A

Decorative Weather Globe

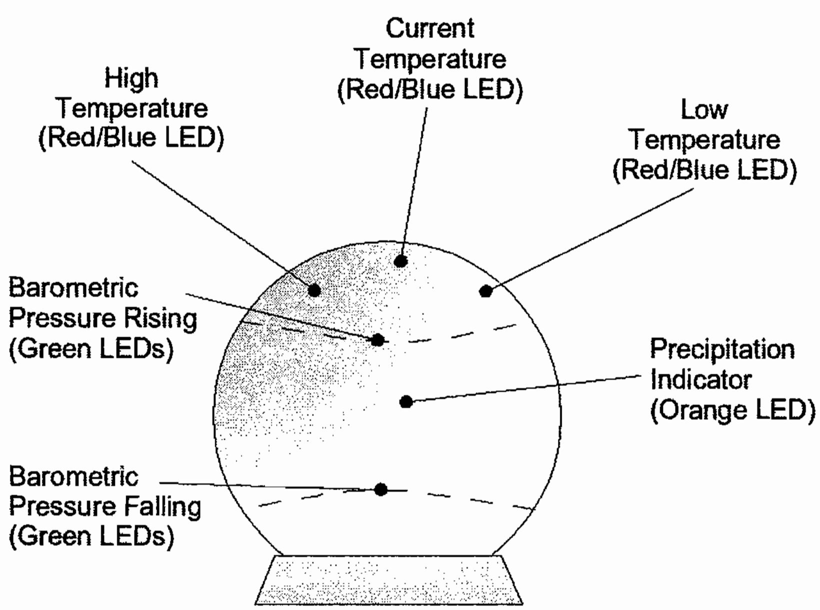
EECS 129A Senior Design

Klefstad, R.

Jay Thomson
Dave Selinger
Jesse Blazina

Abstract:

A Decorative Weather Globe is an advanced weather station that produces an intuitive display representing the current weather. This Globe will be stationary on a coffee table or another easily seen location inside a home. It an excellent informative tool for understanding the gist of the current weather conditions by casually glancing at it from across the room. The DWG can be configured for any geographic area by accessing the DWG system from a PC and fine tuning the weather settings. A user of this product will never forget a jacket on a cold day and will also know if it's raining or not before they go outside.



Description

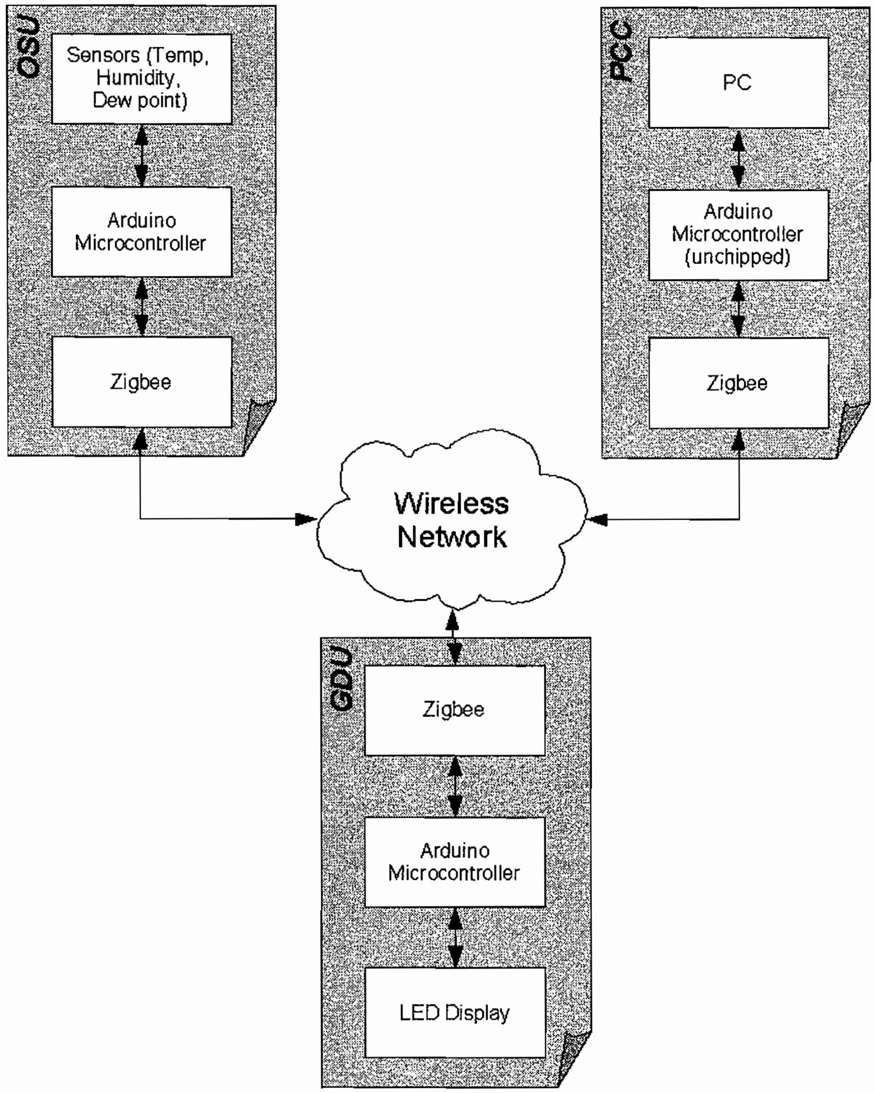
This project is a Decorative Weather Globe, or DWG, which comes to the consumer as an advanced weather station network. The intended use case for this project is a retail product that is purchased at a retail store. After purchase, the customer opens the package at home and plugs in the three devices. Shortly thereafter, the weather sensor data comes streaming in and the globe display will reflect the current data by modulating a triple color LED from red as hot, to blue as cold. If there are specific weather conditions, thunder and lightning or a tornado for instance, then the globe will alert the customer.

One of the hardest challenges is making a display that adapts to the weather in Alaska just as easily as the weather in Texas. If the customer needs or opts to calibrate the globe display, then they can do that from the PC console using the supplied software and a USB port connected to the microcontroller. After customization, the globe display will know what the hot and cold temperature ranges are for the local climate where the customer lives. The customer can also elect to compare their local climate to Internet weather data and follow historical trends over recent sensor data. After the globe display is ready and configured, the customer knows the relative temperature by simply observing the globe display's temperature LED. If there are unusual weather conditions, a snow storm for instance, then the globe display will blink as an alert message. If the temperature exceeds the normal range of temperatures configured by the customer, then the temperature LED will blink while simultaneously displaying the color of the temperature. By using simple heuristics, it is possible to communicate complex weather information while glancing at the globe display; an example scenario for instance, is deciding to get an umbrella or jacket while going out the door to work in the morning. While the DWG can demonstrate all of these features, it will integrate with any decoration scheme better than a text-based LCD display.

There are three parts to the DWG system: the outside sensor (OSU), the globe display (GDU), and the PC console (PCC). Each component of the system communicates wirelessly, through the Zigbee platform, to the other microcontrollers. The OSU will broadcast temperature, humidity, and barometric pressure measurements to the PCC and the GDU. The PCC will interpret the sensor data, log it for later analysis, and will enable dynamic behavioral control of the OSU and GDU for calibration purposes as well as for custom weather settings. The GDU will interact with the raw sensor data from the OSU and will display information about the weather from the relative measurements in a visually appealing and intuitive manner.

The weather station market for sensing an outside temperature and displaying it indoors is saturated with decent quality, low price products. However, all of these products are very simplistic, customization is relatively impossible, and they are an eyesore for living rooms. Furthermore, a roughly equivalent product on the market costs several times more for similar features. For these reasons, our senior design team decided that we could make something much better.

System Level Block Diagram



System Overview

Outdoor Sensor Unit (*OSU*)

- Receivers outdoor temperature, humidity and dew point.
- Transmits Data to other nodes

PC Console (*PCC*)

- Receives Internet weather and broadcasts it over wireless channel
- Logs data from OSU and can do post processing (trends, average etc)
- Configures GDU parameters

Globe Display Unit (*GDU*)

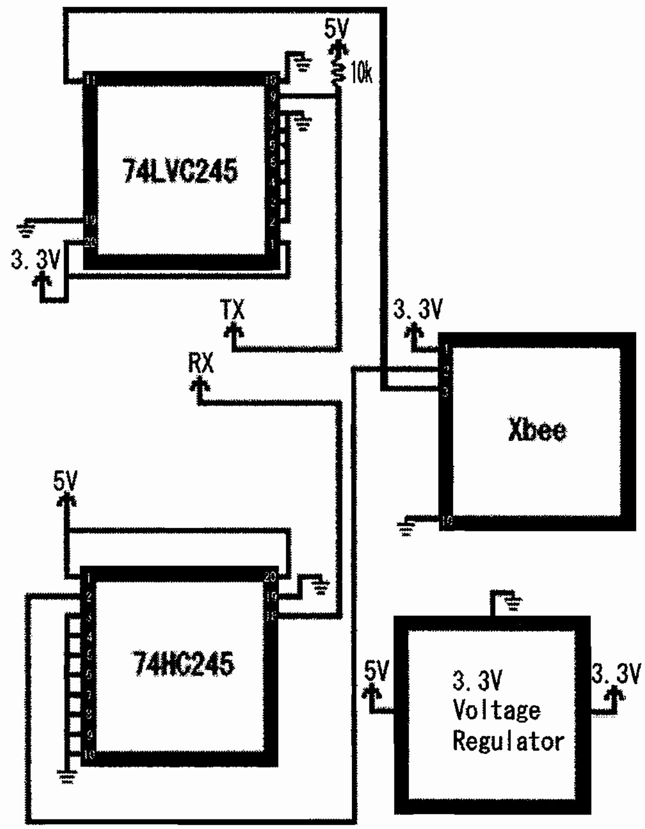
- Receives data from PC and OSU and displays it via a network of LED's
    - Elements to Display
        - Temperature High (Internet)
        - Temperature Low (Internet)
        - Weather Prediction (Rain, Cloudy, Snow, etc)
        - Current Temperature
        - Barometric Pressure Trend

- Receives configuration commands from PC on how to display certain items
    - Temperature LED flashes above/below a specified temperature
    - Weather Prediction gage signals if anything but sunny
    - Signal if there is a sudden drop or increase in barometric pressure
    - Signal if there is a sudden drop or increase in humidity

Circuit Level Documentary

Wireless Module:

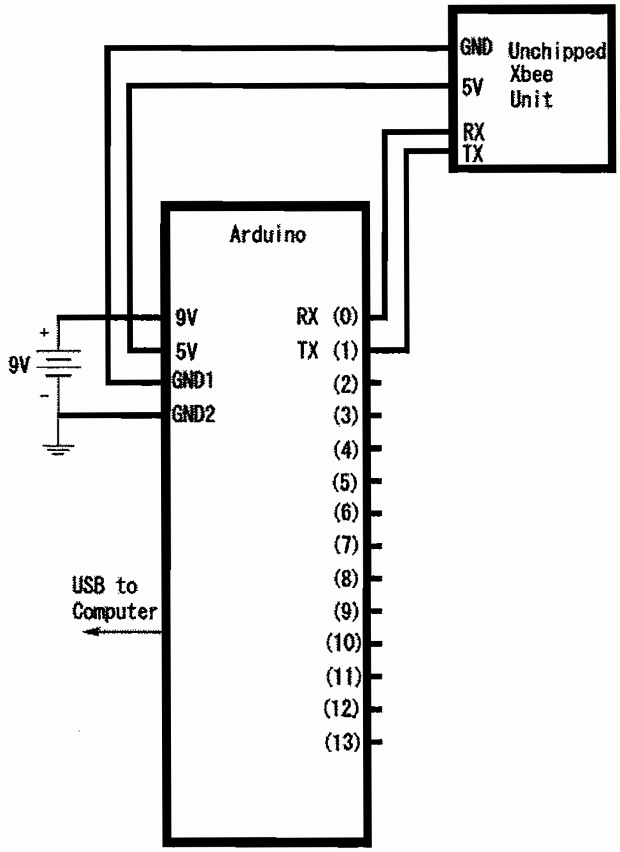
The following is a schematic for the wireless module which will be duplicated of our three modules. It essentially contains the Xbee wireless chip and two buffers, one for transmitting and one for receiving.



Wireless Module Schematic

PC Console:

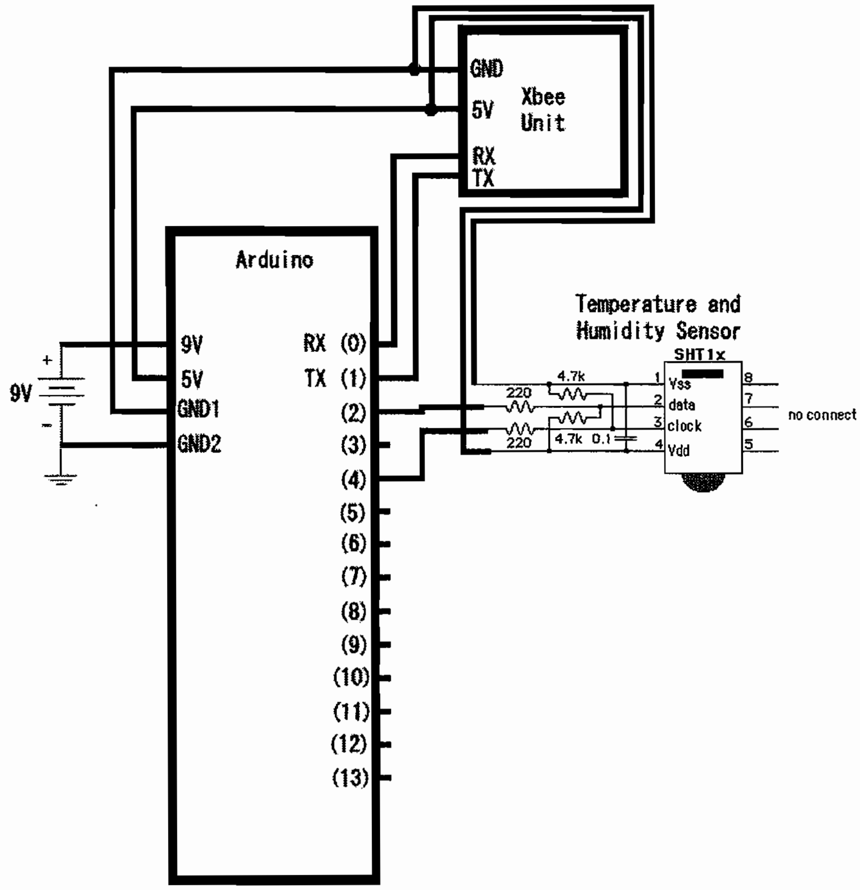
The PC hardware will simply be an Arduino board with a Xbee interfaced with a USB connection. Using an Arduino platform that is unclipped, we achieve a direct serial communication to the Xbee through the USB port.



PC Console Schematic

Outdoor Sensor Unit:

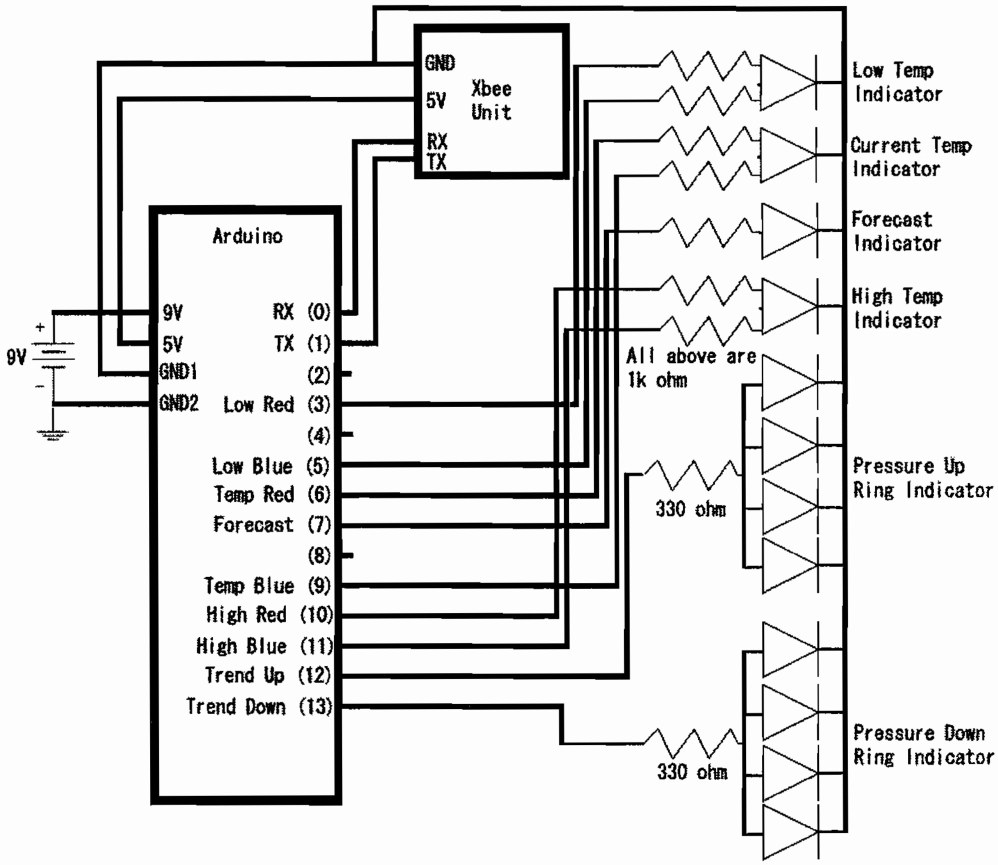
The outdoor sensor will be similar to the PC schematic except that it will additional have an Atmega168 and sensors connected to the Arduino board.



Outdoor Sensor Schematic

Globe Display Unit:

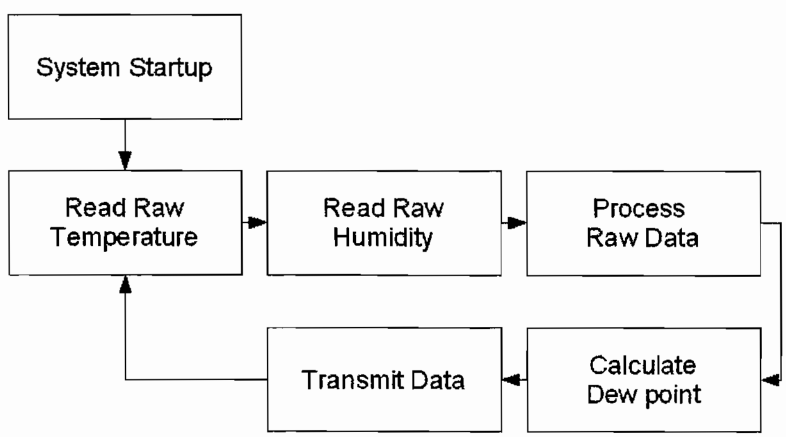
The GDU will additional have numerous LEDs connected to the Arduino.
The following is a unit that will be used in the overall view of the globe. Each of the three temperature indicators will be operated using two PWMs from the Arduino. The remaining LEDs are operated on a digital signal. The following schematic for the globe. There are LEDs setup to indicate the current temperature, the high temperature, the low temperature, change in barometric pressure, and weather.



Globe Display Unit

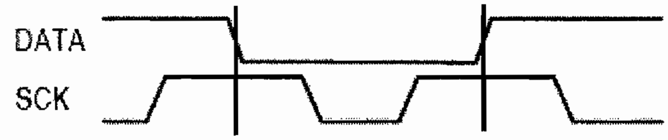
Software Description

Outdoor Sensor Unit:



Outdoor Sensor Unit Software Flow

The outdoor sensor unit essentially reads the temperature and humidity values, calculates the dew point and transmits it all over the wireless network. The sensor combination, temperature and humidity sensor on our outdoor unit is a digital sensor with a two wire interface consisting of a clock and a data line. In order to read data off the unit the following is the sequence to start a transmission:

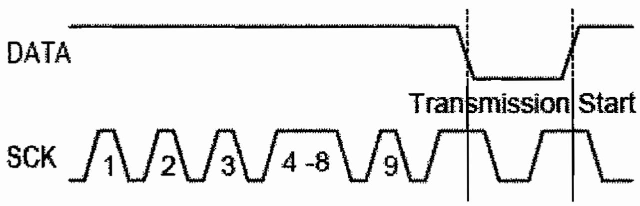


Transmission Start

You then must transmit a code to tell the sensor what you like to do. Below are the following codes which are transmitted by changing the data line relative to what your transmitting and pulsing the clock.

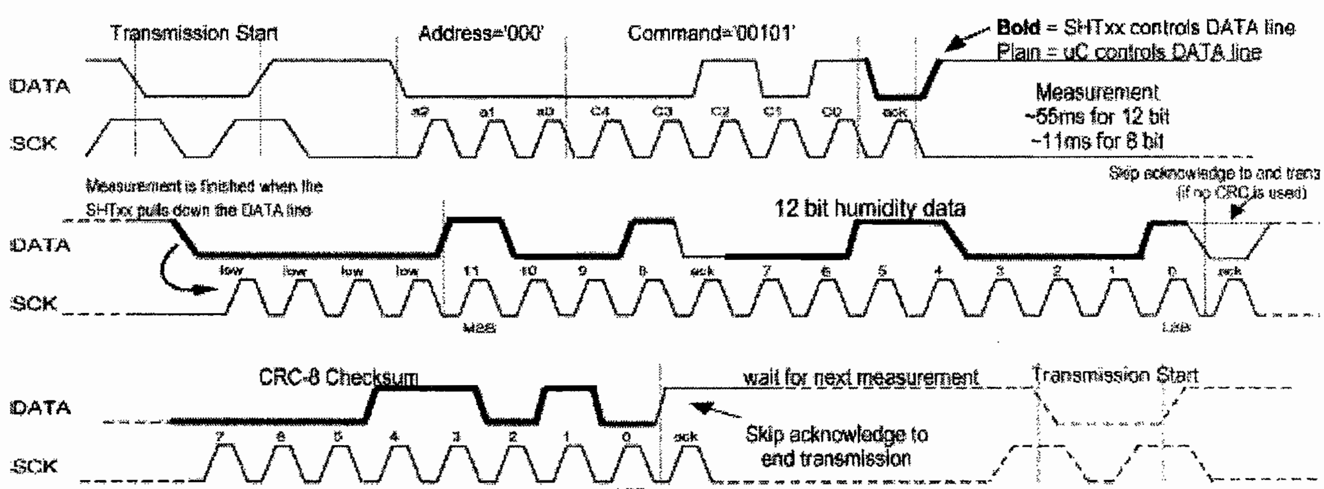
Command	***Code***
Measure Temperature	00011
Measure Humidity	00101
Read Status Register	00111
Write Status Register	00110

After the transmission of the appropriate code, a last ACK pulse clock is sent and the DATA line goes high. The sensor then processes the request. Once it's done the data line goes low and we can read off our value. This is done by pulsing the clock and reading the DATA line. For reading the temperature and humidity, two bytes of data are read followed by two a CRC byte. The status registers are used for reading and setting various parameters on the sensor. More information can be found on the data sheet. In case the there is an error, the sensor can be reset with the following sequence:



Reset Connection

Below is a timing example of reading the humidity off of the sensor:



Example Timing Diagram to read Humidity

Note: Timing diagrams are used from the SHT15 Data sheet

Once the raw humidity and temperature data is read off it must be converted to a readable format. We also calculate the dew point from the temperature and humidity Below are the following equations:

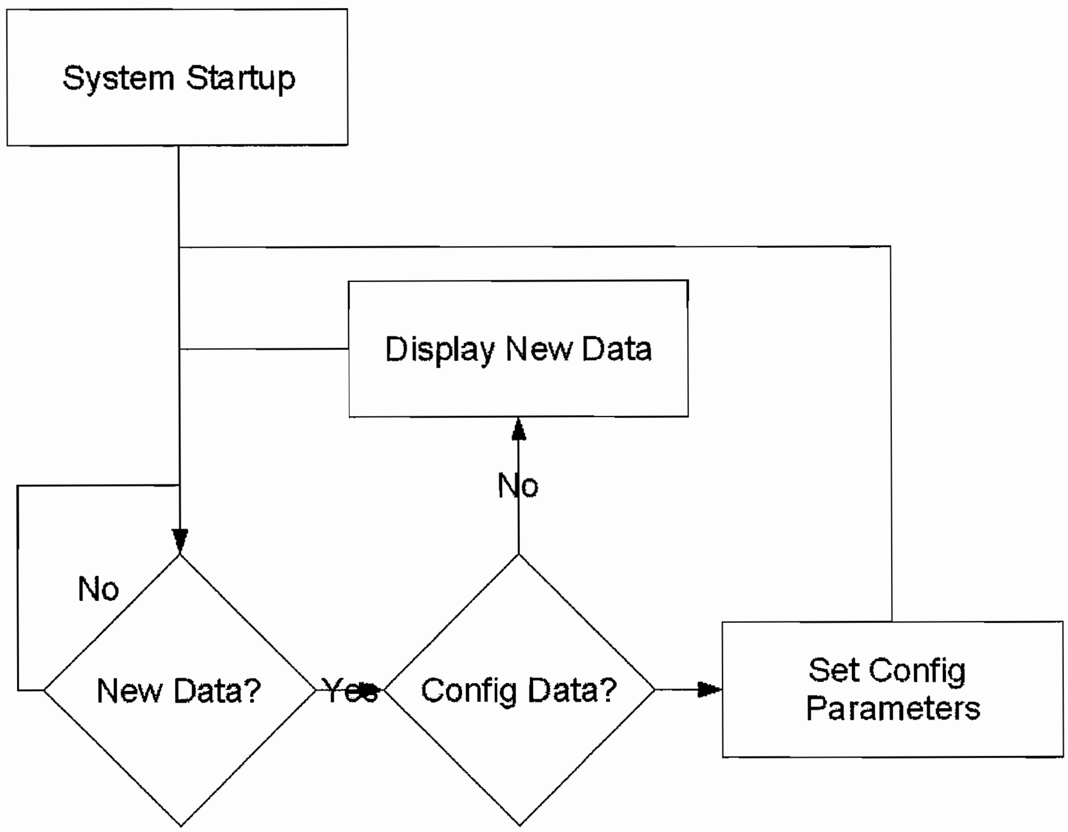
$RH = -4 + 0.0405 * RHD + -2.8 * 10^{-6} * RHD$
Where RH = Relative Humdidty and RHD = Raw Humidity Data

$T = -40.0 + 0.01 * RT$
Where T = Temperature and RTD = Raw Temperature Data

$$H = \frac{(\ln(RH) - 2)}{0.4343} + \frac{(17.62 * T)}{(243.12 + T)}$$
$$DP = \frac{243.12 * H}{17.62 - H}$$
Where RH = Relative Humidity and T = Temperature

Using the serial commands which are described below, we transmit the data over our wireless network were the PC Console logs and records and the Globe unit displays.

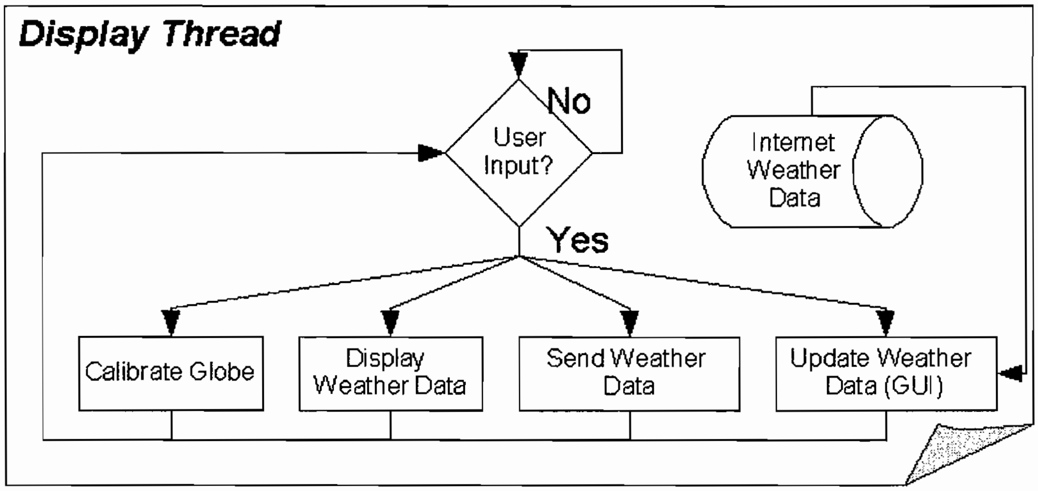
Globe Display Unit:

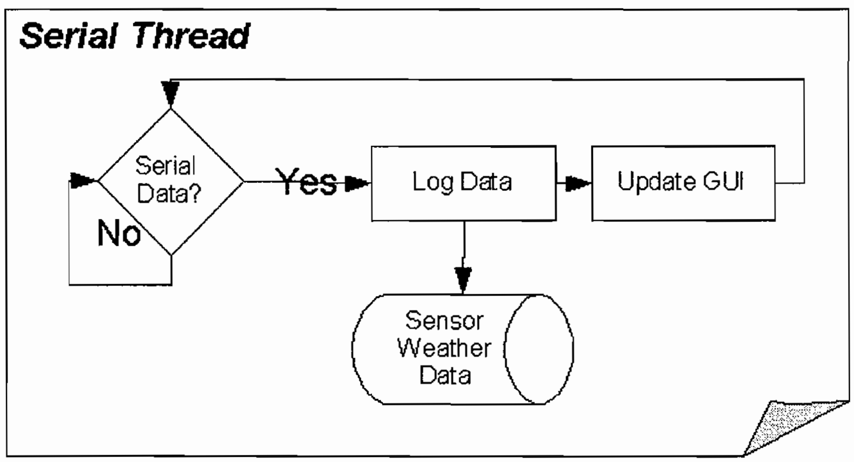


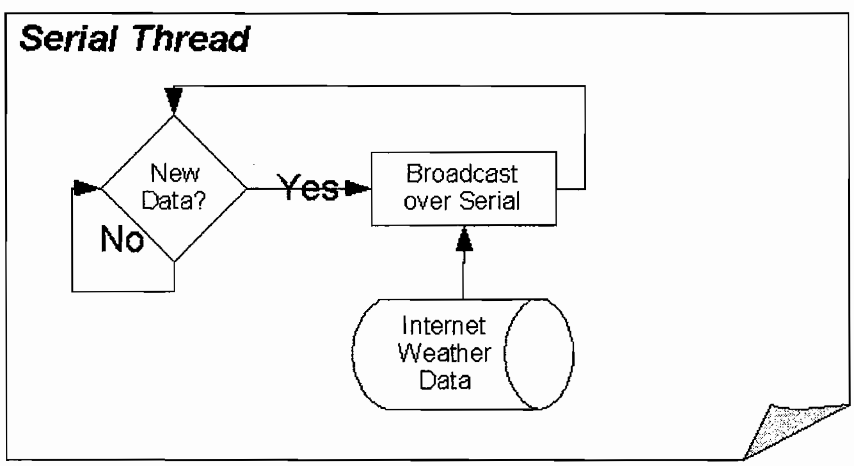
Globe Display Unit Software Flow

The globe unit software reads data thats transmitted over the wireless network. The software essentially polls for new data. It has certain parameters for configuring the globe that can be set via the PC Console. The data is transmitted in human readable data formats. For example the temperature would be sent as 20 degrees. The software translates this based on a configurable high and low value into a range in order to set the intensity of the LED's using the PWM outputs on the Arduino.

PC Console:



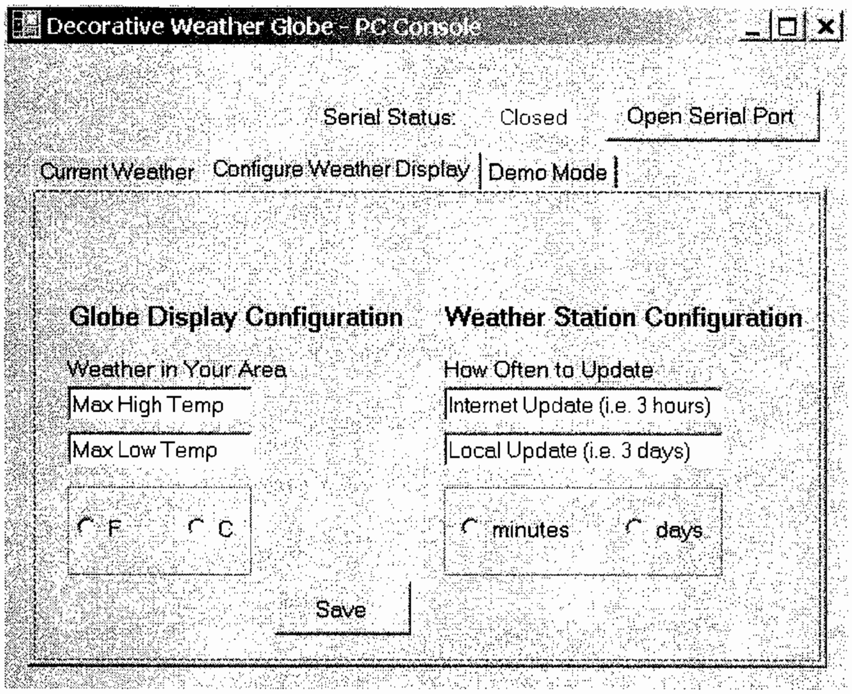




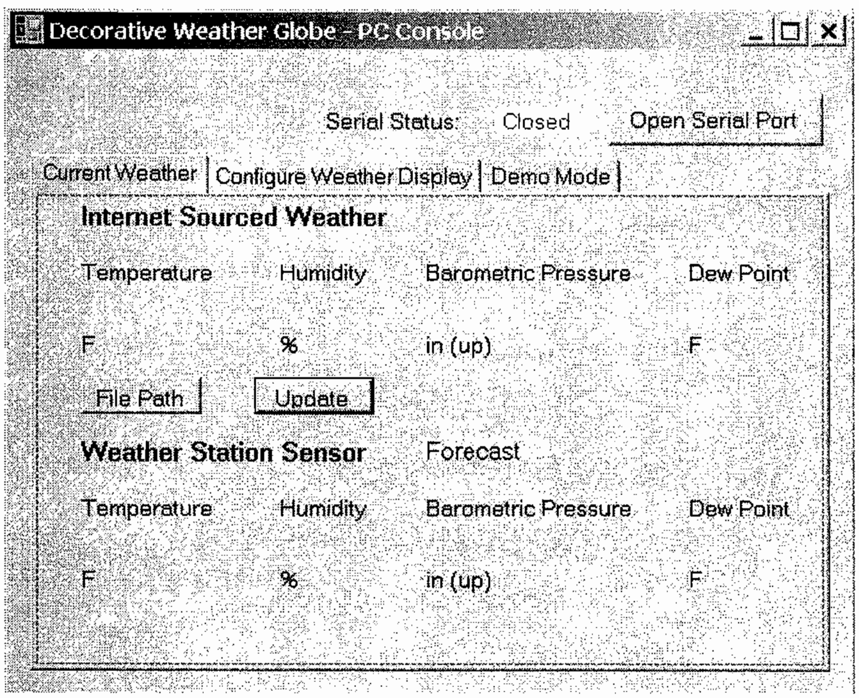
Our PC Client is a custom application written in C#. Essentially it allows a graphical display of weather data from the Internet and our local sensors. We broadcast the weather data achieved from the Internet over our wireless connection to be displayed on our globe unit.

GUI Layout Screenshots:

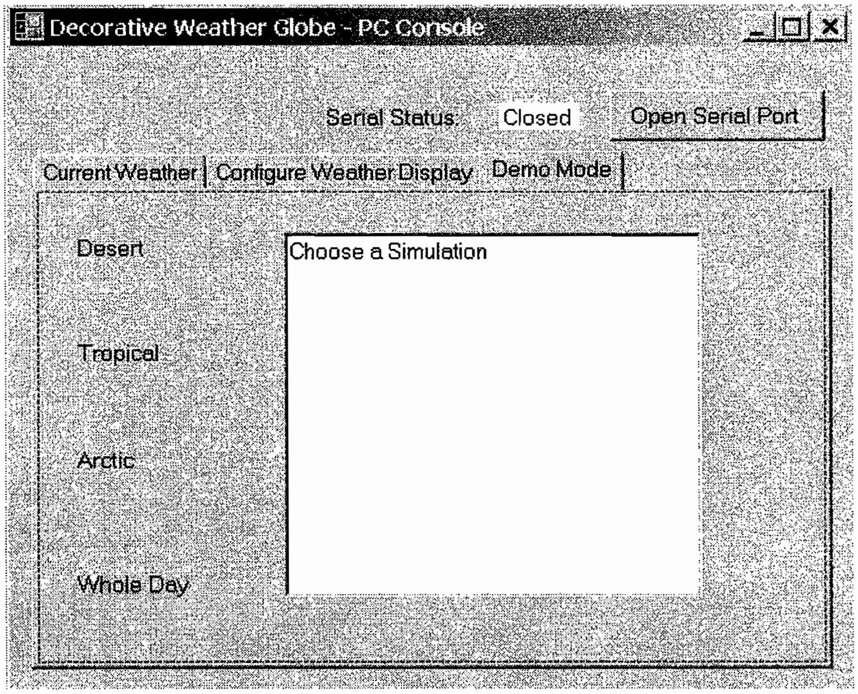
Below is a few example screen shot's from our PC Console client software:



As you can see from above, there's certain configuration settings we can set as far as how often we update and what the max and min's for your local area are generally. This is used in calibrating the LED's on the GDU.



This screen shot shows the local data, both from the Internet and from the sensor unit.



Because weather doesn't change fast enough to provide a reasonable demo, we've added this function so we can demonstrate the globe's abilities to display weather.

Serial Command Language:

Below is the three byte format for sending data and setting commands between each unit.

Byte 1	Byte 2	Byte 3	Description
C	H	Byte Value (C)	Set High Temperature for LED Calibration
C	L	Byte Value (C)	Set Low Temperature for LED Calibration
D	H	Byte Value (C)	Current High Temperature
D	L	Byte Value (C)	Current Low Temperature
D	T	Byte Value (C)	Current Temperature
D	U	Byte Value (%)	Current Humidity
D	D	Byte Value (C)	Current Dew Point
D	R	Byte Value (0,1,2)	High/Low Barometric Pressure Trend (zero, up down)
D	P	Byte Value	Forecast Value (See Values Below)

Forecast Values			
Number	***Description***	***Number***	***Description***
0	CLEAR	12	SMOKE
1	FEW CLOUDS	13	HAIL
2	SCATTERED CLOUDS	14	HAZE
3	BROKEN CLOUDS	15	ICE CRYSTALS

4	OVERCAST	16	SAND
5	DRIZZLE	17	SNOW GRAINS
6	RAIN	18	SNOW
7	FROZEN RAIN	19	SNOW SHOWERS
8	SHOWERS	20	LIGHTNING
9	MIST	21	THUNDERSHOWERS
10	TORNADO		

System Test Plan

Outdoor Sensors Unit

- Ensure proper calibration of temperature sensor
  - Test cold temperatures (e.g. Inside a freezer)
  - Test hot temperatures(e.g. Position sensor over boiling water)
  - Test moderate temperatures (e.g. Touch and release the sensor with fingers)
- Ensure proper calibration of pressure sensor
  - Place pressure sensor indoors and compare sensor with a valid barometer
  - Place pressure sensor outdoors and compare sensor with a valid barometer
- Compare the sensor data with Internet recorded forecast data for accuracy

PC Console

- Ensure that weather data from the Internet is properly loaded into the program
- Ensure each function of the program properly operates

Globe Display Unit

- Ensure that the current temperature indicator blinks whenever it exceeds the current low or current high temperatures
- Ensure that changes to configurations do not cause the LEDs to behave unexpectedly
  - Perform error corrections if necessary after each configuration change
- Test the sturdiness of the position of parts within the globe
  - Shake the globe
  - Roll the globe
- Test the proper functionality of the LEDs using data transmitted from the PC Console

Wireless Network

- Test correct operation with extraneous commands
- Test the range of the of viable communication between units

Cost Analysis

Outdoor Sensor Unit				
Part Number	***Manufacturer***	***Quantity***	***Description***	***Cost***
Arduino-USB	SparkFun Electronics	1	Arduino USB Board	$31.95
ATMega168	Atmel	1	AVR 28 Pin 20MHz 16K 6A/D - ATMega168	$4.11

SHT15	Sensirion	1	Triple Output Humidity and Temperature Sensor - SHT15	$26.95
OD221JE	Digikey	2	220 Ohm Resister	$0.42
SR221E224ZAA	Digikey	1	0.22 Capacitor	$0.35
74HC245N	NXP Semiconductors	1	IC TXRX OCTAL BUS 3TATE 20DIP	$0.58
SN74LVC245AN	Texas Instruments	1	IC OCT BUS XCVR TRI-ST 20-DIP	$0.60
LD1117V33	STMicroelectronics	1	IC LDO REGULATOR +3.3V TO-220	$0.77
XB24-AWI-001-ND	MaxStream Inc.	1	MODULE ZIGBEE 1MW W/WIRE ANT	$19.00
691104	Various	1	10k Ohm Resistor	$0.99
276-1991	Radio Shack	2	20-Pin Retention Contact	$0.69
			Total	$68.71

Globe Display Unit				
Part Number	***Manufacturer***	***Quantity***	***Description***	***Cost Per Unit***
Arduino-USB	SparkFun Electronics	1	Arduino USB Board	$31.95
ATMega168	Atmel	1	AVR 28 Pin 20MHz 16K 6A/D - ATMega168	$4.11
Comp-TPL	SparkFun Electronics	3	Triple Output LED RBG	$1.95
	Various	6	1k Ohm Resistor	$0.99
	Various	4	330 Ohm Resistor	$0.99
74HC245N	NXP Semiconductors	1	IC TXRX OCTAL BUS 3TATE 20DIP	$0.58
SN74LVC245AN	Texas Instruments	1	IC OCT BUS XCVR TRI-ST 20-DIP	$0.60
LD1117V33	STMicroelectronics	1	IC LDO REGULATOR +3.3V TO-220	$0.77
XB24-AWI-001-ND	MaxStream Inc.	1	MODULE ZIGBEE 1MW W/WIRE ANT	$19.00
691104	Various	1	10k Ohm Resistor	

276-1991	Radio Shack	2	20-Pin Retention Contact	$0.69
385-303	Hampton Bay Fan and Lighting	1	Light Fixture	$5.68
52156	Ace Hardware	1	Doll Rod 7/16"	$1.99
30242	Ace Hardware	3 ft	14 AWG Copper Wire	$0.19 - ft
17371	Ace Hardware	1	Household Goop Adhesive	$4.49
33180	Ace Hardware	1	Electrical Tape	$0.79
			Total	$79.36

PC Console				
Part Number	***Manufacturer***	***Quantity***	***Description***	***Cost***
Arduino-USB	SparkFun Electronics	1	Arduino USB Board	$31.95
74HC245N	NXP Semiconductors	1	IC TXRX OCTAL BUS 3TATE 20DIP	$0.58
SN74LVC245AN	Texas Instruments	1	IC OCT BUS XCVR TRI-ST 20-DIP	$0.60
LD1117V33	STMicroelectronics	1	IC LDO REGULATOR +3.3V TO-220	$0.77
XB24-AWI-001-ND	MaxStream Inc.	1	MODULE ZIGBEE 1MW W/WIRE ANT	$19.00
271-1126	Radio Shack	1	10k Ohm Resistor	$.99
276-1991	Radio Shack	2	20-Pin Retention Contact	$0.69
			Total	$35.77

Summary

For the most part, we were able to accomplish many of the goals we planned for this project. We were able to properly construct a single prototype of all three unit types that would be on our planned wireless network (the PC Console, the Sensor Unit, and the Globe Unit).

The implementation of the PC Console is primarily software based. This part of our network functions as we originally planned. It has much potential be expanded including to not only record weather data but to show trends and graphs of historical data. The sensors unit also can easily be expanded to include additional weather sensors. Originally we had planned to include a barometric pressure sensor. A pressure sensor was purchased however it was of the wrong type for measuring atmospheric pressure. Due to lack of funds we decided not to repurchase a new one, as the original couldn't be returned.

The globe unit was the most difficult to implement. Many of our original ideas did not work so well when we actually put everything together. One of the challenges we encountered was that some of the LEDs we used weren't very bright, limiting how well they could be seen through the translucent surface of the globe. A possible solution resolve this matter would simply be to obtain brighter LEDs without draining too much current from the overall electrical network. Another challenge with the globe unit was figuring out a way to mount the LEDs within the globe itself. We currently have no simpler way of mounting the LEDs within the globe in a simpler manner without allowing parts inside the globe to be seen from outside. We have several ideas, including using flexible plastic pieces which can be bent when placing the globe in the unit, however we can out of time to try and implement it.

As for the specifics as for the wireless network, we had originally planned to transmit much more information than we actually implemented. One of the factors for this reduction during the implementation was that there was a limit to what information we could display on the globe unit using only LEDs. Furthermore, instead of sending just three-byte commands between units, we planned to send full strings, including destination and originating information for each message. As we tried to implement the wireless command structure, we felt that it our original ideas were unnecessary and needed to be truncated. Late in the project, we contemplated on switched the commands to an SMS based transmission but felt that out current systems would suffice for the scope of this project. We also had seldom encountered with noise in the wireless systems that could have been circumvented by using capacitors in the electrical network.