SMART FIRE DETECTOR

By

Jens Foyer

Kok Jian Xiong James

Final Report for ECE 445, Senior Design, Spring 2016

TA: Benjamin Eng

4 May 2016

Project No. 34

Abstract

The device my team created is used to detect unsafe breathing conditions for humans and high object temperatures in front of the device. This is accomplished through 3 sensors, a carbon monoxide (CO) sensor, a photoelectric sensor, and a thermopile array. An alarm system and display will notify people in the area when action should be taken to leave the area or provide ventilation. Users can interact with the system through 3 buttons. We found that the best use of our device is in a kitchen or other cooking areas where there is the largest risk of fire.

Contents

1 Introduction

1.1 Purpose

The objective of this project was to design a fire detection system that would provide homeowners with a reliable and quick way to detect a fire, so that proper emergency protocol can be followed and potentially increasing survivability in an emergency. From [11], it has been estimated that there are around 445 cooking fires per day, with unattended equipment being the leading cause. While there are many products that tout smart networking capabilities and combined sensors, these products lack the quick response of an infrared thermal sensor to detect heat emanating from a fire or heat source. Our system aims to address this issue by being a more effective and faster way to detect fire.

1.2 Function & Features

To encompass the various threats that homeowners would face in their homes, we decided to use three sensors packaged in simple integrated alarm as a prototype. The sensor array consists of an infrared thermopile array, a carbon monoxide sensor and a photoelectric smoke detector. To make it highly adaptable to different changes in technology, we decided to use a programmable chip to allow users to update the alarm system with newer algorithms. In addition, the system design is modular, allowing for rapid prototyping, testing and replacements. With a user interface consisting of a liquid crystal display (LCD) and input buttons, users can monitor the data being collected by the alarm.

1.3 Block Diagram

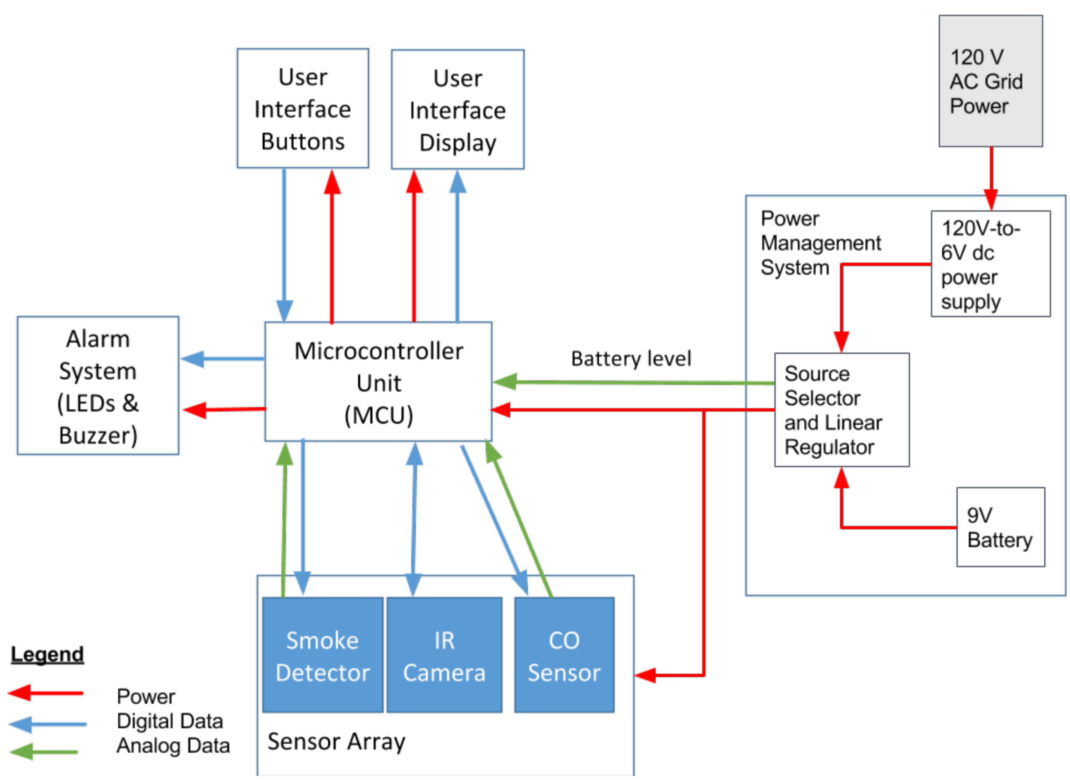

Figure 1: Block Diagram of Fire Alarm

1.4 Block Descriptions

1.4.1 Microcontroller Unit (MCU)

We used an ATMEGA328P for the microcontroller. It is responsible for communication between the different modules, processing data from user inputs and sensor data and to determine whether there is a safety issue. This chip is chosen as it has a very high reliability and sufficient amount of inputs, outputs, and memory to manage the sensors.

Input	5V Power Supply, I2C Serial Interface, 4 Analog ADC
Output	Digital Output, PWM Digital Outputs, LCD Display Data, Sensor Mode Select Lines

1.4.2 Power Management System

The power management system is responsible for supplying 5 volts direct current (DC) to the sensor array, MCU, alarm system, and user interface. The main power for the device comes from a standard US wall outlet delivering 120 volts alternating current. A 6-volt adapter lowers the voltage of the wall outlet and converts to DC current. In the interest of safety, a 9-volt battery is used for emergency power. Fires can cause electrical failures and power loss from the outlets. In the event of losing power from the wall outlet, the power source instantly switches to the battery.

Input	9V/2A AC-DC Power Supply, 9V Battery
Output	5V/1A DC Regulated Supply, Battery Voltage Readout

1.4.3 User Interface System

The user interface consists of an LCD and three push-buttons. The buttons on the device are Reset, Silence, and Menu, which allow for user feedback. The LCD shows the current maximum temperature observed, carbon monoxide concentration, photoelectric sensor voltage reading, and battery voltage level.

Input	5V DC, Pushbuttons, Digital I/O, PWM Digital I/O
Output	LCD readout, RGB LED

1.4.4 Alarm System

The alarm system functions as a visual and auditory feedback, which grabs the attention of people nearby. The buzzer and two red ultra-bright LEDs let the user know when they need to evacuate the area, provide ventilation, or change the battery. The alarm system is enabled by the MCU when there is an active alarm.

Input	5V DC, PWM Digital I/O
Output	LED Output, Alarm Sound

1.4.5 Smoke Detector

The smoke detector uses an IR LED and photodiode pair in a smoke chamber to detect scattering caused by smoke particles. The photocurrent generated by the photodiode is amplified using an op-amp amplifier circuit to generate the output.

Input	5V/100mA DC, Digital Select (LED Power)
Output	Analog Voltage (0-5V)

1.4.6 Infrared Thermopile Array

The thermopile array uses long-wave infrared to determine the temperature of an object at a distance. It communicates with the MCU using I2C, provides the temperature of the device, and a 16x4 pixels temperature image at a 50^{o} field-of-view (FOV).

Input	5V/7mA DC (down-regulated to 2.6V)
Input/output	I^2C Communications

1.4.7 Carbon Monoxide Sensor

The carbon monoxide sensor uses a tin oxide (SnO2) semiconductor to measure the concentration of CO gas by measuring the resistance of the semiconductor. As CO is absorbed by the semiconductor, the resistance changes.

Input	5V/140mA DC, MCU Digital Select (Power Mode Select 1.4V/5V)
Output	Analog Voltage (0-5V)

2 Design

2.1 General Design Alternatives

While designing the project, we came out with small variations on the sensor array and power management system. For example, we were deciding between the MLX90621 thermopile array and the FLIR Lepton Camera, but decided with the thermopile array for its cheaper cost and computational limits. Cost was the main driving factor in our design, the intention was to make it affordable compared to commercially available smart smoke detectors.

2.2 Main Board Design

The main printed circuit board (PCB) holds all components of the power supply and alarm system. It also contains the connections to the sensor array, LCD, and MCU. Using headers allows for modular testing and customization of the sensor placement. The schematic of the main PCB is shown in Figure 2. The sensor array is attached to the main PCB through the multiple headers located at the edges of the board, shown in Figure 3.

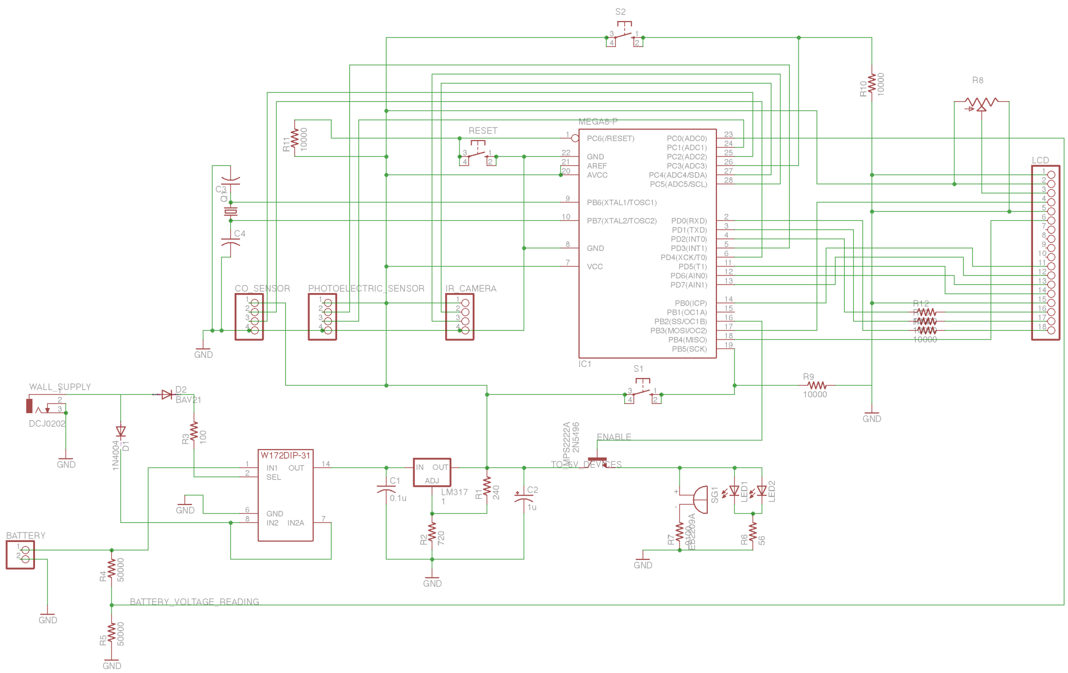

Figure 2: Main PCB Circuit Schematics

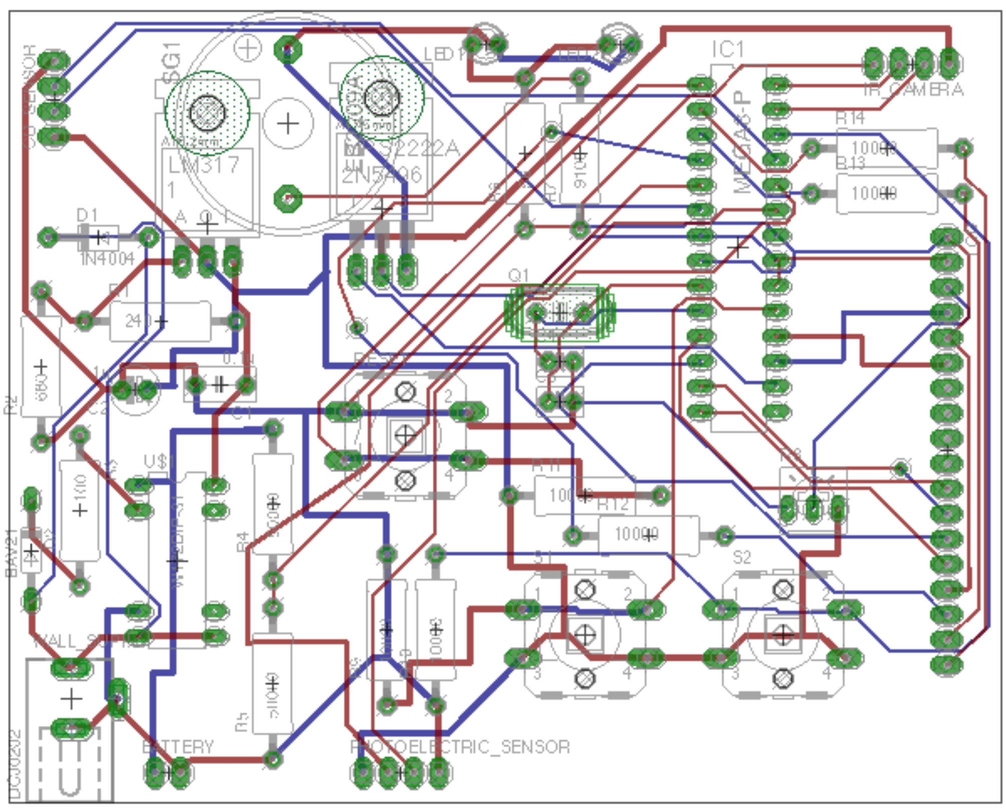

Figure 3: Main PCB Board Layout

2.2.1 Power Management System

The power management system was designed to safely power all the components of the device. To operate during a power outage, a reed relay is used to quickly change the power source of the device. This is an improvement on the original design method using a MOSFET because it can switch faster. A diode was also added from the wall supply to the normally-open terminal of the relay. This stops the input capacitor of the voltage regulator from affecting the coil voltage on the relay used for determining when the main power supply is delivering power. The resistors used with the linear regulator were selected using Equations (1) and (2) from [15].

<u>LM317 Operating Voltage calculations</u>

$$V_{out} = 1.25\ V\left(1 + \frac{R_2}{R_1}\right) = 5.0\ V \tag{1}$$

$$\frac{R_2}{R_1} = 3 = \frac{720\ \Omega}{240\ \Omega} \tag{2}$$

After moving the components to the main PCB, the output voltage increased from 5.01 to 5.3 volts, which was outside of the desired range. The resistor R_2 was changed from 720 Ω to 680 Ω, which resulted in an output voltage of 5.03 volts. The updated schematic of the power management system is found in Figure 4.

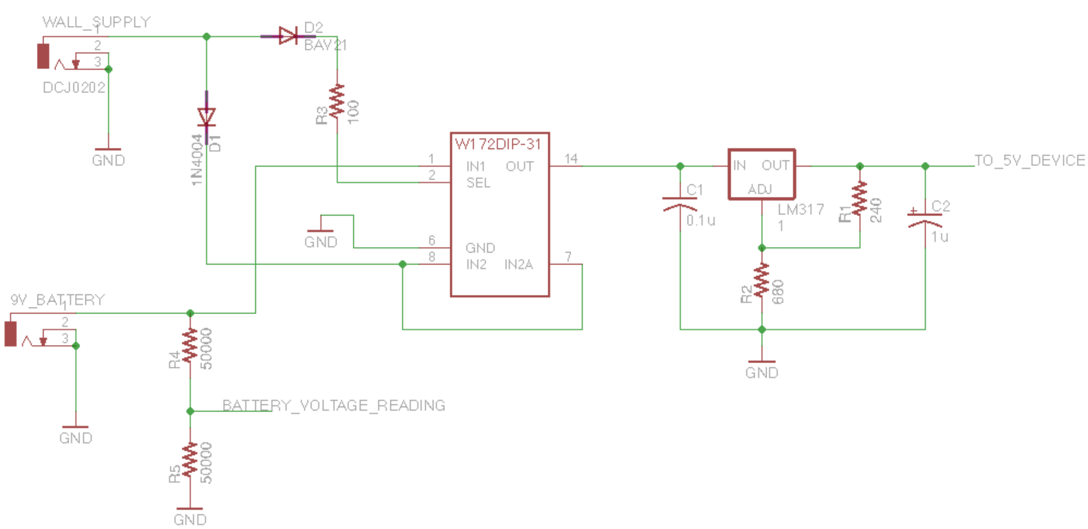

Figure 4: Power Management Circuit Schematics

2.2.2 Alarm System

The alarm system was designed to alert people near the device when an alarm is activated. The alarm system is activated through the MPS222A bipolar junction transistor [17] with the base pin connected to the MCU. The LEDs were changed from being in series to parallel after the design review, since it allows the LEDs to be brighter. Choosing the best voltage for the base pin would determine the current into the alarm system, which controls how loud the buzzer is and how bright the LEDs are. Table 1 shows the measured current with each associated base voltage. Five volts was selected as the best base voltage because the current draw is not significant and it gives the most noticeable alarm. The final circuit schematic for the alarm system is in Figure 5.

Table 1: Current into Alarm System with Varying Base Voltage

V_{in}	V_{base}	I_{in}
5.01 volts	4 volts	25.1 mA
5.01 volts	4.5 volts	33.1 mA
5.01 volts	5 volts	41.3 mA

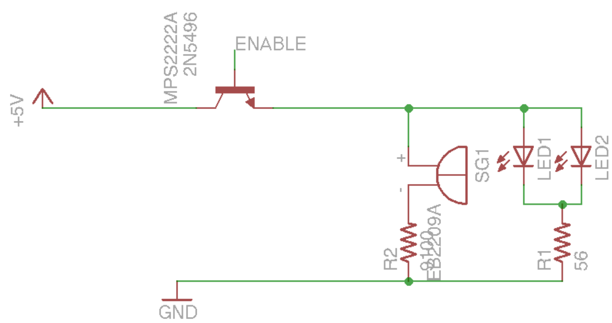

Figure 5: Alarm System Circuit Schematics

2.3 Sensor Array

The modular design of the sensors allows for quick prototyping and testing. The individual sensors are designed as separate circuits, and are designed to work independently using 4 pins each. To standardize the pinouts with the main board design, we used the following convention in Table 2.

Table 2: Sensor Module Pinouts

Sensor Module Pins	I2C sensors	Analog sensors
1	+5V	+5V
2	I2C Data	Switch Select
3	I2C Clock	Analog Voltage
4	GND	GND

2.3.1 Thermopile array

Updating the design from the design review, we tested the original design with level shifters. It was determined that they were not necessary since the device was able to communicate to the MCU without them. In addition, the voltage regulator was not providing sufficient voltage to the sensor, so we made a small modification to the resistor combination to slightly increase the voltage output of the regulator, as shown in Equations 3 & 4.

During the testing of the device, we made a mistake in the board design and used a mirror image of the actual sensor. Quickly redesigning & manufacturing a new board with the correct orientation fixed this.

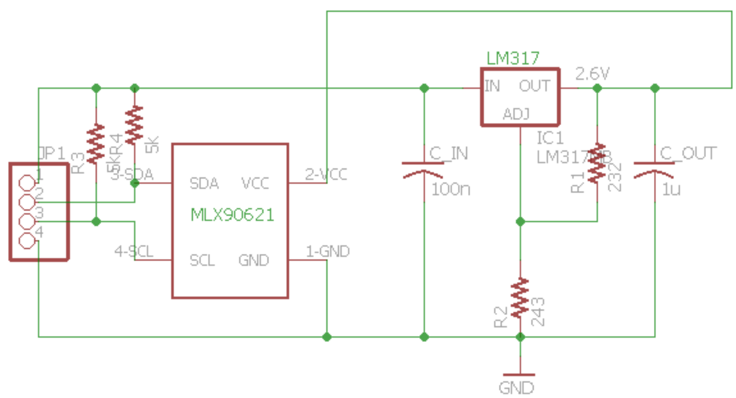

Figure 6: Thermopile Array Circuit Schematics

<u>LM317 Operating Voltage calculations</u>

$$V_{out} = 1.25\,V\left(1+\frac{R_2}{R_1}\right) = 2.6\,V \quad (3)$$

$$\frac{R_2}{R_1} = 1.08 = \frac{243\ \Omega}{232\ \Omega} \quad (4)$$

After performing verification tests, we determined the temperature thresholds to be used for the thermopile array as a gauge of different dangerous temperatures typically encountered in a kitchen, at which we would sound the various alarms to warn occupants. The temperature thresholds are shown in Table 3.

Table 3: Temperature Alert Modes

Alert Mode	Temperature Threshold
High	240°C
Medium	200°C
Low	180°C
Device Temperature High	45°C

2.3.2 Carbon Monoxide Sensor

The carbon monoxide sensor design remained largely unchanged from that of the design review, except that the load resistor was fixed at 4.75 kΩ. The 150-second heating cycle was managed by the MCU and coded into the software of the alarm system.

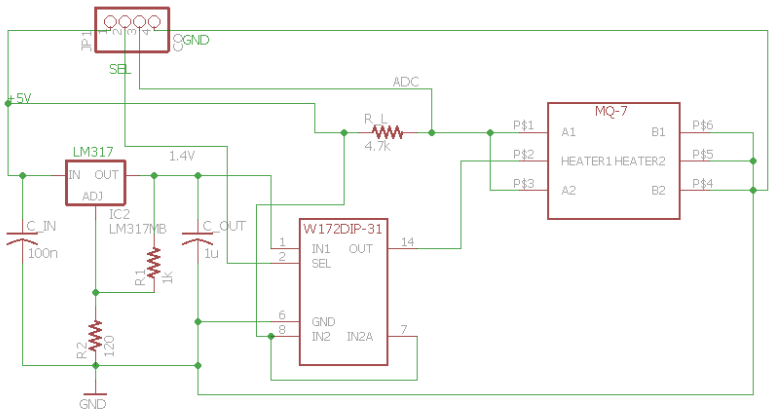

Figure 7: CO sensor Schematics

To supply the MQ-7 device with 1.4 V at the low part of the heating cycle, we used a voltage regulator circuit with parameters given in Equations (5) and (6). To provide a reliable means of switching the voltage levels between 1.4 V and 5V, we used a magnetic reed relay controlled by the MCU through pin 2 of the sensor.

$$V_0 = 1.25\,V \times \left(1 + \frac{R_2}{R_1}\right) = 1.4\,V \tag{5}$$

$$\frac{R_2}{R_1} = 0.12 = \frac{120\Omega}{1000\Omega} \tag{6}$$

The design was to use a voltage divider with the load resistor and the sensor resistance as the resistor pair and to use the MCU ADC to detect changes in the resistance of the sensor, using Equations (7) and (8).

$$V_{ADC} = 5V\,\frac{R_{sensor}}{R_{sensor} + R_{Ref}} \tag{7}$$

$$R_{sensor} = \frac{-V_{ADC}\,R_{Ref}}{V_{ADC} - 5V} \tag{8}$$

2.3.4 Photoelectric Smoke Detector

After performing some tests with the IR LED, photodiode and various op-amps, we chose the LM741 op-amp, as it had the best dynamic range for the design and had a good gain. The resistor was set at 2.5MΩ, which was determined earlier to be the optimal resistance for amplifying the signal.

In addition to the selection of the appropriate resistor values, a significant bug was found in the design of the circuit schematics during the design review process, which was not found until the prototyping phase of the smoke detector. The + terminal of the op-amp was not grounded [8], leaving the voltage of the photodiode floating and producing erroneous results. This was fixed and the PCB design and schematics were modified to account for the changes.

Due to the limits of actual op-amps, I found that the op-amp I used, the LM741, did not have the rail-to-rail voltage outputs, but rather had a saturation voltage output of 4.63V for source voltages of 0 and +5V. This limits a small portion of voltages available, but it should not have any large bearing on the sensor design.

As we tested the device, we found that the op-amp pinouts were also mirrored. To provide a quick fix to the problem, we mounted the op-amp on the underside of the PCB, allowing us to reuse the original PCB design. The schematic is found in Figure 8.

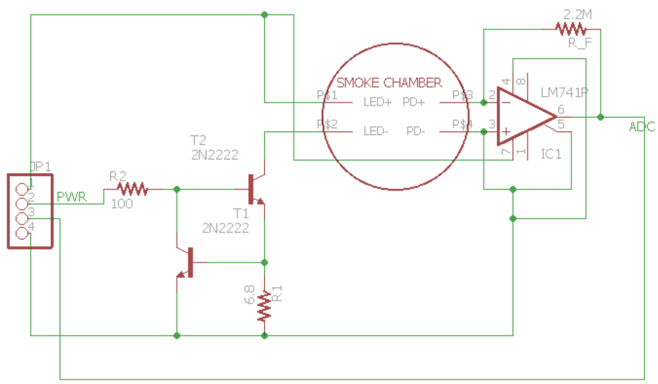

Figure 8: Photoelectric Smoke Detector Circuit Schematics

From the circuit, we can obtain the output voltage using ideal op-amp conditions: $V_{ADC} = i_p\ R_f$, for $V_{ADC}\ < V_{in} = 5V$.

As the calculations of the design review assumed the ideal case, we did multiple experimental tests with canned smoke to determine the threshold levels for smoke detection, and we arrived at the threshold voltage of 2.5V. The sensor was rather sensitive to even a small amount of smoke, often raising the voltage levels to about 3.5 V on certain occasions.

From the design review, we also grossly underestimated the current draw of the IR LED in the design, which happened to draw about 100mA of current when powered on, causing some parts of the circuit to overheat. However, using a 10% duty cycle on the LED using the digital switch line from the MCU mitigated the large amount of current drawn. However, this prevented us from reading the sensor instantaneously and resulted in a 10 second reading cycle, which we deemed to be sufficient for the setup, given that most typical smoke alarms also take about 20 seconds to respond.

2.4 Software Flowchart

To manage the many alarm modes and various states, we designed a finite state machine using C++ code to properly determine which state the alarm is in. As seen in the program flowchart below in Figure 9, we have an initialization phase, where we set up the various sensors and the MCU for the system. In addition, the Emergency Detected states are entered when the MCU reads sensor data above the thresholds determined from [16], which sounds the alarm. To exit the Emergency Detected states, the Silence button must be pressed.

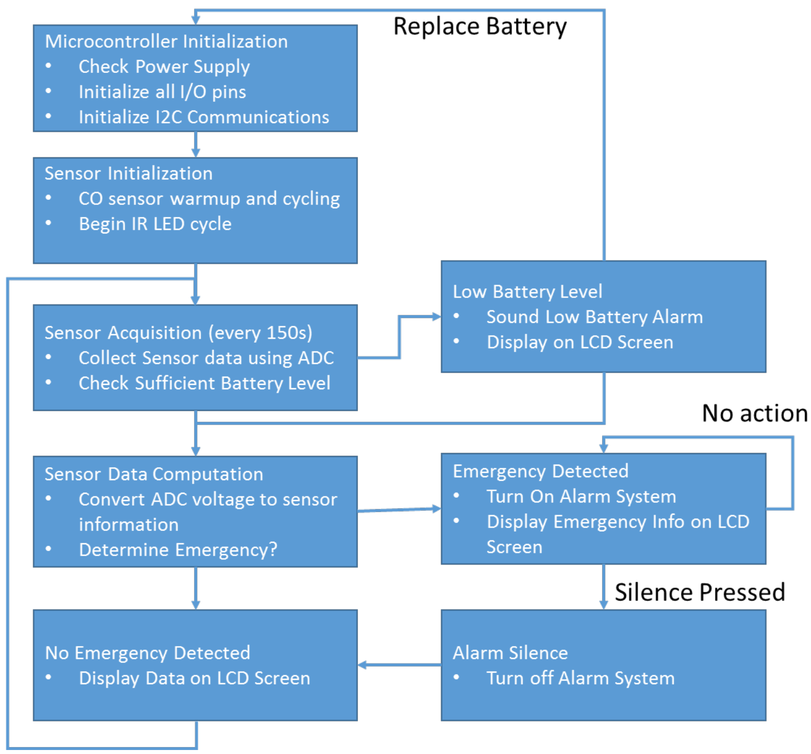

Figure 9: Software Flowchart Diagram

3 Design Verification

The prototype was a feasible proof of concept, where we have shown that the product should be able to function upon some refinement to the prototype. The sensors individually have proven themselves to be effective in detecting the dangers, but some more work would have to be done to properly integrate them as a system.

3.1 Main PCB

3.1.1 Power Supply

To verify that the power management system outputs the correct voltage and switches to the battery when power is lost, the main outlet power supply was simulated with a waveform generator. A square wave was used to oscillate between 0 and 8 volts, which is how the power supply acts when switched on and off. The resulting output is shown in Figure 10 in channel 2 (green) and the power supply is represented in channel 1 (yellow). The steady-state output voltage remained within the allowed range.

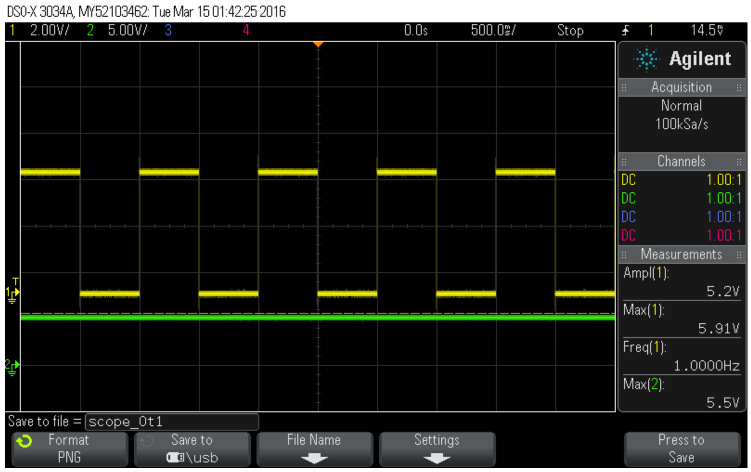

Figure 10: Oscilloscope Image of Input and Output of Power Management System

3.1.2 ATMEGA328P (MCU)

To test if the input and output pins of the ATMega328P were working, we simulated the input conditions first. Using the function generator, we generated a 3.1V DC signal and the ATMega328P read this signal as a digital high, which was expected. Then, we generated a 1.4V DC signal, and the ATMega328P read the signal as a digital low, as expected. Then, we programmed the ATMega328P to output a digital high and low on various selected pins, and using a voltmeter, measured the voltage of those pins, the voltage of the high was about 4.8V and the low voltages was about 0.6V. To test the ADC, we used a slowly varying ramp function from the function generator, and the ATMega328P read the voltage levels to within 0.05V of the output, as compared to the oscillator output.

3.2 Sensor Array

3.2.1 Thermopile Array

To test the thermopile array, we first checked the voltage output of the regulated output, being measured to be 2.62V. Then after connecting the sensor module to an Arduino Uno, we read out the temperatures from the sensor using a serial interface.

Placing various objects below 100℃, we then measured the temperature of the objects with a contact thermometer and checked against the sensor readings. Using a butane lighter, we placed it at a distance of about 40 cm and measured the temperature of the flame, the sensor saturated at 348℃. To simulate a fire at a longer distance, we utilized a hot plate of 20 cm diameter at a distance of 2.5m. The sensor managed to read the temperature readout of 328℃ before the hotplate melted.

3.2.2 Carbon Monoxide Sensor

To properly calibrate the CO sensor, we used a canned CO source, containing 2800 ppm of CO and enclosed both our sensor and a CO alarm with digital readout in an air-tight container. We measured the resistance of the semiconductor over the heating cycle every second. We then plotted the resistance over time, varying the CO concentration in the container. As we see in the Figure 11 below, it shows the progress of one run being made. There is a visible trend that increasing CO concentrations would decrease the resistance of the semiconductor, as seen in the datasheet for the sensor [3].

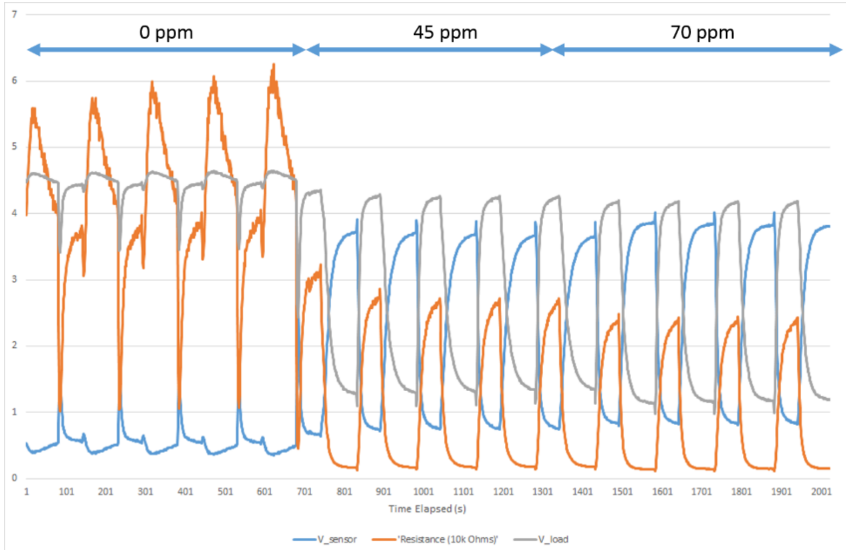

Figure 11: CO Sensor Raw data

Collecting more samples over several runs, we plotted them and used a quadratic regression fit to obtain a relationship to approximate the concentrations from the resistance of the semiconductor, as seen in Figure 12 below.

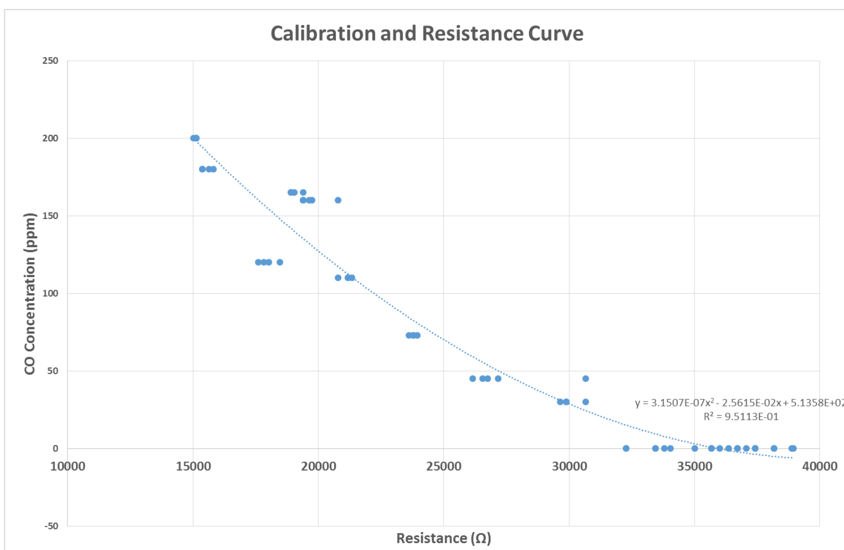

Figure 12: Regression plot of relationship between CO Concentrations and Resistance

With the regression in place, we tested the sensor again, and obtained the measured ppm to be within +/- 8ppm of the commercial CO detector value. As such, our device did not perform to specifications, but it may be due to the calibration method, as the CO detector has a significantly large delay in the response to CO concentration and may not be accurate. We would need a better standard to calibrate our sensor against.

As part of the verification process, we found that the CO sensor also responded to different environmental conditions. The first was the humidity level of the surroundings, which would lead to unpredictable readings. This can be resolved by adding a humidity sensor. Next, the presence of other contaminants in the air caused the sensor to function incorrectly, particularly the presence of ethanol, a key ingredient in the canned smoke test. This was accidently found to cause erroneous high readings of CO gas since the sensor also does react with ethanol, as seen in its datasheet.

3.2.3 Photoelectric Smoke Detector

A canned smoke source[1] is available commercially and used widely in the fire safety industry as a testing method for smoke detectors. As we are unable to find a reliable means to test the sensor, due to the lack of research knowledge or the simple absence of tests, we devised our own means of testing the device. We carried out testing by repeatedly spraying the smoke chamber with the canned smoke source and measuring the output of the photodiode and amplifier circuit. As we notice from the testing, we obtained a 21 out of 24 (87.5%) correct identification, the remaining 3 being false negatives. This is slightly below the specification, but due to the statistically small sample, the data being collected is still inconclusive and would require more testing to verify the sensor.

3.3 Program Flow

After integrating the system, we tested the programming of the system by triggering all of the sensors using the appropriate tests. Upon detecting the signal, the program did enter into the Emergency Detected state, sounded the alarms, and displayed the relevant information on the LCD. Upon the press of the Silence button, the program returned to the original state for 60 seconds, after which the alarm could be triggered again.

In addition, as the timing of the heating cycle was rather critical, we measured the relative clock drift of the program to be at 56 milliseconds over 1 hour, which is rather negligible and hence could be ignored.

[1] Active Ingredients: LPG, Ethanol, 1,2,3-Propanetriol

4 Costs

4.1 Parts

Prices are quoted as per unit cost

Part	Quantity	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
MCU (Atmel ATMega 328P)	1	$4.30	$1.83	$4.30
Crystal Oscillator 16.000 MHZ HC49/US	1	$0.95	$0.45	$0.00
Triad 6V DC Supply	1	$8.00	$5.00	$8.00
Thermopile Array (Melexis MLX90621ESF-BAA-000-SP)	1	$74.38	$54.10	$80.00
Carbon Monoxide Gas Sensor – MQ7	1	$7.25	$6.53	$7.25
LM317 Linear Regulator	3	$0.60	$0.25	$0.00
SPDT Reed Relay (W172-DIP-31)	2	$6.95	$9.86	$6.95
Photoelectric Smoke Chamber	1	$10.00	$2.00	$10.00
IR LED	1	$0.75	$0.10	$0.00
IR Photodiode (LTR-516AD)	1	$0.70	$0.10	$0.00
Op-Amp LM741	1	$0.66	$0.23	$0.00
Speaker	1	$1.50	$1.00	$0.00
Super Bright LEDs (red)	2	$0.25	$0.10	$0.00
LCD Character Display	1	$14.95	$14.95	$18.95
Large Push Buttons	3	$0.25	$0.10	$0.25
Other Miscellaneous parts (transistors, Resistors, capacitors, diodes, etc.)		$10.00	$1.00	$3.00
Custom PCB Fabrication	4	$10.00	$0.10	$0.00
Product Case	1	$22.00	$4.00	$22.00
Total Costs		$190.39	$123.02	$160.7

4.2 Labor

Name	**Hourly Rate**	**Hours**	**Total=Hourly Rate x 2.5 x Hours**
James Kok	$30.00	200	$15,000.00
Jens Foyer	$30.00	200	$15,000.00
Total		400	$30,000.00

4.3 Total Costs

Section	**Cost**
Labor	$30000
Parts	$160.70
Grand Total	$30160.70

5 Conclusion

5.1 Accomplishments

The fire detector passed all of the tests that were applied. This is critical for building a device that is designed to offer safety to the user. When exposed to dangerous temperatures or test gases, the alarm system was successfully activated. The device was also able to switch between power sources without resetting the MCU. Our project is a large step toward reducing the number of deaths from home fires.

5.2 Uncertainties

The fire detector functioned as it was supposed to during our tests, but it can be difficult to know how accurate our readings are. In testing the photoelectric sensor, there is no standard available for a reasonable price that determines the threshold levels for detecting particles in the air. For the carbon monoxide sensor, the ideal calibration method includes using a $400-500 carbon monoxide detector. Our calibration was performed with a similar and less expensive device, but with much less accuracy. This can affect our device's sensitivity. During our final demo, we had to use a soldering iron to test the thermopile array instead of an actual fire. The temperature readings would likely be much more accurate for such a large source.

5.3 Safety Statement

Our project deals with the issue of fire safety, so we would need to test our product with real fires to ensure that the product is reliable and useful. As such, we would have to take the following precautions during testing of the sensors:

- Have sufficient ventilation during testing
- Have fire extinguishing equipment nearby in case of emergencies
- Know the emergency evacuation route and the emergency contact numbers
- Do not inhale any dangerous fumes

Our project deals with carbon monoxide sensors, so we would have to use a CO test source to calibrate and characterize the sensor. Also, we use canned smoke to simulate smoke in the photoelectric detector, which may pose inhalation risks. As such, we would have to take the following precautions:

- Work in a well ventilated space, preferably in a fume hood
- Use caution when injecting CO gas into test chamber
- One person work, the other observe at a distance & never work alone
- Ensure test chamber is well ventilated after use
- Do not inhale any fumes or smoke

5.4 Ethical considerations

As this product deals with the safety of consumers, we would have to continue to redesign the product to meet the standards required for the safety of our users. As such, we would not release the product until such qualifications are met, such as Underwriter Laboratory standards.

5.5 Future work

To better improve this product, we would have to modify some of the sensors and power management system to make them much more reliable and more cost effective. Next, to make it a more useful fire detection system, we propose making a mesh network of sensors connected wirelessly to enhance the detection rate of dangers. With that in mind, we could adapt the system to be part of a smart home system and connect it to the internet to allow home owners to be more aware of their homes. To make a better infrared thermal camera system, we could use a higher resolution camera at the expense of higher cost. Using a camera with a higher resolution would allow for better fire identification. In addition, using higher quality sensors would allow for a more effective sensor fusion.

References

[1] Melexis FIRray: 16X4 Far Infrared Array Datasheet [Online].
Available: http://www.melexis.com/Asset/Datasheet-IR-thermometer-16X4-sensor-array-MLX90620-6099.aspx
[2] Implementing a smoke detector with the MSP430F2012. Retrieved February, 2016
Available: http://www.ti.com/lit/an/slaa335/slaa335.pdf
[3] MQ-7 Carbon Monoxide Detector Datasheet:
Available: https://cdn.sparkfun.com/datasheets/Sensors/Biometric/MQ-7%20Ver1.3%20-%20Manual.pdf
[4] System-Connected Carbon Monoxide Detectors:
Available: http://www.systemsensor.com/en-us/documents/system-connected_co_detector_app_guide_coag103.pdf
[5] Smoke detector design and smoke properties. Richard W. Bukowski, George W. Mulholland. Bukowski, Richard. Washington Dept. of Commerce, National Bureau of Standards, National Engineering Laboratory, Center for Fire Research: 1978.
Available: http://hdl.handle.net/2027/uiug.30112101584602
[6] 172DIP PCB Mount Miniature Reed Relay Datasheet
Available: http://www.serelays.com/library/section8/105A_172DIP.pdf
[7] 16x2 LCD Character Display
Available: https://www.sparkfun.com/datasheets/LCD/GDM1602K-Extended.pdf
[8] Photodiode Monitoring with Op-Amps
Available: http://www.ti.com/lit/an/sboa035/sboa035.pdf
[9] Arduino: From Arduino to Microcontroller on a Breadboard. Retrieved February, 2016.
Available: https://www.arduino.cc/en/Tutorial/ArduinoToBreadboard
[10] Atmel 8-Bit Microcontroller with 4/8/16/32KBytes In-System Programmable Flash Datasheet Summary & Complete [Online].
Available: http://www.atmel.com/Images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Summary.pdf
[11] M. Ahrens, "Home Fires Involving Cooking Equipment", NFPA, 2016.
[12] BSS138 N-Channel Logic Level Enhancement Mode Field Effect Transistor Datasheet
https://www.adafruit.com/datasheets/BSS138.pdf
[13] SMOKE DETECTOR TECHNOLOGY AND THE INVESTIGATION OF FATAL FIRES
http://www.interfire.org/features/smokedetector.asp
[14] Carbon Monoxide Levels & Risks by NCI
http://www.cloudhvac.com/webapp/GetFile?fid=%7B4725DFB6-DF9E-418D-8152-022F99F95E93%7D
[15] 3-Terminal Adjustable Output Positive Voltage Regulators [Online]. Available:
http://pdf.datasheetcatalog.com/datasheet/vishay/74809.pdf
[16] Occupational Health Guideline for Carbon Monoxide [Online]. Available:
http://www.cdc.gov/niosh/docs/81-123/pdfs/0105.pdf
[17] General Purpose Transistors. Available: http://www.onsemi.com/pub_link/Collateral/MPS2222-D.PDF

Appendix A Requirements and Verifications Table

Requirements	Verification	Verified
IR Camera/Thermopile Array a) Voltage Source is providing at least 2.5V and at most 2.7V of Power b) Interfaces with MCU c) Measures Temperature that is accurate to +/- 5.0 K d) Able to detect a small fire through high temperature	IR camera/Thermopile Array a) Connect with 5V power supply, measure the voltage at the output of the linear regulator, it should be 2.6 to 2.7 V b) Connect/Wire Up with MCU, programmatically read the thermopile sensor using I2C c) Measure Temperature using a thermocouple or calibrated IR camera and compare against IR camera readings d) Set up a simulated kitchen fire & programmatically read the temperature readouts and check whether alarm signal on MCU has been raised	(a) to (c) verified (d) not yet verified
Carbon Monoxide Sensor a) Voltage source to the heater element must comply with heating cycle b) Provide Carbon Monoxide ppm readings to +/- 5 ppm, Computed & read & displayed as a value on the MCU c) MCU triggers alarm when above thresholds (35 & 70 ppm)	Carbon Monoxide Sensor a) Connect with MCU, programmatically set the heating cycle. measure voltage over the heating cycle, ensure voltage is +/- 0.2V, duty cycle of 40% and cycle is 150 +/- 1 s long b) Connect Sensor to MCU. Collect Carbon monoxide in a container and measure it with against another calibrated sensor or using calculated concentrations and measure programmatically with the CO sensor, measurements should be within +/-5 ppm c) Connect Sensor to MCU and program the MCU, perform (b) at 40ppm & 80ppm, test whether alarm signal on MCU is triggered	Verified (b) not to spec, accuracy of +/-8ppm
Photoelectric Smoke Detector a) Measure and determine required thresholds for smoke detection, by measuring the scattering when smoke is not present and when smoke is present	Photoelectric Smoke Detector a) Connect to MCU, measure analog output b) Repeatedly check whether alarm signal is triggered by the MCU when a. Canned smoke spray on the smoke chamber	Verified

b) Triggers when smoke is detected at least 9 out of 10 times (>90%) c) Able to detect smoke from an actual fire source at least 9 out of 10 times (>90%)	b. And/or Using a butane lighter/candle and smother the flame to generate smoke at a distance no less than 3 feet c) Set up a simulated kitchen fire check whether alarm signal on MCU has been raised	
Power Management System a) Must Supply 5V +/-0.15 V at a minimum of 200mA b) Must be able to switch from wall outlet to battery power and vice-versa with minimal interruption to microcontroller c) Battery to last for at least 1.5 hours under normal load d) Battery to last for at least 30 minutes under full alarm load e) MCU must be able to read battery voltage to +/-0.1V	Power Management System a) Measure the voltage at the output and current through a 25 ohm resistor using a digital multimeter when the device is on under both mains and battery mode. Check that the voltage is 5 volts +/- 0.15 volt b) Switch from wall outlet to batteries by turning on/off wall supply and check whether voltage of voltage source remains constant using a oscilloscope c) Obtain a new 9V battery and attach a 25 ohm resistor from the output to ground. Measure voltage at the output and current through the resistor every 5 minutes, until the output voltage drops below 4.9 volts. Measure how long it takes for the voltage to drop below 4.85 volts using a stopwatch. d) Obtain a new 9V battery and attach a 22 ohm resistor from the output to ground. Measure voltage at the output and current through the resistor every 5 minutes, until the output voltage drops below 4.9 volts. Measure how long it takes for the voltage to drop below 4.9 volts using a stopwatch e) Connect the power supply to the MCU, measure programmatically the voltage of the battery & compare against the voltmeter reading of the battery	Verified
Alarm System a) Must be able to sound buzzer loud enough when voltage is supplied	Alarm System a) Using a function generator, supply 5 volts to the input and 4 volts to the enable pin. Then measure the	Verified

b) Must be able to light up LED when voltage is supplied c) Both buzzer and LED must activate when given a high enable signal from the MCU	current into the alarm system with a multimeter and check that it is at least 20 mA. b) Using a function generator, supply 5 volts to the input and 4 volts to the enable pin. The LEDs should light up c) Connect the enable pin of the alarm system to pin 16 of the MCU and programmatically set pin 16 to 4 volts. Check that the buzzer and LEDs are activated	
User Interface Buttons a) When Button is pressed, the output must be High (V>3V) b) When Button is not pressed, output must be low (V<1.5V)	User Interface Buttons a) After connecting the 5V power supply to the input and a 1000 ohm resistor from the ouput to ground, measure the output voltage using a multimeter when the button is pressed and check that it is more than 3 volts. b) After connecting the 5V power supply to the input, measure the output voltage using a multimeter when the button is not pressed, and check that it's less than 3 volts.	Verified
Display Interface (16 x 2 Character LCD) a) Display Text from Microcontroller b) Determine the required contrast level to view the LCD display c) RGB LED control able to represent combinations of colours	Display Interface (16 x 2 Character LCD) a) Connect to a MCU, Set programmatically the text string "Hello World" on the MCU, check String printed on the Display unit b) Perform (a), then adjust potentiometer to obtain maximal visual contrast c) Connect to a MCU, Set programmatically the PWM levels for each colour and cycle through various intensity levels	Verified