MODULAR AUTONOMOUS HOME LIGHT

By

Cary Chai (caryzc2) [Intro, High Level Requirements, Phone App Subsystem, Schedule, Ethics]

Makomborero Tizora (mtizor2) [Block Diagram, PCB Schematics, Component Choices]

Samuel Darmamulia (sid2) [Switch Subsystem, Sensor Subsystem, Tolerance Analysis, Costs]

Design Document for ECE 445, Senior Design, Spring 2020

TA: Chi Zhang

27 February 2020

Project No. 52

Contents

1. Introduction

1.1 Objective

People love the convenience and the energy savings that automation can bring. It's for this reason that home automation is so prevalent today. One area that has seen developments in automation is lighting. Especially in the bustle of everyday life, it is often easy to forget to turn off the lights which wastes energy and money. On average, a household uses $200 a year for electricity on lighting, not including the cost of bulbs used overtime. [1] Modern commercial buildings often come built with some of these automation features pre-installed. For example, in Electrical and Computer Engineering Building (ECEB), lighting is automated, but in many older buildings and especially in apartments and personal homes, automatic lighting is not implemented as there is currently no cheap, modular solution for autonomous lighting without having to hire an electrician and rewire the circuitry or purchase an entire expensive smart home ecosystem.

Our goal is to create a cheap modular solution which can be implemented on manual light switches to automatically control lights without needing to rewire a building's circuitry. This way, typical individuals and families can afford to have autonomous lighting installed without needing to hire an electrician to. This will be achieved through two modules. One that will attach externally to existing light switches and a second module that will hold a sensor unit consisting of two passive infrared (PIR) sensors. These two units will communicate with each other to provide modular, reliable automatic lighting to families. We will also have a phone app that will allow for manual control of the sensors as well. In addition, current solutions for occupancy detection which focus on full room motion detection, there is a significant failure rate for still occupants. [11][12] Therefore, we aim to implement our occupancy detection using a sensor unit focused on the entrance to detect as people enter and leave the room instead of the motion in the entire room. The sensor unit will communicate with a central computing unit through Bluetooth. After an entry is detected, the computer will increment its count by one. When someone leaves, the device will count down one. At zero, the device will know that there are currently no occupants in the room.

1.2 Background

There are companies such as SwitchBot which are working towards easy implantation of internet of things (IoT) homes. However, their products are focused on manual switches that can all be controlled from your smartphone and still require human input. In addition, their products are very expensive. For example, SwitchBot's Button Presser costs $29. You then need to purchase a SwitchBot Hub, and the Mini runs for $49. This is roughly $80 for just setting up a manual system for lighting which you can control from a phone app. [2] The environment is not integrated with existing motion sensors, and regardless, these would need to be purchased separately.

Other solutions use motion sensing to turn on and off lights in a room. However, these rely on constant motion and are prone to failure when there is little to no motion in rooms. Our solution aims to be a modular solution that will have the full capabilities to autonomously control lighting in a room. It will also implement occupancy detection based not on persisting motion in the room but instead on the number of people currently in the room based off a count from the entrance.

1.3 Visual Aid

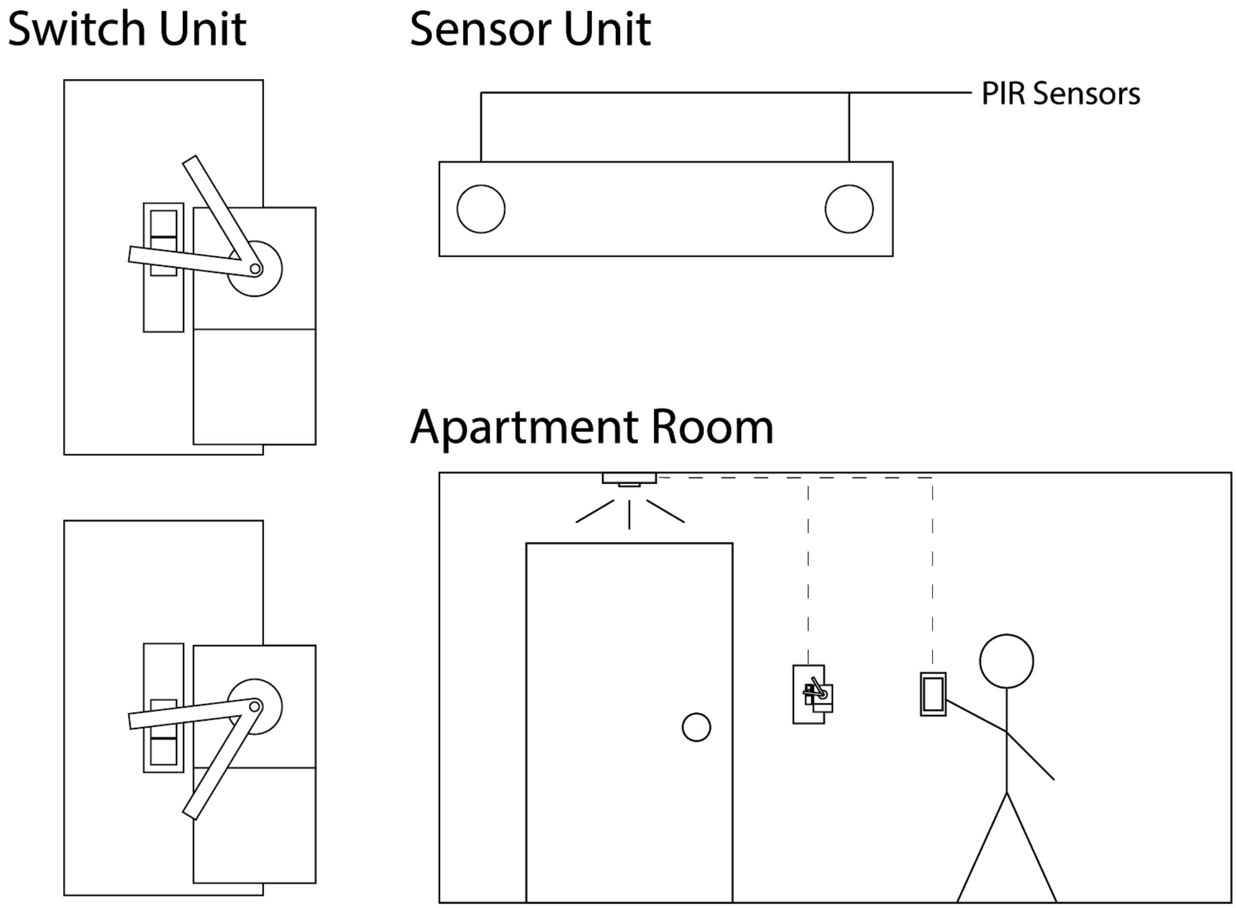

Figure 1: Visual Aid

On the left side of figure 1, we can see the switch unit and the design of how the motor will turn the switch on and off. On the top right of the figure is the sensor unit design. There are two infrared sensors on each side so the system will be able to figure out if the person left or entered the room based on which sensor triggers first. The placement of the sensor can be seen in the lower right side of the figure. The sensor unit will be placed above the entrance while the switch unit will be attached to the light switch. The sensor unit will communicate with the switch unit using Bluetooth for when to flip the switch. The phone will also communicate with the sensor unit directly through Bluetooth.

1.4 High Level Requirements List

- The sensor unit will be able to count the number of people in an apartment bedroom with 90% accuracy under normal daily usage for a week
    - *Normal daily usage will assume average walking speeds in an apartment setting and distance between individuals entering the room for personal space. It will also be

assumed that individuals will not be trying to actively avoid the sensor by moving along the wall instead of the center. More detailed explanation in the controlled test below.

- The switch unit will be able to flip on and off toggle light switches whenever a signal is received
- The phone will be able to communicate with the sensor unit when at your apartment
- Be able to last at least half a year on battery power

2. Design

2.1 Block Diagram

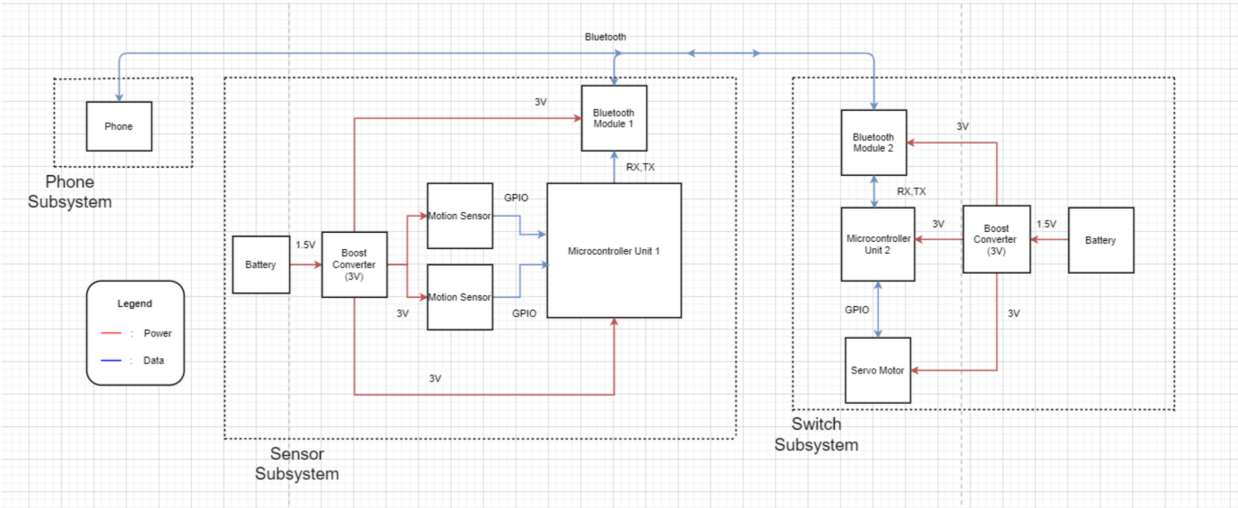

Figure 2: Block Diagram

Our block diagram has 3 subsystems - the phone subsystem, the sensor subsystem, and the switch subsystem. In the sensor subsystem, there are 2 motion sensors that will be able to count the people entering or leaving and give that information to the switch subsystem 1. If there are people in the room, the sensor subsystem will then communicate with the switch subsystem 2 through Bluetooth Module 2 and send a command to use the servo to flip the light switch. The phone subsystem can also communicate with the sensor subsystem to tell it to reset the count in the room and send a command to the Switch Subsystem to turn off the lights. All these subsystems are needed to fulfill our high-level requirements.

2.2 Switch Subsystem

The purpose of this subsystem is to flip the toggle light switch on/off. We decided to use a servo motor because it is lightweight, has enough force to flip a switch, and requires a smaller amount of power to run then a motor. For communication between the switch and sensor subsystems, we chose to use Bluetooth. The reason being, we wanted communication between the units to be subject to as few external factors as possible. If we used a Wi-Fi module and the Wi-Fi in the apartment went down, the units would not be able to communicate anymore which would effectively make our device useless. Bluetooth avoids this issue. We also chose to use a boost converter in order to boost our 1.5 V batteries to ~3.3 V. We can't use a linear regulator, because linear regulators do not boost the voltage, but steps the voltage down.

Requirements	Verification
1. Microcontroller needs to be able to receive data from the Bluetooth module 2. The servo motor needs to be able to generate at least 194 g of force to flip the toggle light switch. To convert to torque, we multiply 9.8 [m/s^2] * 0.194 [kg] * 0.0254 [m] = 0.048 [N*m] and we convert that to 0.4895 [kg-cm], which is below the torque given by the servo motor (2.5[kg-cm]) 3. Needs to be able to run 2.8 V - 3.3 V with a tolerance of ± 5%	1. Microcontroller Verification a. Connect the microcontroller on the switch subsystem PCB to the computer through mini USB. b. From the computer, we will run a script in the Arduino IDE that will output the data being sent from the Bluetooth module. 2. Servo Motor Verification a. To verify this, we will place the servo motor near a light switch. b. Run a script that will make the servo motor run. a. See if the servo motor has enough force and can turn on/off the toggle light switch 2. Use an oscilloscope to measure input voltage and the voltage at each of the components and make sure everything is 2.8 V - 3.3 V with a tolerance of ± 5%

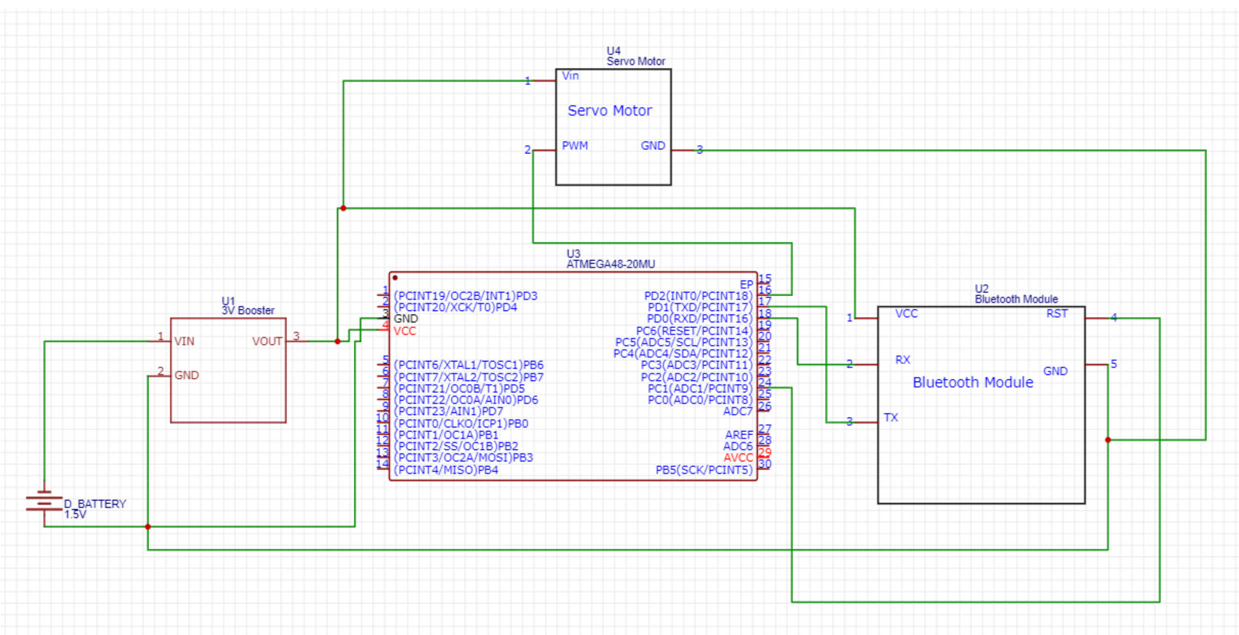

Figure 3: First Draft PCB of Switch Subsystem

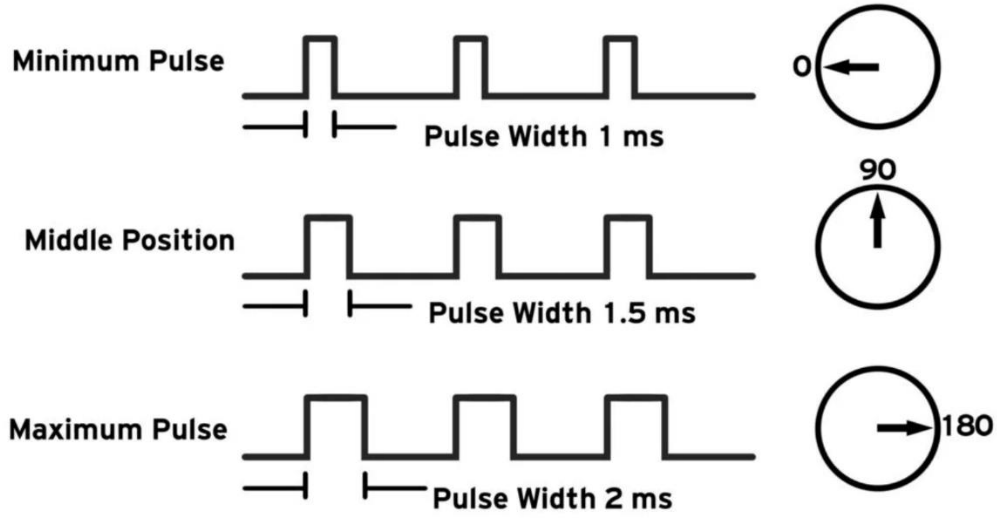

Figure 4: Timing Diagram of Servo [7]

2.3 Sensor Subsystem

The purpose of this subsystem is to allow the system to detect and count the number of people inside the room by using two PIR sensors. We will communicate this information to the switch subsystem using a Bluetooth module. We chose to use a Bluetooth module and boost converter for the same reason stated in the "Switch Subsystem" section. We chose PIR sensors as they require low power to run. Other methods exist for occupancy detection such as cameras using computer vision. However, they require a lot more power to operate effectively. For example, Reolink Argus is a battery powered security camera that requires 4 1600 mAh batteries that can only last 800 minutes, which is only 13 hours of active time. This camera consumes 11,428.57 mAh per day while active, and our sensor subsystem uses 82.56 mAh per day while active. [17] The PIR sensors help make our device run on battery power for a greater amount of time.

Occupancy Detection

Assuming a standard apartment room has one entrance, let us call the first PIR sensor that is closest to the door "Sensor A" and the PIR sensor farther from the door "Sensor B." We can detect people entering or leaving by which sensor triggers first. If Sensor A gets triggered and then Sensor B gets triggered, we know that the person is entering. If it is the other way around where Sensor B is triggered first before Sensor A, we know that the person is leaving the room. If one of the sensors is triggered and the other sensor has not triggered, then the count will not change. Apartment doorways are also typically narrow and only fit one person at a time. Under normal usage, this method for occupancy detection should be sufficient. We will create a CAD casing for the sensors to minimize the chances of miscounts when more than one person is below the sensors.

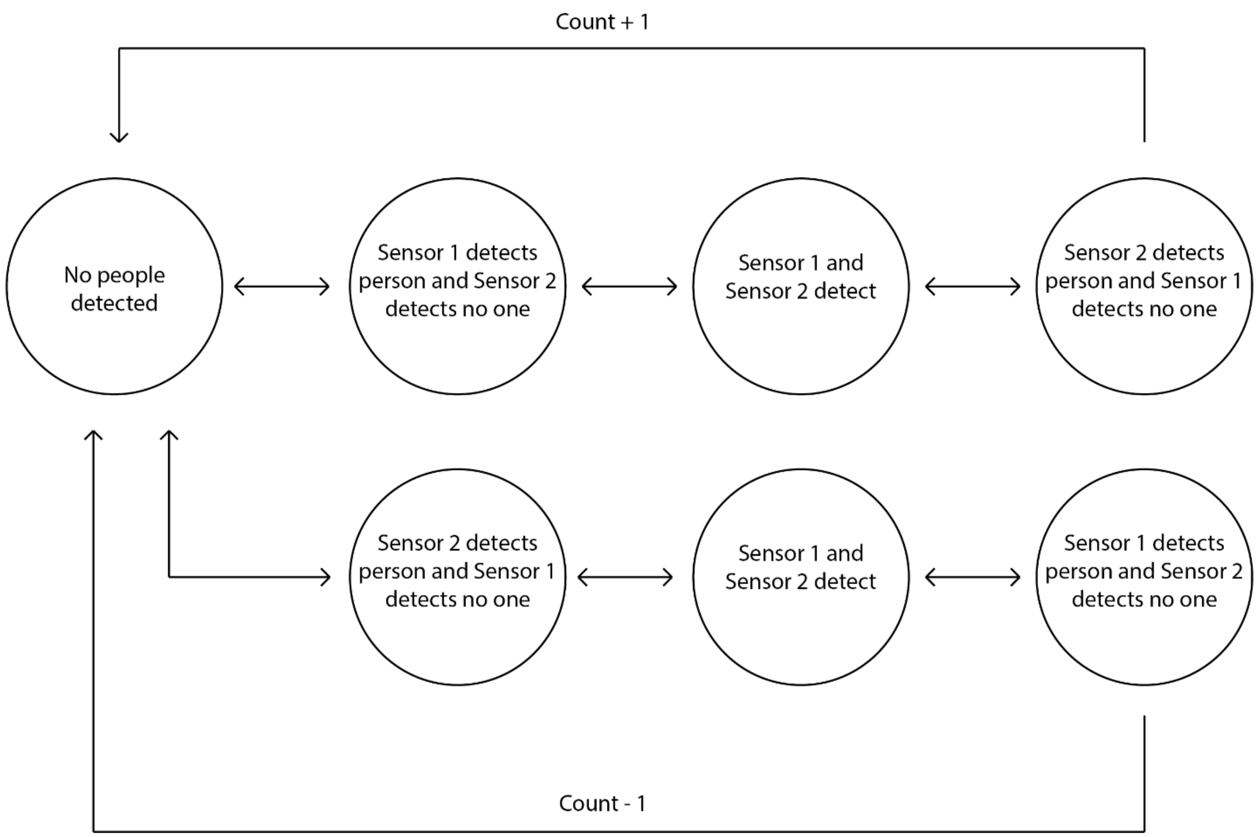

Figure 5: Sensor State Diagram

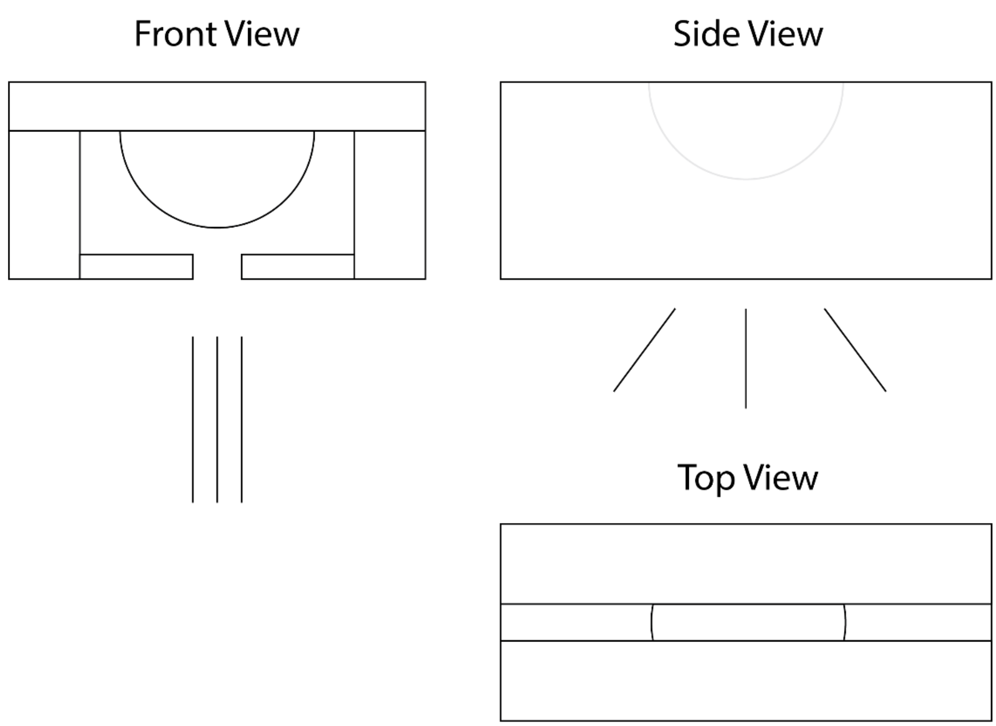

Figure 6: PIR Sensor Casing

Requirements	Verification
1. Motion Sensor needs to be able to detect motion from at least 2-3 meters away. 2. Microcontroller needs to be able to receive motion sensor data 3. Bluetooth Module needs to be able to receive data from the microcontroller and then transmit it to servo motor subsystem and the phone app 4. System should be able to work of 2.8 V - 3.3 V with a tolerance of ± 5%	1. Motion Sensor Verification a. Will connect the motion sensor to an independent voltage source. b. Connect the output line of motion sensor to an oscilloscope. c. Will wave a hand in front of the motion sensor and make sure that the output line goes high when motion is detected. 2. Microcontroller and Motion Sensor Communication Verification a. Connect the microcontroller which is on sensor subsystem printed circuit board (PCB) to the computer via micro USB. b. From the computer, we will run a script in the Arduino IDE that will continuously output the output line value. c. We will move across the sensor and see if there is the correct feedback. 3. Bluetooth Module Data Transmission a. Connect the microcontroller which is on sensor subsystem PCB to the computer via micro USB. b. From the computer we will run a script using the Arduino IDE that will allow us to open a serial monitor which will verify that the Bluetooth module is receiving data from the microcontroller. c. To test communication between the sensor subsystem and the switch subsystem, connect the microcontroller on the switch system PCB to a PC through the micro USB. d. Use the serial monitor to verify that data is being received from the sensor subsystem PCB. 4. Will use an oscilloscope to measure input voltage and the voltage at each of the components and make sure everything is 2.8 V - 3.3 V with a tolerance of ± 5%

True Data Verification and Demoing

We'll demo in one of the ECEB rooms like where we gave the presentations.

Beforehand, we will begin by testing in our own apartment. We will start testing in a controlled test environment with a wide range of cases to simulate normal entrances and exits of the room. Controlled velocities and distances between people entering the room will be observed and recorded.

Variables to test:

- Every mph from 1-6 to simulate slow walking up to a brisk jog for when apartment users run out the room. We will verify our speeds with a separate unit with a velocity sensor
- Having 1-5 people entering an apartment room from the same direction and alternating directions with distances of 1-3 feet between each of them

Once the camera passes the controlled testing, and as time allows, we will get a raspberry pi camera running OpenCV for people detection and set it to count the number of occupants and compare it with the sensor data under normal apartment usage in an uncontrolled testing environment.

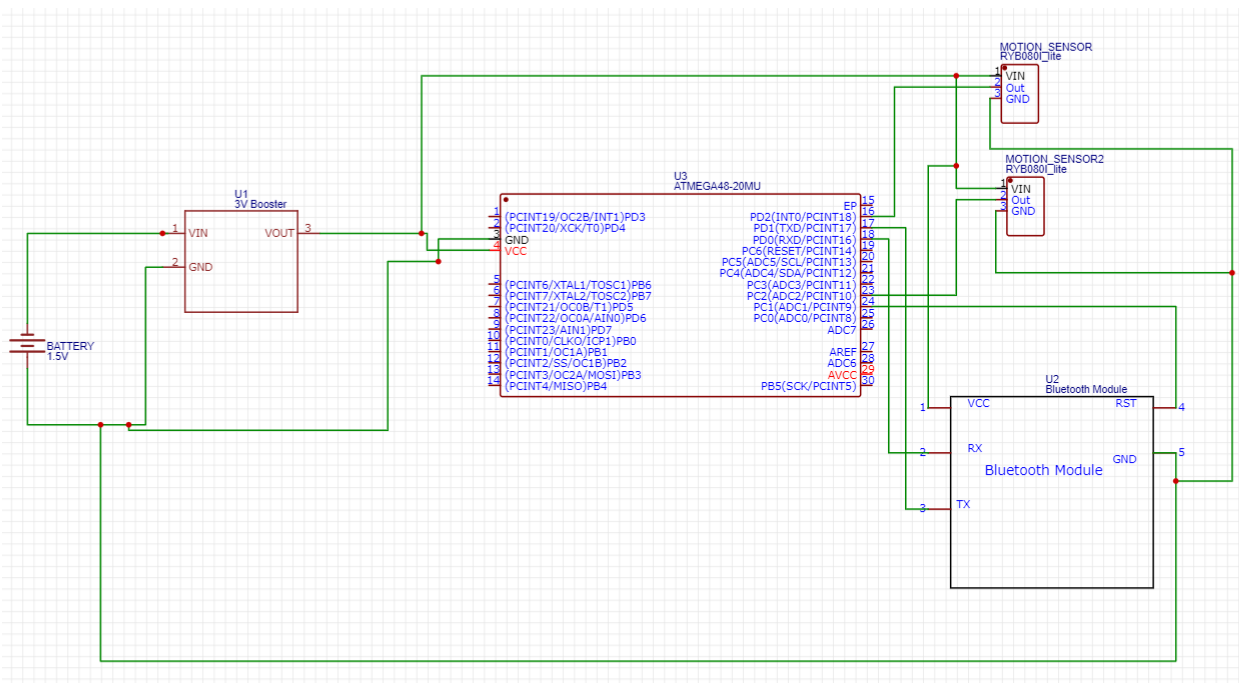

Figure 7: First Draft PCB of Sensor Subsystem

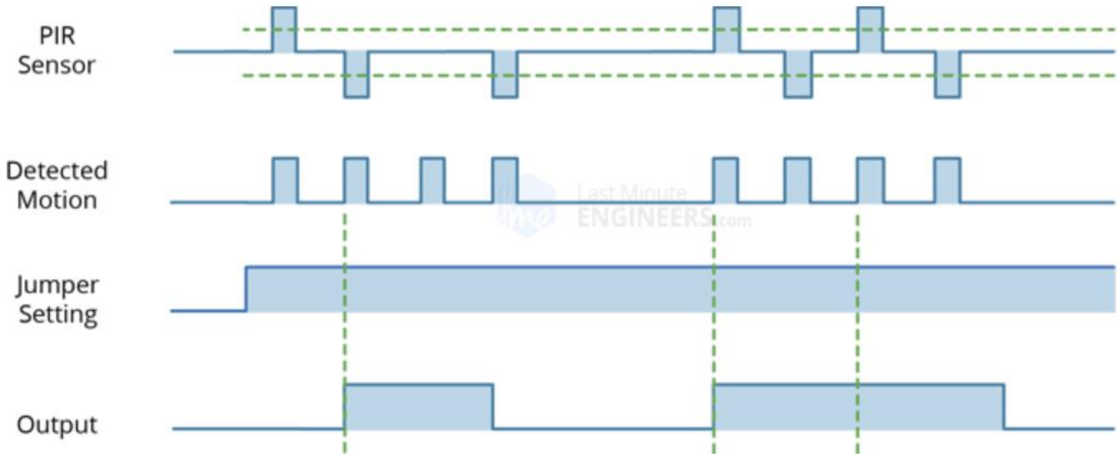

Figure 8: Timing Diagram of PIR Sensor [8]

2.4 Phone App Subsystem

The purpose of this subsystem is to allow apartment owners to control the switch units and turn off the lights in a room from anywhere in the apartment manually with their phone. The sensor unit will then communicate to all connected switch units to shut off the lights. The phone app will have an interface which provides an option to turn off and on the lights from the phone. In addition, the phone app will display information about your light usage in the apartment. For this subsystem, we chose to use a Bluetooth connection over a Wi-Fi module as one of our high-level requirements is to have our device run at least half a year on battery power, so apartment owners don't need to change batteries as often. Wi-Fi modules require a lot more power than BLE and would have greatly increased the frequency the batteries would need to be changed. Even though Wi-Fi is faster being able to transfer data at speeds up to 1.3 Gbps and has a much higher bandwidth at 11 Mbps vs 800 Kbps for Bluetooth [10], for our use this case these speeds are unnecessary as we are not transmitting that much data. Bluetooth is ideal for sending sensor data. In addition, for building the phone app itself, we will be using React Native.

Requirements	Verification

1. Phone needs to be able to connect with the Bluetooth module in the sensor unit.
 	
2. Send data to the MCU to control the states in the sensor
	1. Bluetooth Connection Verification:
	    1. Connect the microcontroller on the sensor subsystem PCB to the computer through mini USB.
	    2. From the computer, we will run a script in the Arduino IDE that will output the data being sent from the Bluetooth module. We will have an output when the Bluetooth is paired and connected
	    3. Connect the phone to Bluetooth and check the output data sent from the Bluetooth module

unit and turn off and on the lights	2. For testing the ability to send data to the MCU: 1. Verify this after completing the first verification for this subsystem to make sure there is a Bluetooth connection. 2. Run a script to program the MCU. The script will have the MCU hold a positive value count of occupants in the room that will constantly increment upwards. By hitting the button to turn off the lights in the phone app, the count of people in the room will be reset. When pressed again, it will restore the previous count. 3. Press the button in the phone app to reset the count the lights and observe the output.

2.5 Tolerance Analysis

We decided to use battery power for our units to improve modularity of our devices and make it easier to implement for the average apartment owner as well as not take up valuable outlet space. Keeping this in mind, low power consumption is a critical component to the success of our project. The reason being, if our power consumption is too high, the battery lifetime will be too short, and the frequency for which the batteries in our product would need to be changed would be very high. This is why we set a benchmark of having our customers need to change the batteries at max once every six months. We emphasized the low power aspect to reach this goal. In addition, we also decided to use 2 D batteries(E95BP-4), which have a capacity of 16500 mAh each, instead of 2 AA batteries(E91), which have a capacity of 3000 mAh each. [14]

Power Consumption of Sensor Unit:
ATmega in Active Mode = 1.5 [mAh] [16]
Bluetooth in Active Mode = 1.5 [mAh] [13]
MSP430-PIR Motion sensor in Active Mode (x2) = 220 [microAh] x 2 units = 440 [microAh] [15]

Total power consumption = 3.44 [mAh]
We are using 2 D batteries, so 16500 [mAh] * 2 = 33000 [mAh]
Therefore, 33000 [mAh] / 3.44 [mAh] =9593.023 [hours] = 399.709 [days] = 1.095 [years]

Power Consumption of Switch Unit:
Servo motor = 100 [mAh]. However, the servo will run at most 2 seconds per activation. 2 [seconds] is 0.000555556 [hour]. Therefore, the power consumed per activation is 100 [mA] * 0.000555556 [hour] = 0.0555556 [mAh per activation]
Let us say that the motor is used an average of 2 times an hour.
0.0555556 [mAh per activation] * 2 [activations] = 0.1111112 [mAh]
ATmega in Active Mode = 1.5 [mAh]
Bluetooth in Active Mode = 1.5 [mAh]

We do not find the power consumption per activation for the microcontroller and Bluetooth, because those parts are always on.

Total power consumption for that scenario is 3.11 [mAh]

We are using 2 D batteries, so 16500 [mAh] * 2 = 33000 [mAh]

Therefore, 33000 [mAh] / 3.11 [mAh] = 10610.932 [hours] = 442.122 [days] = 1.21 [years]

3. Cost and Schedule

3.1 Cost

Part	Units	Cost Per Unit	Total Cost
Part #: ATmega328p Manufacturer: Microchip Description: Microcontroller	2	$2.01	$4.02
Part #: RYB080I_lite Manufacturer: Reyax Description: Bluetooth	2	$10.30	$20.60
Part #: MSP430-PIR Manufacturer: Mouser Description: Motion Sensor	2	$13.79	$27.58
Part #: Product #169 Manufacturer: Adafruit Description: Servo Motor	1	$5.95	$5.95
Part #: DD0606SA-B Manufacturer: Eletechsup Description: Boost Converter to 3.3V	1	$9.99	$9.99
Part #: E95BP-4 (4 batteries/pack) Manufacturer: Energizer Description: D Batteries	1	$7.98	$7.98
Part #: LC2015 Manufacturer: Sci-Supply Description: D Battery Holders	1	$7.49	$7.49
Total Cost of Parts			$83.60

Total Cost of labor = $50 * 2.5 * 13 hours per week * 16 weeks * 3 people			**$78,000**
Grand Total Cost			**$78,083.60**

3.2 Schedule

Week	**Cary**	**Sam**	**Makombo**
2/24	Design Document	Design Document	Design Document
3/2	Work on BLE communications between sensor and switch units	CAD sensor and switch unit casings	Design PCB's
3/9	Prototype PIR sensor occupancy detection	Prototype Switch Unit	Verify PCB's and finalize first PCB order
3/16	Spring Break	Spring Break	Spring Break
3/23	Finish PIR sensor prototype and unit testing	Finish switch unit prototype and unit testing	Rework PCB
3/30	Build out application for communication between phone and sensor unit	Verify, test, and debug PIR and switch units	Solder components onto PCB
4/6	Combine project components and test and debug	Combine project components and test and debug	Combine project components and test and debug
4/13	Testing and Debugging	Testing and Debugging	Testing and Debugging
4/20	Mock Demo, Debugging	Mock Demo, Debugging	Mock Demo, Debugging
4/27	Demo & Presentation	Demo & Presentation	Demo & Presentation

4. Ethics and Safety

Regarding ethics and safety, we have identified a few potential ethics and safety risks. The first safety concern is the location of the PCB. Being an electrical device that will be present in rooms, there is a chance that the PCB could fry, and users could harm themselves. In order to protect against this, we must ensure that there are adequate measures in place against electrostatic discharge. For our design we plan to use two ground planes that we will stitch together with vias. Using two ground planes can help bypass ESD to ground and prevent it from passing through the active parts of the PCB. [10]

In addition, due to the multisystem aspect of our project, there are privacy considerations regarding the communication protocols between subsystems that we need to address in order to follow ACM's General Ethics Principles 1.6. For connection between the sensor and switch units, information is sent and received using Bluetooth. Steps should be taken to ensure that the Bluetooth unit cannot be eavesdropped on or be taken control of by a third party. Otherwise, they will have access to information on the comings and goings of people in their own homes. They would also be able to take control of the lights controlled by the switch unit. [3]

To secure our Bluetooth network, we can provide a pairing mode between the sensor units and utilize BLE's security mode 4. This is a service level security measure which will allow for authentication, encryption, and authorization of data sent between the devices. [4] In this way, third parties will be unable to connect with the sensor and switch units to receive information from them unless they have already previously paired with them. This will greatly mitigate the risk of unwanted connections as a user will need to be already in the apartment to activate the pairing mode of the devices.

For collaboration during the project, our group will adhere to IEEE Code of Ethics Rule 7: we will make sure throughout the entire design process that we will always seek honest feedback from one another. [5] In addition, being a project related to energy efficiency. Our project is following the ACM General Ethics Principle 1.1, which is to contribute to humanity and well-being, by helping the environment. [6]

References

[1] “How Much Money Can You Save by Turning Off the Lights?” Mr. Electric. [Online]. Available: https://mrelectric.com/blog/turn-off-lights-save-money. [Accessed: 14-Feb-2020].

[2] switch-bot.com. (2020). [online] Available at: https://www.switch-bot.com/bot [Accessed 24-Feb-2020].

[3] Mdpi.com. (2020). [online] Available at: https://www.mdpi.com/1999-5903/11/9/194/pdf [Accessed 14-Feb-2020].

[4] “Secure your Bluetooth wireless networks and protect your data,” TechRepublic. [Online]. Available: https://www.techrepublic.com/article/secure-your-bluetooth-wireless-networks-and-protect-your-data/ [Accessed: 24-Feb-2020].

[5] “IEEE Code of Ethics,” IEEE. [Online]. Available at: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 14-Feb-2020].

[6] “The Code affirms an obligation of computing professionals to use their skills for the benefit of society.,” Code of Ethics. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 14-Feb-2020]

[7] “How Does a Servo Work?” ServoCity.com. [Online]. Available: https://www.servocity.com/how-does-a-servo-work. [Accessed: 24-Feb-2020].

[8] Olimex, “MSP430-PIR,” Olimex. [Online]. Available: https://www.olimex.com/Products/MSP430/Starter/MSP430-PIR/. [Accessed: 24-Feb-2020].

[9] Admin, “How ESD Affects PCBs?,” Rush PCB, 10-Dec-2018. [Online]. Available: https://rushpcb.com/how-esd-affects-pcbs/. [Accessed: 28-Feb-2020].

[10] J. Laukkonen, “Bluetooth vs. Wi-Fi: Wireless Tech in Your Home, Office, and Car,” Lifewire, 24-Feb-2020. [Online]. Available: https://www.lifewire.com/bluetooth-vs-wi-fi-4088218. [Accessed: 28-Feb-2020].

[11] J. Ahmad, H. Larijani, R. Emmanuel, M. Mannion, and A. Javed, “Occupancy detection in non-residential buildings – A survey and novel privacy preserved occupancy monitoring solution,” Applied Computing and Informatics, 07-Dec-2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2210832718303284. [Accessed: 28-Feb-2020].

[12]. P. Raykov, Y. P. Raykov, E. Ozer, E. Ozer, G. Dasika, G. Dasika, A. Boukouvalas, M. A. L. NCRG, M. A. Little, Aston University, Aston University, Holdings, ARM Holdings, Holdings, Alexis Boukouvalas University of Manchester, University of Manchester, Ncrg, University of Kaiserslautern, Saarland University, Max Planck Institute for Informatics, and University of Washington, “Predicting room occupancy with a single passive infrared (PIR) sensor through behavior extraction,” Predicting room occupancy with a single passive infrared (PIR) sensor through behavior extraction | Proceedings of the

2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, 01-Sep-2016. [Online]. Available: https://dl.acm.org/doi/10.1145/2971648.2971746. [Accessed: 28-Feb-2020].

[13] TechDesign, "RYB080I_lite - REYAX eStore - Buy Direct," TechDesign. [Online]. Available: https://www.techdesign.com/market/reyax/product/RYB080I_lite?packing=Unit&gclid=EAIaIQobChMImuv6xJvr5wIVkobACh1mzgmpEAQYAyABEgKCTfD_BwE&fbclid=IwAR35oklScLO3caWibYXEPB2gwTwlluweQvDbjZl71jizV_vpN9yyBIfSdcc. [Accessed: 28-Feb-2020].

[14] "EN95 - Battery, Industrial, 1.5 V, D, Alkaline, 17 Ah, Raised Positive and Flat Negative, 34.2 mm," Newark. [Online]. Available: https://www.newark.com/energizer/en95/alkaline-zn-mno2-battery-1-5v/dp/48F915. [Accessed: 28-Feb-2020].

[15] Olimex, "MSP430-PIR," Olimex. [Online]. Available: https://www.olimex.com/Products/MSP430/Starter/MSP430-PIR/. [Accessed: 28-Feb-2020].

[16] "ATMEGA328P-PU Microchip Technology / Atmel: Mouser," Mouser Electronics. [Online]. Available: https://www.mouser.com/ProductDetail/Microchip-Technology-Atmel/ATMEGA328P-PU?qs=K8BHR703ZXguOQv3sKbWcg==&gclid=EAIaIQobChMIgMvh28Lz5wIVyMDACh1hYQQgEAAYASAAEgKYp_D_BwE. [Accessed: 28-Feb-2020].

[17] Olivia, "Battery-Powered Security Cameras: Battery Life & Battery Types," Reolink, 21-Jan-2019. [Online]. Available: https://reolink.com/battery-powered-security-cameras-battery-life-battery-types/?fbclid=IwAR3ogDTKZMXP4hGp86mZJ1tZwFkThCLnNcqakrOtu0bRiXvRGnDlnA7C5g8. [Accessed: 28-Feb-2020].