

INTERSTAATLICHE HOCHSCHULE FÜR TECHNIK BUCHS
AUTOMATING DATA COLLECTION FROM HEAT PUMP MONITORING SYSTEMS

A Major Qualifying Project Report

Submitted October 31th, 2018

to the faculty of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Bachelor of Science

AUTHORS

Evan Duffy

Milap Patel

FACULTY ADVISORS

Michael J. Ciaraldi

Professor, Department of Computer Science

Worcester Polytechnic Institute

And

John M. Sullivan

Professor, Department of Mechanical Engineering

Worcester Polytechnic Institute

SPONSOR

Stefan Bertsch, PhD

Head of the Institute for Energy Systems

Interstaatliche Hochschule für Technik Buchs

Abstract

The goal of this Major Qualifying Project was to create an automated solution to download, store, and back up data coming from PLCs installed in homes across Switzerland. PLCs at these locations monitor sensors attached to heat pumps within the homes of individual volunteers. Previously, NTB engineers manually downloaded data once a week because the process was too tedious and time consuming. To aid NTB engineers the team automated the download process and created an efficiency analysis program.

Acknowledgements

The team would like to acknowledge several individuals for the help they provided to make this project possible. Firstly our on-site advisor: Stefan Bertsch, as well as our WPI advisors: Michael Ciaraldi and John Sullivan. The team would also like to thank some of the engineers for helping us extensively with the technical aspects of our project: Matthias Barthold, and Rupert Fritz.

In addition, the team would also like to thank some people who made this project possible through organization between NTB, WPI and the Switzerland Project Center: Emine Cagin, and Nancy Burnham.

Executive Summary

A branch of The Interstate University of Applied Sciences Buchs (NTB), The Institute of Energy Systems (WPZ), uses Programmable Logic Controller (PLC) modules to monitor the efficiency of heat pump systems in real world environments to compare against the efficiency of the same heat pumps under test environments. There is currently an entire ecosystem of software and hardware being used to collect, analyze, maintain, and display the data on all the heat pumps.

The current system requires an engineer to take time to download the data from each of the PLC modules on a weekly basis. These files are uploaded to a database for later processing. This project seeks to automate the process of downloading files from PLCs and then uploading them to the database.

The NTB engineer currently uses the software suite that ships with the PLC modules, which is made by B&R Automation, to download the files from the PLCs. The design created first was to automate the process that was already used by a human, allowing it to be completed automatically and quickly. This would also enable it to run nightly rather than weekly.

The first design was to make a program that utilized FTP to download the files from the PLC and then upload them to the database website, using a local machine as an intermediate, backup location. This program would run every night, allowing data collection to be much more frequent than before. If a PLC were unavailable or if it were running out of space, an email would be sent to the engineers at NTB telling them about the problem. It was known that the PLC had an FTP server available but it was also discovered that the hosting service used for the database website, HostPoint, had an FTP service available as well.

In addition to the program that would run nightly, it was decided that a graphical front-end should be provided to the user as well. Using PyQt5, a helpful Python user interface library,

a graphical front-end was created. This allows the user to choose any day, month, or year to download data from any one or more PLCs.

Furthermore, a heat pump analysis program was also made to aid NTB engineers with analyzing the field unit heat pump systems, heat pumps operating in the real world, for their efficiency. Currently, NTB engineers consider the vapor compression cycle as a black box. As a result, they only analyze the efficiency of the external systems without considering the compressor efficiency itself. This program will allow them to gain the knowledge of the compressor efficiency which may provide them with a more in depth analysis of the system efficiency as a whole.

While implementing these programs the team ran into several roadblocks. One such roadblock involved getting the program to iterate over the PLCs. At this point it was discovered to be impossible connecting to more than one PLC at a time or programmatically choose a PLC using the manufacturer supplied software. This led to researching ways to automate LinkManager, but it was found that there were no feasible ways to do so.

Midway through the project the team spoke with some coworkers and decided that a different approach was needed to make the program succeed in the time span of our project. The team decided to change the design to have the PLCs send emails nightly and have another program download the emails with the data files attached. It would then organize and upload them to the database hosting service, HostPoint.

The team also created a heat pump analysis program that would take in the data files gathered from the PLCs and analyze them for their efficiency. This provides the engineers with information on how the heat pump is actually operating, which at the time of this project was not considered by the engineers at NTB.

Throughout the project, the team made sure to create modular software for all the programs so that anyone can extend or use our functionality in the future. This modularity also allows for the software to be changed to support a change in NTB's infrastructure.

If NTB later decides to purchase a full enterprise license of B&R software that would enable the use of LogTunnel or a similar service, then the original program will become much more useful, since it would allow for the user to be able to download any day's data from any number of PLCs all at once. For the time being it can be slightly useful as it at least eliminates the use of FileZilla, but still requires the use of the manufacturer's software which would require switching between PLCs manually.

By the end of the project, the team had created two programs; one for downloading data and one for data analysis. It was decided that our mail-downloader program would be the main program that would be run nightly by cron on HostPoint to gather files from the PLCs. The initial program would be used as an "in case of emergency" program where the user can manually download data if there is a failure with the main program. Furthermore, the team also created heat pump analysis program that would aid NTB engineers in analyzing the heat pumps themselves.

Table Of Contents

Abstract.....	i
Acknowledgements.....	ii
Executive Summary.....	iii
Table Of Contents.....	vi
Table Of Figures.....	viii
Chapter 1: Introduction.....	1
Chapter 2: Background.....	3
2.1 Previous MQP Work.....	3
2.2 Current System For Collecting Data.....	4
2.3 B&R Automation.....	4
2.3.1 B&R Hardware.....	5
2.3.2 B&R Software.....	5
2.4 Network Diagram For Current Data Transfer.....	7
2.5 Thermodynamics Background.....	8
2.5.1 Vapor Compression Cycle.....	10
2.5.2 Finding Vapor Compression Parameters.....	13
2.5.3 Calculating Thermal Energy Exchanged At Condenser.....	16
2.5.4 Calculating Work Energy Of Compressor.....	17
2.5.5 Calculating Actual Compressor Exit Energy.....	17
2.5.6 Calculating COP Of The Vapor Compression System.....	18
2.5.7 Calculating Efficiency Of The Compressor.....	19
2.6 Field Units.....	20
2.6.1 What Is Measured At The Field Units.....	21
2.7 Test Bench Heat Pump System.....	22
2.8 Background Used In Implementation.....	24
Chapter 3: Methodology.....	25
3.1 First Design For File Downloader Program.....	25
3.1.1 User Interface.....	27
3.1.2 Major Roadblock.....	28
3.2 Mid-Term Requirement Changes.....	29
3.2.1 Second Design For File Downloader Program.....	30
3.2.2 Re-Enabling Emails From PLCs.....	31
3.3 Documentation.....	31

3.4 Heat Pump Analysis Program.....	32
3.5 How The Field Heat Pump Units Will Be Analyzed By The Program	33
3.5.1 Compensate For Missing Measurements	34
3.6 Testing Assumptions On The Test Bench Heat Pump.....	35
3.6.1 Collecting Test Bench Heat Pump Data	35
3.6.2 Testing Different Assumptions Of Delta T's.....	39
Chapter 4: Results	41
4.1 Verifying The Program Results For Test Bench Data Analysis	41
4.2 Results Of Testing Different Delta T's On Test Bench	42
Chapter 5: Conclusion and Future Work	45
5.1 Future Work: Expandability	45
References.....	47
Appendices.....	48
Appendix A: Charts and Tables	48
A.1 HostPoint Resource Usage By Approach	48
Appendix B: Screenshots Of The Programs	49
Appendix C: Test Bench Instance One Analysis And Results	50
C.1 Manual Calculations And Interpolations:	51
C.2 Heat Pump Analysis Program Result:.....	53
Appendix D: Test Bench Instance Two Analysis And Results	54
D.1 Manual Calculations:	54
D.2 Heat Pump Analysis Program Result:.....	55
Appendix E: Test Bench Instance Three Analysis And Results.....	56
E.1 Manual Calculations:.....	56
E.2 Heat Pump Analysis Program Result:	57
Appendix F: GitHub Repository Links.....	58
F.1 Solution 1:.....	58
F.2 Solution 2:.....	58
F.3 Analysis Program:	58

Table Of Figures

Figure 1: Network Diagram For Manual Method Of Data Transfer	8
Figure 2: Concept Of A Refrigeration Cycle	9
Figure 3: Concept Of A Heat Pump Cycle	10
Figure 4: Demonstration Of How Water Heater Works And How Components Are Used	11
Figure 5: Demonstration Of How Water Heater Works And How Components Are Used	13
Figure 6: Vapor Compression Cycle On A p-h Diagram With The Four States	14
Figure 7: Vapor Compression Cycle On A t-s Diagram With The Four States	14
Figure 8: Field Unit Heat Pump Setup	21
Figure 9: Test Bench Heat Pump And Associated Sensors Setup	23
Figure 10: Screenshot Of File Downloader Program	28
Figure 11: Chart Showing The Flow Rate Of Water Throughout The Testing Process	36
Figure 12: Chart Showing The Temperatures Of Air Throughout The Testing Process	37
Figure 13: Chart Showing The Temperatures Of Water Throughout The Testing Process	38
Figure 14: Chart Showing The System Pressures Throughout The Testing Process	39
Figure 15: Chart Showing The Temperatures Of Refrigerant Throughout The Testing Process	39
Figure 16: Chart Showing The Percent Errors In Calculating The Ideal COP With Assumptions	43
Figure 17: Chart Showing The Percent Errors In Calculating The Efficiency With Assumptions	44

Chapter 1: Introduction

A branch of The Interstate University of Applied Sciences Buchs (NTB), The Institute of Energy Systems (WPZ), uses Programmable Logic Controller (PLC) modules to monitor the efficiency of heat pump systems in real world environments to compare against the efficiency of the same heat pumps under test environments. There is currently an entire ecosystem of software and hardware being used to collect, analyze, maintain, and display the data on all the heat pumps.

The current system requires an engineer to take time to download the data from each of the PLC modules on a weekly basis. These files are then uploaded to a database for later processing. This project seeks to automate the process of downloading files from PLCs and then uploading them to the database.

The NTB engineer currently uses the software suite that ships with the PLC modules, which is made by B&R Automation, to download the files from the PLCs. The design created first was to automate the process that was already used by a human. Doing so would allow it to be completed automatically and quickly, thus enabling it to be run nightly rather than weekly.

Unfortunately the team was unable to secure access to a full license to this software due to the restricted time frame of this project, so using this software suite as originally intended was not possible. A change to our approach was required halfway through the project.

The second design chosen was an approach where files would be emailed, downloaded and organized by a program running on NTB's hosting service, HostPoint. By automating the data download process, the team allowed the NTB engineers to spend their time more effectively elsewhere, as well as lowered data latency and increased throughput.

Furthermore, a heat pump analysis program was also made to aid NTB engineers with analyzing the field unit heat pump systems, heat pumps operating in the real world, for their efficiency. Currently, NTB engineers consider the vapor compression cycle as a black box. As a result, they only analyze the efficiency of the external systems without considering the compressor efficiency itself. This program will allow them to gain the knowledge of the compressor efficiency which may provide them with a more in depth analysis of the system efficiency as a whole.

Chapter 2: Background

The Institute of Energy Systems, a branch of NTB, regularly tests heat pumps in their labs and in the field. They use test data gathered in both situations to analyze the heat pumps for their efficiency. In a field test, sensors and a PLC (Programmable Logic Controller), which collect data, are installed on a heat pump that is actively running in the home of a volunteer participating in the program.

The reason that heat pumps of volunteers are being tested is because NTB needs to make sure that these heat pump systems operate as well in the field as they do in the lab. This allows NTB to make a more accurate assertion about the efficiency of the model of heat pump. In the end, the program allows consumers to make better-informed decisions when buying a heat pump.

2.1 Previous MQP Work

In 2017, there was an MQP working with the Institute of Energy Systems at NTB to create a web application to coalesce and display the data collected from the PLCs. Since then a system had been implemented allowing data to be uploaded to a File Transfer Protocol (FTP) server where the data is parsed, analyzed, and uploaded into an SQL database where it can be accessed by the MQP students' website [2].

The previous MQP greatly increased the readability and organization of the data collected from the PLC modules, however it left a gap in the data-flow between the PLCs and the FTP server. The PLCs do not have any way to create outgoing FTP client connections, so they are unable to upload the files on its own. A human needs to somehow get the data from the PLC to the FTP server. This project aimed to automate this gap in a reliable and straightforward way.

2.2 Current System For Collecting Data

Every Monday, an engineer at NTB logs into B&R Automation's LinkManager, which is software for connecting remotely to PLCs. This allows him to, one at a time, connect to a PLC via a proprietary tunneling system, login with FileZilla (a file transfer application) and download the files. This process alone takes significant time. The engineer needs to repeat the same process for all sixteen of the PLCs currently installed in the field, which takes long enough so that they only have time to collect data once a week.

The most significant bottleneck of this process is the use of LinkManager. LinkManager is designed to bridge the local network to a "SiteManager," which is an on-site router, created by B&R Automation, that allows the proprietary tunneling system to expose the PLC on the local network of the computer running LinkManager. Due to limitations with LinkManager, there is no way to connect to more than one SiteManager at a time, at least out of the box.

Pros to this system include the fact that there is a consistent schedule for data collection and if there is a problem, it is noticed by an engineer as soon as they log in to retrieve the data. However, the pros end there. The list of cons for this system is much longer: Someone takes half an hour or more out of their day to do the data collection, if someone isn't available then the data collection doesn't happen, if a PLC is offline no one will notice until Monday when they check on them, and the list goes on. This process is very low-tech for a fairly high-tech system, and needed to be updated to fit in with the rest of the WPZ ecosystem.

2.3 B&R Automation

B&R Automation is the company that manufactures the PLC modules, and all accompanying software and hardware, used by the Institute of Energy Systems at NTB.

2.3.1 B&R Hardware

B&R Automation have a massive ecosystem of hardware, of which NTB uses PLC modules, and SiteManagers.

2.3.1.1 PLC Module

PLC modules are modular computers that are equipped with sensors and a real time clock. The on board computer has an operating system that allows programs to be written in traditional Structured Text. The OS also allows the software to read and write to files and use the network for various functions.

2.3.1.2 SiteManager

A SiteManager is a router that allows the network to be bridged by a client using LinkManager. The advantage to using the SiteManager is that it does not rely on a port being open in the household of the volunteer, and also isolates the PLC from the home network of the volunteer.

2.3.2 B&R Software

B&R Automation has a large ecosystem of software for various functionality and features pertaining to their hardware ecosystem. Some of this software, however, is not available under certain licenses, such as the research license held by NTB.

2.3.2.1 LinkManager

LinkManager is a program that runs on a Windows machine that will create a tunnel between the local network of the Windows machine and the SiteManager. LinkManager interacts

with the user via a local web page that uses JavaScript to control a daemon-like process that runs in the background. The LinkManager daemon can only be talked to via the console web page, and no API exists to automate its features. To automate the features of LinkManager, the console web page would need to be reverse engineered.

2.3.2.2 GateManager

GateManager is a centralized server that stores information about each site manager and gives the LinkManager a 'Gateway' to access the SiteManager.

2.3.2.3 LogTunnel

LogTunnel is an addon feature of LinkManager that allows many SiteManagers to be grouped together in a way that would allow all the PLCs to be exposed on the local network at once. This is useful because it allows a way to programmatically access all the PLCs, rather than requiring human intervention to switch between each PLC in LinkManager.

Unfortunately, LogTunnel is a feature that requires a full enterprise license of B&R software. At the time of the project, NTB had a research license, so using LogTunnel was found to be impossible for this project, as it would take longer than the remaining time in the project to upgrade the license.

2.3.2.4 PLC Structured Text Language

The PLCs are programmed in a traditional Structured Text language that is historically associated with the programming of PLCs. The B&R version of structured text has some convenient library functions, for instance those used to send emails. B&R also provides many example programs that were used to help create the software that runs on the PLCs.

The current software solution, created by an engineer at NTB, logs the sensor data once every 10 seconds to an Excel file, and every day a new Excel file is created. A feature was added during this project that has the PLC send an email every day with the full Excel file with all the data from the previous day. This feature will be discussed further later in this report [7].

2.4 Network Diagram For Current Data Transfer

The diagram below shows the network layout of the different networks in play when collecting data. The red section is the remote network in the home of a volunteer. The green section is NTB's network and the blue section is the network for HostPoint, the hosting service used by NTB. At NTB, a client initializes a connection using LinkManager. GateManager provides a tunneled connection to the SiteManager in the home of a volunteer, represented by the yellow line. The SiteManager then makes a LAN connection to the PLC (Labeled SPS in this diagram because SPS is German for PLC), which sensors are connected to on the heat pump system. The client then uses this connection to download the needed Excel files from the PLC and which are then uploaded to HostPoint over a regular WAN connection.

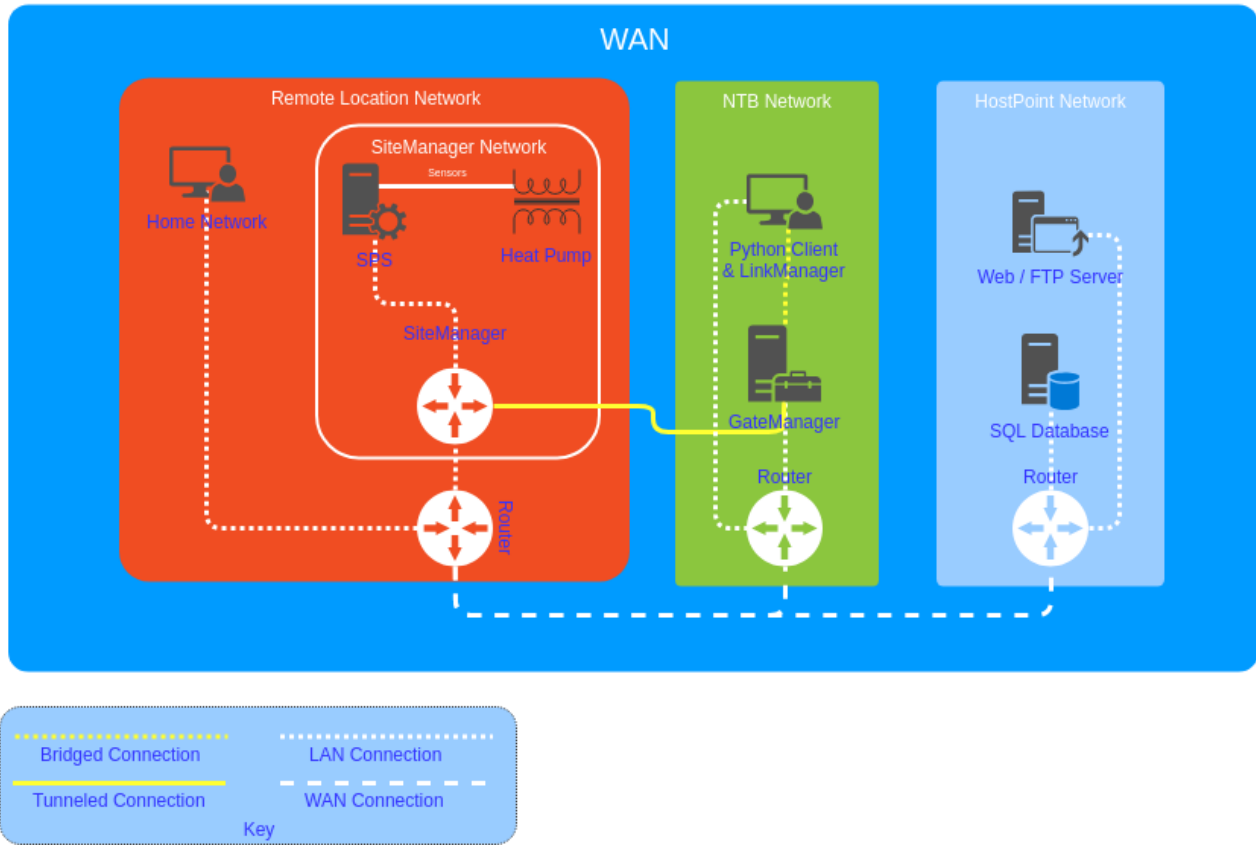


Figure 1: Network Diagram For Manual Method Of Data Transfer

2.5 Thermodynamics Background

There are two major types of vapor compression systems in use today: a refrigeration system and a heat pump system. A refrigeration system transfers heat from a low temperature reservoir to a higher temperature reservoir, essentially cooling the desired medium [6]. An example of this would be a refrigerator. The inside of the refrigerator acts as the low temperature reservoir where heat is extracted. The outside of the refrigerator acts as the higher temperature reservoir where heat is dissipated. This essentially decreases the temperature of the desired medium which is the space inside of the refrigerator as described in figure two below.

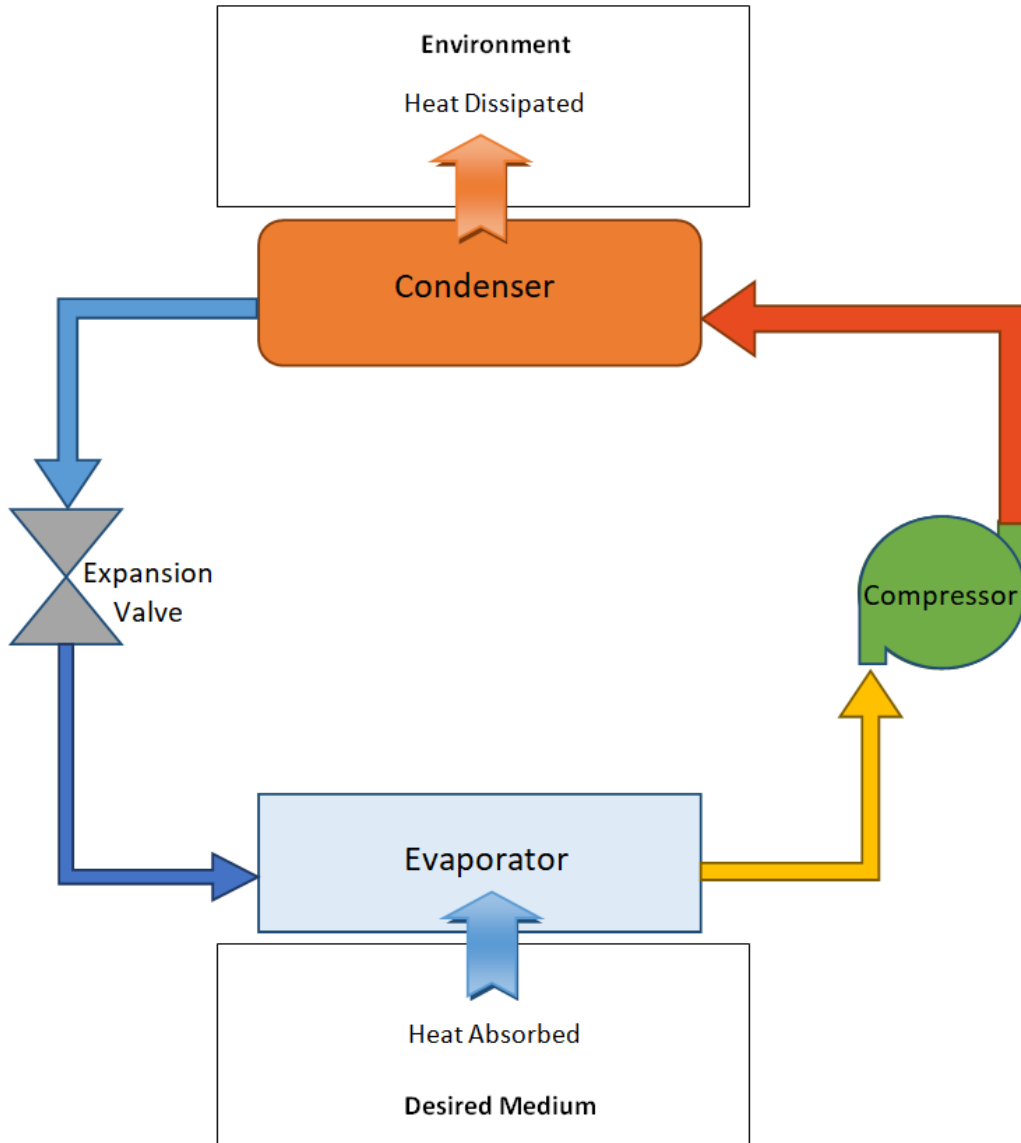


Figure 2: Concept Of A Refrigeration Cycle

A heat pump system, however, does the opposite. It transfers heat from a higher temperature reservoir to a lower temperature reservoir, essentially heating the desired medium [6]. An example of this would be a water heater where heat is added to the water from the surrounding environment, as described in figure three below. This essentially increases the temperature of the water.

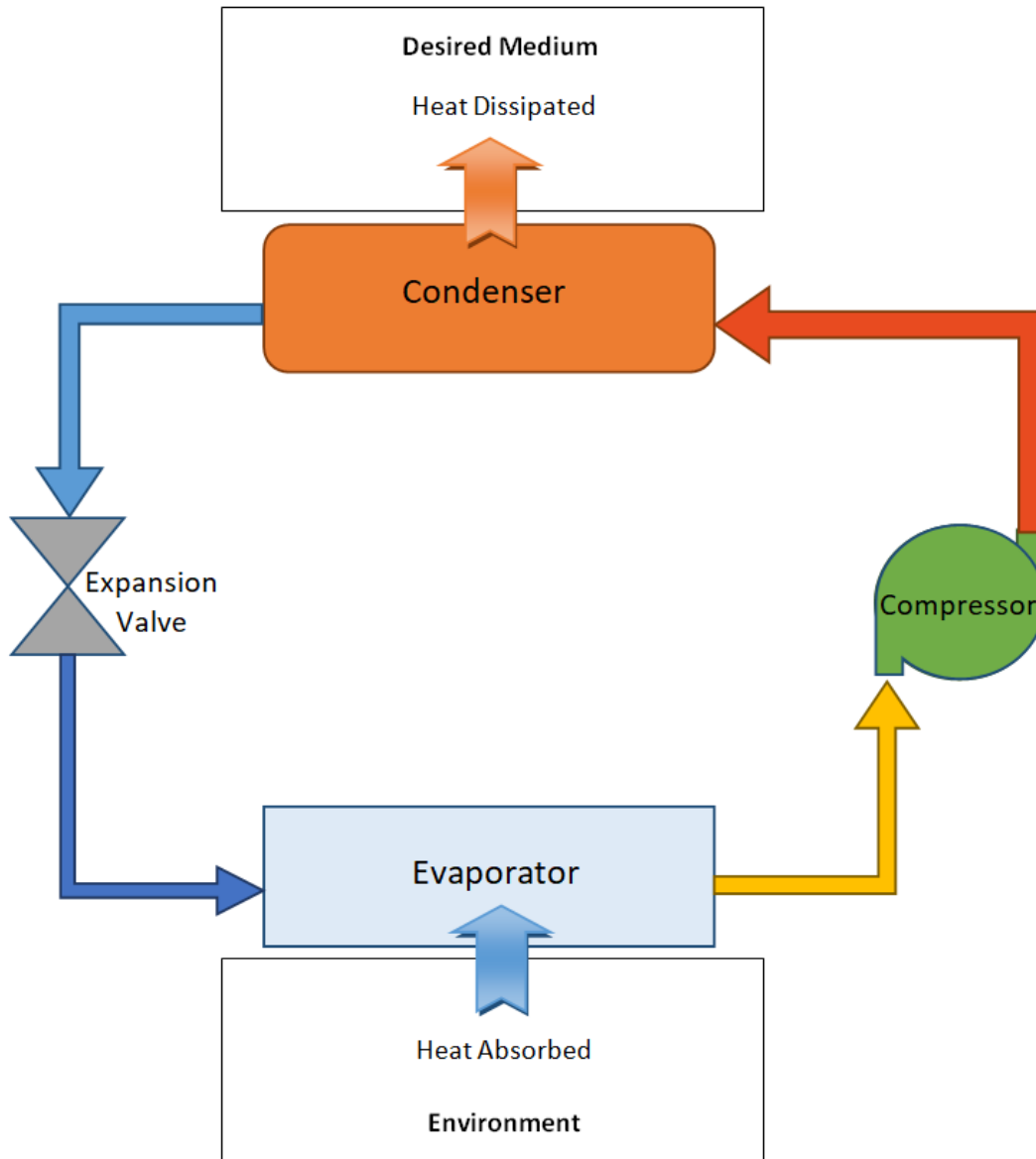


Figure 3: Concept Of A Heat Pump Cycle

2.5.1 Vapor Compression Cycle

The vapor compression cycle is used to transfer heat from one reservoir to another. In this cycle, there are four major components required to transfer heat: an evaporator, a condenser, a compressor and an expansion valve.

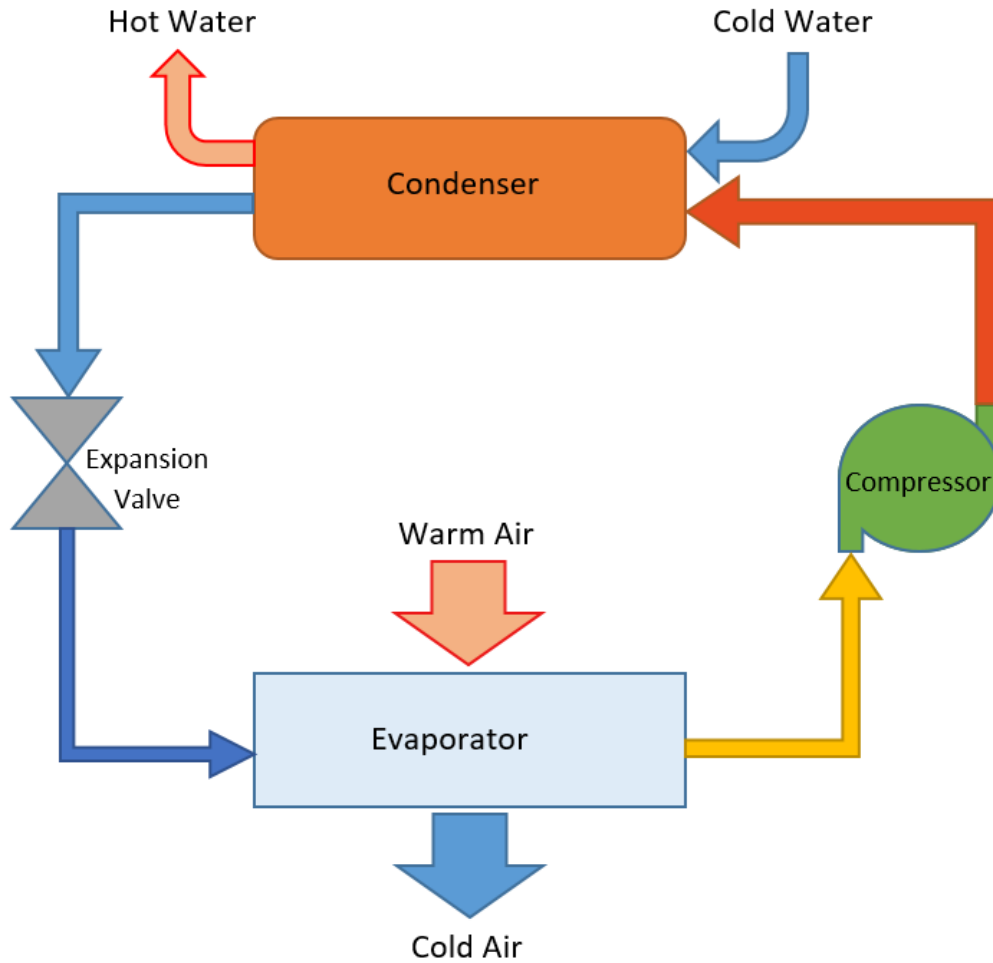


Figure 4: Demonstration Of How Water Heater Works And How Components Are Used

Heat is absorbed into the refrigerant at the evaporator. Refrigerant entering the evaporator is a liquid and vapor mixture and its temperature is also below the temperature of the medium it is absorbing the heat from. As the refrigerant travels through the evaporator, it absorbs heat which increases the refrigerant's temperature. Refrigerant exiting the evaporator is either purely saturated vapor, where the refrigerant has reached saturated vapor temperature, or it is superheated vapor where the temperature of the refrigerant has exceeded saturated vapor temperature. The pressure of the refrigerant does not change through the evaporation process and this pressure is defined as the low pressure of the vapor compression cycle [6].

The refrigerant then flows into the compressor where the pressure of the refrigerant is increased from low pressure to a higher pressure. This higher pressure is defined as the high pressure of the vapor compression cycle. As the pressure of the refrigerant is increased, the temperature of the refrigerant also increases. Refrigerant exiting the compressor is a superheated vapor [6].

The refrigerant is then pumped into the condenser where heat is dissipated from the refrigerant into the surrounding medium. The temperature of the refrigerant is greater than the medium that heat will be dissipated into. As the refrigerant travels through the condenser, the temperature of the refrigerant drops from the superheated vapor temperature to either saturated liquid temperature, the temperature at which the refrigerant will be liquid at high pressure, or subcooled liquid temperature, the temperature at which the refrigerant is cooled below the saturated liquid temperature at high pressure. During the condensation process of the refrigerant, the pressure stays constant at high pressure [6].

Refrigerant then flows through the expansion valve where the pressure of the refrigerant drops from high pressure to low pressure. During this expansion process, the liquid refrigerant at high pressure is changed back into liquid and vapor mixture at low pressure. This refrigerant mixture then flows into the evaporator for the cycle to repeat [6].

To further clarify system operation, exit of the expansion valve to the entrance of the compressor is considered to be the low pressure side. From the exit of the compressor to the entrance of the expansion valve is considered to be the high pressure side. It is assumed the pressure is constant along these paths and that the compressor is generally rated for an exit pressure where the regulator controlling the compressor has exit pressure settings.

2.5.2 Finding Vapor Compression Parameters

Properties, such as temperature, pressure, enthalpy, and entropy, must be calculated at the various states of the vapor compression cycle to gain an understanding of how the refrigeration system or a heat pump system is performing. There are four major states in the cycle that must be calculated [6][4]. For the consistency and readability of this report, state one will be set at the exit of the evaporator, state two will be set at the exit of the compressor, state three will be set at the exit of the condenser, and state four will be set at the exit of the expansion valve as described by figure five below. Furthermore, figure six shows the vapor compression cycle on a pressure and enthalpy (p-h) diagram with the four states whereas figure seven shows the vapor compression cycle on a temperature and entropy (t-s) diagram with the four states.

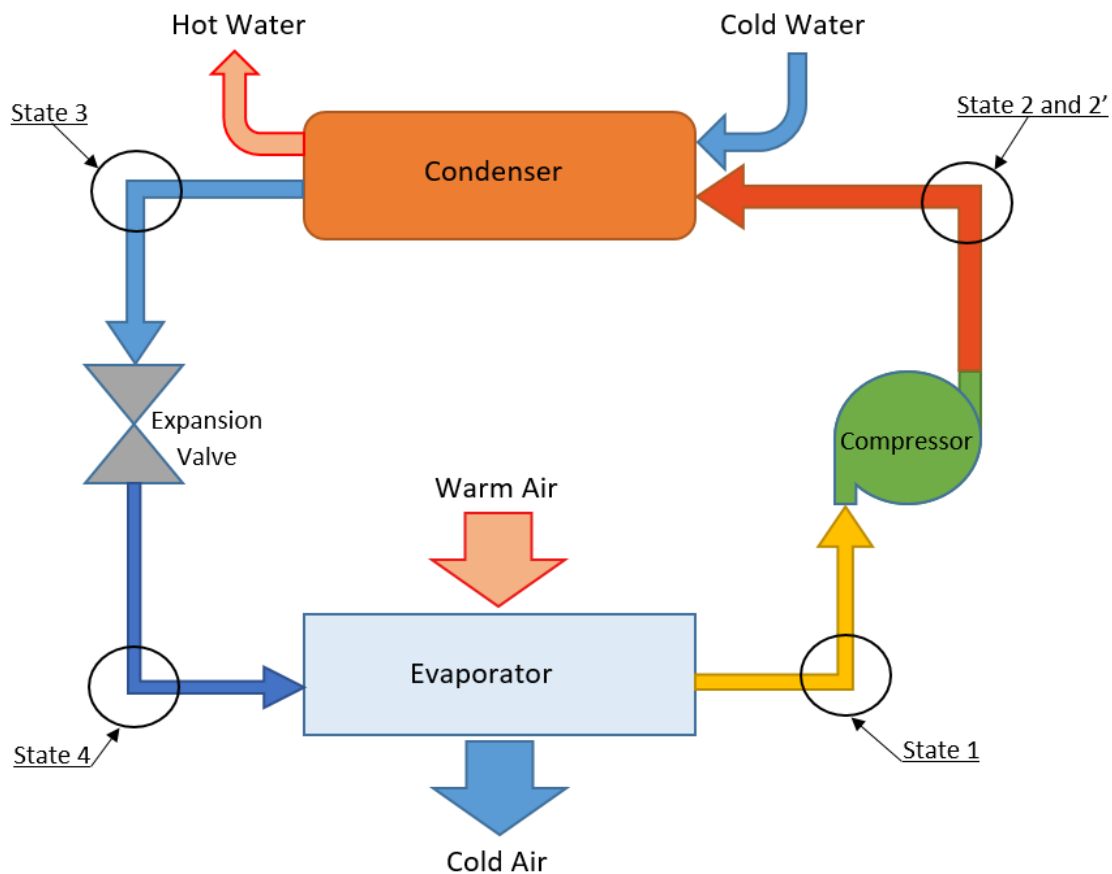


Figure 5: Demonstration Of How Water Heater Works And How Components Are Used

2.5.2.1 Finding State One Properties

State one pressure, known as the low pressure of the system, and temperature must be known to accurately calculate enthalpy and entropy. With the known temperature and pressure values, enthalpy and entropy values may be looked up in the thermodynamic properties table of the given refrigerant [4].

If one of the two required values is unknown, a valid assumption of the refrigerant being as a saturated vapor form must be made at this state to find the unknown values as shown in figures six and seven [4].

2.5.2.2 Finding State Two Properties

State two pressure, known as the high pressure of the system, must be known to accurately find the other three values. In the ideal case of isentropic compression, where the process is reversible and there is no loss of work energy, the entropy is constant going from state one to state two as shown in figure seven. With the given high pressure and state one entropy, the other unknown values may be found from the given refrigerants' thermodynamic properties table [4][5].

2.5.2.3 Finding State Three Properties

State three temperature and high pressure values must be known to accurately find the other two unknown values from the thermodynamic properties table of the given refrigerant. If temperature is unknown, a valid assumption of the refrigerant being as a saturated liquid form at this state must be made to find the unknown property values as shown in figures six and seven [4].

2.5.2.4 Finding State Four Properties

State four pressure, low pressure, must be known to accurately find the other three unknown values. Furthermore, the enthalpy value at state three will be the same as state four, as shown in figure six. With the low pressure value and state three enthalpy value, the other unknown values may be found at state four from the thermodynamic properties table of the given refrigerant. The quality, fraction of the liquid and vapor mixture being vapor, may also be found with the state three enthalpy value [4].

2.5.2.5 Finding State Two Prime Properties

State two prime properties account for inefficiency of the compressor as shown in figures six and seven. Work energy of compressor, mass flow rate of the refrigerant, and state one enthalpy must be known to calculate state two prime enthalpy. If the mass flow rate of the refrigerant is unknown, thermal energy dissipated at the condenser and state three enthalpy must also be known in order to calculate state two prime's enthalpy. With state two prime's enthalpy and the high pressure value known, other unknown values, temperature and entropy, at state two prime may be found from the the thermodynamic properties table of the given refrigerant [5].

2.5.3 Calculating Thermal Energy Exchanged At Condenser

Thermal energy dissipated at the condenser from the refrigerant is calculated by the change in the enthalpy, between state two prime and state three, per unit mass times the mass flow rate of the refrigerant [4]. The following equation calculates thermal energy dissipated from the refrigerant at the condenser:

$$Q_H = \dot{m}(h_{2'} - h_3)$$

Thermal energy dissipated at the condenser is also equal to the energy absorbed by the fluid being heated at the condenser. This is equal to the mass flow rate of the fluid multiplied by the temperature change of the fluid multiplied by the heat capacity of the fluid [4]. The following equation is used to calculate thermal energy absorbed into the fluid that is being heated:

$$Q_H = C_p \dot{m}_f (T_{out} - T_{in})$$

2.5.4 Calculating Work Energy Of Compressor

Electrical work energy of the compressor is directly related to the mass flow rate of the refrigerant multiplied by the enthalpy change between state one and state two prime [4]. This is equal to heat energy gained from the compressor per unit mass of refrigerant going from state one to state two prime as described by the equation below:

$$W_{in} = \dot{m} (h_{2'} - h_1)$$

2.5.5 Calculating Actual Compressor Exit Energy

State two prime's enthalpy can be calculated by first solving both, thermal energy dissipated equation and electrical work compressor equation for mass flow rate of refrigerant as described below:

$$\frac{Q_H}{(h_{2'} - h_3)} = \dot{m}$$

$$\frac{W_{in}}{(h_{2'} - h_1)} = \dot{m}$$

Then both of those equations can be set equal to each other and solved to find the enthalpy at state two prime as described below:

$$h_{2'} = \frac{h_1 Q_H - h_3 W_{in}}{Q_H - W_{in}}$$

Furthermore, if the value of the thermal energy dissipated from the refrigerant at the condenser is unknown, then the equation for thermal energy absorbed by the fluid at the condenser, as discussed in section 2.5.3, can be substituted in as described below:

$$h_{2'} = \frac{h_1 (C_p \dot{m}_f (T_{out} - T_{in})) - h_3 W_{in}}{(C_p \dot{m}_f (T_{out} - T_{in})) - W_{in}}$$

2.5.6 Calculating COP Of The Vapor Compression System

Coefficient of performance, COP, measures the performance of the vapor compression system. It is the ratio of the thermal energy output to work input. There are two types of COP for a vapor compression system. The first type is the ideal COP where the system is assumed to have isentropic compression of the refrigerant through the compressor. The second type is the actual COP where the system takes into account the effects of the compressor's inefficiency on the system [6]. The ideal COP is always better than the actual COP. Ideal COP can be compared with actual COP to determine if the system is efficiently running or if it can be improved.

Furthermore, there are two different ways of calculating efficiency for vapor compression system depending on whether the system is used as a heat pump system or a refrigeration system [6]. This project specifically focuses only on heat pump systems and therefore only heat pump equations will be explained and used.

Ideal COP for heat pumps are calculated as the ratio of thermal energy output, which is heat energy lost between state two and state three, to work energy input, which is the energy gained between state one and state two [6]. The following equation calculates ideal COP for heat pumps:

$$COP_{Ideal} = \frac{\text{Ideal Heat Energy Output}}{\text{Ideal Work Input}} = \frac{h_2 - h_3}{h_2 - h_1}$$

The same theory is used for calculating actual COP as above. However, instead of using state two values, state two prime values are used to account for compressor's inefficiency [6].

Actual COP for heat pump is calculated using the following equation:

$$COP_{Actual} = \frac{\text{Actual Heat Energy Output}}{\text{Actual Work Input}} = \frac{h_{2'} - h_3}{h_{2'} - h_1}$$

2.5.7 Calculating Efficiency Of The Compressor

Efficiency of the compressor indicates how well the compressor is running verses ideal isentropic conditions [6]. The following equation is used to determine the efficiency of the compressor:

$$\eta = \frac{\text{Ideal Compressor Work}}{\text{Actual Compressor Work}} = \frac{h_2 - h_1}{h_{2'} - h_1}$$

2.6 Field Units

Field units currently being monitored are specifically heat pumps. These heat pumps are primarily used to heat water. Heated water may then be used as hot water provided to the house or it may be used as a medium to heat the house. If the heating mode is for hot water, then the water being heated is circulated between the condenser and hot water storage tank. If the heating mode is for heating the house, then the hot water is circulated from the condenser to the entire house and back. If the house uses both modes of heating, then the modes are toggled by a switch valve located at the entrance of the condenser and the exit of the condenser. Figure eight below describes the general setup of the heat pumps.

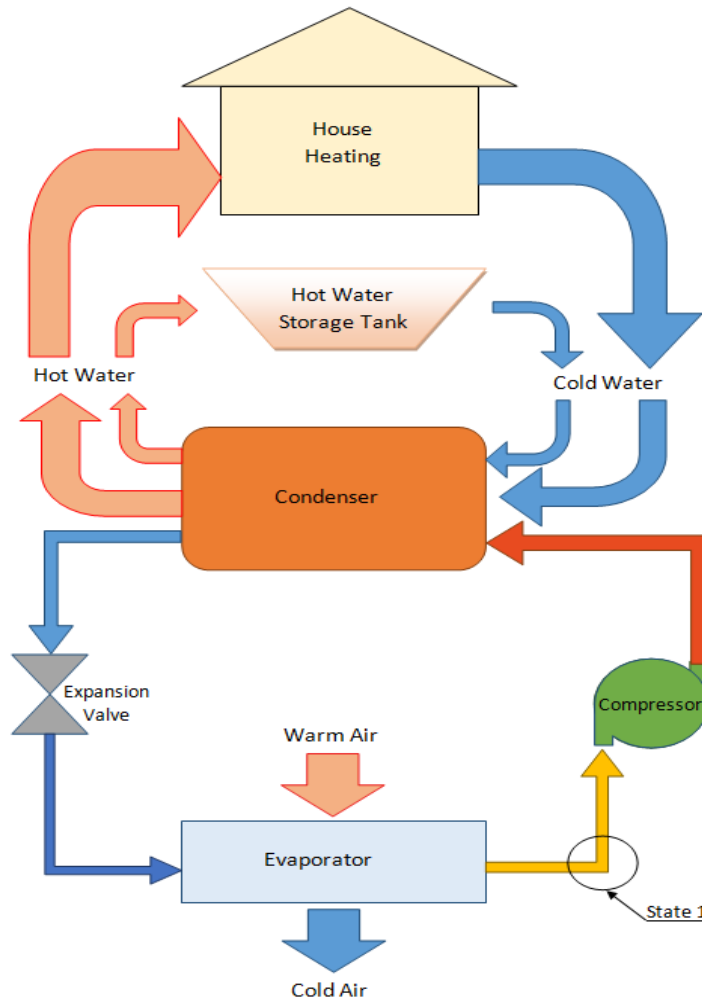


Figure 8: Field Unit Heat Pump Setup

2.6.1 What Is Measured At The Field Units

Currently there are no measurements taken from the field unit heat pumps related to the vapor compression cycle because it is treated as a black box. This means the high and low pressures of the system, the mass flow rate of the refrigerant and the temperatures of the refrigerant are not measured. However, the temperatures of the fluids entering and exiting the evaporator and condenser are measured. The flow rate of the fluid being heated is also measured along with the power consumption of the compressor. The table below describes where the measurements are taken, the type of measurements taken and associated sensor names.

Sensor Number	Sensor Name (In German)	Sensor Measurement Description	Units Measured In	Factor
03	Sein1	Temperature of water entering condenser	°C	100
09	Aaus1	Temperature of water exiting condenser for hot water	°C	100
22	Vaufl	Flow rate of water going through condenser for hot water	l/hr	1
01	Saus1	Temperature of water exiting condenser for house heating	°C	100
21	Vsenke	Flow rate of water going through condenser for house heating	l/hr	1
27	PelV	Power usage by the compressor	W	1
07	Qein1	Temperature of air entering evaporator	°C	100
05	Qaus1	Temperature of air exiting evaporator	°C	100

Table 1: List Of Field Unit Sensors And Their Measurement Description

2.7 Test Bench Heat Pump System

The test bench heat pump is located in the laboratory of NTB. This heat pump serves as a water heater where it gains heat from the surrounding air. It is primarily used to test various aspects of the vapor compression cycle. Unlike the field units, this test unit does measure parameters of the vapor compression cycle including the high and low pressures of the system, the temperatures of the refrigerant at the four states of the cycle, the temperatures of the fluids entering and exiting the evaporator and condenser, flow rate of the fluid through the condenser and the power consumption of the compressor. This test unit, however, does not measure the flow rate of the refrigerant. Table two and figure nine below depict where the measurements are taken, the type of measurements taken and their associated sensor names.

Sensor Number	Sensor Name (In German)	Sensor Measurement Description	Units Measured In	Factor
1	T_Wasser_Vorlauf	Temperature of water entering condenser	°C	1
2	T_Wasser_Ruecklauf	Temperature of water exiting condenser	°C	1

3	V_dot_Wasser	Flow rate of water going through condenser	l/min	1
4	P_elek_Verdichter	Power usage by the compressor	W	1
5	T_ueberhitzt	Temperature of refrigerant at state one	°C	1
6	p_Verdampfung	Pressure of refrigerant at state one [the low pressure]	bars	1
7	T_unterkuehlt	Temperature of refrigerant at state three	°C	1
8	p_Kondensation	Pressure of refrigerant at state three [the high pressure]	bars	1
9	T_expandiert	Temperature of refrigerant at state four	°C	1
10	T_luft_ein	Temperature of air entering evaporator	°C	1
11	T_luft_aus	Temperature of air exiting evaporator	°C	1

Table 2: List Of Test Bench Heat Pump System Sensors And Their Measurement Description

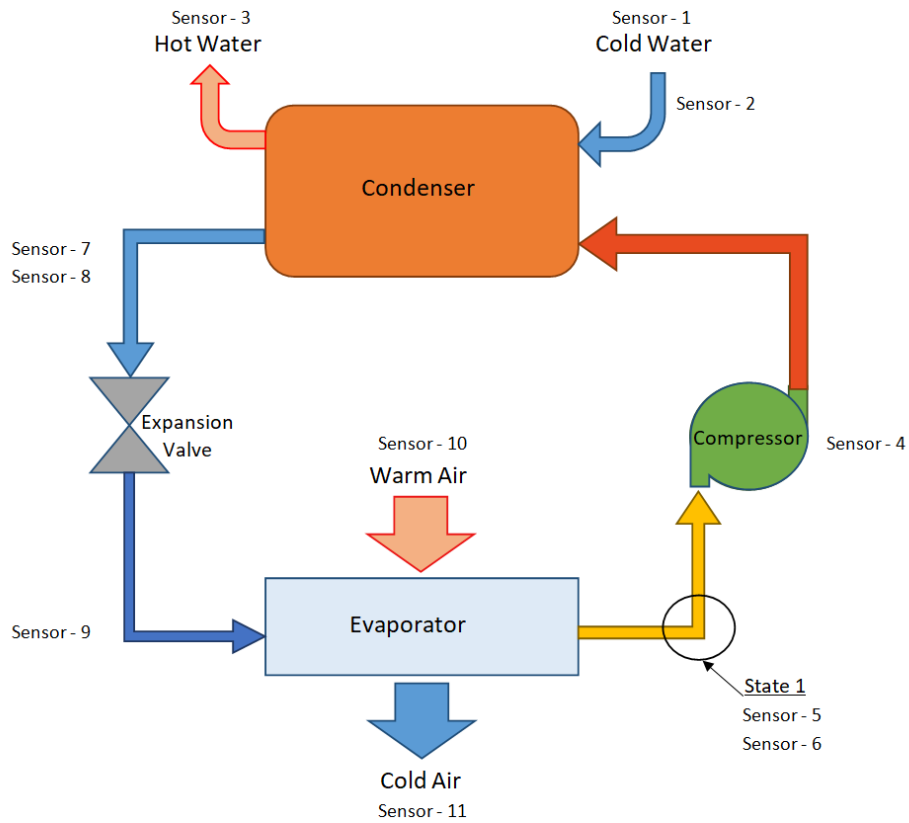


Figure 9: Test Bench Heat Pump And Associated Sensors Setup

2.8 Background Used In Implementation

The information provided in this background section was used as the primary concepts in creating the programs discussed in the methodology section. The thermodynamic process for analysis, as discussed in this section, was turned into code which was used to analyze field units. The code was then tested on the NTB's test bench heat pump to assess its accuracy.

Chapter 3: Methodology

Initially, the goal of the project was to automate the existing process for data file transfer without many modifications. At first this seemed like a straightforward process but as time went on many roadblocks arose that prevented the completion of this approach. As such the team had to make program changes mid way through the project. Furthermore, it was also decided that creating a heat pump analysis program would be also beneficial for NTB engineers to use for their efficiency analysis process.

3.1 First Design For File Downloader Program

The first design was to make a program that utilized FTP to download the files from the PLC and then upload them to the database website, using a local machine as an intermediate, backup location. This program would run every night, allowing data collection to be much more frequent than before. If a PLC were unavailable or if it were running out of space, an email would be sent to the engineers at NTB telling them about the problem. It was known that the PLC had an FTP server available and, after some research, it was found that the hosting service used for the database website, HostPoint, also had an FTP service available.

When deciding on a language to use, Python was chosen because the MQP team and engineers at NTB are familiar with it, and it has many frameworks and libraries (e.g. ftplib, smtplib) that would help create the functionality needed in a simple, straightforward way. Another plus is that several versions are already installed on HostPoint, in case the program were to run there. The pros and cons for different languages considered can be found in Table 1.

Language	Pros	Cons
Python	<p>MQP team familiar.</p> <p>NTB engineers familiar.</p> <p>Many libraries.</p> <p>System Agnostic.</p>	Slow.
PHP	<p>Used on the database website.</p> <p>NTB Engineers are familiar.</p> <p>System Agnostic.</p>	MQP team not familiar.
Java	<p>MQP team familiar.</p> <p>System Agnostic.</p>	
JavaScript/NodeJS	<p>MQP team familiar.</p> <p>System Agnostic.</p> <p>Many libraries.</p>	Not installed on HostPoint
C/C++		<p>Harder to write networking code.</p> <p>Not system agnostic.</p>

Table 3: Pros And Cons Of Different Environments For Running The Program

This Python program would use ftplib’s FTP object to download data from a PLC, save it on the local hard drive, then use another FTP object to upload it to the destination folder on HostPoint. It would iterate the same process over all the PLC modules. When downloading data,

if there was a connection error or if the PLC's storage was almost full, it would send an email using smtplib's SMTP object (with TLS as security).

FTP is not considered secure, since all protocol communication is done in clear text, but unfortunately the PLCs do not have SFTP support. Although FTP is not secure, LinkManager's tunnel does use encryption. Running the FTP through the tunnel makes it secure between the SiteManager and the local PC at NTB.

3.1.1 User Interface

In addition to the program that would run nightly, it was decided that a graphical front-end should be provided to the user as well. Using PyQt5, a system-agnostic user interface library that allows What-You-See-Is-What-You-Get (WYSIWYG) design, a graphical front-end was created. This allows the user to choose any day, month, or year and download data from any one or more PLCs. Figure ten below is a screenshot of this program. More screenshots can be found in Appendix B.1.

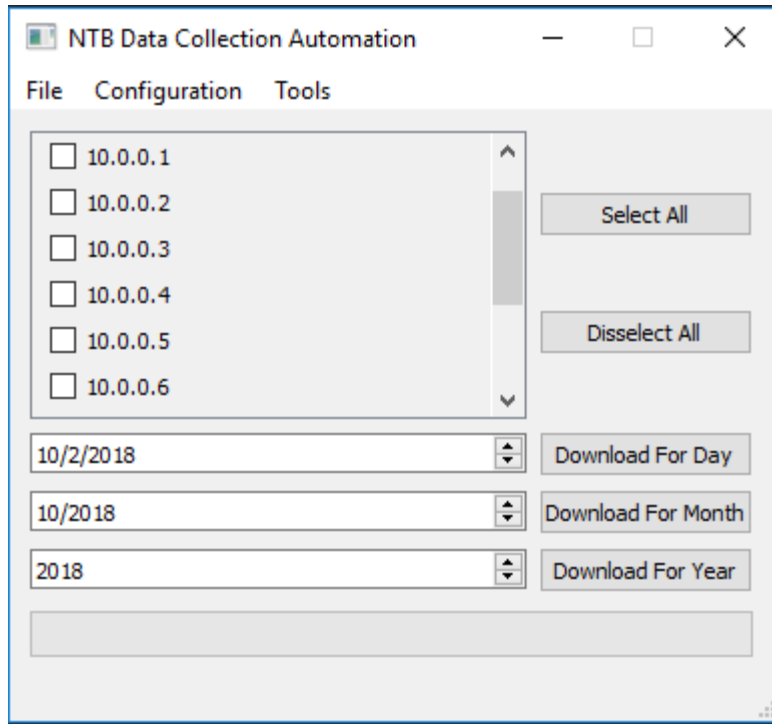


Figure 10: Screenshot Of File Downloader Program

This program was fairly easy to create because when creating the initial command line program, functionality was broken up into different files and classes so that they could be used modularly any way required in the future. For instance in the UI program, the SPSLib class was used heavily because it was needed to connect to SPS (PLC in German) modules, however the email classes were not used because that functionality was not needed for the UI.

3.1.2 Major Roadblock

An early issue was getting the program to iterate over the PLCs. It was at this point that it was discovered to be impossible to connect to more than one PLC at a time using LinkManager, and could not programmatically choose a PLC on LinkManager. This led to researching ways to automate LinkManager.

Our first idea was to see if LinkManager had an Application Programming Interface (API) that would let us login and logout of PLCs using Python. After much research it was

found that there had previously been a poorly supported API for LinkManager but it was removed several releases ago.

The next option was to find if there was a plugin or add-on to LinkManager that would allow us to achieve such functionality. After several phone calls with B&R support, the team turned towards LogTunnel. As defined in section 2.3.2.3 of this paper, LogTunnel is an add-on to LinkManager that allows the user to connect to a single ‘virtual’ SiteManager that will expose all the PLC modules on the local network. This would allow us to connect once using LinkManager and then simply iterate over the addresses on the local network, eliminating the need to switch PLCs using LinkManager. LogTunnel, however, is a feature of LinkManager that requires an Enterprise license of B&R software, so it was not usable with NTB’s Research license, and this option was scrapped.

At this point, the idea was considered to use a web-scraping technique to programmatically press the buttons on the LinkManager console webpage. This is not a sustainable way to attack this problem because if B&R releases an update that changes the IDs or names of elements on the webpage, then it will cause instability or break our program completely.

3.2 Mid-Term Requirement Changes

Midway through the term, after wrestling with LinkManager for most of the time spent on this project, the team spoke with some coworkers and decided that a different approach was needed to make the program succeed in the time span of our project.

During the first week of the project, it was explained that in the past NTB had the PLC send an email with the files as the attachments to an address, then someone would download the

files and upload them to HostPoint. They stopped doing so about two years prior to this project because there was a bug in the email system in the PLCs’ operating system, causing files to become corrupted frequently.

Since then the bug has been fixed, so it was decided that a program should be created that would simply download files from the email attachments, back them up to NTB’s cloud, and upload the files to HostPoint where they would be processed. The firmware engineer decided that he would add the email functionality back into the PLCs’ software to allow us to accomplish this. At this point, the initial program was put on the back burner and the second program began production.

3.2.1 Second Design For File Downloader Program

The second design took many of the same design aspects from the first, however several changes were made. It was decided that it be best for the program to run on HostPoint, which is a BSD (Berkeley Software Distribution) system (based on Unix). Running the software on HostPoint would eliminate the maintenance of an extra machine, but would contribute towards the HostPoint compute quota. Our pros and cons list can be found in Table 2 below.

System	Pros	Cons
HostPoint	Less maintenance. MQP team is familiar	Off site.
Local Windows Machine	High control.	Maintenance. Windows update could break software.

NTB Servers	Lower maintenance than a local machine. Managed by NTB IT.	Requires work by NTB IT. Not sure what resources NTB has available.
-------------	---	--

Table 4: Pros and Cons Of Different Environments For Running Code

It was decided that the software should use imaplib’s IMAP_SSL object to securely download attachments from the emails. The team also decided that it is useful to use a thread pool to allow for concurrent downloads from the email server. The metrics for each method considered can be found in Appendix A.3. Once downloaded, the attachments are parsed out of the emails as base64 strings and are decoded. After being saved to the hard drive, the program determines which files are new and copies them into the destination folder on HostPoint as well as backing them up to NTB’s cloud storage using the Webdav protocol.

3.2.2 Re-Enabling Emails From PLCs

To re-enable emails from the PLCs with the files as the attachments, a special program needed to be added to the PLCs’ operating system. The team worked with the firmware engineer to adapt an example program written in B&R Structured Text and modified the configuration of the SiteManager and PLC to be able to send emails with the files attached.

3.3 Documentation

All the programs have extensive documentation on how to install and use them as well as the way they work under the hood. All documentation can be found on the WPI project page for this MQP and in all the repositories.

3.4 Heat Pump Analysis Program

The heat pump analysis program analyzes the field unit heat pumps to determine their efficiency under working conditions. Each field unit selected within the program will be analyzed for its thermodynamic properties at the four states of the vapor compression cycle. Those properties will then be used to find the coefficient of performance of the system and the efficiency of the compressor.

The heat pump analysis program will take two independent thermodynamic property tables for the given heat pump, a saturation table and a superheated table, along with a data file for the selected heat pump as its inputs. The program will then parse the data file with the all sensor measurements into lists of individual sensor and their corresponding measurements. Using those lists the program will find the thermodynamic properties and calculate the efficiency of the given heat pump for each timestamp within the data file.

Once the program finishes analyzing the entire data file, it will record the thermodynamic properties of the four states and the efficiency of the system for each time stamp in an output Excel file. The output Excel file will be organized by timestamps in a row wise fashion with the efficiency and the thermodynamic properties of each state recorded within that row following the associated timestamp.

The program will also check for errors, such as if there are required sensor measurements missing, if the calculated thermal energy dissipated is negative, or if the calculated state two prime's enthalpy is larger than largest value found in the thermodynamic properties table. Furthermore, the program also provides the type of heating mode in use, such as if it is either in water heating mode or house heating mode and whether the heat pump is turned on or off. This information is also recorded in the output Excel file for each associated timestamp.

This program can primarily be used to understand the vapor compression cycle of the field unit heat pumps since they are currently treated as a black box. Furthermore, this program can also be used to find the reasons for errors or irregularities in NTB's data analysis of the field unit heat pumps to determine if they are caused by the heat pump system itself or by external sources.

3.5 How The Field Heat Pump Units Will Be Analyzed By The Program

Field unit heat pumps will be analyzed using the following sensor measurements parsed from the data files: temperature of fluid entering the evaporator such as the surrounding environmental air, temperatures of the fluid entering and exiting the condenser such as the water being heated, flow rate of the fluid going through the condenser such as the water, and the electrical energy consumed by the compressor. Using the temperatures of the fluid entering and exiting the condenser along with its flow rate, the thermal energy absorbed by the fluid will be calculated first. Second, the thermodynamic properties at state one will be found using the saturation table of the given refrigerant and the temperature of the fluid entering the evaporator. State two and state three thermodynamic properties will be found next. State two properties will be found using the temperature of the fluid entering the condenser and the superheated table of the given refrigerant. State three properties will be found using the saturation table. After that, state two prime's enthalpy will be calculated and using the superheated table, the other thermodynamic properties at this state will also be found. Lastly, both, ideal and actual, coefficient of performance of the system will be calculated along with the efficiency of the compressor.

If the compressor is off, the analysis will stop for the current timestamp and move onto the next one. The same will be done if any errors occur where it will record results up to the point of error before moving onto the next timestamp.

3.5.1 Compensate For Missing Measurements

Four assumptions are made to analyze the vapor compression cycle of the given heat pump to compensate for the missing measurements as discussed in section 2.6. The first two assumptions are made at state one where the first assumption involves temperature of the refrigerant being twenty degrees Celsius below the fluid's temperature entering the evaporator. The Second assumption involves the refrigerant at state one being a saturated vapor. The Third assumption involves temperature of the refrigerant at state three being twenty degrees Celsius above the temperature of the fluid entering the condenser. The last assumption, assumption four, involves the refrigerant at state three being a saturated liquid.

These assumptions are used primarily because they are reasonable assumptions as described in the "Pinch Analysis For the Effective Use" guide book and integrated process design research paper [1][3]. In the Pinch Analysis guide book, it was recommended that a minimum temperature difference of ten to twenty degrees Celsius must be used for sufficient heat transfer between the evaporator and the surrounding air where heat transfer coefficients are low. However, a minimum temperature difference of thirty to forty degrees Celsius must be used between the heat dissipating medium and heat absorbing medium for sufficient heat transfer within heating and cooling applications [1]. Furthermore, in the research paper, a minimum temperature difference of twenty degrees Celsius was assumed for heat transfer in determining the heating and cooling requirements for an industrial park [3]. Taking these two documents into

account, an initial conclusion of using a minimum temperature difference of twenty degrees Celsius was reached to analyze the field unit heat pumps.

Furthermore, these assumptions are also verified for their validity through running a test on the test bench heat pump system as discussed in section 3.6 below. The results of running that test, as discussed in chapter four, showed that using a minimum temperature difference of nineteen degrees Celsius as opposed to twenty degrees Celsius would produce slightly less error in efficiency calculations using assumptions for heat pumps.

3.6 Testing Assumptions On The Test Bench Heat Pump

Assumptions made to analyze field units were tested on the test bench heat pump system to determine their validity. This was accomplished by comparing the actual analysis results with the assumed analysis results. Actual analysis results used measurements taken from the test bench system such as high and low pressure measurements and the temperature measurements of the refrigerant at the four states. Assumed analysis results used assumption, as described in the section 3.5.1, to analyze the test bench heat pump system.

Analysis was done using the heat pump analysis program described in section 3.5 with a slight modification. The analysis were done twice, one time with actual measurements and one time with assumptions, for each timestamp in the test data file.

3.6.1 Collecting Test Bench Heat Pump Data

Test bench heat pump was run for three hours collecting measurements from each sensor every two seconds. Total amount of data samples collected during this testing phase was over six thousand samples. During the testing period, three parameters were varied to replicate actual

behavior of the heat pumps like the field units. Parameters that were varied are flow rate of the water being heated, frequency of the evaporator fans, and target temperature of the heated water.

3.6.1.1 Varying Flow Rate Of Fluid Being Heated

Flow rate of water was varied seven times during the experiment between five liters per second to twenty liters per second. For each variation interval, the other two parameters were also varied to make the measurements as realistic as possible by encompassing all the other variable variations. Furthermore, changing the flow rate would simulate realistic demand of hot water from the hot water heater such as the ones out in the field. Figure eleven below shows the flow rate of water during the data collection phase.

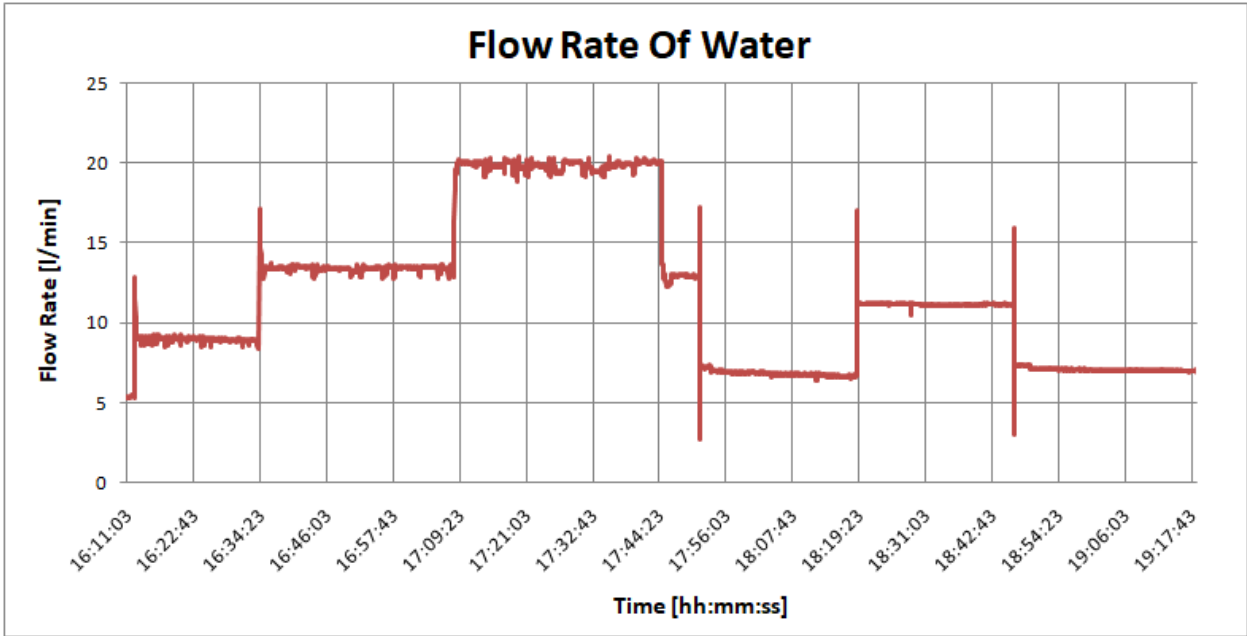


Figure 11: Chart Showing The Flow Rate Of Water Throughout The Testing Process

3.6.1.2 Varying Frequency Of The Evaporator Fans And Room Temperature

The frequency of the evaporator fans was varied to change the flow rate of air flowing through the evaporator. This was done to vary the amount of thermal energy absorbed into the refrigerant from the air flowing through the evaporator. The temperature of the air in the room

was also varied over time. These variations would ultimately simulate heat pumps undergoing exposure to different air temperatures and varying weather conditions such as the field units. Actual frequency measurements and flow rates of air were not collected. However, figure twelve below shows the varying temperatures of the air entering and exiting the evaporator as a consequence of varying flow rate with constant thermal energy absorption.

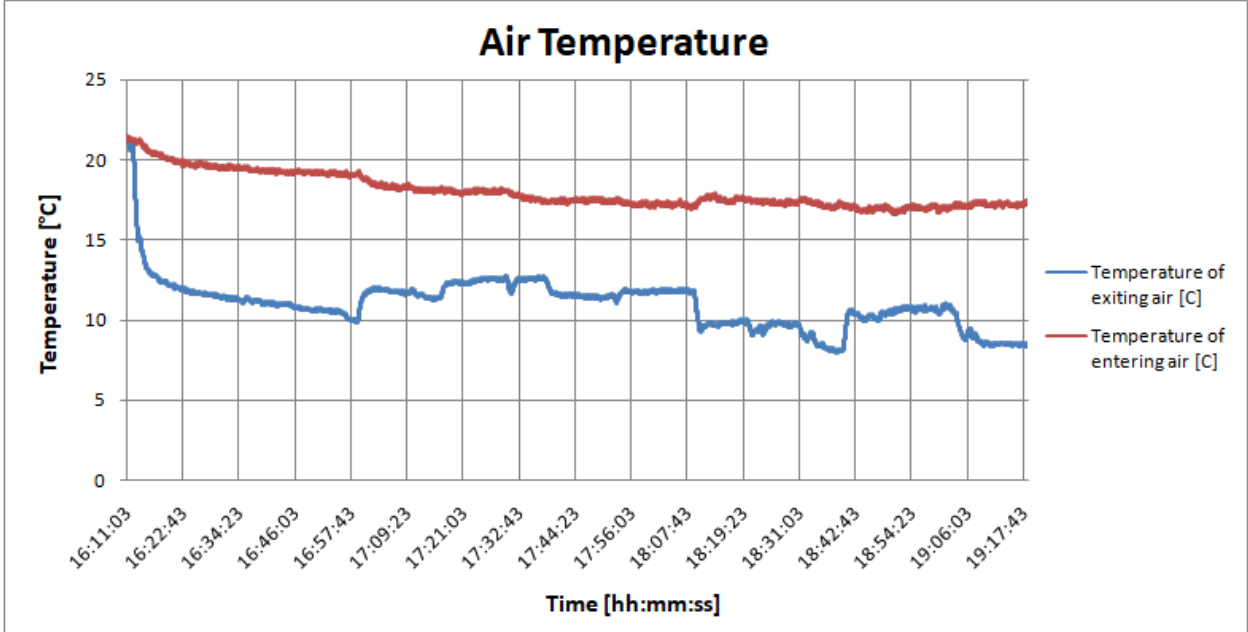


Figure 12: Chart Showing The Temperatures Of Air Throughout The Testing Process

3.6.1.3 Varying Target Temperature Of The Heated Water

Target temperature of hot water was varied over time between seventeen and twenty-five degrees Celsius. This variation was not purely independent because the flow rate of water also influenced the actual hot water temperature. The purpose of this variation was to simulate the effects of different hot water temperature needs on the heat pump cycle as it may be seen at the field unit locations. Figure thirteen below shows the temperatures of water entering and exiting the condenser.

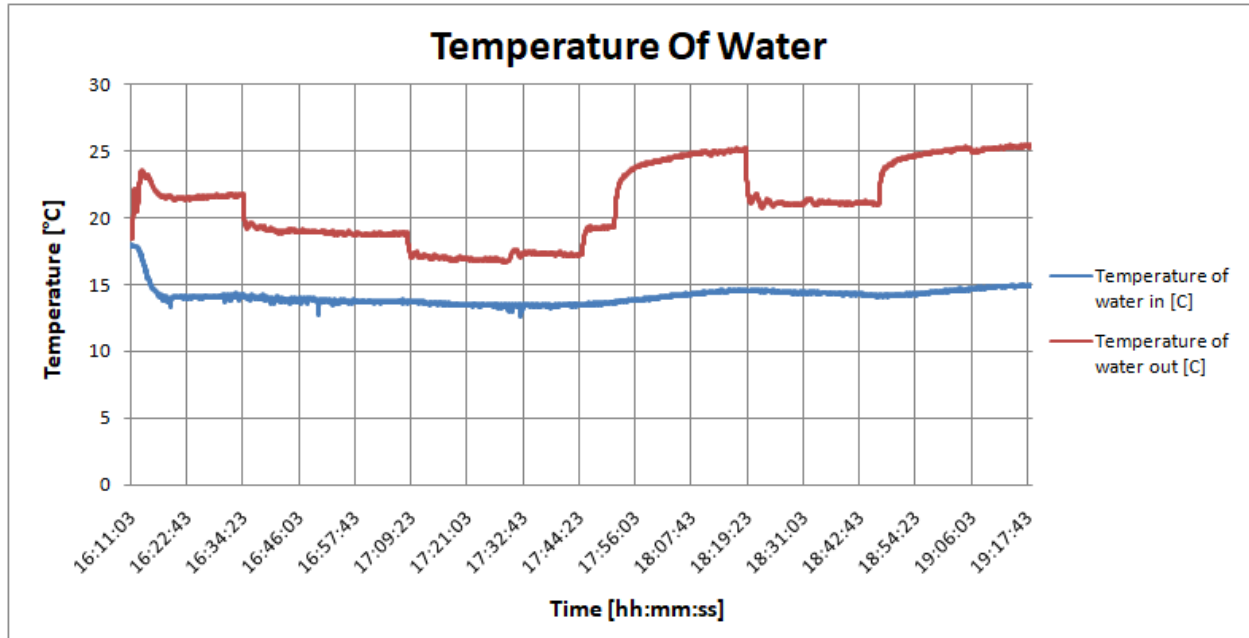


Figure 13: Chart Showing The Temperatures Of Water Throughout The Testing Process

3.6.1.4 Resulting Pressure Changes And Refrigerant Temperatures

Varying the parameters mentioned in the sections above also influenced the high and low operating pressure of the system. Furthermore, those variations influenced the refrigerant temperatures at the four states as well. Pressure and refrigerant temperature variations can be used to further support the validity of the assumptions by having a wider range of operating value that will be analyzed and compared. Figures fourteen and fifteen below show the pressure variations and the refrigerant temperature variations respectively.

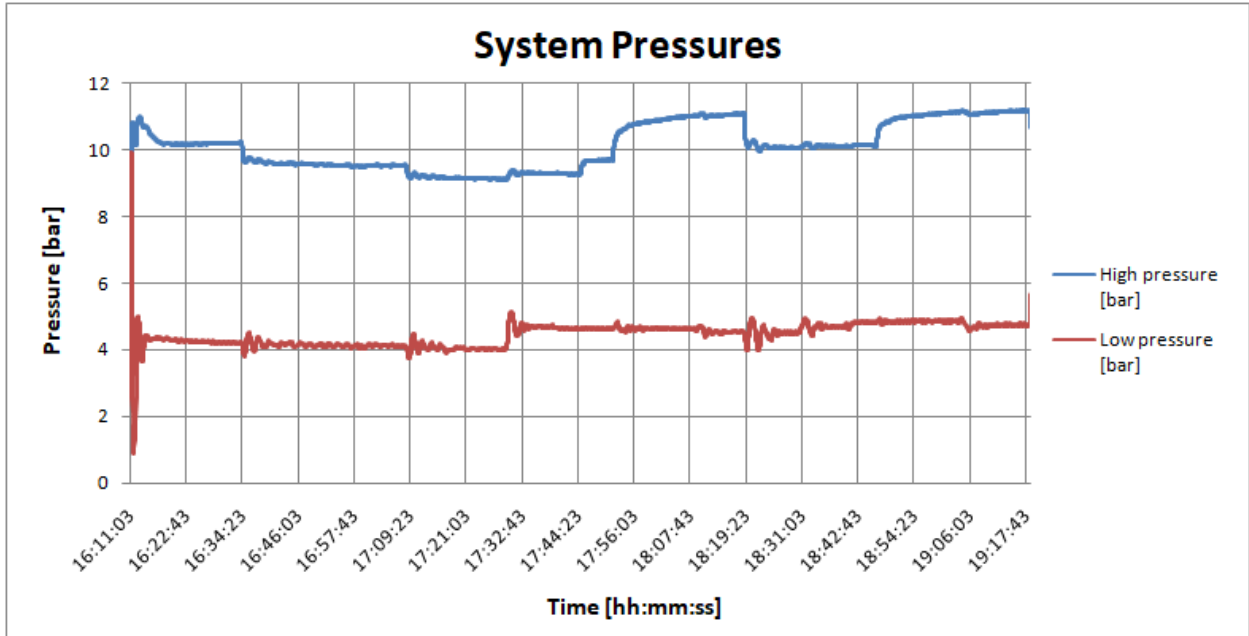


Figure 14: Chart Showing The System Pressures Throughout The Testing Process

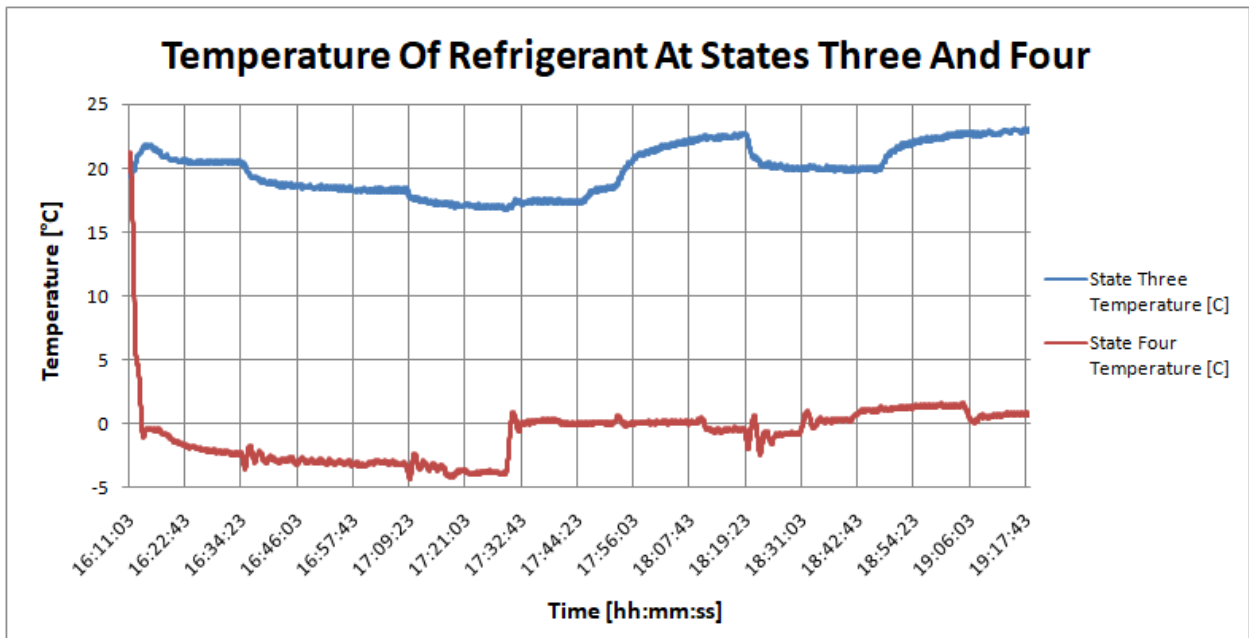


Figure 15: Chart Showing The Temperatures Of Refrigerant Throughout The Testing Process

3.6.2 Testing Different Assumptions Of Delta T's

Different assumptions of temperature differences between fluid and refrigerant at states one and two were also tested. Temperature differences of fifteen, eighteen, twenty, twenty-two

and twenty-five degrees Celsius were tested during the analysis procedures on the same test bench data file. This was done to verify which temperature difference assumption would provide the most accurate calculations.

Chapter 4: Results

Test bench heat pump analysis results are discussed in this section. Primary goal of this section will be to select the most accurate assumptions and prove that those are valid for use when analyzing the field units. Furthermore, this section will also discuss and prove that the heat pump analysis program is accurately computing results for both test bench analysis and for field unit analysis.

4.1 Verifying The Program Results For Test Bench Data Analysis

Results of analyzing three different instances of data measured from the test bench heat pump system are reviewed for program's accuracy. The first instance was explicitly used during the development process of the heat pump analysis program where data was visually extracted directly from the LabView software used to operate the heat pump. Thermodynamic properties at the four states of the vapor compression cycle were then found manually using the extracted data. Both thermodynamic properties table and pressure and enthalpy diagrams were used to understand the process of finding the results. Coefficients of performance and the efficiency of the compressor were also calculated manually for the same reason. Those calculation processes were then coded into the program and tested with the same measurement values. During the program development stage, program testing was accomplished by inputting the measurement values into the program manually.

Results for both of those analysis methods varied by less than one percent when compared for this instance. The LabView software overview for this instance, manual analysis, and results of the program's analysis can be viewed in the appendix C.

Analysis of the next two instances, instance two and instance three, were picked randomly from the test bench's data file created during the test bench heat pump testing phase. Analysis were done manually and by the program for both of these instances. Measurements, however, were entered automatically into the program directly from the test bench's data file for these two instances.

Results of both manual analysis and program analysis were then compared. As in instance one, the results varied by less than one percent which made this program reliable for analyzing the test bench heat pump. The results, manual analysis and the thermodynamic property values at the four states of the vapor compression cycle may be viewed in appendix D and E for these two instances.

4.2 Results Of Testing Different Delta T's On Test Bench

Results of testing different assumptions of temperature differences, as discussed in 3.6.2, are analyzed using the test bench's data file to determine the most accurate assumption among them. For each assumed temperature difference, fifteen, eighteen, twenty, twenty-two, and twenty-five degrees Celsius, an average percent error caused by results calculated using assumptions versus results calculated using actual measurements were reviewed. Specifically, only the ideal coefficient of performance and compressor efficiency errors were reviewed because they indirectly incorporate the other results within their calculations.

A conclusion was drawn after the review where the test results showed that nineteen degrees Celsius temperature difference assumption would produce the least error when analyzing field unit heat pumps despite the initial twenty degrees Celsius temperature difference

assumption being scientifically valid, as discussed in section 3.5.1. Table five and figures sixteen and seventeen below show the test results.

Assumed Temperature Difference (°C)	Type Of Calculation	Average Result With Assumptions	Average Result With Actual Measurements	Difference Between The Two Averages	Percent Error Calculating Result Using Assumptions
15	Ideal COP	13.29	10.10	3.19	31.57
	$\eta_{\text{compressor}}$	0.41	0.55	-0.14	25.90
18	Ideal COP	10.72	10.10	0.62	6.18
	$\eta_{\text{compressor}}$	0.52	0.55	-0.04	6.46
20	Ideal COP	9.46	10.10	-0.64	6.32
	$\eta_{\text{compressor}}$	0.59	0.55	0.04	7.48
22	Ideal COP	8.47	10.10	-1.63	16.17
	$\eta_{\text{compressor}}$	0.67	0.55	0.12	21.76
25	Ideal COP	7.28	10.10	-2.81	27.86
	$\eta_{\text{compressor}}$	0.80	0.55	0.25	44.62

Table 5: Analysis Results For Each Tested Temperature Difference Assumption

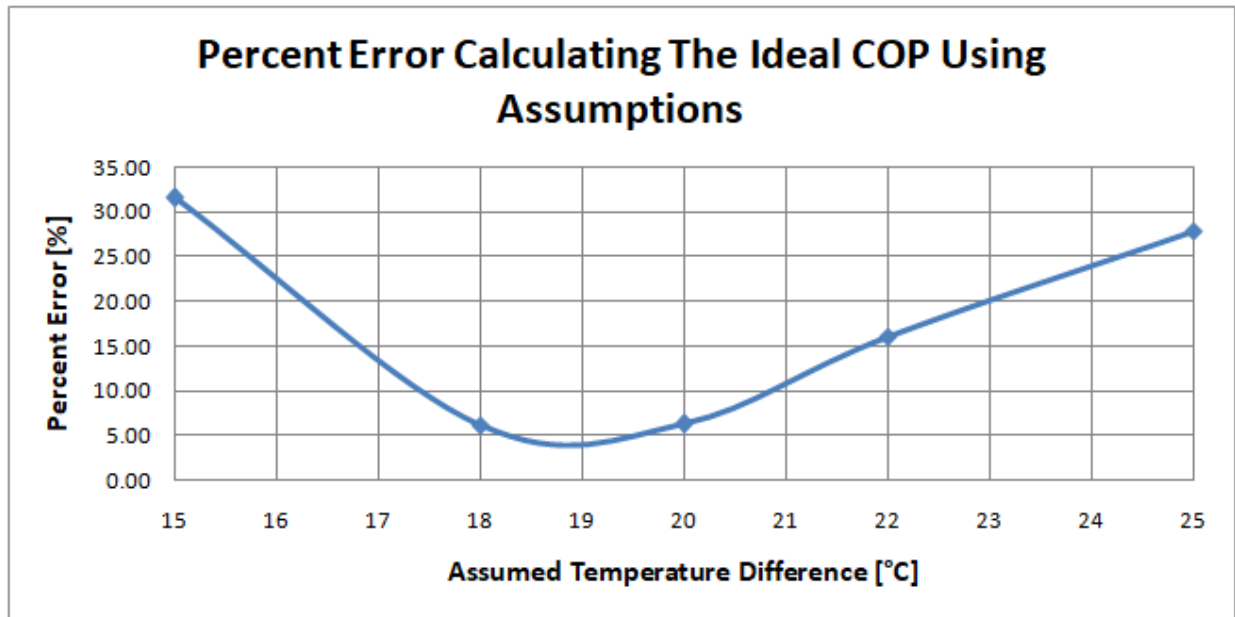


Figure 16: Chart Showing The Percent Errors In Calculating The Ideal COP With Assumptions

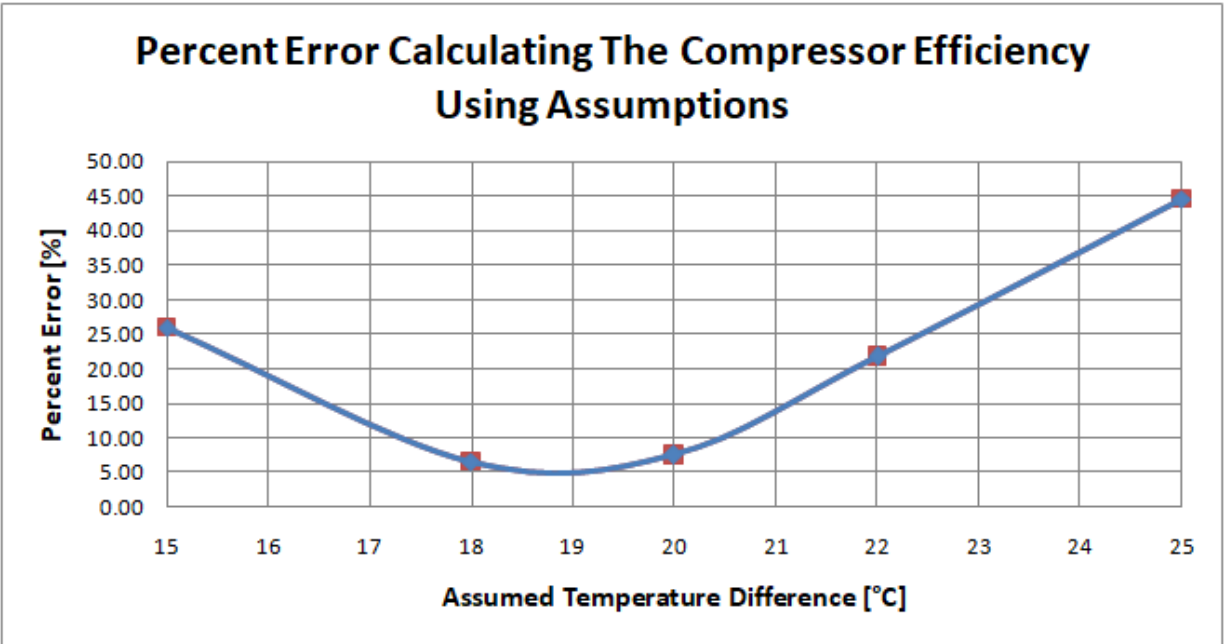


Figure 17: Chart Showing The Percent Errors In Calculating The Efficiency With Assumptions

Chapter 5: Conclusion and Future Work

After overcoming a few challenges along the way, the team had created software that increases the throughput and lowers the latency of NTB's data collection process. The team had created two major programs to downloading files from the PLCs, one of which with two different front-ends: graphical and command line. It was decided that our second, mail-downloader program would be the main program that would be run nightly by a cron job on HostPoint. The initial program would be used as an "in case of emergency" program where the user can manually download data if there is a failure with the main program. With these programs, what used to happen manually once per week now happens nightly and automatically.

Heat pump analysis program was also made along with the data transfer programs for NTB engineers to use. The heat pump analysis program provides an analysis of the vapor compression cycle for each of the field units they are monitoring. This would particularly be beneficial for them in their system analysis process of the field units where they can determine if the reason for lower system efficiency is caused by the heap pump itself or by other external factors.

5.1 Future Work: Expandability

Throughout the project, the team made sure to create modular software so that anyone can extend or use our functionality. For instance, if anyone in the future needs to interface with the PLC modules, they can simply take our SPSLib.py file and import it into their project. Another example is if anyone needs to interface with HostPoint they can use our

HostpointLib.py file. This modularity also allows for the software to be changed to support a change in NTB's infrastructure in a fairly painless manner.

If in the future, NTB decides to purchase a full enterprise license of B&R software that would enable the use of LogTunnel or a similar service, then the original program will become much more useful, since it would allow for the user to be able to download any day's data from any number of PLCs all at once. For the time being it can be slightly useful as it at least eliminates the use of FileZilla, but still requires the use of LinkManager and switching between PLCs manually.

References

- [1] CANMET Energy Technology Centre, *Pinch Analysis: For the Efficient Use of Energy, Water and Hydrogen*. 2003, pp. 28. Available: <https://www.nrcan.gc.ca>
- [2] K. Carrero, M. Piazza, & K. Sposato, *Interstaatliche hochschule für technik buchs heat pump database and search tool implementation*. 2017.
- [3] J. Geldermann, M. Treitz, H. Schollenberger, J. Ludwig & O. Rentz, *Integrated process design for the inter-company plant layout planning of dynamic mass flow networks*. Karlsruhe: Universitätsverlag Karlsruhe, 2007, pp. 421. doi:10.5445/KSP/1000006540
- [4] M. J. Moran & H. N. Shapiro, *Fundamentals of engineering thermodynamics fifth edition*. Wiley, 2006, ch. 3 & 5.
- [5] M. J. Moran & H. N. Shapiro, *Fundamentals of engineering thermodynamics fifth edition*. Wiley, 2006, ch. 6.
- [6] M. J. Moran & H. N. Shapiro, *Fundamentals of engineering thermodynamics fifth edition*. Wiley, 2006, ch. 10.
- [7] Remote maintenance - B&R industrial automation. Available: <https://www.br-automation.com/en/products/software/remote-maintenance/>

Appendices

Appendix A: Charts and Tables

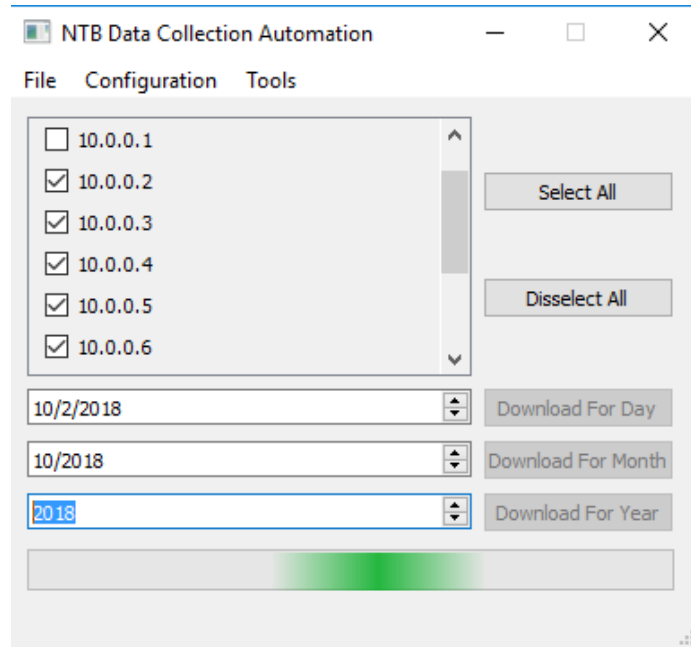
A.1 HostPoint Resource Usage By Approach

Approach	Wall Clock Time	CPU Time	Memory
Single Thread	98.28s	9.83s	167MB
12 Threads	40.10s	10.41s	290MB
24 Threads	47.30s	11.24s	436MB

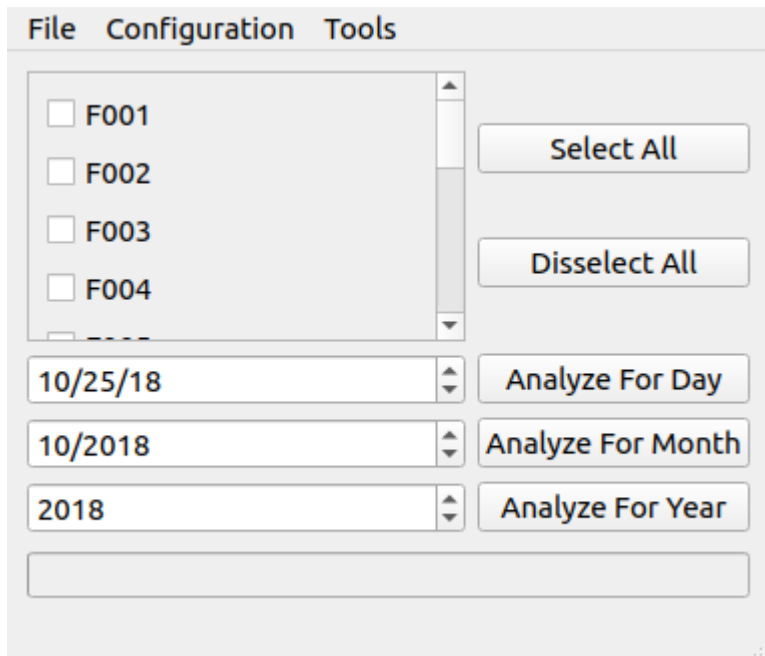
Note: HostPoint's Server Has 24 Cores. CPU_Count/2 and CPU_Count were used as the number of threads

Note2: The files downloaded in this benchmark are an older format that is larger than the files that are being used today, so in real world scenarios it will likely be faster and use less memory.

Appendix B: Screenshots Of The Programs



Screenshot Of File Downloader Program Running



Screenshot Of Heat Pump Analysis Program

Appendix C: Test Bench Instance One Analysis And Results

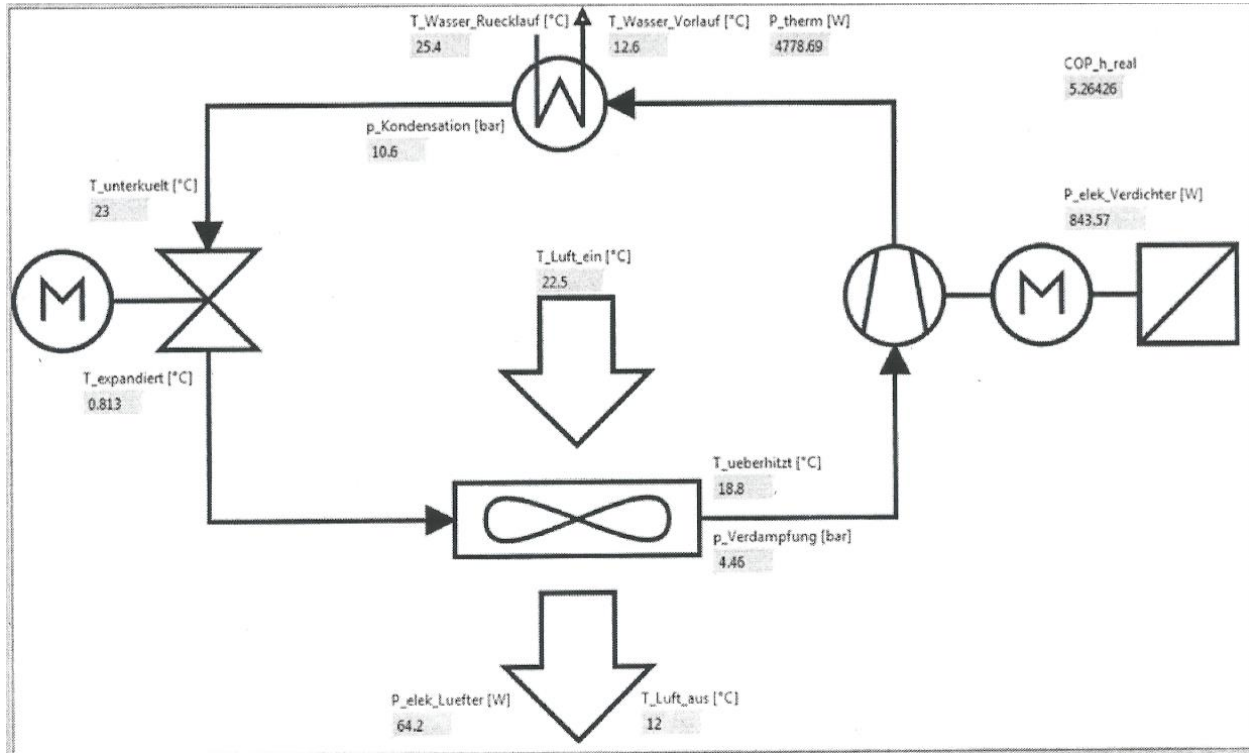


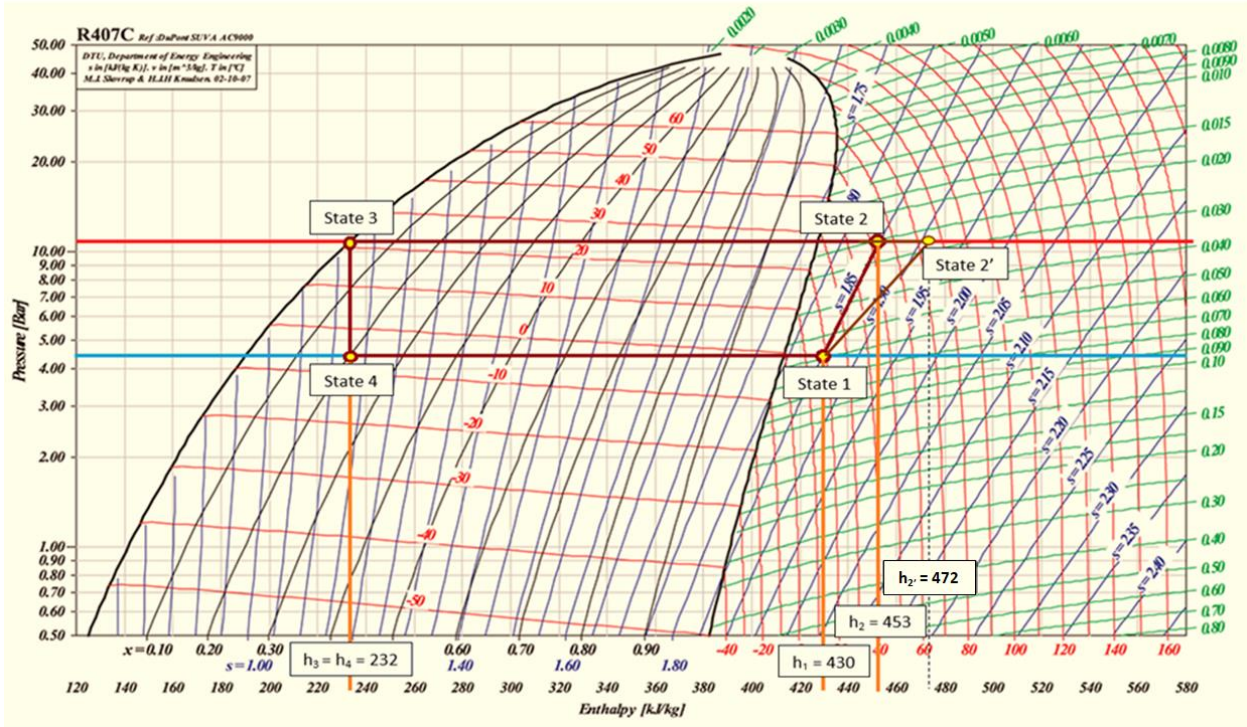
Image Of The LabView Software For Instance One Analysis

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))	Quality
1	18.8	446	429.77	1.8504	1
2	54	1060	453.04	1.8504	1
2'	74	1060	472.23	1.9159	1
3	21	1060	232.02	1.1114	0
4		446	232.02		0.2

Table Of Thermodynamic Properties For Each Of The Four States

Ideal COP	9.50
Actual COP	5.66
$\eta_{\text{compressor}}$	0.55

Calculated Efficiencies



Pressure And Enthalpy Diagram

C.1 Manual Calculations And Interpolations:

Key	
Color Code	Discription
	Property Values Kept Constant
	Variable Property Values
	Interpolated Property Values / Desired Units Conversion
	Desired Property Value / Results
	Values From The Property Table / Measurement Values

Color Key Used For Interpolation And Calculations

Calculating Thermal Energy Exchanged In The Condensor (QH)							
Flow rate of water (l/min)	Flow rate of water (kg/s)	Cold temperature of water (°C)	Cold temperature of water (°K)	Hot temperature of water (°C)	Hot temperature of water (°K)	Cp water (J/(kg*°K))	Thermal Energy Output to water (W)
5.24	0.08908	12.6	285.6	25.4	298.4	4185.5	4772.407552

Thermal Energy Dissipated Calculation

State 2' - Calculating Actual Enthalpy Change				
Thermal Energy Produced (kW)	Total Power Consumed (kW)	Enthalpy At State-1 (kJ/kg)	Enthalpy At State-3 (kJ/kg)	Enthalpy At State-2' (kJ/kg)
4.7724	0.8436	429.77	232.02	472.23

State Two Prime's Enthalpy Calculation

State 1 - linear interpolation			
Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
15	430	426.90	1.8444
	446	426.58	1.8393
	450	426.50	1.8380
20	430	431.10	1.8590
	446	430.78	1.8539
	450	430.70	1.8526
15	446	426.58	1.8393
18.8		429.77	1.8504
20		430.78	1.8539

State One Properties Interpolation

State 2 - linear interpolation			
Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
50.0	1000	449.80	1.8451
51.8		451.51	1.8504
55.0		454.50	1.8596
55.0	1100	453.10	1.8475
56.0		454.05	1.8504
60.0		457.90	1.8620
51.8	1000	451.51	1.8504
54.3	1060	453.04	
56.0	1100	454.05	

State Two Properties Interpolation

State 3 - linear interpolation			
Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
21.0	1052.3	231.60	1.1100
21.3	1060	232.02	1.1114
22.0	1081.8	233.20	1.1153

State Three Properties Interpolation

State 2' - linear interpolation			
Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
75.0	1000	473.60	1.9161
73.6		472.23	1.9122
80.0		478.40	1.9298
75.0	1100	472.40	1.9045
74.8		472.23	1.9184
80.0		477.30	1.9184
73.6	1000	472.23	1.9122
74.3	1060		1.9159
74.8	1100		1.9184

State Two Prime's Properties Interpolation

C.2 Heat Pump Analysis Program Result:

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
1	18.8	446	429.77	1.85
2	54.32	1060	453.04	1.85
2'	74.33	1060	472.23	1.91
3	21.26	1060	232.02	1.11
4	18.8	446	232.02	

State Properties Found By The Program

Thermal Energy (kW)	4.77
Ideal COP	9.5
Actual COP	5.66
$\eta_{\text{compressor}}$	0.55

System Efficiencies Calculated By The Program

Appendix D: Test Bench Instance Two Analysis And Results

T_luft_aus	T_luft_ein	T_Wasser_Vorlauf	T_Wasser_Ruecklauf	T_expandiert	T_ueberhitzt	T_unterkuehlt	p_Kondensation	p_Verdampfung	V_dot_Wasser	P_elek_Verdichter
9.742159	17.71724	14.452439	24.761272	-0.400819	14.758813	22.342449	11.012897	4.576145	6.751492	890.131078

Measurements Collected In Test's Data File For Timestamp 18:11:45

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))	Quality
1	14.8	458	426.18	1.8356	1
2	51	1101	449.33	1.8356	1
2'	71	1101	468.25	1.8904	1
3	23	1101	234.22	1.1187	0
4		458	234.22		0.2

Table Of Thermodynamic Properties For Each Of The Four States

Ideal COP	9.29
Actual COP	5.56
$\eta_{\text{compressor}}$	0.55

Calculated Efficiencies

D.1 Manual Calculations:

Calculating Thermal Energy Exchanged In The Condensor (QH)							
Flow rate of water (l/min)	Flow rate of water (kg/s)	Cold temperature of water (°C)	Cold temperature of water (°K)	Hot temperature of water (°C)	Hot temperature of water (°K)	Cp water (J/(kg*°K))	Thermal Energy Output to water (W)
6.75	0.11475	14.45	287.45	24.76	297.76	4185.5	4951.75

Thermal Energy Dissipated Calculation

State 2' - Calculating Actual Enthalpy Change				
Thermal Energy Produced (kW)	Total Power Consumed (kW)	Enthalpy At State-1 (kJ/kg)	Enthalpy At State-3 (kJ/kg)	Enthalpy At State-2' (kJ/kg)
4.9517	0.8901	426.18	234.22	468.25

State Two Prime's Enthalpy Calculation

D.2 Heat Pump Analysis Program Result:

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
1	14.76	457.61	426.2	1.84
2	51.05	1101.29	449.37	1.84
2'	70.7	1101.29	468.26	1.89
3	22.65	1101.29	234.23	1.12
4	14.76	457.61	234.23	

State Properties Found By The Program

Thermal Energy (kW)	4.95
Ideal COP	9.28
Actual COP	5.56
$\eta_{\text{compressor}}$	0.55

System Efficiencies And Thermal Energy Dissipation Calculated By The Program

Appendix E: Test Bench Instance Three Analysis And Results

T_luft_aus	T_luft_ein	T_Wasser_Vorlauf	T_Wasser_Ruecklauf	T_expandiert	T_ueberhitzt	T_unterkuehlt	p_Kondensation	p_Verdampfung	V_dot_Wasser	P_elek_Verdichter
8.540527	17.33028	14.878718	25.332157	0.926372	14.214607	22.939547	11.16273	4.726079	7.073188	900.937168

Measurements Collected In Test's Data File For Timestamp 19:13:49

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))	Quality
1	14.2	473	425.52	1.8310	1
2	50	1116	448.18	1.8310	1
2'	67	1116	464.89	1.8881	1
3	23	1116	235.01	1.1213	0
4		473	235.01		0.2

Table Of Thermodynamic Properties For Each Of The Four States

Ideal COP	9.41
Actual COP	5.84
$\eta_{\text{compressor}}$	0.58

Calculated Efficiencies

E.1 Manual Calculations:

Calculating Thermal Energy Exchanged In The Condensor (QH)							
Flow rate of water (l/min)	Flow rate of water (kg/s)	Cold temperature of water (°C)	Cold temperature of water (°K)	Hot temperature of water (°C)	Hot temperature of water (°K)	Cp water (J/(kg*°K))	Thermal Energy Output to water (W)
7.073188	0.120244196	14.878718	287.878718	25.332157	298.332157	4185.5	5261.03

Thermal Energy Dissipated Calculation

State 2' - Calculating Actual Enthalpy Change				
Thermal Energy Produced (kW)	Total Power Consumed (kW)	Enthalpy At State-1 (kJ/kg)	Enthalpy At State-3 (kJ/kg)	Enthalpy At State-2' (kJ/kg)
5.2610	0.9009	425.52	235.01	464.89

State Two Prime's Enthalpy Calculation

E.2 Heat Pump Analysis Program Result:

State	Temperature (°C)	Pressure (kPa)	Enthalpy (kJ/kg)	Entropy (kJ/(kg*°K))
1	14.21	472.61	425.54	1.83
2	50.06	1116.27	448.22	1.83
2'	67.46	1116.27	464.91	1.88
3	23.14	1116.27	235.02	1.12
4	14.21	472.61	235.02	

State Properties Found By The Program

Thermal Energy (kW)	5.26
Ideal COP	9.4
Actual COP	5.84
$\eta_{\text{compressor}}$	0.58

System Efficiencies And Thermal Energy Dissipation Calculated By The Program

Appendix F: GitHub Repository Links

F.1 Solution 1:

https://github.com/duffyevan/NTB_Solution1

F.2 Solution 2:

https://github.com/duffyevan/NTB_Solution2

F.3 Analysis Program:

https://github.com/duffyevan/NTB_Analysis