A METHODOLOGY TO DEVELOP A CONTROL SYSTEM APP FOR DATA CENTER COOLING

by

VARUN RAHUL

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

DEC 2016

Acknowledgements

I would like to thank Dr. Dereje Agonafer for making me a part of the EMNSPC family. He has been instrumental in shaping me as a professional and as an engineer.

I would thank Dr. Abdolhossain Haji-Sheikh for being a part of the defense panel and adding insight to developing this project.

I thank Dr. Pavan Nuthi for taking time out from his Post Doc and Research at UTARI, Ft. Worth, TX. I have had the privilege to work under Dr. Pavan Nuthi at UTARI and I owe him a lot in helping me push my limits and grow as an engineer.

This thesis would have been impossible without the support and help of my teammate Anay Darekar. I'm eternally indebted to you for this, my friend.

Special thanks to Ishwariya Srinivasan, Neeraja Gopal, Surendar Kumar Parchur for bearing with me and all my doubts related to LabView. Also thanks to Dr. Betsegaw Gehbrihowt and Abhishek Guhe for their guidance and advice.

I would like to thank Ms. Sally Thompson and Ms. Debi Barton for their constant support and guidance.

Also thanks to my colleagues of Project 4, IAB team and the entire EMNSPC team for their love, guidance and support throughout my journey at University of Texas at Arlington.

And to keep cheering me by saying 'Varun is working on some hot topic!'

Thank you, God.

This is for us Mum and Dad. This is our achievement. Thank you for everything.

November 28, 2016

Abstract

A METHODOLOGY TO DEVELOP A CONTROL SYSTEM APP FOR DATA CENTER COOLING

Varun Rahul, MS

The University of Texas at Arlington, 2016

Supervising Professor: Name Dereje Agonafer

DCIM (Data Center Infrastructure Management), are tools that are available in the industry to serve the purpose of controlling different parameters in the data center. But their cost is significantly high. The idea of developing a tool that is easily accessible at highly competitive costs will be the USP for its development. While developing such a software tool, there will be a few points that will be thoroughly emphasized upon: The software tool will have the flexibility of being used on any Operating System. It will be portable i.e. it can be set up on any device that supports a display and has an interactive screen. There will be proper documentation (setting up blueprints, source files) setup so that it can be modified by the end user to be more suitable based on the data center's facilities such as the DAQ system and the cooling system (for example Indirect/Direct Evaporative Cooling).

The dry bulb temperature (T_{DB}) and relative humidity ($\emptyset$) in the inlet of a server must always be in a recommended zone. As such, it is important to know to know T_{DB} and always to operate the data center servers within the recommended zones. Developing a desktop or mobile based application or tool not only gives the end user control over the cooling method (ex. DEC VS IEC) but also serves to highlight ways in which one can reduce the usage of resources such as infrastructure, capital and energy. The main aim of the tool is to communicate with the cooling system and to ensure that the conditions (Temperature and Relative Humidity) inside the data center facility remain in the recommended zone (ASHRAE recommended zone or allowable region) irrespective of the conditions outside the data center. The proposed control system will have the following sequence of operation: The outside air information enters the control system via a DAQ (Data Acquisition) system for analysis. For a given barometric pressure, any two thermodynamic properties are sufficient to specify the location of the data on a psychrometric chart (ex. Dry Bulb temperature and Wet Bulb temperature). The controller applies the required logic to decide based on the condition of the outside air if it is suitable to be directly sent into the data center or if any form of Evaporative Cooling/ Humidification (EC/H) process is required. Upon performing the necessary calculation to estimate the required amount of cooling, the controller then dictates the cooling units to provide the additional cooling only when necessary based on the design specification (ex. ASHRAE Recommended Zone 1). This tool can help in deciding which cooling systems are best suited based on the location of the data center.

Table of Contents

List of Illustrations

List of Tables

Chapter 1 Introduction

Data Centers and their importance

Data Centers have been the heart of the IT industry. There isn't much about them that hasn't already been said, demonstrated and documented and updated regularly. With the advancements in technology every day, there are continuous efforts in making data centers more advanced in terms of infrastructure and energy utilization. Various organizations have contributed towards such advancements and one such organization taking large strides towards maximizing data center energy efficiency is the American Society for Heating American Society of Heating, Refrigerating and Air-Conditioning Engineers (ASHRAE).

In the age of developing a means to sustain on renewable energy sources man has tried to push the limits of technology in developing green data centers that utilize resources judiciously yet try to achieve maximum output. Data Centers are quite common in today's day and age and most companies' whose basis for operation is data, utilize large spaces to store data that can be accesses around the clock any time of the year.

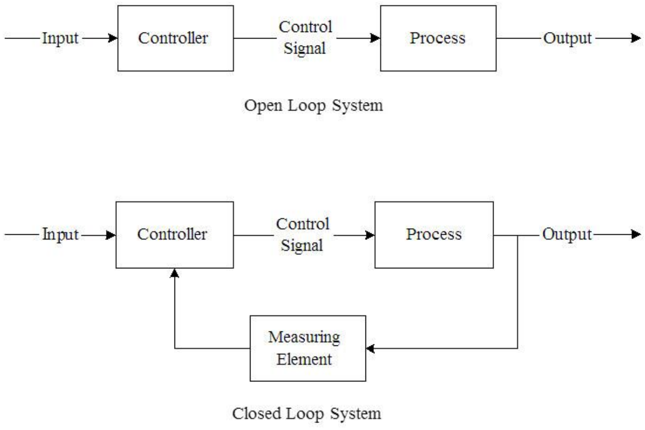
In the past two decades, there has been an exponential increase in the amount of data that has been generated and consumed. IDC [1] in 2011 reported that with that Zettabyte barrier being broken in 2010, with the onset of IoT (Internet of Things), there is an expected nine-fold increase in the next half of this decade. To maintain such huge volumes of data having a data center capable of operating at different loading conditions, throughout the year. The data center must be equipped with systems to support the IT and maintain safe operating conditions and optimum computing performance. An energy efficient data center is of vital importance. This significantly cuts down the cost of cooling the room and is done so by using various techniques such as air side economization (ASE), Indirect evaporative cooling (IEC) and Direct evaporative cooling (DEC) and

Direct Expansion Cooling. The focus of this thesis work is to document the operation of the in the mentioned zones by using Economizer and DEC mode.

Data Center Control Systems and Infrastructure Management Tools

Control System Architecture

A control system is any mechanical/ electronic/ electromechanical system designed and developed to monitor a certain set of variables. A typical control is either Open Loop or Closed Loop. The basic difference between open and close loop control system is the presence of feedback in the closed loop system. The feedback helps in timely updating of values that keep getting passed or cycle through in the system. Most control systems that are commonly used and are semi-autonomous or completely are closed loop control systems. A robust control system is important for the functionality, energy efficiency and reliability of a data center. The control system design for a data center can be dictated based on a performance specification standpoint. Over the past few years the type of control system that has been commonly implemented is called a ‘distributed control’. This type of control system employs a digital control system with a collection of control modules. Each module is a sub system and contains its own specific functions like data acquisition and control. The control of each system is independent and the any user utilizing the control system usually has control over most set points and some aspects of the control system. [2]



Open Loop System

Closed Loop System

Figure 1-1 Open Loop vs Closed Loop Control System [10]

Control system and Mechanical Reliability

Types of mechanical systems

A well designed distributed control system can help achieve both reliability and good energy efficiency. The greatest challenge is the continuous operation of the data center, 24 hours a day, 7 days a week for all 365 days with negligible interruptions. Although earlier redundant systems were installed along with regular systems to facilitate a failsafe, this was in little consideration to the installation and operating costs. With the advancement in technology and developments in the electronics industry, these redundant systems got removes slowly by means of using active and passive monitoring systems that not only kept tabs on the control system but also individually monitor each component's heath and predict its repair or failure. Also in the race to make the data centers more energy efficient or "green", companies began to incorporate economizers that used the outside air to be directly sent into the data center. This air, of course, is filtered and rid of any sort of detrimental material before being sent into the data center. Although this method of economization is cost effective and goes easy on the environment, many companies in the past were reluctant to transition from CRAC units to the Economizer based data center cooling technique. Many data centers hadn't transitioned as the changeover from mechanical cooling to free cooling was deemed risky. [2]

A typical Datacom environment is supported by the following three major types of mechanical systems:

- Central mechanical plant
- Datacom environment equipment spaces
- Support space environment equipment

Central plant usually consists of cooling towers, pumps and chillers. These three components usually account for the maximum energy consumption. An important goal while designing the central plant is to develop an architecture that is highly reliable and has a high speed of response while striving for energy efficiency. [2]

A Datacom environment equipment spaces involve planning and placing the main equipment strategically with the cooling units to achieve maximum cooling efficiency. The cooling required for the Datacom equipment is governed by the ASHRAE recommended envelope as dictated in the *Thermal Guidelines for Data Processing Environments* (ASHRAE 2009) [2]. The systems are designed to operate in the specified cooling bracket. While doing so, utilizing techniques like the hot aisle cold aisle arrangement for stacking the Datacom units and operating the cooling units to achieve a temperature close the dictated one will help in maximum energy savings and efficiency. [2]

Support spaces are general easy going in terms of energy utilization as they usually consist of equipment such as transformers, UPS etc. that require constant cooling and ventilation. These spaces can be kept cooled by using the same air that’s let in to cool the Datacom equipment. [2]

ASHRAE Thermal Guidelines for Data Processing Environment

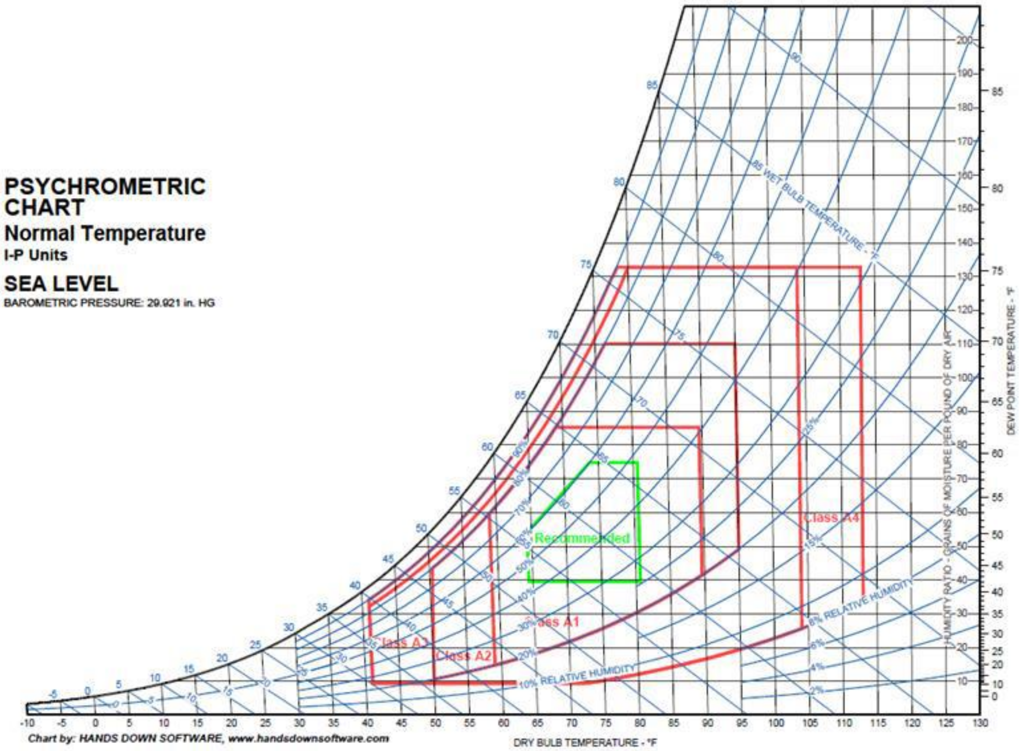


Figure 1 – 2 ASHRAE T.C. 9.9 Allowable regions and Recommended Zones [1]

Under the ASHRAE T.C. (Technical Committee) 9.9. Mission Critical Facilities, Technology Spaces and Electronic Equipment have collaborated with different Datacom equipment companies, research institutes and other governmental and non-governmental organizations to bring about improvements in the data center cooling techniques. Over the years, the agenda for the Technical Committee has shifted from one with emphasis on performance to efficiency and energy savings. [2]

The Thermal Committees' agenda for the latest update was to broaden the operating envelope by widening the temperature and humidity range of operation. Since the year 2008, the temperature and humidity range was widened allowing a larger number of economizers to be used to cool the IT. This further translated in to outside air directly being sent in to the data center without much conditioning, thus calling it “free cooling”. Methods like Air Sided Economizer and Water Sided Economizer are replacing mechanical cooling techniques in a race to be more energy efficient. [1]

Historical Perspective

Computer when introduced were quite large. Their components were of titanic proportions and naturally due to its kind of spread out layout, the power density wasn’t comparable to what it is today. But there were challenges in cooling and maintaining the IT when seen from a facilities and infrastructure standpoint. Computer Room Air Conditioners or CRACs were the first to be introduced in the cooling unit lines. These were meant to dissipate the heat from the devices like computer peripherals, tape drives, cathode ray monitors etc. [2]

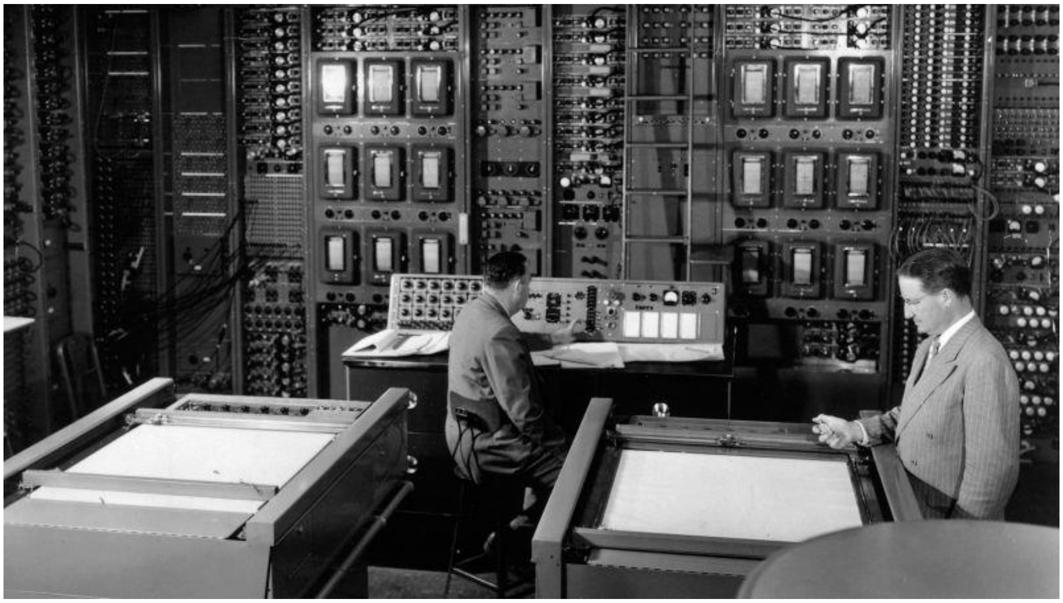


Figure 1-2 Typical Computer room in the 1950s-60s era [9]

With advent of the semiconductor industry, the size of the “computer room” was reduced to a matter of a few inches or a couple of feet. The shift from a macroscopic to a microscopic level of tech allowed the clubbing of many IT modules together and hence helped clubbing units in a more planned way. This also greatly helped in infrastructure planning and development. This however came with a cost. The cost being Power Density. The miniaturization of IT equipment resulted a consequential growth of power densities at all levels of the system.[2]

To put things in perspective, the chart below shows the exponential rise in Heat Load (KW/rack) over the past 20 years alone. ASHRAE has constantly recorded and documented these changes in its journals to define reference points and milestones. [2]

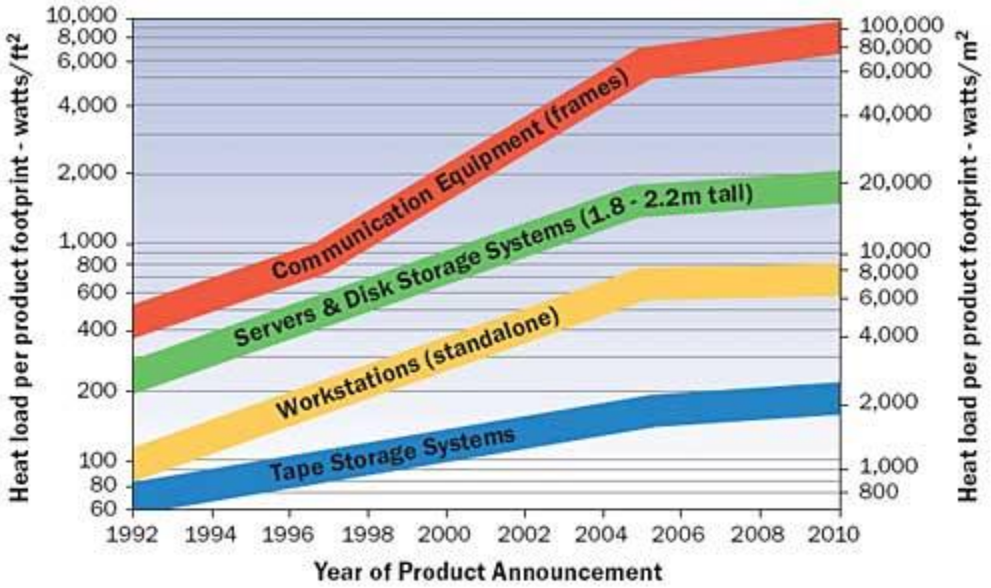


Figure 1-3 ASHRAE power trend chart (ASHRAE 2005c) [2]

In an effort to understand the total power usage at not just a system level by a component level, there was a study conducted by the Lawrence Berkley National Laboratory (LBNL) in 2001, where in 22 Data Centers were studied and their energy usage was analyzed (2007c), and a variation in power utilization ratio was concluded. The ratio was between amount of power consumed by IT versus amount of power provided to the entire facility. The ratio of 33% and 75% was commonly observed, the remaining percentage power used by auxiliary equipment such as lighting, UPS (Uninterrupted Power Supply) etc. The following figure shows a distribution of the power among the equipment housed in a data center [2]

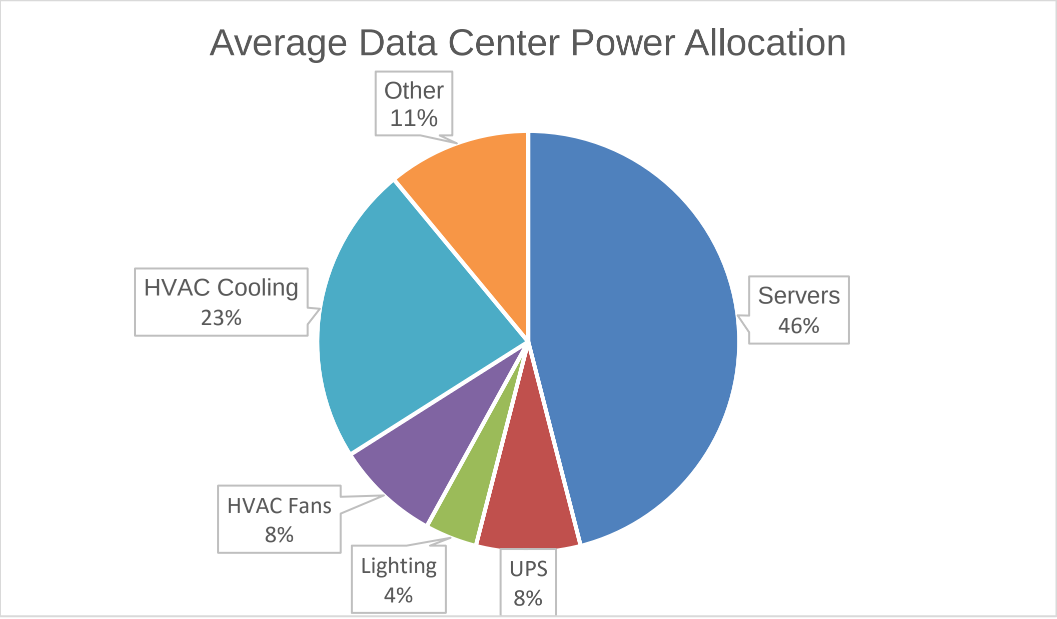


Figure 1- Energy consumption of HVAC controls and energy management (LBNL2007a) [2]

Chapter 2 Psychrometric Properties

Psychrometric properties are the governing parameter when trying to understand, analyze, design or modify the cooling system of a data center.

Dry Bulb Temperature ($T_{\mathrm{d}b}$): It is the temperature that is measured by a thermometer that is simply kept in the ambient. There are no external factors affecting its readings. It is expressed in three units mostly, °F(Fahrenheit), °C (Celsius) and K (Kelvin). On a psychrometric Chart, Dry Bulb Temperature is always found on the horizontal axis.

Wet Bulb Temperature (T_{wb}): The Wet Bulb Temperature unlike the Dry Bulb Temperature also accounts for the moisture content of the air. It represents the lowest temperature that can be achieved in the ambient. It is represented as parallel 45° lines on the psychrometric chart. It is expressed in three units mostly, °F(Fahrenheit), °C (Celsius) and K (Kelvin).

Dew Point Temperature (T_{dp}): The Dew Point Temperature is theoretically the lowest possible that can be achieved before condensation begins. It is represented as horizontal lines originating from the vertical axis on the right. It is expressed in three units mostly, °F(Fahrenheit), °C (Celsius) and K (Kelvin).

Relative Humidity (R.H.): It represents the ratio of the amount of moisture present in the air to the maximum amount of vapor that can be present in air at that given temperature. It is represented in terms of percentage (%) and can be read on the psychrometric chart as the curved lines from 10% to 100% (saturation).

Specific Humidity: It is defined as the ratio of the mass of water vapor in air to the total mass of air. It is represented as percentage (%) and represented by horizontal lines just like Dew Point Temperature.

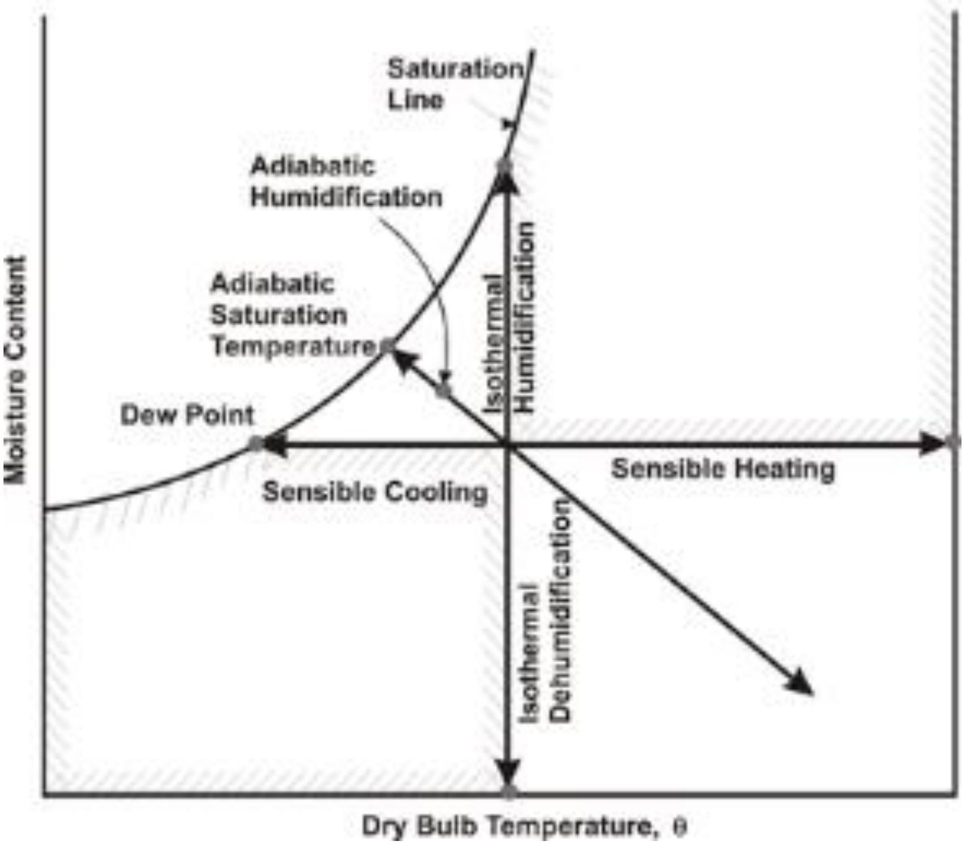


Figure 2-1 Types of thermodynamics processes plotted on a Psychrometric Chart [11]

Allowable Region and Recommended Zones

A Brief Intro to Environmental Classes

Under the Thermal Guidelines for Data Processing Environment (ASHRAE 2009), a committee formulated and updated rules on the different classification of regions which were consider suitable to operate the cooling units, thus dictating the equipment manufacturers the design criteria while developing such data center cooling units. There are four main classes (1-4), and an additional class called the NEBS (Network Equipment Building Systems) that is used to typically used to characterize the Datacom environment [1]:

Class 1: This class consists of rigorously controlled environment and parameters such as Dry bulb temperature, Relative Humidity etc. and the kind of equipment that operate in this zone are typically the type one would find for enterprise servers or storage products.

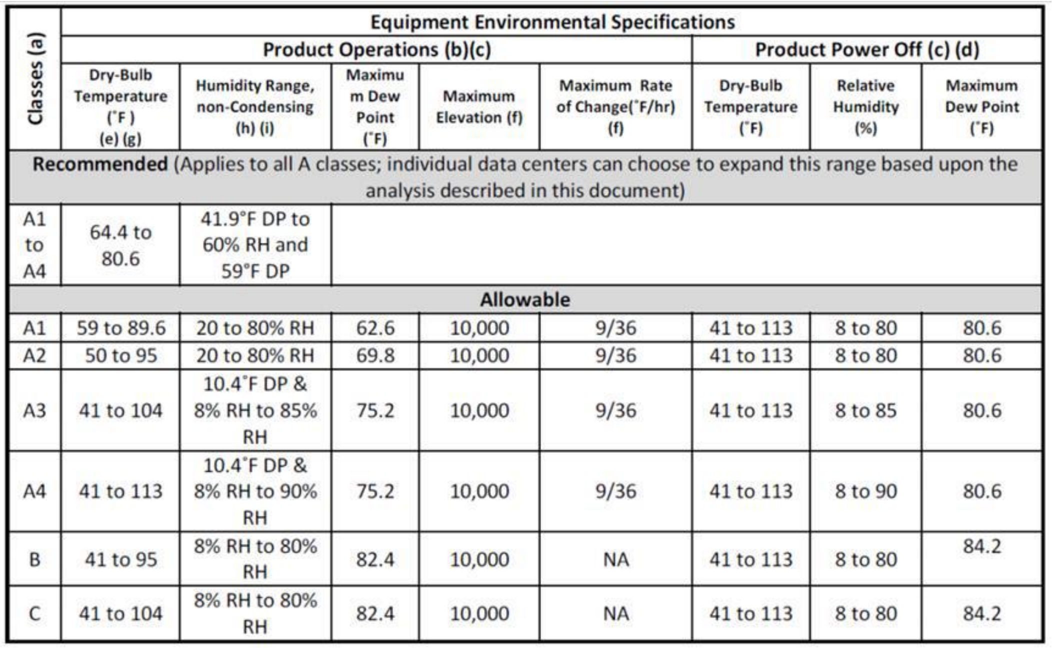
Class 2: This zone typically houses work stations, PCs, Office equipment and hence the equipment used are designed for a small environment to have control over relative humidity, dew point and dry bulb.

NEBS: Products typically designed to operate in this region are switches, routers, small or medium transport equipment. There is control over relative humidity, temperature and dew point (Telecordia 2001, 2002).

Regions 3 & 4: These regions aren’t designed for Datacom equipment. [2]

Envelope Considerations

Table 2-1 ASHRAE 2011 Thermal Guideline Classes [2]



Classes (a)	Equipment Environmental Specifications							
	Product Operations (b)(c)					Product Power Off (c) (d)		
	Dry-Bulb Temperature (˚F) (e) (g)	Humidity Range, non-Condensing (h) (i)	Maximu m Dew Point (˚F)	Maximum Elevation (f)	Maximum Rate of Change(˚F/hr) (f)	Dry-Bulb Temperature (˚F)	Relative Humidity (%)	Maximum Dew Point (˚F)
Recommended (Applies to all A classes; individual data centers can choose to expand this range based upon the analysis described in this document)								
A1 to A4	64.4 to 80.6	41.9°F DP to 60% RH and 59°F DP						
Allowable								
A1	59 to 89.6	20 to 80% RH	62.6	10,000	9/36	41 to 113	8 to 80	80.6
A2	50 to 95	20 to 80% RH	69.8	10,000	9/36	41 to 113	8 to 80	80.6
A3	41 to 104	10.4˚F DP & 8% RH to 85% RH	75.2	10,000	9/36	41 to 113	8 to 85	80.6
A4	41 to 113	10.4˚F DP & 8% RH to 90% RH	75.2	10,000	9/36	41 to 113	8 to 90	80.6
B	41 to 95	8% RH to 80% RH	82.4	10,000	NA	41 to 113	8 to 80	84.2
C	41 to 104	8% RH to 80% RH	82.4	10,000	NA	41 to 113	8 to 80	84.2

Cooling Systems

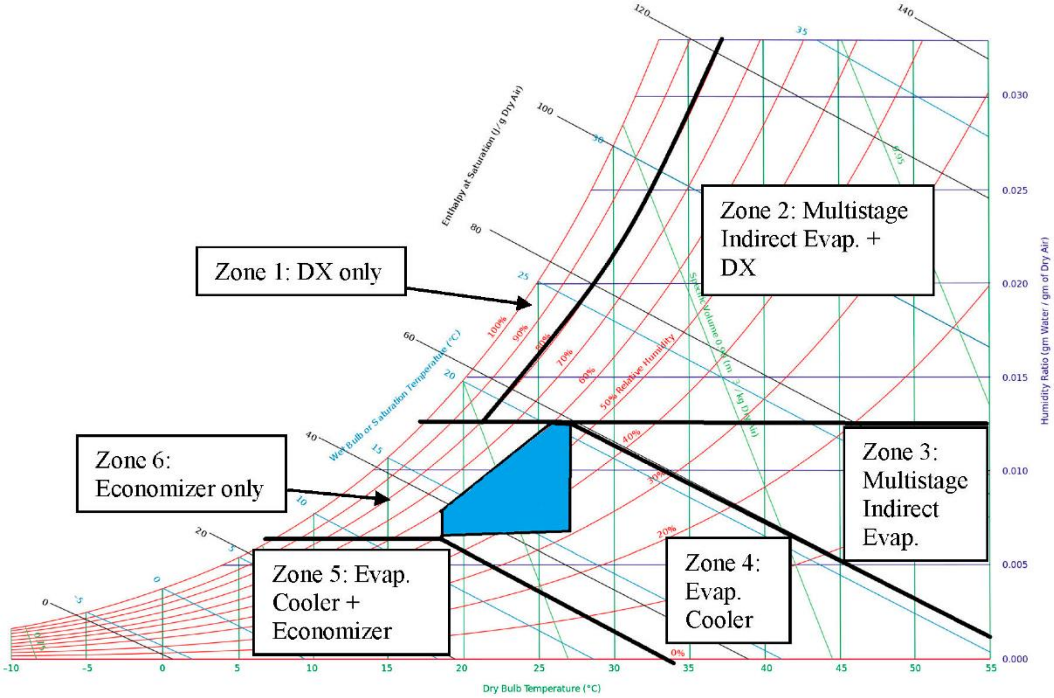


Figure 2-2 Different Zones in Psychrometric Chart [12]

Cooling systems are an integral part of the data center infrastructure. The cooling system is the heart of the data center infrastructure as they operate around the clock throughout the year irrespective of the outside air conditions to keep the dry bulb temperature and relative humidity inside the data center within the recommended zone. The cooling systems chosen to operate in a data center unit and keep it cool are selected based on different psychrometric conditions.

The commonly used cooling systems for data centers are:

Economizer

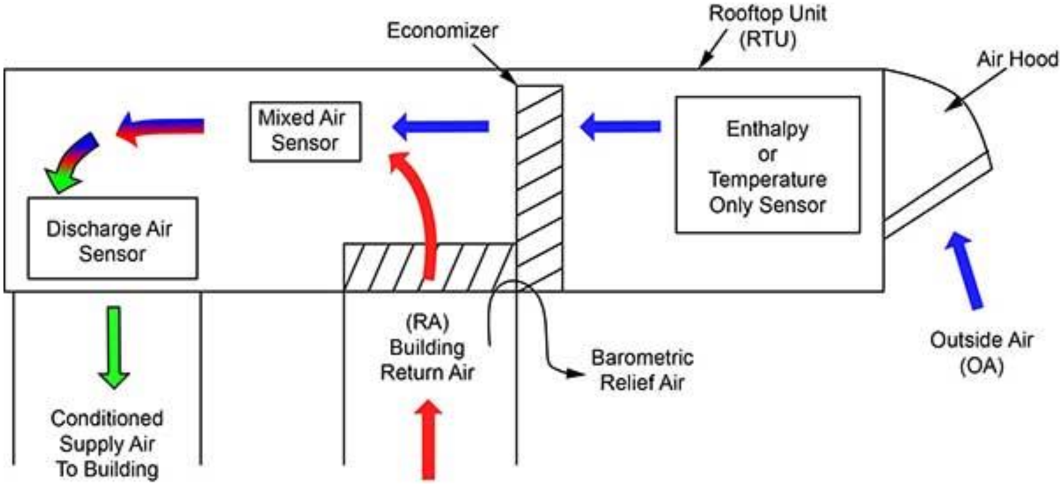


Figure 2-3 Economizer operation in an air handing unit [13]

Dubbed as ‘free cooling’, this cooling system involves inducting the outside air as is, into the data center without any conditioning or change in its temperature and humidity. The air, is subjected to thorough filtration as this outside air contains all sorts of impurities, dust, grass, dirt and other particulate matter that can contaminate the otherwise hermetically maintained Datacom environment. Economizer are also further divided into 2 parts. [2]

Air Sided Economizers

This kind of system uses outside air either fully or partial for cooling. The system inducts outside and if required mixes it with the return air by modulating the return air dampers. This is configuration is best suited for centralized air handlers. The dampers used to control the inlet and return air can be actuated based on active monitoring of either dry bub of inlet and supply or dry bulb and relative humidity (enthalpy economizers) of the inlet and supply. If necessary dehumidification may also need to be carried out. As such it is recommended to perform enthalpy based economization as it gives a better control over the actual Datacom conditions.[2]

Water Sided Economizer

Water side economizers involves water being cooled by the means of a cooling tower and doesn't use any form of mechanical cooling. Water sided economizers are of two types: direct and indirect. The basic difference between the two being, the method of heat transfer to cool the water. [2]

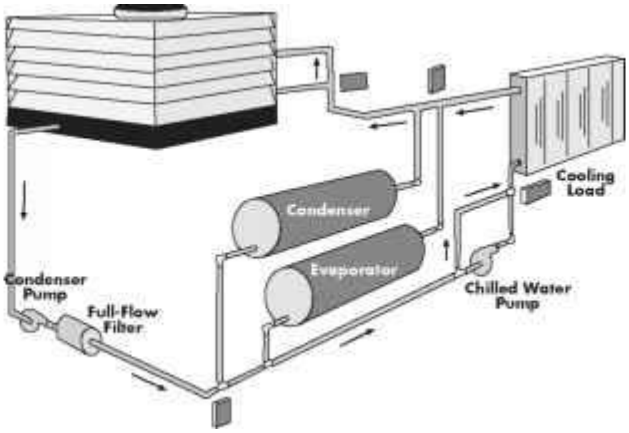


Figure 2-4 Water Sided Economizer Component and Flow Directions [14]

Return Air Cooling

In this method of air conditioning, the warm air coming from the data center is mixed with the outside air. This is done if the temperature and relative humidity of the air is not in the recommended zone and its temperature and humidity value needs to be increased. The amount if air to be mixed is metered by means of sensors and the DCIM control system.

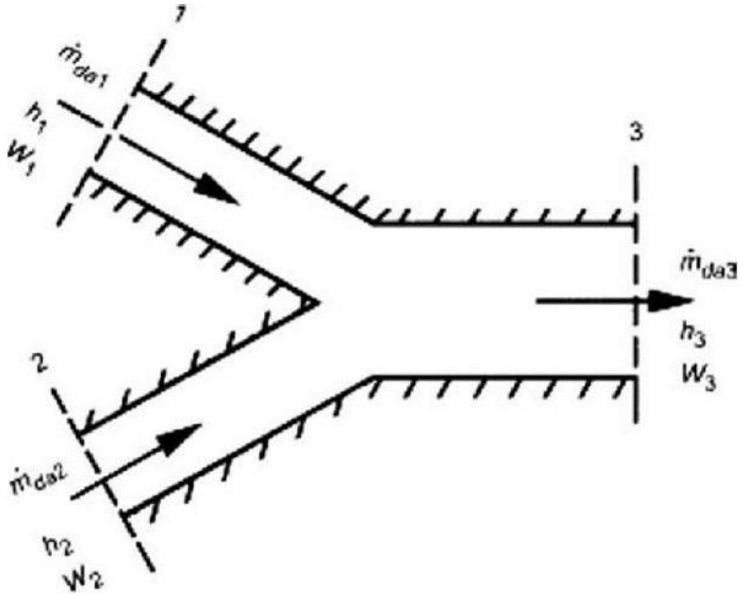


Figure 2-5 Mixing of Return Air [15]

Direct Evaporative Cooling

This is one of the most widely used cooling methods just after DX cooling. This form of cooling is typically used in places of high dry bulb temperature and low relative humidity. For example, arid climatic zones. In this method of air conditioning,

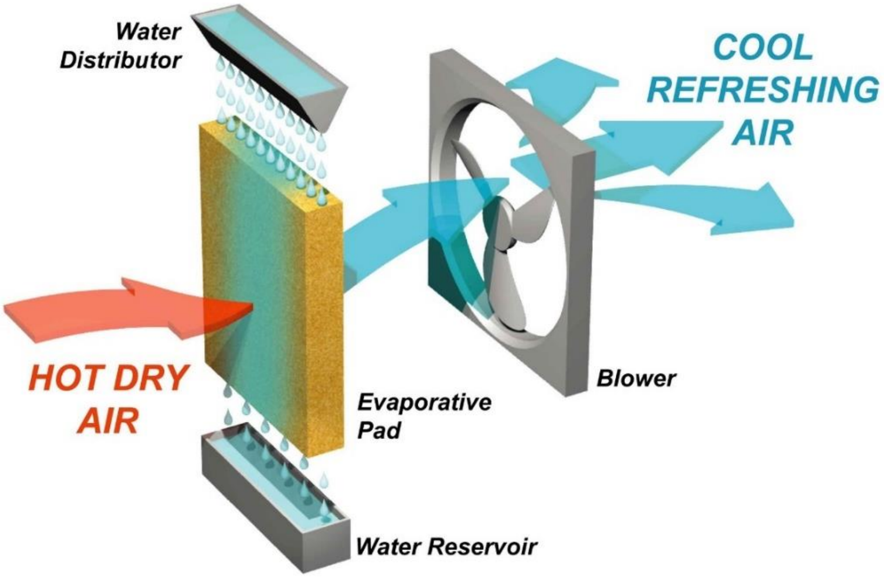


Figure 2-6 Direct Evaporative Cooling setup [16]

the air is subjected to atomized water droplets that mix with it and reduce the dry bulb temperature and increase the relative humidity as required. Once the air is in the recommended zone and fit to be sent into the data center for cooling. The warm and dry outside air is inducted into the cooling unit and then made to pass through a porous media. The porous media is made out Cellulose or glass and the commonly used ones are Celdek and Glassdek. Water is made to flow through the cooling media from top and collected in a reservoir at the bottom of the media. The air passes through the media and comes in contact with the atomized water. The temperature of the water drops and humidity increases. The fan at the end of the unit pulls the air in and felicitates the mixing of the air streams.

Chapter 3 Methodology to Develop the App

The following chapters are broken into sub sections. As we progress, there will details on the methodology adopted to develop the control system and the app.

Groundwork

The project began with understanding the context of the topic and its relevance in the industry. Since DCIM (Data Center Infrastructure Management) tools already existed in the industry, the primary aim was to understand its working and analyze the shortcomings and work on it. The points that almost stood out and urged the team to work on this project were:

1. The cost of the existing software
2. The lack of a modular system approach.

1. Cost of the existing software

Most of the DCIM software available in the market are proprietary in nature, i.e. they are a registered and licensed product of an organization and hence the cost associated with it is also high. Hence one of the objectives of this project was to reduce that cost and work in the availability of such tools. By making them open source and easily available, there are chances that such a tool can have much better updates and reach a much wider audience.

2. Lack of a modular system approach

Among the different software that the team had analyzed there was an observation that the DCIM tools were designed to operate on very specific type of systems. They didn’t have any means of including or removing a system when needed. A sophisticated plug and play of sorts.

The following sections deal with the work flow sequence, on how the project was developed over different phases. Each phase involved developing and testing the app by means of different DAQ system. This chapter discusses all the systems in detail. Each DAQ system is discussed in detail and the process of acquisition to processing and displaying is discussed.

Data Acquisition The process as the name suggest involves collecting or acquiring data. The data for this process was the outside air information. The data collected would then proceed to be analyzed. A few configurations for a DAQ (Data Acquisition System) were developed or predeveloped and used.

Agilent 34972A

Instrument information and setup

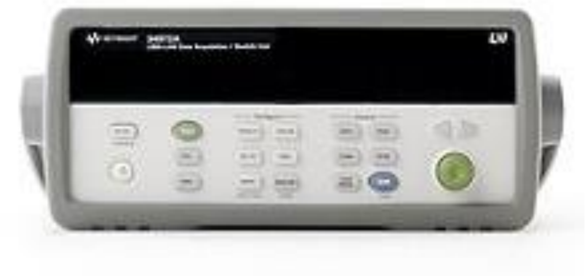


Figure 3-1 Agilent 34972A LXI Front [17]

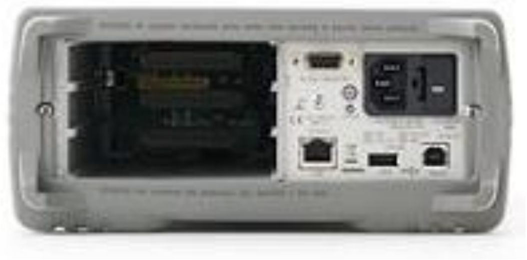


Figure 3-2 Agilent 34972A LXI Back [18]

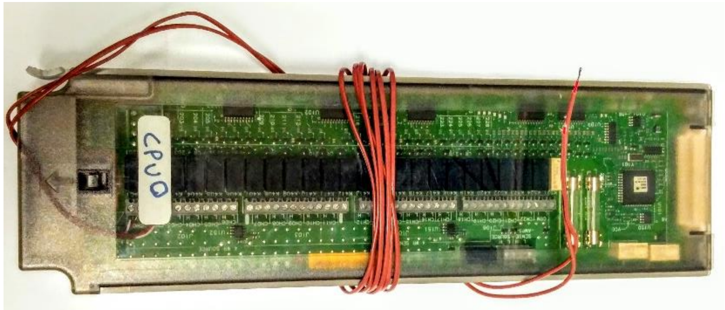


Figure 3-3 Agilent 34901A and Thermocouples

The Agilent 34972A LXI Data Acquisition/ Data Logger Switch Unit was the main DAQ system used. LXI stands for LAN eXtensions Instrument. Following are the specifications of the instrument used [Web Address]:

1. 3-slot LXI data acquisition unit with 6 ½-digit DMM (22-bit) and 8 plug-in modules to choose from.
2. Measures 11 different input signals including temperature with thermocouples, RTDs and thermistors; DC/AC volts or current; 2- or 4-wire resistance; frequency and period. Accepts all 34970A switch and control plug-in modules, and is backward compatible with the 34970A SCPI command set
    - 1Gbit LAN and USB 2.0 for easy connectivity to the PC
    - USB memory port for data storage or transfer
    - Graphical Web interface for easy set up and control
    - Free Agilent BenchLink Data Logger 3 software to create tests without programming [3]

In either of the 3 slots provided a 20 channel, 34901A multiplexer can be inserted. The multiplexer can switch between 60 channels per sec and address a broad spectrum of data acquisition problems [4]. The 34972 A has in built settings to show temperature readings of various temperature measurement devices such as Thermocouple, Thermistor, 2 wire Ohms and 4 wire Ohms RTD. Under thermocouple options, the DAQ unit can support J, K, E, T type of thermocouple. For this study, a J type thermocouple as it has an operating range of 32°F to 1382°F (0 °C – 750 °C) which was sufficient for the experiment. There wasn’t a need for other thermocouples that are developed and calibrated to operate in extreme temperatures. [4]

Following steps outline how the DAQ system was set up and used to record the readings.

1. Two thermocouples were connected to one channel each in the 34901A multiplexer and inserted in to the Agilent 34972A LXI DAQ system.

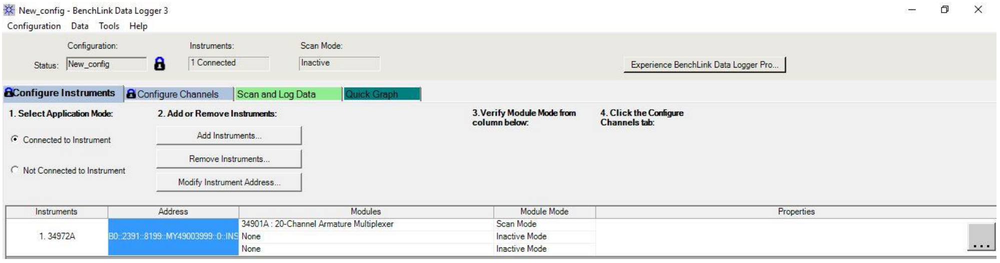
2. Ensure the DAQ is powered on and set the DAQ to measure temperature using the job dial on its display and choose Thermocouple. Choose J type under that.

3. Display the readings on the monitor by pressing the ‘view’ button.

4. Now to view the readings on the Computer/PC, download the Keysight BenchVue and Agilent BenchLink Data Logger 3 software from Keysight.com. This software(s) assist in interfacing the DAQ device via the Computer/PC.

5. Once the software(s) are loaded and opened, select the device from the menu and choose the channel numbers to which the J-type Thermocouple shave been connected in the 34901A Multiplexer. Change DC to °C/°F based on the desired unit for measurement.

6. If the steps have been followed correctly, a live display of the readings should be available. The frequency in which the readings are taken in and recorded can be set to the required interval.



Configuration:
Status: New_config
Instruments: 1 Connected
Scan Mode: Inactive

1. Select Application Mode:

● Connected to Instrument

○ Not Connected to Instrument

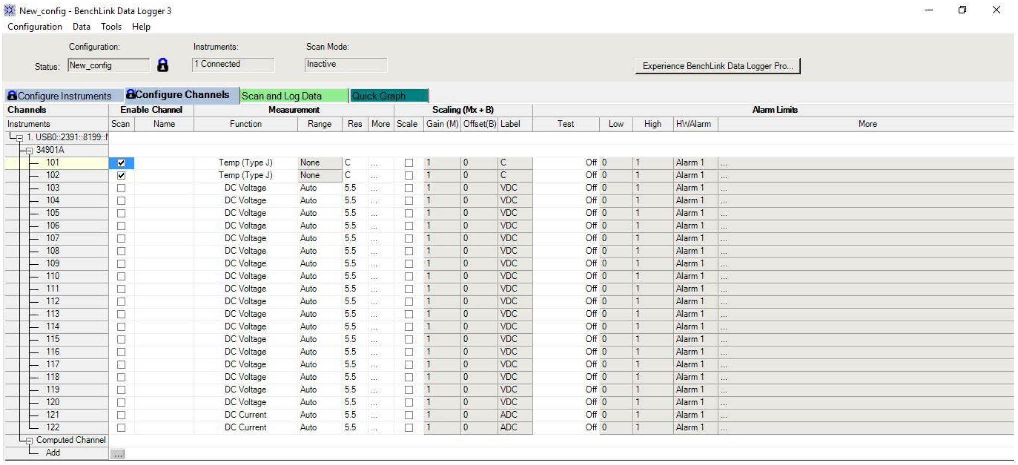
2. Add or Remove Instruments:

3. Verify Module Mode from column below:

4. Click the Configure Channels tab:

Instruments	Address	Modules	Module Mode	Properties
1. 34972A	B0::2391::8199::MY49003999::0::INS	34901A : 20-Channel Armature Multiplexer	Scan Mode	
		None	Inactive Mode	
		None	Inactive Mode	

Figure 3-4 BenchVue Data Logger Interface to Configure the instrument


New_config - BenchLink Data Logger 3
Configuration Data Tools Help
Experience BenchLink Data Logger Pro...
Configure Instruments
Configure Channels
Scan and Log Data
Quick Graph

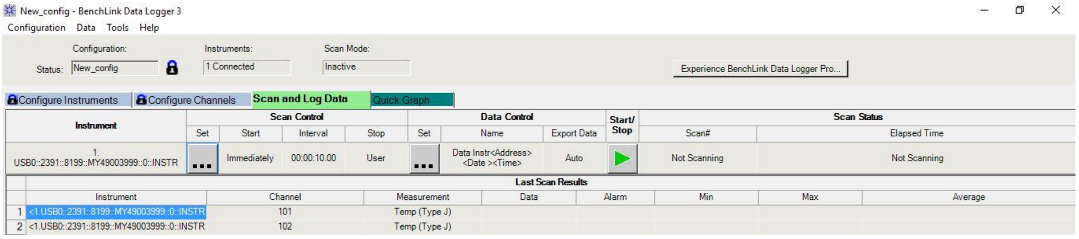
Configuration: Status: New_config

Instruments: 1 Connected

Scan Mode: Inactive

Channels	Enable Channel		Measurement				Scaling (Mx + B)				Alarm Limits				
Instruments	Scan	Name	Function	Range	Res	More	Scale	Gain (M)	Offset(B)	Label	Test	Low	High	HWAlarm	More
1. USB0::2391::8199::															
34901A															
101	☑		Temp (Type J)	None	C	...	□	1	0	C	Off	0	1	Alarm 1	...
102	☑		Temp (Type J)	None	C	...	□	1	0	C	Off	0	1	Alarm 1	...
103	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
104	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
105	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
106	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
107	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
108	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
109	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
110	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
111	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
112	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
113	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
114	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
115	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
116	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
117	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
118	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
119	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
120	□		DC Voltage	Auto	5.5	...	□	1	0	VDC	Off	0	1	Alarm 1	...
121	□		DC Current	Auto	5.5	...	□	1	0	ADC	Off	0	1	Alarm 1	...
122	□		DC Current	Auto	5.5	...	□	1	0	ADC	Off	0	1	Alarm 1	...
Computed Channel															
Add	...														

Figure 3-5 Section to Configure the 34901A multiplexer channels


New_config - BenchLink Data Logger 3
Configuration Data Tools Help
Experience BenchLink Data Logger Pro...
Configure Instruments
Configure Channels
Scan and Log Data
Quick Graph

Configuration: Status: New_config

Instruments: 1 Connected

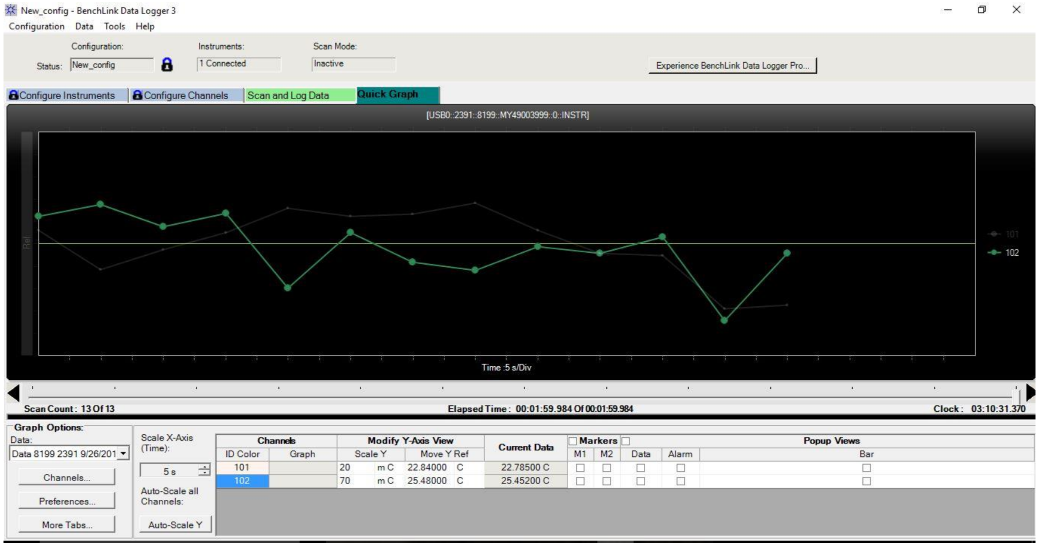
Scan Mode: Inactive

Instrument	Scan Control				Data Control			Start/ Stop	Scan Status	
	Set	Start	Interval	Stop	Set	Name	Export Data		Scan#	Elapsed Time
1. USB0::2391::8199::MY49003999::0::INSTR	...	Immediately	00:00:10.00	User	...	Data Instr<Address> <Date ><Time>	Auto	▶	Not Scanning	Not Scanning

Last Scan Results

	Instrument	Channel	Measurement	Data	Alarm	Min	Max	Average
1	<1.USB0::2391::8199::MY49003999::0::INSTR	101	Temp (Type J)					
2	<1.USB0::2391::8199::MY49003999::0::INSTR	102	Temp (Type J)					

Figure 3-6 Section to Scan and Log Data



Configuration: Status: New_config

Instruments: 1 Connected

Scan Mode: Inactive

Scan Count : 13 Of 13

Elapsed Time : 00:01:59.984 Of 00:01:59.984

Clock : 03:10:31.370

Data: Data 8199 2391 9/26/201

Scale X-Axis (Time): 5 s

Auto-Scale all Channels:

Channels		Modify Y-Axis View				Current Data	□ Markers □				Popup Views
ID Color	Graph	Scale Y		Move Y Ref			M1	M2	Data	Alarm	Bar
101		20	m C	22.84000	C	22.78500 C	□	□	□	□	□
102		70	m C	25.48000	C	25.45200 C	□	□	□	□	□

Figure 3-7 Live display of readings

For the experiment the values of Relative Humidity and Dry Bulb Temperature were chosen to represent the outside air information but since the DAQ had no direct way to measure RH, one of the thermocouples was modified to measure the Wet Bulb Temperature. To convert a regular thermocouple into a Wet Bulb Temperature measuring thermocouple the following modifications were made:

1. In a cup/glass water was filled till ¼ of the cup's volume.

2. One end of a single piece fabric was inundated in the water. The other end was wrapped around the thermocouple.

3. Due to capillary action the entire cloth should be wet in a short while (based on the thickness of the fabric).

4. Close the cup with a lid and make sure the cloth and thermocouple part is exposed to the ambient.

5. If the apparatus is setup correctly, the DAQ should now read the Wet Bulb Temperature of the surrounding.

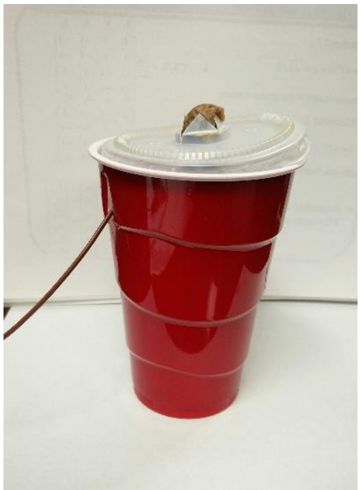


Figure 3-8 Wet Bulb Temperature Thermocouple apparatus setup

Interfacing via LabView

Accessing and displaying the data

To develop the control system app on LabView, the readings of the thermocouple need to be brought into LabView. This involves connecting the device and bringing the readings in a specific order. To do the driver library for 34972A and 34901A needs to be downloaded and installed. The steps for this is discussed in Appendix A.

The order to be followed to pull values from the Agilent 34972A DAQ into LabView is as follows:

Initialize→ Reset→ Configure Thermocouple→ Temperature Auto Zero→ Configure Scan→ Measure Thermocouple→ List to Array→ Close

This workflow was adopted to access the channels and to tap into the data being recorded, by correctly specifying the channel numbers and thermocouple type.

Defining this methodology took a while and adequate documentation is done in this report to establish the method to setup and use the apparatus.

Note

The author would like to add that setting up the equipment can be time consuming and sometimes even the most trivial of errors can cause unforeseeable delays in the project and hence requests anyone who uses this device to dedicate a small portion of their project report as a setup guide and troubleshooting section. This way proper setup procedure and debugging (if any) can be carried out with relative ease. A similar note in their thesis is also encouraged to carry forward this courtesy.

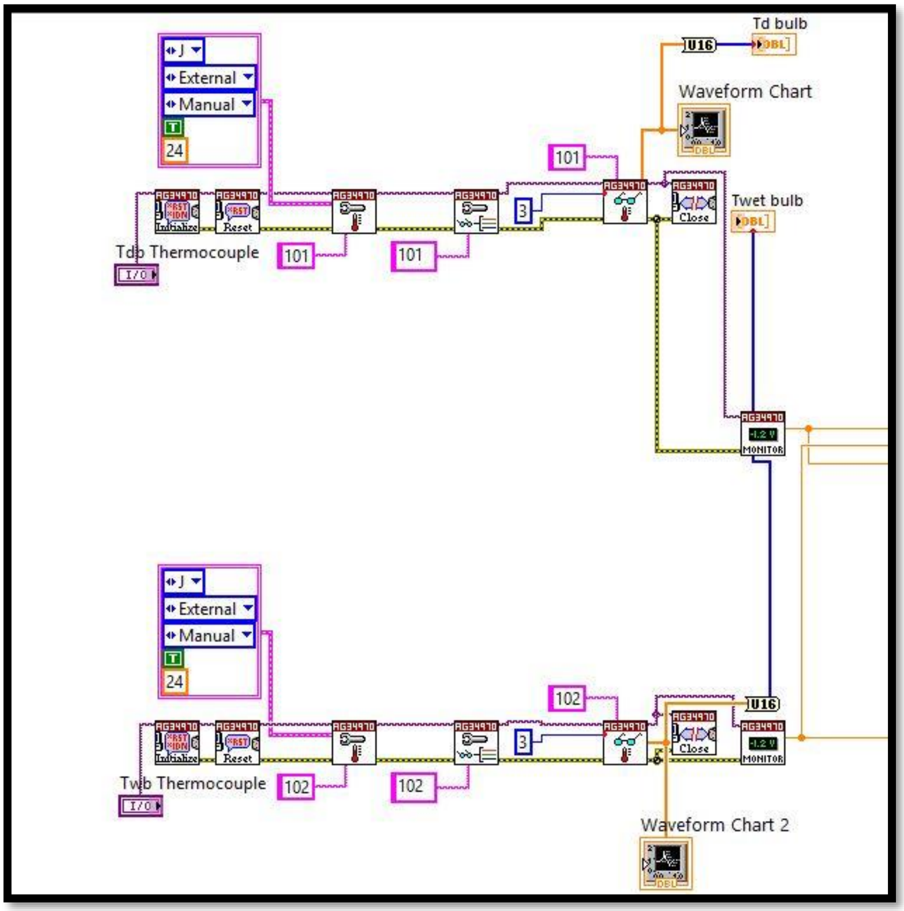


Figure 3-9 Data Acquisition loop using Agilent 34972A

Once the data was accessed and brought into LabView, it was displayed on the main screen by the means of a text indicator and a graph to show the variation of the temperature over time.

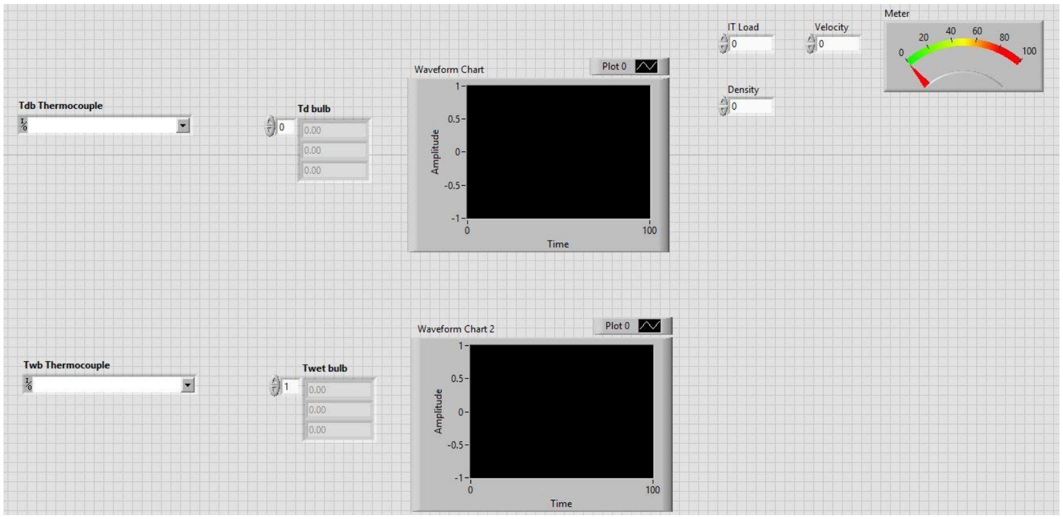


Figure 3-10 The front of the Agilent 34972A based DAQ system.

Once the data has been received by LabView from the Agilent 34972A unit, the Dry and Wet Bulb Temperature values and then sent into the MATLAB structure. The MATLAB structure is constructed based on the following criteria:

Region 0 (< 52°F DB and < 41.9°F DP): When outside air conditions lie within this region, the target supply air dry bulb temperature is 65°F. The outside and return air dampers modulate to mix both airstreams. If required, the EC/H system stages on to provide the necessary humidification for maintaining wet bulb temperature of the supply air at 54°F and the dew point temperature at 42°F. [1]

Region 1 (> 52°F DB and < 41.9°F DP): This region calls for 100% outside air. The return air dampers are completely closed and the outside air dampers are fully open. EC/H stages on to provide the required humidification or cooling. The supply air dry bulb temperature is maintained between 65° F and 80°F while the dew point temperature is maintained at 42°F. [1]

Region 2 (> 65°F DB and > 41.9°F DP and <80°FDB and < 59.0°FDP and < 65%RH): In this region, too the return air dampers are completely closed and the outside air dampers are fully open. 100% outside air is admitted. The EC/H system remains off, since no evaporative cooling or humidification is required. The outside air is delivered into the data hall "as is" (after filtration). [1]

Region 3 (> 80°F DB and > 41.9°F DP and <65. 76°FW B): The economizer is at 100% in this region as well, meaning that outside air is not mixed with return air. EC/H stages on to provide required humidification or cooling. The supply air dry bulb temperature is maintained at 80°F while dew point temperature is maintained between 42°F and 59°F. [1]

Region 4 (> 80°F DB and > 41. 9°F DP and > 65. 76°FW B): Once more, the dampers modulate to bring in 100% outside air. EC/H stages on to provide the required humidification or cooling. The supply air dry bulb temperature is maintained at 80°F while dew point temperature is kept above 59°F. [1]

Region 5 (80°FDB and > 59.0°FDP and > 65.76°FWB): In this region, the dampers modulate to mix outside air with return air to increase cold aisle temperature as necessary for reducing cold aisle relative humidity to a 65% maximum. The supply air temperature is maintained between 65°F and 80°F. The dew point temperature is kept above 59° F. The direct evaporation system is bypassed, since no evaporative cooling or humidification is required. [1]

Region 6 (> 65°F DB and < 59.0°F DP and> 65 and < 63°FWB or > 65°FDB and> 41.9°FDP and> 65%RH and < 59.0°FDP or < 65.76°F D.B. Again, the dampers modulate to mix outside air with return air to increase cold aisle temperature as necessary for reducing cold aisle relative humidity to a 65% maximum. The supply air temperature is maintained above 65°F. and the dew point temperature is kept below 59°F. The direct evaporation system is bypassed, since no evaporative cooling or humidification is required. [1]

Region 7 (Unacceptable OA conditions): When outside air is inadmissible to the datacenter (such as excessive smoke or dust particulates in the air), the external dampers are shut. [1]

The MATLAB code returns a value between 0 & 7 and as per the number a subsequent case structure is triggered. There are 4 cases possible in NiyanTron:

1. Economizer
2. Direct Evaporative Cooling
3. Return Air Mixing
4. Rejection of outside air

As the Sub VI is triggered based on the associated conditioning operation, an interface gets displayed on the screen and the file proceeds to be saved in the specified directory.

NOTE

The last part was not achieved for the Agilent 34972A DAQ model as an alternative DAQ was formed using the Arduino 2650 MEGA board and the DHT 11 Temperature and Humidity sensor.

Arduino based DAQ Model

Device information and setup

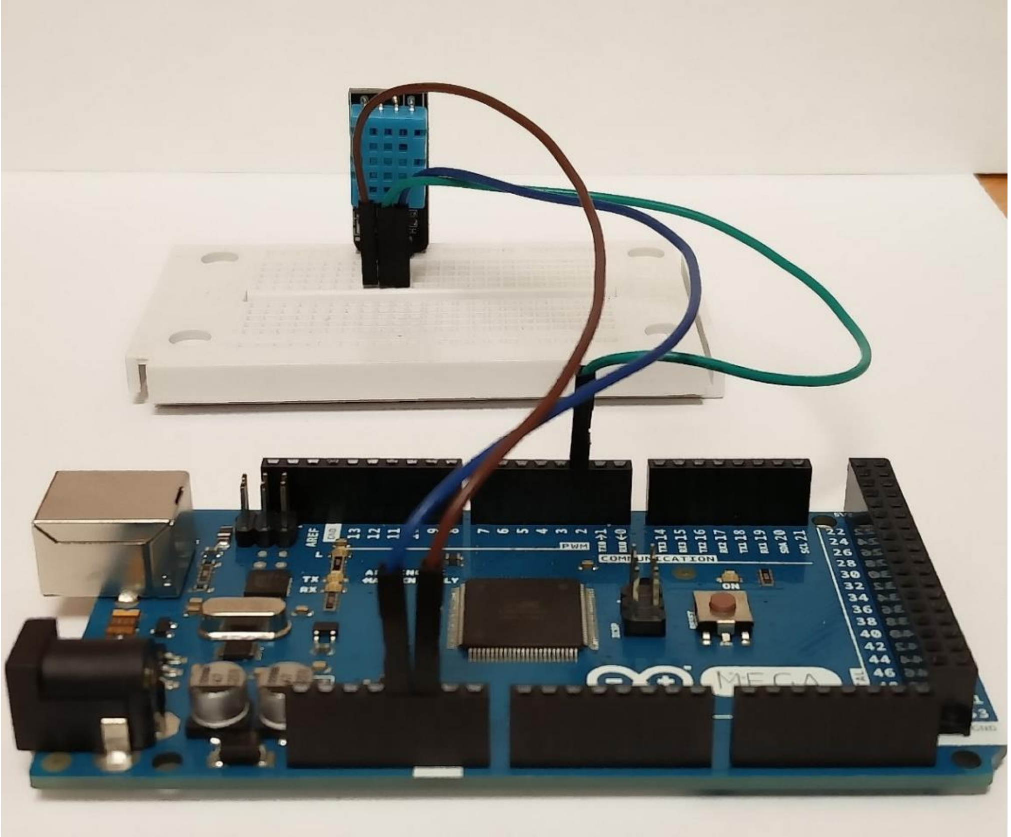


Figure 3-11 DHT 11 Temperature and Humidity sensor and Arduino MEGA 2650 (top to bottom)

Arduino MEGA 2650

Table 3-1 Technical Specifications of Arduino MEGA 2650 [5]

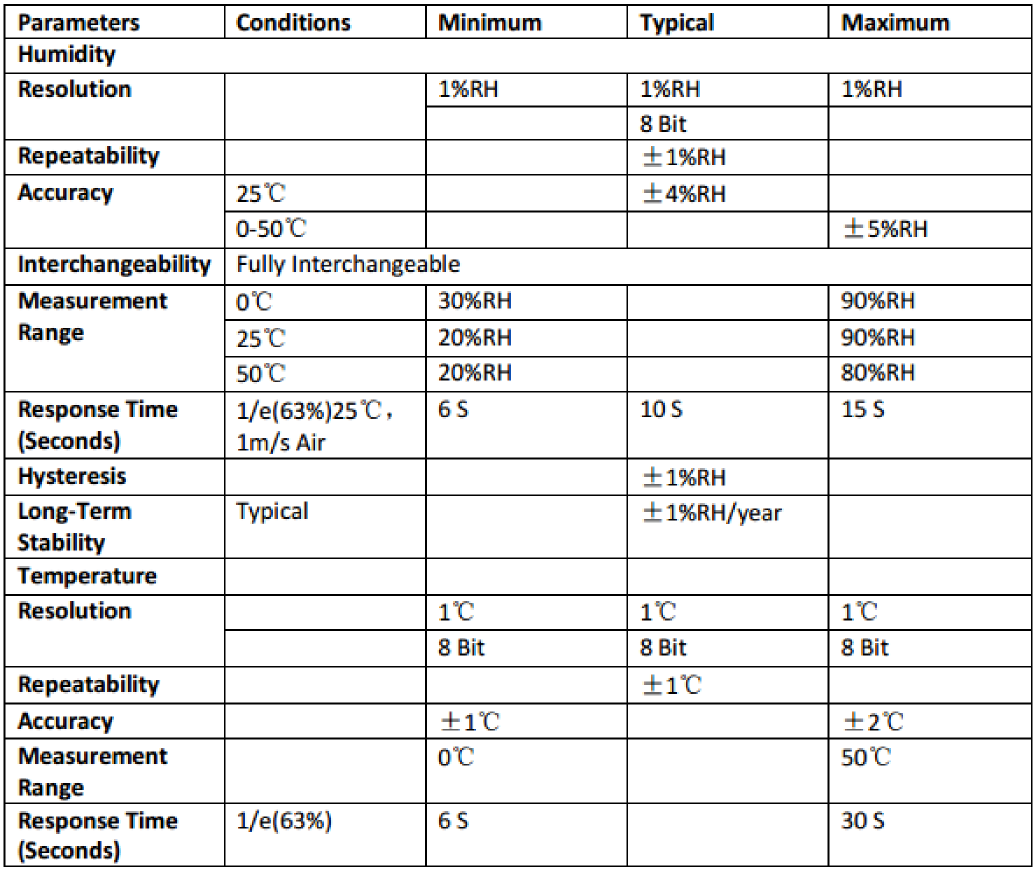
Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA

Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
LED_BUILTIN	13
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Table 3-2 Overview of DHT 11 sensor [6]

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT 11	20-90%RH 0-50 ℃	±5％RH	±2℃	1	4 Pin Single Row

Table 3-3 Detailed technical specifications of DHT 11 sensor [6]

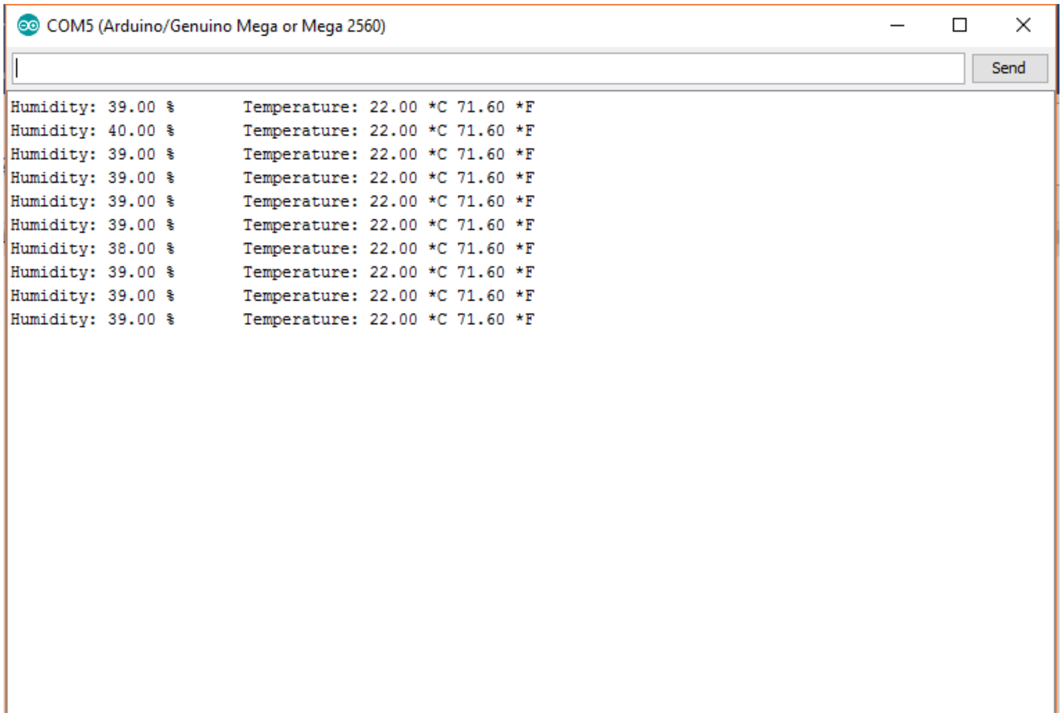


Parameters	Conditions	Minimum	Typical	Maximum
Humidity				
Resolution		1%RH	1%RH	1%RH
			8 Bit	
Repeatability			±1%RH	
Accuracy	25℃		±4%RH	
	0-50℃			±5%RH
Interchangeability	Fully Interchangeable			
Measurement Range	0℃	30%RH		90%RH
	25℃	20%RH		90%RH
	50℃	20%RH		80%RH
Response Time (Seconds)	1/e(63%)25℃，1m/s Air	6 S	10 S	15 S
Hysteresis			±1%RH	
Long-Term Stability	Typical		±1%RH/year	
Temperature				
Resolution		1℃	1℃	1℃
		8 Bit	8 Bit	8 Bit
Repeatability			±1℃	
Accuracy		±1℃		±2℃
Measurement Range		0℃		50℃
Response Time (Seconds)	1/e(63%)	6 S		30 S

Setting up of the sensor is simple and it doesn’t take more than a few minutes to have it up and running. To test if the sensor is working, there are two things that need to be done.

1. Physically connect the Arduino board with the sensor and see if the LED on the DHT sensor is glowing.
2. Flash the DHT11 code using the Arduino software.

The Arduino software is easily available and can be downloaded at Arduino.cc. It is installable on the common operating systems (Windows, Mac OSX, Linux). Once the code is flashed on to the Arduino (after choosing and connecting to the correct COM port), the serial monitor can be opened (Ctrl + M) and the local Dry Bulb Temperature and Relative Humidity values are displayed.



```
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 40.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 38.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
Humidity: 39.00 %       Temperature: 22.00 *C 71.60 *F
```

Figure 3-12 Arduino Display of Temperature and Humidity

Now that the data can be displayed on Arduino's serial monitor, the next step in the data acquisition process is to pull in the data into LabView and sort it as needed. To set up the data acquisition loop, following steps need to be followed:

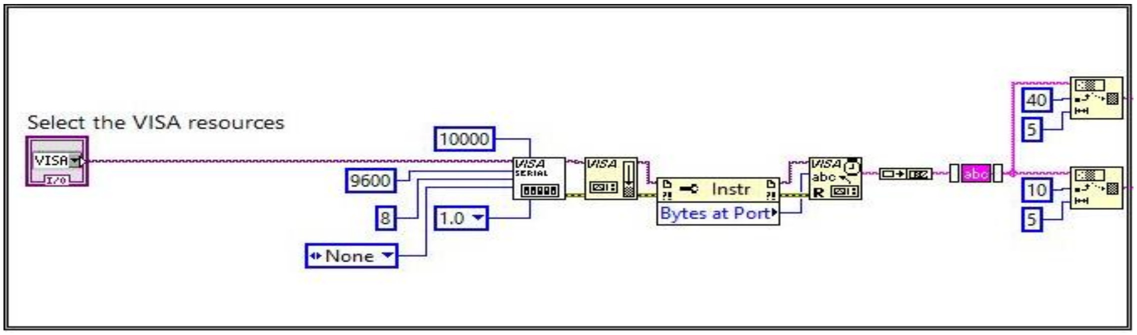


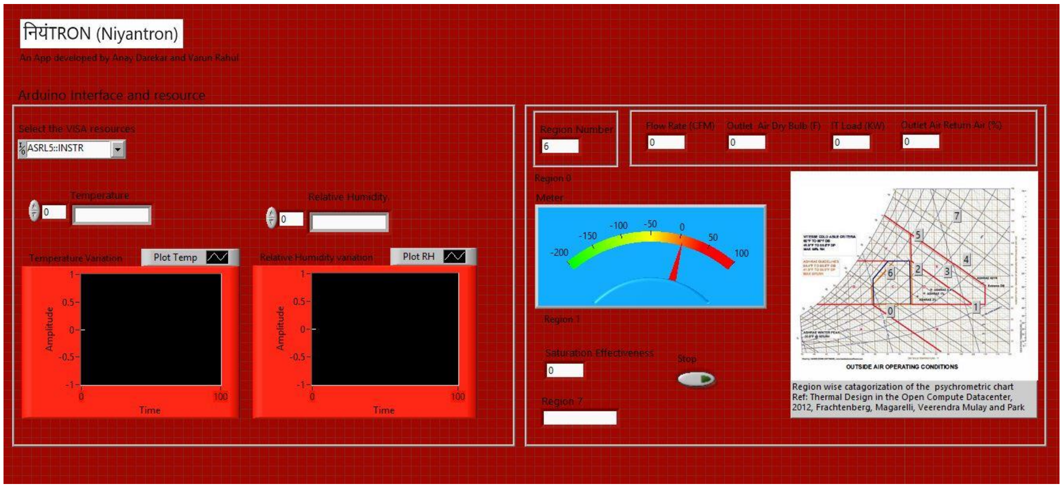
Figure 3-13 Data Acquisition loop in LabView to extract data from Arduino

1. The loop involves accessing the COM port to which the Arduino is connected. Once the port is accessed, the data is pulled in via the Visa Configure Serial Port, where the necessary details regarding the incoming data is specified.
2. The details that to be specified here are: Visa Resource Name, Baud Rate, Data Bits, Parity and Flow Control.
3. The information then proceeds to the Property Node. The option of Bytes at ort is selected here. This block specifies the amount of information coming in in the form of Bytes.
4. The next block is the Read VISA. This reads the specified number of Bytes coming in from the device specified in the Visa Resource Name and returns the data in the read buffer.
5. After concatenating the strings, the incoming data is split and sorted for the required value (of Temperature and Relative Humidity) by using the String Subset block. Here the exact characters with offsets is specifies for extraction.
6. Before the data is sent to the MATLAB code for further processing, there is a Scan from String, to access the Sub String done in the previous block.
7. Once the values are passed to the MATLAB code, based on the specified condition the MATLAB code returns a value between 0 & 7 and as per the number a subsequent case structure is triggered. There are 4 cases possible in NiyanTron:

1. Economizer
2. Direct Evaporative Cooling
3. Return Air Mixing
4. Rejection of outside air

As the Sub VI is triggered based on the associated conditioning operation, an interface gets displayed on the screen and the file proceeds to be saved in the specified directory.

The following figures show the Arduino based DAQ interfaces



नियंTRON (Niyantron)

An App developed by Anay Darekar and Varun Rahul

Arduino interface and resource

Select the VISA resources: ASRL5::INSTR

Temperature: 0

Relative Humidity: 0

Region Number: 6

Flow Rate (CFM): 0

Outlet Air Dry Bulb (F): 0

IT Load (KW): 0

Outlet Air Return Air (%): 0

Saturation Effectiveness: 0

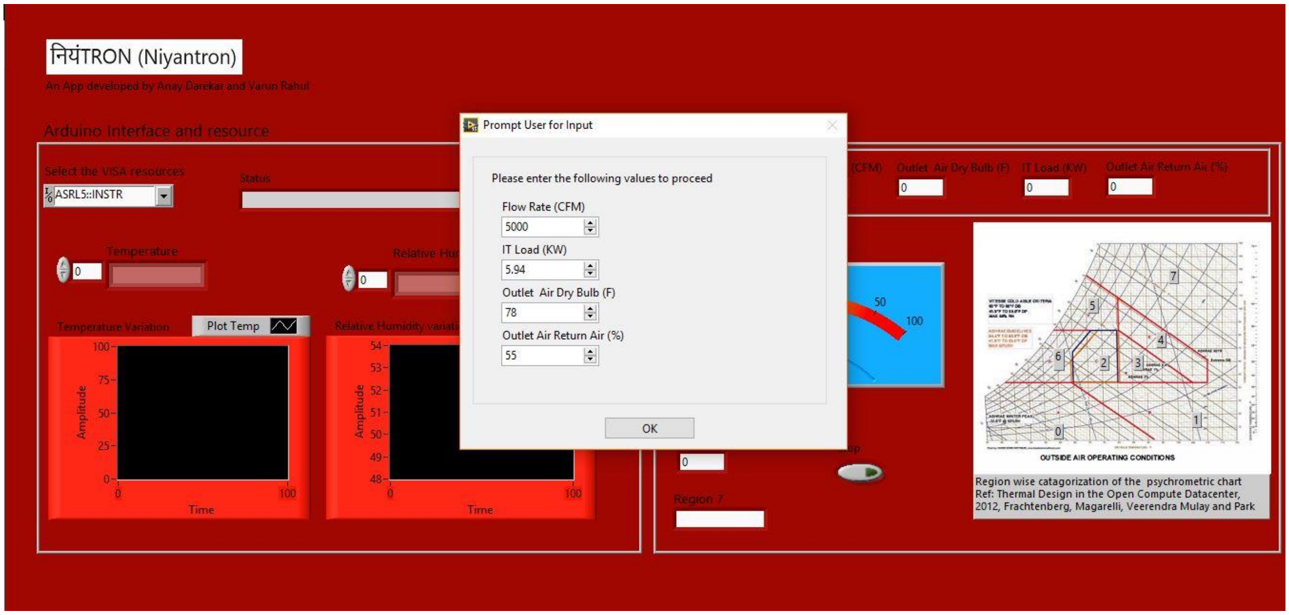
Region wise catagorization of the psychrometric chart
Ref: Thermal Design in the Open Compute Datacenter, 2012, Frachtenberg, Magarelli, Veerendra Mulay and Park

Figure 3-14 Arduino based interface of NiyanTron

The interface developed is simple and easy to use. It has an interactive display with options to choose the kind of Instrument used. There is also live display of data coming in and the data that is being processed in the background or in other interfaces.

Just as discussed previously, based on the region number in the case structure, the Sub VI corresponding to the type of cooling or conditioning that needs to be performed is triggered and the VI pops up. If any additional information needs to be supplied for the VI to perform further calculations, it can be done so by entering it in the respective dialogue boxes that appear.

The back end is discussed in the next page. The diagram is divided in to different sections: DAQ loop → MATLAB Processing→ Case Structure→ Sub VI→ Save to file.



नियंTRON (Niyantron)

An App developed by Anay Darekar and Varun Rahul

Arduino Interface and resource

Select the VISA resources: ASRL5::INSTR

Temperature: 0

Relative Hu: 0

Prompt User for Input

Please enter the following values to proceed

Flow Rate (CFM): 5000

IT Load (KW): 5.94

Outlet Air Dry Bulb (F): 78

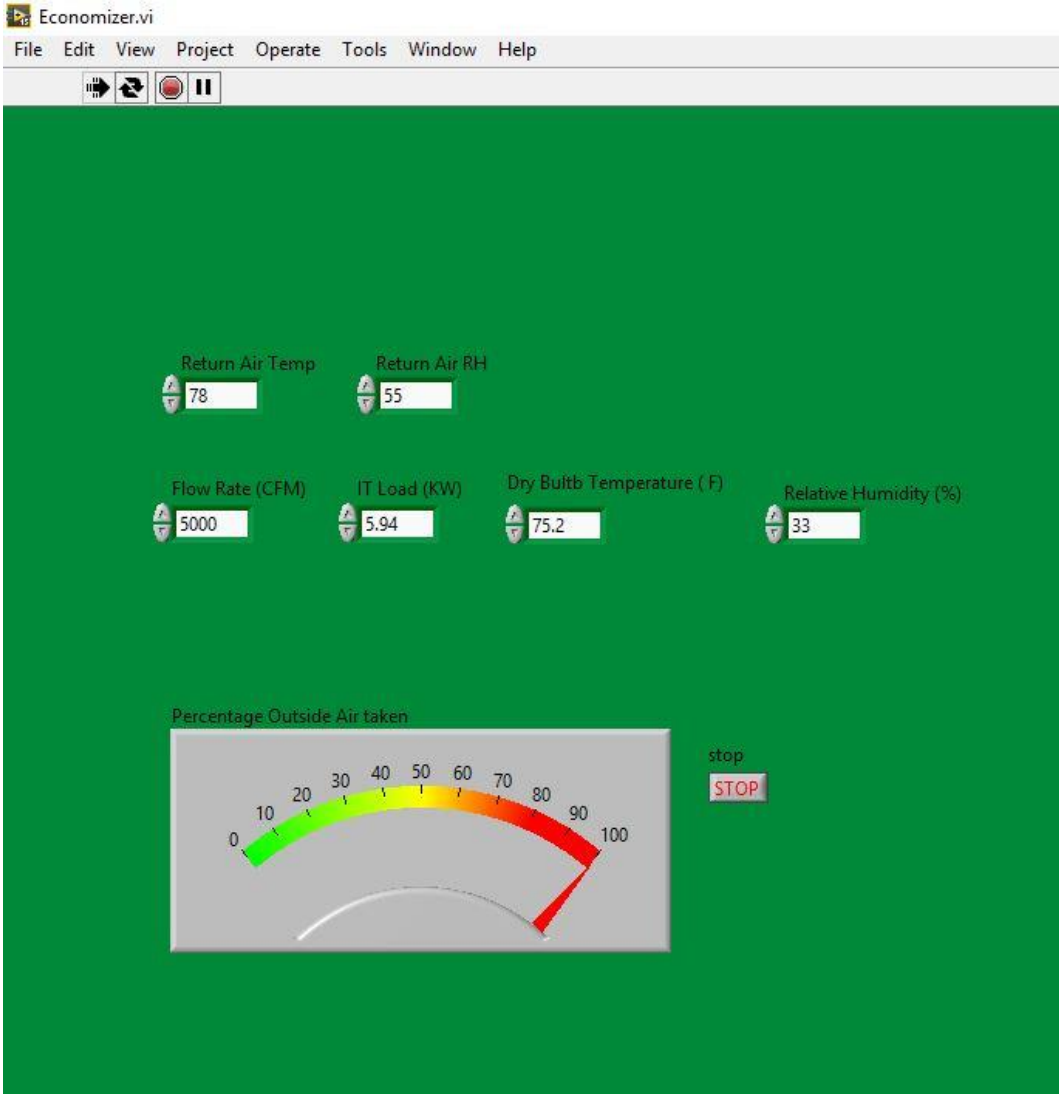
Outlet Air Return Air (%): 55

(CFM) Outlet Air Dry Bulb (F): 0 IT Load (KW): 0 Outlet Air Return Air (%): 0

Region 7

Region wise catagorization of the psychrometric chart
Ref: Thermal Design in the Open Compute Datacenter, 2012, Frachtenberg, Magarelli, Veerendra Mulay and Park

Figure 3-15 Initial prompt to enter values



Return Air Temp 78

Return Air RH 55

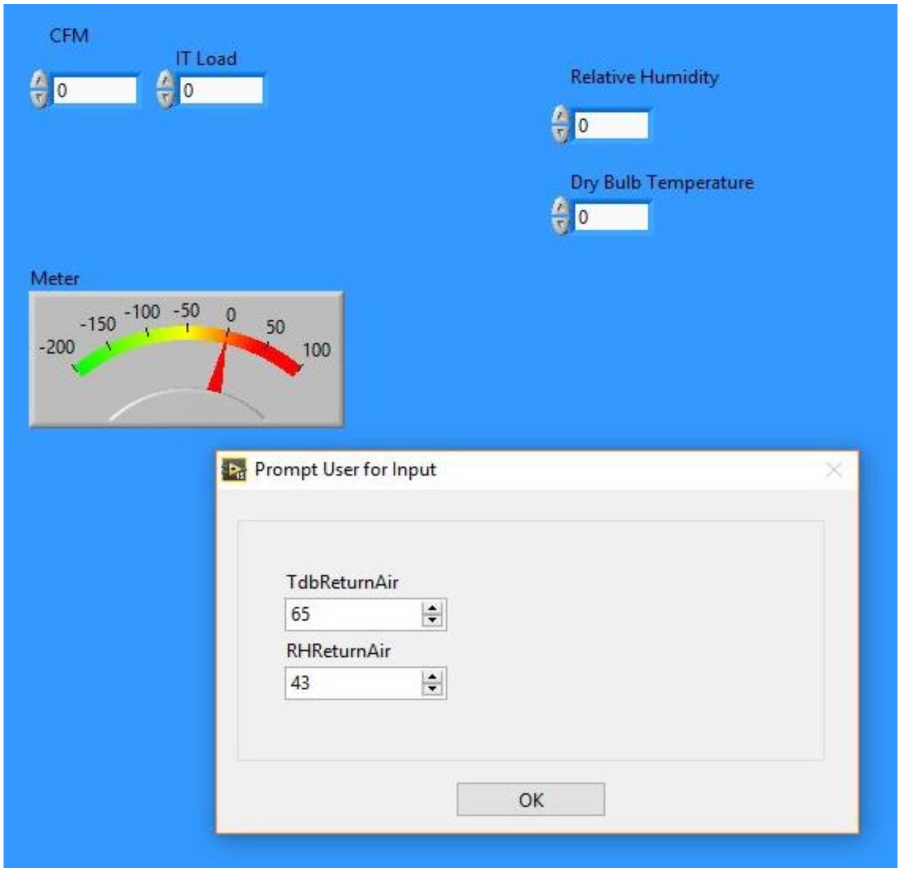
Flow Rate (CFM) 5000

IT Load (KW) 5.94

Dry Bultb Temperature (F) 75.2

Relative Humidity (%) 33

Figure 3-16 Economizer Interface



CFM 0

IT Load 0

Relative Humidity 0

Dry Bulb Temperature 0

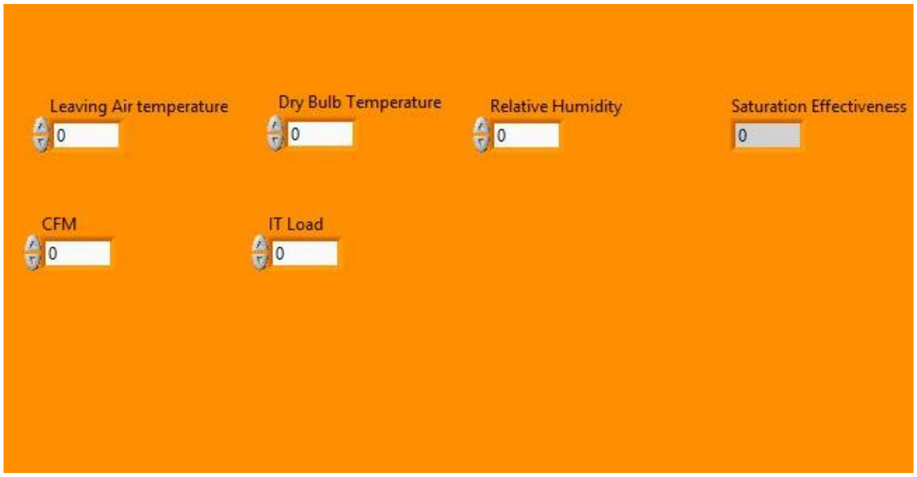
Meter

Prompt User for Input

TdbReturnAir 65

RHReturnAir 43

Figure 3-17 Return Air interface



Leaving Air temperature 0

Dry Bulb Temperature 0

Relative Humidity 0

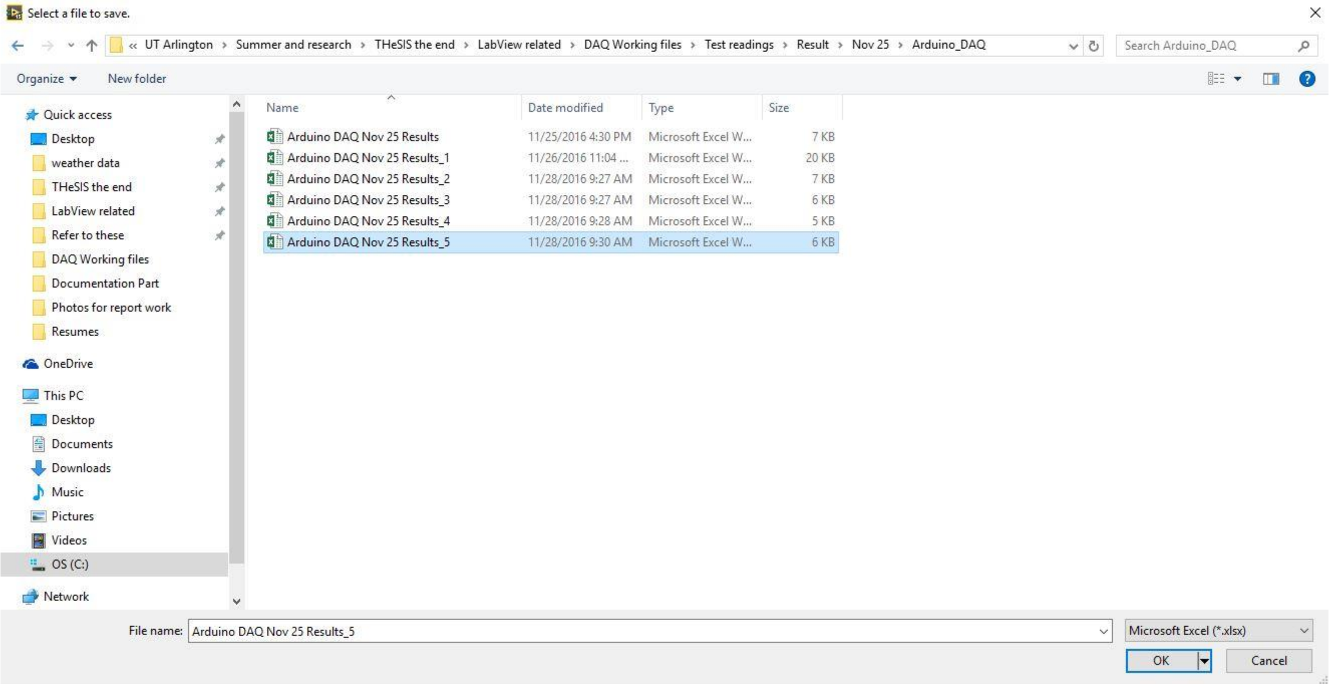
Saturation Effectiveness 0

CFM 0

IT Load 0

Figure 3-18 Direct Evaporative Cooling interface

Once the data is processes, the results are saved in a tabular format, as an Excel sheet. The fie name and location can be selected as per choice in a window as shown below



Name	Date modified	Type	Size
Arduino DAQ Nov 25 Results	11/25/2016 4:30 PM	Microsoft Excel W...	7 KB
Arduino DAQ Nov 25 Results_1	11/26/2016 11:04 ...	Microsoft Excel W...	20 KB
Arduino DAQ Nov 25 Results_2	11/28/2016 9:27 AM	Microsoft Excel W...	7 KB
Arduino DAQ Nov 25 Results_3	11/28/2016 9:27 AM	Microsoft Excel W...	6 KB
Arduino DAQ Nov 25 Results_4	11/28/2016 9:28 AM	Microsoft Excel W...	5 KB
Arduino DAQ Nov 25 Results_5	11/28/2016 9:30 AM	Microsoft Excel W...	6 KB

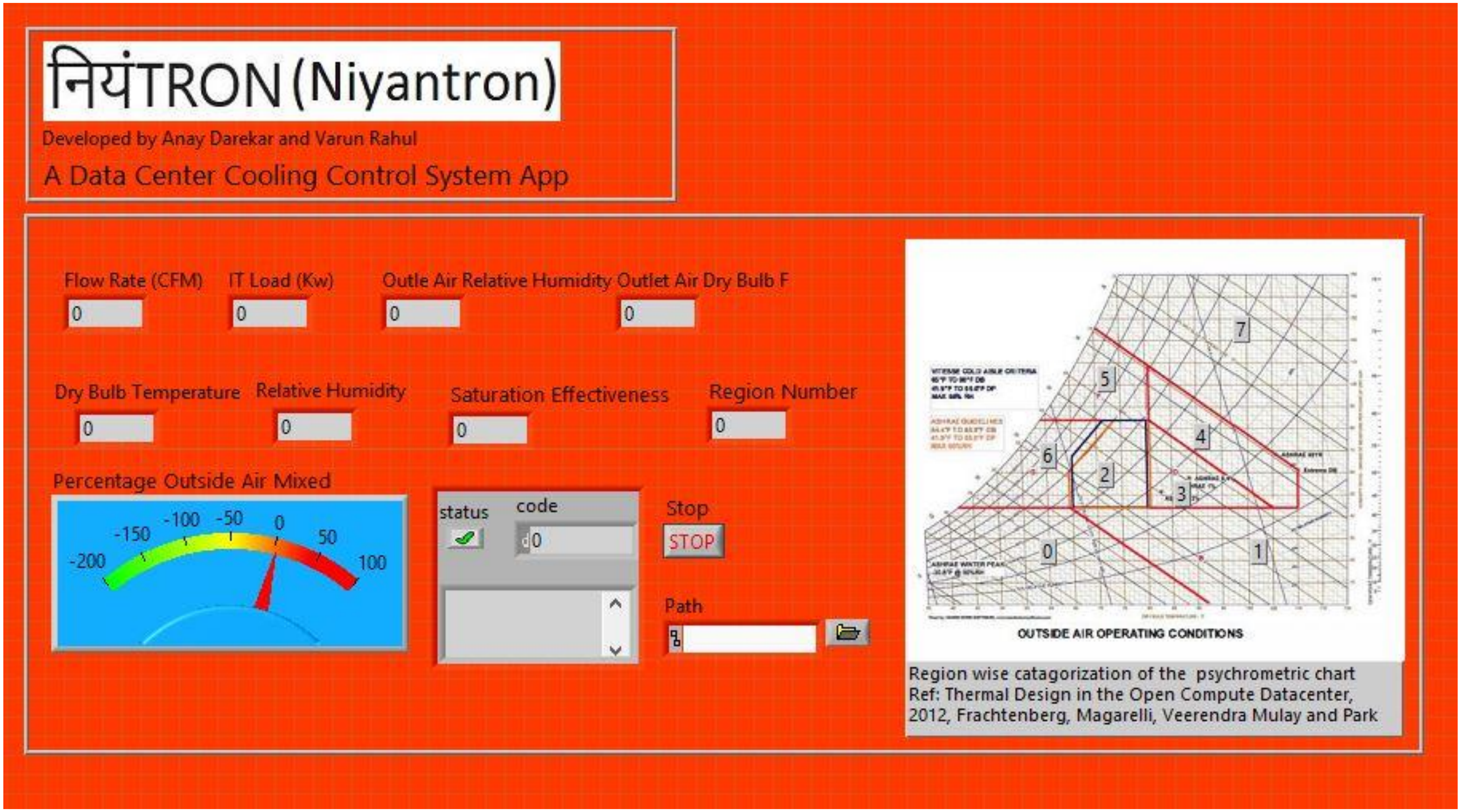
File name: Arduino DAQ Nov 25 Results_5

Microsoft Excel (*.xlsx)

Figure 3-19 Prompt to save file

Once the file is saved, the loop continues. The saving of the file is programmed to be on the outside of the loop, so that it is not prompted during every iteration. The loop is iteration is set at 24 to represent the number of hours of the day. It is done so to also simulate each data point to correspond to an hour of the day. By making a few changes in the code, the frequency and duration to receive the data can be altered to suit the system.

Since this form of DAQ configuration is meant for recording live data, the results are discussed in the Results, Discussions and Conclusions chapter.



नियंTRON (Niyantron)

Developed by Anay Darekar and Varun Rahul

A Data Center Cooling Control System App

Flow Rate (CFM): 0

IT Load (Kw): 0

Outle Air Relative Humidity: 0

Outlet Air Dry Bulb F: 0

Dry Bulb Temperature: 0

Relative Humidity: 0

Saturation Effectiveness: 0

Region Number: 0

status: ✓

code: d0

Region wise catagorization of the psychrometric chart
Ref: Thermal Design in the Open Compute Datacenter, 2012, Frachtenberg, Magarelli, Veerendra Mulay and Park

Percentage Outside Air Mixed
-200
-150
-100
-50
0
50
100
Stop
STOP
Path
7
5
6
2
4
3
0
1
OUTSIDE AIR OPERATING CONDITIONS

Figure 3-20: NiyanTron - Excel Based DAQ

Excel Based DAQ Unit

If the app needs to be tested for a location other than a current location to gauge the cooling system configuration and requirement and the cooling patterns based on the loading conditions, inputting the weather data can also help in identifying or addressing the aforementioned needs.

For this purpose, the Arduino based DAQ has been replaced by a DAQ system that accepts a file in the form of an excel sheet containing the values of the Dry Bulb Temperature and Relative Humidity. Rest of the program is the same as the Arduino based DAQ model. The data that is saved as an excel sheet can be plotted as a graph to understand the interdependencies and variations of different parameters.

The concept of using this DAQ configuration helped analyze the cooling over different cities spread over USA: Dallas, TX, Chicago, IL, Tucson, AZ, Albuquerque, NM.

For the purpose of this study and this thesis, the regions that have been studied and discussed are Dallas, TX and Chicago, IL.

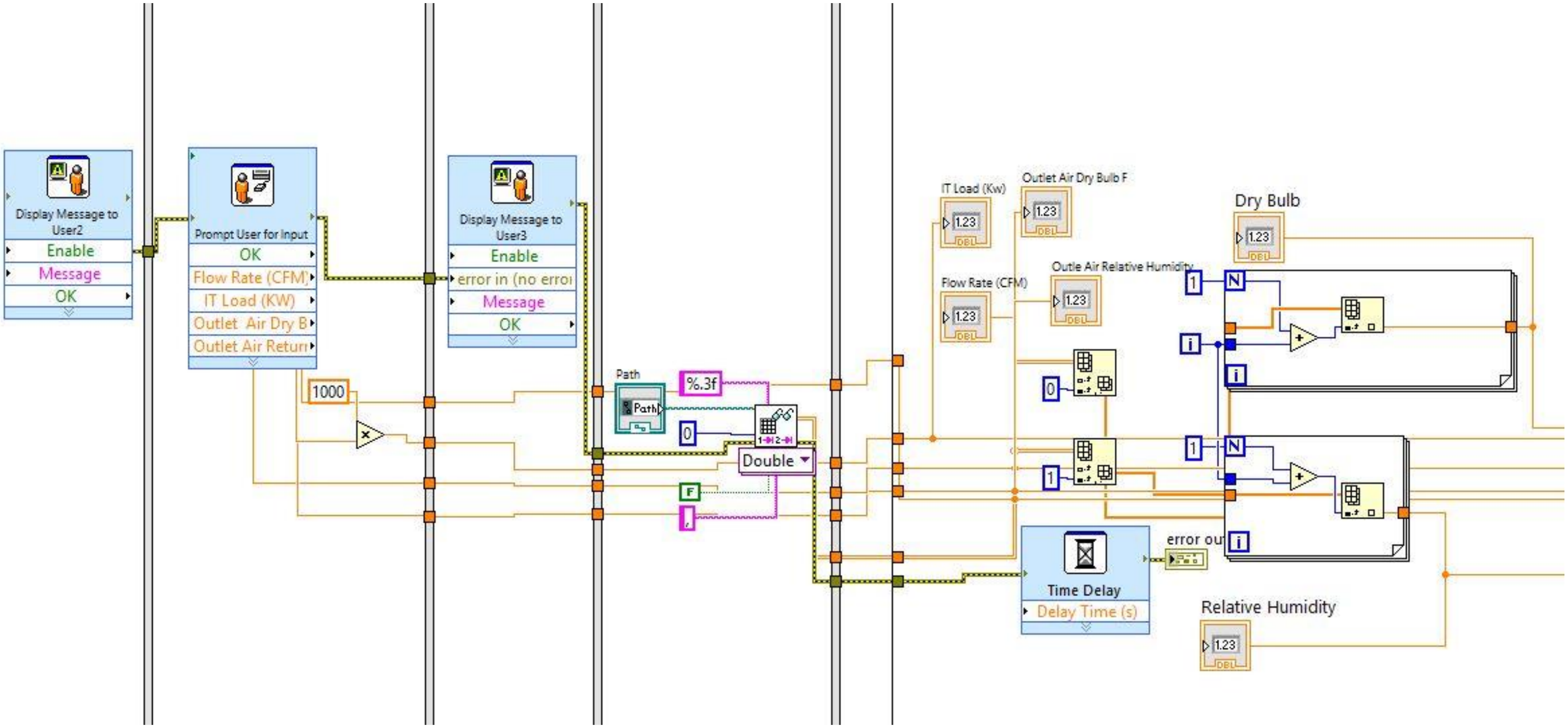


Figure 3-21 Back End Part -DAQ Part

The following are the steps before the actual data is sent into the MATLAB script for processing. The initial steps are meant to setup the variable pertinent to the calculations in the later stages.

1. A dialog box prompts introduces the App and gives a brief information about it.
2. Once the above is closed, a dialog menu pops up, asking for the following:
    1. Flow Rate (CFM)
    2. IT Load (KW)
    3. Supply Air Dry Bulb Temperature
    4. Supply Air Relative Humidity
3. The Next step is to access the file location where the weather data is saved. This could be at any user desired location on the device being used.
4. Once the data is accessed, iterative loops taken in data as required and send it to the code for processing.

The MATLAB code then decides the region where the incoming data lies in and accordingly sends the region number and the data to the case structure and the corresponding sub VI for further processing.

Cooling Systems design

This section discusses the cooling system that was proposed to go as a demonstrative setup for the App’s hardware implementation part. The cooling system was designed and developed as a part of the project’s first phase, the next phase being building the system with the components mentioned in Chapter 4. The design was entirely built in Solidworks, with different views to display the cooling configurations involved in it.

The cost analysis of the cooling system is discussed in the next chapter. This section is meant only to shed light on the possibility of developing a custom cooling unit to demonstrate a self-sufficient project.

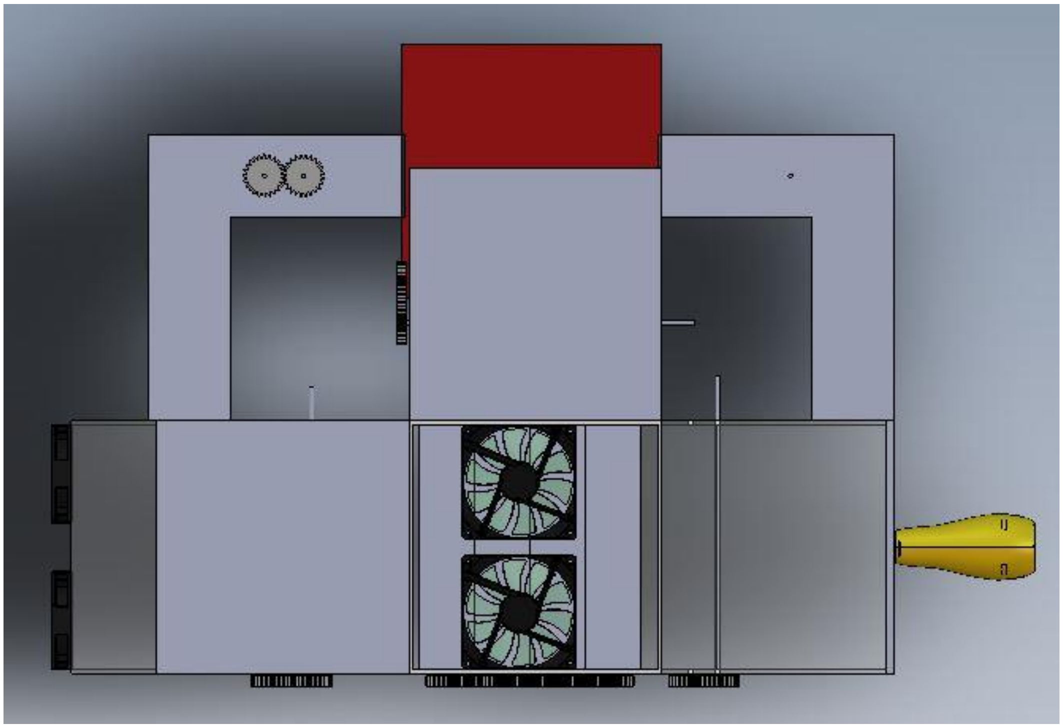


Figure 3-22 Top view of the Cooling System

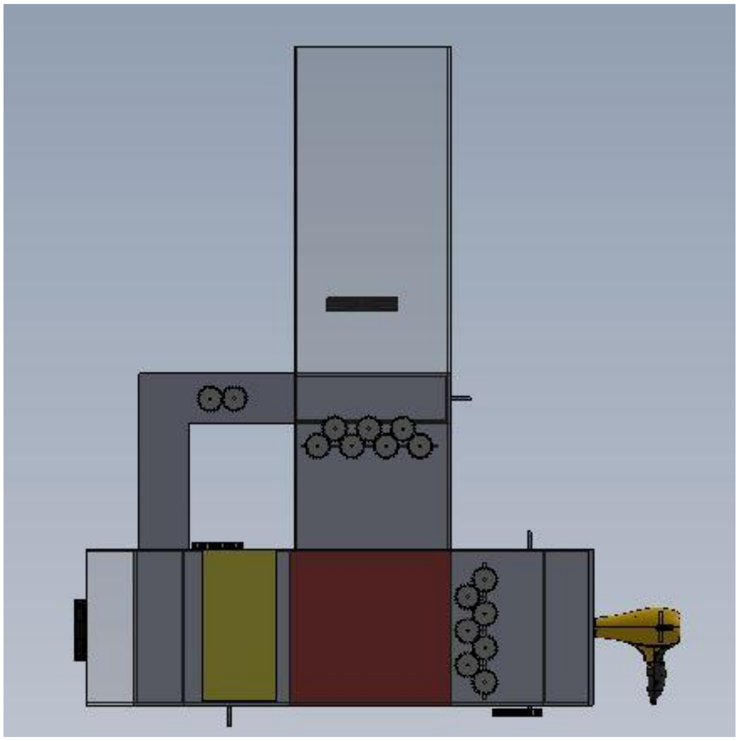


Figure 3-23 Front View of the Cooling System

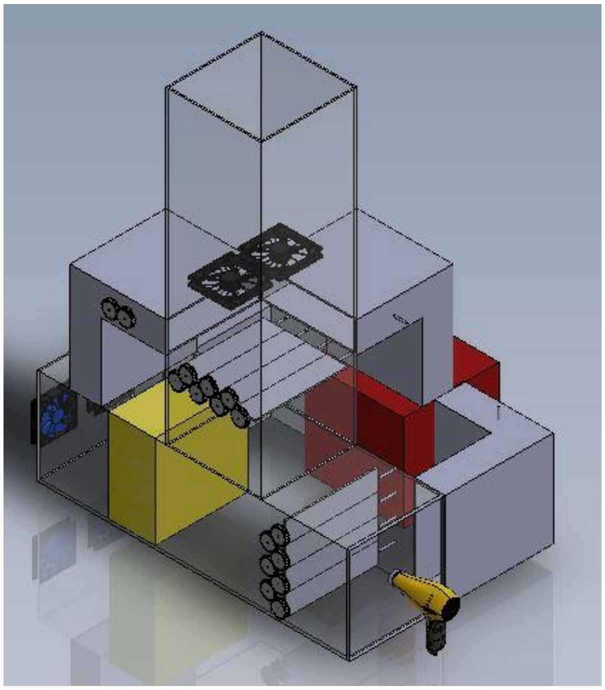


Figure 3-24 Isometric View

Component design

The cooling system design was made with the same aim as the app, i.e., to develop the system that is scaled down and making the setup economical by using easily available components and 3-D printing the ones that aren’t easily found.

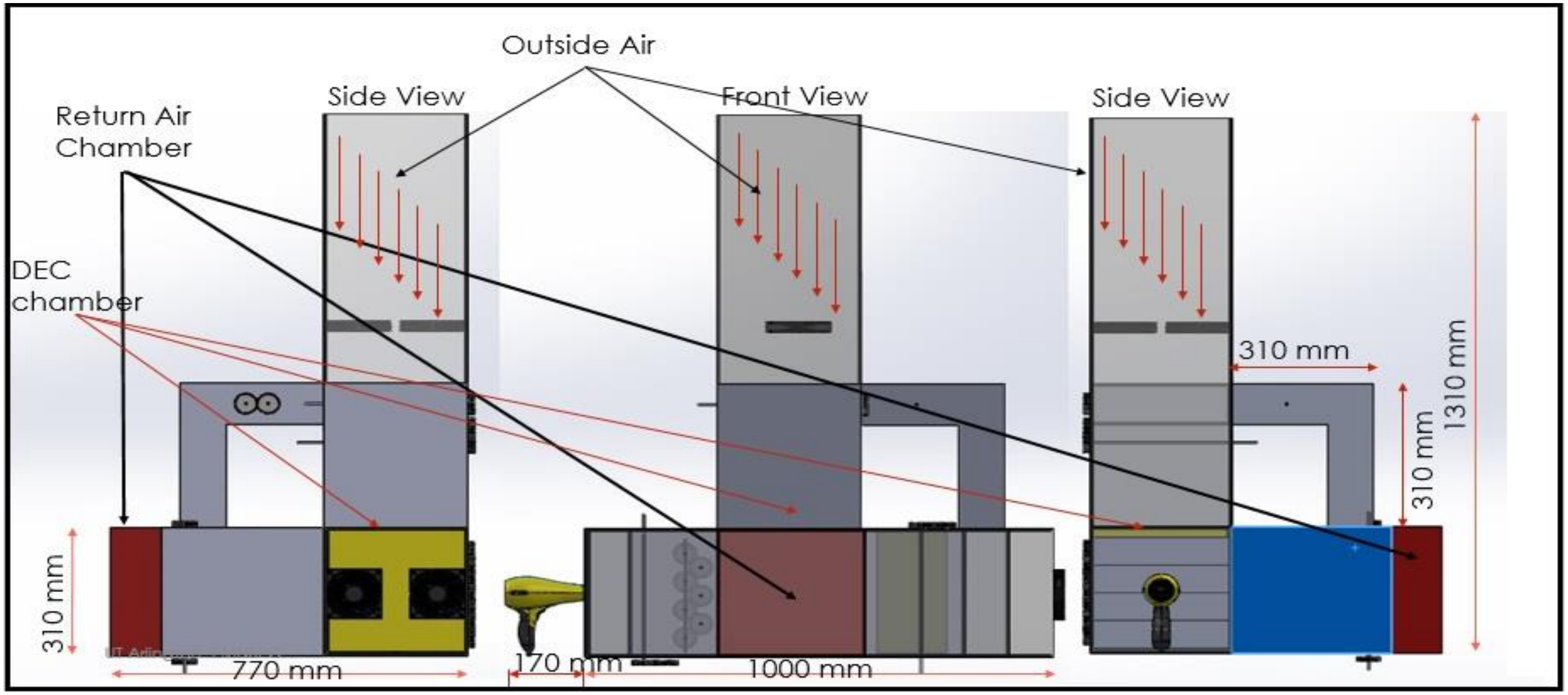


Figure 3-25 Detailed Sketch of the Cooling Systems

Chapter 4 Results, Discussion and Conclusion

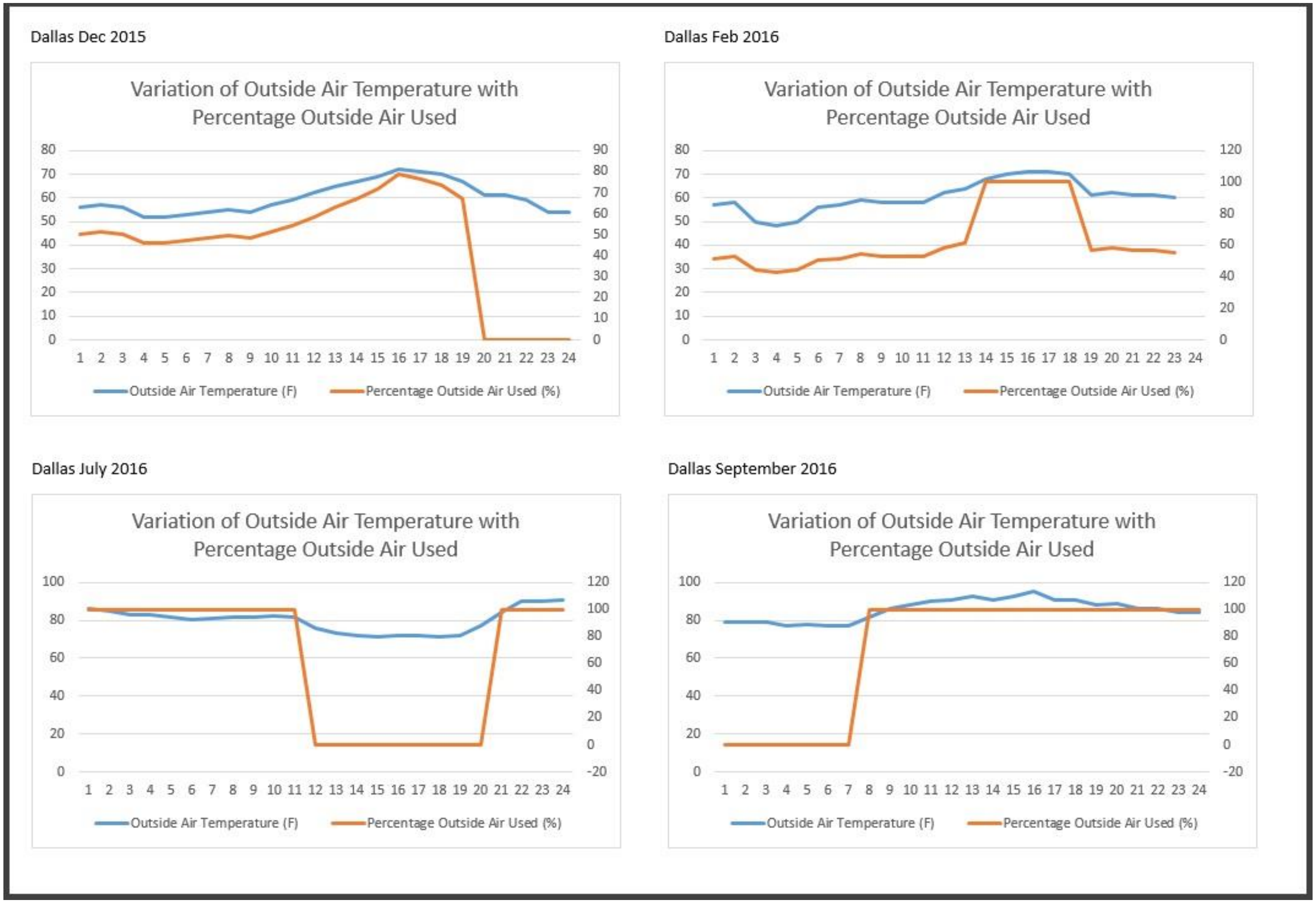
This section discusses the results of the app developed and talks about the scope of the app implementation. It also mentions the future work and suggests improvement that can be made in the existing app.

Results

This section discusses the variation in temperature and humidity of the incoming air and supply air (going into the data center). The supply air properties are calculated based on the calculations done in each sub VI which in turn represents a cooling system. Although on must remember that the calculation part is omitted only in the Economizer section (as the air is sent into the data center without any conditioning).

The weather data chosen to be analyzed was taken for the months December 2015, February 2016, July 2016 and September 2016. This was done to cover the main seasons – Winter, Spring, Summer and Fall, so that the system can be tested for a wide range of data points. The 15th day of each month was chosen as weather data reference point. The selection of the date was trivial. The weather data for the above-mentioned months were taken for Chicago, IL and Dallas, TX [8]. The Arduino DAQ was used to record and plot live data and was done on November 27, 2016. 24 data points are plotted for each reading to represent the 24 hours of the day.

Final Dry Bulb Temperature and Relative Humidity after DEC and Return Air mixing falls in ASHRAE Recommended Zone. This give an indication that the analytical part of the app works perfectly. The scope of improvement can be in making the control system robust and have a faster rate of response.



Dallas Dec 2015

Dallas Feb 2016

Dallas July 2016

Dallas September 2016

Figure 4-1 Variation of Dallas weather over 2015-16

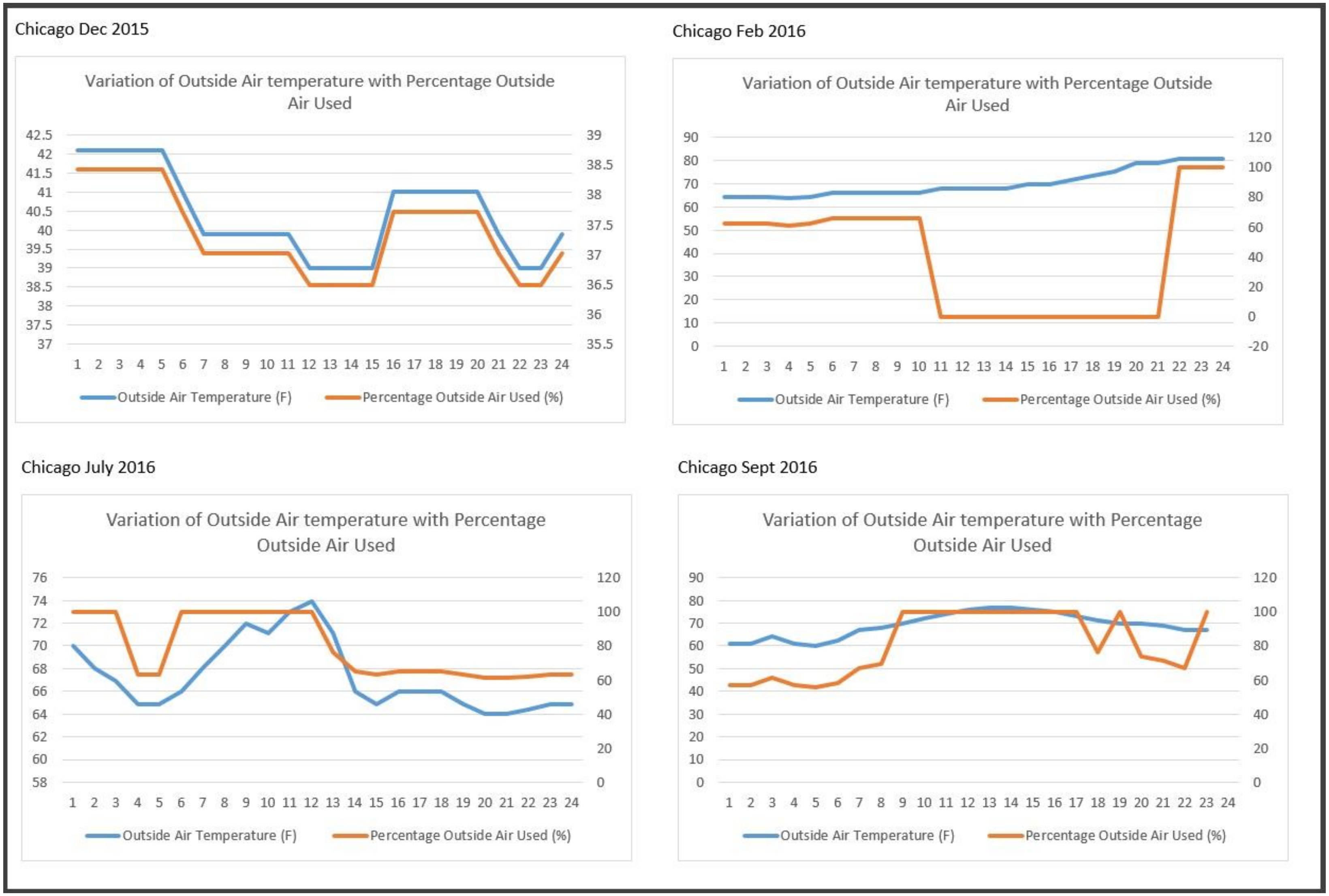


Figure 4-2 Variation of Chicago, IL weather over 2015-16

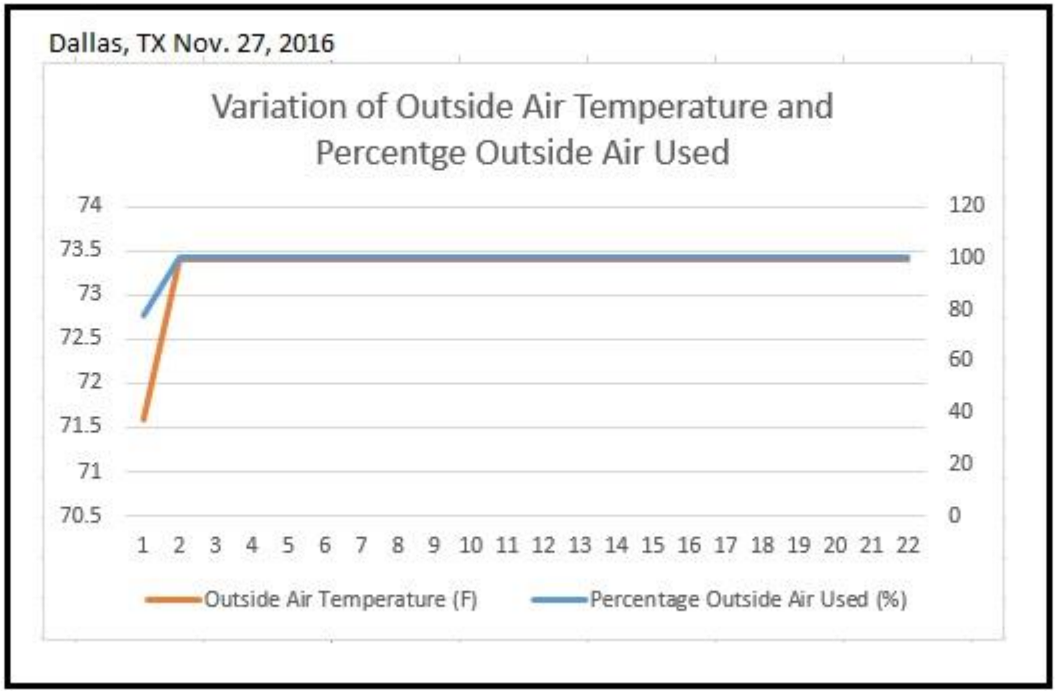


Figure 4-3 Variation of Dallas weather Nov. 28,2016

Discussions and Future Work

The response of the app to the cooling processes was as expected with not a significant amount of deviation in the results. This goes to show that initial implementation of the app has been successful with a series of simulated test conditions. The next step would be to go ahead with porting the technology with a proper industry read data center cooling system capable of carrying out the documented operation in this app.

That being said, on the app more other cooling techniques such as Indirect Evaporative Cooling and DX Cooling need to be added so that the app addresses the control of all types of cooling systems. This implementation can again be done in as first as a simulated test setup and then proceed to hardware integration.

The existing hardware, DAQ configuration and control system on the whole can be improved and made more robust by :

1. Implementing wireless sensors. This way we can localize the sensing process and also by placing the sensors between racks in the data center facility, we can have a better idea about the variation of temperature on the data center and accordingly direct the cooling towards the rack that require it.
2. Setting up of the sensors will in a way facilitate a means of subsequently developing a learning algorithm for the app, and the app can improve upon it's predictive control by sensing different heat load conditions.

To improve upon the scope of implementation of the app, it can be developed for Android and iOS. This will not just ensure a higher reach among the professional but also achieve significant portability.

Before attempting the hardware integration, the app can be tested for regions other than the United States of America, to see how consistent the results from the app are.

Appendix A

MATLAB Code

```
syms  Tdp Twb region a b c d e f g;
syms TdpFarenheit gamma2 TwbCenti

fprintf('Enter the measured temperatures \n');
%Tdb= input('Enter dry bulb \n');
%Tdp= input('Enter dew point \n');
%Twb= input('Enter Wet bulb');
%RH= input('Enter the realtive humidity');

TdbCenti= (Tdb-32)/1.8000;
fprintf('Tdb in centigrade =');
disp(TdbCenti);
%TwbCenti= (Twb-32)/1.8000;
%fprintf('Twb in centigrade =');
%disp(TwbCenti);
%calculation for Tdp in centigrade and Relative Humidity
%Es=6.112*(2.718^((17.67*TdbCenti)/(273.3+TdbCenti)));
%fprintf('Es=')
%disp(Es);
%Ew=6.112*(2.718^((17.67*TwbCenti)/(273.3+TwbCenti)));
%fprintf('Ew=')
%disp(Ew);
%E=Ew-((0.00066)*(1+0.00115*TwbCenti)*(TdbCenti-TwbCenti)*(1013));
%fprintf('E=')
%disp(E);
%RH=100*(E/Es);
ABHReturn=((6.112*(2.718^((17.67*TdbCenti)/(243.5+TdbCenti)))*RH*2.1674)/(273
.15+TdbCenti))/1000;
gamma=(log(RH/100)+((17.67*TdbCenti)/(243.5+TdbCenti)));
fprintf('value of gamma is');
disp(gamma);
Tdp=(243.5*gamma)/(17.67-gamma);
fprintf('Relative humidity is = ')
disp(RH)
fprintf('Tdp Dew point temperature in degree Centigrade is =')
disp(Tdp)
TdpFarenheit= ((Tdp*1.8)+32);
fprintf('Tdp in Fahrenheit is =')
disp(TdpFarenheit)
```

```
%calculation for Twb
TwbCenti=TdbCenti*atan(0.151977*((RH+8.313659)^0.5)) +atan(TdbCenti+RH)-
atan(RH-1.676331)+0.00391838*((RH)^1.5)*atan(0.023101*RH)-4.686035;
fprintf('Twb in Centigrade=')
disp(TwbCenti)
Twb=((TwbCenti*1.8)+32);
fprintf('Twb in Fahrenheit=')
disp(Twb)

if(Tdb<=52&&TdpFarenheit<=41.9&&Twb<46.8)
   fprintf('REGION IS A \n');
   region=0;
%elseif
 %   fprintf('POINT LIES IN REGION B,C,D,G,E');
elseif (Tdb>52&&TdpFarenheit<41.9&&Twb<46.8)
   region=1;
   fprintf('region is B \n')
elseif
(Tdb>65&&Tdb<80&&TdpFarenheit>41.9&&TdpFarenheit<59&&Twb<66.137)
   fprintf('Region is C \n')
   region=2;
elseif(Tdb>80&&Tdb>80&&Twb<65.76)
   fprintf('Region is D \n')
   region=3;
elseif(Tdb>80&&TdpFarenheit>41.9&&Twb>65.76)
   fprintf('Region is E \n')
   region=4;
elseif(Tdb<80&&TdpFarenheit>59.0&&Twb>65.76)
   fprintf('Region is F\ n')
   region=5;
else
   fprintf('POINT LIES IN REGION G');
   region=6;
end
switch region

case 0
   disp('cooling equipment required is Return Air');
case 1
   disp('cooling equipment required is Economizer');
case 2
   disp('cooling equipment required is Economizer');
case 3
```

```
    disp('cooling equipment required is Economizer (if req, DEC)');
case 4
    disp('cooling equipment required is Economizer (if req, DEC) ');
case 5
    disp('cooling equipment required is Return Air, DEC bypassed');
case 6
    disp('cooling equipment required is Return Air, DEC bypassed');
end
```

Appendix B

Arduino Code [7]

```
#include <DHT.h>
// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain
#include "DHT.h"
#define DHTPIN 2     // what digital pin we're connected to
// Uncomment whatever type you're using!
#define DHTTYPE DHT11   // DHT 11
//#define DHTTYPE DHT22   // DHT 22 (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)
// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 4 (on the right) of the sensor to GROUND
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor
// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
 //Serial.println("DHTxx test!");
  dht.begin();
}
void loop() {
  // Wait a few seconds between measurements.
  delay(2000);
  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (it’s a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);
  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
```

```
  Serial.println("Failed to read from DHT sensor!");
  return;
}
// Compute heat index in Fahrenheit (the default)
float hif = dht.computeHeatIndex(f, h);
// Compute heat index in Celsius (isFahreheit = false)
float hic = dht.computeHeatIndex(t, h, false);
Serial.print("Humidity: ");
Serial.print(h);
Serial.print(" %\t");
Serial.print("Temperature: ");
Serial.print(t);
Serial.print(" *C ");
Serial.print(f);
Serial.println(" *F\t");
//Serial.print("Heat index: ");
//Serial.print(hic);
//Serial.print(" *C ");
//Serial.print(hif);
//Serial.println(" *F");

}
```

References

Literature references

[1] ASHRAE TC 9.9, “2011 Thermal Guidelines for Data Processing Environments – Expanded Data Center,” pp. 1–50, 2011

[2] Best practices for datacom facility energy efficiency. (2008). Atlanta, GA: American Society of Heating, Refrigerating, and Air-Conditioning Engineers.

[3] [Brochure]. (n.d.). Retrieved from http://literature.cdn.keysight.com/litweb/pdf/5989-8706EN.pdf?id=2231563

[4] (n.d.). Retrieved from http://www.keysight.com/en/pd-1000000089:epsg:pro-pn-907A/multifunction-module-for-34970a-34972a?nid=-33237.536880685.00&cc=US&lc=eng

[5] Arduino - ArduinoBoardMega2560. (n.d.). Retrieved December 12, 2016, from https://www.arduino.cc/en/Main/ArduinoBoardMega2560

[6] (n.d.). Retrieved from http://www.micropik.com/PDF/dht11.pdf

[7] A. (2016). Adafruit/DHT-sensor-library. Retrieved December 12, 2016, from https://github.com/adafruit/DHT-sensor-library

[8] Weather Forecast & Reports - Long Range & Local | Wunderground | Weather Underground. (n.d.). Retrieved December 12, 2016, from https://www.wunderground.com/

Image References

[9] (n.d.). Retrieved from https://i.kinja-img.com/gawker-media/image/upload/s--ZW-dXt0v--/c_scale,fl_progressive,q_80,w_800/17zlnftieiizbjpg.jpg

[10] (n.d.). Retrieved from http://www.qingonggroup.com/pic/other/2015-10-21-20-17-345.jpg

[11(n.d.). Retrieved from] http://image.slidesharecdn.com/2-150214032338-conversion-gate01/95/2-psychrometric-chart-1-638.jpg?cb=1423905939

[12] @. (2016). A History of ASHRAE Technical Committee TC9.9 and its Impact on Data Center Design and Operation | Electronics Cooling. Retrieved December 12, 2016, from https://www.electronics-cooling.com/2012/12/a-history-of-ashrae-technical-committee-tc9-9-and-its-impact-on-data-center-design-and-operation/

[13] What Is An Economizer – AKA Mixing Box? Economizers Class 101! (2015). Retrieved December 12, 2016, from http://blog.micrometl.com/what-is-an-economizer-aka-mixing-box-economizers-class-101/

[14] (n.d.). Retrieved from http://www.civilengineeringhandbook.tk/heat-recovery/images/7583_588_755.jpg

[15] Psychrometrics (Energy Engineering). (n.d.). Retrieved December 12, 2016, from http://what-when-how.com/energy-engineering/psychrometrics-energy-engineering/

[16] Richmueller, P. B. (2012). D.I.Y. Inspired Evaporative Cooler Design for Remote Military Applications. Retrieved December 12, 2016, from https://muellerdesignlab.wordpress.com/2012/04/20/diy-evaporative-cooler-design/

[17] 2016, A. L. (2016). Keysight / Agilent 34970A / 34972A data loggers from Amplicon. Retrieved December 12, 2016, from https://www.amplicon.com/MandC/product/Data-Panel-Keysight-2792.cfm

[18] Feedback, & Keysight, C. (2000). 34972A LXI data acquisition / data logger switch unit. Retrieved December 12, 2016, from http://www.keysight.com/en/pd-1756491-pn-34972A/lxi-data-acquisition-data-logger-switch-unit?cc=US&lc=eng

Biographical Information

Varun Rahul (Jan 9,1992) is from Pune, India on. He completed his Bachelors in Technology from Amrita School of Engineering, Bangalore, in the field of Mechanical Engineering in 2014. Varun joined The University of Texas at Arlington in January 2015 for the MS program in Mechanical Engineering. He joined the EMNSPC team in Summer 2015 and started contributing to the team’s research. By taking a diverse selection of courses in Thermal, Fluids, Robotics and Unmanned Vehicles, Electronic Packaging to name a few. Over his graduate coursework, Varun also worked as a Graduate Research Assistant at The University of Texas at Arlington Research Institute, Fort Worth, TX. He worked in the Bio-Medical Technologies Division. The work is centered on Product Development in Rehabilitative and Predictive Technology. Varun was also conferred the Roger Goolsby Graduate Student Fellowship scholarship for his good academic standing. Varun graduated from The University of Texas at Arlington on November 28,2016 and plans to join the industry and