Automated Window Temperature Regulator

Derik Lee, Hersh Singh, Louis Wright
Team 22

Design Document for ECE 445: Senior Design
Fall 2019

September 25, 2019

Introduction **1**
Objective 1
Background 2
High-level requirements list 2
Design **3**
Block Diagram 3
Physical Design: 4
Block Design 6
Control Module 6
Motor Module 7
User Interface Module 8
Sensor Module 8
Requirements and Verifications 14
Control Description 18
Tolerance Analysis 19
Cost and Schedule **21**
Cost Analysis 21
Labor 21
Materials 21
Grand Total 22
Schedule 22
Discussion of Ethics and Safety **24**
Citations **25**

1. Introduction

1.1. Objective

Most houses and buildings tend to drive a lot of energy in the form of electricity into the HVAC (Heating, Ventilating and Cooling) system. This system is used to regulate and maintain temperature and air quality at a comfortable level for the occupants. As the threat of climate change grows, new technology is emerging to curb the impact of energy consumption and find alternative and efficient methods to electricity. Efficient interior climate control can be used to limit the power consumption HVAC systems within apartments and homes. As an alternative to using the air conditioning system, we can harness the outside climate to help regulate the interior temperature and air quality. This will take the strain off the HVAC system, resulting in lower power consumption. A simple and common way for the outside climate to enter the feedback loop between the HVAC and room is to open a window! Opening the window a certain amount can dictate how much the exterior temperature changes the temperature inside. Using this natural diffusion of air and temperature allows for less electricity to be consumed as well as a natural cross-ventilation for the room.

Our proposed solution is to use a window that is attached to a motor that opens and closes accordingly when given a certain desired interior temperature. The window will have sensors that measure rain and temperature, which will be fed into a microcontroller, which will compute the size of the opening of the window so the outside temperature can help regulate the interior temperature. An IR sensor will be attached for safety reasons in case there is obstruction as well as be used to calculate the position of the window. Once the sensor data is calculated, the microcontroller will drive the motor to change the position of the window to efficiently maintain the desired temperature (cool or heat). The system will be powered by a wall adapter and will have multiple modules for the sensors, motor, microcontroller and user interface.

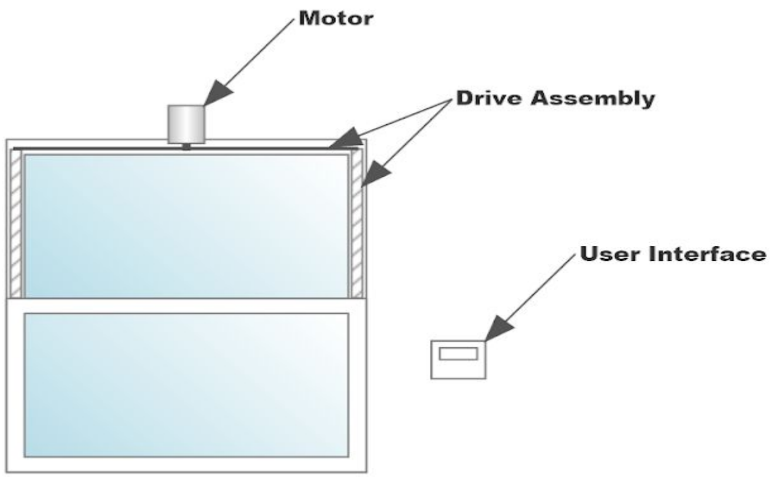


Figure 1: Visual representation of System

1.2. Background

For a typical homeowner, energy expenditure becomes a costly monthly fee. If people are looking to save money, most of the time they will turn off their thermostat or match it with the outside temperature. This does help, but using heating and cooling in a building can build up cost. The typical heating and cooling can make "up about 42% of your utility bill" [1]. Electricity is just one part of the utility bill, as there are water costs and sometimes natural gas costs as well. For reference, "The average monthly residential electricity bill in Illinois is $87… and is… less than the national average of $107 per month" [2]. By helping reduce the use of electricity and natural gas in temperature management, a homeowner could save hundreds of dollars a year.

For cooling savings, the best time to open a window would be during the evenings and nights. If someone were to leave a window open at night, it could end up disastrous if there was a nighttime storm, high humidity, or a spike in a particulate matter like pollen. The consequences could be damaged furniture, damaged paper products, high AC unit work for humidity, or allergic reactions at night and in the morning. By having the window itself manage opening and closing, leaving a window open would be inconsequential as if conditions become undesirable, the window will close itself.

1.3. High-level requirements list

- Window should compare outside temperature and indoor temperature to open and close at proper times.
- Window should stay open/closed based off of particulate matter in the air (< 30μg/m3), humidity (< 60%), and/or rain.
- User should be able to adjust temperature threshold for the window, and adjust the window opening manually if desired
- Ensure that the safety sensors top the movement of the window frame in case of obstruction that is detected

2. Design

2.1. Block Diagram

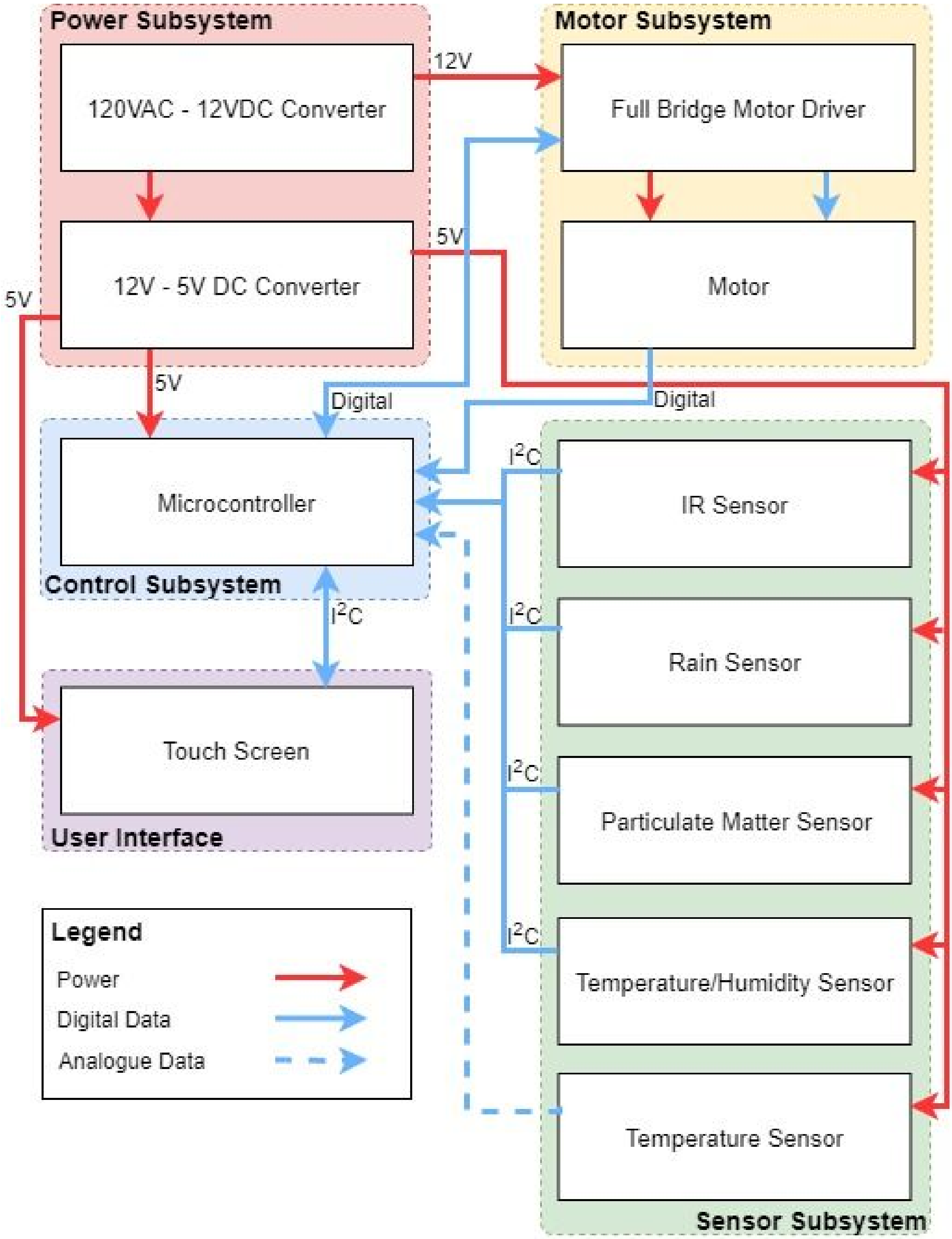


Figure 2.0: System Block Diagram

2.2. Physical Design:

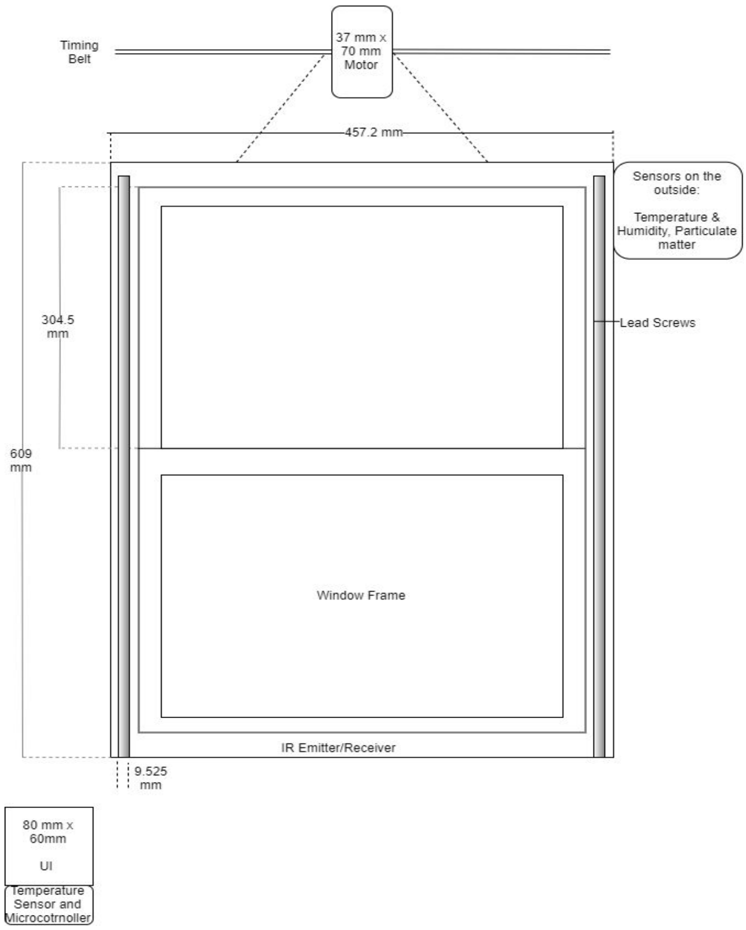


Figure 2.1: Physical Design

We will be working with an 18 inch by 24 inch, vertically hung window. On the sides of the frame we will have two lead screws that turn and carry the window frame up and down. These two screws will be driven by a timing belt that is attached to a 12 V DC motor with encoder that is attached to the top of the window. On the outside of the window at the to we will have an encasement that contains the temperature/humidity sensor and the particle matter sensor. On the bottom of the window, on the inside, we will have an IR Emitter Receiver that will help us detect any obstruction. Finally near the window we will have the touch screen UI along with the temperature sensor and microcontroller on the inside of the window. The sensors will be PCB mounted along with the microcontroller. The motor driver will be right next to the motor on top as well. The encoder will calculate the position of the window.

2.3. Block Design

2.3.1. Power Module

The power module is required to run the motor when the control module requires the window to be open or closed and is needed to supply power to the microcontroller, user interface, and sensors.

2.3.1.1. 120VAC-12VDC adapter

To supply power to the system, we will use the HitLights PWR-12V-060-30-UK. It claims an efficiency of >85%, which, although not ideal, is the most efficient given the budget available. For details regarding power and current requirements, see tolerance analysis (pg. 21)

Data:

Input Voltage	120VAC
Output Voltage	12VDC
Output Current	8A
Output Power	96W

Table 2.0: 120VAC to 12VDC Adapter Data

2.3.1.2. 12VDC to 5VDC

A TPS62133RGTR buck converter by Texas Instruments will be used to supply a 5V rail to the control, sensor and user interface subsystems. We require a steady output voltage for this component, as most dependant components require an input voltage range of ±5%

Input Voltage	3-17 V
Output Voltage	.9 - 6 V
Output Current	3 A
Output Power	1.2 - 10 W

Table 2.1: 12VDC to 5VDC Buck Converter Data

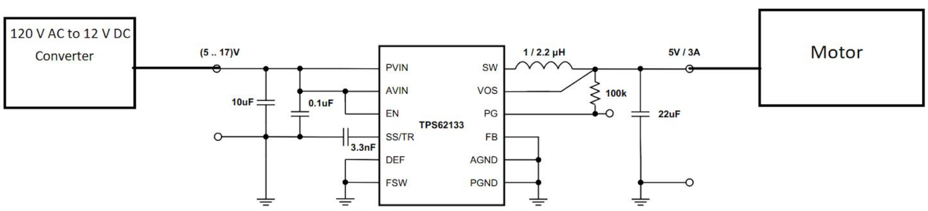


Figure 2.2 Buck Converter Schematic [11]

2.3.2. Control Module

The control module consists only of the microcontroller, which receives data from the peripheral devices, processes this information, then decides whether to activate the motor driver according to a set of predetermined parameters.

2.3.2.1. Microcontroller

Our design uses the ATMEGA328-PU microcontroller by Microchip Technology/Atmel. The microcontroller has 3 bi-directional ports (two 8-bit and one 7-bit), which we will use to interface with the various sensors, the motor driver and the touch screen. The microcontroller will read the data provided by rain, humidity and temperature sensors and make a decision as to whether to open or close the window (see Fig. 2.4 for details). Additionally, it will use the IR sensor inputs and the current sense input (from motor driver - pins PB5 & PB6 in Fig) to detect obstructions, and to determine when the window is in the closed position. Data from the motor encoder will then be used to determine the relative position of the window. The IR sensors and the temperature/humidity sensor will run off a single I2C interface.

Data:

Supply Voltage	5V
Supply Current	0.2mA

Table 2.2: Microcontroller Data

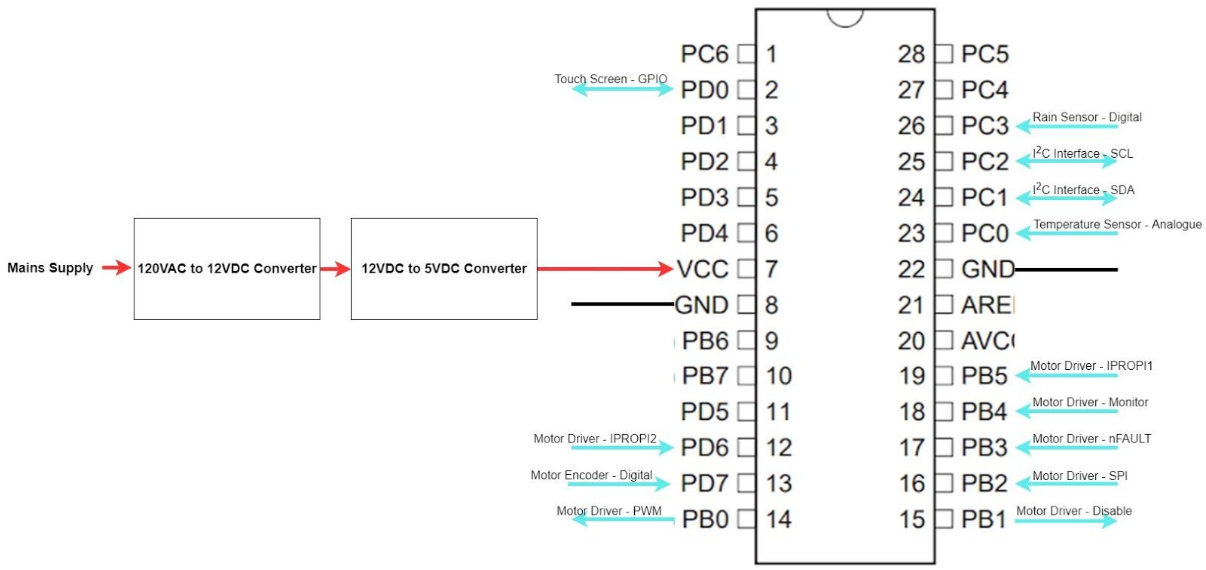


Fig 2.3: Microcontroller pinout

2.3.3. Motor Module

2.3.3.1. Full bridge motor driver

To provide power to the motor, we will be using the DRV8873SPWPR full bridge motor driver by Texas Instruments. It will provide ±12V to the motor, depending on the input from the microcontroller, at up to 10A, which is sufficient to handle the peak motor current draw of 5.5A. Additionally, it will provide a current sense data output, which the motor driver can use to detect any faults in the motors' operation.

Data:

Supply Voltage	4.5 - 38 V
Logic Inputs	5 V
Output Current	10 A

Table 2.3: Motor Driver Data

2.3.3.2. 12V Brushed DC Motor with rotary encoder

We will be using the motor with part number 4754 from Polulu. This motor will be a 12 V brushed DC geared motor with a gear ratio of 70:1. The force required to pull the window up is equivalent to lifting a mass of 1 kg. The gear that is attached to the motor to go on the timing belt has a diameter of 5.5 cm. The calculation for the minimum amount of torque is:

Torque from the screws:

$$N \times \mu \times (Clamp\, force)$$

$$(1 \times 9.8) \times 0.2 \times 9.525 \ = \ 18.669\ N * mm$$

Torque with gear ratio:

$$18.669 \times 2 = 37.338\ N * mm$$

Convert to Kg:

$$33.338 \times 0.102 = 3.80848\ Kg * mm$$

For maximum efficiency, the motor can operate at a torque at or below 32 Kg*mm, and the torque needed is much less at a window weight of 1 Kg. The friction of the window is not accounted for, but there is a very small amount of friction pulling the window up, so it is negligible. At maximum efficiency, the motor runs at 130 RPM, which will be fine for speed since the screws move ⅛ of an inch every rotation, which is 41.275 cm every minute. Since there is not much movement once the window is set, it does not need to move large distances instantaneously, so this speed will be sufficient.

Data:

Gear Ratio	70:1
No Load Speed	150 RPM
No Load Current	0.15 A
Stall Current	5.5 A
Maximum Power	10 W
Stall Torque	270 Kg*mm
Torque @ Max Efficiency	32 Kg*mm
Speed @ Max Efficiency	130 RPM

Table 2.4: Motor Data

2.3.4. User Interface Module

The user interface will allow the operator to either control the window position manually or set it to automatic mode, in which case the window will open and close according to the measurements made by the sensors.

2.3.4.1. Touch Screen

The touch screen is something we need as a user input to our system. It will be a simple display that allows the user to set the temperature they want their house to be at and the window will adjust accordingly. There will also be an off and on option so that one can turn the automatic window function off if it is not needed. The touch screen we are using is the model UNO R3 TFT Touch Screen by Elegoo.

Data:

Supply Voltage	3.3-5 V
Supply Current	120 mA

Table 2.5: Touch Screen Data

2.3.5. Sensor Module

The sensor module has all the sensors for our system and relays information back to the microcontroller. The information will come from a temperature sensor, IR sensors, a rain/humidity sensor, and a particle sensor. Taking all this data into account it will send information to the control module for what action the window should take.

2.3.5.1. Temperature Sensor

We need an indoor temperature sensor as well, which we will use model TMP235A2DBZR by Texas Instruments. We chose this sensor because it is compact, as it is a surface mount chip, and it is fairly accurate over a good spread of temperatures. The output will be an analog signal voltage output with voltages scaling based off temperature.

The temperature sensor can run at 5 V, which is ideal for what we want. The temperature range is definitely large enough for our purposes. This chip is very low power, with a maximum at 20.7-49.5 µW. Accuracy is also very good at a difference of 0.5 C.

Pins:

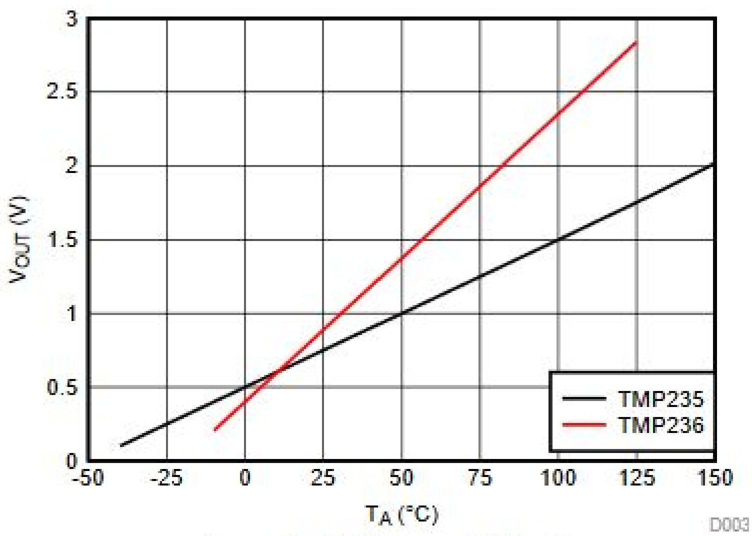
1	Vdd
2	Vout
3	GND

Table 2.6: Temperature Sensor pins out

Data:

Input Voltage	2.3-5.5 V
Current	< 9 µA
Temperature Range	-40-150 C (-40-302 F)
Accuracy	+/-0.5 C (0.9 F)
Output	Analog signal, the voltage is proportional to temperature

Table 2.7: Temperature sensor data



I_{OUT} = 0 µA, C_{LOAD} = 1000 pF

Figure 2.8: Temperature Sensor Voltage Characteristics [3]

2.3.5.2. Temperature and Humidity Sensor

For the outdoor temperature and humidity sensor, we will use HDC1080DMBR by Texas Instruments. Since indoor temperatures do not need a humidity check since AC keeps humidity at a certain level, it is not needed. We do however, need to check the outside humidity. Checking humidity is important since humidity can do a lot of damage on property as well as health depending on the level. High humidity will also increase the AC run time, which would increase electricity usage. This is a good temperature sensor because it is very accurate, only deviating by +/- 0.1 C. The humidity sensor is very accurate as well, deviating only by +/- 2 % Relative Humidity. Although the humidity sensor does have an age to it, it only changes by +/-0.25 % RH per year, which is a very slow decline.

This temperature and humidity sensor is good because it has a large operating temperature for both the temperature sensor and the humidity sensor, having a range of -20-85 C. This chip can also run at 5V, but has a large voltage range of 2.7 to 5.5 V. The output of the chip will be an I2C serial data line. This chip will be connected to the PCB as a surface mount.

Pins:

1	SDA (Serial Data Line)
2	GND
3	NC
4	NC
5	VDD
6	SCL (Serial Clock Line)

Table 2.8: Temperature/Humidity sensor pin out

Data:

Input Voltage	2.7-5.5 V
Current	Sleep Mode: Average 150 nA 1 μA at highest temperature Active Mode: 125-250 μA depending on temperature and Vdd
Temperature Range	-20-70 C (-4-158 F)
Accuracy (Humidity)	+/-2 %RH

Accuracy (Temperature)	When 5 C < T < 60 C: +/-0.1 C F: When 41 F < T < 140 F: +/- 0.18 F
I2C Clock Frequency	10- 400 kHz
I2C Clock Low Time	$\geq$1.3 μs
I2C Clock High Time	$\geq$0.6 μs

Table 2.9: Temperature/Humidity sensor Data

2.3.5.3. IR Sensor

The IR sensor we chose is model VCNL4020C-GS08 from Vishay Semiconductors. This IR Sensor will be part of a PCB since it is a surface mounted chip. We chose this chip because we wanted an IR sensor that fit on a window without blocking anything and this sensor is very small and all it will need is a PCB so it can work.

The IR sensor sends out information using I2C. The recommended maximum voltage only goes to 3.6 V, compared to the 5 V we plan on powering all sensors with. Although the absolute maximum voltage rating for this chip is 5.5 V, we will add resistors to the PCB to create a voltage divider so that the voltage can be dropped to 3.6 V. Although the maximum current seems very high at 4 ma, this is a maximum current with 250 measurements per second and the LED is being pulsed. Usually, current consumption on standby is 1.5-2 μA, and with lower measurement speeds, would reach 0.64 mA at most. Power is not too large of a factor if we limit the measurements and how often it is turning on. There is also an interrupt function that can be used on this chip.

Pins:

1	IR anode
2	SDA (Serial Data Line)
3	INT
4	SCL (Serial Clock Line)
5	VDD
6,7	NC
8,9	GND
10	IR cathode

Table 2.10: IR sensor pin out

Data:

Input Voltage	2.5-3.6 V
Input Voltage for IR Anode	2.5-5 V
Max Current	4 mA
INT High Level Voltage	1.7-5 V
INT Low Level Voltage	≤0.4 V
Output Type	I2C
I2C Clock Rate Range	3400 kHz

Table 2.11: IR sensor data

2.3.5.4. Rain Detector

The rain detector is a model a13082300ux1431 from Uxcell. Most rain detectors have similar form of some kind of flex resistor. We chose this rain detector because the price is low and the sensitivity can be adjusted using a potentiometer. The rain detector is necessary because we need to check to see if it is raining outside before a window can be opened. At the moment, we have not found a correlation between rain and humidity, so we will have both detectors.

This rain detector also comes with its own board, which can output a digital switch output or an analog output. The output will be high when there is no rain and low when rain is detected. The power required for the rain detector will be no more than 0.0495-0.075 W. This detector can also run at 5 V, with a range of 3.3.-5 V.

Data:

Input Voltage	3.3-5 V
Input Current	≤15 mA
Output	Active Low Digital Output

Table 2.12: Rain sensor data

2.3.5.5. Particulate Matter Sensor

The particulate matter sensor is a dust sensor (Model PPD42NS by Shinyei Corporation). The particulate matter sensor will be on the outside of the window. The function of this sensor is to detect particulate matter levels outside like dust and pollen, things that are detrimental to the inside environment. This sensor was chosen because it is robust since it can handle temperatures from 0-45 C. It also can detect particles larger than 1 μm, which is a good range for detecting pollen and dust (Ex: the size of pollen is 6 μm).

The device operates in voltages between 4.5-5 V and the power needed for this device goes to a maximum of 0.405-0.495 W. The output of this device will go straight into the microcontroller since it is a digital signal (PWM). Based on the concentration of particles, the duty cycles will hit a low pulse occupancy percent up to 16%.

Data:

Voltage Input	4.5-5.5 V DC
Current	Up to 90 mA
Operating Temperature	0-45 C (32-115 F)
Output	PWM wave, negative logic High: > 4 V Low: < 0.7 V Unit Wave time: 30 sec

Table 2.13: Particulate matter sensor data

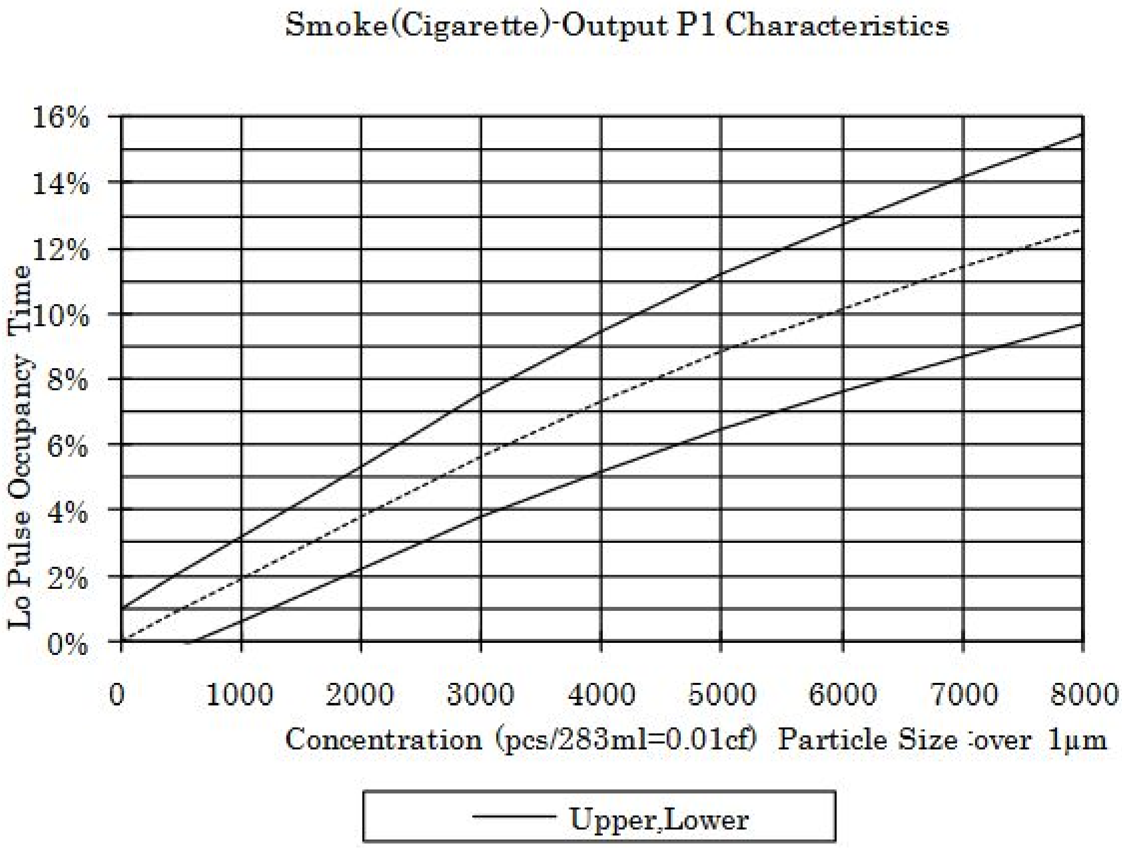


FIgure 2.5: Particulate matter sensor characteristics [4]

2.4. Requirements and Verifications

Module	Requirements	Verification
120V-12V Converter	Convert 120VAC to 12VDC from a wall plug with ≥85% efficiency	1. Connect adapter to wall outlet 2. Measure input and output voltage and current using DMM 3. Calculate efficiency
12V-5V Converter	Converts 12VDC to 5VDC with ≥95% efficiency	1. Connect converter to 12VDC supply 2. Measure input and output voltage and current using DMM 3. Calculate efficiency
	Maintains steady output under all load conditions (within 5% 5V at output)	1. Attach variable load to output 2. Observe output behaviour on oscilloscope with varying load, including during startup and shut down (simulated by switching power source on and off)
Microcontroller	Responds correctly to all given input combinations ● Temperature ● Humidity ● Rain ● Obstructions (IR sensors) ● Current window position	1. Check logic using computer simulations on code 2. Connect input pins to relevant simulated interfaces (I2C, Analogue, PWM) 3. Connect oscilloscope to motor driver output pin 4. Vary inputs and measure output using DMM 5. Check that output matches control flowchart
	Interfaces correctly with touch screen	1. Connect touch screen to microcontroller via I2C interface 2. Check logical response at output of microcontroller for any functional touch inputs

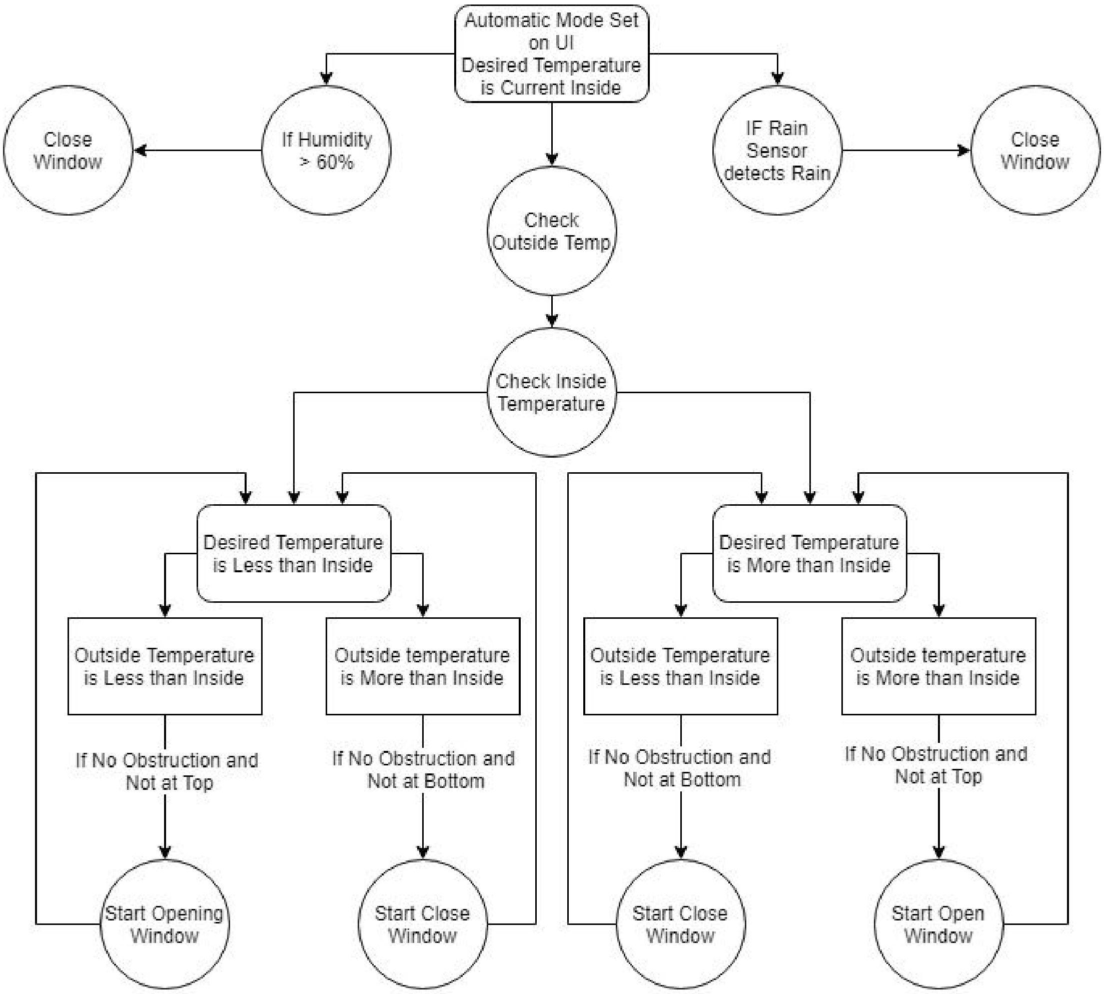
<table>
<tr><td rowspan="1">Temperature Sensor (indoor)</td><td>Measures temperature to within ±1°C</td><td>1. Place sensor in small air conditioned room
2. Allow AC to reach thermostat set temperature
3. Measure output of temperature sensor using oscilloscope
4. Vary AC temperature to extremes
5. Check sensor matches thermostat temperature</td></tr>
<tr><td rowspan="2">Temperature/Humidity Sensor (outdoor)</td><td>Measures temperature to within ±1°C</td><td>1. Place sensor in small air conditioned room
2. Allow AC to reach thermostat set temperature
3. Measure output of temperature sensor using oscilloscope
4. Vary AC temperature to extremes
5. Check sensor matches thermostat temperature</td></tr>
<tr><td>Measures humidity to within ±5%</td><td>1. Place sensor in various environments of different humidity levels (e.g. outside, room with HVAC, sauna)
2. Measure actual humidity using sensor of known accuracy
3. Read data from I2C interface and compare to measured value</td></tr>
<tr><td rowspan="2">IR Sensors</td><td>Detects obstructions with accuracy ≥95%</td><td>1. Mount sensor in window and attach to microcontroller
2. Put the system in manual mode
3. Start the window closing, then obstruct the window with a rod
4. Check that window stops</td></tr>
<tr><td>Detects closed position of window</td><td>1. Move window to closed position
2. Check that IR sensors provide correct output</td></tr>
</table>

Rain Sensor	Detects light rain with accuracy ≥80%	1. Spray sensor with spray bottle on mist setting 2. Check output with DMM
	Detects heavy rain with accuracy ≥95%	1. Drop single drop of water onto sensor 2. Check output with DMM
Particulate Matter Sensor	Measures particulate matter content to within ±5μg/m3	1. Place sensor in various environments of different known PM levels 2. Measure actual PM concentration using sensor of known accuracy 3. Read data from I2C interface and compare to measured value
Motor Driver	Outputs ±12V (within ±5%) current up to 5.5A (stall current of motor)	1. Attach driver to 12VDC source 2. Attach variable load to output 3. Apply logic inputs for forward, reverse and brake functions 4. Vary load between 2.2Ω and 80Ω to simulate different motor operating modes 5. Use oscilloscope to observe output
Motor	Opens and closes window 25cm in 30 seconds	1. Attach motor to power supply and window set up 2. Measure time taken for window to fully open from closed position 3. Measure time taken for window to return to closed position
	Encoder provides accurate relative position of motor (within 0.1°)	1. Check datasheet

Touch Screen	Displays information described in Figure 2.6	1. Configure display with power source and microcontroller 2. Check spacing and color of display
	Responds to touches accurately	1. Configure display with power source and microcontroller 2. Touch all buttons and check output to see if it matches 3. Touch all non functional parts of the screen, ensure no response

Table 2.13: Requirements and Verification of Design

2.5. Control Description



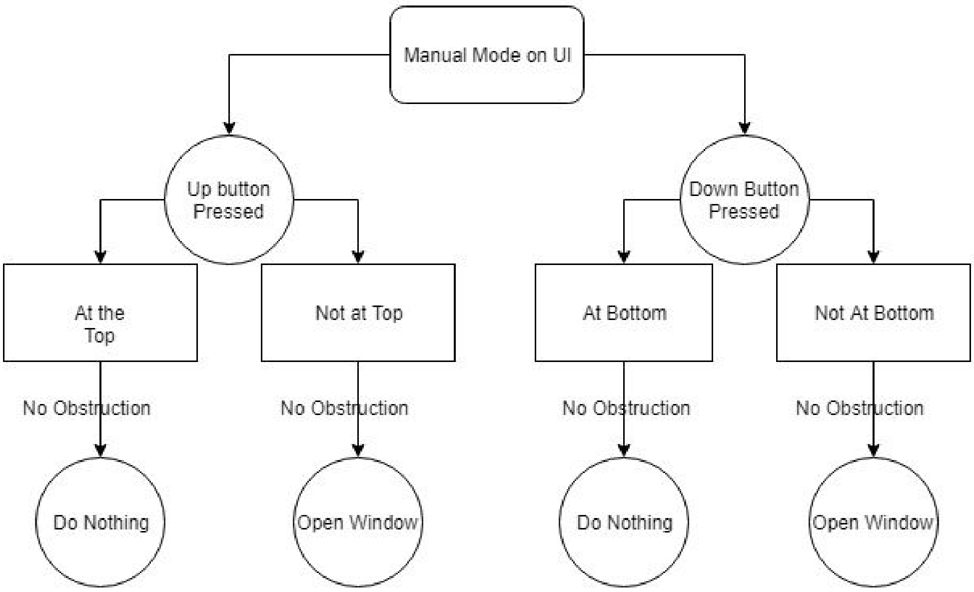


Figure 2.6: System operational flow diagram

In our design (Figure 2.6), there are two different modes that can be processed by the microcontroller. For the first mode, we set a desired temperature on the UI. We then get data from both the inside and outside temperature sensors. Depending on if the desired temperature is more or less than what we desire, we either close or open the window. If there is precipitation outside then the entire system goes and keeps the window closed to make sure that the water is not entering the facility. If there is no precipitation, the control module continues on. If it is colder outside than it is inside and the temperature we want is less than the current inside temperature we will open the window, if it is warmer outside than inside then we will close the window. On the other hand if the desired temperature set is more than the current if it is warmer inside than outside we will close the window and if it is warmer outside than it is inside, we will open the window. Through this entire process we will keep track of the position of the window. This way we can detect if the window is at the maximum or minimum position and we can stop the motor from moving the frame. Finally, another check we constantly will be doing is that if if the IR detects and obstruction on the window, then we will halt all processes and stop the motor from running and moving any direction. We plan to send data to the UI to show that there is obstruction, stopping the window from moving up or down.

2.6. Tolerance Analysis

The power module of this system is critical to its success, as all other modules rely on it to function. Additionally, the main purpose of the system is to save energy, and the largest source of power loss will be in the 120VAC to 12VDC adapter, so it is essential that this component satisfies efficiency requirements.

The 120VAC to 12VDC adapter is rated to 8A. If the current draw exceeds this rating, it would be detrimental to the system, causing potential overheating, which may be a fire hazard.

Subsystem	Current Required (A)
Sensors	30u
User Interface	30m
Motor	5.5
Microcontroller	2m
Total:	5.5

Table 2.14: System current draw breakdown

As seen in Table 2.9, the maximum total system current draw is 5.5A, which is 2.5A below the maximum rating of 8A. This gives us a margin for error of 45%. Table 2.10 contains a breakdown of the power consumption of all major system components.

Component	Power Consumption (W)
120VAC to 12VDC adapter	0.02*
Sensors	1.5e4
User Interface	0.15
Motor	0.09**
Microcontroller	0.01
Total	0.27

Table 2.15: System power consumption breakdown

*Assuming motor is active 5% of the time, drawing an average of 0.15A at 148RPM (See Figure 2.6, τ = 3.995), the power loss is given by: $0.15(VI_{active}T_{active} + VI_{idle}T_{idle})$

**Assuming motor is active 5% of the time, drawing an average of 0.15A at 148RPM (See Figure 2.6, τ = 3.995), the power usage is given by: $VI_{active}T_{active}$

Considering that a typical HVAC system will use up to 5000W [10], the power consumption of this system is negligible.

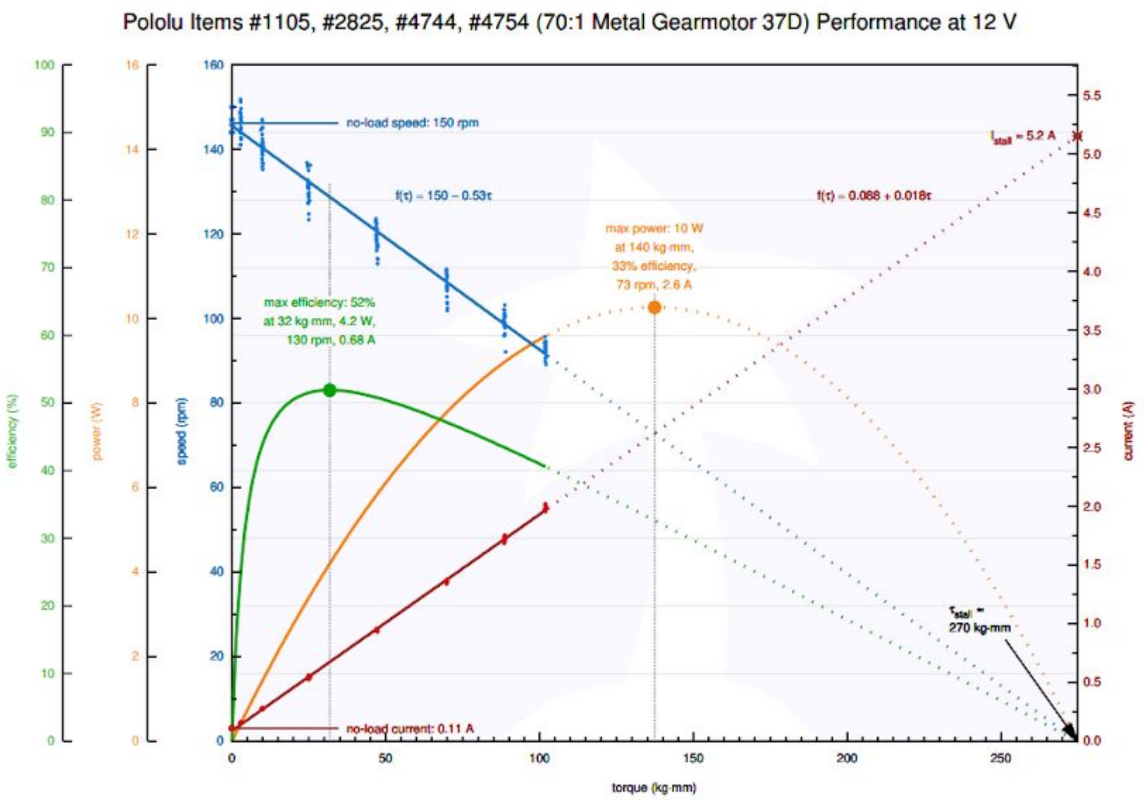


Figure 2.7: Performance characteristics of motor [9]

3. Cost and Schedule

3.1. Cost Analysis

3.1.1. Labor

Based on an average graduate salary of $76,079 [8] for UIUC electrical engineering graduates, our hourly rate is estimated to be $35/hour per person. Assuming 10 hours of labour per person per week, and 16 weeks of development, the total labor cost will be:

$$\$35 * 10\ hours/week\ *\ 10\ weeks\ *\ 2.5\ *\ 3\ members\ = \$26\,250$$

3.1.2. Materials

Description	Manufacturer	Part No.	Quantity	Cost
Sensor Subsystem				
Board Mount Temperature Sensor	Texas Instruments	TMP235A2DBZR	1	$0.90
Board Mount Humidity and Temperature Sensor	Texas Instruments	HDC1080DMBR	1	$2.80
Rain Sensor	Uxcell	a13082300ux1431	1	$14.16
Particulate Matter Sensor	Seeed Studio	101020012	1	$18.50
IR Sensor	Vishay Semiconductors	VCNL4020C-GS08	3	$8.70
Motor Subsystem				
DC Geared Motor with 64-bit Encoder	Polulu	4754	1	$39.95
Full Bridge Motor Driver	Texas Instruments	DRV8873SPWPR	1	$4.04
Control Subsystem				
Microcontroller	Microchip Technology	ATMEGA328-PU	1	$1.95
Power Subsystem				
120VAC - 12VDC Converter	HitLights	PWR-12V-060-30-UK	1	$26.69
12V - 5V DC Buck Converter	Texas Instruments	TPS62133	1	$2.15
User Interface				
LCD Touch Screen	Fujitsu	N010-0554-T703	1	$12.52
Other				
Window*	-	-	1	$80.00
Mechanical Parts*	-	-	-	$20.00
Miscellaneous Electrical Parts	-	-	-	$10.00
PCBs	-	-	3	$30.00
TOTAL				$272.36

Table 3.1: Material cost breakdown

* Window and mechanical part costs are covered by the machine shop, and thus are estimated to give a more accurate material cost.

3.1.3. Grand Total

Taking into account both the labor and material costs, the grand total for this project will be $26522.36

3.2. Schedule

Week	Derik	Hersh	Louis
9/16	Research Sensors	Research Motors	Research Power
	Research component specifications	Research component Specifications	Research window and motor specifications
9/23	Research component specifications	Research component specifications	Research window and motor specifications
	Start design documentation	Start design documentation	Start design documentation
9/30	Finish design documentation	Finish design documentation	Finish design documentation
	Place Parts Order	Find exact parts to order	Find exact parts to order
10/7	Research PCB TTL chips	Order all leftover parts	Test any parts in this week
10/14	Start PCB design	Start PCB design	Start PCB design
	If applicable, start PCB assembly If not, Test new components	Test motor subsystem on window	Test all of the new components
10/21	Finish motor system on window	Finish PCB assembly	Start adding sensors to the window
10/28	Add user layout to the window	Program the microcontroller	Finish adding all the sensors
11/4	Test user layout on window and systems	Program the user layout	Start debug of sensors and motor subsystem

11/18	Debug programs, sensors, and motors	Debug programs, sensors, and motors	Debug programs, sensors, and motors
11/25	Thanksgiving Break	Thanksgiving Break	Thanksgiving Break
12/2	Final Demo Preparation	Final Demo Preparation	Final Demo Preparation
12/9	Final Report	Final Report	Final Report

Table 3.2: Project Schedule

4. Discussion of Ethics and Safety

We will ensure that our project will take into account the safety and concerns of the user. The ethical side of our project will be to maintain the veracity of our claims that this project will help save energy. Because this is an automated system which runs unsupervised, we want to be transparent with concerns we have ourselves and convey them to the user. We will be transparent with safety issues we see and warn the user to follow the IEEE code of ethics; more specifically, the first point where we will “disclose promptly factors that might endanger the public or the environment,” [5]. Another IEEE Code of Ethics point that we will follow is “avoid injuring others, their property, reputation, or employment by false or malicious action,” [5]. We will ensure that the window will have sensors and overrides any action to stop movement in case of obstruction.

Another safety concern we have is the sensors we have are electronic and may be exposed to hazardous weather. Water can cause the circuitry to short circuit, which presents a fire hazard and risk of electrocution, which could cause damage to the home, window and anyone close by. We shall make sure that all circuits and sensors are properly shielded from any potential situations that could lead to this concern. Next, we also want to make sure that while working on the system, we take all the safety precautions in the lab. Working with power, circuits, and tools we could cause hazards like explosions and electrical fires if we are not careful. If we are careful and follow all lab procedures from the ECE 445 guidelines and ask for help on concerns we have, we will be protecting ourselves, classmates from any dangerous situations in the lab.

Another ethics concern we foresee whether our promise of reducing energy consumption in homes is true. We want “to be honest and realistic in stating claims or estimates based on available data;” [5]. Our main goal is to make an environmentally friendly window system to reduce power consumption homes and advertise the product as such only if it truly does curb wasted energy. Considering that the typical HVAC unit uses up to 5000W [10], and this system uses an average of 0.27W (see tolerance analysis for detail, pg. 22), if implemented properly, this should not be a concern.

Finally, as a group of three, we will follow the Code of Ethics and hold each other accountable to follow these concerns of ethics and safety. Point nine states “assist colleagues and co-workers in their professional development and to support them in following this code of ethics,” [5]. Working together to ensure we meet these standards will ensure that our project will be made without malicious intent and ensure that we take into account the safety of the user at all phases.

5. Citations

[1] Energy.gov. (2019). *Home Heating Systems*. [online] Available at: https://www.energy.gov/energysaver/heat-and-cool/home-heating-systems [Accessed 12 Sep. 2019].

[2] Electricity Local. (2019). *Champaign, IL Electricity Rates*. [online] Available at: https://www.electricitylocal.com/states/illinois/champaign/ [Accessed 12 Sep. 2019].

[3] Texas Instruments. (2019). *TMP23x Low-Power, High-Accuracy Analog Output Temperature Sensors.* [online] Available at: http://www.ti.com/lit/ds/symlink/tmp235.pdf [Accessed 01 Oct. 2019]

[4] Seeed Studio. (2019) *Grove - Dust Sensor.* [online] Available at: https://www.seeedstudio.com/Grove-Dust-Sensor-PPD42NS.html [Accessed 01 Oct. 2019]

[5] IEEE.org. (2019). *IEEE Code of Ethics*. [online] Available at: https://www.ieee.org/about/corporate/governance/p7-8.html [Accessed 13 Sep. 2019].

[6] US EPA. (2019). *Particulate Matter (PM) Basics | US EPA*. [online] Available at: https://www.epa.gov/pm-pollution/particulate-matter-pm-basics [Accessed 16 Sep. 2019].

[7] Illini Success, (2018), *Illini Success Annual Report 2017-2018*, pg. 19. [online]. Available at: https://uofi.app.box.com/s/ml9oh48vawrhm7e019kw5ix4j91j605p [Accessed 28 Sep. 2019]

[8] Elegoo.com. (2019) *ELEGOO UNO R3 2.8 Inches TFT Touch Screen*. [online] Available at:https://www.fujitsu.com/downloads/MICRO/fcai/touchpanels/control-board-single-input.pdf [Accessed 01 Oct. 2019]

[9] Pololu Robotic & Electronics. (2019). *37D Metal Gearmotors,* [online] Available at: https://www.pololu.com/file/0J1706/pololu-37d-metal-gearmotors.pdf [Accessed 10 Oct. 2019]

[10] EnergyUseCalculator.com. (2019). *Electricity Usage of a Central Air Conditioner,* [online] Available at: http://energyusecalculator.com/electricity_centralac.htm [Accessde 16 Sep. 2019]

[11] TI.com. *(2019) TPS6213x 3-V to17-V, 3-A Step-Down Converter In 3x3 QFN Package*, [online] Available at: https://www.ti.com/lit/ds/symlink/tps62133.pdf [Accessed 3 Oct. 2019]