Daniel Wilson

Dan Millikin

Patrick Walker

Austin Riegsecker

Sean Mince

Project Description:

The idea of this project is to use arduinos to control the temperature in a room that has only hot and cold temperature adjustments. The arduinos should be able to communicate with each other and adjust the temperature in the room to keep it within 3 degrees of the set point. The hardware requirements will be as follows:

* 2 x Arduino Uno
* 1 x LCDKeypad shield
* 3 x Servo motor
* 2 x nRF24L01 2.4 ghz wireless transcever module
* 2 x nRF24L01 power board
* 1 x LM34 Temperature sensor
* 2 x Arduino Power supply

One arduino will have the LCD Keypad shield, temperature sensor, and a wireless transcever. This unit will be used to set the temperature, fan speed, turn the heater/ac unit on and off, and take room temperature readings to transmit back to the base unit.

The other arduino will have the three servos and the other wireless transcever. This unit will receive the data from the first arduino and adjust the respective settings on the heater/ac unit.

Available controls on heater/ac unit:

* fan speed
* on / off
* temperature (high / low)

With this project, we wish to emulate the functionality of an actual thermostat by having three temperature control modes including heat mode, ac mode, and auto mode. In heat mode, the unit will heat the room until it hits the set point then turn the unit off until it drops below the setpoint - 3 degrees, in ac mode it will cool the room until it reaches the setpoint and resume cooling when the room is 3 degrees warmer than the setpoint, and finally in auto mode the unit will automatically determine how to adjust the temperature to keep it within 3 degrees of the setpoint at any given time.

Menu options for arduino:

* current temperature
* set temperature
* fan speed
* power (on/off)
* control mode (heat/ac/auto)

Methods that will need to be written:

TransmitData() – Transmits data from one arduino to the other

ReceiveData() – Receives data from the first arduino and parses it into usable code

Will need a general loop method for both arduinos, as is required by the ide

UpdateSetPoint() – Updates where the ideal temperature should be when user changes it

CheckTemp() – checks temperature from temp sensor to see if adjustments are necessary

ChangeTemp() – writes the temp change (high or low position) to the servo

ChangeFanSpeed() – writes the fan speed (high, medium, low) to the servo

ChangePower() – writes the power position (on/off) to the servo

WriteScreen() – writes data to the screen for the menu on remote unit

The buttons on the lcd keypad will probably be updated from an interrupt method that will do what the button action should make it do

It has been decided that the raspberry pi receiving the weather information will send that information to the base unit arduino when it is put into weather mode and the base unit will adjust the indide weather depending on what the outside weather is like. Additionally, the team has decided to add an additional arduino with an ethernet shield to display current information about the weather and thermostat on a website housed on the arduino.

Weather Tracking Specifications

The team is currently considering multiple possible options to handle the real time weather tracking aspect of the project. The goal is to supply our Arduino-powered temperature controlling environment with real time weather statistics. We plan to program the Arduinos to fluctuate the controlled temperature according to the temperature outside. To satisfy this we are entertaining ideas including a self written program that will pull data from Project Weatherman, open source code that utilizes Project Weatherman, multiple different bash scripts that will pull data from Weather.com and other various weather tracking websites, and Arduino weather station builds.

Utilizing Project Weatherman is currently our top option. Weatherman displays weather information from WeatherBug on the command line and is entirely written in bash. It runs on most UNIX/Linux systems and will provide the team with all the information we need. Work to produce a final product will entail the writing of a program that will grab the supplied information from Weatherman. 1 Arduino device will be hardwired into the computer running said program (Will possibly be run on the Arduino itself), and will wirelessly transmit to the second Arduino device that will be controlling indoor temperature control.

The bash script option is probably the easiest and least time consuming. The script would grab information from weather.com and provide basic weather information for a few hour period of time. This grabbed information would then be parsed for outside temperature information which will then be used by the Arduino to determine the appropriate inside temperature settings.

Also being considered is the development of an Arduino based weather station. For this option, code would need to be written, and the device would need to be built from scratch. There is available open source code that could aid in the development process. In terms of hardware, this option would require of the purchase of the following, in addition to the already owned Arduinos:

1. ESP8266 Microcontroller
2. DHT11 Humidity and Temperature sensor
3. BMP185 Barometer sensor
4. Rain Drop sensor
5. Soil Humidity sensor
6. Breadboard or Custom ProtoShield
7. Jumper Wire