Real-Time Fire Escape Plan

Team 20 - Alex Makeever, Samir Kumar, Sujal Sutaria

ECE 445 Design Document - Spring 2020

TA - Johan Mufuta

Table of Contents

1 Introduction

1.1 Objective

Current building fire escape routes are rigid and do not adapt to quickly changing situations during a fire. Fire alarms will only tell you that there is a fire, but not the location of that fire. This presents a major problem because if a fire has engulfed a predetermined escape route, already frightened people can panic and be trapped in a building without the knowledge of where to go. For example, a hospital has many patient rooms and escape routes, and occupants who are often disoriented and unaware of the fire escape routes. When the fire alarm is triggered, mass confusion and panic can occur as patients frantically try to find an exit based on posted signs which convey no information regarding the status of the danger. According to the National Fire Protection Agency, there were an estimated 5,750 fires with an average of 2 civilian deaths in health care facilities between 2011-2015 [1]. Since the leading death toll in these fires is with ill residents [1] who are presumably unfamiliar with the building, our goal is to reduce the number of deaths by directing them out of the building automatically.

In addition, a minor flame can quickly turn into a major fire within 30 seconds [2]. When this happens, a raging fire can produce thick black smoke that fills nearby areas. This smoke can develop within just one minute [3], and inhalation of this smoke can cause permanent lung damage. Carbon Monoxide alone is dangerous at concentrations as low as 50 parts per million [4]. This problem is especially relevant to potentially panicked building occupants as they might run directly into a smoke filled room or hallway. Asphyxiation is the leading cause of fire deaths, exceeding burns by a three-to-one ratio, so knowing where to run in order to escape is extremely important.

Our goal is to create a system which will provide the safest escape route possible. This system will adapt to a fire based on temperature and smoke density from probes located throughout the building. This data will be used to generate a route which will guide building occupants through a safe and quick path to the nearest exit through an interconnected array of LED arrows on the floor. A phone application will also be developed to assist with escape and display a comprehensive heat map of the building.

1.2 Background

Currently, the focus of innovation has primarily been in residential homes, whether that is aiding the action of leaving the building, or installing smarter home fire alarms. There is little to no discussion on how to improve fire escape plans for commercial and industrial buildings.

Additionally, while some projects such as Google Nest [5] have created networks that provide more information regarding the location of danger, no solution on the market gives visual indication of the safest escape route. Although knowing only the location of a fire may be helpful for a small building where occupants are very familiar with the layout, such as a residence, having real-time indication arrows to aid in the escape for occupants unfamiliar with the layout of the burning building can potentially save thousands of lives.

The ideal application for our system is a large building with many exits, hallways, and rooms with an asymmetrical layout. It will need to be able to guide all occupants of a building to the nearest exit with minimal risk of encountering a fire. The application will also need to remove any user decision making as any wrong decision by the user would only delay escape.

1.3 High-Level Requirements List

- The location of unsafe areas should be updated using smoke and temperature data at a minimum frequency of once every 15 seconds.

- The system should display the optimal exit route on an LED display system based on thermocouple and smoke sensor data within 30 seconds of a fire alarm trigger event.

- The phone application should display a map of the building and show changing heat and smoke data in real time within 30 seconds of the fire alarm trigger event.

2 Design

The “Master Logging” block will be the main controller of the system. The microprocessor will be loaded with a maze solving algorithm that determines safe paths to exit the building. The maze in this scenario will be a blueprint of the building that marks all possible exits. These safe paths are updated in real time using feedback from the sensor modules. The “Sensor Module(s)” are comprised of a smoke and temperature sensor. They will be placed strategically around the building and will communicate air quality and heat data wirelessly using WiFi to a web server. Once the logging board has determined the optimal path, it sends the optimal path data via WiFi to store on the web server for utilization by the “Floor LED” block. This block will be LEDs placed on the floor at an intersection of hallways and at exits. This block fetches the direction data from the web server and lights up the corresponding LED indicating the correct direction to move. An example of this floor sign system at an intersection is seen in Figure 2.

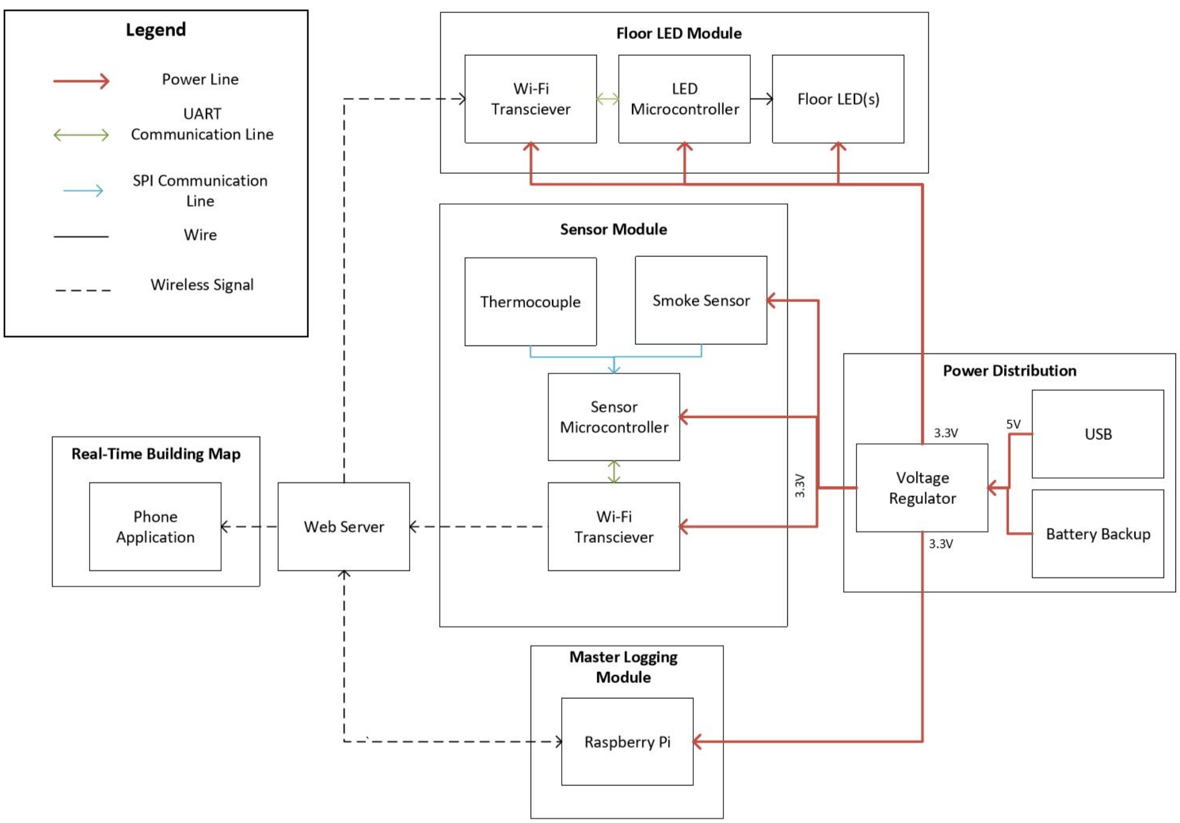

Figure 1. Block Diagram

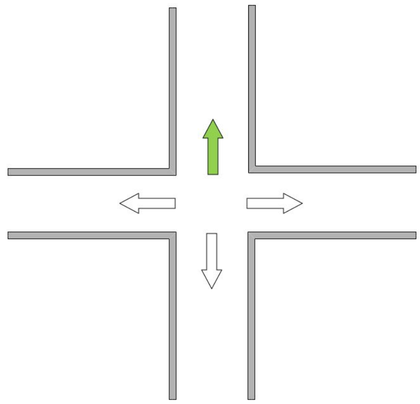

Figure 2. LED Arrow System

2.1 Power Distribution

The power distribution system is a sub-module which will be common to the Sensor and Floor LED modules. Its purpose will be to provide the necessary voltages to power each of the integrated circuits throughout the design.

2.1.1 Main Power

120 VAC wall power must be converted to 5 VDC to power the sensor and floor LED modules. To do this we will use a wall adapter that connects to our PCBs through a barrel jack connector.

2.1.2 Battery Backup

In order to ensure that the system is operational in the event that the fire causes an electrical outage in the building, each of the modules will contain a backup battery system. The batteries will be placed in a pack and connected directly to the power system of the PCB.

As shown in Figure 3, the output of the 5V power adapter will control a PMOS transistor. This transistor will turn on whenever the main power is lost and will conduct such that the battery is then connected to the system. Figure 4 shows the source switchover as the main supply loses power.

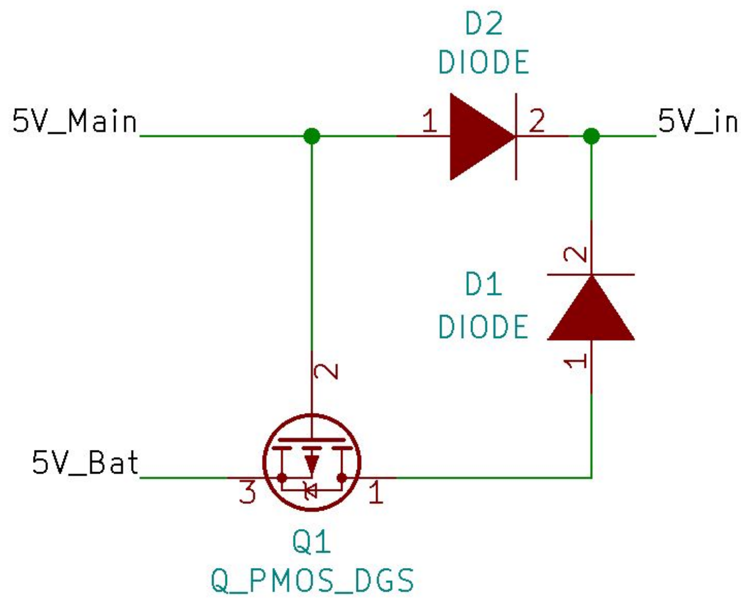

Figure 3. Battery Backup Circuit

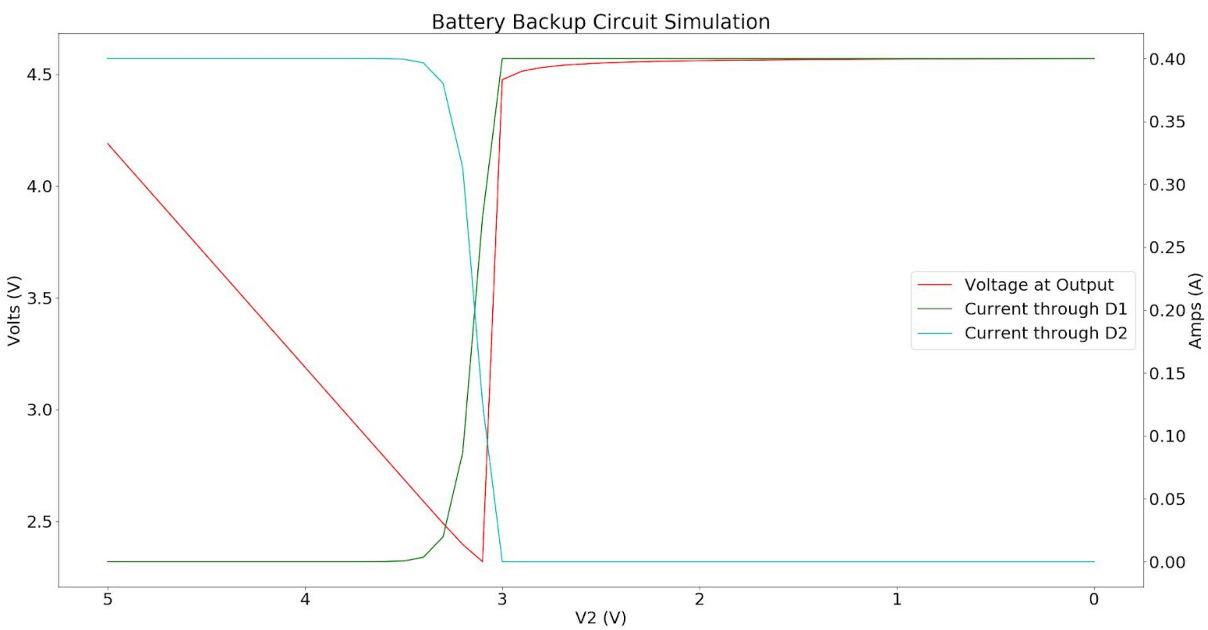

Figure 4. Battery Backup circuit simulation as main power (x-axis) is swept from 5V to 0V.

Requirement	Verification
1. The system outputs 300-500mA, at a minimum of 3.3V, for 1 continuous hour.	1. a. Disconnect the 5V power adapter and connect the

	battery. b. Connect a multimeter to the output of the battery backup system. c. Connect a DC electronic load to the output of the battery backup system. Set the load to constant current mode with a setpoint of 500mA. d. Begin a timer and wait for at least 1 hour to ensure that the output of the backup system is at least 3.3V for the entire duration.
1. Supplies at least 3.3V within 10μs of main power removal.	1. a. Connect the 5V power adapter and battery to the PCB. b. Connect an oscilloscope probe to the output of the 5V power adapter and the output of the battery backup circuit. c. Remove the 5V power adapter. d. Using the oscilloscope, measure the time between the removal of the 5V power adapter and the stabilization of the output voltage to +/- 5% of its nominal value.

2.1.3 Voltage Regulator

The voltage regulator will translate the 5V power provided by the battery/USB to the 3.3V required by many of the integrated circuits utilized on the PCBs. In some cases, an integrated circuit used on the board may have the option of using a 3.3V or 5V supply. In these cases, we will use the 3.3V power provided by the voltage regulator due to the reduced noise. The output of the regulator will be decoupled with multiple capacitors of mixed impedances to also improve noise immunity.

In order to protect the battery when it is powering the circuit, we will also utilize the enable pin on the voltage regulator. As shown in Figure 5, we will do so by making use of a voltage monitor IC which holds the enable pin high unless the voltage measured at the sense pin reaches the value of the internal voltage reference of 400mV. The sizing of the resistor divider to create our desired functionality is shown in Section 2.6.1.

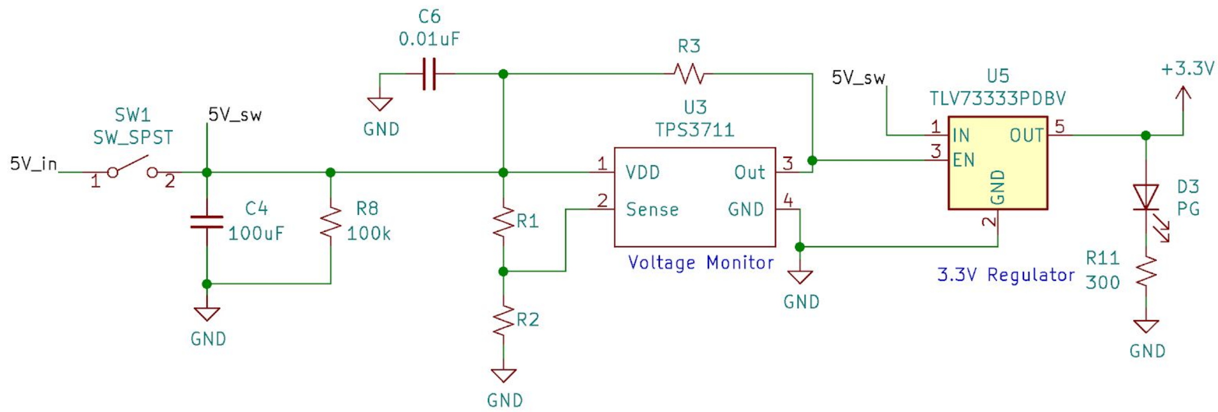

Figure 5. Voltage Regulator Schematic

Requirement	Verification
1. Step down a maximum of 5.4V input to a 3.3V output +/- 0.2V for up to 500mA of constant current load.	1. a. Connect a 5.4V power supply to the input of the regulator. b. Connect a DC electronic load to the output and set the load to constant current mode with a setpoint of 500mA. c. Use a multimeter to measure the output of the voltage regulator and ensure that the voltage is within 0.2V of 3.3V.
2. Disable power when the supply voltage reaches below 3.4-3.6V.	2. a. Connect a 5V variable power supply to the input of the regulator circuit, indicated by the +5V node in Figure 5. b. Connect a digital multimeter to the Vout pin on the circuit shown in Figure 5. c. Step down the 5V supply in increments of 0.1V until it reaches 3.5V. Ensure that the multimeter reads 0V at some point after the power supply goes below 3.4-3.6V.

2.2 Sensor Module

The Sensor Module allows the system to determine the location and intensity of a fire located in the building. It will take in the temperature and smoke density in the air and transmit the data to the web server to be processed by the master logging module.

2.2.1 Thermocouple

The thermocouple will be the method by which the system determines the ambient temperature. As the fire increases in intensity, it will be assumed that the ambient temperature will likewise increase significantly. This information will be used to generate the safest, shortest path to the exit. It should also be used to determine if a certain path can be traveled despite the presence of a fire. In certain cases, it may be possible for an escapee to travel through an area where a fire is present if the temperature and smoke content is deemed to be at a safe level.

We will use a K-Type thermocouple due to its wide temperature range and long lifespan at high temperatures. Before the thermocouple data is passed to the sensor microcontroller, it must first be converted from an analog to a digital value through a very precise Analog to Digital Converter (ADC) designed specifically for reading thermocouple data. This device will be the MAX31855K, as shown in Figure 9. The device will communicate with the sensor microcontroller over the SPI communication protocol.

Requirement	Verification
1. The thermocouple module must be able to report digital temperature values to the sensor microcontroller over SPI.	1. a. Set an oven to a temperature of 300°F and place the thermocouple junction inside the oven with the rest of the electronics outside. b. Program the microcontroller to select the thermocouple ADC. c. Connect an oscilloscope to the thermocouple MISO signal to display the thermocouple temperature. d. Use a well-calibrated thermometer placed alongside the thermocouple junction as a reference to determine accuracy of this module.

2.2.2 Smoke Sensor

The Smoke sensor will be the method by which the system determines the parts per million (ppm) concentration of carbon monoxide in the immediate area. A higher ppm detected is assumed to be a larger density of smoke in that area. The sensor is used to determine unsafe air quality conditions and combined with the thermocouple, will determine if the area is safe to travel through en route to an exit.

In order to interface with the smoke sensor, the device will require an ADC which can communicate with the microcontroller over SPI. Since the microcontroller only has one SPI peripheral, we will also need to use a decoder for the Chip Select signal so that only one of the slave devices (the Smoke Sensor ADC or the Thermocouple ADC) is controlling the SCLK and MISO logic buses, as seen in Figure 9.

In addition, since the smoke sensor we have chosen (MQ7) is powered off of 5V, we will require a 5V boost converter to provide appropriate power, as shown in Figure 6. The high switching frequency of the device requires a significant amount of decoupling capacitors and physical isolation in order to decrease noise addition to the circuit.

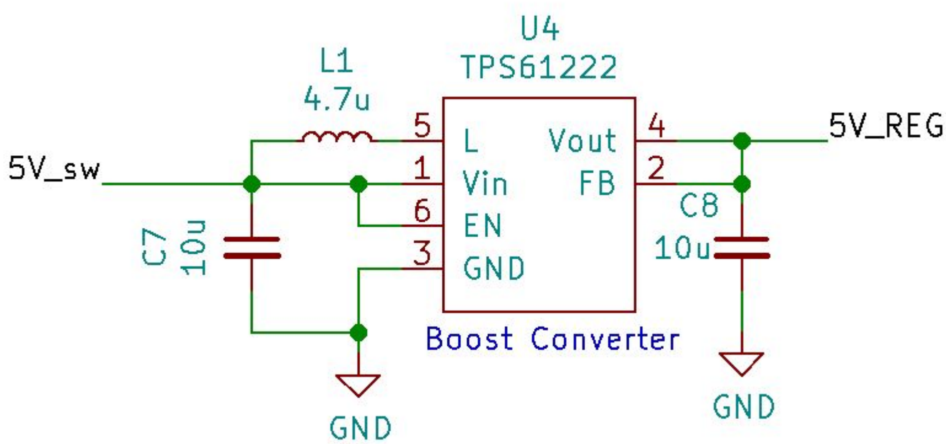

Figure 6. 5V Boost Conversion

Requirement	Verification
1. The smoke sensor module must be able to detect Carbon Monoxide and report the measured values in digital format to the sensor microcontroller.	1. a. Write a simple microcontroller program to toggle the values of arbitrary GPIO pins as the smoke density reaches 100PPM, 200PPM, and

	300PPM (These thresholds are picked arbitrarily in order to show the operation of the sensor; they may need to change based on testing conditions). b. Perform this test outside, somewhere away from flammable objects: set a small, controlled fire underneath the smoke sensor and use a digital multimeter to measure the voltage at the GPIO pins to verify that the sensor module is sending data to the microcontroller.

2.2.3 Sensor Microcontroller

The Microcontroller is the module that processes sensor data. It will take the digital sensor data and hand it off over UART to the Wi-Fi module to be sent to the web server. It will also act as the main controller for the Wi-Fi module, interfacing with the Wi-Fi module's RTOS to tell it when and where to send data.

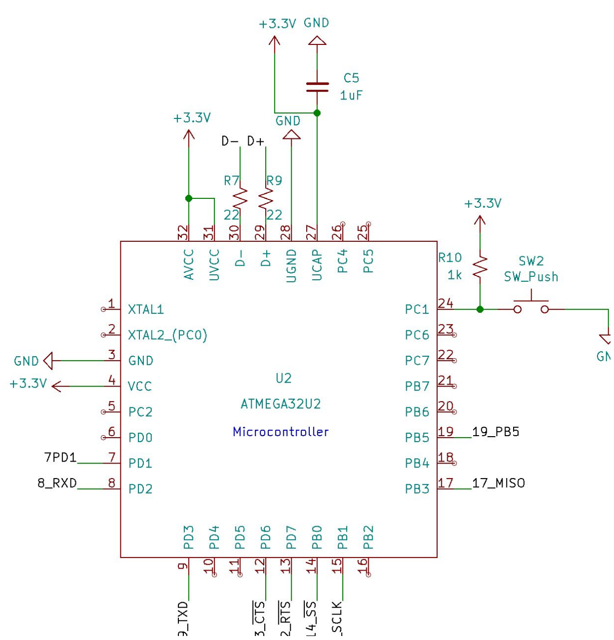

Figure 7. Sensor Microcontroller

Requirement	Verification
1. The microcontroller must be able to select which sensor it reads SPI data from.	1. a. Connect two oscilloscope probes, one to the chip select pin of the smoke sensor ADC and the other to the chip select pin of the thermocouple ADC. b. Program the microcontroller to drive the PB5 and SS control signals with different control values. i. e.g. 00, 01, 11, 10 c. Confirm that the correct ADCs are selected for each control pairing.
2. The microcontroller must be able to interface with the Wi-Fi module over UART with a baud rate of 74880 +/- 5%.	2. a. Send an arbitrary packet of data from the microcontroller to the Wi-Fi module over UART. b. Use an oscilloscope probe placed at the UART TX and RX pins to measure the time between each bit transfer.

2.2.4 Sensor Wi-Fi Transceiver

The purpose of the Wi-Fi transceiver within the sensor module is to send sensor data to the web server. This data will then be pulled from the server by the master logger module and by the phone application.

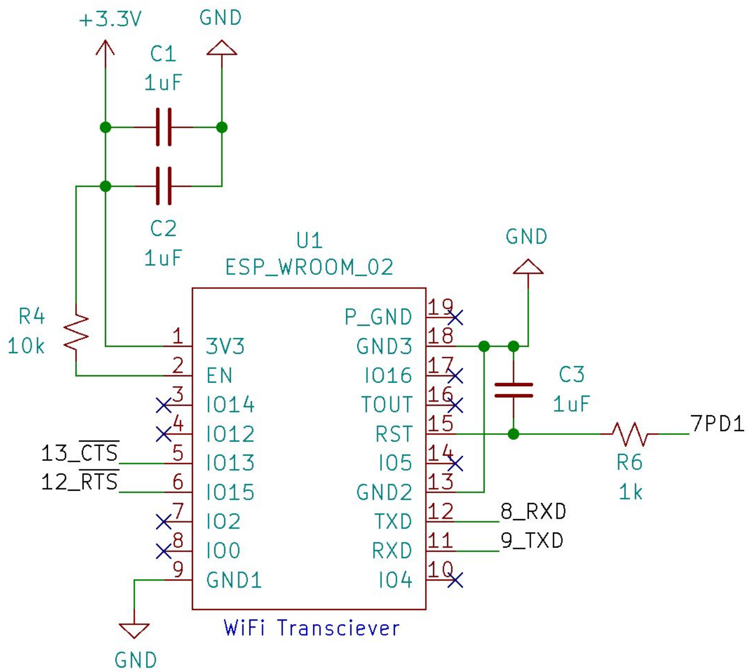

Figure 8. Wi-Fi Transceiver Schematic (Sensor Module and LED Module)

Requirement	Verification
1. The Wi-Fi module must be able to connect to the web server and send error-free data within 1s.	1. a. Using a serial connection between a PC and the Wi-Fi module, send the necessary “AT commands” to the Wi-Fi module to connect to the web server. Verify that the module responds with “OK”. b. Start a timer and send an arbitrary byte of data to the web server. Verify by logging into the web server that the chosen byte of data was received and saved by the server within the allotted time.

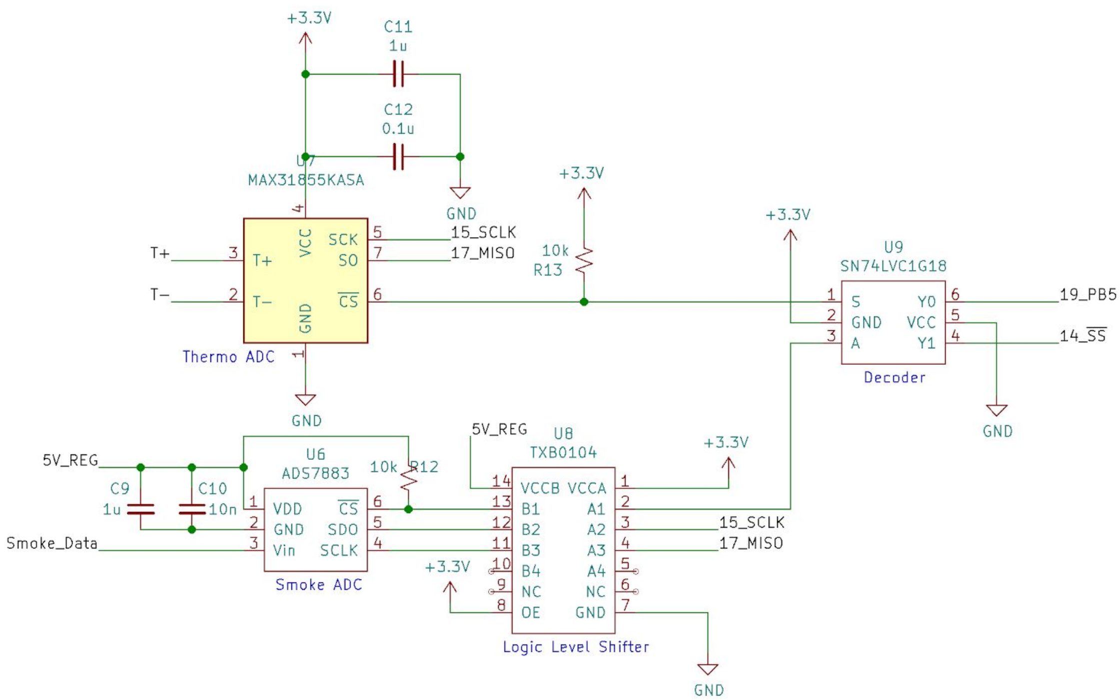

Figure 9. Sensor Interface Schematic

2.3 Master Logging Module

The Master Logging Module will act as the main information hub for the system. All of the sensor data will be fetched from the server to the logger. This data will then be processed in the context of the maze implementation of the building in order to calculate the locations of the floor LEDs which should be turned on. The collected data will also be sent to a server to be further processed by a phone application.

2.3.1 Raspberry Pi

The microprocessor is the device which will determine what parts of the building are most dangerous based on the sensor data and utilize a maze solving algorithm to find the safest and quickest path through a building. The map of the building as well as the locations of each of the smoke sensors and floor LEDs will be mapped out in the on-chip memory of the processor. This will allow programming of the chip to follow walls and avoid locations where sensors read a high temperature or smoke value. Once the path is determined, the directions will be sent to the Wi-Fi module to be sent to the LED modules.

This algorithm will be developed in Python, as the Raspberry Pi comes preloaded with Python. The algorithm will work as follows: the building will be mapped into a graph data structure. Every intersection, dead end, and room will be considered a vertex and every hallway and doorway will be considered an edge. These edges will be weighted, based on the length of the hallway in meters. The weighting of edges and vertices will change based on smoke and temperature readings. Every exit in the building will be mapped to a vertex, whether it be a doorway or a window, and the solution will be a list of edges from a vertex to an exit vertex. Every other non-exit vertex will be labeled as a start vertex, as every vertex in the graph requires a solution so no matter where the user is in the building, they will have a way to escape.

There will be no negative edge weights as distance and danger will always increase the edge weights. This graph will also be non infinite as no building is infinitely large. Therefore, we are able to utilize the A* search algorithm in order to solve the graph as A* is guaranteed to provide a solution for finite graphs with positive edge weights if using an admissible heuristic. Every time we run the algorithm, we will have to update the edge weights from the web server to account for the sensor data. On every start vertex in our maze, we will utilize A* in order to find the closest exit vertex in our building graph. Once the algorithm runs through every node in the building, a solution will exist for every vertex in the graph, which means an exit path will exist for every location in the building. The A* search algorithm will be using the Euclidean distance heuristic to determine which path is most ideal as we can label each vertex with a coordinate system. This heuristic will also be used to determine which vertex to path-solve on. The algorithm will have a list of all the vertices in the building and their distance to each exit. The vertex with the highest distance will be the next chosen vertex to solve for as this will progressively solve for more edges as the path is created, reducing the need to perform an A* for every node.

As each path will be non-cyclic, no vertices will have more than one exit destination. For example, if a user enters an intersection, they will not have to decide which path to use, as only one of the connected edges will lead to the exit in the quickest way possible. This exit edge will be used to show the user which direction to travel, as this edge will be selected to be displayed on the floor LED array. This data will be pushed onto the web server in order to be sent to the floor LED array. See Figure 14 for the flowchart of this algorithm.

Requirement	**Verification**
1. The maze solving algorithm is able to compute an efficient path through the building to find the shortest and safest path to the exit for each vertex within 10 seconds.	1. a. Generate a random finite maze that contains no negative weights. b. Manually create a list of every single path possible, from every start node to every exit node, with no repeating edges or vertices. c. Pair each path with its associated path weight d. Search the list to find the path

	with the lowest weight e. Start a timer in code f. Execute the A* algorithm on the predetermined graph g. End the timer and verify that it is under 10 seconds h. Compare this A* path with the lowest weight path from the list to validate that the lowest weight path has been generated by the A* algorithm

2.4 Floor LED Module

The Floor LED modules will be the main visual method by which the system directs escapees safely out of the building. At each intersection with more than one possible direction to turn, there will be a set of LEDs which will remain off unless the system determines that the safest escape route involves turning in their direction.

2.4.1 LED Microcontroller

The main role of the microcontroller in this module will be to parse the data fetched via WiFi from the web server. This data will contain the device addresses of the LEDs which should turn on or off. Since there will be multiple Floor LED modules, each of the microcontrollers will be programmed with the device addresses of the LEDs to which it connects. This way, when the microcontroller matches its device addresses to those which are sent by the logging module, it can turn on the correct LED(s). Once a new path has been determined by the master logging board, the microcontroller will send a done flag back to the web server to confirm that the change in path has been completed.

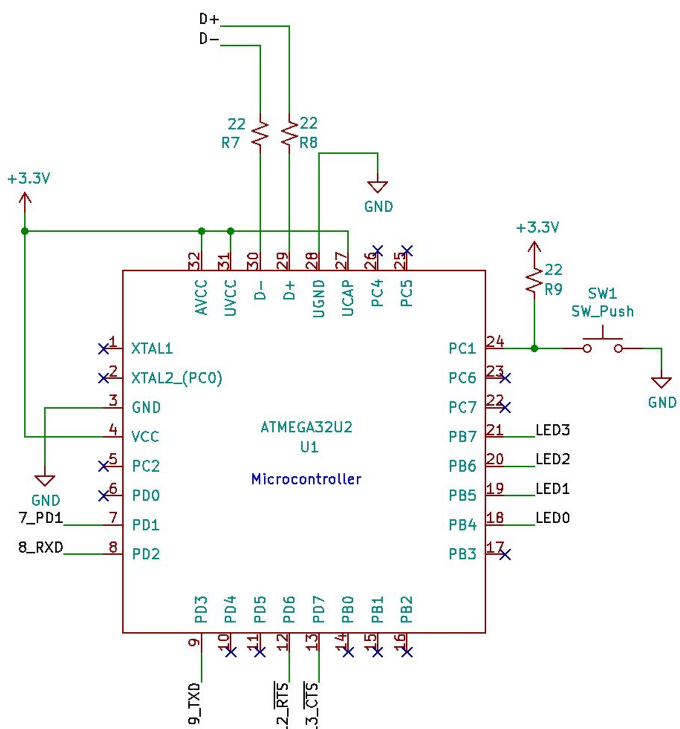

Figure 10. LED Microcontroller

Requirement	Verification
1. The microcontroller must have a working on-chip bootloader such that the code is saved on power-down.	1. a. Connect the 5V power adapter and disconnect the battery. b. Program the microcontroller using USB to repeatedly toggle the output of an arbitrary GPIO pin. c. Place an oscilloscope probe at the chosen GPIO pin. d. Reset the microcontroller power by either toggling the power switch or by disconnecting and reconnecting the 5V power adapter. e. Verify on the oscilloscope that the GPIO pin is still toggling as designated in the loaded program.
2. The microcontroller must be able to interface with the Wi-Fi module over UART with a baud rate of 74880 +/- 5%.	2. a. Send an arbitrary packet of data from the microcontroller to the Wi-Fi module over UART. b. Use an oscilloscope probe placed at the UART TX and RX pins to measure the time between each bit transfer.

2.4.2 LED Wi-Fi Transceiver

The purpose of the Wi-Fi transceiver in this module is to pull the direction data from the web server and pass it onto the microcontroller via UART.

Requirement	Verification
1. The Wi-Fi module must be able to connect and fetch data from the web server within 1s.	1. a. Using a serial connection between a PC and the Wi-Fi module, send the necessary “AT commands” to connect the Wi-Fi module to the web

	server. b. Put an arbitrary byte of data on the web server. c. Use a microcontroller to send the required commands to the Wi-Fi module over UART to pull the data from the server. d. Use a counter within the microcontroller code to determine the length of time it takes before the data is received. Light up an LED if this time is less than one second.

2.4.3 Floor LED

The floor LEDs will be arrays of LEDs located on the floor at each possible intersection with three or more connecting hallways throughout the building. The LEDs at each intersection will all be connected to a single microcontroller, but only the LEDs which follow the safest exit path will be turned on at a time. See Figure 2 for an example layout of an intersection.

Long wires connecting the LEDs to the microcontroller will result in a large resistive voltage drop. This makes powering these components from GPIO pins impractical due to the large amount of current required. Instead, we will power the LEDs with MOSFETs whose gates are controlled by the microcontroller pins. This way, the current will be provided by the power distribution system rather than the microcontroller.

Requirement	**Verification**
1. LEDs adapt to a changing escape route within 2 seconds of an updated route being calculated.	1. a. Place map data manually onto the web server. b. Wait for LED module to fetch data and update accordingly. c. Place new map data manually onto the web server. d. Time how long it takes for all LED modules to switch outputs.

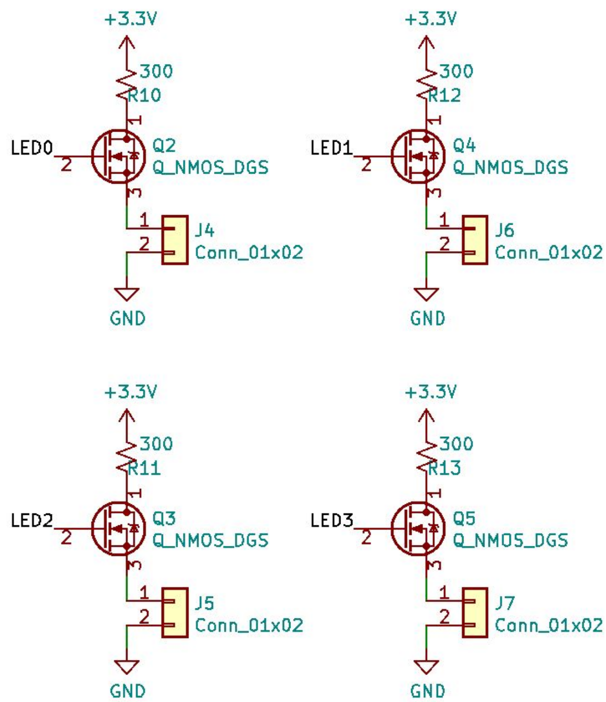

Figure 11. Floor LEDs

2.5 Real-Time Building Map Phone Application

This app will be developed on Android to show a floorplan of the building and all of the escape routes. The app would also show detailed heat and smoke data throughout the building. It will interface with the web server to receive heat and smoke data and output this on the map based on where the corresponding sensors are located on the map.

2.6 Web Server

We will use Google's Firebase service for our web server. The server will provide the handshake between the hardware modules and the master logger module through Wi-Fi. This will allow the sensor modules, LED modules, Raspberry Pi, and Android software to send and receive data asynchronously.

2.7 Calculations

2.7.1 Sensor Module Battery Capacity

Sensor Module			
Function	**Parts to Use**	**Voltage Input**	**Current Draw (Theoretical Max) (mA)**
Wifi and Antenna	ESP-WROOM-02	3.3V	170
Thermocouple	240-080	N/A	0
Thermocouple ADC	MAX31855KASA +T	3.3V	0.9
Smoke Sensor	MQ-7	5V	150
Smoke Sensor ADC	ADS7833	3.3V	3
Microcontroller	ATmega32U2	3.3V	6
Voltage Regulator	TLV733P	2.7V-5.4V	60
PG LED	445 Lab	2.6V	5
Boost Converter	TPS61222	3.5-4.5V	300
Logic Level Shifter	TXB0104	5V	0.01
Logic Decoder	SN74LVC1G18	3.3V	0.01
Voltage Monitor	TPS3711	3.3V	11
Total			705.92

From the table above, the maximum current consumption of the entire sensor module is 705.92mA. If we are to discharge the battery at a rate of roughly C/3 (C is the total current capacity of the battery) or lower, our current capacity for the sensor module will need to be:

$$Battery\ current\ capacity\ (mAh)\ =\ Total\ current\ draw\ (mA)\ *\ 3$$

$$Battery\ current\ capacity\ =\ 2117.76\ mAh$$

A standard Alkaline AA battery is rated for 2000mAh, so we will be able to use one to power the sensor module while discharging it at a rate slightly lower than C/3.

2.7.2 LED Module Battery Capacity

LED Module			
Function	**Parts to Use**	**Voltage Input**	**Current Draw (Theoretical Max) (mA)**
Wifi and Antenna	ESP-WROOM-02	3.3V	170
Microcontroller	ATmega32U2	3.3V	6
Voltage Regulator	TLV733P	2.7V-5.4V	60
PG LED	445 Lab Parts	2.6V	5
Floor LED (x4)	445 Lab Parts	3.3V	20
Voltage Monitor	TPS3711	3.3V	11
Total			272

From the table above, the maximum current consumption of the entire LED module is 272mA. If we are to discharge the battery at a rate of roughly C/7 (C is the total current capacity of the battery) or lower, our current capacity for the sensor module will need to be:

$$Battery\ current\ capacity\ (mAh)\ =\ Total\ current\ draw\ (mA)\ *\ 7$$

$$Battery\ current\ capacity\ =\ 1904\ mAh$$

A standard Alkaline AA battery is rated for 2000mAh, so we will be able to use one to power the sensor module while discharging it at a rate slightly higher than C/7.

2.7.3 Resistor Divider Sizing for Voltage Monitor Circuit

The minimum voltage that we will allow for the supply to reach, $V_{mon(UV)}$,will be

$$V_{mon(UV)}\ =\ V_{out}\ +\ V_{dropout}$$

Where $V_{dropout}$ is the dropout voltage of the linear regulator, which is equal to 0.2V, and V_{out} is the output voltage of the linear regulator, which is equal to 3.3V. Therefore,

$$V_{mon(UV)} = 3.5V \ .$$

From the TPS3711 datasheet, we know that

$$V_{mon(UV)} = (1 + \frac{R1}{R2}) * V_{IT-}$$

Where V_{IT-} is the value of the internal voltage reference, equal to 0.4V. So,

$$\frac{R1}{R2} = \frac{V_{mon(UV)}}{V_{IT-}} - 1$$

Therefore,

$$\frac{R1}{R2} = 7.75 \ .$$

In order to save power and not sink battery current in the case of the circuit being off, we'll want both of these resistors to have a very high value. We can use:

$$R_1 = 1M\Omega = 7.75 * R_2$$

$$R_2 = 129.03k\Omega.$$

The power consumption when the output of the voltage monitor is low will therefore be no larger than:

$$P = \frac{V^2}{R_1+R_2}$$

$$P = 10.9mW \ .$$

2.7.4 Allowable Baud Rate Error for UART Communication

The WiFi module and microcontroller both communicate via UART. However their communication speeds are different. The datasheet shows the maximum receiver baud rate error that can be tolerated with the following calculation:

$$R_{slow} = \frac{(D+1)*S}{(S-1)+(D*S)+S_F}$$

Where:

- R_{slow} : The ratio of the slowest incoming data rate that can be accepted in relation to the receiver baud rate
- $D = 8$: The sum of character size and parity size
- $S = 16$: Samples per bit
- $S_F = 8$: First sample number used for majority voting

$$R_{fast} = \frac{(8+1)16}{(16-1)+(8*16)+8}$$

$$R_{slow} = 0.9536 = 95.36\% \textit{ of the receiver baud rate}$$

- WiFi (Receiver) baud rate = 74880 bps
- Microcontroller baud rate = 76800 bps

$$R_{fast} = \frac{74880}{76800} = 0.975 = 97.5\% \textit{ of the receiver baud rate}$$

This ratio is within the tolerance of acceptable baud rate error

2.7.5 Algorithm Runtime

To calculate the runtime of this algorithm, we will first begin with the runtime of the A* search algorithm. The worst case runtime of A* is the same as the runtime of the Breadth First Search Algorithm, which is

$$O(V + E)$$

where V is the number of vertices and E is the number of edges. This would mean that A* would also have the same worst case runtime of $O(V + E)$. Our algorithm, however, requires A* to run multiple times a cycle as we need to perform it on every vertex in the graph. If A* runs V number of times, the worst case runtime for our algorithm would be

$$O(V^2 + VE)$$

2.8 Flowcharts

2.8.1 Sensor Microcontroller Flowchart

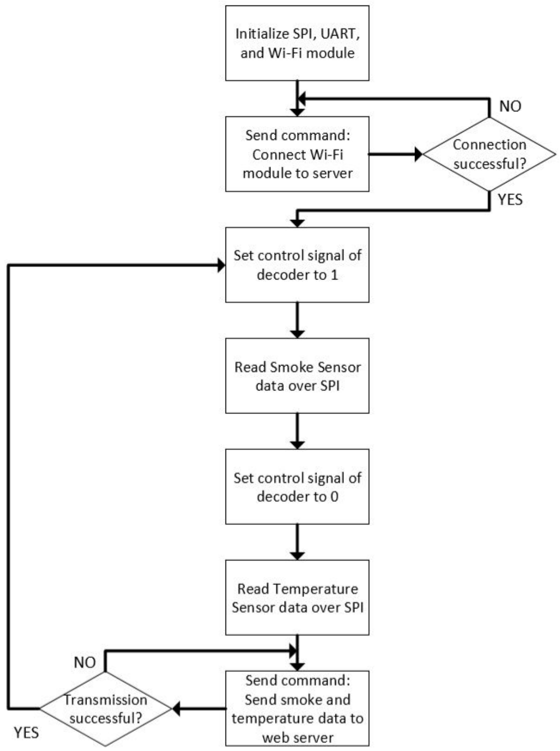

Figure 12. Sensor Microcontroller Code Flowchart

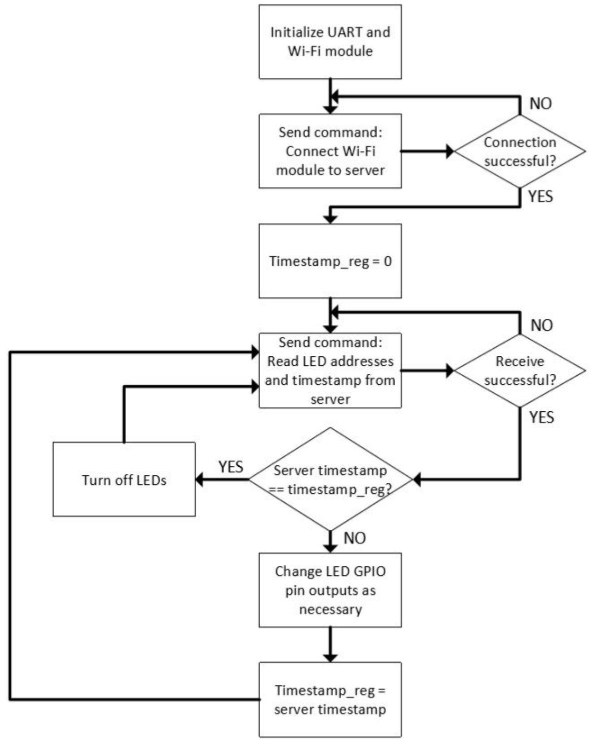

Figure 13. LED Microcontroller Code Flowchart

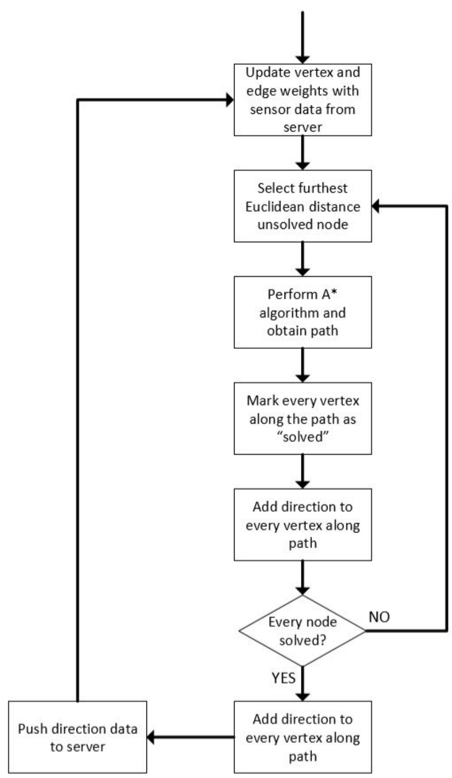

Figure 14. Maze Solving Algorithm

2.9 Tolerance Analysis

The largest reliability issue is the placement of the sensor modules throughout the building. If the sensor modules are placed too far apart, it is possible that they may not pick up on the change in temperature. In order to determine the furthest distance each sensor must be placed from one another in order to detect a change in temperature due to fire, we can use the equation for heat transfer by convection:

$$\dot{q} = h(T_{hot} - T_{cold})A$$

Where $\dot{q}$ is the heat transfer per unit time (W), h is the convective heat transfer coefficient of the material (W/(m^2*K)), T is the temperature (K), and A is the area of exposure (m^2). In our case, we will be measuring the temperature of the air, so we will need to find the convective heat transfer coefficient of air. We will use a worst-case value of

$$h = 10 \frac{W}{m^2 * K}$$

since stagnant air in a building is likely flowing slowly due to density changes as the air heats up [8]. We will also use a worst-case value for the heat transfer of a building fire of 2.9MW [9]. Lastly, the temperature difference will be between that of a burning building fire, determined to be about 600°F [4], and room temperature, 70°F. Therefore, the maximum permissible area between the fire and a single heat module can be calculated as:

$$A = \frac{\dot{q}}{h * (T_{hot} - T_{cold})}$$

$$A = 1000\, m^2 .$$

This tells us that in order to successfully detect a fire anywhere in the building, the system will need to have a temperature sensor density of:

$$A^{-1} = 0.032 \frac{sensors}{m^2} .$$

This value is also based on the assumption that there are no obstructions, such as walls, to the convective heat transfer. In reality, there would be many walls throughout a building. As such, we would likely place at least one sensor in every room. However, the above analysis proves that we can use a single sensor for a hallway as long as:

$$L = \frac{\sqrt{A}}{2} = 16m .$$

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Labor

Name	Hourly Rate	Hours	Total	Total * 2.5
Alex	$40	200	$8,000	$20,000
Samir	$40	200	$8,000	$20,000
Sujal	$40	200	$8,000	$20,000
Total				$60,000

3.1.2 Parts

Sensor Module					
Function	Parts to Use	Vendor	Individual Cost	Quantity	Total Cost
Wifi and Antenna	ESP-WROOM-02	Digikey	$2.70	1	$2.70
Thermocouple	240-080	Digikey	$9.99	1	$9.99
Thermocouple ADC	MAX31855KASA +T	Digikey	$5.47	1	$5.47
Smoke Sensor	MQ-7	Sparkfun	$7.25	1	$7.25
Smoke Sensor ADC	ADS7833	Mouser	$5.10	1	$5.10
Microcontroller	ATmega32U2	Digikey	$2.98	1	$2.98
Voltage Regulator	TLV733P	Digikey	$0.37	1	$0.37
Boost Converter	TPS61222	Digikey	$1.14	1	$1.14
Logic Level Shifter	TXB0104	Digikey	$1.17	1	$1.17

Logic Decoder	SN74LVC1G18	Mouser	$0.37	1	$0.37
Voltage Monitor	TPS3711	Mouser	$2.11	1	$2.11
Passives (resistors, capacitors, etc.)	N/A	Digikey	$0.10	26	$2.60
Total Per Board					$41.25
Number of Boards				3	$123.75

LED Module

Function	Parts to Use	Vendor	Individual Cost	Quantity	Total Cost
Wifi and Antenna	ESP-WROOM-02	Digikey	$2.70	1	$2.70
Microcontroller	ATmega32U2	Digikey	$2.98	1	$2.98
Voltage Regulator	TLV733P	Digikey	$0.37	1	$0.37
Logic Level Shifter	TXB0104	Digikey	$1.17	1	$1.17
Logic Decoder	SN74LVC1G18	Mouser	$0.37	1	$0.37
Voltage Monitor	TPS3711	Mouser	$2.11	1	$2.11
Passives (resistors, capacitors, etc.)	N/A	Digikey	$0.10	24	$2.40
Total Per Board					$12.10
Number of Boards				4	$48.40

Master Logging Module

Function	Parts to Use	Vendor	Individual Cost	Quantity	Total Cost
Master Logger	Raspberry Pi	Amazon	$42.00	1	$42.00
Total Per Board					$42.00
Number of Boards				1	$42.00

$$Estimated\ Total\ Cost\ =\ Parts\ Cost\ +\ Labor\ Cost$$

$$Parts\ Cost\ =\ \$123.75\ +\ \$48.40\ +\ \$42.00$$

$$Estimated\ Total\ Cost\ =\ \$60,214.15$$

3.2 Schedule

Week	Item	Responsibility
3/2/2019	Finalize initial LED Module schematic, buy parts	Alex
	Finalize initial Sensor Module schematic, buy parts	Samir
	Research SD card programming of Raspberry Pi for non-volatile storage	Sujal
	Purchase Raspberry Pi	All
3/9/2019	Breadboard and test LED Module circuit using Arduino as microcontroller	Alex
	Breadboard and test Sensor Module circuit using Arduino as microcontroller	Samir
	Finalize initial LED Module PCB design	Alex
	Finalize initial Sensor Module PCB design	Samir
	Begin writing maze solver code to Raspberry Pi	Sujal
3/23/2019	Populate PCBs	Alex
	Finalize LED Microcontroller code	Alex
	Finalize Sensor Microcontroller code	Samir
	Set up web server	Sujal
	Finalize Raspberry Pi code and ensure device connectivity with web server	Sujal
3/30/2019	Test and debug issues with initial LED Module	Alex
	Test and debug issues with initial Sensor Module	Samir

	Finalize LED Module PCB for last revision	Alex
	Finalize Sensor Module PCB for last revision	Samir
	Begin writing phone application	Sujal
4/6/2019	Continue Debugging	All
4/13/2019	Populate PCBs	Alex
	Start Full Integration Testing	All
4/20/2019	End to End Testing	All
4/27/2019	Final Demonstration, last minute tweaks	All
5/4/2019	Final Presentation	All

4 Safety and Ethics

4.1 Safety

Our users do not interact with our system in the usual way. The system only directs people to a safe exit, so our biggest safety issue is with the floor LED system. It is possible for building occupants to misinterpret the signs, leading to people not leaving the building in an efficient and safe manner. In order to avoid misinterpretation, we have tried to simplify our sign system as much as possible. As seen in Figure 2, the system will be arrows placed on the ground, and the arrow that lights up shows the correct direction to move.

In a similar light, if the system malfunctions or fails, it may end up directing people into an unsafe area. Our solution to this problem revolves around the floor LED subsystem to not latch a value sent by the master logging subsystem. Once the master logger sends a packet of information, the floor LED PCB will light up the corresponding arrow for a short duration, and then turn off all arrows. It will then wait for the next set of instructions from the master to relight an arrow. This way if the logger fails, there will be no arrows lit up to potentially misdirect people.

Another safety issue that could occur is in the case that no exit has a safe escape path. If all exit paths are compromised, the escapee may still be trapped in a room or hallway without our system having any ability to help them. Ideally, the path-finding algorithm would contain backup escape methods such as climbing out a window or onto the roof. However, neither of these cases are ideal and could still lead to injuries. While additional functionality of the phone application is not currently in our work scope, the phone application can provide room for additional functionality to solve these problems in the future such as showing the fire department the location of the trapped person.

Batteries also have inherent safety issues, especially Lithium-based batteries [7]. We use a battery for the backup power for each of our modules, however, we elected to use Alkaline batteries due to their much lower fire risk. Even so, we designed a small circuit within the power distribution module which will sink current from the batteries only in the condition that the main power is lost and the total battery voltage is at a safe level (>3.5V). Additionally, we placed a diode at the output of the battery so that no current is sunk from the circuit into the battery.

4.2 Ethics

It is clear that this project deals very closely with the safety of the public, which is covered in the IEEE Code of Ethics #1: “to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment.” In order to comply with this code, we will test our system very thoroughly in a variety of situations and environments before releasing it to the market. We will also be sure to provide ample documentation and warnings to ensure its proper use. If ever we discover a flaw in our design, we will ensure that this is known to the users of the system before we work tirelessly to fix it.

In addition, our project relies on the use of sensor data to provide information regarding an emergency. This can possibly infringe on IEEE Code of Ethics #3: “To be honest and realistic in stating claims or estimates based on available data.” We must be very careful in our calibration of this system so that when it is in operation, the sensors do not use the available data to lead users astray and into unsafe situations. In order to make sure we abide by this code, we have researched unsafe levels of Carbon Monoxide and heat and we will work towards designing our sensor module such that it follows these limits as closely as possible. Throughout the project, we will continue to test and adjust our design to ensure that all data is used appropriately.

5 References

[1] National Fire Protection Agency, ‘Structure Fires in Health Care Facilities’, 2017. [Online]. Available:
https://www.nfpa.org//-/media/Files/News-and-Research/Fire-statistics-and-reports/Building-and-life-safety/oshealthcarefacilities.pdf

[2] Ready.gov, ‘Home Fires’, 2019. [Online]. Available: https://www.ready.gov/home-fires

[3] National Research Council Canada, ‘Toxic Gases and Vapors Produced at Fires’, 1971. [Online]. Available: http://web.mit.edu/parmstr/Public/NRCan/CanBldgDigests/cbd144_e.html

[4] Puroclean, ‘How Does a Fire Spread in a Building?’, 2016. [Online]. Available: https://www.puroclean.ca/blog/how-does-fire-spread-building/

[5] Google, ‘Google Nest Protect’, 2020. [Online]. Available: https://store.google.com/us/product/nest_protect_2nd_gen

[6] Archibald Tewarson, ‘Smoke Emissions in Fires’, 2008. [Online]. Available: https://iafss.org/publications/fss/9/1153/view/fss_9-1153.pdf

[7] Arrow Electronics, ‘Basic Battery Safety: Very Different Chemistries, Very Different Concerns’, 2015. [Online]. Available:
https://www.arrow.com/en/research-and-events/articles/basic-battery-safety-very-different-chemistries-very-different-concerns

[8] Fishlock, Mark, ‘Physics of Fire - Fire Energy Release Rates’. [Online]. Available: http://www.highrisefirefighting.co.uk/docs/physicsO2.pdf

[9] American Institute of Aeronautics and Astronautics, ‘Heat Release Rates of Burning Items in Fires’, 2000 [Online] Available:
http://www.firebid.umd.edu/docs/Kim%20-%20HRR%20of%20Burning%20Items%20in%20Fires.pdf

6 Appendix A

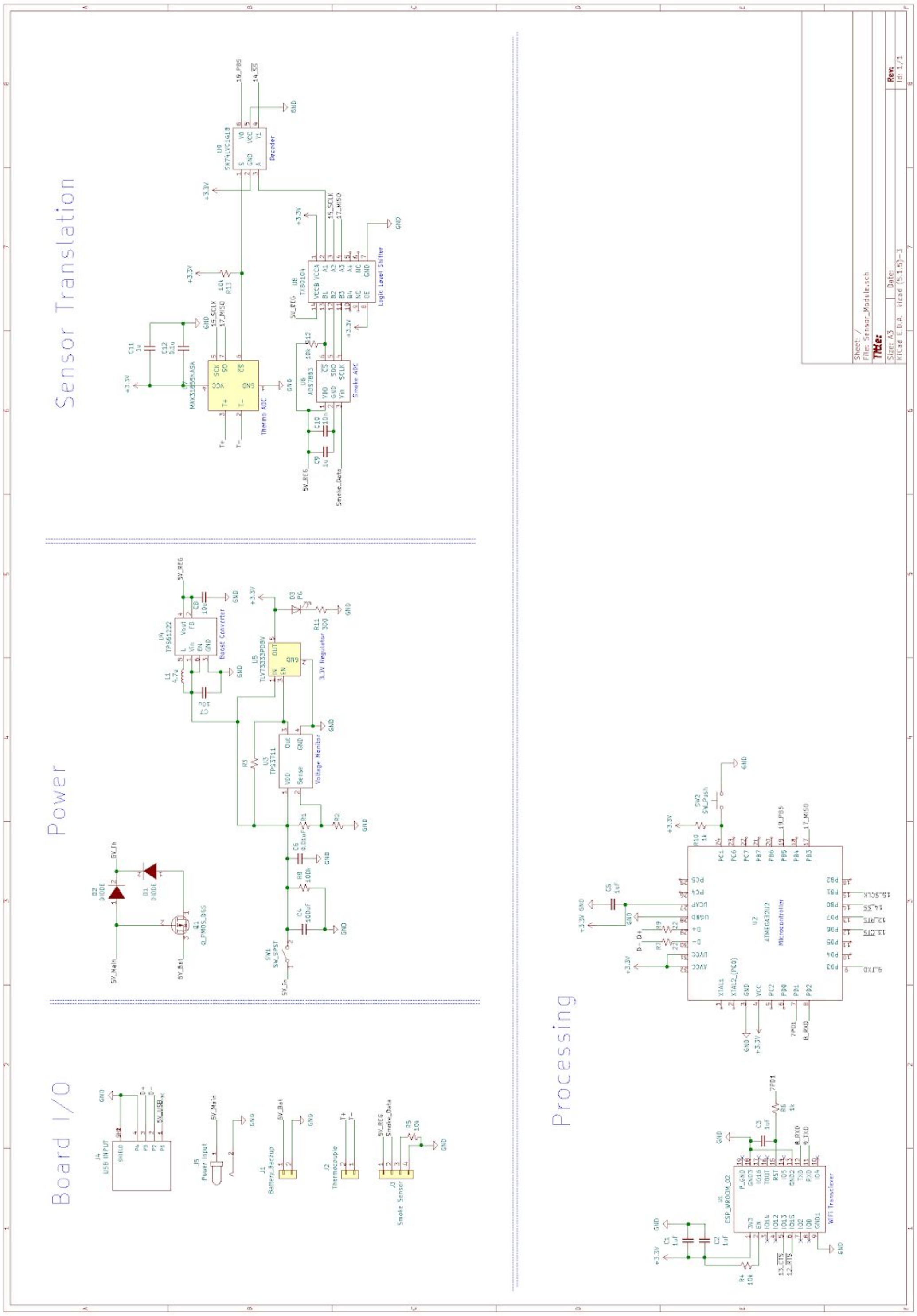

Figure 15. Sensor Module Full Schematic

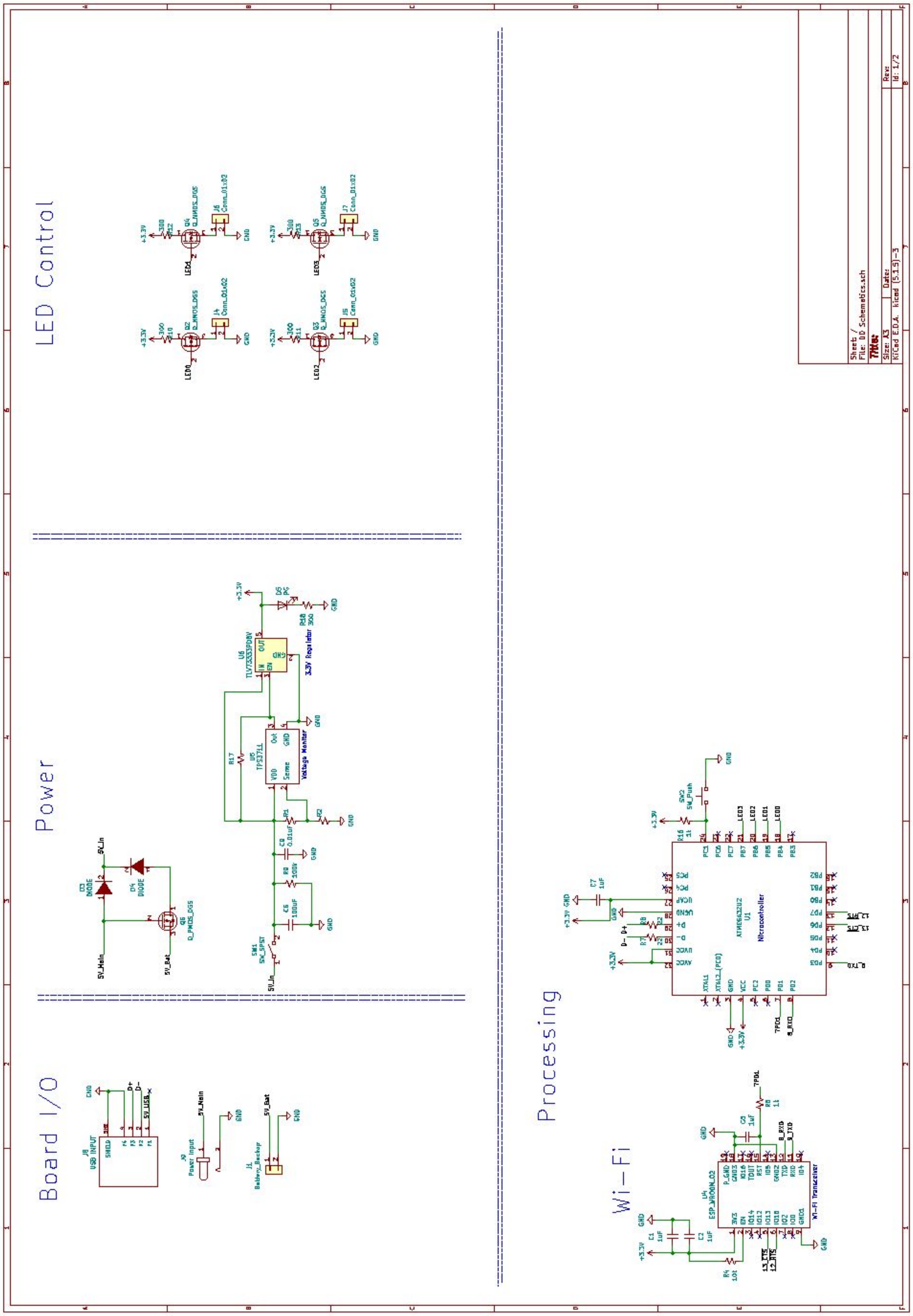

Figure 16. LED Module Full Schematic