Web-based Weather Responsive Window System

By

Fengling Wang

Hanyu Wang

Kaishen Wang

Final Report for ECE 445, Senior Design, Spring 2017

TA: Michael Fatina

2 May 2017

Project No. 84

Abstract

Web-based Weather Responsive Window System enables user to adjust window to different levels remotely via a web server or manually by pressing buttons. The system can quantify current weather of the city that user lives in. Based on the weather situation, it will control the air conditioner to turn on or off and change window's opening level to reach a desired room temperature. To make the system safe and reliable, object sensor is used to avoid people close to the window getting physically injured while it is moving. This report provides details for hardware and software design of the system and verification of main components.

Contents

1. Introduction

Window is an essential furnishing and protection of our home. Sometimes we open window to let fresh air flow into the house or to let the room have a comfortable temperature, and sometimes we close it because of a rainy weather. One problem that can bring inconvenience is that people are not always staying in the house, so they cannot have the control of window all the time. Weather is also occasionally changeable and bad weather can cause detrimental effect far more beyond people's expectation. For example, rain damages properties inside the house if the window is left open; wind blows clothes away if a housekeeper forget to close the window.

To solve this problem, we designed a web-based weather responsive window system which can open window to different levels with remote instructions sent through web or with weather forecasting information from the Internet. The system has two operating modes, auto and manual mode and we define the window to have four different levels with equal sequential distance. Specifically, level 0 corresponds to window being fully closed, level 3 corresponds to window being fully open while level 1 and 2 are intermediate levels. The window system has a few novelties. Firstly, window's opening level can be adjusted either by the web-based system or by user through Internet without presence at home. In addition, compared to weather-responsive windows using various kinds of sensors, the system uses online authentic weather information, which makes control of the window more accurate. Last but not least, the system supports different levels of opening and an LED display to denote whether air conditioner should be turned on.

1.1 Objective

Goals and benefits to potential customers of the weather-responsive system include:

- User can open the window to different levels remotely through a web server in auto mode or through physical buttons in manual mode.
- In auto mode, user inputs the local city's name, a desired level of the window, and a desired room temperature on the website interface. User can either order the window to switch to the entered desired level, or let the system decide which level to go to. To avoid possible damages brought by bad weather, the system will fully close the window.
- If any object remains close to the window and possibly affects window's motion to be stuck, the window will stop operation immediately.
- To achieve energy efficiency, the system will only update the window and the air conditioner's status periodically unless it encounters an extreme weather condition or receives direct instructions from user.

2. Design

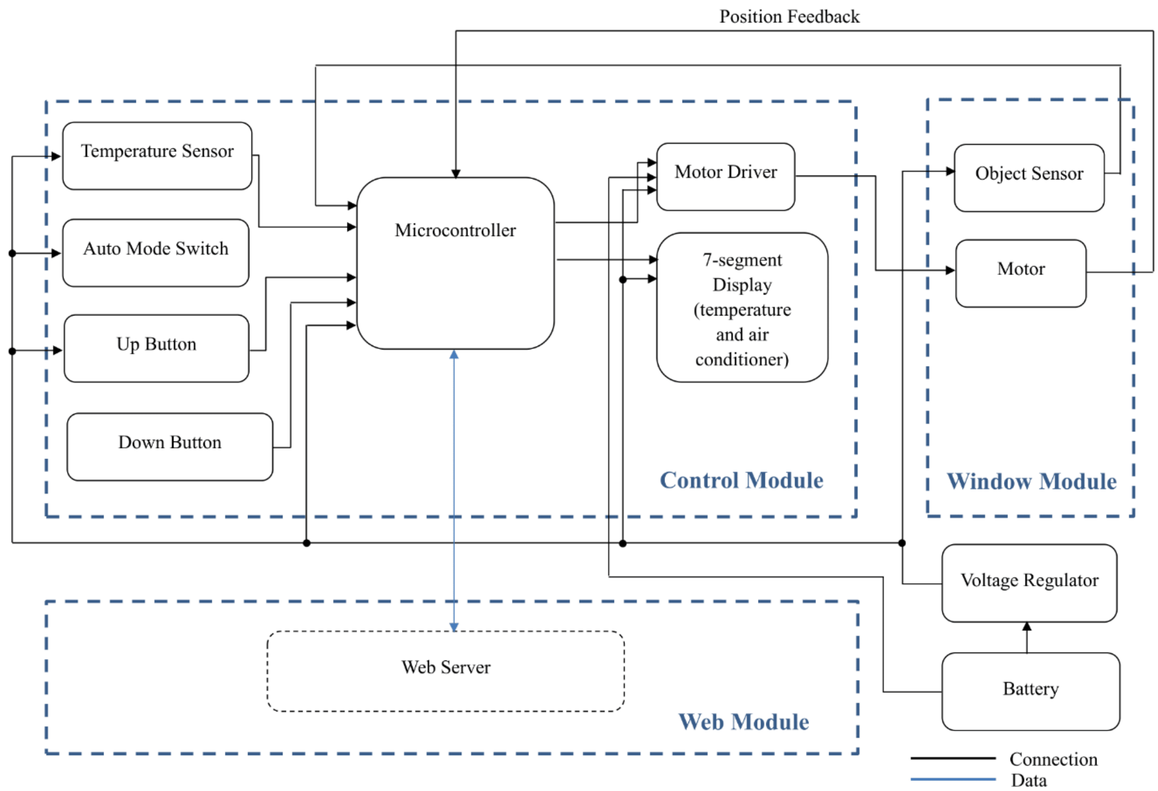

Figure 1 Block diagram of three modules and power supply.

Our project consists of control module, web module, window module and power supply. First of all, the control module has a Wi-Fi incorporated microcontroller which communicates with other chips and the web server. The temperature sensor measures current room temperature and then microcontroller will order the 7-segment display to show it. An LED on the display is also assigned to show the status of the air conditioner, denoted as "AirCon". Secondly, the web module contains a web server with a comprehensive development of front end and back end. The server can grab useful weather information and send it to the control module. Lastly, the window module is composed of an infrared sensor and a gear motor. These two components will work with a screw actuator and an 18*24 inches window in a whole. The motor can receive a signal from control module, then it changes the window to a corresponding level. The IR sensor will detect if any person or object is blocking the window. As for power supply, we use a 6V battery to directly power the motor driver and the battery will also be regulated to power other chips involved in the project.

2.1 Physical Design

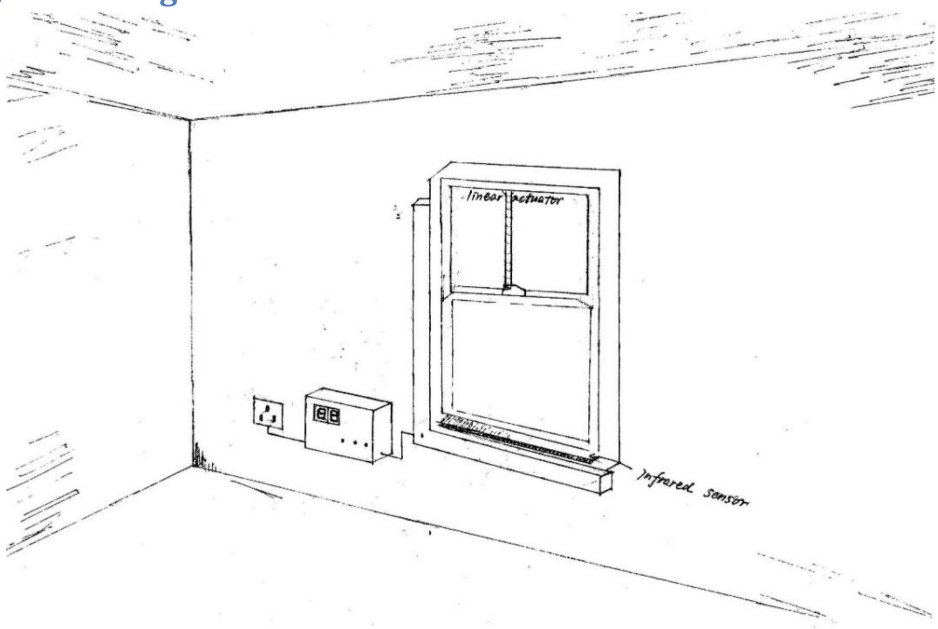

Figure 2 Physical Design of the Window.

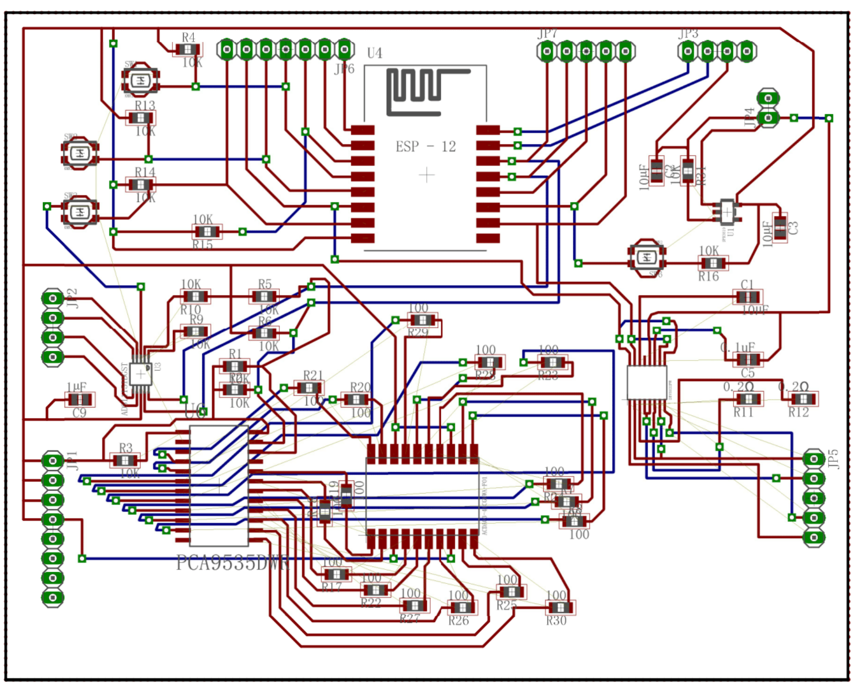

Figure 3 Printed Circuit Board Design for the Entire Project.

2.2 Power Supply

The major power-consuming components on the PCB include microcontroller chip, Analog-to-Digital converter (ADC), I/O expander, 7-segment display and four switches.

ESP8266 chip used as microcontroller can take voltage 3.0 - 3.6V and has operating current 80mA in average; ADC has a wide supply range 2.0 - 5.5V and consumes current of 150μA; I/O expander takes voltage 2.3 - 5.5V and has current consumption of 1μA in maximum; the 7-segment display contains 16 LEDs. It needs forward voltage 2.0 - 2.5V and takes 30mA DC current in maximum.

We used a 6V battery which has 4 AA alkaline batteries connected in series. In all, we supply each component with 3.3V regulated from 6V. Voltage of 3.3V lies in all the supply ranges above except the one of 7-segment display. Hence, we need to add protective current-limiting resistors to take up a portion of 3.3V voltage to operate the display.

Each pair of connecting pins, one in I/O expander and the other one in 7-segment display should have a resistor in series with them to protect the LED from burning caused by a large current. Since we power all the chips at 3.3V and the 7-segment display's forward voltage is about 2.2V [1], then

$$V_{resistor} = 3.3V - 2.2V = 1.1V$$

Let LED take 10mA current, so the proper resistance we would choose is,

$$Resistance = 1.1V / 10mA = 110 \text{ Ohms}$$

The total power consumed by components on the printed circuit board will be,

$$P_{on\text{-}board\ components} = 3.3V * (80mA + 150\mu A + 1\mu A + 30mA)$$

$$= 0.36W$$

The gear motor we chose is intended for operation at 12V with stall current 5A. According to the datasheet [1], the motor can also run at 6V so we use a 6V battery. Since the motor driver can provide maximum motor power voltage of 10.8V, the motor can have

$$I_{maximum} = 5A * \frac{10.8V}{12V} = 4.5A$$

In addition, DRV8833 dual H-bridge can provide 1.5A output current per bridge and 3A in total if parallel connecting the two bridges, the motor would receive 3A. Although it doesn't reach the 4.5A current for greatest efficiency, it is sufficient enough to drive the motor. Therefore, the power consumed by motor will be,

$$P_{motor} = 6V * 3A = 18W$$

2.3 Control Module

Control module will have the following components. There are three functioning switches and one switch left empty. The AUTO switch is used to switch between auto mode and manual mode; the UP and DOWN buttons are used to move the window to an expected level when system being in manual mode; the temperature sensor will measure current environment temperature and microcontroller will show it on the 7-segment display.

2.3.1 Microcontroller

Microcontroller receives information from the web server constantly and decides to what level should the window be moved to and whether to turn on the air conditioner. With relative information prepared, the microcontroller will then communicate with the 7-segment display and the H-bridge motor driver. To guarantee safety, when the IR sensor detects anything getting in the way, microcontroller will stop sending power signals to disable the motor. A more detailed implementation logic can be found in Figure 5.

We chose ESP8266 12E Wi-Fi chip to be the microcontroller. Since it has a NodeMCU development board integrated with a USB port, it is convenient to load program into the chip for testing.

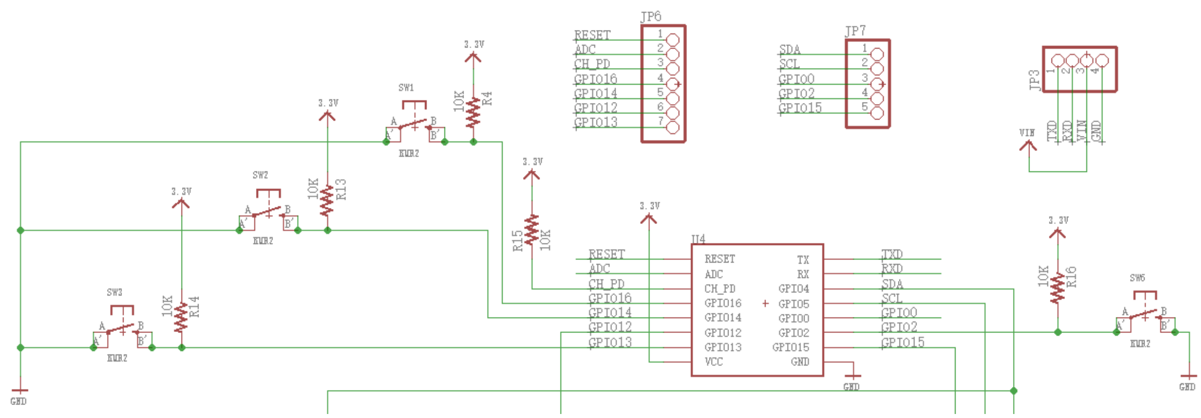

Figure 4 Schematic for Microcontroller's Pin Connections.

As shown in the table below, we assigned each pin with different functions according to ESP8266's datasheet [2]. V_{cc} is connected to 3.3V and GND is connected to PCB itself, which we set to be GND. Other unmentioned pins are left empty.

Table 1 Microcontroller Pin Assignments

Pin Name	Type	Function
GPIO16	I/O	Up Button
GPIO14	I/O	Down Button
GPIO12	PWM0	A_IN1 and B_IN1 of Motor Driver
GPIO13	I/O	Button (Not used)
(Continued)		

GPIO4	I/O	SDA
GPIO5	I/O	SCL
GPIO2	I/O	Auto Mode Switch
GPIO15	PWM1	A_IN2 and B_IN2 of Motor Driver

Considering possibility that we may need to access the signals from microcontroller later, we set up three pin headers for all the pins on the chip except V_{cc} and GND. One for pins on the left-hand side, one for pins on the right-hand side and another one for USB-to-TTL cable connection.

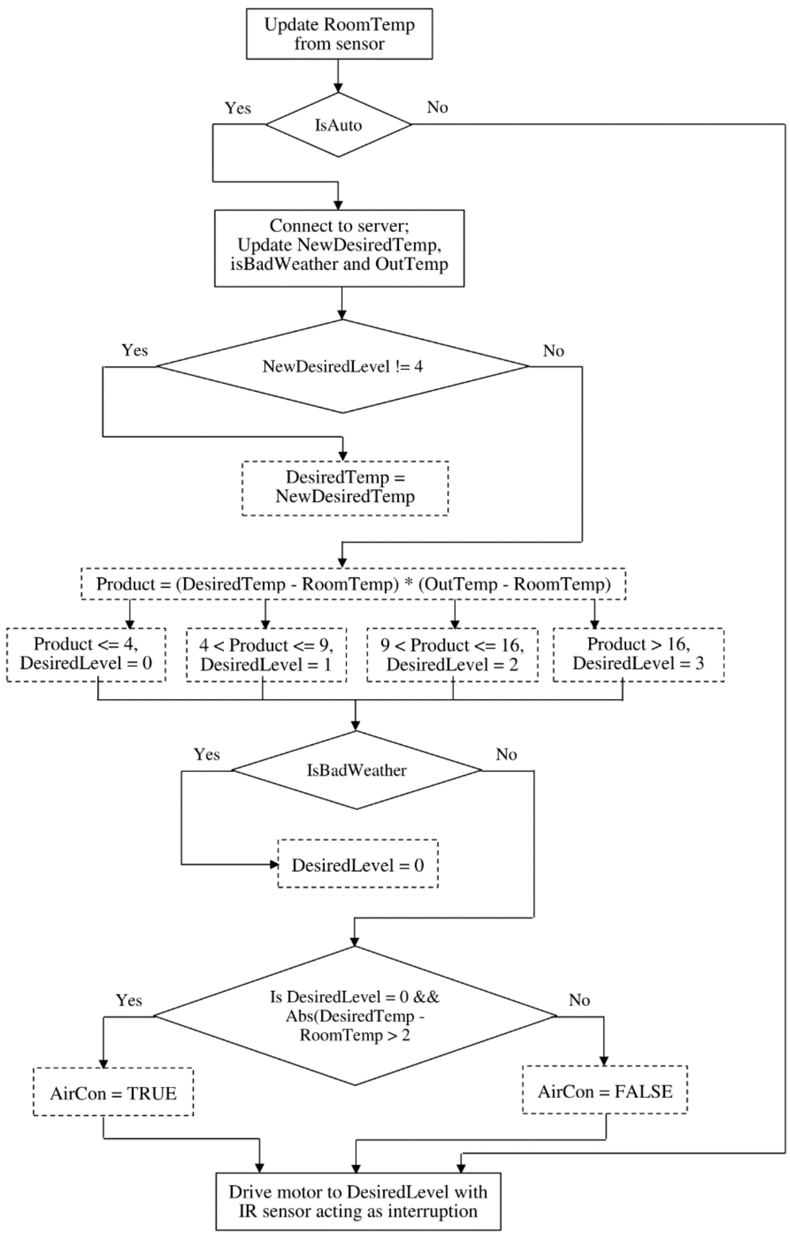

Figure 5 Microcontroller Logic Flow Chart.

2.3.2 Switches

In our design, the window can operate in two modes, auto mode and manual mode. We use one button to switch between the two modes. To handle with manual mode, we assigned anther two buttons, UP and DOWN which allow user to manually open or close the window for one level, 2cm in distance. Particularly, these two buttons will be ignored so won't have effect on the system in auto mode.

Each of the three buttons is assigned a GPIO pin on microcontroller to send signal to it. In our program, we use a function called "digitalRead" to keep track of the buttons' status: the reading will become 1 if button is pressed and 0 if released.

2.3.3 Temperature Senor

To provide an intuitive denotation of room's environment, we place a temperature sensor LM35DZ/NOPB beside the window to measure current room temperature and show it on the 7-segment display.

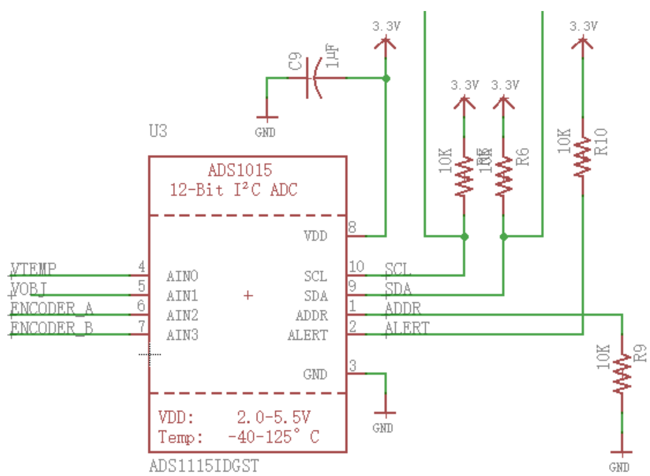

Figure 6 Schematic for ADC chip's Pin Connections.

Since the sensor outputs analog signal, it can't directly connect with microcontroller chip but goes to a 16-bit Analog-to-Digital Converter first. The ADC chip will convert the sensor's analog signal to reasonable digital value, and then send it through SCL and SDA to microcontroller for processing.

Based on the equation used for ADC in 5V system [3],

$$\frac{5}{1023} = 0.004882814 = \frac{Analog\ Voltage\ Measured}{ADC\ Reading}$$

we deduced a function which can give accurate temperature in degree of Celsius for our board driven by 3.3V,

$$Temperature = (ADC\ Reading * 0.004882814 - 3.9) * 10;$$

2.3.4 Seven-segment Display

The two-digit 7-segment display ACDA03-41CGKWA-F01 is used to show current room temperature in degree of Celsius. To program it, it needs to be connected with a 16-bit I2C I/O expander to allow the microcontroller to control the 16 LEDs in the display. In on-board circuitry, we assigned the left eight pins of I/O expander to be corresponding with the left eight pins of the 7-segment display and the same thing for the pins on right-hand side.

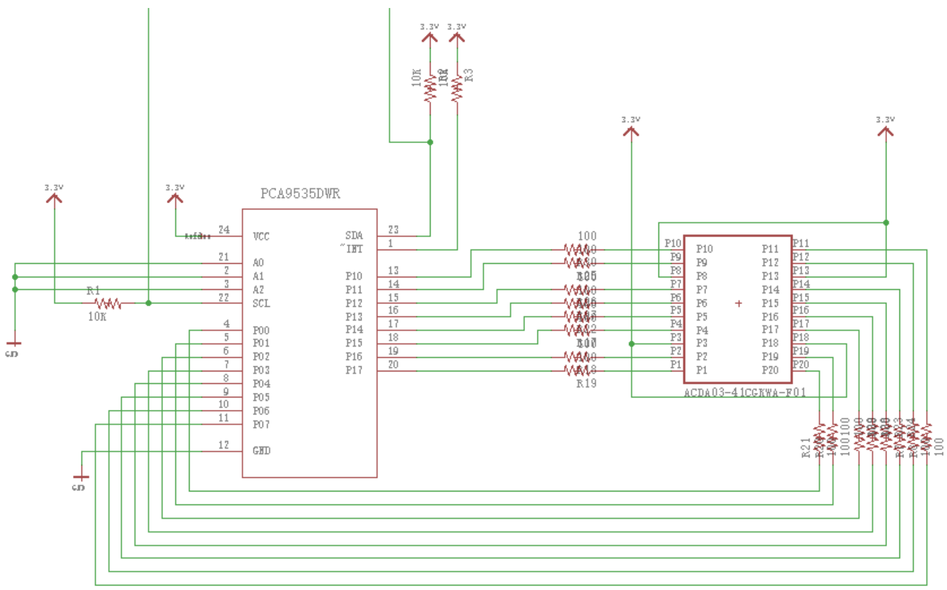

Figure 7 Schematic for I/O Expander and LEDs Connections.

In our program, we established two 10 * 2 two-dimensional arrays to store the mappings between the 7-segment display' pins and LEDs that represent higher and lower digit respectively.

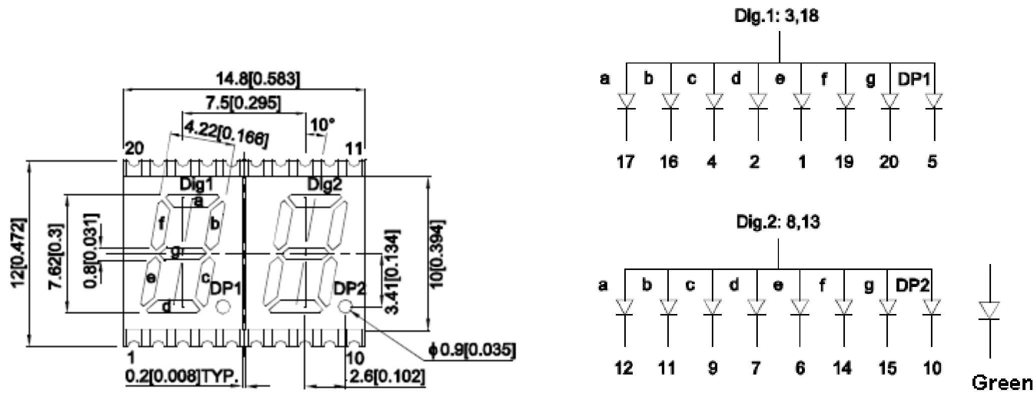

Figure 8 Seven-segment Display's Digit-to-LED Mappings.

We made the top half of the digits a, b, and f to be represented by Mapping[x][0] and the bottom half c, d, e, and g to be represented by Mapping[x][1]. We figured out the denotation by calculating the tens digit and units digit and then fetching the bytes we need from an initialized array, for example, the array for lower digit's LEDs as follows

Table 2 Lower Digit's LEDs Mappings

Units Digit x	Mapping[x][0]	Mapping[x][1]
0	B11100000	B00001110
1	B10000000	B00000010
2	B11010000	B00001100
3	B11010000	B00000110
4	B10110000	B00000010
5	B01110000	B00000110
6	B01110000	B00001110
7	B11000000	B00000010
8	B11110000	B00001110
9	B11110000	B00000110

2.3.5 Motor Driver

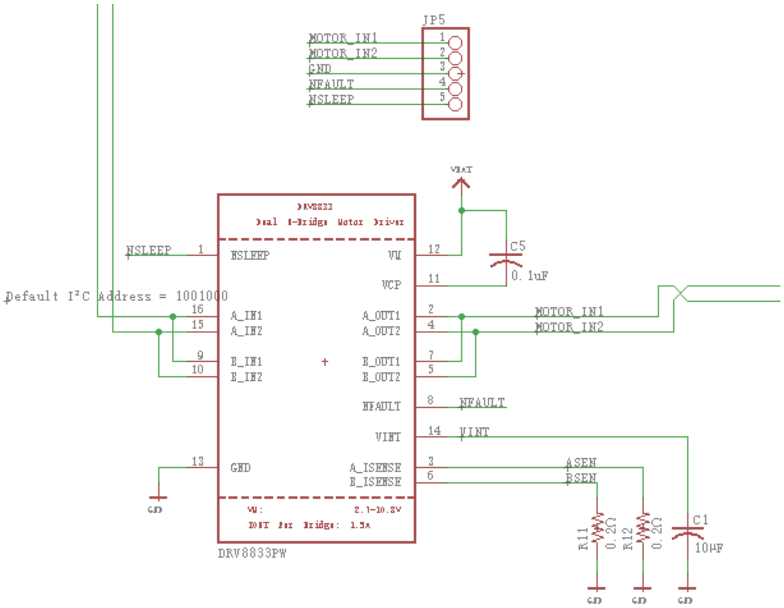

Figure 9 Schematic for Motor Driver's Pin Connections.

The motor driver we use is DRV8833 Dual H-Bridge Motor Driver. It is powered by a 6V battery directly. To control the direction and speed of the motor, we utilize Pulse width modulation, i.e. PWM. By swapping the input pin that receives PWM, the direction that motor is moving in can be reversed. As shown in Table 3, we use Forward or Reverse PWM, fast decay. For implementation, we perform function analogWrite(1024) on xIN1 to send a PWM signal which can drive the motor at the greatest speed, and set xIN2 to 0. If to make the motor to move in the opposite direction, we perform reverse things on xIN1 and xIN2.

A_IN1 and B_IN1 take in PWM from pin GPIO 12 while A_IN2 and B_IN2 take PWM from pin GPIO 15 on the microcontroller. We connect A_OUT1 and B_OUT1 in parallel to provide a greater power to the red power input wire of the motor. A_OUT2 in parallel with B_OUT2 will be connected to the other power input wire of motor.

Table 3 PWM Control of Motor Speed [4]

xIN1	xIN2	Function
PWM	0	Forward PWM, fast decay
1	PWM	Forward PWM, slow decay
0	PWM	Reverse PWM, fast decay
PWM	1	Reverse PWM, slow decay

2.4 Window Module

2.4.1 Object Sensor

For safety control, we installed an infrared break-beam sensor on the bottom left frame of the window. It consists of an emitter side that sends out a beam of human-invisible IR light, and a receiver sensitive to the same light. Compared to PIR sensor, break-beam sensor is faster and allows better control to detect any motion. If any person or thing break in, it will block the infrared light transmitted from emitter to receiver, and microcontroller can be informed of that. Same as the temperature sensor, the object sensor also produces analog signal which need to be converted in ADC chip; then a digital version of reading will be sent to microcontroller through SCL and SDA.

On the 7-segment display, the decimal point following the first digit will indicate if there is any object getting close to the window's frame and block the infrared beam.

2.4.2 Motor

Our actuator is made of a thick screw which is about 30.5cm long in length. One side of it is fixed at the top center of the window and the other side is fixed at the top of bottom half window. Considering great friction and gravity applied on the window when it is moving up, we chose an effective motor, 100:1 metal gear motor 37D*73L mm with a 64 CPR encoder. This gear motor is a powerful 12V DC motor with a 102.083:1 metal gearbox and is integrated with an encoder that provides a resolution of 64 counts per revolution of the motor shaft [5].

2.5 Web Module

The web server requires such information from user: city where he lives in, desired window level and desired temperature. The server uses Yahoo weather API to determine the weather condition and the city's average temperature. Based on the condition, it will decide whether the city is undergoing a bad weather like raining or snowing and save such four parameters: desired window level, desired room temperature, current temperature and a Boolean value indicating if bad weather is happening now.

The server then handles requests sent by the microcontroller and replies with the saved information. The front end can show the connectivity between server and microcontroller.

The web server is implemented on C-Panel and it can be visited at http://ece445group84.web.engr.illinois.edu.

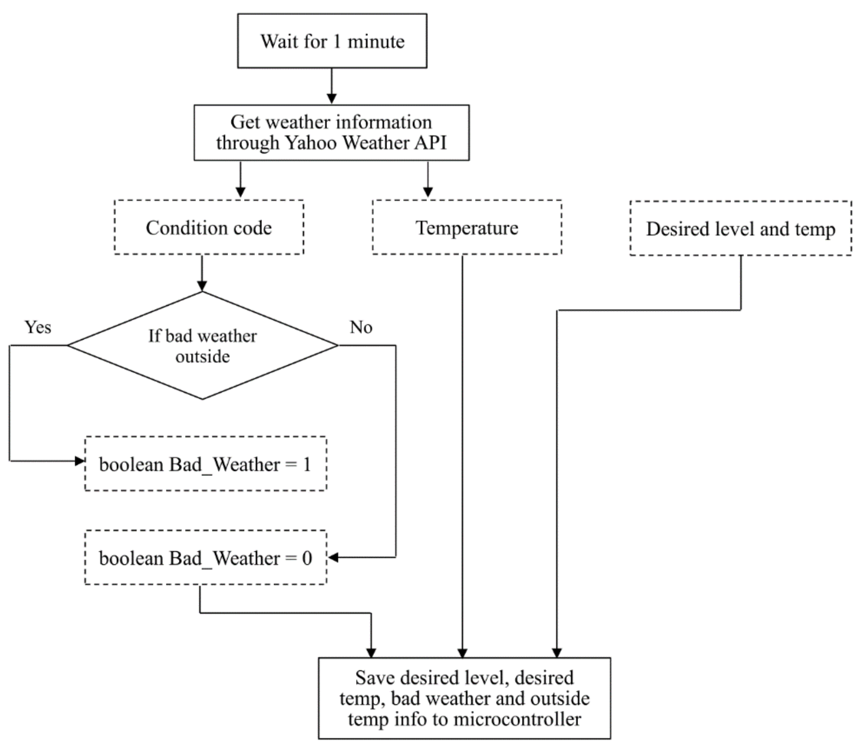

Figure 10 Web Server Logic Flow Chart.

3. Design Verification

3.1 Microcontroller

We tested a few possible combinations of city and desired temperature for our level determination algorithm. Four representative cases are shown below,

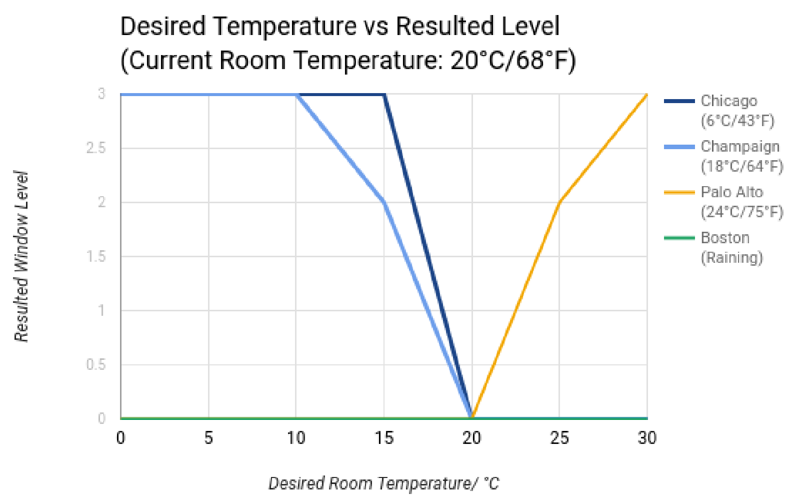

Figure 11 System Responses to Four Cities' Weather and Desired Temperature.

3.2 Switches

We tested the three switches with a standard switch sample code provided online [6]. The Boolean flag in the program will be set to its previous value's opposite once the button is pressed for a time longer than the debouncing time. For example, if the Boolean variable is 0 at initial, then it would be set to 1 at the time when button is pressed.

3.3 Motor Driver

We tested the motor driver with a standard motor sample code provided online [7]. The motor is driven in a speed positively correlated to the analog value written to the motor pin. To enable the motor to run in two opposite directions, we wrote a full speed of value 1024 to motor's pin A_IN1 and 0 to motor's pin A_IN2, and the motor operated in one direction. Later we wrote a full speed of value 1024 to pin A_IN2 and 0 to pin A_IN1, and the motor operated in the other direction.

3.4 Object Sensor

We used Arduino code to constantly read the input analog value from object sensor. The input value is positive and stably remains around 11450 when there is no object in between the IR emitter and IR receiver, and negative and stably remains around -154 when there is an object in between.

3.5 I/O Expander

We wrote two two-dimensional arrays of 8 bit masks in the program indicating which LED segment should be on for lower or higher digit. With bit operation, we could use the I/O expander to drive the 7-segment LED display and show any two-digit value we want. We verified correctness of the mapping by displaying values from 0 to 99 every half second.

4. Costs

4.1 Parts

Table 4 Parts Costs

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
Temperature Sensor LM35DZ/NOPB	Texas Instruments	1.86	1.86	1.86
Green Two-digit 7-Segment Display ACDA03-41CGKWA-F01	Kingbrihgt	4.16	4.16	4.16
16 Bit Analog to Digital Converter ADS1115IDGST	Texas Instruments	0.65	6.51	0.65
Linear Voltage Regulator 8-SOIC SPX3819S-L/TR	Exar Corporation	0.89	4.45	0.89
Tactile Switch SPST-NO KMR211GLFS	C&K	0.49	2.94	1.96
10k Ohm Chip Resistor ERJ-6GEYJ103V	Panasonic Electronic Components	0.02	0.45	0.26
100 Ohm Chip Resistor ERJ-6ENF1000V	Panasonic Electronic Components	0.03	0.66	0.48
0.2 Ohm Chip Resistor ERJ-6BSFR20V	Panasonic Electronic Components	0.48	2.40	0.96
0.1µF Ceramic Capacitor 0805 08055C104KAT2A	AVX Corporation	0.03	0.33	0.03
1µF Ceramic Capacitor 0805 CC0805KKX7R7BB105	Yageo	0.07	0.73	0.07
10µF Ceramic Capacitor 0805 0805ZD106KAT2A	AVX Corporation	0.11	1.09	0.33
Linear Voltage Regulator 3.3V SPX3819M5-L-3-3/TR	Exar Corporation	0.72	2.16	0.72
USB to TTL Serial Cable 954	Adafruit	9.95	9.95	9.95
ESP8266 ESP-12E Serial Wi-Fi Wireless Transceiver Module	AI-Thinker	9.87	9.87	9.87
IR Break Beam Sensor - 3mm LEDs	Adafruit	1.95	1.95	1.95
Printed Circuit Board	PCBWay	10.40	52.00	10.40
100:1 Metal Gearmotor 37Dx73L mm with 64 CPR Encoder	Pololu	39.95	39.95	39.95
Total				**84.49**

4.2 Labor

Table 5 Labor Costs

Name	Hourly Rate/$	Total Hours Invested/hr	Total/$
Fengling Wang	30	220	6600
Hanyu Wang	30	220	6600
Kaishen Wang	30	220	6600
Total	**90**	**660**	**19800**

5. Conclusion

5.1 Accomplishments

The whole system is reliable and complicated. From practical point of view, it realizes multiple functionalities which could significantly facilitate people's daily life. From technical point of view, the system uses technology from a lot of areas, such as PCB design, front end and backend development, programming, signal processing, and integration of hardware and software.

5.2 Uncertainties

The object sensor needs to be integrated into the whole program to allow the microcontroller to interrupt the running status of the motor when any object is found near the window.

To avoid the window moving back and forth, the boundary condition when desired temperature is close to current room temperature needs to be explicitly handled.

5.3 Ethical considerations

We use weather information from free Yahoo API. We should obey the term described at a Yahoo's webpage for developer [8]. We should pay respect to professional creations of other people. Per Imperative 1.6 in the ACM Code of Ethics, "Specifically, one must not take credit for other's ideas or work, even in cases where the work has not been explicitly protected by copyright, patent, etc." [9].

Another concern about our project is that the strong power supplied to the actuator may leak and hurt people or small animals. Number 9 of the IEEE Code of Ethics has a requirement like, "avoiding injuring others, their property, and reputation" [10]. With help of the infrared sensor, our project becomes safer to user.

5.4 Future work

To make the project to become a commercial product, we may want to add functionalities in two more aspects. The first one is that user may want to change the desired temperature by a physical button. The second one is that we need to enhance power supply for the system to be applicated on more kinds of window.

References

[1] “Surface Mount Display,” online, digikey.com, accessed May 1 2017. [Online]. Available: http://www.kingbrightusa.com/images/catalog/SPEC/ACDA03-41CGKWA-F01.pdf

[2] “ESP8266ex Datasheet,” online, arduino.org, accessed May 3 2017. [Online]. Available: http://download.arduino.org/products/UNOWIFI/0A-ESP8266-Datasheet-EN-v4.3.pdf

[3] “Analog to Digital Conversion,” online, sparkfun.com, accessed Mar 26 2017. [Online]. Available: https://learn.sparkfun.com/tutorials/analog-to-digital-conversion

[4] “DRV8833 dual h-bridge motor driver,” online, ti.com, accessed Feb 25 2017. [Online]. Available: http://www.ti.com/lit/ds/symlink/drv8833.pdf

[5] “100:1 Metal Gearmotor 37Dx73L mm with 64 CPR Encoder,” online, pololu.com, accessed May 1 2017. [Online]. Available: https://www.pololu.com/product/2826

[6] “Switch,” online, sparkfun.com, accessed Apr 30 2017. [Online]. Available: http://www.arduino.cc/en/tutotial/switch

[7] “Arduino Lesson 13. DC Motor,” online, sparkfun.com, accessed Apr 30 2017. [Online]. Available: learn.adafruit.com/adafruit-arduino-lesson-13-dc-motors/arduino-code

[8] “Yahoo APIs Terms of Use,” online, yahoo.com, accessed May 3 2017. [Online]. Available: https://policies.yahoo.com/us/en/yahoo/terms/product-atos/apiforydn/index.htm

[9] “ACM code of ethics and professional conduct,” online, acm.org, accessed Feb 17 2017. [Online]. Available: https://www.acm.org/about-acm/acm-code-of-ethics-and-professional-conduct

[10] “IEEE Code of Ethics,” online, ieee.org, accessed Feb 17 2017. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html

Appendix A Requirement and Verification Table

1. Control Module

1.1 Microcontroller

Requirement	Verification	Points
1. When in non-auto mode, the microcontroller should be able to adjust the window to different levels using UP and DOWN button. The response time should be less than 2 second. 2. When in auto mode, and the user enter a city that is not in bad weather condition, the microcontroller can get the desired level and adjust the window level to the desired one. The response time should be less than 30 seconds. 3. In auto mode, when user enters a desired level of 4 (which means don't care), the microcontroller should open the window to the level which in long term can drive the room temperature towards the desired one. Meanwhile, we should see that the right dot on seven segment display denoting the Air Conditioner will be on if and only if the window is closed and the room temperature doesn't equal to the desired one. The response time should be less than 30 seconds. 4. If the outside weather condition is bad (rainstorm, tornado, etc.), the window will be closed.	1. Use button to adjust window level from 0 to 3, and back from 3 to 0. Use timer to measure the response time. 2. Find a city not in bad weather condition. And enter the city and desired level of 1 to the webpage, see that the window level will be adjusted to 1. And enter desired level of 0 to the webpage, see that the window level will be adjusted to 0. Use timer to measure the response time. 3. Enter desired level of 4 to the webpage. Enter the desired temperature equals to the outside temperature, see that the microcontroller will adjust the window to level 3. Change the desired temperature to somewhere between the inside room temperature and the outside temperature. See that the microcontroller will adjust the window to level 1 or 2. Change the desired temperature to room temperature. See that the microcontroller will adjust the window to level 0. Check if the right dot on seven segment display is on in appropriate situations. Use timer to measure the response time. 4. Enter a city that is in bad weather condition, and set the desired level to non-zero, see that the window will not be opened. If we set the desired temperature equal to the outside temperature, the window still cannot be opened.	15

1.2 Other Components

Requirement	Verification	Points
1. Temperature Sensor (LM35) (a) The program should calculate the correct temperature based on the voltage output from the temperature sensor. The value should be within an accuracy of 95%.	1. Temperature Sensor (LM35) (a) Compare the calculated temperature with the real temperature read from the room air conditioner.	2
2. Seven Segment Display (ACDA03-41CGKWA-F01) (a) Temperature displayed on the LED must be of the same value as the one read from the temperature sensor and generated by the program. (b) Each segment should be in series with resistor of appropriate resistance to have moderate luminosity.	2. Seven Segment Display (ACDA03-41CGKWA-F01) (a) Write code to read current temperature from the temperature sensor and print it to monitor. Compare the value with the displayed number. (b) Measure the voltage across each segment. The voltage should be around 2.2V	5
3. Switch (a) Should have a pull-up voltage; outputs high when released and low when pressed (b) The program flags, e.g. Auto, Up and Down, should be set to its opposite Boolean value once the according switch is pressed.	3. Switch (a) Use program to print digital reading of the pin which connects to the switch. The output should be 1 when button is released and 0 when pressed. (b) Print flag value before and after the switch is pressed. It should change from TRUE to FALSE or from FALSE to TRUE.	3
4. I^2C-compatible Analog-to-Digital Converter (ADS1115) (a) With Adafruit_ADS1115 library, the program can read pin A0-A3 values. (b) The chip can communicate with microcontroller through SDA and SCL successfully.	4. I^2C-compatible Analog-to-Digital Converter (ADS1115) (a) Connect a working temperature sensor to each pin. Warm or cool down the sensor. The program should output increasing or decreasing value. (b) Same as (a).	3
5. I/O Expander (PCA9535) (a) The program should set all the 16 bit pins as output. Each bit should output low to turn on LED, and high to turn off LED. (b) The program should have a digit mapping from each output pin to	5. I/O Expander (PCA9535) (a) Write a blinking program to alternate each bit output between high and low. The 7-segment display LED driven by this chip should blink.	5

each segment in the 7-segment display such that it should show the integer value input from the program.	(b) Input the program with values from 0 to 99 and the dots and check if the 7-segment display displays the corresponding values.	

2. Web Module

2.1 Web Server

Requirement	Verification	Points
1. The web server should be able to read following information: city that user lives in, the desired temperature and desired window level. The back end should fetch correct weather information based on the city. Such information should be available in 1 minute: the desired window level, the desired room temperature, whether it is in bad weather condition outside, and the outside temperature. 2. The web server should be able to communicate with the microcontroller through TCP and HTTP. The Web server should be able to handle http request and send back correct information on: the desired window level, the desired room temperature, whether it is in bad weather condition outside, and the outside temperature. 3. The web server should be able to show connection condition. The "connection established" should be updated within 30 seconds, and the "connection lost" should be updated in 3 minutes.	1. Manually enter different values and click on buttons. Use curl command to check if the fetched content is the same as the true weather conditions. 2. Use application called "postman" to send http request, check the reply message is correct. 3. First, check that the web page shows "connection not viable". Use application called "postman" to send http request. Check that the web page shows "connection is viable". Wait for 3 minutes, and see if the web page shows "connection not viable" again.	10

3. Window Module

3.1 Infrared Sensor

Requirement	Verification	Points
1. Window stops moving when object is detected, regardless of the way microcontroller was trying to move the window.	1. Simulate situations such that microcontroller will move the window. While the window is moving, place any object between the upper part and bottom part of the window and see if the window stops.	2

3.2 Dual H-Bridge Motor Driver

Requirement	Verification	Points
1. The motor driver can communicate with the microcontroller successfully. 2. The motor driver should drive motor in two directions, clockwise and anti-clockwise. 3. The motor should be driven in relatively safe and fast speed. The window should be adjusted by one level up/down in 10 seconds.	1. Use a small motor for simplicity. Analog Write to each motor pin, the pin should output some significant voltage. 2. Use a small motor for simplicity. Analog Write to motor pin A with significant value, and the other pin B with 0 value. The motor should rotate clockwise, and vice versa. 3. Given the full speed (1024) we use Analog Write, the motor should adjust the window to a level up or down in 10 seconds.	5

Appendix B PCB Schematic