March 11, 2010

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, British columbia
V5A 1S6

Re: ENSC 440 Design Specifications for a Flame Extinginuishing Intelligent System

Dear Dr. Rawicz:

Enclosed is our document, "Design Specifications for a Flame Extinginuishing Intelligent System", which describes both high level and low level design specifications of our product.

Our Flame Extinginuishing Intelligent System (FlexiSys) product will be able to address current inefficient methods of extinguishing flames. By utilizing motors, sensors, and algorithms, FlexiSys will be able to detect and extinguish a flame, minimizing both fire and water damage to any structure.

These design specifications include the physical, electrical, and mechanical aspects of our system, and the integration needed to complete our working model. Specific parts will be listed and detailed in preparation for our production model. In addition, future possible designs will be discussed as well.

If any there are any questions or concerns, please feel free to contact me by email at ENSC440-fire@sfu.ca or by phone at (604) 780-3392.

Sincerely,

Kelvin Ho
President and CEO
FlexiSys – Flame Extinguishing Intelligent System

Design Specifications

For a Flame Extinguishing Intelligent System

Project Team: Kelvin Ho
Ken Zheng
Peter Zheng
Luke Dang

Contact Person: Kelvin Ho
Ensc440-fire@sfu.ca

Submitted to: Dr. Andrew Rawicz – ENSC 440
Steve Whitmore – ENSC 305
School of Engineering Science
Simon Fraser University

Issued date: March 11, 2010
Revision: 1.0

Executive Summary

Sprinkler systems and other similar fire extinguishing systems all share three critical flaws: the inability to correctly determine a flame, the inability to aim the extinguishing material, and lastly, the inability to control the amount of extinguishing material. These factors lead to excess water damage and failure to extinguish the flame.

These design specifications for our Flame Extinguishing Intelligent System will provide detailed descriptions for both the design and development of our product and the justification behind our choices. These specifications will explain the high level specifications of the entire system, followed by detailed specifications for two of our major subsystem designs: the physical and electronic system design. In addition, we include detailed test plans to ensure the stability and the robustness of our system.

The system overview section details several simple points regarding the overall components. Included in this section are the physical properties, the mechanical build, and the electronics involved in the system. Furthermore, the physical and mechanical design section provides detailed figures and critical information such as the materials used and the structural integrity. The electronic system design specifications will list the components such as the microcontroller, the sensors, and motors. In addition, justification for the part choices will illustrate the depth of research prior to the design of this system.

The working model is designed to be a proof-of-concept model and will not be actually installed in buildings. Once the working model is complete, optimizations will be performed in both the physical and electrical aspects of design. The production model will minimize the footprint through contract manufacturing and costs will be reduced by bulk purchases from manufacturers.

These design specifications were based on our functional requirement documentation: *Function Specification for a Flame Extinguishing Intelligent System* [1]. As we proceed with our designs, the specifications in this document will be updated accordingly as well. This also extends to our functional specification documentation.

Table of Contents

List of Figures

List of Tables

Acronyms

DC	Direct Current
EEPROM	Electrically Erasable Programmable Read-Only Memory
I/O	Input/ Output
IR	Infrared
Pa	Pascal
psi	Pound per inch
PWM	Pulse Width Modulation
RAM	Random Access Memory
RX	Receiver
SRAM	Static Random Access Memory
TX	Transmitter
USB	Universal Serial Bus
VDC	Voltage Direct Current

Glossary

Baud Rate	Modulation rate for electronic device, similar to symbols per second or pulse per second.
Infrared Thermometer Module	Non-contact temperature sensor.
Servo Motor	an automatic motor that provides error feedback for position correction.

1. Introduction

FlexiSys is a prototype flame extinguishing intelligent system which will detect flames in an area and proceed to extinguish the flames. Our product will ultimately be mounted on the ceiling, similarly to a passive fire sprinkler. Furthermore, we will minimize the power and space requirement, such that our product can replace current passive sprinkler systems with ease. We wish to obtain this goal, along with the following design specifications and requirements by April 15th, 2010.

1.1 Scope

This document describes the design specifications requirements chosen in order to satisfy and fulfill the functional requirements listed as priority II in *Functional Specifications for a Flame Extinguishing Intelligent System* [1]. As these design specifications were constructed specifically for our working model, they may change as we proceed onwards to our production model. In addition, theoretical calculations and discussion of alternatives will be included in the scope of this document.

1.2 Intended Audience

This design requirements documentation is created primarily for our team members of FlexiSys, and in addition, to interested parties that may be interested in investing in our company. It will be our basis for our engineers as they build each component to an agreed design. In addition, it will provide detail information for our test engineers once the product has been completed. Lastly, it will serve as a reminder and written measure during the entire life cycle of our product.

2. System Overview/Overall System Design

The Intelligent Flame Extinguishing System is a system that can be modeled at a high-level as shown in Figure 1.

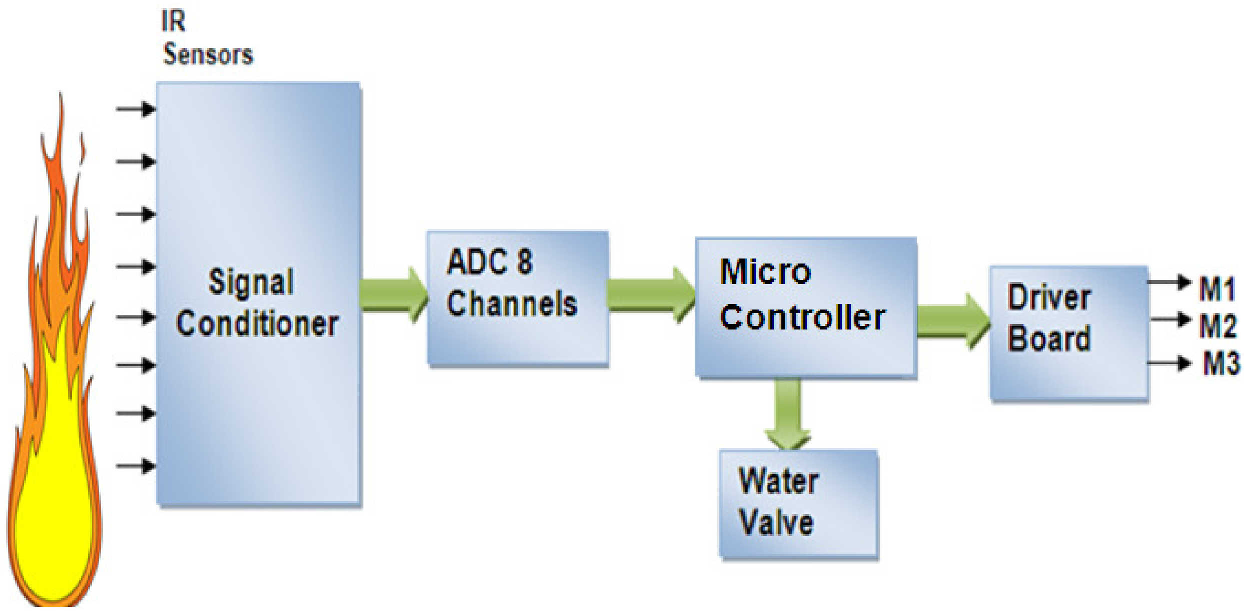

Figure 1: Block Diagram for the Fire Fighting System

The working theory is as follow: The IR sensor stack constantly rotates to monitor the environment temperature. When a fire starts, there is significant infrared light emitting from the flame. When the infrared light enters the monitoring range of any IR sensor, the IR sensor will send an analog signal to the ADC, which will convert the analog signal to digital signal. This digital signal will then be sent to the microcontroller informing that there is a fire. By controlling servo motors M1, M2 and M3 (which is connected to the nozzle), the microcontroller will direct the nozzle to the location where the fire is. Once the nozzle rotates to the direct location of the fire, the microcontroller immediately switches on the water valve to let the pressurized water to flow. When the fire is extinguished, the IR sensor will notify the microcontroller. In order to ensure the fire is totally extinguished, the microcontroller will keep the water valve on for an extra 30 seconds after it receives the fire-off signal from the IR sensor. Unlike current sprinkler systems which will still stay on even if the fire is off, our fire fighting system will reduce the water damage to the building.

Design details that are common to all parts of the fire fighting system will be discussed in upcoming subsections, while design details specific to individual parts of the fire fighting system can be found in their respective sections.

2.1 Physical Properties

In this section, we explain how all the parts are assembled together. The finished system will resemble Figure 2 below. The complete system will be roughly 330 mm in height, 210 mm in width and 2 kg in weight. During rotation, the system will require 210 mm radius of rotation.

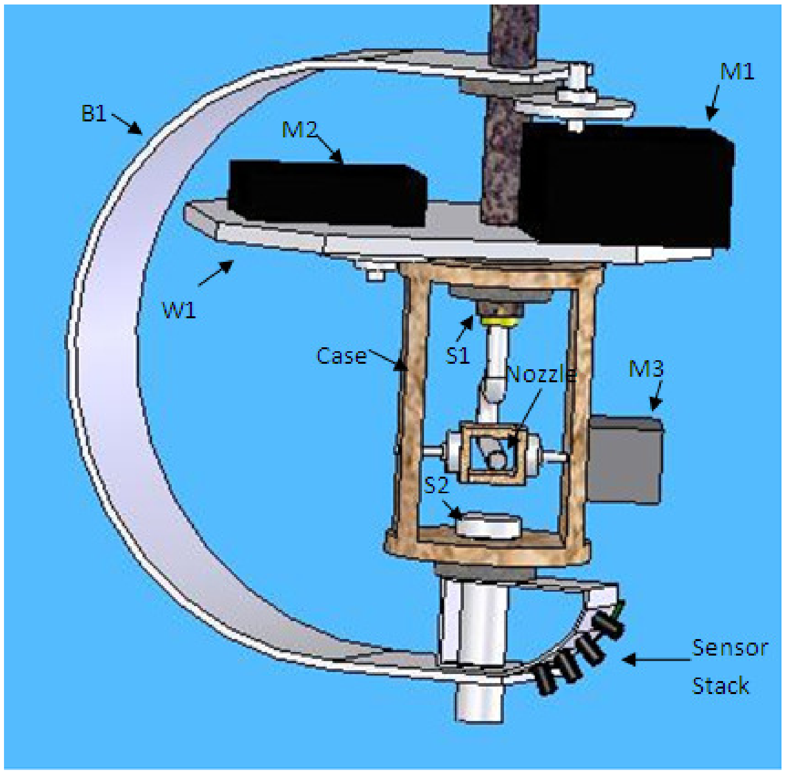

Figure 2: Overall System

Like current sprinkler systems, the water pipe is concealed inside the ceiling with one end protruding out where the whole system is mounted onto. The end of the water pipe goes through one of the holes of the C bracket (B1) which is secured by a nut to prevent motion. The top end of B1 is connected to a gear which is in contact with another gear from M1. The wood board (W1) is clamped on the water pipe and is stationary with respect to the water pipe. Servo motors, M1 and M2 are mounted on top of W1. M1 is used to rotate B1 which ultimately rotates the sensor stack at the bottom of B1. Below W1 is the case which is connected to the water pipe through a bearing so that the case can be freely rotated by M2. The case is used to rotate the nozzle in an azimuthal direction. The nozzle is connected to the shaft (S1), in the middle of the case. This shaft can be rotated by M3 through a series of gears. By controlling M3, the nozzle can swivel up and down as needed. There is a small shaft (S2) at the bottom of the case which is connected to the lower end of B1. The metal bracket for the sensor stack is connected to this shaft so the whole sensor stack will rotate with B1, but the center casing will be independent. When M2 rotates B1, B1 turns S2, and S2 rotates the sensor stack so that they can monitor the space in 360° angle.

2.2 Power Supply

The following power requirements for the flame extinguishing intelligent system must be met:

- Voltage: 6VDC and 12VDC
- Maximum Current: 3A
- Number of connectors: minimum 4
- Can be run from an electrical outlet
- AC to DC conversion
- Regulated

The maximum current is determined by summing the current requirements for all electronic components. The solenoid valve requires supply voltages of 12V, while the motors require 6V. The four connections to the power supply include: one for the solenoid valve, two for the two servo motors, and one for the DC motor.

We are going to borrow a power supply from school to prove the concept of our fire fighting system as this will limit the cost of the prototype. In the final product, the fire fighting system will be connected to a backup power supply to ensure that the system is always on and active when the main building power is lost.

2.3 Sensor placement

The MLX90614 infrared thermometer module is a temperature sensor with a 10 degree field of vision, which has been chosen as the main sensor. These sensors have been selected due to their accuracy, range and easy interfacing. In our design, there are 4 MLX90614 sensors assembled together in a stack of sensors, illustrated in Figure 4 below. Using 4 sensors in combination will allow the system to have an approximate range of 65 degrees. From our preliminary research, flames and fires are most present within the lower 60 degrees. In addition, our mechanical system currently can only support these 4 sensors, as we are trying to minimize the amount of space needed for our product. The 4 holes for the sensors in the stack have a diameter of 9mm to accommodate the size of the barrel of the sensor.

Figure 3: Sensor MLX90614 Infrared thermometer module

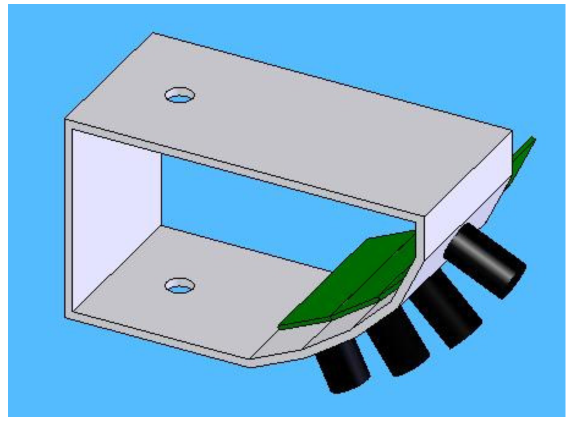

Figure 4 : Sensor stack assembly – it can hold up to 4 sensors.

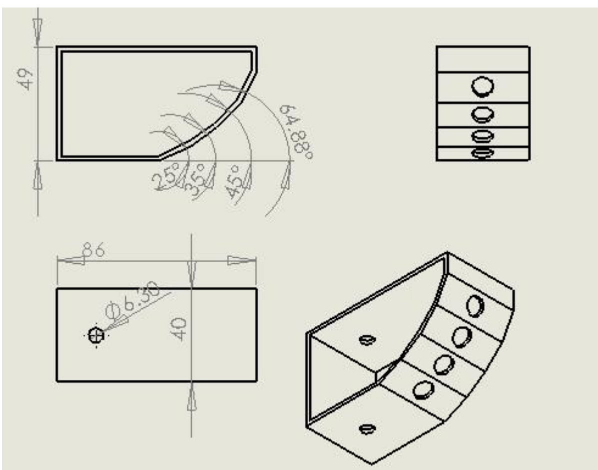

Figure 5: Dimensions for the sensor stack

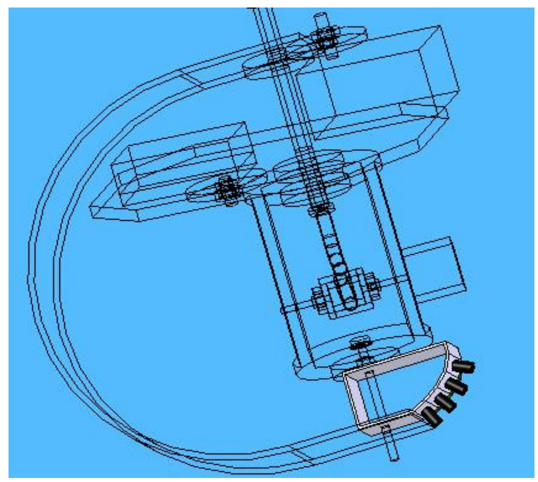

Figure 6: Sensor placement in assembly

3. Physical and Mechanical Design

For our entire physical system, Figure 2 above provides an overview of the FlexiSys design. In designing the system, we took into consideration the friction and degrees of freedom. For the construction of the system, plexiglass was a consideration, but given the brittle nature and the difficulties in joining plexiglass, wood was a better choice. The ease of obtaining wood and joining pieces together was definitely a motivation for choosing wood as our base material.

We initially considered using bearings to provide near frictionless rotation, however, we found out that the bore diameters of the bearings are made only in standard sizes. Unfortunately, our pipe is not provided in standard sizes so we decided to use two metal plates and self lubricated washers as bearings instead for our working model. The production model will utilize contract manufacturing, so standard sizes would not be an issue.

3.1 The Base

The base is a stationary part that holds the motors which rotates the sensor stack and the casing - the dimensions are shown in Figure 7 and Figure 8 below. The hole in the middle of the base plate is where the water pipe will go through.

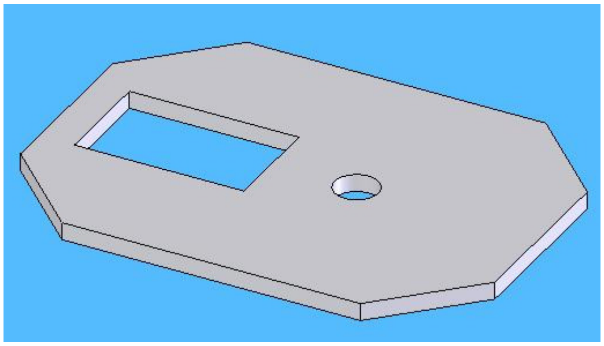

Figure 7: Base plate

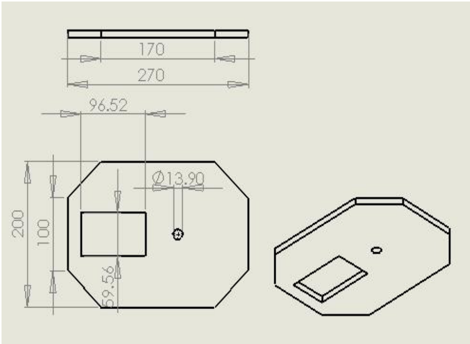

Figure 8: Base Plate dimensions

3.2 Casing

The dimensions of the casing are shown in the Table 1 below. The casing is made of wood and connected together with glue and nails. The hole on top of the center case is for the water pipe, while the bottom hole is for the shaft attached to the sensor stack as shown in Figure 9 and Figure 10 below. The casing will be able to rotate around the pipe by a motor in conjunction with self lubricated washers as illustrated in Figure 10.

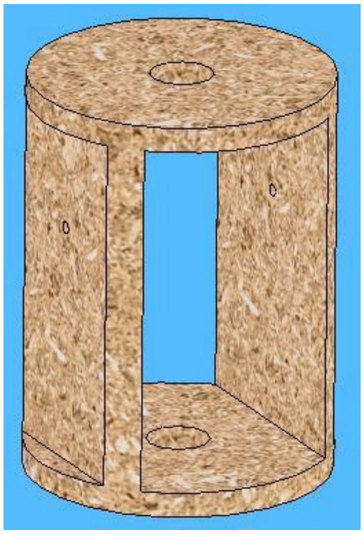

Figure 9: Center Casing

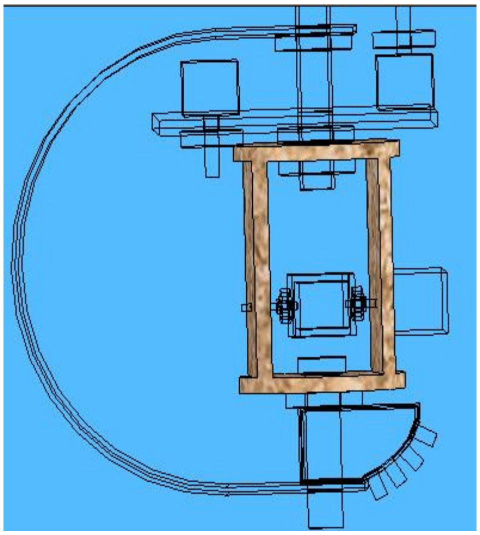

Figure 10: Casing in assembly

Table 1: Casing dimensions

	Dimensions in mm
Radius	110
Height (from top to bottom plane)	160
Top bore diameter	13.90
Bottom bore diameter	6.30
Bore diameter for DC motor	4.7625

3.3 C-shape Main Bracket

This C shaped bracket translates rotational motion from the motor to the sensor stack as illustrated in Figure 11 and Figure 12. A flexible metal is used for to create this curve as wood is generally inflexible and may crack during operation - the dimensions are shown in Figure 13.

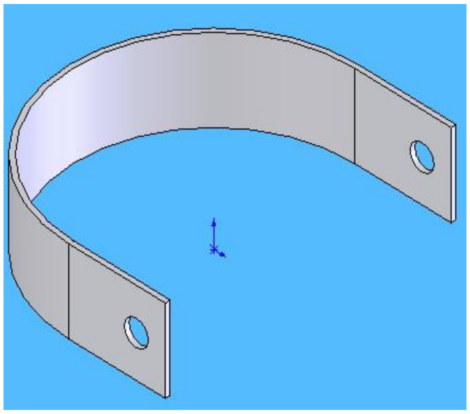

Figure 11: C Bracket Design

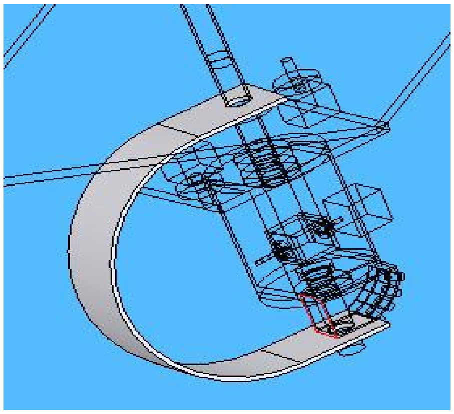

Figure 12: C Bracket in assembly

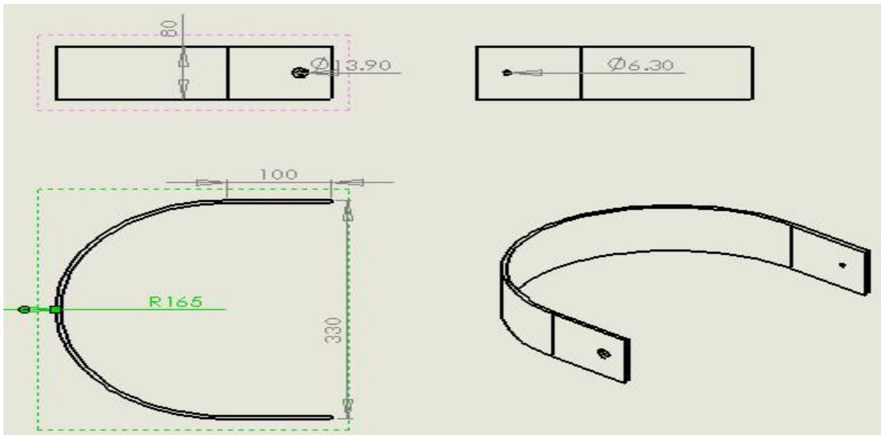

Figure 13: Bracket dimensions – top, front and isometric

3.4 Sensor stack

The following sensor stack is made with thin metal with 4 holes on the curve for the sensor cylinders, and two larger holes for a shaft. There are 4 holes on the metal to allow the sensor head to stick out as illustrated in Figure 14 and Figure 15. The sensors will be held together with silicon glue.

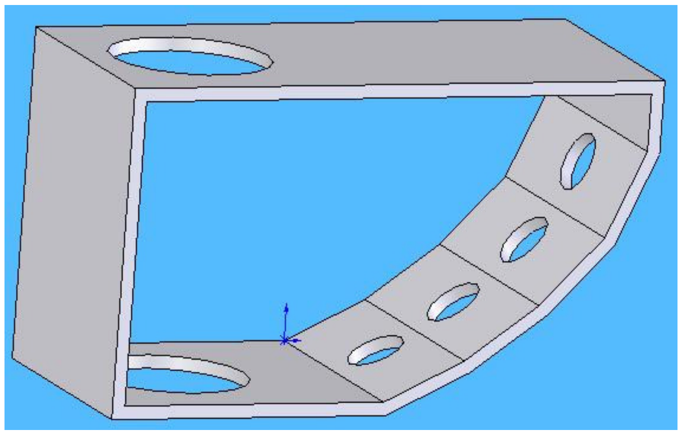

Figure 14: Sensor casing to hold sensors in place

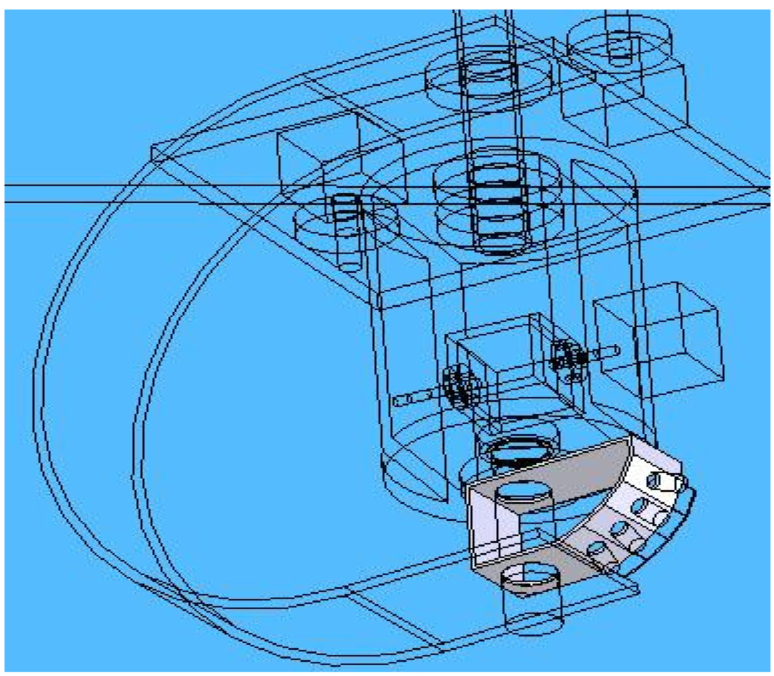

Figure 15: Sensor casing in assembly

3.5 Water Pipe

The water pipe we are using is the standard pipe currently used for sprinkler systems. The water from the city's water supply will have sufficient pressure for our system. According the *Water Pipeline Design Guidelines* [2] at Saskatchewan Environment, the maximum water pressure should not exceed 700 kPa (100 psi) to protect the water pipes. Water pipes should be designed to withstand the transient pressures caused by the water valve turning on and off. As shown in Figure 2, the weight of the case, motors, water hose, bearings and some gears is supported by the water pipe. As mentioned in overall physical properties, the total mass of all of these parts are estimated to be around 2 Kg, so the water pipe has to be strong enough to support the whole system while it is rotating. After consulting with a technical specialist at Home Depot, we decide to use a 6.4 mm steel water pipe for the system.

3.6 Water gun casing

The water hose will be fed through the water hose casing, shown in Figure 16, which will be rotated by a motor. Two clamping hubs with a diameter of 25.4mm will be connected to 25.4mm diameter rods to provide the rotational degrees of freedom. The hubs are illustrated in Figure 16 below, and the water hose casing in the whole system is shown in Figure 17. The dimensions for the casing are shown in Figure 18 below as well.

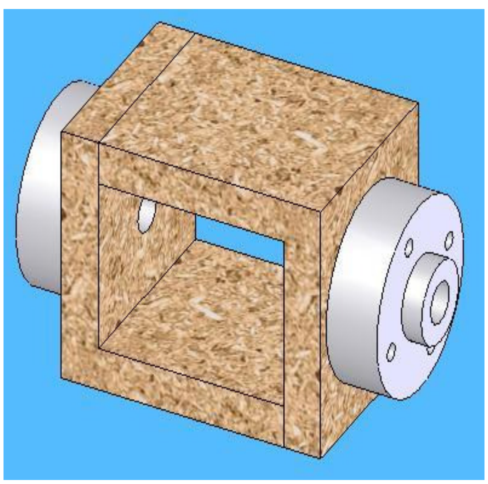

Figure 16: Water hose casing

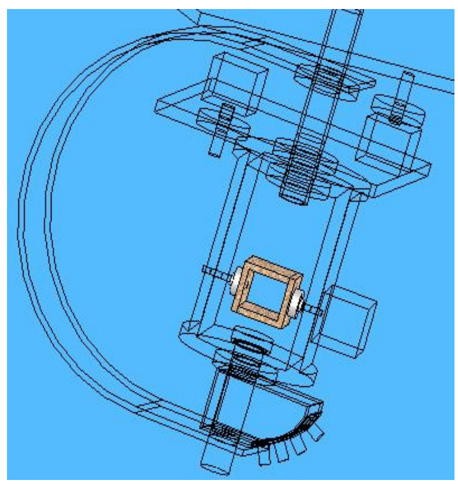

Figure 17: Water hose casing in assembly

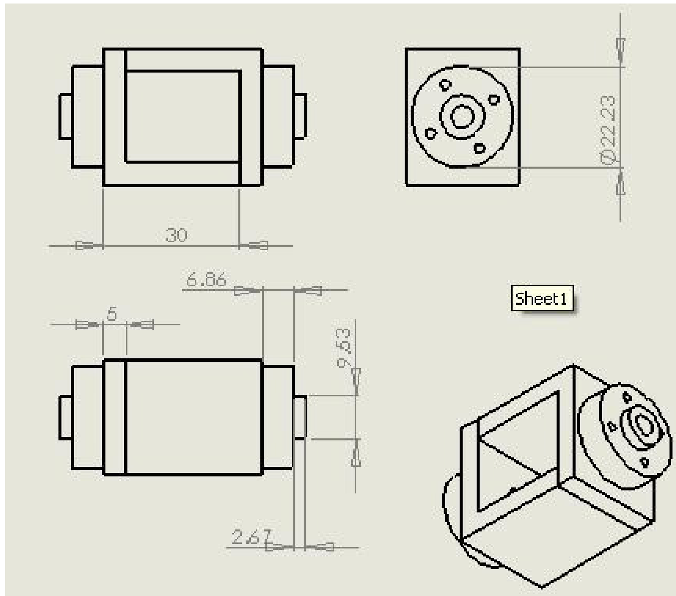

Figure 18: Water hose casing dimensions

4. Electronic System Design

4.1 Microcontroller Design

For the "brain" of the system, we will use the Arduino ATmega1280 microcontroller. Arduino is an open-source electronic prototyping platform with easy to use software and hardware in combination with an active online community. The Arduino Mega has 128kB of flash memory, 8kB of SRAM and 4kB of EEPROM. The total memory size will be plenty for the logic we plan to implement in the system. The hardware board requires a 7-12VDC supply with a maximum input current of 500mA. Another power supply option for this board is through the USB port. This low power consumption property makes this microcontroller ideal for our project.

ATmega1280 board has 54 general digital input/output pins, with 14 pins having the ability to send out Pulse Width Modulation (PWM) signals. Each digital pin operates at a maximum of 5VDC and 40mA, and can be configured to be either a digital input or output. Currently, 12 of the pins will be used in this project, with the usage listed in Table 2 below:

Table 2: List of digital pins used on ATmega1280

Devices	Pins used
Sensor Stack (2 for each sensor)	8
Solenoid Valve	1
DC Motors	2
Servo Motors	1

4.2 System logic

As illustrated by Figure 1 on page 2, the sensor readings will control the behaviour of the whole system. To ensure sensors are operating correctly when the system is started, a test reading will be performed at the initialization. After the initialization tests, the system will proceed to the fire detection loop as illustrated by flowchart in Figure 19 below:

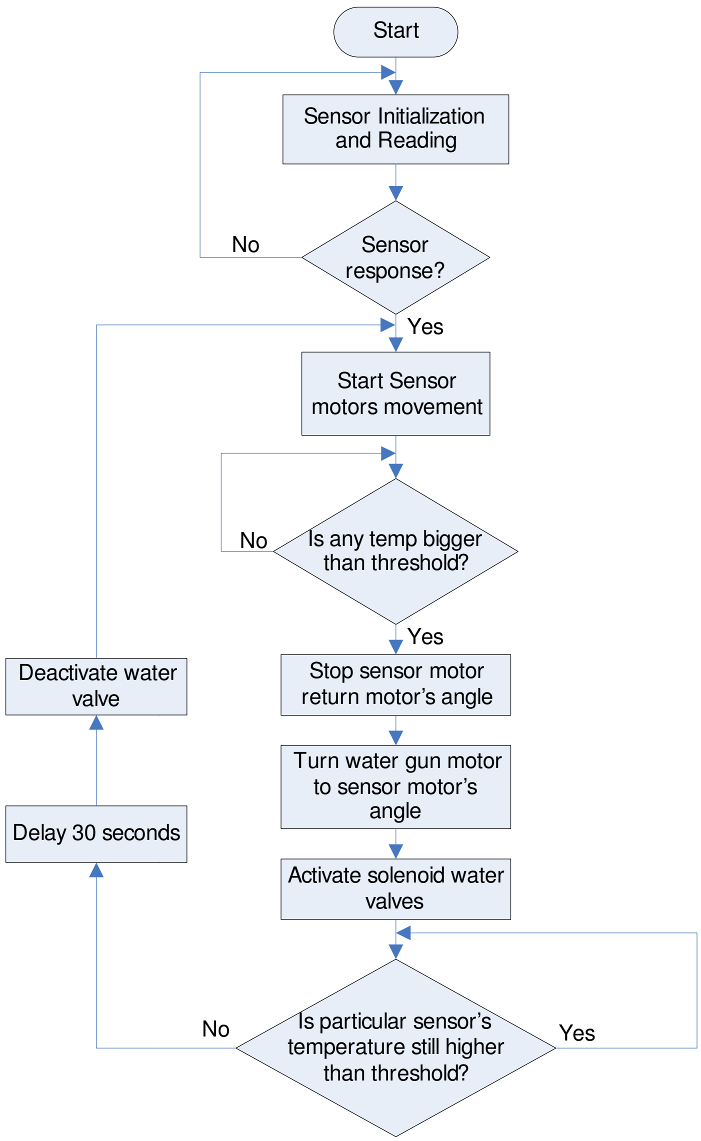

Figure 19: High Level Microcontroller Flow Chart

Once the system starts the sensor motor movement, the microcontroller will constantly check the temperature every millisecond to see if any of the readings exceed 200^{o}C. If the temperature exceeds 200^{o}C, the motor (M1) controlling the sensor stack will stop and the angle between the sensor stack and the nozzle will be sent to the microcontroller. The water gun motor will then rotate with this angle and activate the solenoid water valve. During this time, the system will still constantly check the temperature. Once the temperature drops below 100^{o}C, the system will continue spraying water for 30 seconds as a precaution before deactivating the water valve. The sensor stack will continue to rotate after these 30 seconds. Since our system is meant to be automatic, the system will continue in an infinite loop.

4.3 Thermal Sensors

Our system uses thermal sensors to act as the "eyes" of the system. The model we use is 10^{o} MLX90614 Infrared Thermometer Modules from Parallax. These are non-contact sensors that are able to pick up heat signatures in range of 10 meters. MLX90614 has a fast refresh rate of 1ms and has a low power consumption of 5 VDC and 20mA which makes this sensor ideal for our FlexiSys product.

4.4 Serial data communication with the Arduino

Reading the temperature from the MLX90614 is performed via serial command through one of the serial I/O pins on the sensor. To read the current temperature on module's RAM, the microcontroller needs to send the serial command "*!TEMR*" to the sensor I/O, which will then receive the reading on the same pin from the sensor. On our microcontroller, there is limited receive (RX) and transmit (TX) connections on board. Hence, we will use the NewSoftSerial library provided by Arduino developers to configure general digital I/O pins to output serial commands and receive serial inputs respectively. Figure 20 displays the flow chart to read temperatures from the MLX90614 modules.

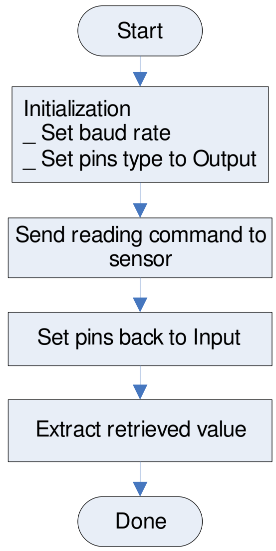

Figure 20: Sensor Initialization and Reading

The following snippet of code performs the steps of the above flow chart:

```
SoftwareSerial Temp10(2,3);    // set 2 communications pins to be 2 and 3
pinMode(2,OUTPUT);             // set pin 2 to output
 pinMode(3,OUTPUT);            // set pin 3 to output
Serial.begin(115200);          // set baud rate to 115200
Temp10.print(0,BYTE);          // send first parameter, empty buffer
 Temp10.print("!TEMc");        // send reading command
 Temp10.print(0x5A,BYTE);      // send sensor address
 Temp10.print(7,BYTE);         // send memory address of reading
pinMode(2,INPUT);              // set pin 2 back to input to catch upper bits
 pinMode(3,INPUT);             // set pin 3 back to input to catch lower bits
```

Pin 2 and 3 will receive binary values of temperature counts. According to the Parallax specification of MLX90614 [7], these counts are taken at rate of 0.02 counts/Kelvin. To convert degrees of Kelvin to Celsius, we perform the following:

```
 // shift upper byte and retrieve whole number
wholeBinNum = lowerByte + upperByte*256;

// convert back to Celcius degree
Temp = (wholeBinNum*0.02) – 273;
```

4.5 Solenoid Water Valve

To enable the water flow, we use the solenoid water valve BBTF-CD-12VDC from ECHOTECH. The valve is normally closed; however, when current is fed to the solenoid, the valve will open. This solenoid valve will provide our extinguishing system with a fast and reliable response. This is one key aspect of the overall product.

The BBTF-CD-12VDC has a maximum pressure limit of 140 psi and requires 12 VDC and 500mA. Since our microcontroller digital pin can only send out a maximum of 5 VDC and 50mA, we need an external power source. A picture of our solenoid valve is displayed in below Figure 21 simple circuit for valve's external power source is shown in Figure 22 below:

Figure 21: Solenoid Valve

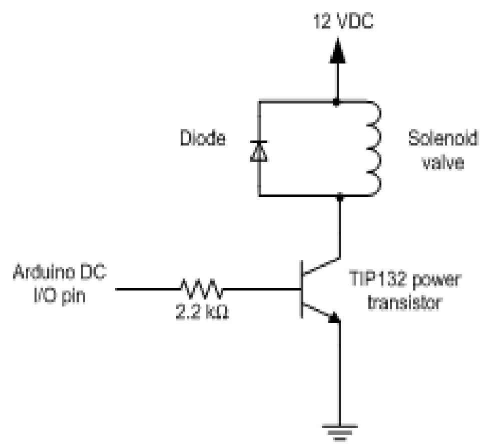

Figure 22: External Power Circuit for Solenoid Valve

When the Arduino DC I/O pin set to high at 5 VDC, the transistor is on and allows current through solenoid valve to turn it on. When DC I/O pin is low, transistor is cut off and does not allow current through the circuit. Hence, the valve is closed. Since our solenoid valve requires a high current supply of 500mA, we need to use the power transistor TIP132 which allows a maximum current of 4A through. In addition, when transistor is cut off suddenly, the solenoid can create magnetic field that creates induced voltage which damage our solenoid valve. Therefore, a protection diode is connected backward across the solenoid which will limit the induced voltage and protect the transistor [8].

4.6 Motor design

4.6.1 Servo Motors

For the water gun movements and sensor movements, we decided to use servo motors SPG425A. The servo motor controlling the sensor stack will give us angle feedback and allow our system to precisely locate the area of fire and coordinate the water gun motor to the same angle. SPG425A is a multi-rotational motor that allows sweeps of 360^{o} and more. The motor is able to create 5.43 N-m torque with maximum rotation speed at 0.32 sec/60^{o}. The maximum operating voltage for these motors is 6V.

To control the servo motor through the Arduino board, we will use "Servo.h" library provided by Arduino developer, and to rotate the motors to certain angle, the following Arduino command will be used:

myservo.write(pos); *// rotate myservo to angle pos*

where: myservo: object of servo class
pos: Angle of rotation

4.6.2 DC Motors

To ensure a large area of coverage for our extinguishing system, an extra DC motor is placed for vertical rotation. The DC motor we have requires a supply of 6V and 30mA. To control the DC motor through the Arduino board, a circuit for external power source is built very similar to external power source for the solenoid water valve. The schematic for the DC motor circuit is shown in Figure 23 below.

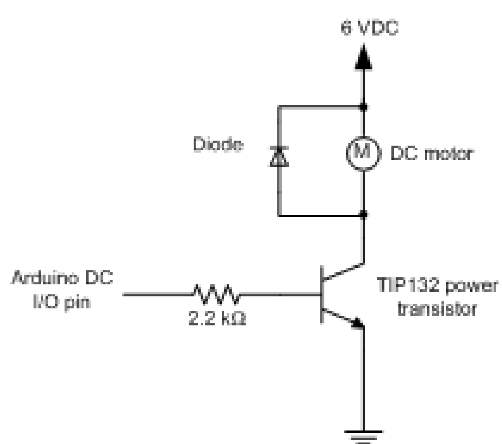

Figure 23: External Power Circuit for a DC Motor

5. System Test Plan

5.1 Unit Testing

5.1.1 Sensor

- Turn one sensor on at a time to ensure each sensor will be able to accurately detect the temperature.
- Turn on all sensors and ensure all sensors are able to detect accurate temperatures.
- Create different temperatures at each sensor and ensure the sensor stack is able to locate the fire.
- Verify that the sensors are stable for long time and able to maintain accuracy (at least 3 hours).

5.1.2 Solenoid Valve

- Verify that valve will not leak water when closed.
- Verify that water will flow through when the valve is turned on.
- Verify that the valve will open within 0.5 seconds after sending signal.

5.1.3 Motors

- Verify that the motors are able to rotate to the intended angle when a signal sent.
- Verify that the motors have enough torque to carry water gun and sensor stack.
- Verify that the motors do not draw more current than intended.

5.2 No Fire Scenario

Conditions: No fire appears within the monitored space.

Expected Observations: The sensor stack must rotate 360° back and forth constantly. The sensor stack completes each 360° rotation in 5 seconds. During this time, the case and the nozzle should be stationary.

5.3 Fire scenario

Conditions: Fire of at least 20cm in height and width appear randomly within 3 meters of the system.

Expected Observations: System stops rotating with sensor pointing in the direction of the fire. Water gun then rotates to the same direction as the sensor and starts shooting water. Once the flame is extinguished, the water gun continues shooting for 30 seconds and then deactivates. The system then resumes to normal sensor sweeping.

6. Conclusion

These proposed design specifications provide details that will fully satisfy the functional requirements of the working model of the Flame Extinguishing Intelligent System. As the development of the model progresses, these design specifications will be the foundation for each component and integration. Further optimizations will be discussed and revisions will be completed when members unanimously agree. In addition, test plans have been included to ensure the safety and robustness of our system. Development has been progressing and our system is on track to be completed by April 15, 2010.

References

[1] FlexiSys, “Functional Specifications for a Flame Extinguishing Intelligent System”, Simon Fraser University, Burnaby, BC, Canada, February 2010.

[2] Saskatchewan Environment, *Water Pipeline Design Guidelines,* 2004. [Online]. Available: http://www.saskh2o.ca/DWBinder/EPB276WaterPipelineDesignGuidelines.pdf. [Accessed: March 04,2010]

[3] Servocity.com, *Set Screw Hubs*, 2008. [Online]. Available: http://www.servocity.com/html/set_screw_hubs.html. [Accessed: March 08, 2010]

[4] Servocity.com, *48 Pitch Aluminum Hub Gears (.250" Face),* 2008. [Online]. Available: http://www.servocity.com/html/48_pitch_aluminum_hub_gears___.html. [Accessed: March 08, 2010]

[5] Servocity.com, *SPG425A-360 360^o Rotation*, 2008. [Online]. Available: http://www.servocity.com/html/spg425a-360_360o_rotation.html. [Accessed: March 08, 2010]

[6] Servocity.com, *SPG425A-BM-360 360^o Rotation*, 2008. [Online]. Available: http://www.servocity.com/html/spg425a-bm-360_360o_rotation.html. [Accessed: March 08, 2010]

[7] Parallax.com, *MLX90614 Infrared Thermometer Modules*, 2008. [Online]. Available: http://www.parallax.com/Portals/0/Downloads/docs/prod/sens/28040-28042-IRThermometer-v2.0.pdf. [Accessed: March 09, 2010]

[8] Kpsec.freeuk.com, *Transistor Circuits*, 2010. [Online]. Available: http://www.kpsec.freeuk.com/trancirc.htm. [Accessed: March 09, 2010]

[9] Arduino.cc, *Arduino Mega*, 2009. [Online]. Available: http://arduino.cc/en/Main/ArduinoBoardMega?action=diff. [Accessed: March 08, 2010]

[10] Arduiniana.org, *NewSoftSerial*, 2010. [Online]. Available: http://arduiniana.org/libraries/NewSoftSerial/. [Accessed: March 05, 2010]

[11] Arduino.cc, *Sweep*, 2009. [Online]. Available: http://arduino.cc/en/Tutorial/Sweep. [Accessed: March 01, 2010]

[12] Letsmakerobots.com, *DC motor control with Arduino*, 2009. [Online]. Avaliable: http://letsmakerobots.com/node/7990. [Accessed: March 10, 2010]