RefriECO **Design Specifications for the DualCooler**

March 13, 2014

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

Re: ENSC 440/305W Design Specifications for DualCooler

Dear Dr. Rawicz,

On behalf of our team, I am enclosing the Design Specifications for our ENSC 440 project. Our product makes refrigerators more energy efficient than what is available on the market at the present time. The technology will utilize the naturally cooler temperatures, of places with winter temperatures that can dip below 5°C, to cool refrigerators.

The attached design specification provides details regarding the design choices we made to build our eco-friendly and energy efficient refrigeration system. RefriECO will make sure that all of the components such as the microcontroller, temperature sensors, motors, ducts, filters, fans and dampers will meet the minimum functional requirements in order to conclude with a final product that will maintain safety and reliability on the long run. We will also follow the test plan that we have attached in this document to test the individual components as well as the integrated system after the final product has been assembled.

Our team is composed of five talented, creative and determined engineering students: Abantika Oishee, Gonsakar Gunasingam, Allan Vincent, Hasan Syed and myself. If you have any questions or concerns, please feel free to contact me by email at rravi@sfu.ca.

Sincerely,

Ranjita Ravi

Ranjita Ravi

Chief Executive Officer (CEO), RefriECO

Enclosure: Design Specifications for DualCooler

RefriECO

DualCooler Refrigeration System

A more environmentally friendly Refrigeration System

Project Team: Ranjita Ravi
Allan Vincent
Gonsakar Gunasingam
Abantika Oishee
Hasan Syed

Contact Person: Ranjita Ravi
(rravi@sfu.ca)

Submitted To: Dr. Andrew Rawicz
Steve Whitmore
School of Engineering Science
Simon Fraser University

Issued: March 13, 2014

Revision: 1.5

Table of Contents

List of Figures

List of Tables

Abstract

RefriECO's DualCooler is an innovative new idea to further curb the energy savings of a refrigerator by installing a system of ducts through a building and routing cold air from the outside into the fridge. The purpose of this design specifications document is to outline and describe in detail the parts and functionalities of RefriECO's DualCooler System. Due to the mechanical, electrical and electronic disciplines required for implementing and designing this refrigeration system, the concepts and theories of thermodynamics, fluid dynamics, electronics and electrical circuits had to be implemented. The circulation of air through the ducts and through the fridge require careful considerations of fluid dynamics and thermodynamics. Everything from pressure to air flow to air temperature through the system is taken into consideration. As for the control systems that control the PWM servo motors, PWM fans and compressor motor require in-depth and thorough descriptions of adjacent circuits that are needed to power the electronic components and supply signals.

1. Introduction

RefriECO's DualCooler is an alternative to existing refrigeration systems such as the ever popular refrigeration cycle (using compressors, condensers, evaporators) and thermoelectric cooling systems. The system will utilize the cold outside air in parts of the world that are commonly subjected to much colder temperatures. The idea is derived off geothermal energy heating systems and based on HVAC central air conditioning systems. It will require a system of ducts through a building or structure that using fans and dampers can circulate and deliver cold air to refrigerators.

Our innovative idea will be demonstrated using a prototype that will be created using a mini fridge. Although this system assumes that buildings already have the required ducting in place, for this demonstration we will require insulated ducts, grills and fans that connect to a test environment. The DualCooler Refrigeration System, that the consumer will purchase, will consist of dampers (controlled by servo motors), filters, relays (to control the compressor) and LCDs to program the input temperature as well as display the actual external and internal temperature.

1.1. Scope

All the design specifications in this document will cover details and justification for most of the functional specifications allotted for this report. It covers the installation, replacement and functions of the refrigeration system for each of its main parts as well as the general specifications.

1.2. Intended Audience

The intended audience for this report are the engineers at RefriECO and the DualCooler team. It provides the in-depth design specifications of each part and will also provide guidelines for the development of this project. This report is also intended to provide information regarding the functions of the refrigeration system for the consumers of refrigerators, with the system implemented. It will also be used to create a test plan in the design specifications for the prototype.

2. General System Overview

2.1. Prototype Model

The Figure 1 and 2 below shows what the completed prototype must resemble, from the front as well as the back. The front portion will house the LCD and buttons which will be the user interface that will allow the user to customize the internal temperature of the fridge. On the back, in the small component section, there is plenty of space to fit all of our electronic and electrical components in an enclosure, to keep the wires out of the users and easy access to the compressor. The servo motors will have to fit above and outside the duct.

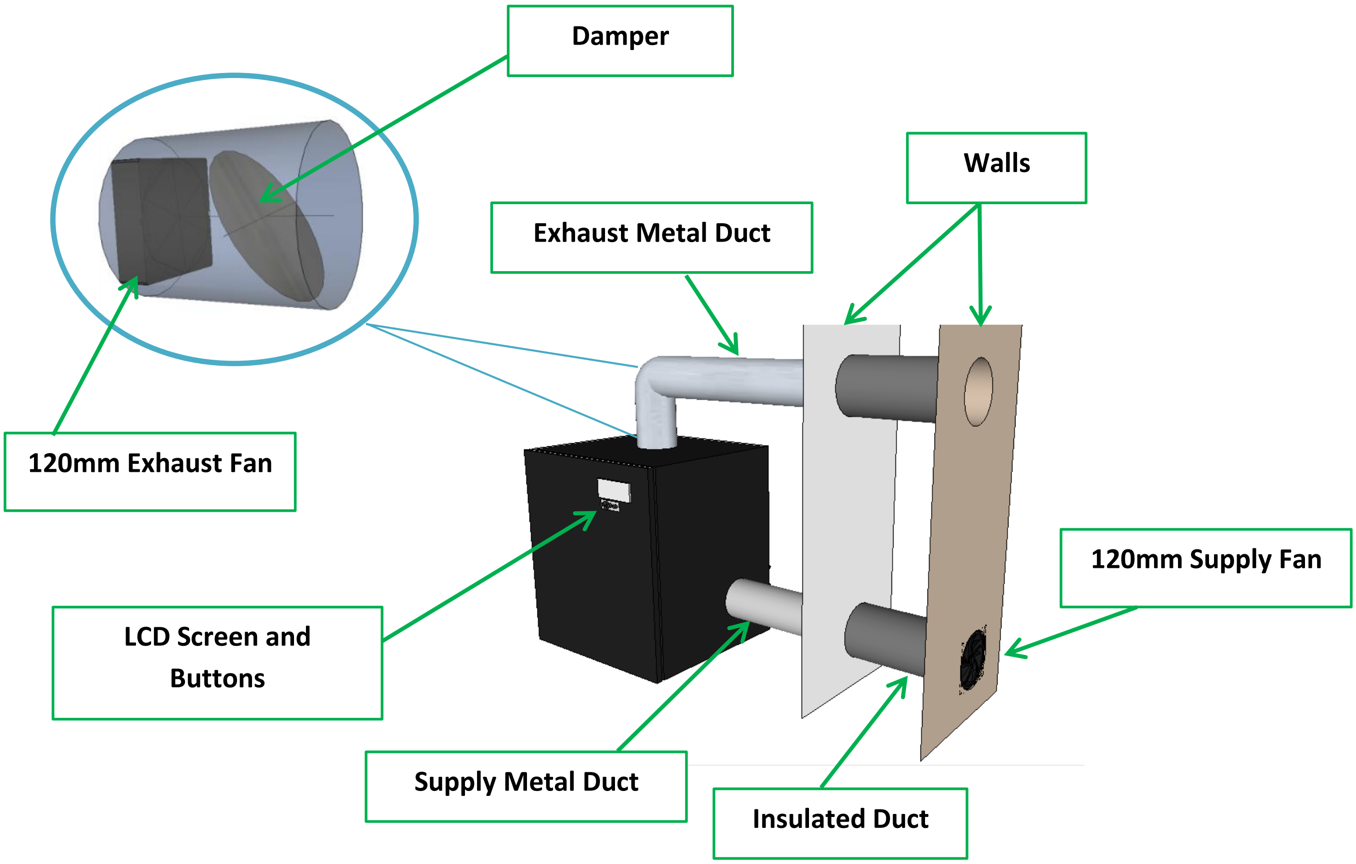

Figure 1: Front View of DualCooler Refrigeration System Prototype

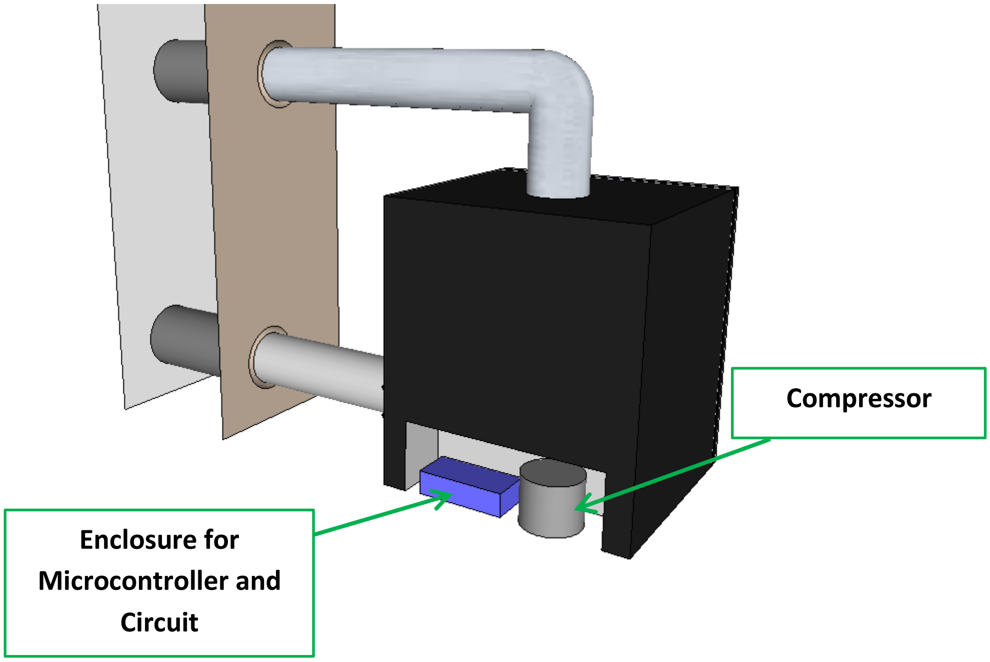

Figure 2: Back View of DualCooler Prototype

The placement of the fans is slightly more complicated. The supply fan needs to be closer to the exterior of the building to pull in the cold air, whereas the exhaust fan needs to be placed in the ducting right above the fridge (for our prototype) in order to pull the warm air from inside the fridge into the duct to be vented. The exhaust fan comes with the refrigeration portion that is sold with the fridge. The supply will be installed in the insulated ducts close to the outside of the building. However, both dampers will be positioned right beside the fridge and will be provided in the retail refrigeration system.

2.2. System Block Diagram and Flowchart

Figure 3 below is an updated block diagram that summarizes all of the inputs and outputs this system will have. Note the greyed out input blocks that represent the pressure sensors and humidity sensors, will be implemented if time permits or for post development plans. As drawn out, the microcontroller will oversee and control all cooling related operations of the fridge.

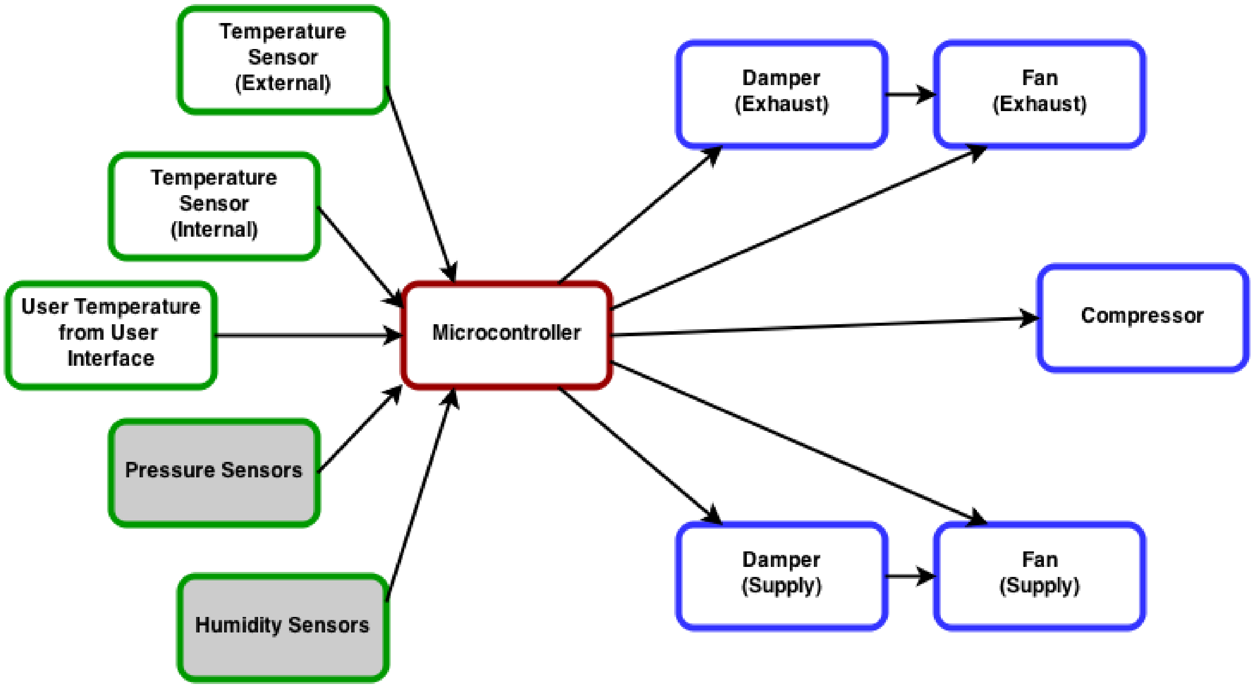

Figure 3: Block Diagram of Microcontroller Interface

The flowchart in Figure 4 on the next page, goes in-depth into explaining just how the block diagram translates into functions and how the software aspect will process the inputs and produce various outputs. We have to take in three main parameters for comparison; the temperature that the user chooses the refrigerator to be at, the temperature of the temperature outside the building (external) and the internal refrigerator temperature. The processor can also just keep polling until the desired temperature is reached. However, this uses more power and it would be wiser to take multiple measurements and estimate the ideal wait period for best results.

Flowchart Legend:

U: Desired Temperature (selected by the user)

I: Internal Refrigerator Temperature

E: External Environmental Temperature (outside temperature)

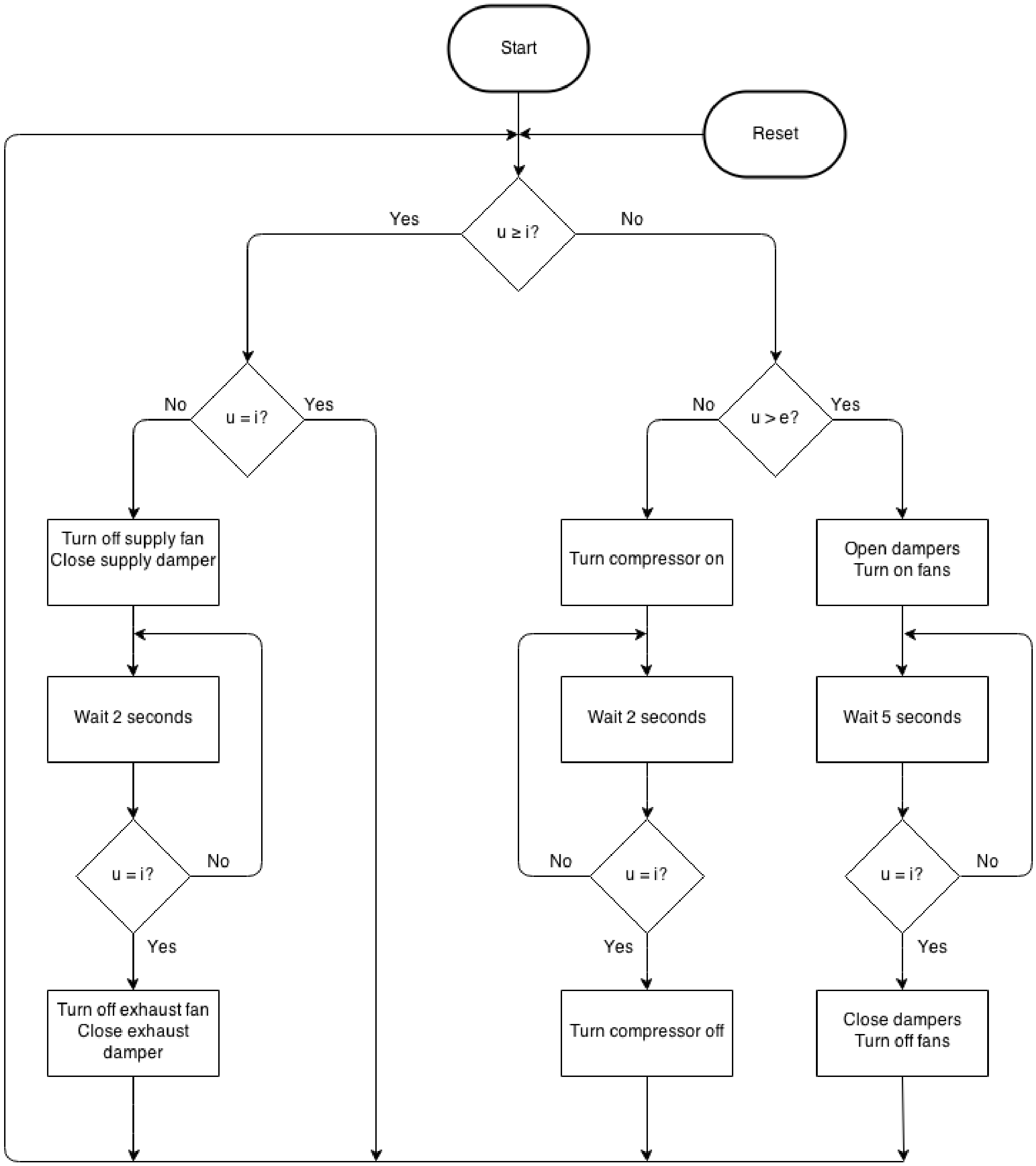

Figure 4: Flowchart of DualCooler Refrigeration System

Listed below in Table 1, is a quick reference table of parts we require for the project. In this section we will briefly describe all the parts and then thoroughly describe the specifications for each part throughout the document.

Table 1: Bill of Major Parts of DualCooler

Parts	Number Required	Purpose
Dampers	2	To close or open the system depending on the condition
Servo Motors	2	Control the position of the damper (open and close)
Temperature Sensor	2	External temperature versus Internal temperature
Fans	2	Promote Exhaust and Supply air flow
LCD Display	1	User interface for setting temperature and displaying outside and inside temperatures
Pushbuttons	5	User interface for setting temperature and toggling through menus
Insulated Ducts	2	Exhaust and Supply ducting through the test environment
Metal Duct	2	Exhaust and Supply ducting that come with the DualCooler Refrigeration System
Microcontroller	1	Controls and processes the array of inputs and sends appropriate output signals.

2.3. Power Overview Table

One of the biggest challenges and risks in this project is the fact that we need to conserve as much energy as possible using this Refrigeration System process. Table 2 below list all the components and the operational voltage, current and power to be able to display a quick estimate of total power consumption for the system.

Table 2: Table of Total Power Consumed by components of DualCooler

	Operational Voltage (V)	Operational current (A)	Power consumption (W)
Supply and exhaust fans	12	0.14 * 2	3.36
Servo motors	5	0.02 * 2	0.2
Temperature sensors	5	0.0015 * 2	0.015
Arduino mc	12	0.250 [1]	3
LCD	5	0.003	0.015
Total			**6.59**

For comparison, the power consumed by a compressor motor is:

*120V *5A* = **1800W**

3. Duct Design Specifications

3.1. Types of Ducts

The duct is considered a static component of the installation through which air flows in the building, connecting all part of the system by either supplying the air or exhausting the air (discharge). Air ducts are the elements where air is distributed across the units using air processing unit or a diffuser unit. The main advantage of using a duct is for quiet operation of air circulation or in words for acoustic purposes. There are three types of ducts available in the market, which are metal, plastic and flexible ducts.

For our refrigeration system, we are using a 5 inch metal (galvanized metal) duct [R3.10-I, R3.7-I]. Since metal is a good thermal conductor, such ducts require thermal insulation which wrapped around the outer duct wall. We are using aluminum and foil pipe insulation wrap to wrap the sheet metal. We considered the flexible ducts as well for the design, while they tend not to last long as sheet metal and also they are cost effective [R3.2-I, R3.6-III]. The shape and size will be adjusted according to the location in sheet metal and not possible to do it with the flexible ducts [R3.2-I]. Minimum thickness of the thermal duct insulation reduces the danger of condensation. Energy losses mainly caused by the air leakage in the duct work joints, we are considering a possible way to reduce the amount of the joint work in ducts to reduce the energy loss and also by sealing the joints will prevent the leakage as well [R3.11-I] [14].

Figure 5: Duct Types: (Left to Right) Flexible duct, semi-flexible, sheet metal (using for our system)

Our system contains two ducts, supply and an exhaust the air ducts; we are using a well-insulated 5inch diameter sheet metal duct with 40 inch long and the cross section of 5" * 30". In the supply duct we are inserting:

- Supply fan (120mm) - Blow the outside air in to the ducts
- Damper (5") - passes the air flow through to the internal structure of the fridge or stops the air.
- Air Filter - Filter the contaminated air

We chose our fan based on the highest and most optimized fan for our system based on various calculations and CFM values. Also there are only standard sizes for computer fans in the market. The fan that works best for us is a 120mm (4.72 in) fan which would only fit in a 5" duct. Damper sizes followed suit with the duct size.

3.2. Air Quality

To get rid of the contaminated air from the environment into the refrigeration system we are using a Honeywell odor air filter which is normally used for air purifiers. The filter located near the supply fan to ensure the easy maintenance procedure in the future. The filter system needs to be changed span time of 2 years of time period. Not only does it filter out odor but other small allergens and particles.

Figure 6: Honeywell Air Purifier Filters

3.3. Heat transfer in ducts

Heat loss calculated by suing the heat transfer per unit area/ length in duct surface per unit temperature difference (˚C)[15]

$$Q_e = U.P.L(\frac{t_e + t_i}{2} - t_a)/1000 \qquad \text{Equation 1}$$

Here,

U = Total heat transfer rate for duct wall (W/m)
P = Duct perimeter (m)

L = Duct Length (m)
Q_e =Heat Variation through the ducts (W)
t_e = Air temperature in the duct (inflow) (˚C)
t_i = Air temperature in the duct (outflow) (˚C)
t_a = Air temperature of surrounding air (˚C)

Using the above equation 1 for our supply and exhaust duct:

Table 3: Calculated Values for Heat Variation Estimation

Cross section Area of the duct (πr^2)	π*2.52 = 19.64 inches2 (0.0126m2)
Length of the duct	40 inches (1.016m)
Assuming Airflow Temperature in the inflow Duct	5 degree Celsius
Air flow Temperature in the surrounding air	16 degree Celsius
U of the Galvanized sheet metal duct	3.8 W/m
Therefore, Q (Heat variation through Duct)	6.811 W

Therefore, ***Q (heat variation through ducts) = 315 W***

3.4. Air pressure

Flow of air or any other fluid occurs because of the pressure difference between the two points. Total pressure in the ductwork always drops in the direction of the air flow. In our design pressure drops also occurs to the physical components such as filters and dampers. Main concern in our design is the static pressure because the fan blows the air in to a closed duct.

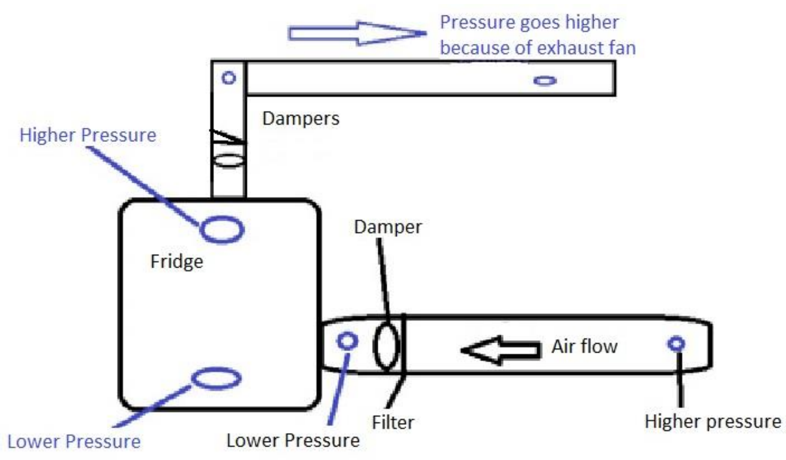

Figure 7: Air Pressure and flow diagram for the DualCooler

Figure 7 above shows the pressure differences at different points in our refrigeration system. It's clear that pressure decreases because of the components and the direction of the flow. Also another interesting pressure difference occurs when we open the fridge door and close them, the pressure difference occurs because of the outside temperature (room temperature) or warm air mix up with the cold air inside the fridge which makes the pressure difference and the warmer air will be sucked by the exhaust fan to keep the pressure inside the fridge at a constant rate.

4. Damper Motor Design Specifications

4.1. Design of Servo Motors

There are several purposes to our servo motors. One of the purposes is to be able to completely seal the air inside the fridge to maintain the desired temperature. Another function is to be able to open the dampers completely so that there is no air resistance when the fans are supplying/ exhausting the air.

4.2. Design Specifications

We needed our servo motors to turn at least 90 degrees in angle to completely close and have no air resistance [R7.1-I]. One of the main requirements of the servos is to have enough static torques to be able to seal the fridge door. It needs to be able to work efficiently in cool temperatures and consume minimum energy [R7.4-I], [R7.7-I]. The motor will not be placed inside the duct but on the outside therefore not coming into contact with any moisture [R7.8-I].

Our specifications match the details of SKU: LS-0006 servo motors to use for our project. Its operating voltage between 4.5V to 6V, this means that our servos can be powered directly by the microcontroller [R7.5-I]. Our selected servo motors have a torque of 1.2 kg/cm [12]. Our ducts have a radius of 2.5 inches, (or 6.35 cm). Which means it will be applying a force of 0.2 kg at the seals of the damper [R7.6-I]. For the dynamic torque calculations, we would need the mass of the damper plate for the moment calculations and a good estimate of the friction forces. The mass of the damper plates is 100 grams and the moment to turn it is 0.6 kg-cm. The rest of the torque would be used as applied torque for the turning motion. A detailed overview of the servo specifications is presented in Table 4 [R7.3-I] [12].

Table 4: Servo Design Specifications

Specifications	Servo Dimensions/Design
Rotation	90 degrees
Voltage Range	4.5-6V
Speed @ 4.5V; @6V	0.12; 0.1 sec/60˚
Torque @ 4.5V; 6V	1.0; 1.2 kg/cm
Dimensions	21.5 x 11.8 x 22.7 mm
Weight	6g

4.3. Energy consumed by the motors

The motors will be supplied with 5V from the microcontroller and use an approximate of 0.02 amps. The motors would be closing or opening a maximum of once in 10 minutes and turning 90 degrees. The motor specs mention that the motor turns at 0.1 sec/60 degrees. This means, our motors will be running for 0.15 seconds at each turn. The motors will be running for approximately 1 second for every hour. Consuming 0.00003 Wh energy in each turn by equation 2.

$\text{Energy in an hour}_{(Wh)} = \text{Power}_{(W)} \times \text{time}_{(hr)}$ Equation 2
= (5V)*(0.02 amps) × (1/3600)= 1/36000 Wh

Energy in a month= (730)(1/36000)= 0.02 Wh Equation 3

Since we are using two servo motors for our project, the energy would accumulate to 0.04 Wh in a month.

5. Fan Design Specifications

(Both Supply and Exhaust)

5.1. Fan Specifications Overview

The fans will be in charge of supplying cold air into the fridge and also of releasing the hot air from the fridge. The fan will be positioned in a way so that the maximum amount of air flow will be directed into the fridge through the ducts and exhaust the warm air from the fridge rapidly. The supply fan will be positioned at one end of the supply duct towards the outside of the house in order to bring in the cold air from the outside. Appropriate safety measurements will be taken while placing the fan at this position such as not letting it come into contact with harsh weather such as rain and snow etc. The exhaust fan will be positioned on top of the fridge to provide proper exhaust of hot air to the outside. As hot air rises, positioning the fan on top of the fridge and not inside the exhaust duct would be the sensible option.

5.2. Type and Size requirements

Fans come in a variety of types, shapes and sizes. The size and the type of the fan required for the mini refrigeration system had to be decided based on the total power consumption of the fans, the speed of the fans as well as the diameter of the fans that would fit perfectly into the duct. For this particular refrigeration system, the only suitable types of fans that we could use were the 5" In-Line Duct fan and a regular PC cooling fan. A 5" In-Line fan had a large power consumption due to the fact that it ran on the regular 120 V power outlet and required about 0.75 Amps therefore resulting in a total power consumption of about 85W in total. The 5" duct fan also created noise levels of up to 57 dB which could be compared the noise level of having a normal conversation at home. This defeated the purpose of the refrigeration system since our focus was to implement the design using as little power and noise as possible [3]. The 5"Duct fan also cost significantly higher than a 12V DC cooling fan and did not fit the total budget cost. Although the 5" In-Line duct fan had a much higher CFM than a 12V DC cooling fan and would result in the refrigeration system being cooled at a much faster rate due to the higher air flow, we decided to implement our refrigeration system using two 12V DC PC fans due to the eco-friendly design concept as well as lower noise level system acceptable for commercial purposes.

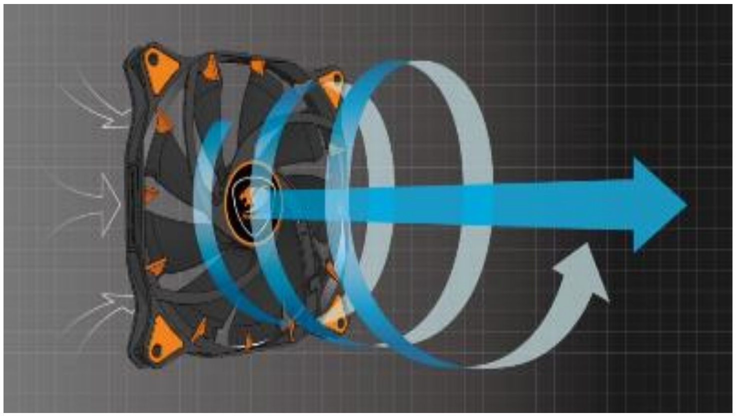

Figure 8: Aiflow is focused in a strong straight direction [2]

Figure 9: Minimum Dispersion to the surroundings [2]

5.3. Design Specifications

The diameter of the supply duct we will be using for the refrigeration system will have a diameter of 5 inches and hence a 120mm (approximately 4.7 inches) fan which will be used inside the duct to supply the air into the fridge [R4.2-I]. The fan that will be used for the supply and exhaust air will be a 120mm 12 V DC fan with a speed of 800 – 1500 RPM. With the help of the diversion blade design, the airflow gets directed in a strong focused direction with minimal dispersion to the surrounding therefore maintaining an optimal cooling efficiency. With the help of these blades, the fan is able to produce up to 70 CFM at its maximum RPM (Rotation per Minute) which is 1500 RPM [R4.1-I]. The blades on the fan are plastic and therefore eliminating the possibility of corrosion on the blades [R4.5-I]. The fan is designed with anti-vibration pads and an aerodynamic design which minimizes the noise levels of the fan and maintaining the noise level to about 17.9dB at the maximum fan speed [R4.6-I][4]. This noise level in comparison is lower than that of the noise level at which a person whispers. The fan needs a constant voltage of 12 volts with a minimum ampere of 0.01 Amps to run at its minimum rotation speed. At the maximum rotation speed it uses up to 0.15 Amps. This results in the fan total power consumption at the minimum rotation speed to be 0.12W and at the maximum rotation speed to be 1.8W [R4.3-I]. Due to the limitations of the micro controller only being able to provide an output voltage of 5V, an external power source will be used to provide power to the supply and exhaust fans. The microcontroller itself will also be powered using an external power supply.

5.4. PWM Applications

The 120 mm fan we are using contains four input pins each representing different functions. One of the most important features of the 12V DC fan is the PWM Signal. PWM stands for Pulse Width Modulation. PWM signal is a digital square wave that can change the duty cycle of the signal by controlling the time that the signal stays on with the frequency of the signal being a constant. This feature enables us to change speed of the fan and control the amount of air being directed into and outside the fridge. Also, manipulating the signal can either increase or decrease the RPM accordingly. The example of the PWM signal is shown in the Figure 10 below

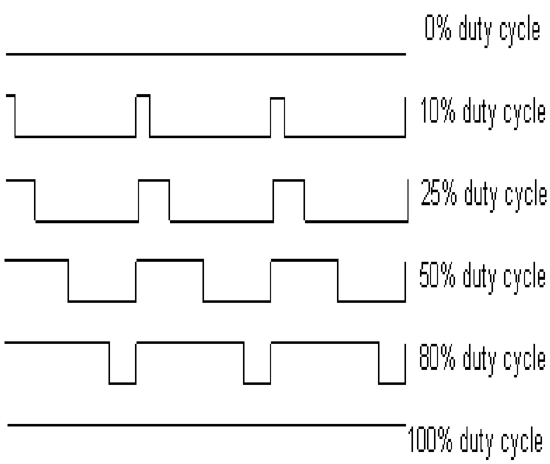

Figure 10: Fan Duty Cycle waveforms

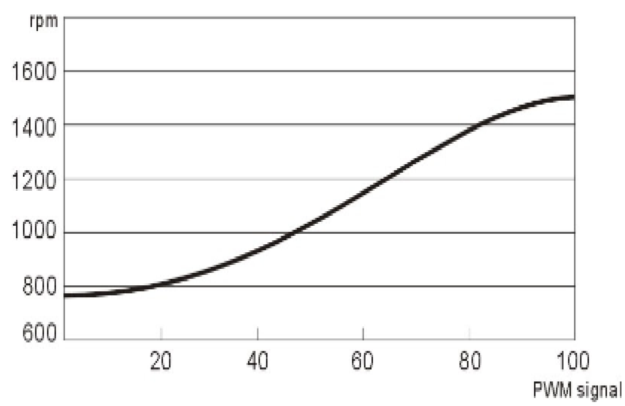

Figure 11: Relationship between PWM signal and RPM

Since the fan requires a minimum RPM to run, the 0% duty cycle would refer to the fan running at its minimum speed. At 100% duty cycle, the fan runs at its full speed and is on 100% of the time. Since the signal being sent to the fan is a digital signal, there is no additional heat being generated. As the 0% duty cycle refers to the lowest number of RPM this DC fan could run, which is at 800 RPM, the speed of the fan can be increased in a linear manner to achieve its highest RPM at 100%. This can be seen in Figure 11.The PWM feature will be used in the refrigeration system to control the amount of supply air that will be brought into the system and to control the amount of exhaust air to be released from the system.

Depending on the temperature outside, it would be necessary to bring the speed of the fan down or up in order to speed up the cooling of temperature inside the fridge. If the temperature outside is extremely low, then the fan would not have to run at full speed and could run at its minimum speed which results in extremely low power usage, and at higher temperatures the fan could run at a higher speed to reduce the time required to cool the refrigeration system. This feature could also be used to exhaust the cold air depending on how warm the air inside the fridge is. If the air inside the fridge is warm, then the RPM could be increased to release that air at a sooner rate and if it's manageable, then the RPM could be used at a slower rate.

Although the PWM feature of the fan has the ability to bring down the speed of the fan to its minimum RPM at 0% duty cycle, our refrigeration system needs to fulfill the condition of the fan turning off when the temperature inside the refrigeration system has been achieved as required by the user. In order to achieve this, we will have to create a circuit using a BJT which would act as a switch to turn the fan on or off. The input to the BJT would be the signal being sent from the Arduino Microcontroller which would then turn on the Fan as required. When the microcontroller does not send a signal through its digital pin to the BJT, the fan will not work, therefore fulfilling our requirement. The circuit diagram can be seen in the Figure below. The diode in parallel to the Fan is used to protect the circuit from having a reverse current. Having the diode protects the fan and the circuit in case high spike voltage is generated. The diode does not let a current pass through if this spike occurs [1]. Figure below shows the circuit diagram to turn the fan on or off.

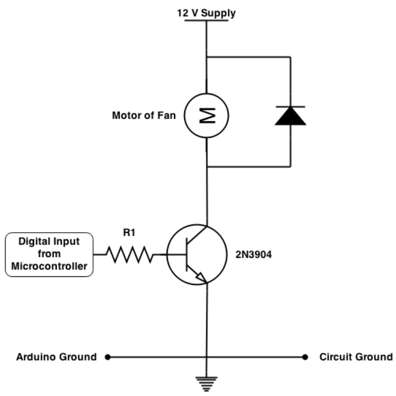

Figure 12: Circuit of fan connected to Arduino

6. LCD Design Specifications

Our design gives our customers the freedom to be able to see the current temperature, and control the temperature inside the fridge. This required a display interface that would allow the user to see the current temperature inside the fridge, outside temperature, and select the desired temperature. Figure 13 shows LCD that would be placed on the outside of the fridge door.

We needed an interface that would minimize the pins used on the arduino and receive commands from the user. We chose to use the monochrome LCD compatible with arduino microcontroller. This screen only uses 2 analog pins on the arduino because it is using I2C multiplexing protocol. It provides 16 X 2 character display with 6 feedback buttons.

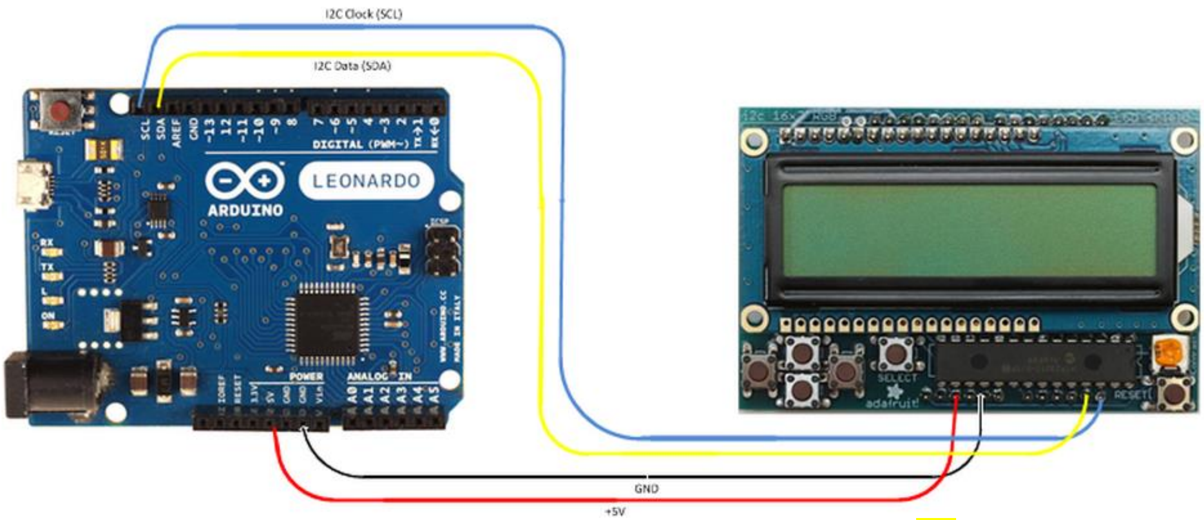

Figure 13: LCD Connections to Arduino, figure is adopted from [12]

6.1. Button feedback design

Our LCD also has 6 built in buttons. These include reset, up, down, left, right and select buttons as shown in figure 14. Our design protocol of the user interface is detailed below [16].

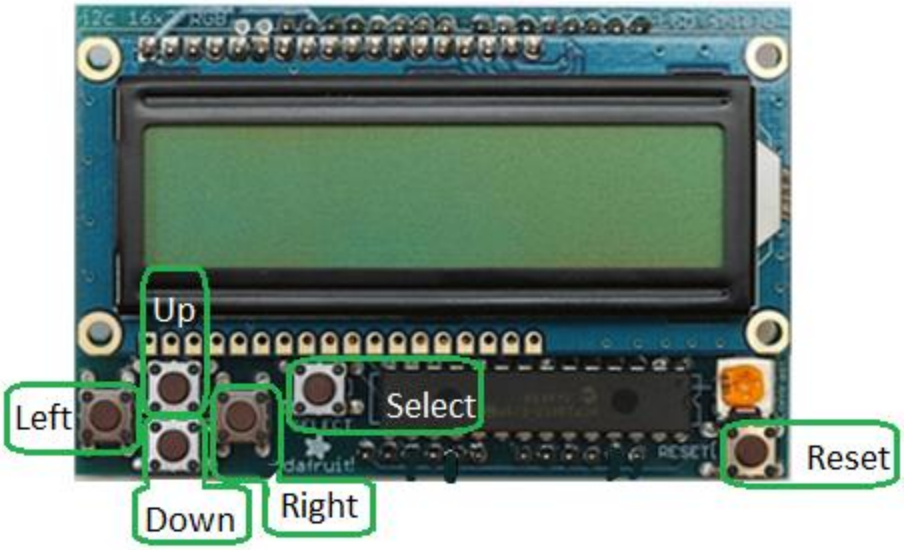

Figure 14: Push Button Identification

Up/ Down buttons:

Up and down buttons would be used to increase and decrease the desiring temperature respectively. In order to override the existing target temperature, the user would first go to the temperature by pressing up and down buttons and then press select to override the previously desired temperature.

Left/ Right buttons:

Left and right buttons would allow the user to select the screen display types and choose between Celsius and Fahrenheit. In order to select the desired display, the user would press the left and right buttons which would allow navigation through the display types available to the user. Once the user is satisfied with the display type, pressing the select button would override the previous display method.

Reset button:

Pressing the reset button at any time would set the target temperature inside the fridge at default of 5 degrees Celsius. The temperature display chosen by the user (Celsius or Fahrenheit) would not change. In other words, if the user had chosen the display type to be Celsius, pressing the reset button would not change this display type.

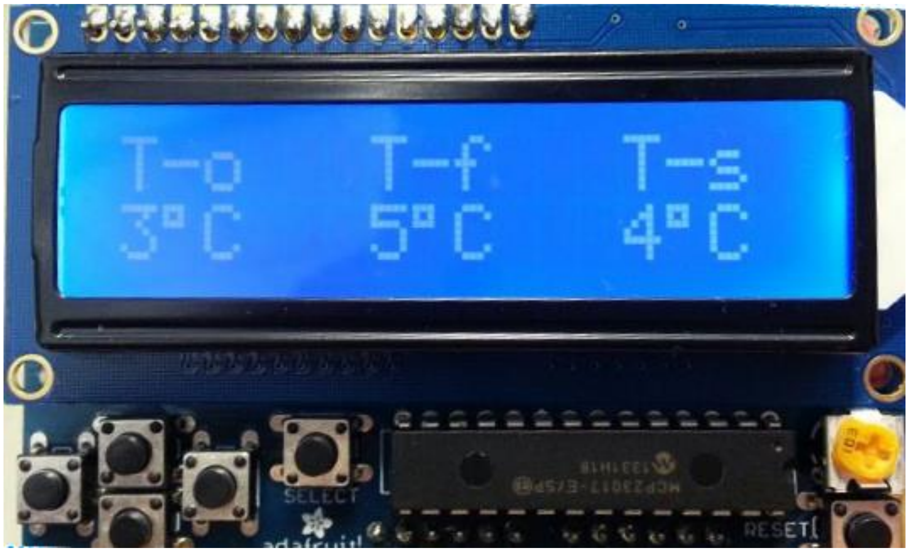

Figure 15: Temperature Displayed

Figure 15 shows one of the display formats available to the user. T-o stands for outside temperature, T-f stands for fridge temperature, and T-s stands for set temperature. Below these three terms is the temperature of that term.

6.2. Voltage, current and energy usage of LCD interface

The LCD operates at 5V which is supplied directly by the arduino microcontroller.

Current draw when the backlight is on: 8.627 mA
Current draw when the backlight is off: 2.682 mA

Table 1 was derived using formulas (4) and (5), given that there are 730 hours in a month.

Power(W)= Voltage(V) × Current(I) Equation 4
Energy(kWh) = Power(W) × time(hr) / 1000 Equation 5

Table 5: Energy Consumption of the LCD

	Current	Power	Energy in a month
Backlight ON	8.627 mA	0.043135 Watts	31.5 Wh
Backlight OFF	2.682 mA	0.01341 Watts	9.8 kWh

Table 1: Energy consumption by LCD

The backlight will be on 10 seconds after the last button pressed by the user. According to the above experimental data, the LCD will consume about 0.15 kWh per year.

Constraints/ Limitations

Since this LCD display is being placed on the front of the fridge door, we would need to maneuver the wires to the back of the fridge such that they don't interfere with the sealing and motion of the fridge door.

The screen size and the number of buttons available are an important constraint of this application for the user. The display protocol devised is quite similar to many of the home appliances such as the thermostat or the microwave oven that use a screen size similar to our application. The buttons and display protocol mentioned above may change if we find a way that is more convenient for the user or saves a significant amount of energy.

7. Temperature Sensors:

7.1. Overview of Temperature Sensors

Considering that one of the main features of this ultra-efficient refrigerator is to accurately read and monitor the temperatures within the exterior and interior structure, it is prevalent to use temperature sensors that would provide extremely high precision readings, making it ideal for our prototype.

Types of temperature sensors:

1. ONE WIRE DIGITAL TEMPERATURE SENSOR - DS18B20
2. WATERPROOF - TEMPERATURE SENSOR (DS18B20)

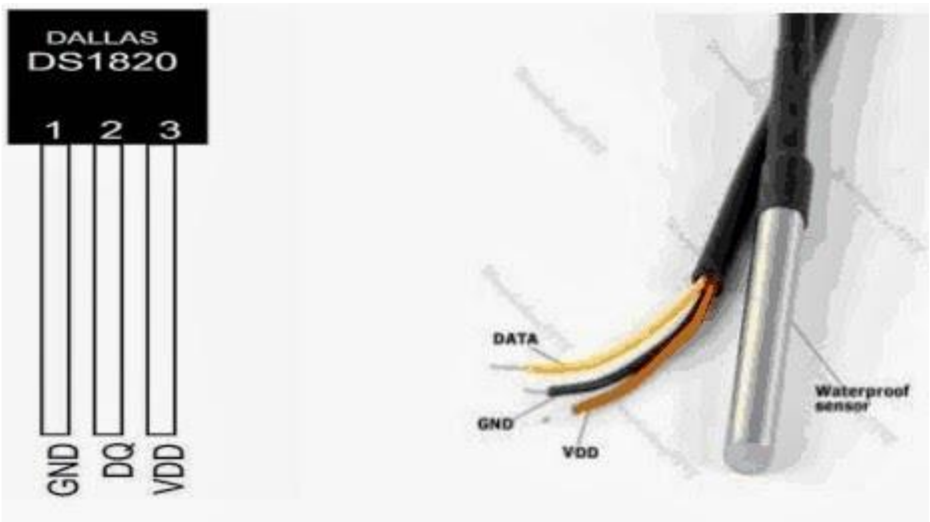

Figure 16:(Left) Pin Assignments for DS1820 temperature sensor; (right) waterproof temprature sensor

7.2. Applications

We have chosen ONE WIRE DIGITAL TEMPERATURE SENSOR - DS18B20 to detect the internal temperature of the fridge. The primary reason for selecting a digital one over an analog one was that, the former provides a more accurate and precise representation of a sensor than that of the latter [R6.3-1]. Also, having a digital temperature sensor will enable the user to have control over the resolution of the sensor.

To measure the outside temperature, we will be using the WATERPROOF - TEMPERATURE SENSOR (DS18B20). It has similar features to that of the one stated above, with additional waterproof capacities.

Since this particular sensor will be placed on the exterior surface, it will be exposed to other environmental elements like humidity and moisture, for which we would require a sensor with impermeable surface [R6.7-I].

7.3. Design Specifications

The DS18B20 is a simple device for measuring the temperatures with an operating range of-55°C to +125°C [R6.8-I] and is accurate to ±0.5°C [R6.2-I], over the range of -10°C to +85°C. The two advantages of using this particular sensor are that, it only needs one pin for the communication, and multiple units can be connected together on the same one wire, thus enabling it to be placed in many different places. Also, the sensor requires no other external component to power it up, since power for reading, writing and performing temperature conversions can be derived from the data line itself. Power supply range is 3.0-5.5V [R6.5-I].

To measure the outside temperature, we will be using the WATERPROOF - TEMPERATURE SENSOR (DS18B20). It has similar features to that of the one stated above, with additional waterproof capacities. Since this particular sensor will be placed on the exterior surface, it will be exposed to other environmental elements like humidity and moisture, for which we would require a sensor with impermeable surface [R6.7-I].

7.4. Wiring with Arduino

As shown in the layout below, GND pin and the VDD pin from the temperature sensor are both connected to the Arduino GND pin. Then, we connected pin 2, the data input/output pin to the Arduino I/O pin. A resistor R1 is required to be connected to the DQ pin to pull it up to 5V.

The waterproof temperature sensor has similar wiring protocol as the one stated above, red wire being the VDD, black the GND and white the DQ.

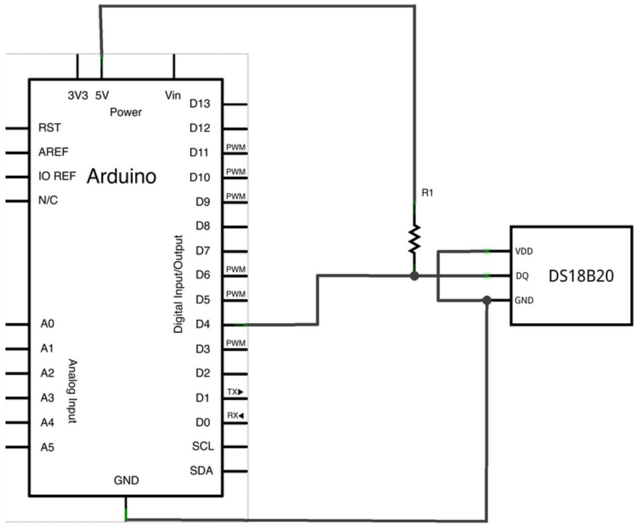

Figure 17: Correct Practice for Wiring the temperature Sensor to the Arduino

7.5. Humidity Sensor

As a part of the post development plan, we are planning to incorporate a humidity sensor to monitor and control the humidity and moisture levels in the air inside the ducts. We would be placing a humidity sensor in the ducts to control the buildup of moisture that could interfere with air flow circulation. The plan is to use a DHT11 Digital Temperature and Humidity sensor that has a measurement range of 20% ～90%RH.

8. Microcontroller Design Specifications

8.1. Types of Microcontrollers

For our refrigeration system, we had initially considered three different options to control the servo motors, fans, various temperature sensors, dampers in the system – a FPGA board, comparator circuits, and various microcontrollers. FPGA devices had a lot of resources and were not specific enough for our purpose. FPGA devices also cost significantly more than a comparator circuit and the microcontrollers. FPGA boards also required a lot of power to carry on its functionality which did not fit our purpose due to the fact that our project focuses on an ecofriendly and power efficient system [9]. A comparator circuit could be used to control the dampers and the fans but since our project required extreme precision in terms of controlling the degree of the dampers to open and the speed of the fan to control the amount of air flow into the fridge, we decided to use the microcontroller for our refrigeration system. Different types of microcontroller such as the Raspberry Pie, Arduino Uno, and BeagleBone Black. The Raspberry Pie and BeagleBone took a minimum of 150mA and 210mA at a minimum 5V respectively on idle conditions which was much higher than what the Arduino microcontroller took which was less than 75 mA [10]. After taking all of the above factors into account, we decided to use the Arduino Uno microcontroller as it fits all our requirements with its ability to ability to efficiently control all of the necessary devices required for the refrigeration system. The Arduino Uno microcontroller also contained a vast amount of resources on the internet and therefore helping out the with the time constraints which could result as the project moves forward.

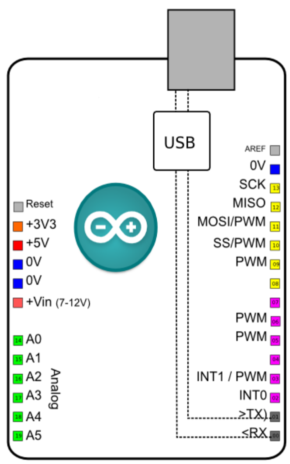

Figure 18: Arduino Uno Schematic [11]

8.2. Microcontroller Requirements

The Arduino Uno microcontroller has a built in flash memory of 32 KB. Since we are not overloading the microcontroller with a lot of devices and with the variety of resources and built in functions available online and the total number of devices that we will be using to control using the Arduino microcontroller, we will be able to fit the entire software code for the refrigeration system in the available 32KB flash memory [R9.1-I]. The microcontroller will be placed in an enclosure at the back of the fridge in an ABS plastic material. [R9.6-I]. The plastic material will prevent it from electrostatic charges protecting the microcontroller [R9.8-I]. Since the microcontroller will be placed behind the fridge at room temperature, it will not have any problems handling extreme temperatures or temperatures beyond its threshold [R9.10-I]. The microcontroller will also be isolated from any moisture in the system [R9.11-I].

Controlling the fans

The microcontroller will be in charge of controlling the speed of the fans and in charge of controlling when the fan would go on and off. The speed of the fans will be controlled using the PWM digital I/O pins on the microcontroller. Since the microcontroller has 6 digital I/O PWM pins and since we are only using two 12V DC fans with PWM signals, it would be more than sufficient for our system functionality to control the amount of air flow. Although the microcontroller has the ability to provide a PWM signal (Described above in the Fan specifications) between 0 – 100%, it does not have the ability to turn off the fan using the PWM signal itself. In order to completely turn off the fan, a BJT (Bipolar Junction Transistor) will be used as a relay to turn the fan on and off. The BJT (2N3904) will be used as a relay for this purpose. The BJT will only turn the fan provided that an input signal has been sent through the microcontroller. The Base junction of the BJT will receive an input from the microcontroller when the required temperature from the user has met and will turn on the Fans to cool the inside of the fridge if the temperature inside the fridge has dropped to a temperature lower than the desired. The fans will be powered using an external 12 V power source due to the limitation of the microcontroller only being able to provide a maximum output voltage of 5V [R9.4-I].

Controlling the temperature sensors

The microcontroller will be in charge of keeping track of the temperature of the outside air and the temperature inside the fridge. Both the temperature sensors used are the exact same model (DS18B20 One-Wire digital temperature sensor) except the fact that the temperature used to detect the outside temperature is waterproof and the temperature sensor used to detect the temperature inside the fridge is a regular temperature sensor. Both the temperature sensors will fit into one of the Digital Input pins of the microcontroller. The microcontroller will be able to process and calculate the temperatures to two decimal points at minimum [R9.7-I]. The temperature sensors will also be powered using the 5V output pin of the microcontroller which is labeled as Red in the Figure 17 above [R9.3-I].

Controlling the Servo Motors

The microcontroller will be in charge of controlling both the servo motors that would turn the damper to its specific positions to let the required amount into the fridge and to exhaust warm air outside the fridge. Both the servo motors will be connected to the Digital Inputs of the microcontroller. The motors will also be powered using the 5V output pin of the microcontroller which is labeled as Red in the Figure 19 in the Appendix [R9.2-I].

Controlling the Compressor

The microcontroller will be in charge of controlling the relay circuit which controls the turning on and turning off of the compressor. A signal from the microcontroller will be sent to the transistor that controls the relay circuit which in turn turns the compressor on or off [R9.5-II].

9. System Test Plan

9.1. Unit Test

Microcontroller Testing

- Supply a stable voltage of 7 – 12V to the microcontroller using an external power adapter
- Use a digital multi meter (DMM) to test the voltage being applied to the microcontroller and ensure it's in the required range
- The microcontroller will be able to draw as much current as it needs and therefore at least a minimum of 500 mA power supply should be used to be on the safe side
- Confirm that the ground and the 5V output pins of the microcontroller are working as required using a DMM
- Confirm that the analog pins A4 and A5 (SCL and SDA) pins are working by sending a High and Low signal to it. Confirm that the pins are able to output a voltage of around 5V and a maximum of 40mA at during its maximum stability.
- Confirm that the all the eight digital I/O pins used for the refrigeration system are working as required by sending a high/low signal to it and confirming the maximum voltage output from the pins are 5V and the maximum current is around 40mA.
- Confirm that the ground signal pins of the Arduino microcontroller work as expected
- Ensuring that the total size of the written code for the refrigeration system is less than 32Kb by compiling and verifying using the Arduino Software Program.

Fans Testing

- Supply a stable voltage using an external power adapter to the Fan circuit
- In order to achieve this, the tests should be conducted of how much voltage is required to power the entire circuit with the BJT in place.
- In order to achieve a stable voltage of 12V across the fan, the Vcc input to the circuit would have to be much higher than the required 12V for the fan and this value would have to be calculated using a DMM across the fan
- The total current across the fan will have to be restricted to a maximum of 0.15A as this is the speed at which the fan runs at full RPM and this value must be supplied using the proper resistor values in the circuit

- The fans should be tested to see if it comes to a complete stop using the required circuit with the proper specifications
- The PWM feature of the fan should be tested from a range of 0 to 100% using the Arduino Digital software algorithm and see that the maximum Reading and minimum Reading can be achieved using the Tachometer pin from the fan at 100% and 0% respectively

Temperature Sensors

- Supply a stable 5V input to the temperature sensors
- Ensure that the circuit input and ground pins are shorted and the data signal is connected to the digital pin of the microcontroller
- Run the application on the Arduino Software that outputs the temperature through the serial output and observe the real time temperature outputs
- Use a blow dryer and blow air onto the temperature sensors and ensure that the temperature increases
- Make sure that the temperature sensors output on the screen have a minimum of 2 decimal points

Duct

- Observe the duct and ensure that there are no holes or leakage in both the supply and exhaust ducts
- Closing one end of the duct and blowing air into the duct from the other side, observe if any air can be felt from anywhere in the duct for leakage
- Using two temperatures, one on the outside and the other inside the fridge, start the fan and take in air through the supply duct and using the temperature sensor inside the fridge, observe how much temperature drop can be observed on the other end of the duct near the inside of the fridge

Compressor

- Send a signal from the microcontroller to the relay that controls the compressor to see if turns on or off depending on the signal

LCD

- Ensure that that conversion and the display on the LCD is appropriate and accurate
- Ensure that all the menu buttons in the LCD is working according to the design by making sure:
    - The user can select an appropriate desired temperature input
    - Conversion between Celsius and Fahrenheit are properly calculated
    - Visually whether the LCD displays the correct characters

Servo Motors

- Supply a 5 Volt stable power to the servo motors
- Ensure that digital signal from the microcontroller can control the servo motors to rotate only 90 degrees
- Ensure that the servo motor has enough torque to turn the dampers in the refrigeration system
- Ensure that the dampers are not an overload to the motor by testing the rotation at various speeds

9.2. Integrated Testing

- The integrated testing will be done in an artificial cold environment. This will be achieved by putting ice inside a chest beverage cooler and connecting one end of the supply duct to the cooler to take in the air. This air will then be then be used to cool the fridge and maintain a cold temperature inside the fridge
- All the individual unit tests will be combined together to test for a working environment
- The water proof temperature sensors will be tested using the microcontroller on the computer to see whether it is outputting the correct value that is inside the cooler test environment
- The servo motors and the dampers connected to them will be tested to see if it opens at a certain degree depending on how cold the detecting outside temperature is and how cool the inside of the fridge is
- The supply fan will be tested to see how to see if its running or not depending on what the temperature is set
- The exhaust fan will be tested to see how the fan is running or not depending on whether the internal temperature is acquired
- The fridge will be left open to avoid the cooling inside the fridge to test whether the exhaust dampers controlled by the servo motors work as required
- Ensure that the compressor turns on by doing a test when the outside temperature goes above the required set value using the Arduino Software code by settings the bounds
- Ensure that the compressor turns off when the outside temperature goes below the required set temperature value

10. Appendix

10.1.I2C Multiplexing protocol theory

I2C bus uses two signal connections to the Arduino, SCL and SDA (A4 and A5 pins on the Arduino Board). The SCL is the clock line that synchronizes the SDA, which is the data line. The devices on this I2C system are either master or slave. The master always drives the clock and initiates the transfer over the bus. Both master and slave can transfer data over I2C bus but only the master can control the clock. In our case, the master is the Arduino microcontroller and the slave is the MCP23017 chip. The master device sends out signals in a sequence. First, a start sequence to alert the slave devices to listen, then the address of the specific slave that it wants to communicate to is sent out. Once it has picked out the corresponding slave, all the rest of the slave will ignore the signal and will ask for the address of the internal register number that it needs to write to. After this step, the data is finally transmitted until no more is left and then a stop sequence is sent.

This is a basic overview of the I2C software protocol. Although we don't need direct knowledge of above sequences, it may come in handy for future debugging and designing purposes.

10.2.Servo Wiring with Arduino

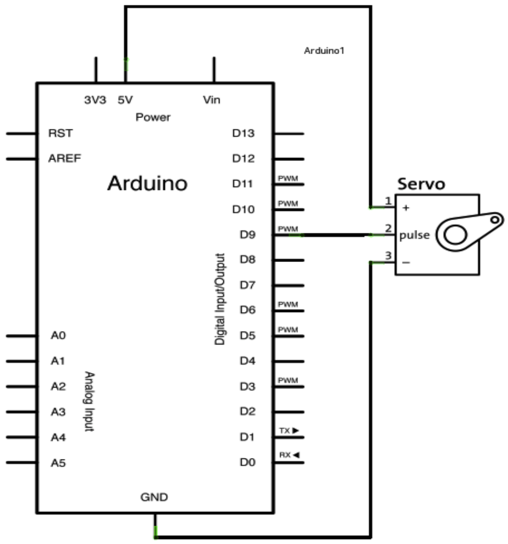

Figure 19: Schematic for wiring a servo motor to the Arduino[8]

10.3.Data for Pressure Calculations

Friction Loss in Ducts $\triangle P = \frac{0.109136\, q^{1.9}}{d\, e^{5.02}}$ = 0.3
△p = Friction or pressure loss for 100ft of duct
$d_e = Equivalent\ duct\ diameter$
Q = Air flow volume (cfm)

Used an online calculation tool to calculate the pressure loss inside the duct and velocity pressure given by:

Duct Pressure Loss and Velocity Presure Results		
Total Duct Loss (inches Water): 0.349	**Velocity Pressure (inches Water): 0.017**	
Job Number: **Client:** **Date:** **Line Number:** **Fluid:** **Duct Type:** ROUND **Duct Diameter (in):** 5 **Flow Rate:** 70 ACFM **Duct Length (ft):** 4 **Viscosity (cP):** 0.018 **Inlet Pressure (PSIG):** 0 **Temperature (F):** 41 **Duct Material:** GALVANIZED METAL **Duct Roughness (ft):** 0.0005	**Fluid Velocity (ft/min):** 513.63 **Reynolds Number:** 23384 **Flow Region:** Turbulent **Friction Factor:** 0.0275 **Density at Inlet:** 0.079 **Specific Volume at Inlet:** 12.61 **Specific Heat Ratio:** 1.4	**Velocity Pressure (inches Water):** 0.017 **Total Duct Loss (inches Water):** 0.349 **Straight Duct Loss (inches Water):** 0.005 **Hood Entry Type:** Plain Duct End **Hood Loss Factor:** 0.93 **Hood Entry Loss (inches Water):** 0.034 **Elbow Type 1:** Mitered turning vanes **Radius / Duct Diameter 1:** 1.50 **Number Of Elbows 1:** 1 **Elbow Sweep 1 (Degrees):** 90 **Elbow Loss Factor 1:** 0.6 **Elbow Loss 1 (inches Water):** 0.01 **Duct Exit Configuration:** Vertical Discharge, No Loss **Exit Configuration Loss (inches Water):** 0 **Equipment Name 1:** Fliter **Equipment 1 Pressure Drop (inches Water):** 0.3

Figure 20: Table of Duct Pressure Loss and Velocity Results

11. References

[1] V. Ryan, “THE DIODE”, [online] 2009, http://www.technologystudent.com/elec1/diode1.htm (Accessed: 9 March 2014)

[2] “Cougar Vortex PWM”, [online] 2013, http://www.cougarworld.com/products/fans/vortex_pwm.html, (Accessed: 9 March 2014)

[3] 5" In-Line Duct Fan, *Global*, [online] 2014, http://www.globalindustrial.ca/p/hvac/exhaust-fans/in-line-duct-fans/in-line-duct-fan-212-cfm?infoParam.campaignId=T9F&gclid=CPfu7IDAjL0CFYeUfgodDUsALQ (Accessed: 9 March 2014)

[4] “Comparative Examples of Noise Levels”, *Industrial Noise Control*, [online] 2010, http://www.industrialnoisecontrol.com/comparative-noise-examples.htm, (Accessed: 10 March 2014)

[5] C. Cunningham, “Connecting a relay to Arduino” (Make), [online] 2009, http://makezine.com/2009/02/02/connecting-a-relay-to-arduino/, (Accessed: 10 March 2014)

[6] “Application of PWM speed control”, *JMC*, [online] 2014, http://www.jmcproducts.com/wp-content/uploads/2012/02/PWM_technical_paper.pdf, (Accessed: 10 March 2014)

[7] µCI – Arduino DS1820 Temperature Sensors and LCD, *µ Control It*, [online] 2012, http://www.ucontrolit.tv/184/, (Accessed: 9 March 2014)

[8] Servo Library, *Arduino*, [online] 2014, http://arduino.cc/en/Tutorial/sweep#.UyLFQ4VPKH9, (Accessed: 9 March 2014)

[9] A. Sarwar, FPGAs vs Microcontrollers, (ElectroDesigns), [online] 2012, http://electrodesigns.net/blog/fpgas-microcontrollers/, (Accessed: 10 March 2014)

[10] M. Leonard, How to Choose the Right Platform: Raspberry Pi or BeagleBone Black?, (Make), [online] 2014, http://makezine.com/magazine/how-to-choose-the-right-platform-raspberry-pi-or-beaglebone-black/, (Accessed: 9 March 2014)

[11] La carte Arduino Uno, *Référence Arduino français,* [online] 2012, http://www.mon-club-elec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.MaterielUno, (Accessed: 11 March 2014)

[12] RP Electronics, Electronic Components & Parts : Motors : Servo Motors," [online] 2012, http://www.bedfordstmartins.com/online/citex.html (Accessed: 14 March 2014).

[13] HVAC Systems: How They Work, edc, [online] 2000, http://www.edcmag.com/articles/hvac-systems-how-they-work, (Accessed: 12 March 2014)

[14] Technical Insulation, ISOVER, [online] 2010, http://www.isover-technical-insulation.com/, (Accessed: 11 March 2014)

[15] Friction Loss in Ducts, The Engineering ToolBox, [online] 2014, http://www.engineeringtoolbox.com/duct-friction-pressure-loss-d_444.html, (Accessed: 12 March 2014)

[16] L. Fried, "RGB_LCD_WIRING_rev1.png," (Adafruit learning system), [online] 2012, http://learn.adafruit.com/assets/1439 (Accessed: 14 March 2014).