Greenhouse Climate Control

Dale Nelsen

Preliminary Project Description
Etec 471, Professor Todd Morton
11-21-2005

Western Washington University
EET

Introduction:

Many people in this day in age have many things going on in their lives. A lot of these activities they don’t really have the time for. For instance, my cousin Tye is in charge of a 250 acre orchard in Wenatchee, WA, and that alone is a full time job. On top of that, he also runs a large nursery 30 miles south in the tri-cities and six 30 by 160 foot greenhouses that are on the orchard. He is running all over the place throughout the week taking care of these three responsibilities. This is why I have come up this project. The system I plan to design will maintain an optimum growth environment inside the greenhouse.

I have a plan to create a climate control system that will do all the work for my cousin. This system will vary the temperature by controlling a fan, heater, and ventilation. It will also water the plants either once a day or twice a day by opening and closing a water valve at the time or times the user inputs for watering. I was going to use moisture sensor to control the watering process but I was told by my cousin that he just needed the system to water at certain time intervals. So I choose to just water on a timely basis. The user could either enter two different times or one time that they would want the plants to be watered.

Functional Description

The needs of a greenhouse are pretty straight forward. The green house needs to be a relatively constant temperature, and it would also be good to keep the correct level of humidity. Also very important, the plants in the greenhouse need to be watered at the correct time. This is exactly what my project will do. First off, there will be two sensors a humidity sensor and a temperature sensor. The outputs of these sensors will be fed into the microcontroller for processing, but not before the signals are conditioned through the use of op amps and filters. The microcontroller is responsible for receiving input data from various sensors, and making adjustments to the system in order to maintain the desired environment. This can be seen in figure one

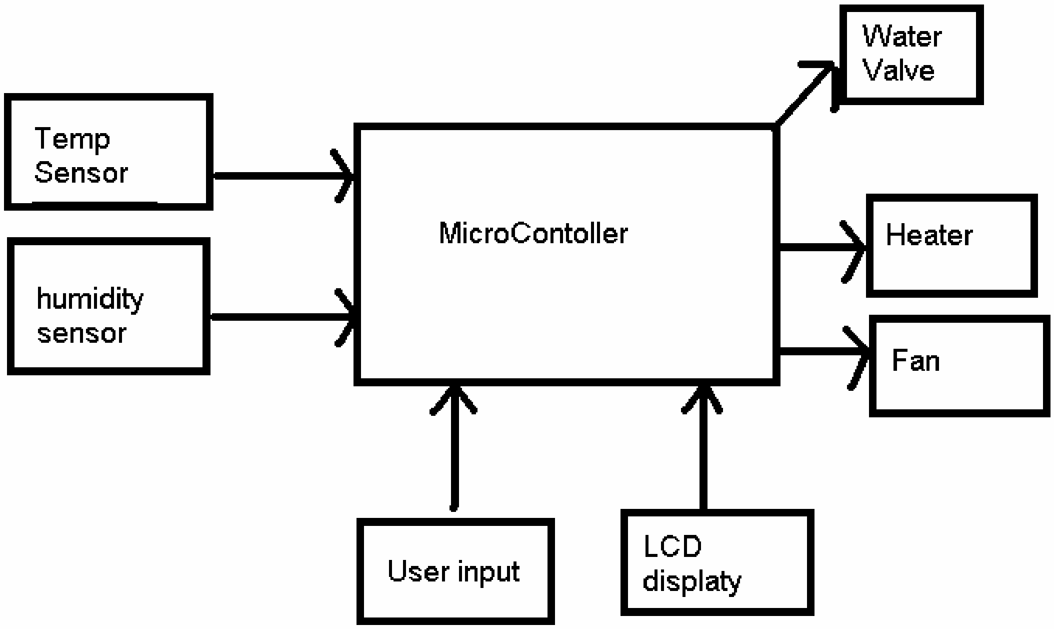

Figure One

The microcontroller not only receives data, it also needs to do something with it. For this, the user will input some parameters to which he or she wants the greenhouse to be. For instance the user wants the greenhouse to be between 70 and 75ºF. To stay with in these parameters the microcontroller will control a heater, fan, and water valve. So if the temperature was 78ºF the fan would come on, and if it were below 70ºF the heater would turn on. This climate control hardware is now powered by 120 VAC from the wall but the microcontroller will control them by using relays. Since these will remain in the on position the microcontroller will just switch on the relay and the water, heater, or fan will be on until the parameters that the user input are met.

The hardware that will be seen outside the controller box will be the temperature sensor, the humidity sensor, plug box, heater, fan, and water valve. Also on the box there will be four buttons for user input and a 2 by 16 Liquid Crystal Display. This will display the time, temperature and the humidity. It will also be the display to input all the climate parameters for the greenhouse. There will also be three Light Emitting Diodes to indicate whether or not the equipment is on. All of the hardware for this project can be seen in figure below with there dimensions (figure two).

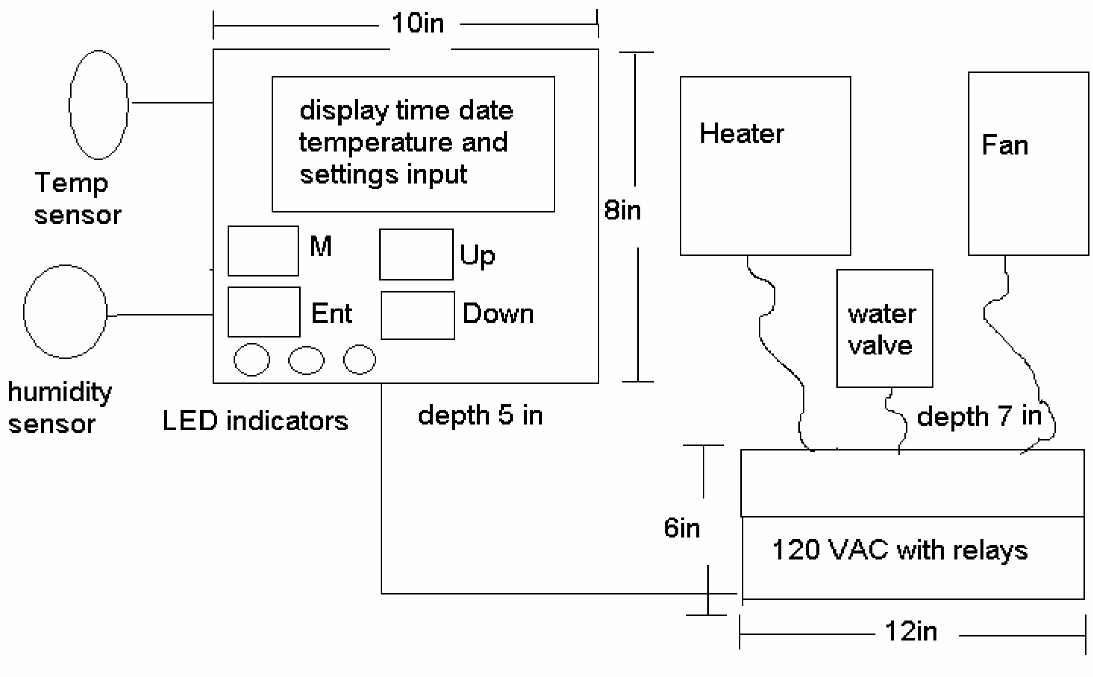

Figure Two

Detailed Functional Description

I looked at some of the different microcontrollers out there and I decided on using the MC9S12DP256. I chose this controller because I felt it had enough input/output lines so if I decided to or needed to add additional parts to my system, I would not be too limited. It also has sufficient memory for the good amount of programming that will be needed. Not to mention the fact that I will be using it in 454 and will have access to the development tools needed.

All of the Pin connections for the microcontroller are shown in the block diagram below (Figure three) and first off I will deal with the input signals and the output control signals. The output of the temperature sensor and the humidity sensor will be input into PAD00 and PAD01 of the microcontroller after their signal was conditioned. The relays for the heater, fan, and water valve will be connected to Port T 0-2. Then the indication LED's will be connected to PT3 and PT4. The LCD display will make use of Port A and Port K. Port A will be used up entirely and 0-2 of Port K. The four buttons for input into the system will use Port B 0-3. A 16 MHz crystal will also be used and will use the EXTAL XTAL. These are connected to the clock and reset generator (CRG) and provides the internal clock signals for the core and all the peripheral modules. Ports M and S will control and receive the real time clock information for the DS1511, a battery packed real time clock. The DS1511 should keep time to with in +-1 minute per month. The clock is also battery backed so the time will not be lost in case of a power loss.

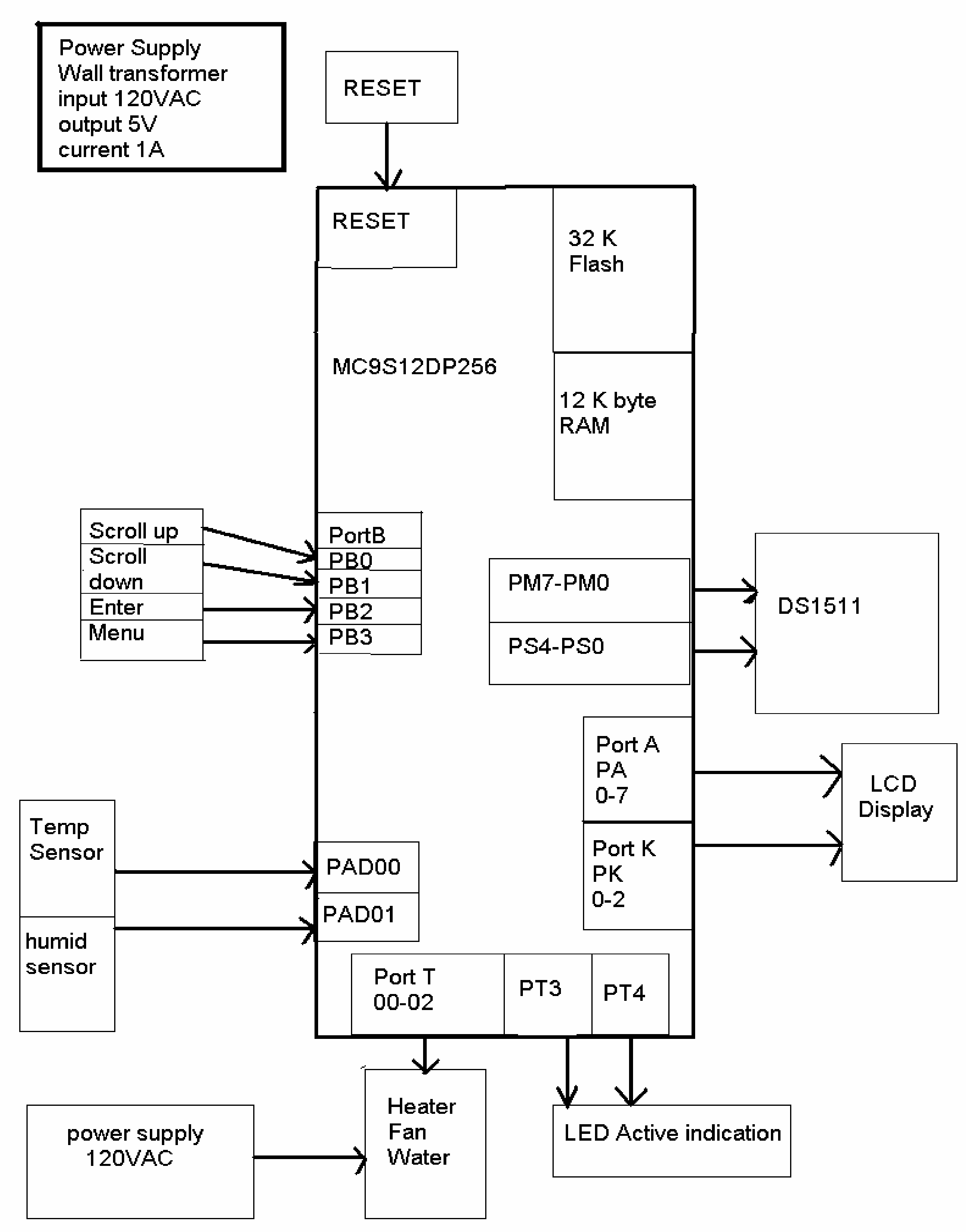

Figure Three

Software Requirements

This project requires a fair amount of complex code. For this reason, I will be using C for the entire program and it will be composed of several different modules with a kernel managing multiple tasks. Here is a preliminary list of the different modules and their tasks.

Module	**Description**
Kernal	This is a MicroC/OS is a real time kernel and will Manage the software modules with task scheduling.
Main_control	This module monitors the state of the system and controls The different hardware and monitors the sensor signal
Display	This module controls what the 2 by 16 LCD displays
Input_detect	This module keeps track of the user input from the four Buttons U, D, M, and E.
Relay_control	Controls the state of the relays to the heater, fan, and Water valve.

User Interface

The user interface will be done through the use of four buttons, M(Menu), E(Enter), D(Down), and U(Up). You can see these in figure two of the entire system. By pressing the menu button the user can enter a step by step process of inputting data into the controller. The information that will need to be entered is the time, the time he/she would like to irrigate, the ideal temperature range, and %humidity. The system will store this information and control the different devices to keep the greenhouse at an ideal climate. The flow diagram of how signals are input to the controller and output from the controller from sensors and to hardware is in figure one.

The User interface is very straight forward. It will make use of four buttons. One is a menu button. It first initializes the set up from the main screen and from there on if menu is pressed it will take you back to the main screen. When in set up mode, pressing up and down will scroll through the options and press enter will take you where you want. Then when setting the different parameters use the numbers on the key pad to enter values for the different parameters like set temp high and set temp low. The enter button is pressed when the desired value is entered. When the user has gone through a whole set up, the system will automatically refresh back to the main screen. User interface is pretty much a set by set process. The diagram below shows this process and what keys to press to move through the user interface (Figure four). A key letter is placed by the arrow to indicate what happens when a certain key is engaged. For instance from the main menu screen there is an arrow going down, to move down to the different option menu button (M) was pressed. Then from there use the up (U) and down (D) keys to scroll through.

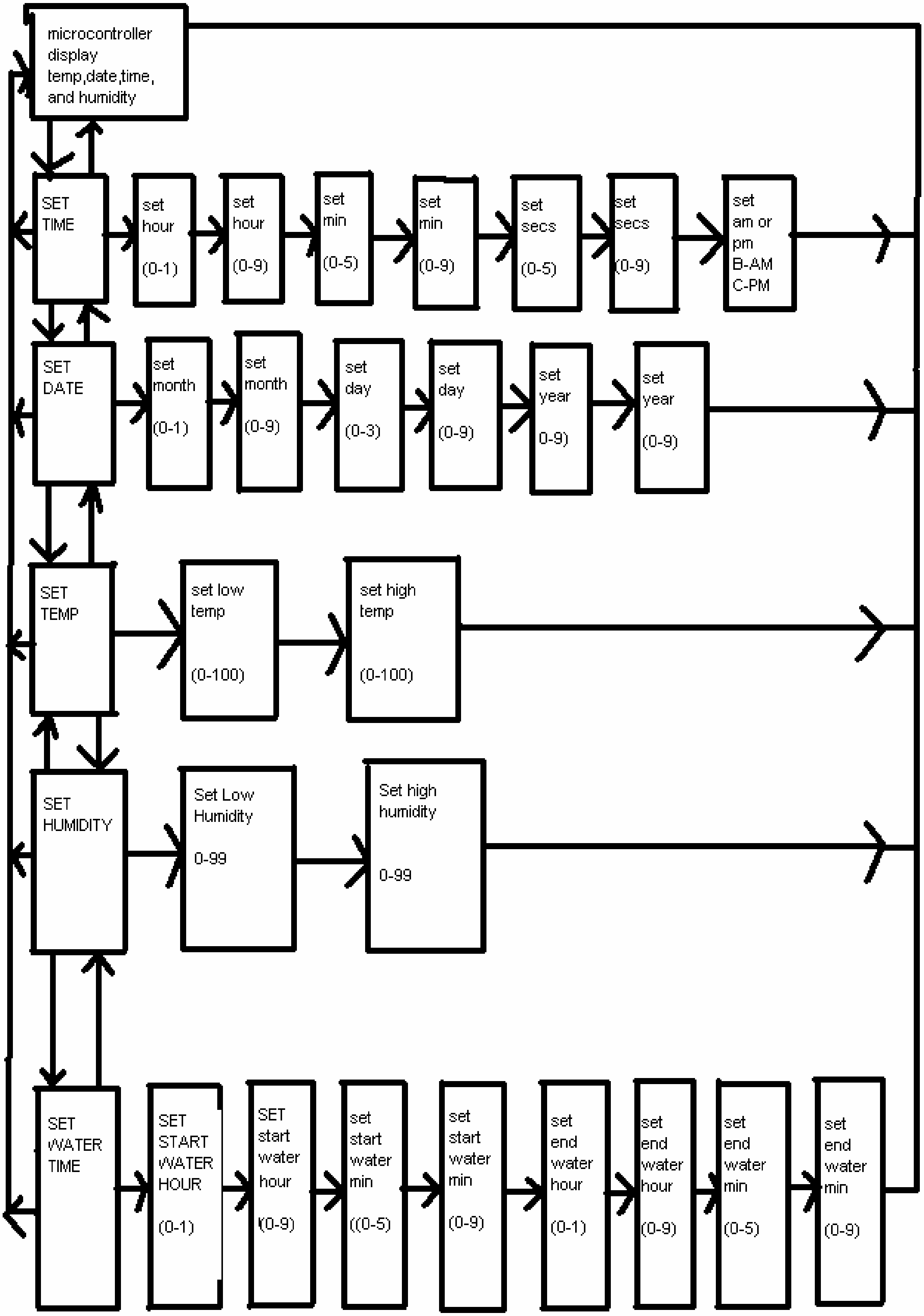
microcontroller display temp,date,time, and humidity
SET TIME
set hour (0-1)
set hour (0-9)
set min (0-5)
set min (0-9)
set secs (0-5)
set secs (0-9)
set am or pm B-AM C-PM
SET DATE
set month (0-1)
set month (0-9)
set day (0-3)
set day (0-9)
set year 0-9)
set year (0-9)
SET TEMP
set low temp (0-100)
set high temp (0-100)
SET HUMIDITY
Set Low Humidity 0-99
Set high humidity 0-99
SET WATER TIME
SET START WATER HOUR (0-1)
SET start water hour ((0-9)
set start water min ((0-5)
set start water min (0-9)
set end water hour (0-1)
set end water hour (0-9)
set end water min (0-5)
set end water min (0-9)

Figure Four

A typical user interface with this system would go a little like this. This example would take place after all the parameters had been set and the user wanted to change the temperature. When the system is running the screen is on its main screen shown in figure five, and then keys will be pressed to move through the user interface. First the Menu button would be pressed and that gets you to some different options which you could scroll through. In this case, you would hit the down key until you reached the screen set temp (figure six). From here, to get to other options, you would press enter and the display would look like figure seven. To set the high temp and the low temp just scroll up or down and press enter when the desired value is read (figure seven). Then to go back to the main running screen just press enter. Now the high temp and low temp are set. The other parameters would be set a lot like this was.

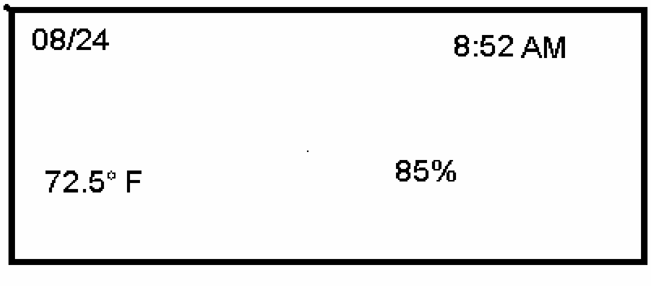

Figure five

Menu
Set temp

figure six

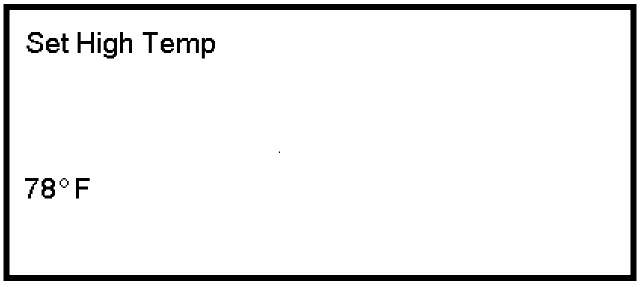

figure seven

This was just a general description of how to set the high and low temperatures for the greenhouse. The rest of the parameters will be set in a similar fashion you would just press menu and scroll until you found the parameter that you wanted to set. All the variables are set in a similar fashion using the Up and down keys to get the level you need the greenhouse to be.

Development Plan

Prototype and materials

The prototype will consist of two boxes. One box will house the outlets and the relays that power the fan, heater, and water valve. This will just have 120 VAC running into it, and also wiring from the controller box. The control box will house the microcontroller, liquid crystal display, and light emitting diodes. It will also house the signal conditioning for the two sensors. The components in the control box will be powered by a wall transformer that supplies 5V at .5A. The two sensors will be external and the wires will run into the control box. The method that will be used on the wiring for the surface mounting of the prototype will be point to point soldering. Outside of the MC9S12DP256 micro controller there will not be too much wiring. There will be the signal conditioning from the sensors, and that will consist of a few op amps and instrumentation amplifiers. The crystal oscillator and the RTC will involve a fair amount of work, and then just the indication LEDs. There may be more that will come up later that I have not thought about but there should be adequate space in the control box to house this.

Schedule

This is my detailed schedule of how I will be developing my project. After this quarter is over and my description is done, I will really start working on the project. During the winter break, I will begin ordering all the parts that I don't already have. Hopefully most of the items do not have much of a wait period for delivery. There are only a few things I need to order. Much of the equipment is found in local stores so that is just a matter of going out and purchasing the parts.

The work on my project starts at the beginning of the winter quarter. I should have most of the equipment by now and I can start assembling parts of my system. I think I will first test the parts that I ordered first to see if the are working correctly so I know if I have to order more. Then I would probably start by building the signal conditioning of the two sensors. This essentially just consists of op amps and resistors and capacitors. When they are built these are complete I will test if I have the desired outputs. Then I will start on the climate control hardware. For this I need to connect the output of the microcontroller, outlets and main wall line to the relays. When all the outside hardware is complete, I will construct the housing all of the equipment. Figure two shows the basic design.

The hardware side of the project should be completed towards the end of the winter quarter, which leaves all of spring quarter to develop all of the software. I hope to have at least started modifying the modules for the program, during the winter quarter. As of right now a have very rough idea of what the program flow will look like but I haven’t used this microcontroller before and I have not written code in C for a few years so I will have to get back into gear. ETEC 454 in the winter quarter will probably do just that by refreshing my brain and learning about the microcontroller. By the end of the spring quarter, I have scheduled three weeks for debugging and testing the system for the demonstration.

Winter Quarter

Week1	**Check outputs of sensors and build the signal conditioning**
Week2	**Finish the S.C. test the output**
Week3	**Connect the hardware(heaters, fans) to relays**
Week4	**Test this hardware**
Week5	**Design cases for system components and housing for outlets and relays.**
Week6	**Build the cases for the components and power outlets.**
Week7	**Test the hardware after enclosing it.**
Week8	**Connect all the parts to the microcontroller**
Week9	**Miscellaneous**
Week10	**Miscellaneous**
Week11	**Dead Week, look at details of the software**
Week12	**Finals Week**

Spring Quarter

Week 1	**Start the software programming Hardware should be done**
Week2	**Modify modules**
Week3	**Modify modules and Hardware Reviews**
Week4	**Write module for sensor signals Hardware Documents due**
Week5	**Calibrate the signals with the microcontroller/ software Review**

Week6	**Write module for the relays and Real time clock**
Week7	**Miscellaneous**
Week8	**Miscellaneous**
Week9	**Test system /Code Reviews**
Week10	**Test system**
Week11	**Demonstration**

Demonstration

For the demonstration I will have the system with the temperature sensor and the humidity sensors in a place where I can very the conditions to display what my project will do. To very the temperature I could maybe blow some warm air on the sensor or stick a cold object on it. For humidity I could surround the sensor in a damp cloth. I will start from scratch on the parameters like it was the first time setting the system up and I will enter the time, temperature, humidity that the system will run on. When entering these parameters I will be sure to set the irrigation time to a time that is close to the real time so the water valve will turn on during demonstration. So In the demonstration I will show that the heater will turn on and the fan will turn on at the proper times.

Project Specification

Temperature :+-1ºF
-Range 40-90°F

Humidity :+-2%
-Range 0-100%

Time
-error +- 1min/month

Display Clock----HH:MM

Display Date-----MM/DD/YY

Electrical Specifications

: 5 V supply at .5 A

: Preliminary Max draw 120 mA

Power dissipation when relays energized

These are the specifications of just the hardware not what they are drawing from the entire system.

Fan :2.5A
Heater :1500W 120 VAC I= 12.5A
Water Valve : .4 A

Preliminary Parts List

Qt	Part list	Part #	source	Price	Max current
1	microcontoller	MC9S12DP256B	Motorola	25.00	65 ma
1	Temp Sensor	LM34	national	1.00	.15 mA
1	Humidity sensor	19C6680	Honeywell	25.00	.5 ma
1	Heater		Sears	20.00	n/a
1	Fan		Lowes	10.00	n/a
3	Relays	N/A	n/a	20.00	25 mA
2	LED	67-1096-ND	Digikey	1.23	5 mA
2	Hardware housing	n/a	n/a	15.00	n/a
10	Resistors	1% ¼ W	Digikey	1.00	4 mA
3	120 V male plug	n/a	Lowes	5.00	n/a
3	120 V female plug	n/a	Lowes	5.00	n/a
1	2by16 LCD	3-1033-ND	Digikey	16.00	3 mA
1	16 MHz crystal	549-016757	Mouser	2.00	2 uA
1	Real time clock	DS1511	Digikey	12.08	15mA
			Total	147.00	120 mA