

**APPLICATION OF LOAD SHIFTING FOR COMMERCIAL HVAC-R SYSTEMS
VIA STATIC AND DYNAMIC EVENT AUTOMATED DEMAND RESPONSE**

A Thesis Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Justin Anthony Martinez
December 2022

Copyright © 2022 by Justin Martinez
All rights reserved.

ACKNOWLEDGEMENTS

I firstly wanted to thank my advisor, Dr. Fran Li, who made this journey through graduate school possible. Allowing me to explore higher education in the tumultuous times of the COVID-19 events and the war in Europe is a kindness I can never repay. Thank you for giving me the opportunity to explore power systems. This amazing time at the University of Tennessee has given me much more confidence in my field and I feel that I have the knowledge necessary to succeed.

Next, I would like to thank my colleague, Vince Wilson. Your support made this work possible. I enjoyed being in the lab with you and wish you the best of luck on your doctorate.

ABSTRACT

Life in the modern era is inextricably tied to energy and the unending, ever-growing need for more. The technologies that drive society, in fields such as communication, infrastructure, and heavy industry, are dependent on that electrical energy being reliable and readily available. This places an extreme importance on the power generation sector as a function of meeting the demand required for stability. However, as seen with increased climate volatility due to misused or otherwise mismanaged resources in the heavy industry, and the uncertain and variable renewable energy generation; there must also be ecological as well as economic consideration when discussing modern energy. Therefore, a balance between sustainability and energy demand must be met, to increase the desire for better use of energy resources.

This thesis focuses on an application of a demand response (DR) program that may increase the performance of energy resources while also maximizing monetary savings for end-users in the commercial sector. This DR may provide support for the uncertain and intermittent nature of renewable generation. By targeting Heating, Ventilation, Air Conditioning, and Refrigeration loads (HVAC-R) utilities have access to non-critical loads that they may be able to shed during times of high energy demand. This allows for the allocation of less resources during peak hours, which may lead to less strain the electrical grid and could increase the threshold for which additional energy resources will need to be constructed.

Based on relay to substitute the analog parts of a refrigerator and a development PC to drive the control logic a closed loop, Automated Demand Response (ADR) is

implemented to utilize both static and dynamic Time of Use (TOU) events as well as Critical Peak Pricing (CPP) events. This enables the end user to respond to two types of unique events, resulting in potential increased energy efficiency, and monetary savings for the end user via load shifting.

TABLE OF CONTENTS

Chapter One Introduction	1
1.1 Background to the Modern Energy Sector and Demand Response	1
1.2 Literature review	4
Commercial sector	4
Industrial sector.....	8
Automated Demand Response	11
1.3 Shortcomings of the Modern Electrical Energy Sector	12
Chapter Two PROPOSED Demand Response application	13
2.1 Problem Statement	13
Justification	13
Solution	14
Potential benefits.....	14
2.2 Application Goals	14
Hardware	15
Software	15
2.3 Proposed Design platform.....	16
Initial design.....	16
Research evolution.....	22
Final design.....	26
Chapter Three Materials, Methods, and experimental PROCESS	28
3.1 Materials	28

Software	28
Hardware	28
3.2 Design platform methods	28
Implementing CPP with static and dynamic time of use	41
3.3 Experimental process	44
Baseline experimentation of analog refrigerator.....	44
Final design experimentation.....	45
Chapter Four Results.....	49
4.1 Energy Efficiency	49
Cooling efficiency.....	49
Steady State efficiency.....	49
4.2 Monetary savings	49
Initial digital with Static TOU operation	49
Improved digital with CPP operation	49
Improved Digital refrigeration operation.....	49
.....	62
Chapter Five Conclusion and future application	68
5.1 Conclusion	68
Results explained for energy consumption and monetary analysis: baseline analog and digital conversion.....	68
Final thoughts: sources of error and implications for the future of automated demand response.....	72
5.2 Future Application	73

Improved system: thermodynamic model, machine learning, and website controls	73
Expanded application: residential and industrial uses	74
List of References	75
Vita.....	81

LIST OF FIGURES

Figure 1-1: Modern energy sector as described above. This may also be called a power system, interchangeably, through the text [1].	3
Figure 1-2: A look at global greenhouse gas emissions to compare energy usage by sector in the US, as shown in figure 3, to emissions worldwide [3].	5
Figure 1-3: Looking at the break down of energy consumption in the US, it is possible to glean insight into which sectors have the highest energy usage as well as which sectors may have potential for DR application [4].	6
Figure 2-1: An illustration of the original design platform architecture.....	17
Figure 2-2: The raspberry pi receives a constant signal from the development pc. Upon receiving this signal, the Rpi interprets the string and then selects the proper channel. If this signal is interrupted a failsafe is enacted that will allow the refrigerator to operate normally.	25
Figure 2-3: A visualization of the design platform's final architecture. The development pc dictates the commands sent to the raspberry pi. The Rpi interprets the signal to either allow for normal control or intelligent control.	27
Figure 3-1: The main refrigeration platform was the FR3201, selected for its analog nature which made modifications easy, as well as its size, which balanced accurate scaling of thermodynamic modeling and accessibility due to weight.....	29
Figure 3-2: A look at the internal components of the platform to give a complete image of the system. This includes the thermocouples for tracking temperature and the wiring for compressor control.	30

Figure 3-3: Highlighted in blue is the K type thermocouple for air temperature readings. This acted as the primary device that would determine system reponse. Pointed to with a red arrow is another K type thermocouple used for reading surface temperatures.	31
Figure 3-4: A surface thermocouple used to record the temperature of the evaporator. A great use for having additional thermocouples is error checking as the values of the thermocouple reads should be relatively close to one another.	32
Figure 3-5: A surface thermocouple used to record temperature at the bottom of the refridgerator. While not directly used to influence system responses, these thermocouples helped to build an understanding of the system thermodyanmics....	33
Figure 3-6: A depiction of the connection from relay to compressor circuit, acting as a substitute for the anolog thermostat of the original system. This is achived with a bypass circuit that will allow for the use of either the analog or digital system.....	34
Figure 3-7: A full image of the temperature collection system, including the DAQ.....	35
Figure 3-8: The DAQ model USB-6001 acted as the primary data collection devcie connecting the thermocouple readings to the development PC.....	36
Figure 3-9: This is a small board used to power the amplifiers.....	37
Figure 3-10: The anolog K type thermocuple amplifier, model AD8495, used to allow for readable output from the thermocouples to the DAQ.	38
Figure 3-11: The Raspberry Pi 4 model B (RPi) worked as the controller for a relay which in turn drove the compressor. If a fault was detected the, Rpi was capable of	

automatically switching to the analog circuit to preserve the integrity of the system.
..... 39

Figure 3-12: A two-channel relay used to control the compressor for the refrigerator. ... 40

Figure 4-1: This graph shows that there are many temperature fluctuations for the original thermostat's initial cooling to steady state. Also shown is the time it took to reach steady state, approximately 1 hour..... 50

Figure 4-2: This energy graph shows the power consumption of each cooling cycle including the total for the system, approximately 30 watt-hours..... 50

Figure 4-3: This graph shows the many cooling cycles of the original thermostats cooling to steady state. Also shown is the time it took to reach steady state, approximately twenty-four hours..... 51

Figure 4-4: This energy graph shows the power consumption of each cycle including the total for the system, approximately 400 watt-hours..... 51

Figure 4-5: This graph shows the many cooling cycles of the original thermostats cooling to steady state. Also shown is the time it took to reach steady state, approximately nineteen hours. 52

Figure 4-6: This energy graph shows the power consumption of each cycle including the total for the system, approximately 500 watt-hours. Note how the first hours resulted in a longer cooling cycle. 52

Figure 4-7: This graph shows comparatively few temperature fluctuations indicating one longer cooling cycle of, approximately 40 minutes..... 53

Figure 4-8: This graph shows the energy consumed for the cooling cycle, approximately thirty-five watt-hours.	53
Figure 4-9: This graph shows a steady drop in temperature, indicating a longer cooling cycle or cycles.....	54
Figure 4-10: This graph shows that there was one long cooling cycle using approximately, 80 watt-hours.	54
Figure 4-11: This graph illustrates the steady cooling of the refrigerator at a rapid rate. This indicates a longer cooling cycle, approximately eight hours.....	55
Figure 4-12: This graph shows the energy consumption of cooling cycle, approximately 170 watt-hours.	55
Figure 4-13: Shows the steady state temperature as consistent with some spikes between cooling cycles.....	56
Figure 4-14: This graph shows there are many cooling cycles and the total energy used in a day, approximately 175 watt-hours.....	56
Figure 4-15: This graph shows the steady state temperature as less consistent than no load with a higher average temperature.....	57
Figure 4-16: This graph shows the energy consumed over 24 hours, approximately 164 watt-hours.	57
Figure 4-17: This graph shows the temperature fluctuations throughout the hours of operation in a day with full load conditions.....	58
Figure 4-18: This graph shows the power used over twenty-four hours being approximately 149 watt-hours.	58

Figure 4-19: This graph shows a wider temperature range and distinct cooling cycles throughout the day.	59
Figure 4-20: This graph shows the energy usage through the day, approximately 262 watt-hours.	59
Figure 4-21: Show cases the increased amount of cooling cycles proportional to the load added.	60
Figure 4-22: This graph shows the energy used during the day as being approximately, 245 watt-hours.	60
Figure 4-23: This graph shows more cooling cycles with a wider temperature range based on the amount of load in the unit.	61
Figure 4-24: This graph shows the energy usage during a twenty-four-hour period being approximately 290 watt-hours.	61
Figure 4-25: Pricing data taken from, Comed an Exelon company, from 2019 [24]. This shows the predicted rates twenty-four hours in advance.	62
Figure 4-26: The real-time rates given by Comed during 2019 [25].	63
Figure 4-27: This graph shows the temperature range of the refrigerator during the STOU test.	64
Figure 4-28: This graph shows energy consumption of approximately 256 watt-hours while showing the ride through of the STOU event.	64
Figure 4-29: This graph shows the temperature range for the winter static TOU event. .	65
Figure 4-30: This graph shows the energy consumption of STOU event, approximately 350 watt-hours).	65

Figure 4-31: This graph shows the temperature performance of the improved digital system which is significantly more precise and has tighter control over the temperature range.....	66
Figure 4-32: This graph shows the energy consumption of the improved digital code. Consuming approximately 260 watt-hours.	66
Figure 4-33: This graph shows the improved digital system temperature range including the WI-FI failsafe activation and subsequent system reset.	67
Figure 4-34: This graph shows the improved digital system which consumed 284 watt-hours.....	67

CHAPTER ONE

INTRODUCTION

1.1 Background to the Modern Energy Sector and Demand Response

The modern energy sector is the cornerstone of globalized society.

Communication which interconnects the world is only possible on the current modern ear scale due to modern energy capacity. Vital infrastructure that allows for the smooth and safe transportation of goods, people, and services relies heavily on a steady source of power to sustain the immense strain placed on the logistics of a globalized society.

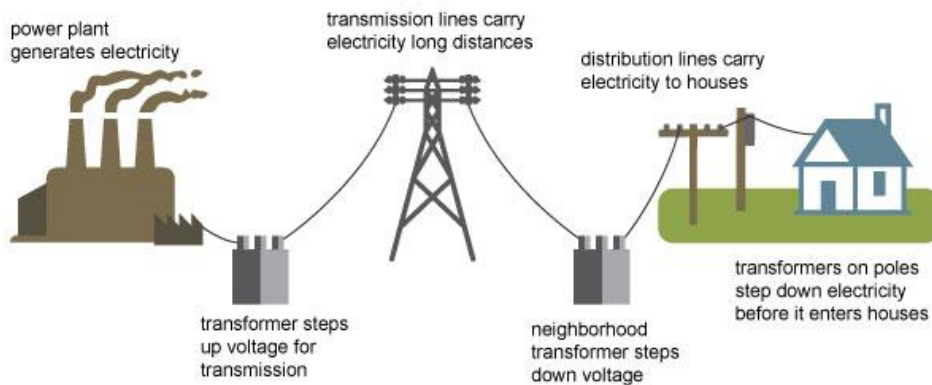
Critically, the standard of living for people around the world has also drastically increased through the availability of technologies such as computers or more advanced medical equipment that are dependent on the energy sector. Higher penetration of more energy dependent technologies such as heating, and air conditioning enables people to live in more extreme biomes with minimized effects on health. Efficient management of energy sector's resources has become increasingly more important as climate change exacerbates the challenges of a climate region, leading to hotter places becoming hotter and colder ones becoming colder.

As seen during the ongoing COVID-19 pandemic, the transition from an open to a quarantined world was facilitated by having access to technologies that provide people with more options. Working from home or working remotely is dependent on having access to the internet, a key component of many businesses today. Therefore, it is very important to have access to the benefits of steady energy.

Traditionally the energy sector is comprised of 3 components: generation, transmission, and distribution. Generation, where the creation of energy takes place, tends to be geographically close to the loads of a region to prevent losses in transmission; however, these generation sources are generally placed remotely from residential districts or highly populated areas. Transmission is how the generated energy is stepped up through a transformer to high voltage lines that then carry energy from plants to an interconnected network linking the power system of a region. Distribution occurs when the energy is then stepped down through substations and is delivered to the various customers: industrial, commercial, and residential [1]. This network can be best visualized in Figure 1-1.

Demand Response (DR) is the response of customer energy demands to better meet the generation available. This is particularly important under the new paradigm of high penetration of renewable energy resources which are uncertain and variable. Frequently, the customer is given incentives to manage reduction in power consumption [2]. DR management is a very important aspect of the energy sector as it offers a way for both the customer and the utility to benefit from more efficient use of energy resources. Continuous effort must be devoted towards the maximization of these resources so that significant economic and ecological factors can be considered. DR exists because of those two primary considerations. The ecological effects of climate change are becoming more evident with the current understanding that the greenhouse effect is a primary driver for climate change. Therefore, it can be concluded that any attempt to mitigate the effects of climate change will place an economic strain on the growth and maintenance of the

Electricity generation, transmission, and distribution



Source: Adapted from National Energy Education Development Project (public domain)

Figure 1-1: Modern energy sector as described above. This may also be called a power system, interchangeably, through the text [1].

energy sector which makes up 25% of world greenhouse gas emissions, as seen in Figure 1-2 [3]. Furthermore, neglect of ecological concerns impacts the economics of the energy sector just as neglect affects any other industry. Better allocation and efficiency of energy resources will result in monetary gain for the benefiting parties.

The pressure to make better use of available resources now applies not only to the generation and transmission side of power systems, but also to the distribution side, *i.e.*, the customers. DR can be easily applied to three sectors of the US distribution network: commercial (12%), residential (17%), and industrial (36%), as seen in Figure 1-3 [4]. The remaining 35%, attributable to transportation, may have a difficult time with DR as the penetration of electric vehicles (EV) is still relatively low as of 2022. This is indicative that energy consumption in the transportation sector has more to do with the fuel for the vehicle and less to do with the generation of power for that vehicle via a power plant. With 65% of the energy sector potentially subject to manipulation in the US, DR is a very promising mechanism for better use of energy resources through allowing the customer to have more control of their energy consumption by giving them monetary incentive while providing flexibility to the utilities in the form of less demand.

1.2 Literature review

Commercial sector

The commercial sector makes up 12% of energy consumption in the US, a not insubstantial proportion [2]. Thus, there has been significant effort in recent years to develop DR programs that target specific load types. From [5] there are four primary load types to consider: type I pure resistive loads, type II compressor-based loads, type III

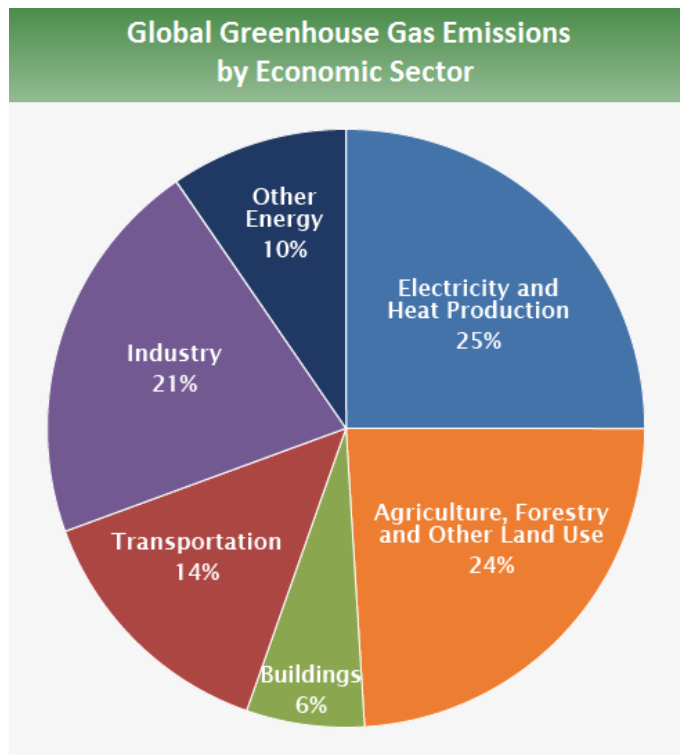


Figure 1-2: A look at global greenhouse gas emissions to compare energy usage by sector in the US, as shown in figure 3, to emissions worldwide [3].

U.S. energy consumption by source and sector, 2020 quadrillion British thermal units (Btu)

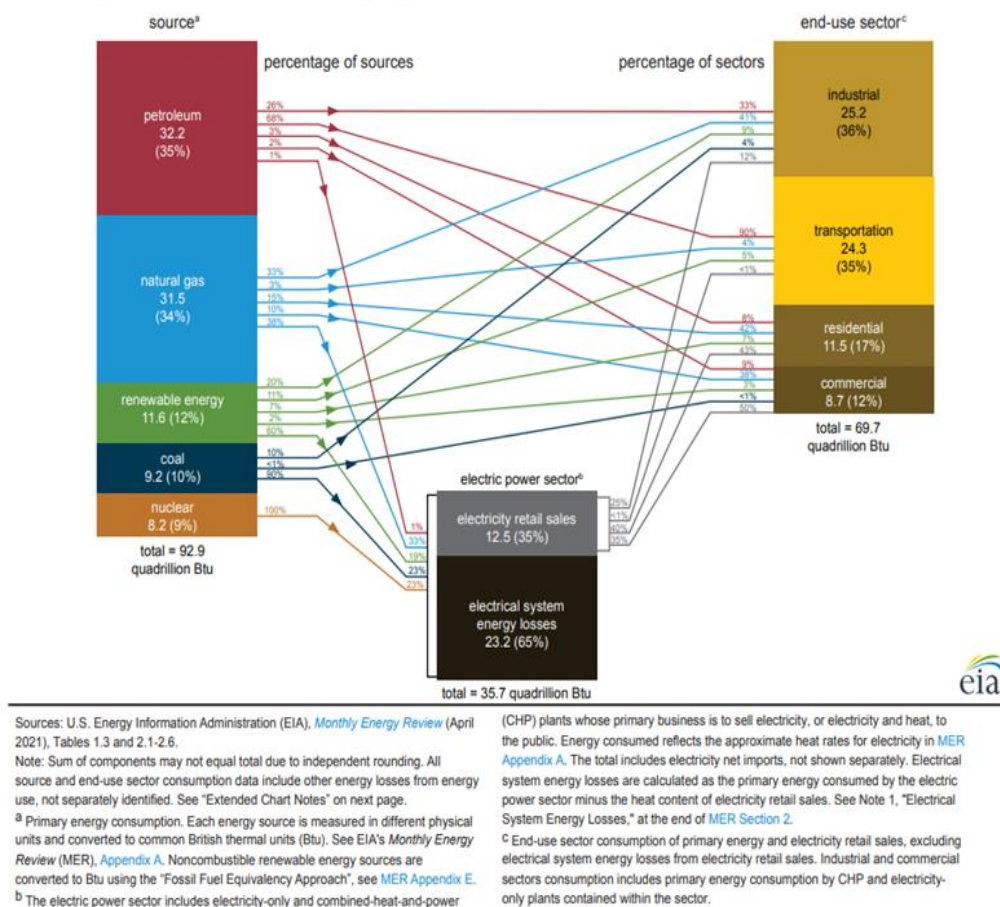


Figure 1-3: Looking at the break down of energy consumption in the US, it is possible to glean insight into which sectors have the highest energy usage as well as which sectors may have potential for DR application [4].

variable speed compressor-based loads, and type IV electric vehicles. Heavy focus can be placed on types I and II, as these loads are the most common and readily available to the current market. Type I loads normally consist of heating elements like Electric Water Heaters (EWH) and Electric Heating Pumps (EHP). Type II loads are driven by a compressor with a single speed, including loads such as HVAC and refrigeration systems [5]. Due to their predictable profiles, both types of loads can be manipulated through several different tools such as load shedding, peak shaving, and frequency control [5][6]. Therefore, due to that predictability, electric utilities are interested in types I and II, as these loads can be remotely controlled and have minimal impact on the customer. This type of load manipulation, defined as the desire to change the patterns of end-user electricity consumption through improvement of efficiency and optimal allocation of power resources, is known as Demand-Side Management (DSM) [6]. DSM is defined very similarly to DR and can be considered interchangeable for the purposes of this thesis.

EHPs can be optimally modeled from [7] allowing for the maximization of energy savings based on the dynamic temperature response of a building given by [8]. Relying on this model, utility companies can schedule events for the customer based on the optimal behavior of the system presented. Similar results can be achieved with HVAC and refrigeration systems, although a different model will need to be derived. Understanding the availability of this modeling is critical, as any type I or type II load that has a sufficient thermodynamic model can be managed in this way.

Participation in commercial DR or DSM can be considered from the perspective of both potential integration and current technologies application. Looking at potential DR integrations, there are two primary considerations for the purposes of this thesis in the commercial sector, (1) renewable energy integration and (2) privacy and security [6]. With new smart grid technologies, it is possible to use scheduling to efficiently allocate power resources to match the load profiles of customers more accurately. In contrast to the traditional power grid where the demand is followed closely hour-to-hour, these strategies use a day-ahead load profiling scheme [9]. However, with the increasing penetration of renewable energy smart grids may struggle due to the variable nature of the generation being used, making the scheduling more complex [10]. Renewable energy thus places, a strain on DR, but may led to customers' being able to pick which form of generation to fulfill their load, leading to increased environmental sustainability [11]. DR is inherently dependent on data acquisition and device communication, meaning that while DR can become increasingly complex, so too do the issues regarding privacy and security. Devices are becoming increasingly smart and interconnected, thereby being able to drive more complex DR applications, but at the cost of heightened security risks such as information breaches or tampering, resulting in the potential to render systems vulnerable and to lower DR effectiveness. Therefore, security and customer privacy must be heavily considered if DR is to succeed and propagate into the future [12][13][14].

Industrial sector

The industrial sector has had many of the same considerations as the commercial sector when talking about recent developments, including the focus on renewable

integration and privacy and security. There may be an even high concentration of effort in the industrial sector for security, as tampering with many of the loads may lead to hazardous or economically devastating results. However, these considerations were mentioned above and will not be restated for this section. An emphasis was placed on the commercial application of DR for this thesis, but it is important to highlight other sectors where DR may be applied. With that said, there are some important developments to mention was speaking about DR in the industrial sector.

The industrial sector takes up three times as much energy as the commercial sector, at approximately 36% [2]. This significant size difference is due not only to the scale of the sector but also to the energy needs of the industry loads. According to [6] there are four major examples of high energy industries that are now participating in DR: aluminum smelting, steel manufacturing, cement plants, and pulp and paper mills. Aluminum smelting, an energy intensive process, allows operators the option to change the amount of energy consumed by changing the tap changers used to supply electricity, thus, effecting the production rates of the plant. However, it is important to note that production and electricity consumption are not proportional, but rather can be modeled through a piecewise function. Therefore, it is possible to optimize the plants electricity consumption for DR purposes such that production is not disrupted to a high degree by allowing the plant to operate at a 25% decrease of consumption of electricity for four hours before disruption occurs. [15-17].

Steel manufacturing is another high energy industry with much of the cost of operation devoted to electricity. From a DR study, it is possible to balance electricity

used and production, resulting in a decrease of 12-52% based on the throughput of the plant. However, interestingly, the smaller the output the more considerable the gains, with large outputs having lessened DR effects [18].

Cement plants are a difficult industry to target for DR as they are time dependent and require significant amounts of electricity throughout the manufacturing process. With that stated, there are three load types which can be categorized as non-shift, semi-shift, and full-shift loads [6]. Each constraint must be considered to understand how DR can be implemented. The output of the plant must not be affected but the electricity used can potentially be shifted to alternative times based on the above load types [19]. This is an excellent example of load shifting which will be discussed in the following section.

Lastly, there are pulp and paper mills which are utilizing DR to benefit from price-based incentive programs. If the thermo-mechanical machines used to create pulp have a larger capacity than the machines they are feeding, the paper processors, then there is room for the operator to lessen the electricity with dynamic responses to electricity price [20].

The industrial sector was considered for this thesis to understand what DR is and where it can be effectively applied. By looking at the recent developments in DR, the commercial sector is the most likely place the application of this thesis's work will succeed, as industrial loads have more to do with high energy processes and less to do with HVAC-R loads.

Automated Demand Response

When discussing DR, several mechanisms can be employed to entice end-users to adopt effective strategies. Two of these methods are Incentive-Based Programs (IBP) and Price-Based Programs (PBP). IBPs aim to lower electricity use during time of high demand using rebates or other monetary benefits [21]. PBPs seek to lower electricity consumption by specifically using the price of electricity to motivate a reduction in that consumption [5]. The common types of DR algorithms that are targeted using IBPs or PBPs are frequency control, load shedding, load shifting, and peak shaving. Frequency control programs target user loads to shed during contingencies allowing the utility companies to support frequency drops. Load shedding is used for similar purposes but can often be a flat drop of a load with the express intent of better matching the generation available. Finally, load shifting, and peak shaving are both programs aimed at lowering electricity usage during time of high demand. By shifting when certain systems use their electricity some appliances such as HVAC and refrigeration systems can ride through times of high demand by pre-cooling themselves. Similarly, by shedding non-critical loads, the utility may save significant resources during peak hour of demand known as peak shaving,

All these types of DR programs rely on primary factor. The end-user or, more precisely, communication and response time with energy requests. Many loads must be shut down or disconnected manually which takes time. Automated Demand response (ADR) seeks to eliminate this issue using standardized communication such that smart devices can be used to implement DR quickly and readily, thus, allowing the customer to

opt for distinct parts of the DR programs they want without having to actively check for events and alerts. This more passive response DSM allows for high degrees of customization as well as faster and more reliable responses from customers, decreasing the time realization of benefits from the programs.

1.3 Shortcomings of the Modern Electrical Energy Sector

The shortcomings of the modern energy sector have less to do with industry practice and more to do with educating the customer. The end-user cannot and should not be expected to understand the intricacies of DR, but the benefits of the programs presented should be understood and weighed, so that a balance can be struck between sustainability and the need for the energy that the world demands. This drive for DR programs will be relevant in the coming years to help support the integration of intermittent renewable energy.

The rest of this thesis will be organized as follows. Chapter 2 is an introduction to the problem statement and proposed design platform including initial approach and the basic evolution of the research. Chapter 3 is the methods and materials behind the construction of the design platform. Chapter 4 will be an explanation of the experimental data. Chapter 5 will cover conclusions drawn from the research topic as well as future research topics that may come from this initial investigation.

CHAPTER TWO

PROPOSED DEMAND RESPONSE APPLICATION

2.1 Problem Statement

The problem statement presented in this thesis will be a retelling of the initial proposal given by Grid Fruit, the client for which this research was developed. It will thus be broken up into three sections: justification, solution, and potential benefits.

Justification

The commercial sale of food is heavily dependent on the capability of refrigeration and, therefore, the cost of electricity is proportional to this demand. As no major recent innovation in commercial refrigeration technology have been observed, the cost of electricity still accounts for 44-62% of the total electrical demand for fresh food retail. According to Oak Ridge National labs (ORNL), 20% of this energy cost could be reduced through increased efficiency, such as optimal defrost and compressor cycles. However, innovation in the field of refrigeration for commercial loads has been discouraged due to the aversion of risk in food retail. Furthermore, the low-profit margins for fresh food as well as competition with online shopping is driving fresh food retail to rely more on frozen foods despite the cost of electricity.

This increased energy demand places a strain on the electrical grid as peak hours demands increased generation. In turn, this affects the price of electricity and further drives up the cost that stores must pay towards their electricity bills. The generation for this increased energy traditionally comes in the form of coal or natural gas power plants thereby increasing pollutants and having a negative effect on the environment [22].

Solution

Grid Fruit aims to save utilities and commercial customers money through the targeting of commercial loads using an “AI-driven Software-as-a-Service product” system. Sensors will feed information into the machine learning algorithm while Grid Fruit monitors the performance of the system. The development of an automated control package that will target commercial loads using optimal conditions will be developed for this project. Each load will be dynamically controlled such that operation is optimally handled for each individual load. Further research into HVAC systems will be targeted based on the findings of this study [22].

Potential benefits

Grid Fruit can optimize how commercial refrigeration defrosts such that the stores’ energy demand during peak hours is reduced up to 58% leading to an overall decrease each day up to 39%. This savings translates to nearly \$31,000 per year [22].

2.2 Application Goals

The scope of this thesis pertains to an ADR application that targets refrigeration systems, a class of HVAC-R systems, for further investigation and analysis. The client, Grid Fruit, is a 3rd party aggregator between utility services and their primary customer base, specifically commercial companies reliant on refrigeration. Grid Fruit wanted a design platform that is capable of intelligent control to provide ancillary services to the electrical grid using ADR integration as the novelty to the application.

1. The explicit goal is to reduce energy usage during peak hours while maintaining food quality.

2. An implicit goal is gain increased energy efficiency using optimization of energy usage.

With these two goals in mind a breakdown of hardware and software requirements is needed to better understand the scope of the research conducted for the desired design platform.

Hardware

The physical controller will adjust operational parameters (e.g., shifting temperature setpoints and/or turning the refrigeration compressors on/off) by receiving and responding to remote commands. In addition, this controller will store refrigeration systems' on/off status, internal temperature, and compressor cycles.

The hardware will directly receive OpenADR signals and respond to these signals.

The needed components including refrigerator(s) and software development toolkit(s) as well as a controller such as Particle Photon or Raspberry pi will be selected.

Software

The software developed will be hosted by the hardware controller

The software will turn commands received wirelessly into control signals that in turn adjust each refrigerator's operation.

2.3 Proposed Design platform

Given the open-ended nature of the design requirements the proposed design platform had significant potential for research topics. In the following section a breakdown of the initial platform will be given as well as the evolution of the research to better match real world requirements. Due to the dynamic nature of research during the events of COVID-19 and for better flow for this thesis additional work done by colleagues will be included but marked distinctly. This will provide a more complete image for the design platform.

Initial design

The initial proposal for the design platform was comprised of three core components: development pc, website user interface, and the raspberry pi as depicted in Figure 2-1. The development pc would contain the control code, which in turn would drive the logic that enables intelligent control. The raspberry pi would communicate with the development PC through a transmission control protocol (TCP) connection so that it may receive the proper control signals. The pi would then interpret these signals and drive a relay that controlled the compressor for the refrigeration unit. This forms the primary closed loop communication for the system. However, the last core for the system is the website user interface in which the end user to able to enable and customize ADR controls while monitoring their system. In the following sections an in-depth description and explanation of each core is given.

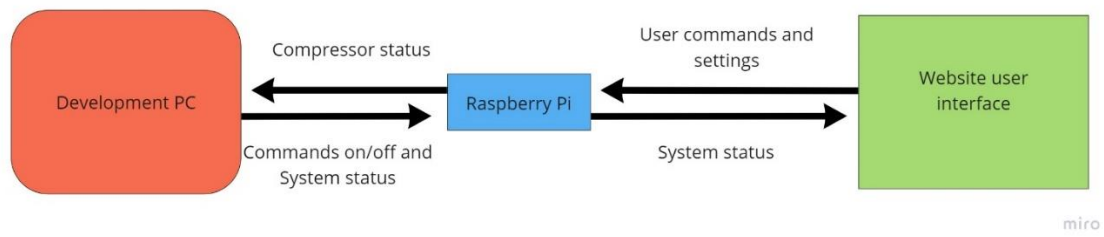


Figure 2-1: An illustration of the original design platform architecture.

Development PC

The development PC is the main computational device for the platform. It serves two primary functions: data acquisition and intelligent control. For data acquisition, there are two essential sets of data that need to be monitored: temperature and power consumption. Temperature is the main benchmark for success as the refrigeration functionality must be maintained for proper utilization of ADR technology. Temperature monitoring is accomplished using thermal couples linked to a National Instruments (NI) data acquisition device (DAQ). The thermal couples (TCs) measure the voltage difference between two metals, then by using a conversion that difference is given as a temperature in degrees Celsius. This information is then passed from the DAQ to the development PC.

At this point, the intelligent control comes into play. The development PC utilizes one program, LabVIEW, to receive, interpret, and communicate information. The temperature data is fed from the DAQ to this program which houses the primary control loop. There is a second control loop which is housed on the Raspberry Pi (RPi) which will be discussed in the following section.

The intelligent control for the refrigerator served four functions: signal processing, dummy control, ADR control, and communication. In the first iteration of the project, signal processing as well as both dummy and ADR control were developed by Vince Wilson. Thusly the specific details regarding those loops will be omitted. However, it is important to comprehensively understand what is happening for better understanding of the work that was done.

The signal processing is where the openADR extensible markup language (XML) file was parsed for pricing data, event start time, and event duration. This is also the step in which the temperature data from DAQ was passed into the program. From here both the XML signals and temperature signals would be passed to the control code which contained two modes: dummy control and ADR control.

For dummy control the refrigerator would operate as normal with the only difference being that the analogy circuit, that drove the compressor's, being replaced with a relay that is driven by the control code. Given a minimum and maximum operating setpoint the refrigerator would stay within these boundary conditions with the addition of a small dead band that would prevent stop/start at the boundaries, essentially replacing the analogy functionality of the refrigerator with a digital replacement.

The ADR control would operate very similarly to dummy control during normal hours of operation. However, the primary use of the ADR control was to incorporate a pre-cooling function before ADR events. This function was the focus of ADR application for this thesis and will be discussed in greater detail after the brief overview of the four functions of the control code.

Lastly, the communication between the development PC and the RPi would be handled. This communication function was achieved, as previously stated, with a TCP connection between the two devices. LabVIEW had a program which allowed for the remote deployment of LabVIEW code from the one device to another. However, this connection was primitive and could only deploy the program while it self's not being able to send and receive signals. Therefore, two programs would need to be developed:

Server-side TCP connection and Client-side TCP connection. The server-side TCP connection would create a listening function at the end of the control loop; this would listen for any programs trying to connect to the server specified at the internet protocol address (IP). The client-side program would be remotely deployed to the RPi and would query the server for a signal response; this would then tie the devices together. In this case once the computation was done during either dummy control or the ADR control stage, a simple Boolean command would be passed to the RPi for use in the secondary control loop.

As previously mentioned, there were two sets of data being collected temperature and power consumption. The former was directly incorporated into the control code as it was necessary for refrigeration functionality. The power consumption was tracked using a wattmeter only during times of experimentation. This data was saved on the device and then used to compare the results from the different sets of experiments.

The primary ADR function was created on the idea of load shifting in which the load during times of high-priced electricity is shut off and moved to a time of lower pricing. The concept pre-cooling was developed from this step. By knowing the schedule of DR events, either time-of-use or critical-peak-pricing, the control code can calculate how long and how cooled the refrigerator will need to be to ride through the event, allowing. This allows for the fridge to be off during the time of the event while still being able to maintain the quality of the food. This process was affected using an interpolation model replacement for the thermodynamics of the refrigerator. A real thermodynamic model was forgone as LabVIEW was unable to handle the differential

nature of thermodynamics. The interpolation was modeled using experiments detailed Chapter 3 of this thesis and results and methodologies will be discussed further there. While the interpolation and pre-cooling are relatively simple concepts, understanding them is critical to perception as how the control code was developed and how it operates in the contexts of this thesis.

Raspberry Pi

The RPi for the design platform served to drive the relay that controlled the compressor. For this task, a Boolean command would be sent to the secondary control loop, which was also known as the driver code. This code was composed in Python and kept separate from the LabVIEW program, that remotely targeted the pi for the purposes of fault protection; this will be discussed in greater detail under the research evolution section below. The driver code would then receive the Boolean command from the development PC utilizing the remotely deployed LabVIEW program. The LabVIEW program would write to a file which was constantly checked by the driver code which is how the communication between the two programs was handled. The driver code would then signal for the general-purpose input/output (GPIO) pins to write the pin to either be high or low depending upon the state that the refrigerator was desired to be in. High for “ON” and low for “OFF”. This pin setting would then either drive the relay to be on, which would complete the circuit and thus turn the compressor on, or it would turn the relay off, which interrupts the circuit and thus turns the compressor off. The closed loop communication for the design platform was completed and gave full control of the refrigerator to the administrative user, operating the control code.

Website user interface

The last core for the design platform was the website user interface meant for ADR customization on the behalf of the end-user, thus allowing for manual control, temperature tracking, refrigerator status, and custom ADR events. The website user interface was hosted locally on the RPi, which allowed for quick real time edits and bug fixing during the research. The architecture was achieved using four languages: SQL, Python, HTML/CSS, and PHP. SQL was the database which would store data and work as the primary place to check for updates and statuses for variables. Python allowed for the creation of a local host for the website as well as simple commands tied to buttons. HTML and the CSS files associated with each page were the languages responsible for the website creation. Lastly PHP allowed the SQL database to communicate with the website user interface, tying in the development PC to the website controls.

It is important to note that the bridge between the SQL and the LabVIEW interface has not been mentioned, this because the core for the thesis was ultimately absorbed by the other two cores and will be discussed in the section below. While the initial proposal did include a website user interface and while some degree of high functionality was achieved, the conclusion was researched that the work would be better suited for future research.

Research evolution

In the following section, an overview of the evolution of the initial design platforms research will be discussed, ultimately leading into the final design section.

Therefore, what decisions were made will be covered as well as a justification to provide a more comprehensive understanding of the choices taken.

The reason that LabVIEW was selected as the central point, to which all other aspects of the project would connect, was due to its highly versatile nature. The initial thought was that LabVIEW had the capability to run the intelligent control logic, drive the relay through the RPi, and host a website which could be edited in real time. This satisfied each of the three main cores and meant that the research would be centralized in one language. However, while this is true in concept, the features that made this possible were locked behind the professional version of the licensing, making it impractical for the design platform as cost was very important to the sponsor. Thus, while the logic core and the RPi core were able to be satisfied with some relatively minor changes, the third core, website control, was no longer viable within LabVIEW.

Based on this knowledge, further work on the website would be pursued using retrofitted HTML/CSS code discussed in the above section. The Smart Home Energy Management System (SHEMS) project from which this code was pulled allowed for easy integration with the research for this thesis. The website user interface was created and achieved much of the functionality that was desired for testing, allowing for the user to send manual controls to the refrigerator at any time via PHP scripts that would connect the website status to that of the status of variables on the SQL database. The SQL database would replace the status file that was being checked by the secondary control loop. However, the final piece that would close the loop was the bridge between the SQL database hosted on the RPi and the LabVIEW program on the development PC. It was at

this juncture that the issues with LabVIEW become apparent with the *coup de grâce* being the failure of LabVIEW to connecting to SQL. A Python library paired with drivers that allow for remote access would act as the bridge. This connection, however, could never be established even with rigorous troubleshooting and work arounds. Therefore, the decision was made switch from a LabVIEW- based centralized language to a more diverse and robust architecture. This new architecture will be discussed further in the final design section below.

Once the new architecture was selected a reevaluation of the research path was conducted, looking specifically for redundancies or work that was not necessary to a minimum viable product in the design platform. With the conversion of the LabVIEW sections into python it was found that a basic user interface (UI) could be created there as well. Finding that the Python code was more than sufficient to fully run the platform it was decided that the website was no longer academically interesting, and its functions were absorbed into the other cores, user interaction now either being automated via initial product installation or simply inserted in the program now run by the administrator side with the custom controls possible from the back end if desired. This would mean custom events, scheduling, minimum and maximum temperatures, and other variables that will be discussed in Chapter 3 - Methods could be changed from the administrative side.

It was decided that a secondary loop would operate the digital control the analog control selection from the RPi. Using a bypass circuit that would detect WIFI, if the connection was interrupted then the circuit would switch between the intelligent control and the analog control as visualized in Figure 2-2.

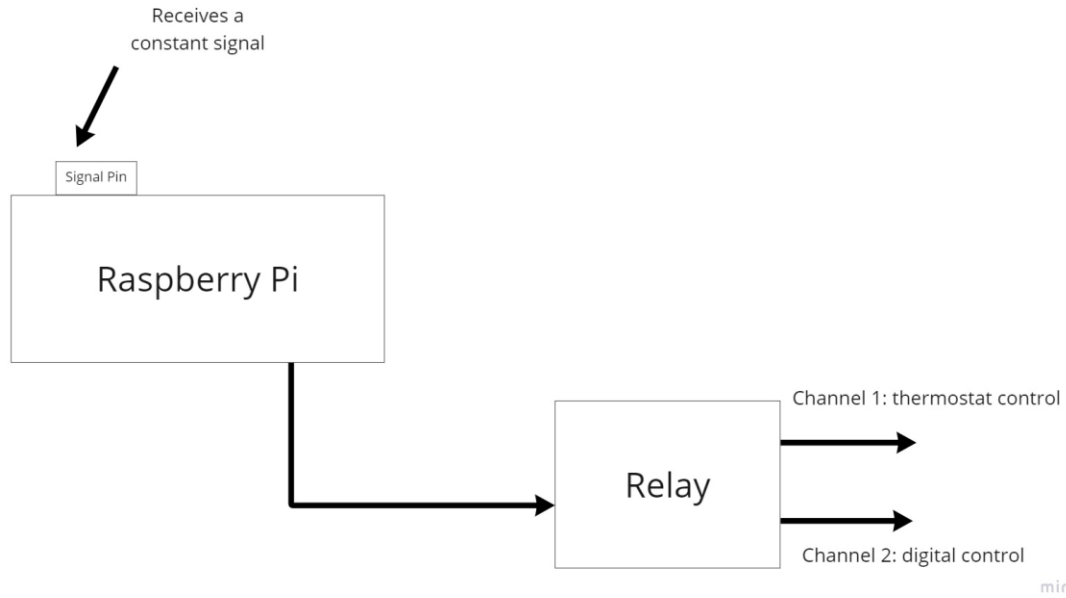
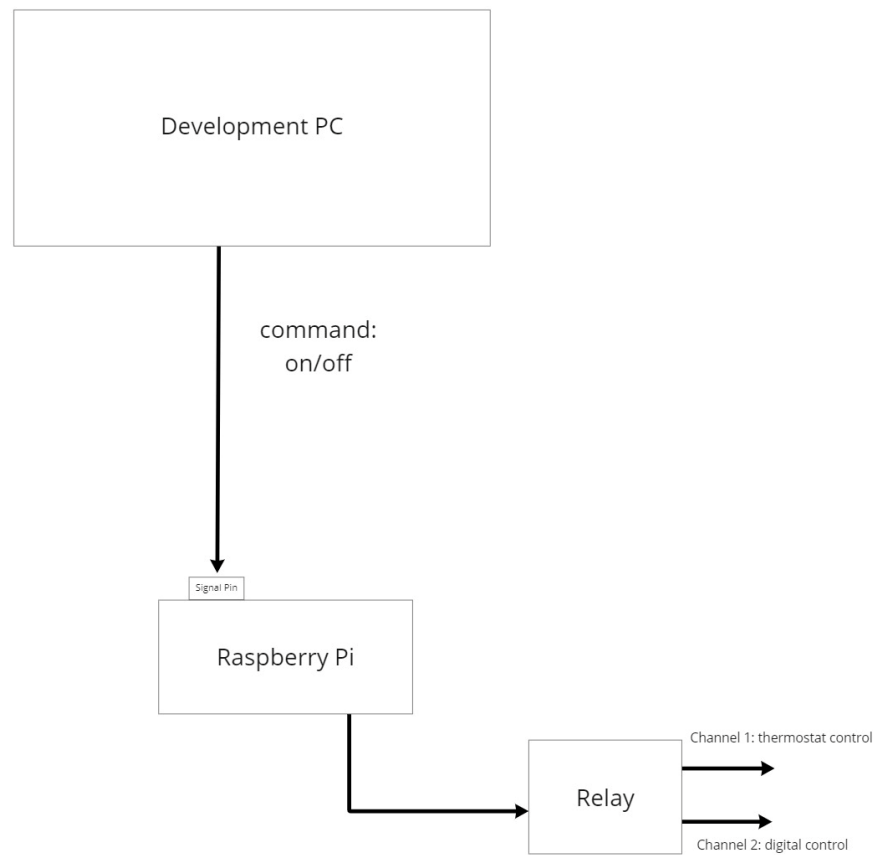


Figure 2-2: The raspberry pi receives a constant signal from the development pc. Upon receiving this signal, the Rpi interprets the string and then selects the proper channel. If this signal is interrupted a failsafe is enacted that will allow the refrigerator to operate normally.

Final design

The final design platform consisted of two main cores, all run from the development PC: the control logic core and the secondary control core. The control logic core runs much the same as it did in the initial design, the main difference being that the LabVIEW element has now been converted into Python. The XML and temperature variables have their own loops in python that extract the necessary components for operation as described in the initial design details. The control logic operates identically as it did previously, but with LabVIEW code now replaced with a more robust Python analog. This means that in the future more complex thermodynamic models may be used, as discussed more in Chapter 5 below. In the control logic code, the means to interact with and edit the variables are also present. This is also into which the website UI was absorbed. While not as aesthetically pleasing or easy to use, it still offers the customization necessary for in depth ADR use. Lastly the secondary control core has been heavily edited. Instead of writing to a file that has driver code constantly checking for the status of the refrigerator, it has been replaced with a very simple remote GPIO driver. This means that after the final calculations in the control code a simple high/low status is communicated to the RPi at the proper pin, driving the relay. The architecture, as visualized in Figures 2-3, has been greatly simplified which gives the design platform many less points of failure, overall increasing its reliability.



miro

Figure 2-3: A visualization of the design platform’s final architecture. The development pc dictates the commands sent to the raspberry pi. The Rpi interprets the signal to either allow for normal control or intelligent control.

CHAPTER THREE

MATERIALS, METHODS, AND EXPERIMENTAL PROCESS

In the following chapter, a breakdown of the materials used in the research design platform will be described. This breakdown will also serve as visual reinforcement for Chapter 2 allowing the reader to have a better understanding of the interconnections of the network previously detailed. The methods in which the setup was used will be described, as well as the experimental process for which results were gathered. This will serve as both a demonstration of capabilities of the research and a detailed explanation of how results were obtained.

3.1 Materials

Software

The two programs used in the final design of the research platform were: Excel by Microsoft and Python. Excel was used for logging data as well as recording and visualizing results. Python was the primary language upon which all control logic was built, while also handling the automation of recording relevant data for Excel.

Hardware

The hardware requirements will be given in the following pictures below Figure 3-1 to Figure 3-12.

3.2 Design platform methods

In this section, an ADR event will be narrated to give a better understanding of the methods used to implement the system described in Chapter 2 of this thesis. It will also illuminate how the materials both hardware and software work with one another to



Figure 3-1: The main refrigeration platform was the FR3201, selected for its analog nature which made modifications easy, as well as its size, which balanced accurate scaling of thermodynamic modeling and accessibility due to weight.



Figure 3-2: A look at the internal components of the platform to give a complete image of the system. This includes the thermocouples for tracking temperature and the wiring for compressor control.



Figure 3-3: Highlighted in blue is the K type thermocouple for air temperature readings. This acted as the primary device that would determine system response. Pointed to with a red arrow is another K type thermocouple used for reading surface temperatures.



Figure 3-4: A surface thermocouple used to record the temperature of the evaporator. A great use for having additional thermocouples is error checking as the values of the thermocouple reads should be relatively close to one another.



Figure 3-5: A surface thermocouple used to record temperature at the bottom of the refrigerator. While not directly used to influence system responses, these thermocouples helped to build an understanding of the system thermodynamics.



Figure 3-6: A depiction of the connection from relay to compressor circuit, acting as a substitute for the analog thermostat of the original system. This is achieved with a bypass circuit that will allow for the use of either the analog or digital system.

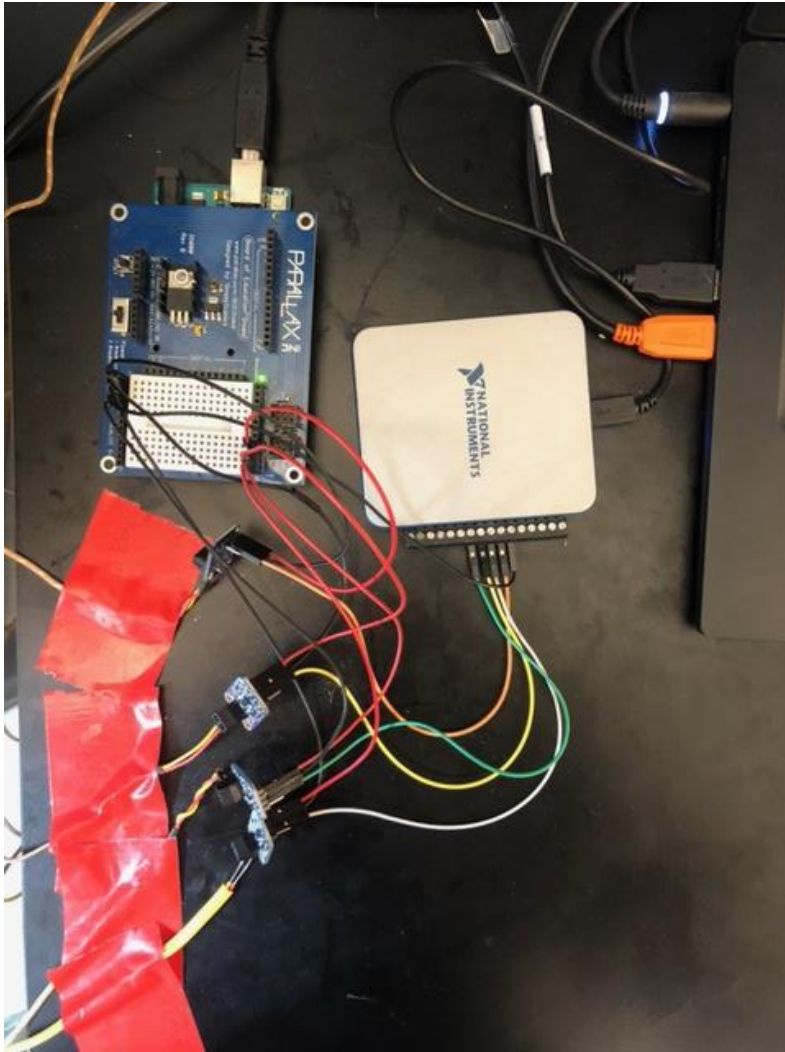


Figure 3-7: A full image of the temperature collection system, including the DAQ



Figure 3-8: The DAQ model USB-6001 acted as the primary data collection device connecting the thermocouple readings to the development PC.

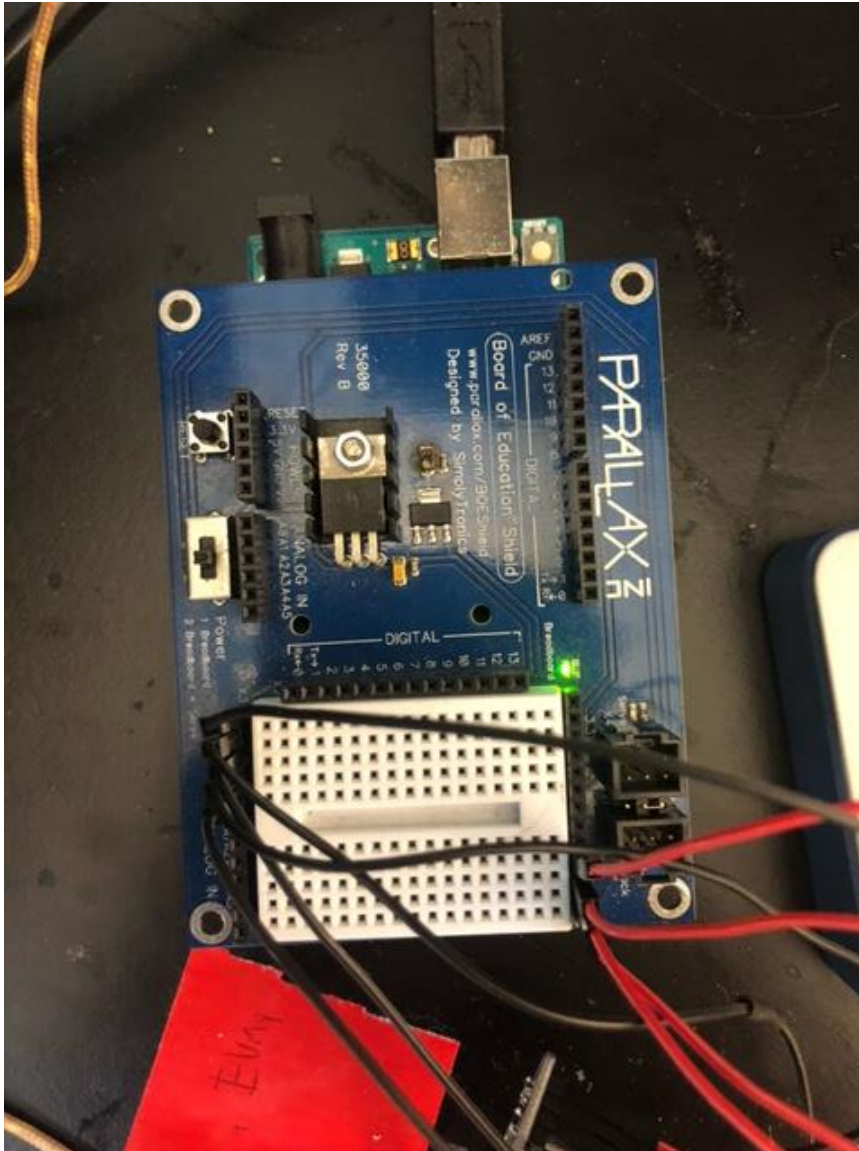


Figure 3-9: This is a small board used to power the amplifiers.

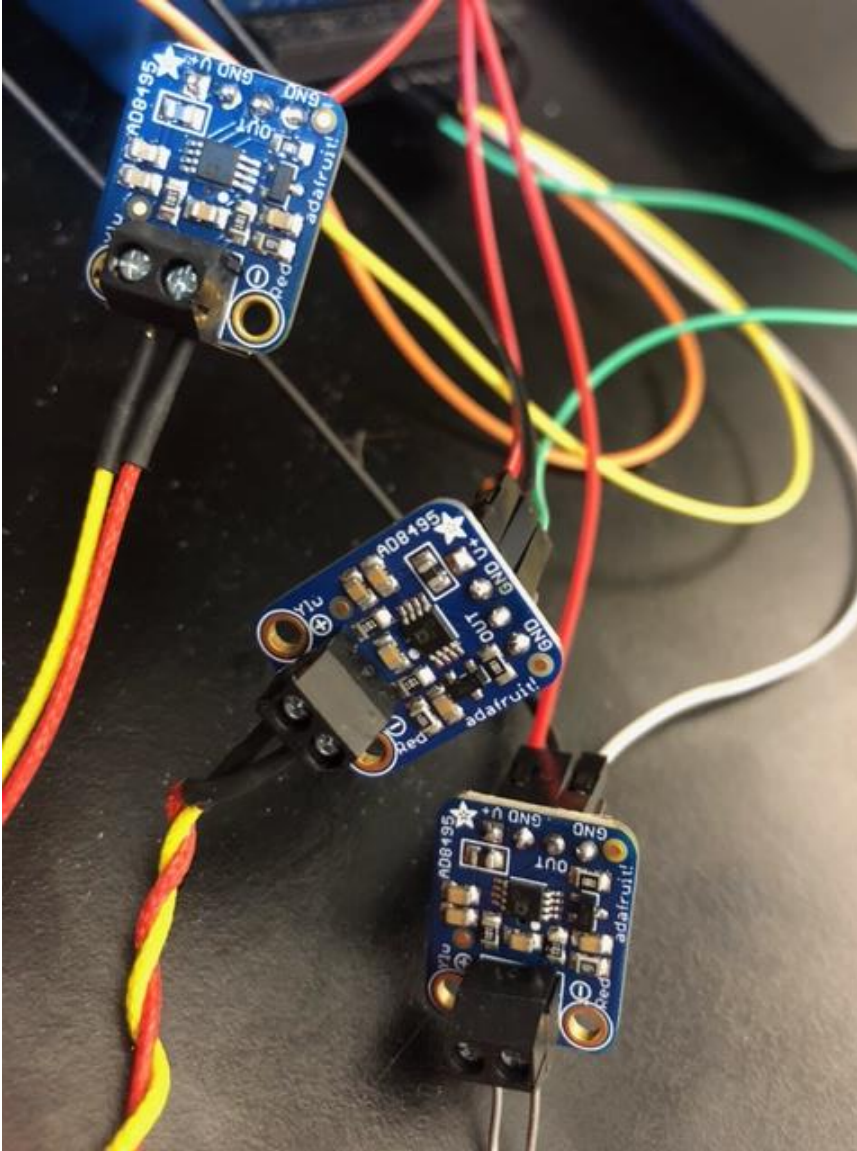


Figure 3-10: The analog K type thermocouple amplifier, model AD8495, used to allow for readable output from the thermocouples to the DAQ.

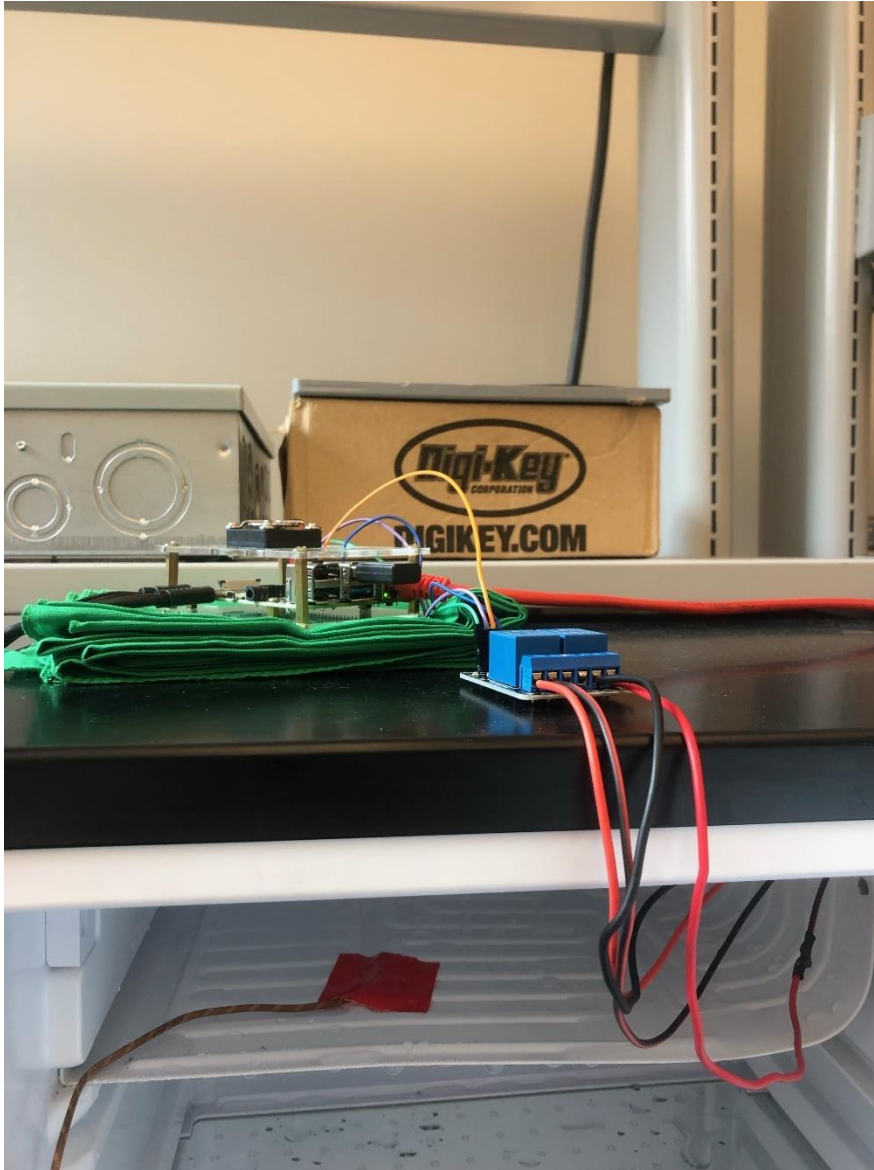


Figure 3-11: The Raspberry Pi 4 model B (Rpi) worked as the controller for a relay which in turn drove the compressor. If a fault was detected the, Rpi was capable of automatically switching to the analog circuit to preserve the integrity of the system.

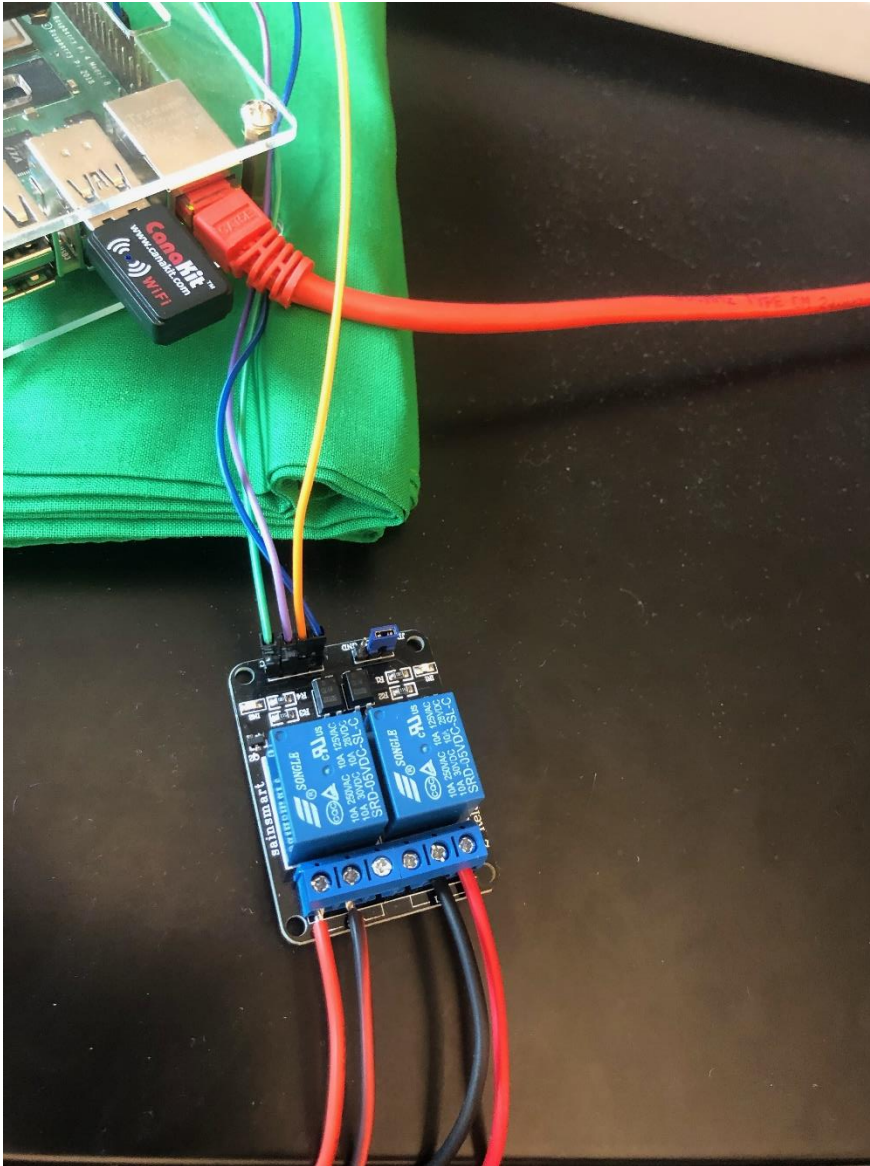


Figure 3-12: A two-channel relay used to control the compressor for the refrigerator.

achieve the desired behavior. Time of Use (TOU) or peak vs off peak is a time block that denotes higher or lower electricity usage. There are two major seasonal TOU blocks that are considered: during the winter months from 5 AM-8 AM, inducing an optional time block of 5 PM to 8 PM, and during the summer months from 12 PM to 8 PM. These events are static and will be referred to as static time of use (STOU) events. OpenADR is a standard for which ADR events are scheduled and communicated to clients from utilities or aggregators. These events will be considered as dynamic time of use (DTOU) events. Both events operate under the principles of Critical Peak Pricing (CPP) in which electricity rates change based on demand.

Implementing CPP with static and dynamic time of use

STOU events are programmed to be seasonal and are not sent via XML files but are determined based on the time of the year and physical location. Thus, pre-cooling conditions and start time for these events are determined daily based on the time and duration of the event. While these parameters are calculated and up until the time of the pre-cooling event the system will operate as a standard refrigerator. During the pre-cooling event the refrigerator will cool to the desired pre-cooling temperature, considering the minimum operating boundary. In this case the desired temperature is hard set to be substantially below the 4.4 °C requirement at 1 °C for ease of operation. Once the pre-cooling event is done the fridge will transition to the STOU event in which the fridge will turn off and monitor the temperature to avoid violating the maximum boundary. After the STOU event the refrigerator will return to standard operation. If the

unit is unable to ride through the entire event, then the compressor will return to normal operation.

For the purposes of this thesis openADR files were substituted using locally generated files compliant with standards that are mandated for opted in clients. This was done as gaining access to live services was not practical for academic research.

An openADR XML file is received from the local utility company or aggregator. The file is then read by the control code and parsed for the relevant date-times and pricing information. The pricing information is archived for later use to determine savings based on system performance during the ADR event. The date-time information is then used to generate unique chunks of time that determine the standard operating procedures for the chunk. These chunks being: pre-event and post-event which will be considered standard time, pre-cooling, and DTOU/ CPP event. DTOU and CPP events are also known as an ADR event. The pre-cooling block is determined based on the current temperature of the refrigerator and time it will take to cool to the lower temperature boundary which will allow the system to ride out the ADR event, while also staying within the minimum and maximum boundaries set by the customer. It is important to note that there is a pre-cooling event for both STOU and DTOU events. The ADR event's time block is set by the information contained in the XML file, specifically start-time and duration. These time blocks are then checked against the STOU event(s) that may be happening. If the DTOU event takes place during the STOU event, then the event is ignored as the refrigerator will already be operating as desired. If there is some overlap, there will be no pre-cooling cycle, but the extra time will be account for until the

maximum temperature boundary is reached. If there is no overlap, then the TOU and openADR event will operate individually.

From here the refrigerator will operate in standard refrigeration mode using the control code logic until the determined time for the pre-cooling event. During the pre-cooling event one long cooling cycle will begin dropping the fridge temperature to that of the predetermined optimal pre-cooling temperature. If this temperature is colder than that of the minimum temperature boundary, then the cooling cycle will stop once the boundary is hit. Only starting again once the dead-band is over. This may result in more than one cooling cycle but should preserve the quality of product inside the refrigerator.

The reason pre-cooling is done is to use energy during times of low price to make the refrigerator slightly colder than normal. This will allow the system to be off during the event, or times of high price, resulting in monetary savings while preserving the quality of food. Pre-cooling start time was calculated using linear interpolation between several pre-recorded cooling curves. These curves correspond to the three loading conditions: no load, half load, and full load. The loading conditions each have unique cooling-rates based on the mass in the system; therefore, three conditions were selected to have a more accurate pseudo-thermodynamic model.

Pre-cooling ends, the control logic will begin operating in ADR event mode, which means that the compressor will be shut off and the temperature will be closely monitored. If the temperature stays within the maximum temperature boundary, then the compressor will remain off for the duration of the event. If the maximum temperature boundary is broken, then the system will detect a fault. Once this fault occurs, the

compressor will automatically turn on to correct the temperature and the system will return to standard operation.

3.3 Experimental process

Baseline experimentation of analog refrigerator

Baseline cooling testing

This experiment was designed to measure the time and energy consumption the system took to cool from room temperature to refrigeration temperature. While different from standard operation, this was done to measure cooling performance in the system, a useful tool in determining efficiency of cooling cycles. Refrigeration temperatures were determined according to FDA guidelines and common commercial practice, the FDA recommending that commercial refrigeration be kept below 4.44 °C [23]. Common commercial practice is to overcool the system, presumably to provide a buffer for the 4.44 °C limit [24]. There were three load conditions all of which were run using the below steps: (1) The no load test in which there was no product introduced to the system, (2) the half load test in which sixteen twelve-ounce water bottles were introduced and (3) the full load test which incorporated thirty-two twelve-ounce bottles of water. These conditions will be used for each of the below experiments unless otherwise mentioned.

1. Measure internal air temperature until steady state temperature was in the room temperature operating range, 20-23 °C
2. Set refrigeration setting to the coldest setting supported by the analog thermostat

3. Once the internal air temperature reached steady state below the recommended refrigeration temperature and oscillated between 3-4.44 °C the test was concluded.
4. All temperature and power results were measured by the DAQ and wattmeter respectively.

Baseline steady state testing

This experiment was set up to determine power consumption over twenty-four hours and is designed to estimating total power consumption over a simulated year. The analog setting that corresponds to the proper temperature for optimal refrigeration was determined. While not precise the closest setting that reached the 4.4 °C limit was picked.

1. Measure internal air temperature until steady state temperature was in the optimal refrigeration range.
2. Set recording devices to capture twenty-four hours of data
3. Upon the expiration of the twenty-four-hour period, all temperature and power results that were measured by the DAQ and wattmeter, respectively, were archived.

Final design experimentation

Cooling efficiency

This test was designed to compare the measured time and energy consumption of the digital system to that of the previously measured analog system.

1. Measure internal air temperature until steady state temperature was in the room temperature operating range, 20-23 °C

2. Set refrigeration setting to the 4.4 °C limit
3. Once the internal air temperature reached steady state below the recommended refrigeration temperature and oscillated between 3-4.44 °C the test was concluded.
4. All temperature and power results were measured by the DAQ and wattmeter respectively.

Steady State efficiency with digital logic

This experiment was set up to determine power consumption over twenty-four hours to estimate total power consumption over a simulated year. The minimum temperature was set to 3 °C, the optimal temperature was set to 4 °C and the maximum was set to 5 °C. The optimal temperature was selected as 4 °C, because this accounted for some overcooling that commercial users may incorporate. The maximum was allowed to be 5 °C as the recommended was 4.4 °C: however, it was noted that there was some oscillation that occurred at steady state temperatures of roughly ± 1 °C. This meant that to prevent a stop-start event at steady state a 1 °C buffer would be needed. The change to the maximum temperature was deemed acceptable as the 0.6 difference is less than +1 °C, which meant that it was likely to trigger the cooling cycle of the system before the system truly reached 5 °C, the cutoff for perishable food being considered spoiled. This, in conjunction with the dead band, allowed for acceptable refrigeration behavior.

1. Measure internal air temperature until steady state temperature was in the optimal refrigeration range.
2. Set recording devices to capture twenty-four hours of data

3. Upon the expiration of the twenty-four-hour period, all temperature and power results that were measured by the DAQ and wattmeter, respectively, were archived.

Steady State efficiency with improved digital logic

This test was done to measure the impact of the improvements made to the digital system. By upgrading the resolution of the thermocouples, having a smaller dead band, and smaller temperature range the energy consumption may present as lower while increasing the quality of the temperature monitoring

1. Measure internal air temperature until steady state temperature was in the optimal refrigeration range.
2. Set recording devices to capture twenty-four hours of data
3. Upon the expiration of the twenty-four-hour period, all temperature and power results that were measured by the DAQ and wattmeter, respectively, were archived.

Steady State efficiency with STOU (Summer and Winter)

This experiment was set to test the efficiency of the STOU events. As previously mentioned, the intelligent control works similarly to the standard digital logic. Therefore, this experiment will run similarly to the above steady State efficiency with digital logic test.

1. Measure internal air temperature until steady state temperature was in the optimal refrigeration range.
2. Set recording devices to capture twenty-four hours of data

3. Analyze STOU event and prepare for either summer or winter
 - a. Automatically response to STOU event
4. Upon the expiration of the twenty-four-hour period, all temperature and power results that were measured by the DAQ and wattmeter, respectively, were archived.

Steady State efficiency with CPP

This experiment was set to test the efficiency of the STOU events coupled with the CPP events. These experiments will work similarly to the above STOU test. However, this experiment is looking specifically for a comparison between how much the STOU and the CPP overlap in terms of efficiency and time. Therefore, the load condition will be set to full load while the CPP event is taken. If the CPP event is sufficiently responded to then both a CPP event and TOU event can be responded to depending on the time of both.

1. Measure internal air temperature until steady state temperature was in the optimal refrigeration range.
2. Set recording devices to capture twenty-four hours of data
3. Analyze CPP event and prepare for pre-cooling
4. Upon the expiration of the twenty-four-hour period, all temperature and power results that were measured by the DAQ and wattmeter, respectively, were archived.
5. Analyze the CPP event to measure the time that can be ridden through. Apply this to the pricing data looking for the possible times that can be load shifted

CHAPTER FOUR

RESULTS

Figures 4-1 through Figure 4-34 will be the results from the experiments.

4.1 Energy Efficiency

Cooling efficiency

Figures 4-1 through 4-12 are the cooling efficiency results from the previous sections

Steady State efficiency

Figures 4-13 through 4-24 are the cooling efficiency results from the previous sections
description

4.2 Monetary savings

Pricing data

Figures 4-25 through 4-26 are the cooling efficiency results from the previous sections
description

Initial digital with Static TOU operation

Figures 4-27 through 4-30 are the cooling efficiency results from the previous sections
description

Improved digital with CPP operation

Figures 4-31 through 4-34 are the cooling efficiency results from the previous sections

Improved Digital refrigeration operation

Figures 4-1 through 4-12 are the cooling efficiency results from the previous sections
description

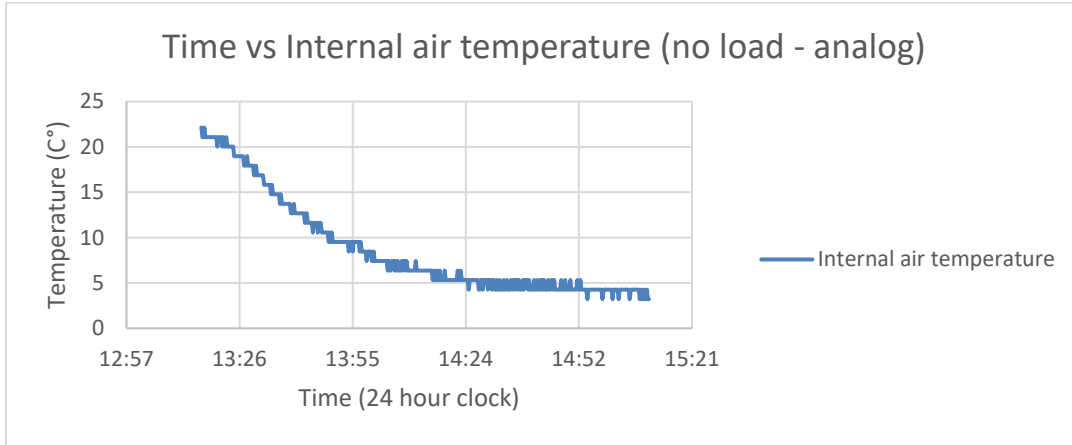


Figure 4-1: This graph shows that there are many temperature fluctuations for the original thermostat's initial cooling to steady state. Also shown is the time it took to reach steady state, approximately 1 hour.

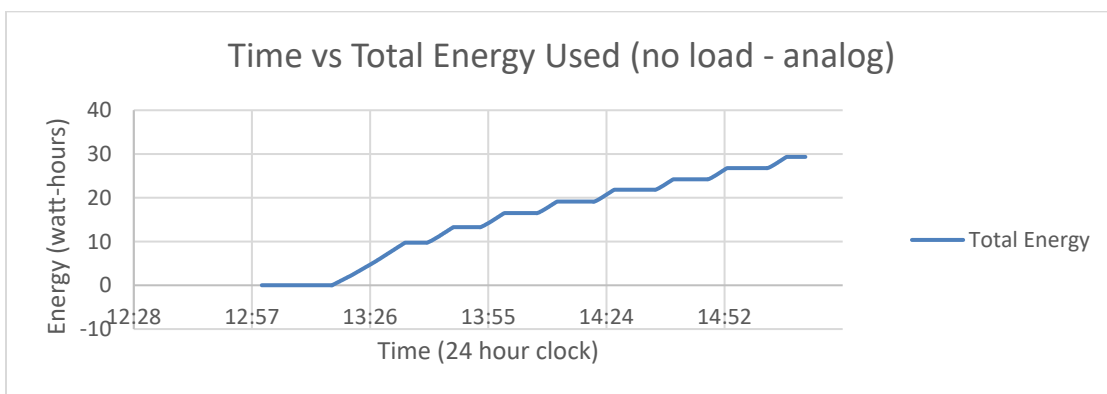


Figure 4-2: This energy graph shows the power consumption of each cooling cycle including the total for the system, approximately 30 watt-hours.

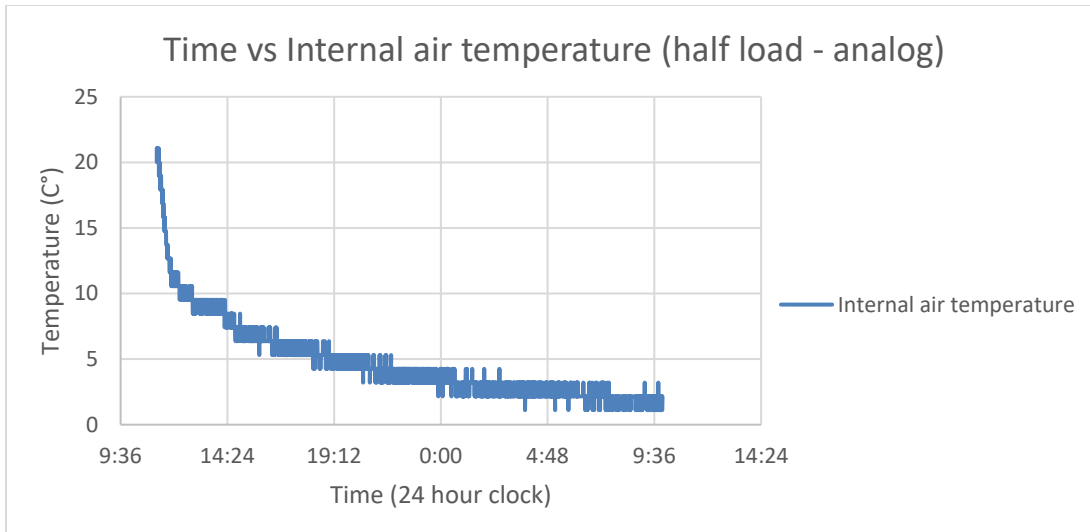


Figure 4-3: This graph shows the many cooling cycles of the original thermostats cooling to steady state. Also shown is the time it took to reach steady state, approximately twenty-four hours.

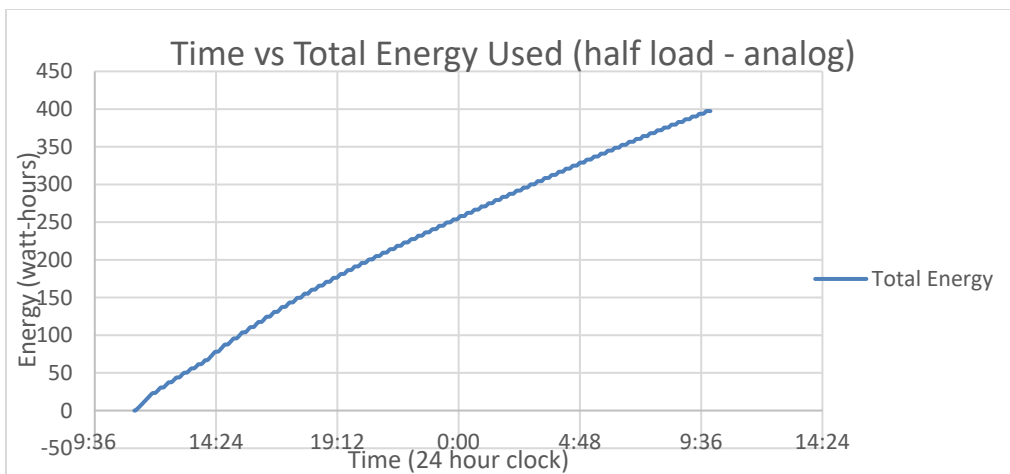


Figure 4-4: This energy graph shows the power consumption of each cycle including the total for the system, approximately 400 watt-hours.

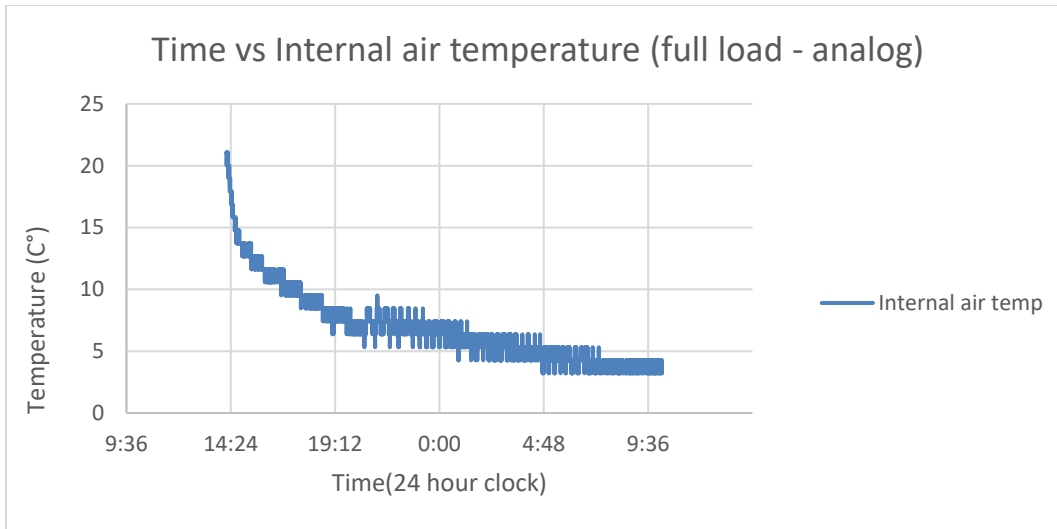


Figure 4-5: This graph shows the many cooling cycles of the original thermostats cooling to steady state. Also shown is the time it took to reach steady state, approximately nineteen hours.

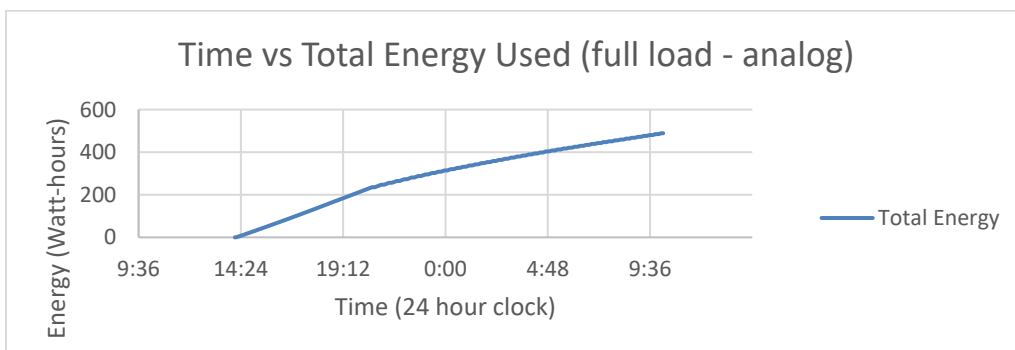


Figure 4-6: This energy graph shows the power consumption of each cycle including the total for the system, approximately 500 watt-hours. Note how the first hours resulted in a longer cooling cycle.

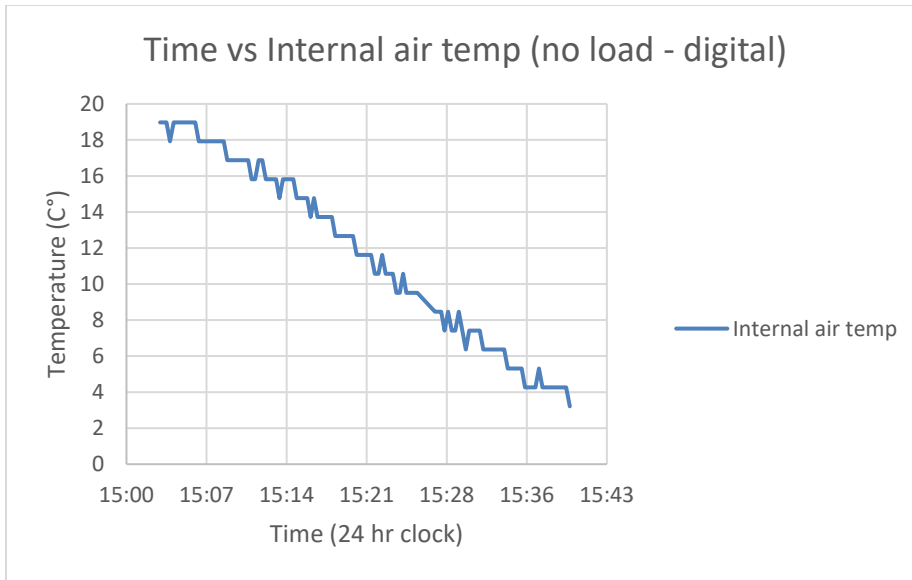


Figure 4-7: This graph shows comparatively few temperature fluctuations indicating one longer cooling cycle of, approximately 40 minutes.

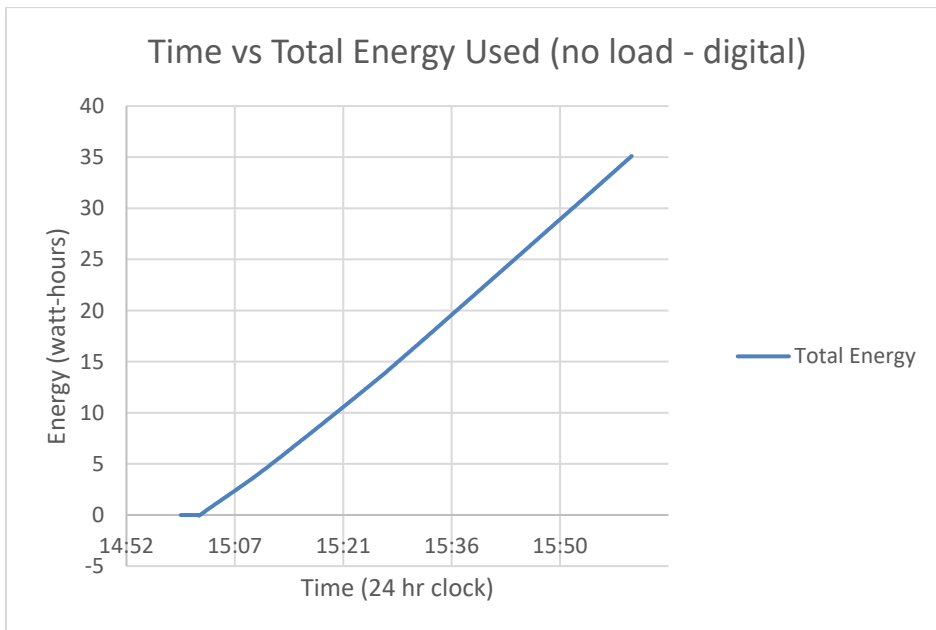


Figure 4-8: This graph shows the energy consumed for the cooling cycle, approximately thirty-five watt-hours.

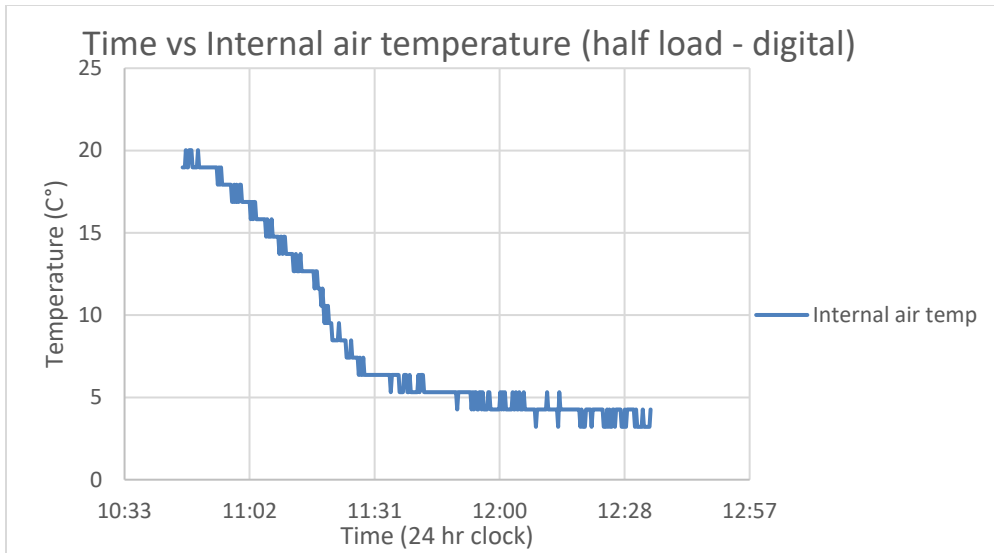


Figure 4-9: This graph shows a steady drop in temperature, indicating a longer cooling cycle or cycles.

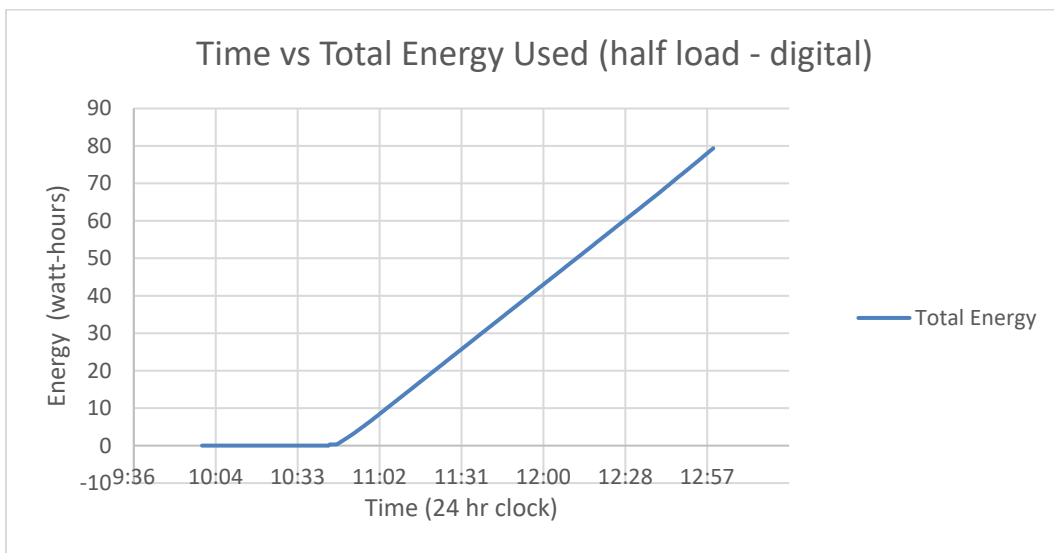


Figure 4-10: This graph shows that there was one long cooling cycle using approximately, 80 watt-hours.

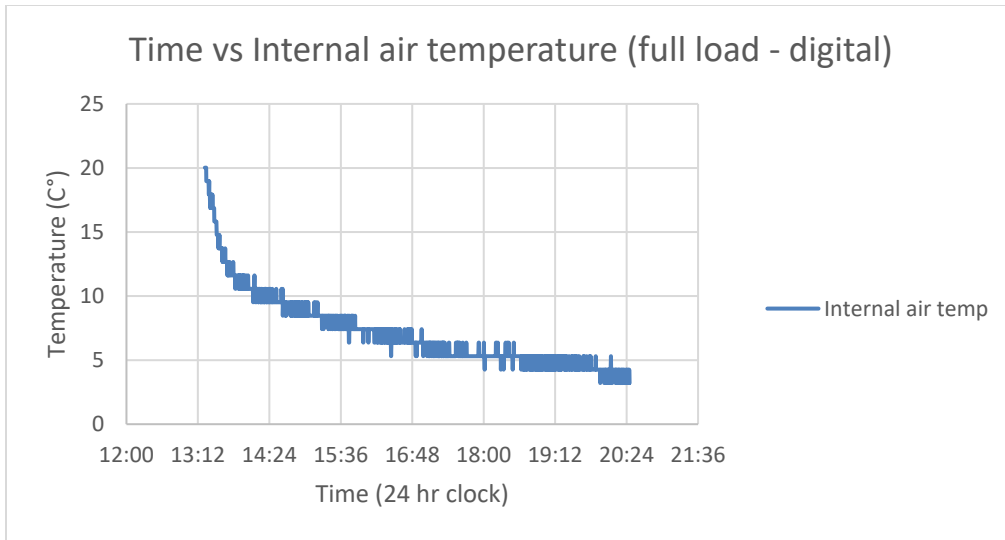


Figure 4-11: This graph illustrates the steady cooling of the refrigerator at a rapid rate. This indicates a longer cooling cycle, approximately eight hours.

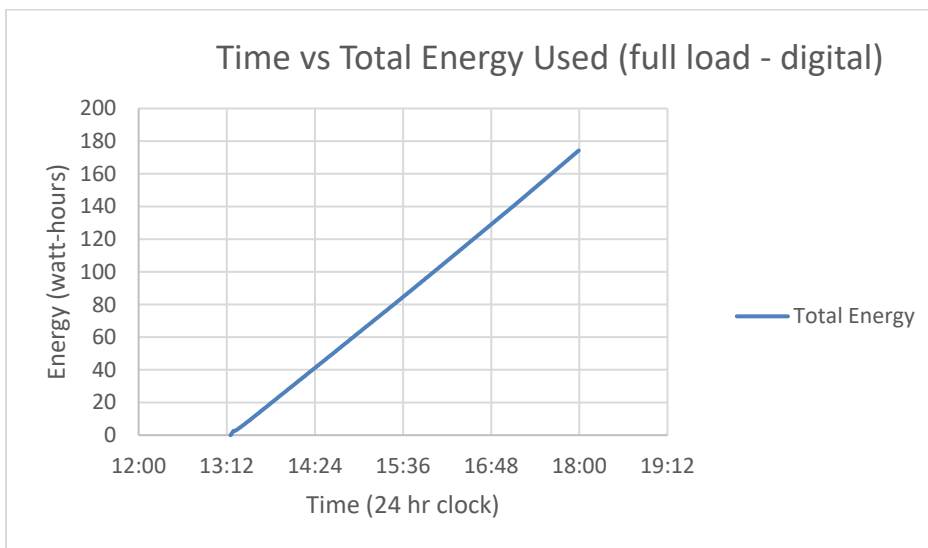


Figure 4-12: This graph shows the energy consumption of cooling cycle, approximately 170 watt-hours.

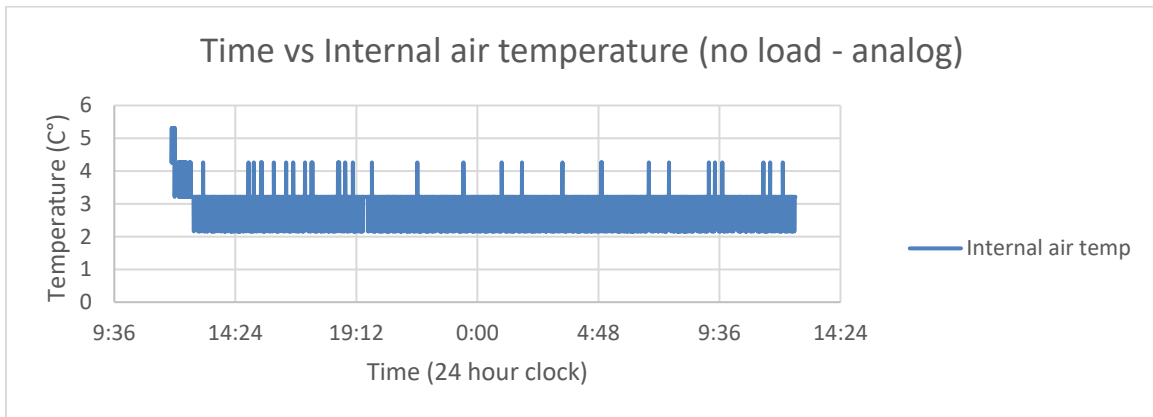


Figure 4-13: Shows the steady state temperature as consistent with some spikes between cooling cycles.

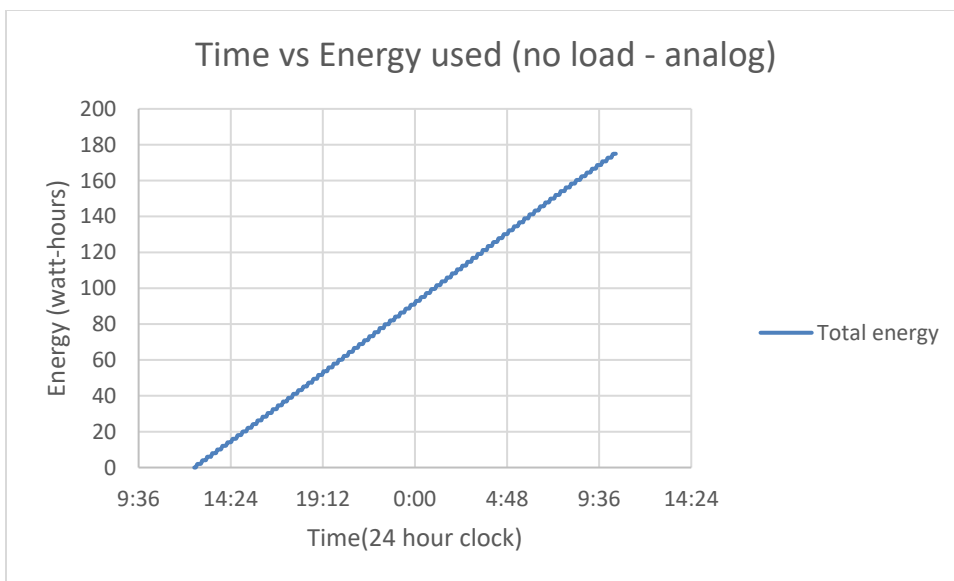


Figure 4-14: This graph shows there are many cooling cycles and the total energy used in a day, approximately 175 watt-hours.

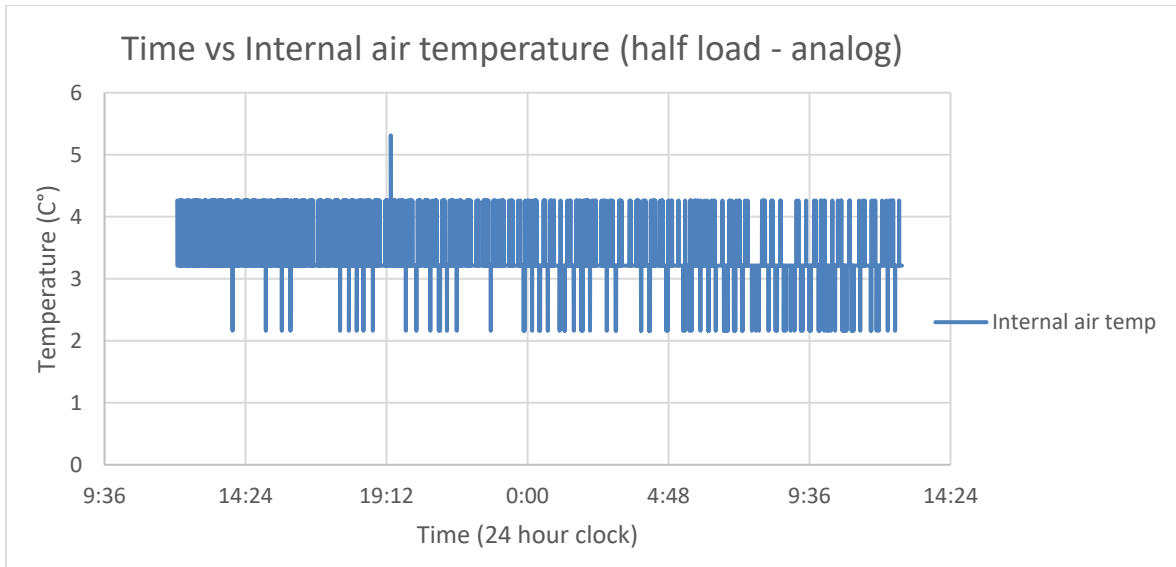


Figure 4-15: This graph shows the steady state temperature as less consistent than no load with a higher average temperature.

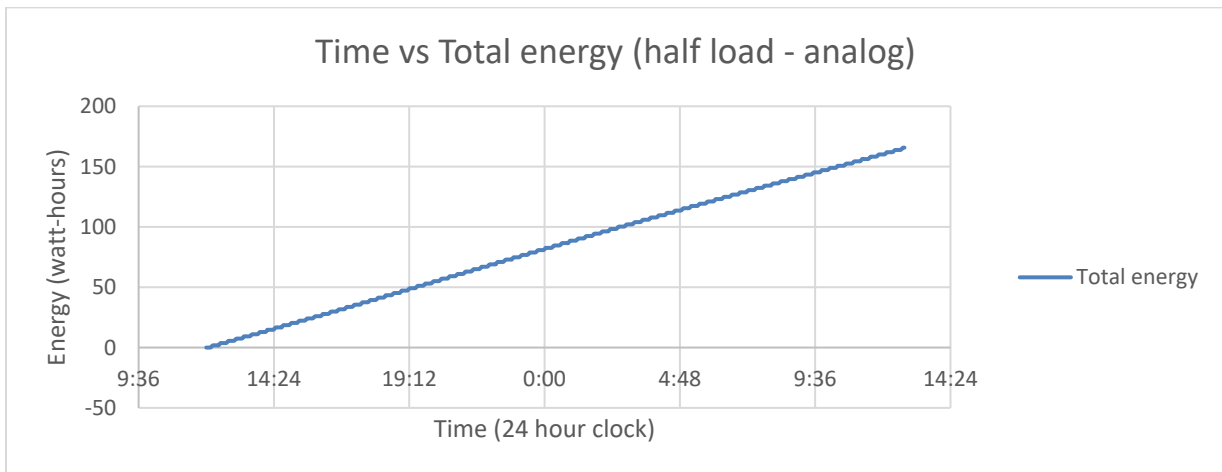


Figure 4-16: This graph shows the energy consumed over 24 hours, approximately 164 watt-hours.

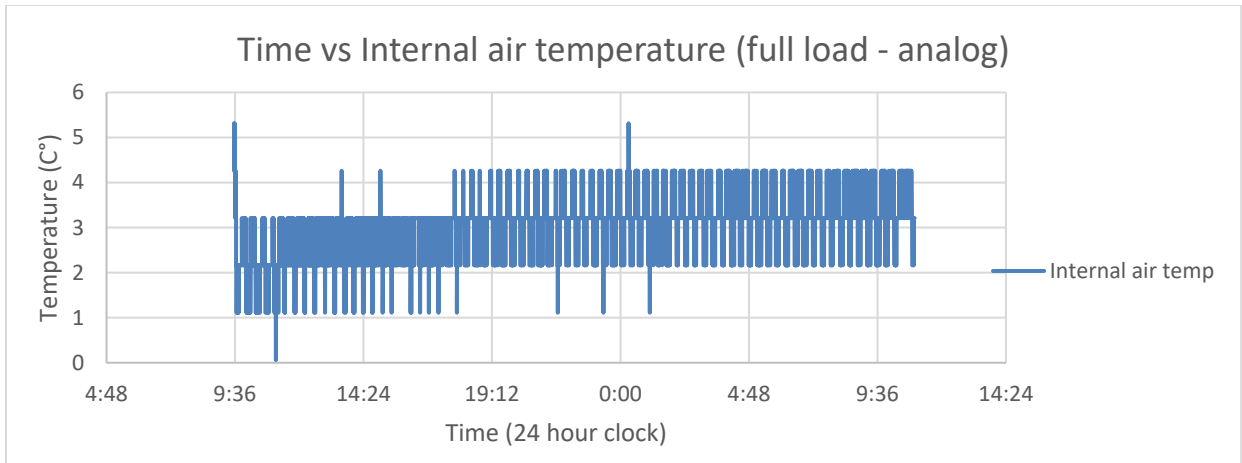


Figure 4-17: This graph shows the temperature fluctuations throughout the hours of operation in a day with full load conditions.

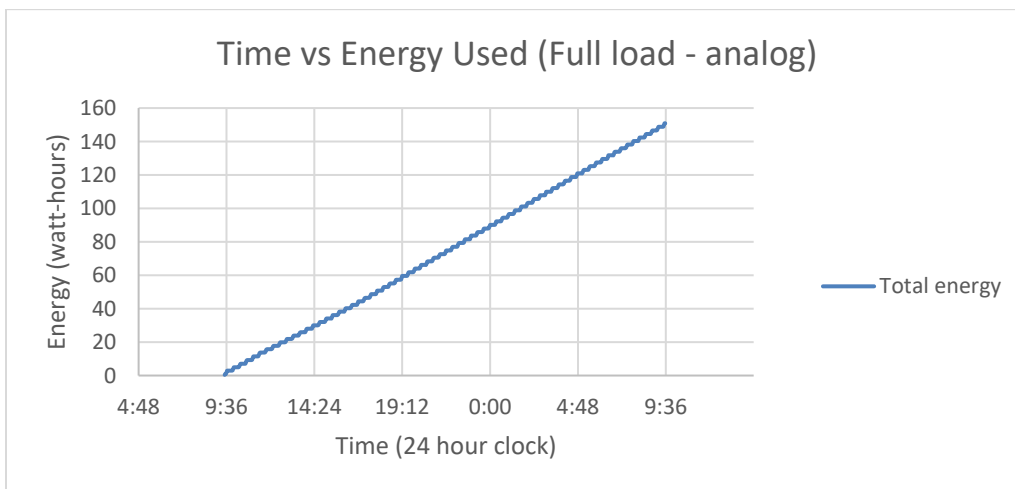


Figure 4-18: This graph shows the power used over twenty-four hours being approximately 149 watt-hours.

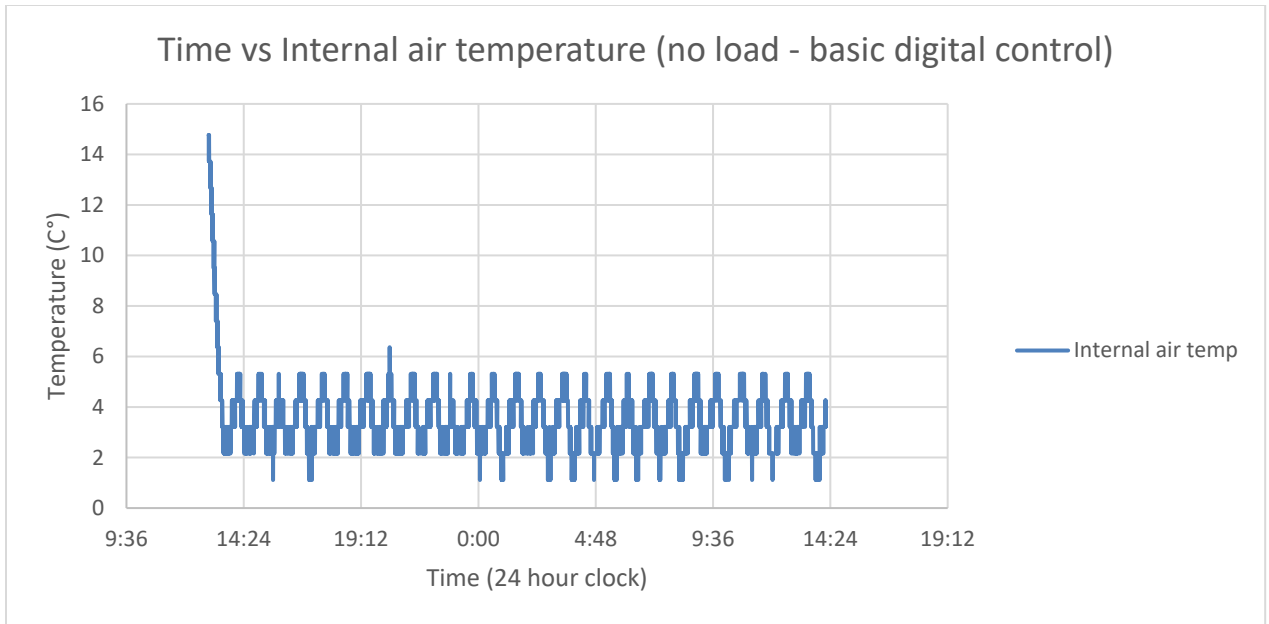


Figure 4-19: This graph shows a wider temperature range and distinct cooling cycles throughout the day.

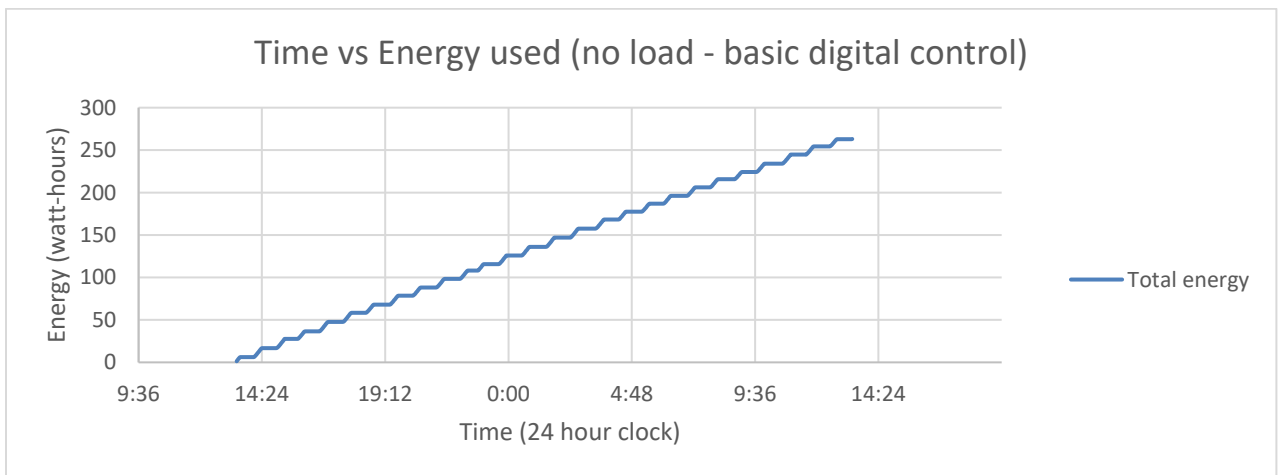


Figure 4-20: This graph shows the energy usage through the day, approximately 262 watt-hours.

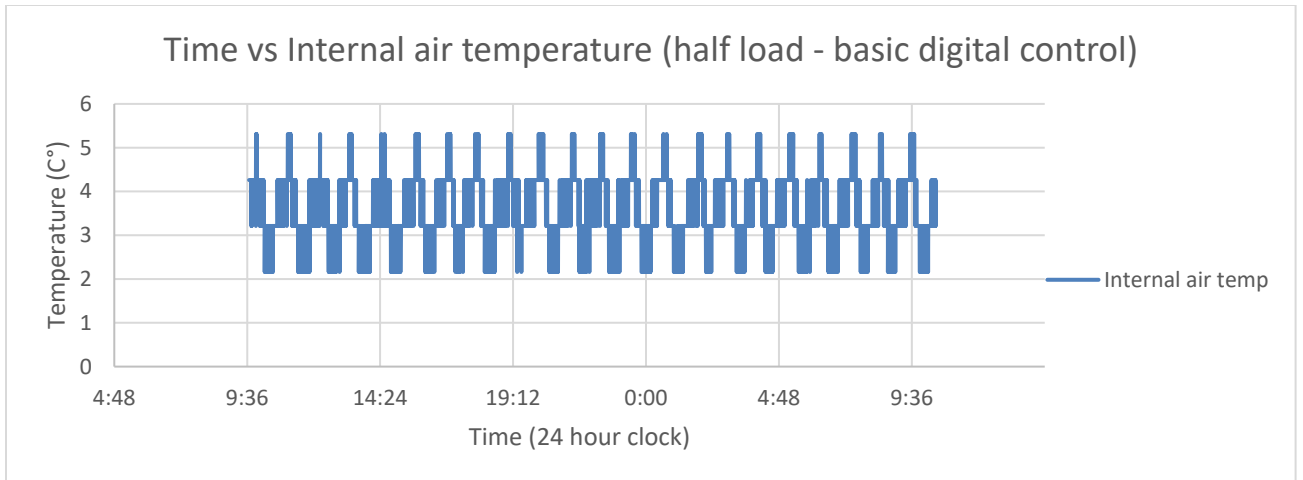


Figure 4-21: Show cases the increased amount of cooling cycles proportional to the load added.

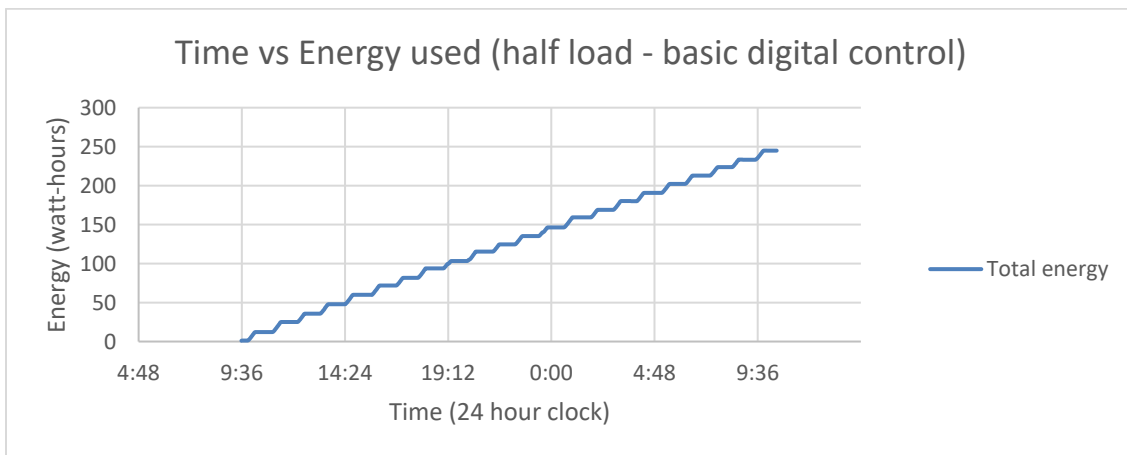


Figure 4-22: This graph shows the energy used during the day as being approximately, 245 watt-hours.

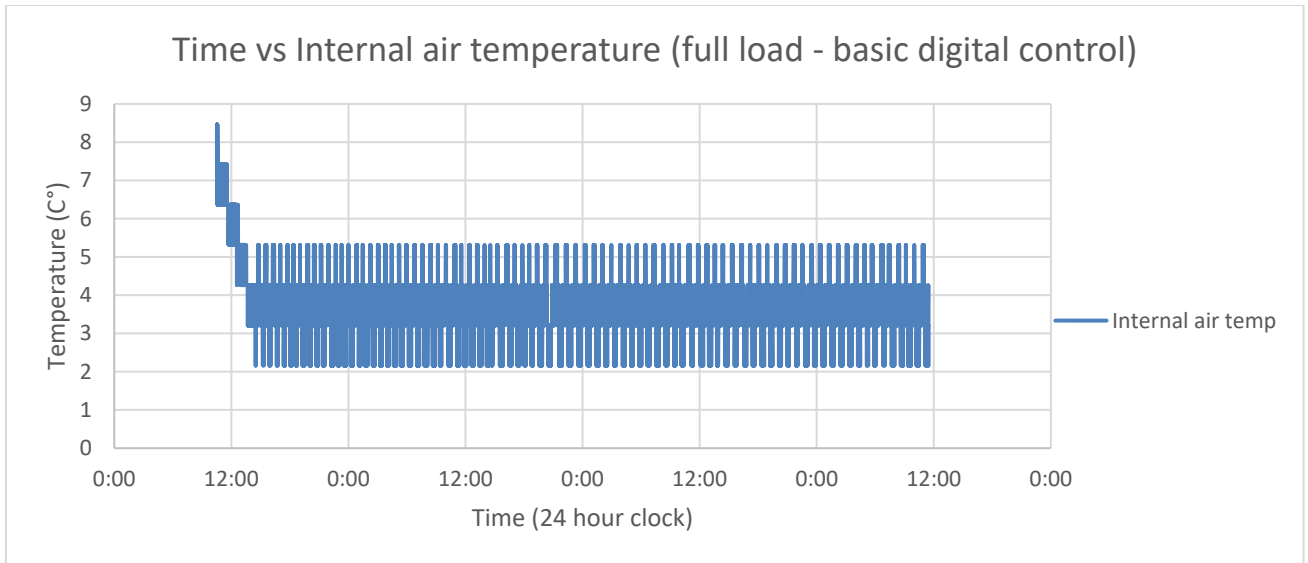


Figure 4-23: This graph shows more cooling cycles with a wider temperature range based on the amount of load in the unit.

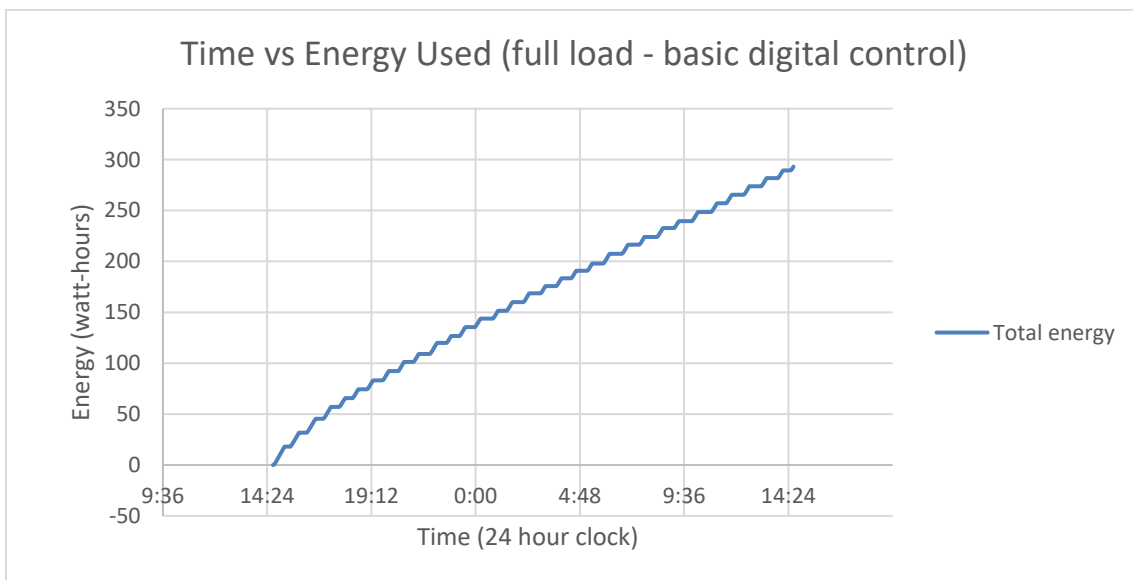


Figure 4-24: This graph shows the energy usage during a twenty-four-hour period being approximately 290 watt-hours.

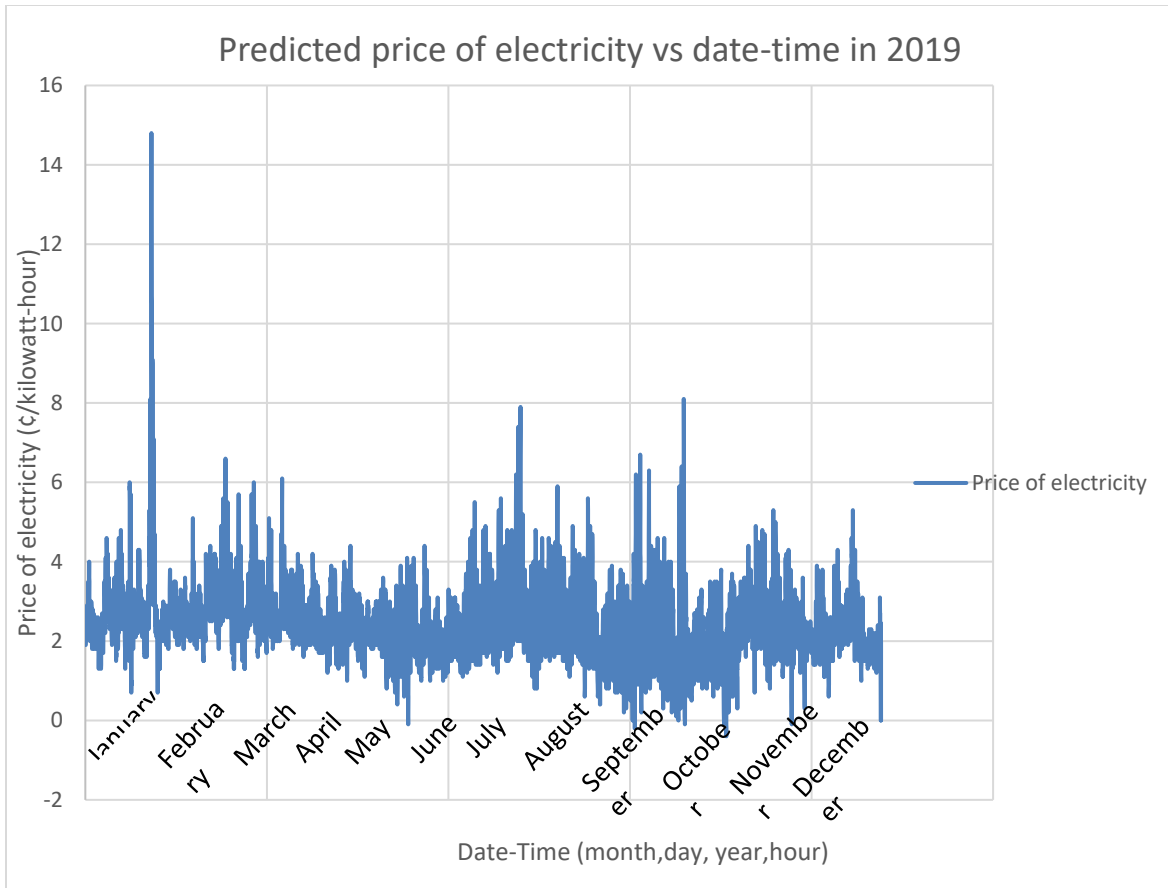


Figure 4-25: Pricing data taken from, Comed an Exelon company, from 2019 [24]. This shows the predicted rates twenty-four hours in advance.

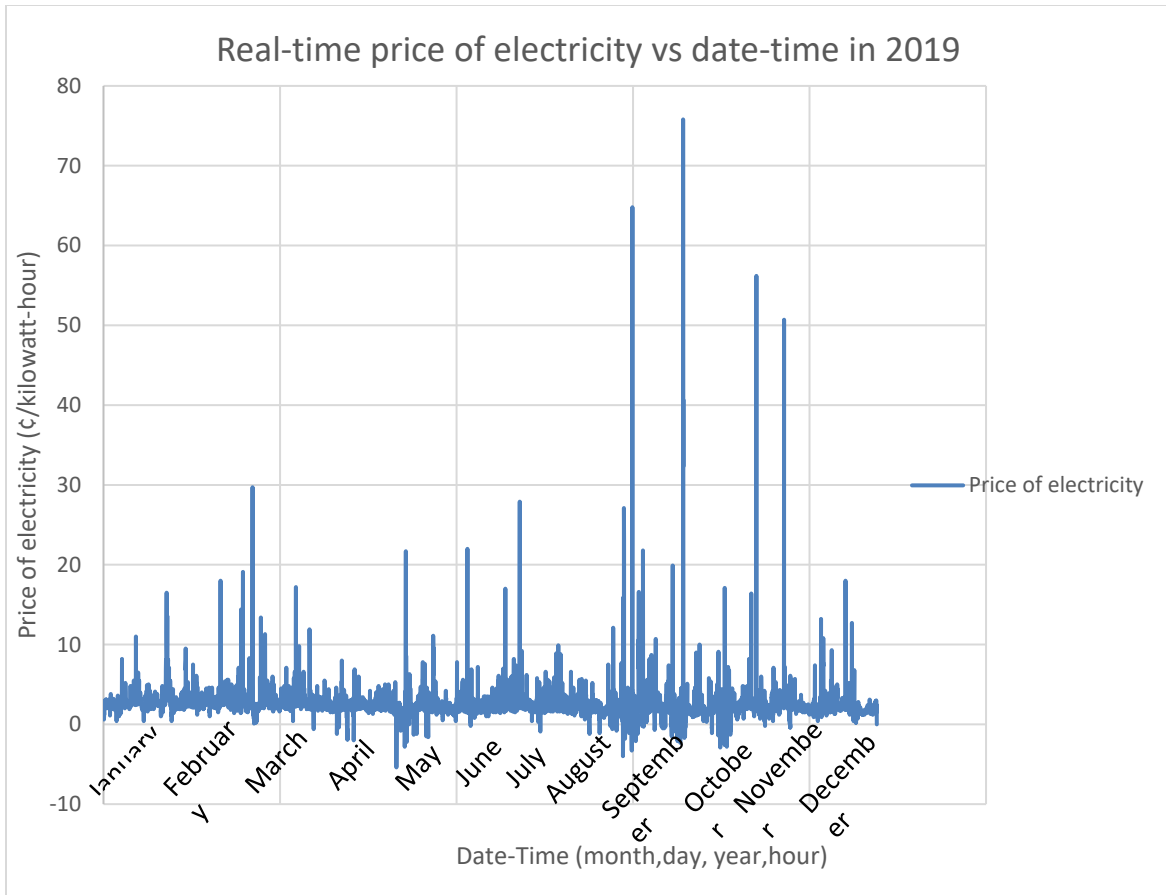


Figure 4-26: The real-time rates given by Comed during 2019 [25].

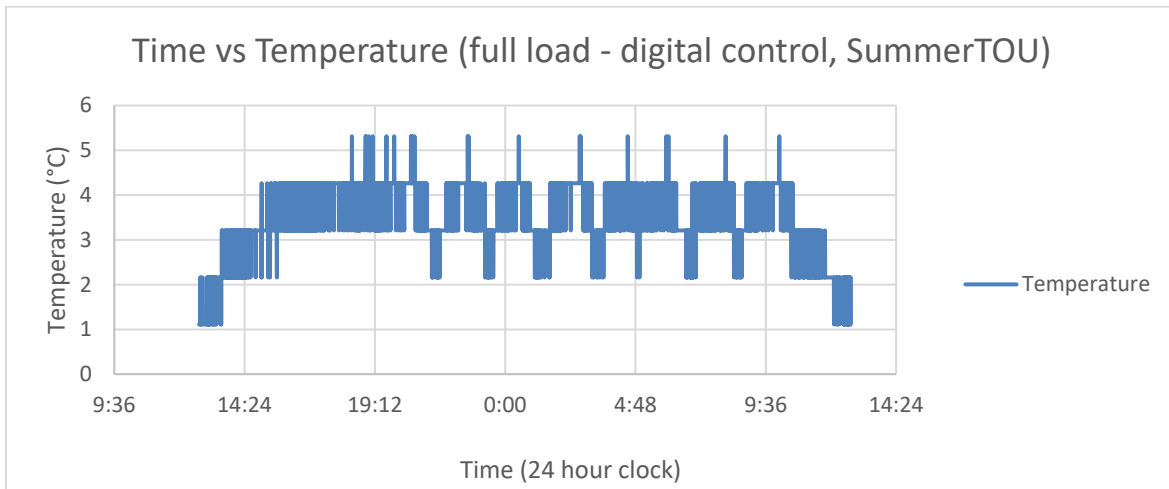


Figure 4-27: This graph shows the temperature range of the refrigerator during the STOU test.

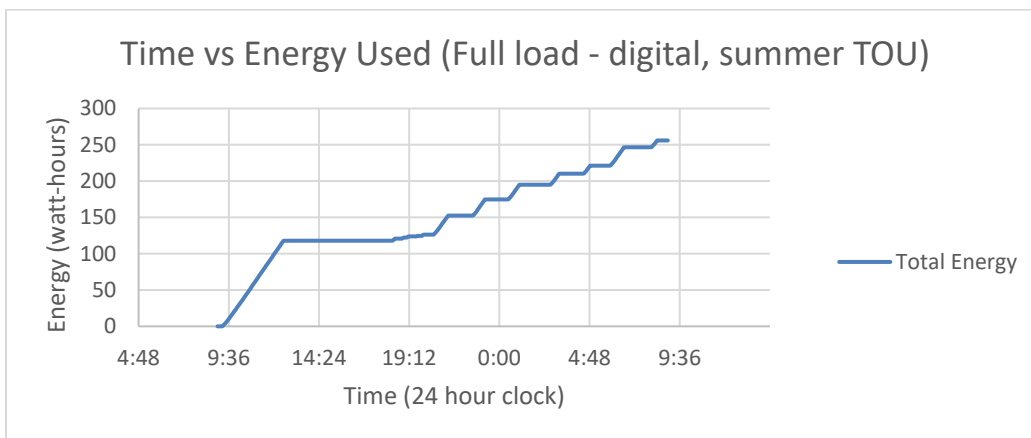


Figure 4-28: This graph shows energy consumption of approximately 256 watt-hours while showing the ride through of the STOU event.

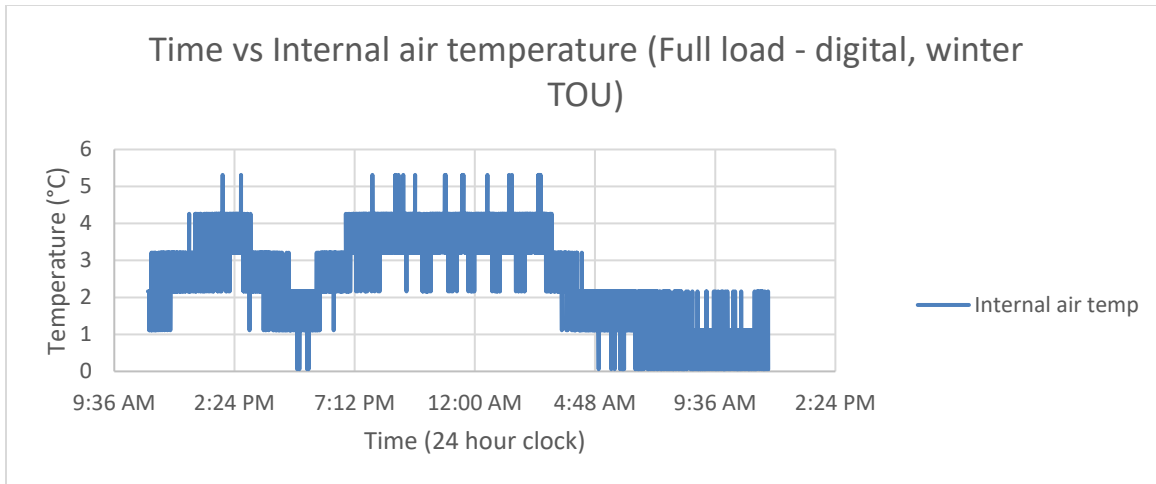


Figure 4-29: This graph shows the temperature range for the winter static TOU event.

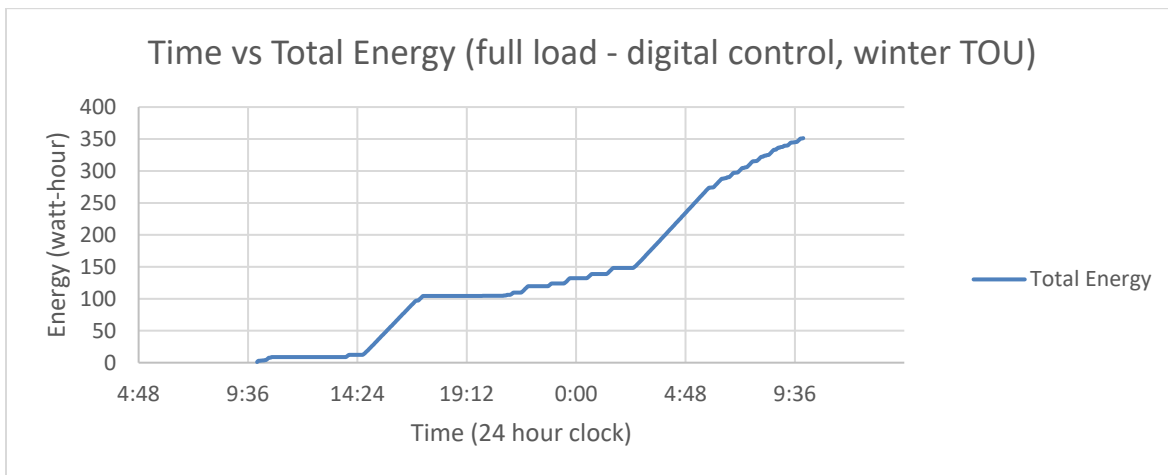


Figure 4-30: This graph shows the energy consumption of STOU event, approximately 350 watt-hours).

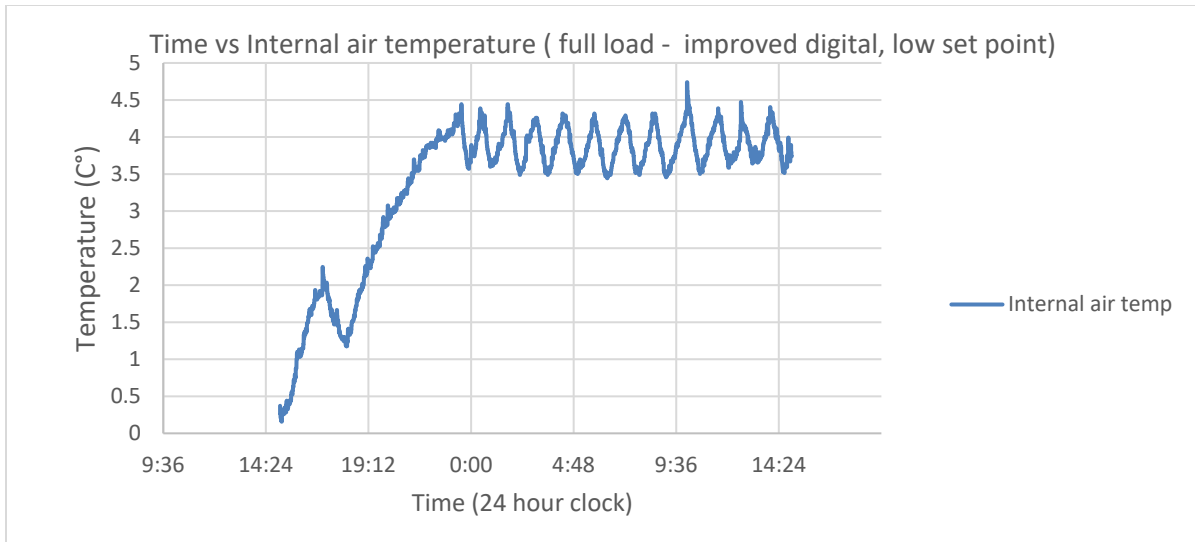


Figure 4-31: This graph shows the temperature performance of the improved digital system which is significantly more precise and has tighter control over the temperature range.

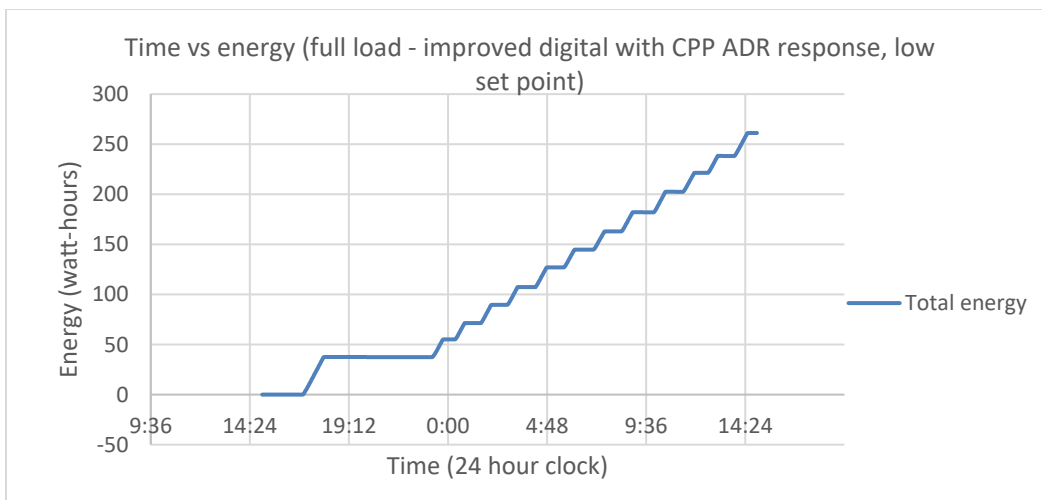


Figure 4-32: This graph shows the energy consumption of the improved digital code. Consuming approximately 260 watt-hours.

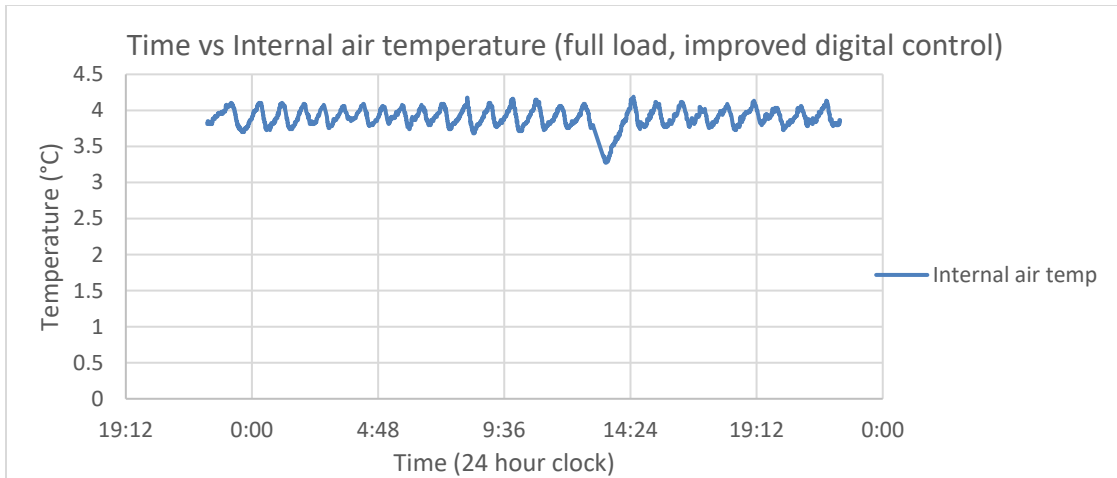


Figure 4-33: This graph shows the improved digital system temperature range including the WI-FI failsafe activation and subsequent system reset.

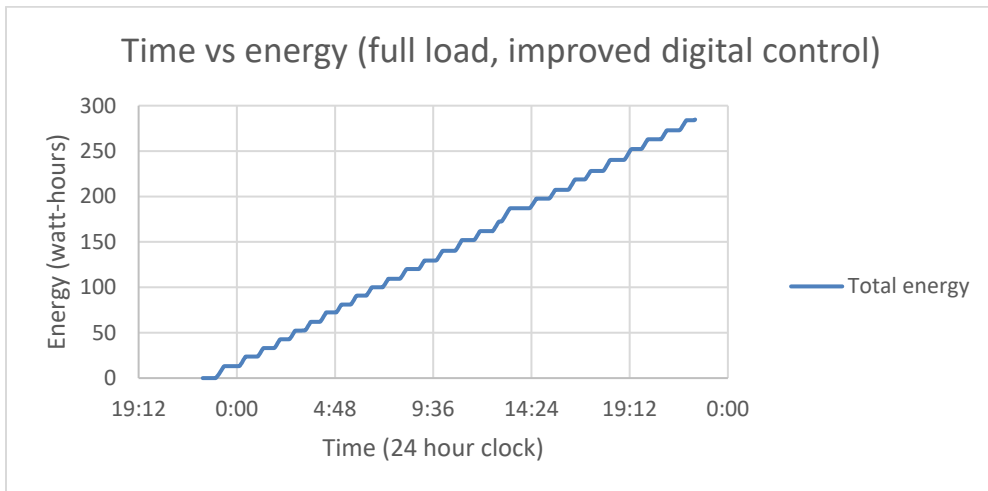


Figure 4-34: This graph shows the improved digital system which consumed 284 watt-hours.

CHAPTER FIVE

CONCLUSION AND FUTURE APPLICATION

5.1 Conclusion

Results explained for energy consumption and monetary analysis: baseline analog and digital conversion

Energy consumption for refrigerators has two primary phases: initial cooling from an unconfigured state to a steady state, and steady state operation. The initial results from the cooling efficiency tests indicated that the analog refrigerator had two main issues when initially cooling: (1) the number of cooling cycles to reach steady state, and (2) the time it took to cool to steady state. In comparison the digital system being used had substantially fewer cooling cycles and cooled more quickly, see Figures 4-1 through 4-12. This led to the hypothesis that the energy consumption was due to the amount of stop/starting the refrigerator had to undergo to reach steady state. This stop/starting then led to increased time for cooling, which in turn meant the compressor was on for a longer amount of time. Therefore, it was erroneously concluded that with more precise temperature control and longer, deeper cooling cycles the stop/starting could be cut out, thus, saving energy.

Then by incorporating load shifting via the ARD events it was found that theoretically if the law of conservation of energy was observed that monetary savings could be had. Therefore, three programs were developed: predicted electricity rates, real-time electricity rates, and predicted electricity rates with real-time CPP responses.

The predicted average cost of electricity from 2019 was calculated by averaging the rates for every hour in the year, this came out to be 2.46 ¢/kW-hr. For this program two conditions were created. One in which no load shifting was done, which then meant the 2.46 ¢/kW-hr would be observed and the other in which load shifting was done. The load shifting has two limits that would dictate the response. The first limit was a 3.8 ¢/kW-hr minimum to trigger a response. This selected as this was the threshold for which events could realistically be had that would not disrupt normal operation to a large degree. Going down to 3.7 ¢/kW-hr meant that the 2nd condition would be regularly violated. That being the 3-hour limit to the length of the event, this was selected as it was the safest length that the lab results showed. With these two conditions in mind, the entire year was analyzed, and the conditions were implemented. From that analysis it was found that there were 145 days that could have load shifting events, ranging from 1-3 hours, with 485 hours throughout the year. This led to a drop in price from 2.46 ¢/kW-hr to 2.21 ¢/kW-hr, saving 10.05% cost throughout the year.

The real-time pricing program worked similarly to predicted cost program. It was determined through analyzing the pricing data that the average cost was 2.45 ¢/kW-hr. From here two conditions were again developed. One in which no load shifting was done so the real-time price would be the same and the other in which load shifting was done. For the load shifting a limit of 3.5 ¢/kW-hr was imposed with the same 3-hour time limit for an event. This meant that there were 735 hours that were responded to, with 236 unique days with events. The price dropped from 2.45 ¢/kW-hr to 1.98 ¢/kW-hr, a savings of 19.16%.

The last program that was developed was a more comprehensive price analysis for load shifting using TOU and CPP. Finding potential TOU events was done utilizing the predicted data by sorting for a pricing threshold at or above 4.0 ¢/kW-hr and then for 3-hour time blocks that broke this threshold the most often throughout the year. From here these TOU events were imposed on the real-time pricing data. Coupled with CPP responses, if large jumps in price would trigger an alert, and mindful of the 3-hour event time limit the price dropped from an average of 2.45 ¢/kW-hr to 2.1 ¢/kW-hr a 14.15% in savings.

Therefore, it was concluded that with the above calculations and the assumption that the refrigerator could keep the same or better energy performance a significant monetary benefit could be achieved.

However, as seen in Figures 4-13 through 4-24, at steady state the stop/starting has the opposite effect for energy consumption. During the initial phase energy consumption was much greater, but during steady state operation the stop/starting was more efficient. This result was likely because the analog cooling system was keeping a tighter control of the temperature operating range. Therefore, the compressor would be on for a much shorter amount of time. While the startup cost is very high for machines, it is offset by a very short period of draw, whereas the deeper cooling cycles of the digital refrigerator would draw consistent power for longer periods. This ultimately meant that the initial hypothesis was incorrect and would need to be reworked.

Initial cooling is normally a very short period compared to steady state operation, as the initial cooling phase only occurs in the opening stage of the refrigerator's use.

Therefore, it was imperative that the digital system performs better or at least as well as the analog system. Given a clear understanding of the initial error, a new hypothesis was formed: More precise temperature control with shorter cooling cycles would result in better performance.

This was achieved by making a few minor changes to the control code and one to equipment. The control code was changed such that the threshold for triggering a cooling cycle was smaller and the dead band after the cooling cycle was shorter, which, ultimately lead to significantly more stop/starting operations in the device. More precise temperature control was achieved by changing the thermocouples to those with a slightly higher resolution, going from a tenth of a degree to a hundredth of a degree. While this change did not affect the architecture of the system it did give more consistent temperature results that had fewer errors.

What this ultimately showed is that the analog refrigerator is simply insufficient for refrigeration as the temperature range was too varied and violated the FDA 4.4 °C too often. On a commercial scale this is unacceptable and would lead to loss of products. While the analog thermostat does have higher energy efficiency it trades its temperature control. This means that while the design platform is less energy efficiency it is a better refrigerator. Furthermore, the design platform methods and ADR algorithms still hold true and would lead to percentage gains in monetary benefits.

Final thoughts: sources of error and implications for the future of automated demand response

There were several sources of error found in the research conducted. The first and most concerning was the operation of the analog refrigerator as it was indicating better energy performance over that of the digital system. However, this was likely due to the inaccuracies in temperature sensing. By having a “good enough” approach in a relatively small refrigerator the temperature range was extremely varied. While not good enough for commercial refrigeration as the 4.4 °C barrier was routinely broken the refrigerator is more than sufficient for keeping water bottles and longer lasting foods safe. This meant that the equipment used for experimentation was insufficient for proper demonstration but was sufficient in showing the design platform and theoretical concepts.

Another source of error was the sample size coupled with human error. As always human error can result in varied data but paired with longer experimentation times it is likely that some mistakes were made. While much of the damage was likely mitigated through rigorous testing, implementing 24-hour tests over data that spans a year means that errors will compound.

Lastly the implications for demand response are profound. While the energy data gathered in this thesis was interesting, but not academically relevant, the concepts of demand response and the algorithms explored are very promising. Monetary savings around 10-20% for end-users are exceptional. Then implemented on a commercial scale could potentially save even more as the ADR on that scale could free up tens of

thousands of kilowatt hours during times of increase demand, lessening the strain on the electrical grid.

5.2 Future Application

Improved system: thermodynamic model, machine learning, and website controls

In the future, the system described by this thesis has room for expanded features and improvements that will make for better performance and/or varied application.

Starting with a proper thermodynamic model the accuracy of the pre-cooling start time and duration could be improved. This was not done due to software limitations and, eventually, time constraints. However, with additional time, a thermodynamic model could be built which would more accurately model the system. This may lead to an improved dead band which when coupled with energy consumption data, could give better steady state performance, drawing a correlation between the amount and depth of cooling cycles.

Furthermore, by adding machine learning it is possible to allow for certain variability in the refrigeration system to be improved. This could be anything from a balance between pre-cooling time and cost off-set to varying the width of the dead band to improve steady state efficiency. Allowing for a machine to access features of operation may lead to increased system performance that otherwise would take hours of observation and experimentation by humans.

Data collection and pattern recognition that could be integrated into this system to enable more complex responses to ADR requests. With the increasing penetration of

smart devices, it may be logical to conclude that machine learning can be tied to these systems to enable more sophisticated applications of ADR.

With the end-user able to set customized responses or desired performance requests, using a website user-interface may further increase the ease of use of an ADR system. Future work may be directed towards integrating increased customizability to an ADR system for the purposes of studying adoption rates of ADR systems. This may either be in conjunction with or perhaps independent of, the benefits received to utilities and by the end-user via ADR.

Expanded application: residential and industrial uses

The areas benefited from HVAC-R systems are wide and highly varied. Nearly every sector has an application of these systems, from the cooling system of an industrial plant to the air conditioning of a single household, requiring future expansion of ADR work in the other sectors outside of commercial use. A particularly interesting field of study may be, as previously mentioned, smart home or smart device integration with ADR signals from utility companies. For example, if a smart device was able to both receive signals from a utility company and affect the AC of a home, the monetary savings available to the household could be increased by ADR programming. Further penetration of the smart devices sector would potentially allow an entire neighborhood or section of a city to respond to utility requests, which may change how we think about DR with automation at the forefront of development.

LIST OF REFERENCES

- [1] U.S. Energy Information Administration - EIA - independent statistics and analysis. Delivery to consumers - U.S. Energy Information Administration (EIA). (n.d.). Retrieved April 7, 2022, from <https://www.eia.gov/energyexplained/electricity/delivery-to-consumers.php>
- [2] Computer and technology use in education buildings continues to increase. Homepage - U.S. Energy Information Administration (EIA). (n.d.). Retrieved April 7, 2022, from <https://www.eia.gov/todayinenergy/detail.php?id=24812>
- [3] Environmental Protection Agency. (n.d.). *Global Greenhouse Gas Emissions Data*. EPA. Retrieved April 7, 2022, from <https://www.epa.gov/ghgemissions/global-greenhouse-gas-emissions-data>
- [4] *U.S. Energy Information Administration - EIA - independent statistics and analysis*. U.S. energy facts explained - consumption and production - U.S. Energy Information Administration (EIA). (n.d.). Retrieved April 7, 2022, from <https://www.eia.gov/energyexplained/us-energy-facts/>
- [5] Shi, Q., Li, F., Hu, Q., & Wang, Z. (2017, August 24). *Dynamic Demand Control for system frequency regulation: Concept Review, algorithm comparison, and future vision*. Electric Power Systems Research. Retrieved April 14, 2022, from <https://www.sciencedirect.com/science/article/pii/S0378779617303061>

- [6] *Comprehensive review of the recent advances in industrial and commercial dr.* IEEE Xplore. (n.d.). Retrieved April 14, 2022, from <https://ieeexplore.ieee.org/document/8681637>

- [7] Papadaskalopoulos, D., Strbac, G., Mancarella, P., Aunedi, M., & Stanojevic, V. (2013). Decentralized participation of flexible demand in electricity markets—part II: Application with electric vehicles and heat pump systems. *IEEE Transactions on Power Systems*, 28(4), 3667–3674. <https://doi.org/10.1109/tpwrs.2013.2245687>

- [8] Crabb, J. A., Murdoch, N., & Penman, J. M. (1987). A simplified thermal response model. *Building Services Engineering Research and Technology*, 8(1), 13–19. <https://doi.org/10.1177/014362448700800104>

- [9] Yu, R., Zhong, W., Xie, S., Yuen, C., Gjessing, S., & Zhang, Y. (2016). Balancing power demand through EV mobility in vehicle-to-grid mobile energy networks. *IEEE Transactions on Industrial Informatics*, 12(1), 79–90. <https://doi.org/10.1109/tii.2015.2494884>

- [10] Liu, Y., Yuen, C., Huang, S., Ul Hassan, N., Wang, X., & Xie, S. (2014). Peak-to-average ratio constrained demand-side management with consumer's preference in residential smart grid. *IEEE Journal of Selected Topics in Signal Processing*, 8(6), 1084–1097. <https://doi.org/10.1109/jstsp.2014.2332301>

- [11] Brooks, A., Lu, E., Reicher, D., Spirakis, C., & Weihl, B. (2010). Demand dispatch. *IEEE Power and Energy Magazine*, 8(3), 20–29. <https://doi.org/10.1109/mpe.2010.936349>
- [12] Vardakas, J. S., Zorba, N., & Verikoukis, C. V. (2015). A survey on demand response programs in smart grids: Pricing methods and optimization algorithms. *IEEE Communications Surveys & Tutorials*, 17(1), 152–178. <https://doi.org/10.1109/comst.2014.2341586>
- [13] Mohagheghi, S., & Raji, N. (2012). Intelligent demand response scheme for energy management of Industrial Systems. *2012 IEEE Industry Applications Society Annual Meeting*. <https://doi.org/10.1109/ias.2012.6374087>
- [14] Yan, Y., Qian, Y., Sharif, H., & Tipper, D. (2012). A survey on cyber security for Smart Grid Communications. *IEEE Communications Surveys & Tutorials*, 14(4), 998–1010. <https://doi.org/10.1109/surv.2012.010912.00035>
- [15] Zhang, X., & Hug, G. (2015). Bidding strategy in Energy and spinning reserve markets for aluminum smelters' demand response. *2015 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*. <https://doi.org/10.1109/isgt.2015.7131854>
- [16] Zhang, X., & Hug, G. (2014). Optimal regulation provision by aluminum smelters. *2014 IEEE PES General Meeting | Conference & Exposition*. <https://doi.org/10.1109/pesgm.2014.6939343>

- [17] Wissner, M. (2011). The smart grid – A saucerful of secrets? *Applied Energy*, 88(7), 2509–2518. <https://doi.org/10.1016/j.apenergy.2011.01.042>
- [18] Zhang, X., Hug, G., Kolter, Z., & Harjunoski, I. (2015). Industrial demand response by steel plants with spinning reserve provision. *2015 North American Power Symposium (NAPS)*. <https://doi.org/10.1109/naps.2015.7335115>
- [19] He, B., Xu, F. Y., Lai, L. L., Yang, C., Lu, S., & Li, D. (2014). A model of demand response scheduling for Cement Plant. *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. <https://doi.org/10.1109/smc.2014.6974393>
- [20] Merkert, L., Harjunoski, I., Isaksson, A., Säynevirta, S., Saarela, A., & Sand, G. (2015). Scheduling and energy – industrial challenges and opportunities. *Computers & Chemical Engineering*, 72, 183–198. <https://doi.org/10.1016/j.compchemeng.2014.05.024>
- [21] Albadi, M. H., & El-Saadany, E. F. (2008, May 21). *A summary of demand response in electricity markets*. Electric Power Systems Research. Retrieved April 17, 2022, from <https://www.sciencedirect.com/science/article/pii/S0378779608001272>
- [22] Thornburg, J. (2020, August). Grid_Fruit_1pg_exec_summary_Aug2020. GridFruit.
- [23] ComplianceMate™ (2020, September 1). *5 ways to Keep Your Restaurant Refrigeration System Cool this summer*. Restaurant Business. Retrieved July 7, 2022, from <https://www.restaurantbusinessonline.com/operations/5-ways-keep-your->

summer#:~:text=While%20the%20FDA%20requires%20restaurant,set%20at%203
8%C2%B0F.

- 80

VITA

Justin Martinez was born in Redding California. They moved to Nashville where they were raised and educated until college. He graduated from the University of Tennessee Knoxville in the Spring of 2020, obtaining a bachelor's degree in Electrical engineering. Justin continued his education under the guidance of Dr. Fran Li, researching power system application in the smart home technology lab. Justin was attached to the Smart Home Energy Management System (SHEMS) where he began his work with demand response. He plans to move to Nashville upon the completion of his degree at UT