/(PO-R1*FO*VO)
1710 WO=W1*DO
1720 REM ** CALCULATE THE ENTHALPY (H1) **
1730 H1= 0.24 * T + (1061.0 +0.444* T) * WO
1740 RETURN
```

Figure 23. (Continued)

```
4000 REM ** TASK 17 : OA/RA REGION 2A OPERATION
4010 REM ** PRIORITY = 30
4020 REM ** ACTIVATED BY TASK 16 FOR PERIODIC EXECUTION WITH PERIOD OF 1 SECOND
4030 REM ** FUNCTION : PERFORM PI CONTROL FOR OA/RA DAMPERS FOR AC-10 & AC-11
4040 REM ** THIS TASK USES 4 SUBROUTINES
4050 REM ** VARIABLES : X(#)= ARRAY FOR MANUAL MODE CONDITION FOR EACH CONTROL LOOP
4070 REM
4100 IF X(3)=0 THEN  GOTO 4140
4110 IF X(7)=0.0 THEN  GOSUB 4200(P1,P2,P3,P4)
4120 IF X1=0.0 THEN  GOSUB 4200 (P1,P2,P3,P4)
4130 GOSUB 4600 (P1,P2,P3,P4)
4140 IF X(4)=0 THEN  GOTO 4180
4150 IF X(8)=0.0 THEN  GOSUB 4400 (P8,P9,Q1,Q2)
4160 IF X2=0.0 THEN  GOSUB 4400 (P8,P9,Q1,Q2)
4170 GOSUB 4800 (P8,P9,Q1,Q2)
4180 DISMISS
4190 GOTO 4100
```

Figure 23. (Continued)

```
4193 REM ** SUBROUTINES FOR TASK 17
4194 REM **
4195 REM ** SUBROUTINE TO INITIALIZE THE PI LOOP ALGORITHM FOR AC-10 OA/RA DAMPER
4196 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 4200 (P1,P2,P3,P4)
4197 REM ** P1=SAMPLE TIME,P2=CONTROLLER GAIN, P3=RESET TIME, P4=CUM. ERROR
4198 REM **
4200 DECLARE (P1,P2,P3,P4)
4210 P1=1.0
4220 P2=1.4
4230 P3=21.0
4240 P4=0.0 REM ** CUM. ERROR INITIALLY ZERO
4250 AOT(1,0)=0.0 REM ** AC-10 ON 100% RA
4260 WAIT 10 REM ** LET DAMPER RESPOND
4270 X1=1 X(7)=1 REM ** RESET MARKERS
4280 RETURN
4388 REM **
4390 REM ** SUBROUTINE TO INITIALIZE THE PI LOOP ALGORITHM FOR AC-11 OA/RA DAMPER
4392 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 4400 (P8,P9,Q1,Q2)
4394 REM ** P8=SAMPLE TIME,P9=CONTROLLER GAIN, Q1=RESET TIME, Q2=CUM. ERROR
4396 REM **
4400 DECLARE (P8,P9,Q1,Q2)
4410 P8=1.0
4420 P9=1.4
4430 Q1=21.0
4440 Q2=0.0
4450 AOT(1,1)=0.0 REM ** AC-11 ON 100% RA
4460 WAIT 10 REM ** LET DAMPER RESPOND
4470 X2=1 X(8)=1 REM ** RESET MARKERS
4480 RETURN
4588 REM **
4590 REM ** SUBROUTINE TO PERFORM PI ALGORITHM FOR AC-10 OA/RA DAMPER
4592 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 4600 (P1,P2,P3,P4)
4594 REM ** P1=SAMPLE TIME,P2=GAIN,P3=RESET TIME, P4=CUM. ERROR
4596 REM **
4600 DECLARE (P1,P2,P3,P4)
4610 P5=AVG(0,2,0,3) REM ** READ IN MA TEMP.
4620 P6=P5-T(2) REM ** T(2)=SET POINT FOR MA CONTROLLER
4630 P4=P4+P6 REM ** CUM ERROR
4640 P7= P2 * (P6 + P1/P3 * P4) REM ** CONTROLLER OUTPUT
4650 AOT(1,0)=P7
4660 RETURN
4788 REM **
4790 REM ** SUBROUTINE TO PERFORM PI ALGORITHM FOR AC-11 OA/RA DAMPER
4792 REM ** THIS SUBROUTINE EXECUTES UPON A GOSUB 4800 (P8,P9,Q1,Q2)
4794 REM ** P8=SAMPLE TIME, P9=GAIN, Q1=RESET TIME, Q2=CUM. ERROR
4796 REM **
4800 DECLARE (P8,P9,Q1,Q2)
4810 Q3= AVG(0,3,0,3) REM **READ MA TEMP
4820 Q4=Q3-T(2) REM ** ERROR; T(2)=CONTROLLER SET POINT
4830 Q2=Q2+Q4 REM ** CUM ERROR
4840 Q5 = P9 * (Q4 + P8/Q1 * Q2) REM ** CONTROLLER OUTPUT
4850 AOT(1,1)= Q5
4860 RETURN
```

Figure 23. (Continued)

VITA

Richard Lee Kissell was born August 28, 1951 in Lima, Ohio. He resided in Columbus Grove, Ohio with his parents while attending elementary and junior high schools. Three years of high school were spent attending Put-in-Bay High School when his family moved to South Bass Island (Ohio) on Lake Erie. In June 1969 he graduated from Port Clinton High School in Port Clinton, Ohio.

In August 1969 Mr. Kissell enrolled in the cooperative engineering curriculum at General Motors Institute, Flint, Michigan. Cooperative work began at that time with Central Foundry Division, Saginaw Malleable Iron Plant in Saginaw, Michigan. He received the degree of Bachelor of Electrical Engineering in July 1974 and continued working at Central Foundry Division through August 1978.

In September 1978 Mr. Kissell entered the graduate program in electrical engineering at The University of Tennessee, Knoxville. Since then he worked as a teaching assistant in digital systems and also as a research assistant in the digital control field at the Systems Research Laboratory in the Electrical Engineering Department. He received the Master of Science degree with a major in Electrical Engineering in June 1980.

Mr. Kissell will be employed with Aluminum Company of America in Alcoa, Tennessee following graduation.

He is married to the former Nancy Hogg of Saginaw, Michigan and has three children.