ECE 445

SENIOR DESIGN LABORATORY

DESIGN REPORT

Automatic Humidity Sensing and Water Refilling Cool-Mist Humidifier

---

Team No. 11

ANDREW SHERWIN
(zyxie2@illinois.edu)
Jalen Chen
(jalenc3@illinois.edu)
Woojin Kim
(wkim51@illinois.edu)

TA: Surya Vasanth
Professor: Dr. Jonathon Schuh

February 22, 2024

Table of Contents

Abstract

Improper indoor humidity levels are a cause for health issues, such as dry skin or inflamed sinus passages. With the majority of humidifiers on the market being manually controlled, this causes an issue when no operator is present. A room can be under-humidified or over-humidified when there is no one to control the humidity output. As a result, we propose a cost-effective humidifier that will mitigate this problem. Concerned about the health of the operator, a cool-mist humidifier design was chosen. This humidifier communicates via 2.4GHz WIFI with multiple remote humidity sensors before determining the need to turn itself on or off. When the humidifier filter needs more water, the humidifier will activate a water valve that will irrigate the filter to the perfect amount. With all the components combined, this humidifier brings a new perspective to the humidifiers already on the market, guaranteeing the safety and comfort of the user. A truly luxurious experience.

Introduction

Problem and Solution

The United States Environmental Protection Agency (EPA) strongly suggests the regulation of indoor humidity to be between 30%-50%. When indoor humidity falls within these ranges, pollutants, such as chemicals, gasses, mold, and other airborne particles, are minimized in the air. This ultimately helps with allergies, respiratory illnesses, and other health issues.

To maximize the health benefits of our product, we decided to implement a cool mist humidifier. Unlike ultrasonic humidifiers and warm mist humidifiers, this product does not disperse bacterial matters, minerals, or cause your nasal passages, as the US Food and Drug Administration (FDA) suggests. To further improve humidifiers on the market, our humidifier features an automatic-on function and an automatic water tank filling function. The solution has three remote humidity sensors. These sensors detect the humidity and communicate the humidity readings with the humidifier. Using the average of these readings, the humidifier decides whether it is necessary to turn itself on or off. Inside the water tank of the humidifier are two water sensors. One sensor detects whether more water is needed, while the other sensor detects when to stop automatically adding water. With all these improvements, we push out a revolutionary humidifier system onto the market that has a high level of autonomy, providing a luxurious product to the end-user.

Visual Aid

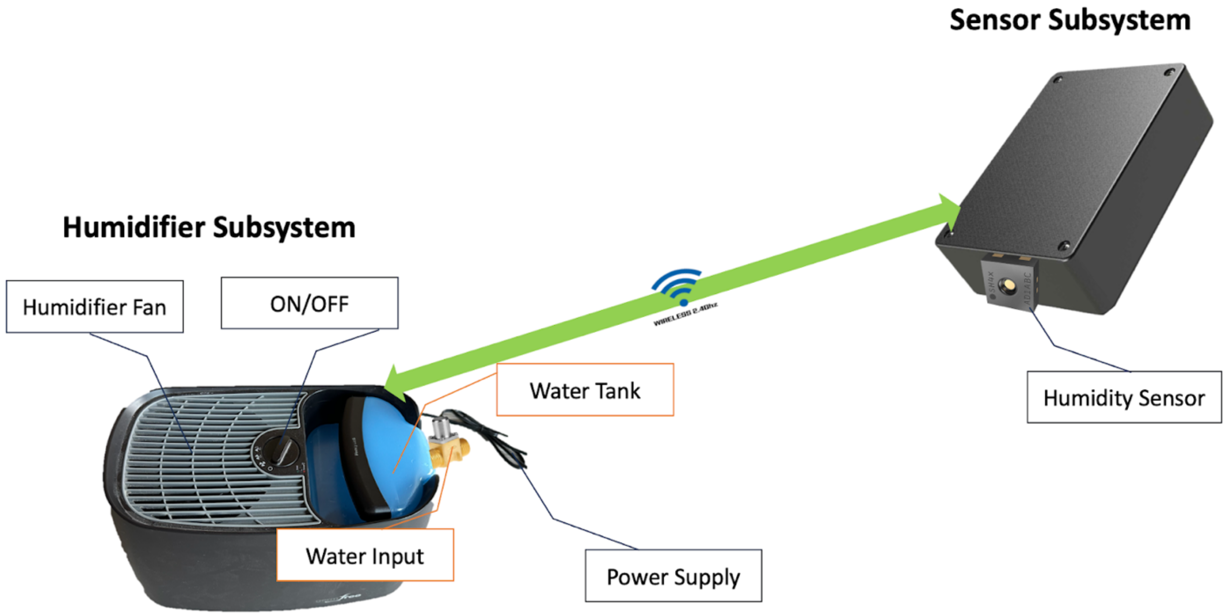

Figure 1: Consumer Visual Aid

The visual aid (*figure 1*) supports better understanding of how our product gives a solution to the suggested problem. Water supplied through water input to the tank will be brought to the filter located under the fan, then the fan will help to blow water out to the air. After the humidifier receives humidity data from the sensors, it will calculate then decide whether to turn the fan on/off.

Physical Design

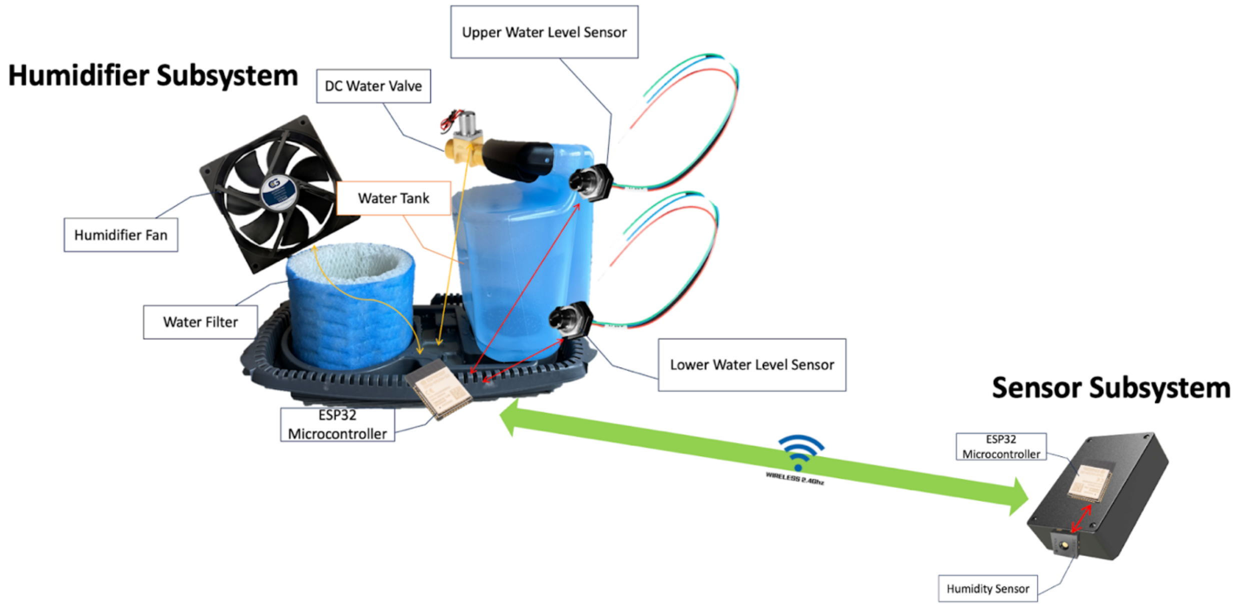

Figure 2 – Component Visual Aid

The physical design offers detailed information about inner components of the product. Two water-level sensors will be located vertically on the water tank to send fill/stop-fill signals to the ESP32 on the humidifier to open/close the DC water valve. It also shows the water filter that must be changed routinely to prevent potential hygiene risks. After reading humidity data, the humidity sensor will send its ESP32, then it will transmit the data to the humidifier ESP32 through 2.4GHz Wi-Fi.

High Level Requirements

Our project would need to achieve a multitude of high-level goals to be sufficiently complete. Some goals would include:

1. The ESP32 can read data from the humidity sensor through the use. This will be done using the ESP32 microcontroller communication through the three different sensors placed around the room. The three sensors with ESP32 microcontrollers will be mounted in their individual PCBs and will communicate with the humidifier's ESP32 microcontroller via 2.4GHz Wi-Fi with the continuous humidity readings. This will be considered as part of our total system, and the extra readings will allow us to determine the correct range at which we should turn in our humidifier fan or turn it off. This array of ESP32s will allow us to use the average humidity to detect the ideal humidity range for the fan's on/off functionality.

2. The filter irrigation system refills the water tank with water depending on readings from two water level sensors. There will be two water level sensors, with one near the top of the tank, and one on the bottom of the tank. The bottom sensor will tell the humidifier's ESP32 microcontroller that the tank needs to refill water. The ESP32 will activate the 12V DC water valve, allowing water to flow into the tank. Once the water hits the tip of the upper sensor, the sensor will signal the humidifier's ESP32 to turn off the 12V DC water valve. The measured max capacity of the water tank should be 4.16 liters. The filter should absorb around 150 milliliters ± 5 milliliters. The volume of water produced, with the humidity conditions of 35% to 50% humidity, is approximately 173 milliliters per hour. This measures up to 4.16 liters every 24 hours.

3. The humidifier should be operating between 35% and 50% humidity. When the humidity falls below 35%, the humidifier will turn on. When the humidity rises above 50%, the humidifier will turn off. This is in accordance with the EPA and FDA government guidelines for indoor humidity ranges. With the remote communication between the humidity sensor's ESP32 microcontroller and the humidifier's ESP32 microcontroller, there is an estimated 44ns response time to the activation and the deactivation of the humidifier's fans.

Design

Block Diagram

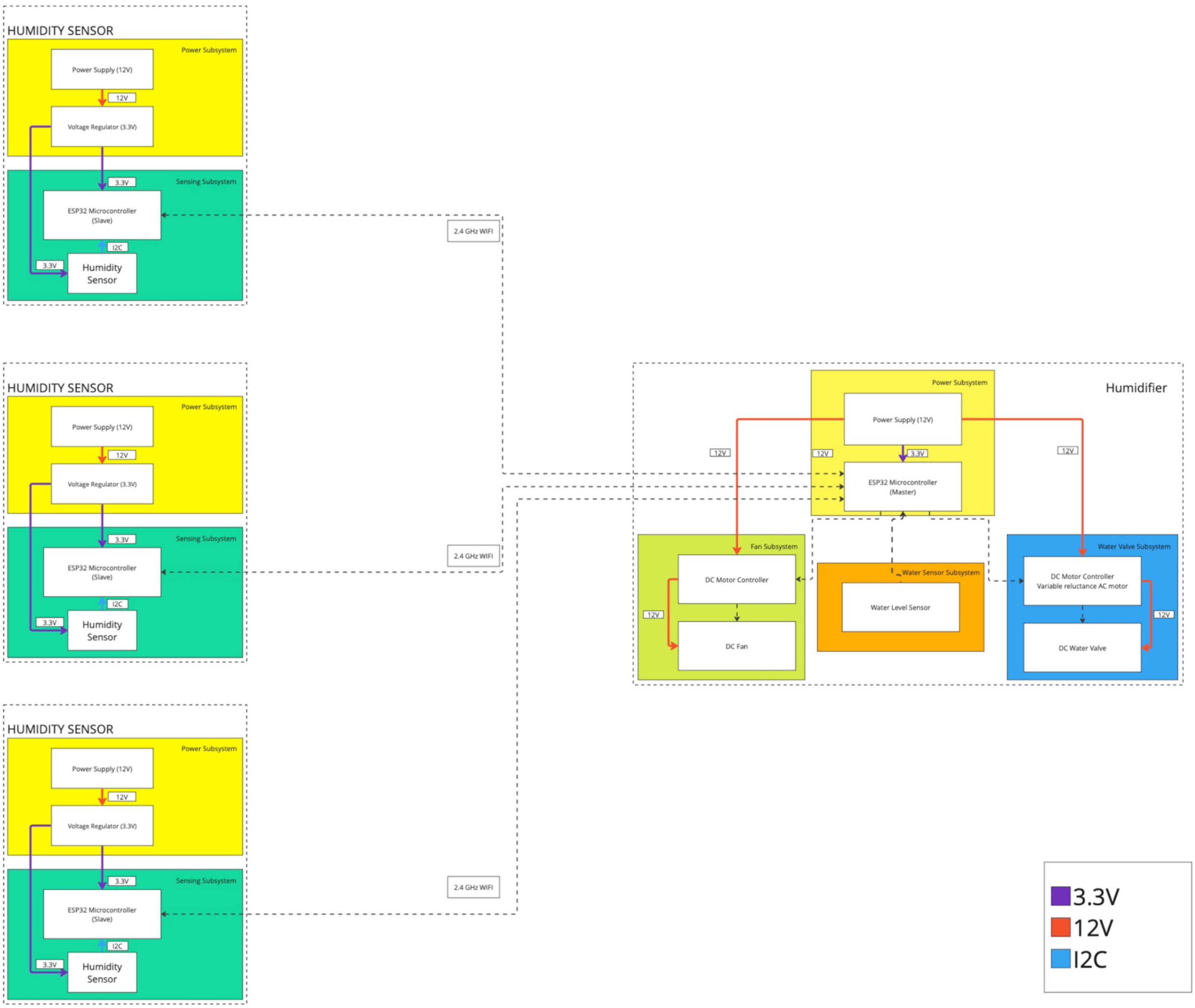

Figure 3 – Overall Block Diagram

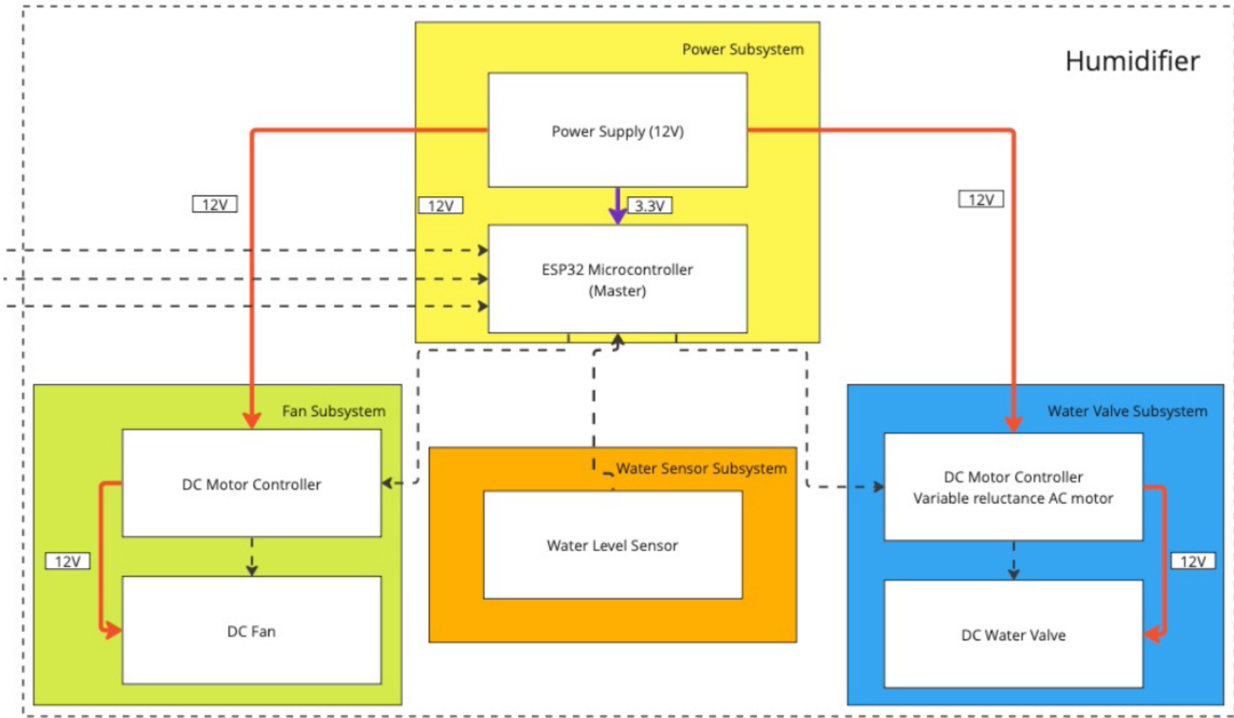

Figure 4 – Humidifier Subsystem Block Diagram (Close-up)

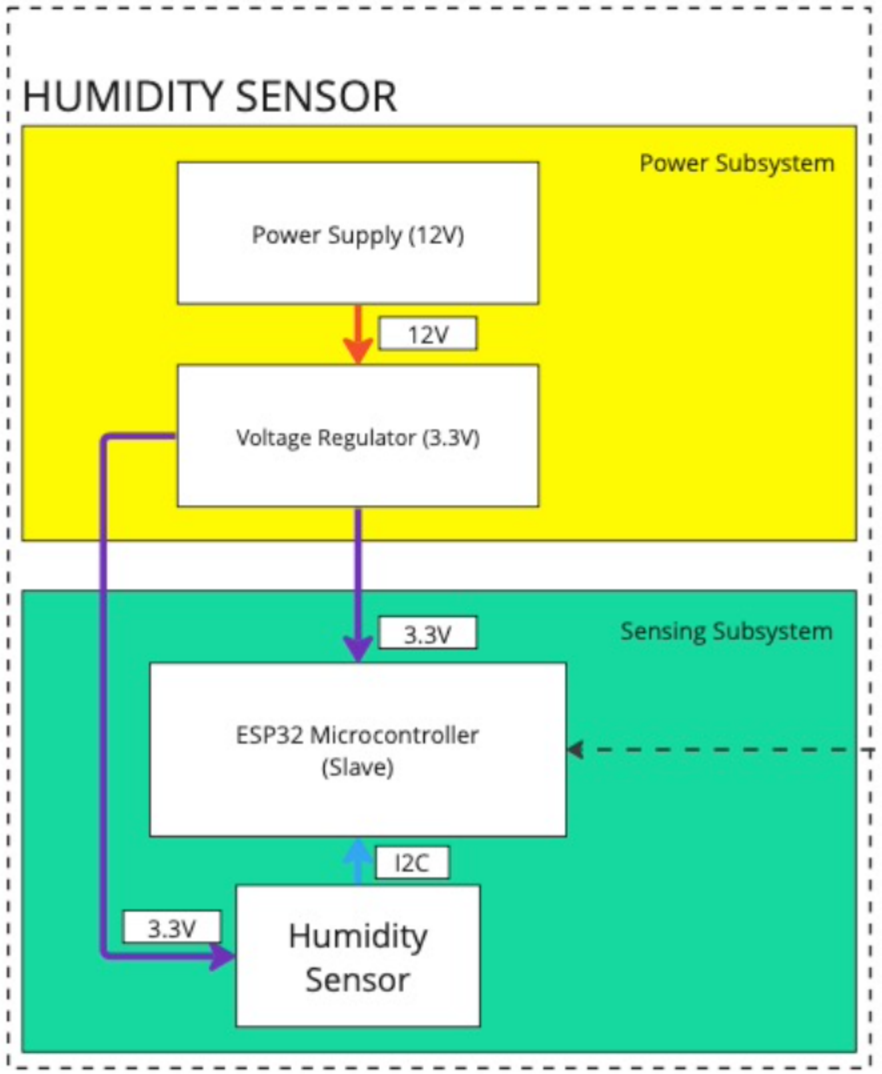

Figure 5 – Humidity Sensor Subsystem Block Diagram (Close-up)

Our block diagram is divided into two main parts. The Humidity Sensor subsystem and Humidifier subsystem. Within the humidifier subsystem consists of the power supply, fan, and water valve subsystems. The power supply subsystem regulates 12V input to 3.3V. This 3.3V is used to power the ESP32 microcontroller in the humidifier subsystem. The power subsystem also sends out 12V to the fan DC motor controller and the water valve DC motor controller. These DC motor controllers send 12V to their respective controlled motors. The ESP32 in the power subsystem also sends an on/off signal to the DC motor controllers. This tells the motor controllers to turn the DC motors on/off. Aside from the communication with the DC motor controllers, the humidifier subsystem's ESP32 microcontroller also communicates via 2.4 GHz Wi-Fi with another ESP32 microcontroller in our next subsystem, the humidity sensor subsystem.

Similar to the humidifier subsystem, the humidity sensor subsystem also consists of the power subsystem. The power supply gives a 12V DC input that gets regulated to 3.3V. The 3.3V is sent to power the ESP32 in each humidifier subsystem. At every humidifier subsystem, there is a sensor subsystem. Inside the sensor subsystem are the ESP32 microcontroller and a SHT45 humidity sensor. These two devices communicate with each other in standard-mode (100kHz) I2C. Aside from communicating with the SHT45, the ESP32 also communicates with the ESP32 in the humidifier subsystem, as mentioned previously.

Subsystem Overview / Requirements

Humidifier Subsystem:

The humidifier subsystem controls the humidity output of the humidifier. This subsystem consists of a power subsystem that converts 110V AC to a 12V DC power. The 12V DC is sent through a voltage regulator. The output is a maintained 3.3V. This 3.3V is used to power the ESP32 microcontroller. The ESP32 microcontroller is the main microcontroller of the entire system. It communicates with the ESP32 microcontroller in the humidity sensor subsystems. In this relationship, the humidifier subsystem's ESP32 is the master, and the humidity sensor subsystem's ESP32s are the slaves. Based on the incoming data, the ESP32 will determine whether to turn on the humidifier's fan or not. The motor for the fan uses 12V from the power supply and starts to rotate to run the fan. This is the main source of propulsion for our fan to produce the water into the air. When the humidifier needs more water, the ESP32 will send a signal to the water valve's motor control system to turn on the water valve to enable the connection to the source of water, such as water pipeline at home, to fill up the 4.16-liter water tank, to wet the filter for X-seconds. This motor will be used to turn the water valve on based on the amount of water needed or shut off the flow of water once the system's water level sensor detected liquid at the "max fill" line. The water valve's subsystem power source will be 12 V DC coming from the barrel jack which was converted from 110V AC from the wall supply. This power was integrated throughout the whole system through 12V to the fan as well as the water valve subsystem, as both are motors that require this voltage to operate properly.

Humidity Sensor Subsystem:

The humidity sensor subsystem detects humidity before relaying it back to the humidifier's ESP32 microcontroller. The subsystem received power from a 110V AC to 12V DC power supply. The 12V power supply is input in a barrel jack on the PCB. The 12V is sent through a voltage regulator that maintains a 3.3V output. This 3.3V is used to power the ESP32 chip in the humidity sensor subsystem as well as the SHT45 humidity sensor. The ESP32 controls the SHT45 chip. The ESP32 in each humidity sensor subsystem serves as the master and the SHT45 is the slave. The ESP32 receives data from the SHT45 via I2C. After receiving the data, the ESP32 in the humidity sensor subsystem sends the humidity sensor's data to the ESP32 in the humidifier subsystem. In this scenario, the ESP32 in the humidity sensor subsystem is the slave, while the ESP32 in the humidifier subsystem is the master.

Subsystem Requirements

Humidifier Subsystem

i. Circuit must be powered by a 12V input source, which is then regulated at 3.3V for respective components. The microcontroller and humidity sensor takes in 3.3V. The motors will be powered by 12 V to operate at the maximum capability. We will use an oscilloscope to verify the voltage that it's the desirable number with +/- 0.5%.

ii. ESP32 on the humidifier board must successfully receive wireless communication signals from remote ESP32 microcontrollers to obtain humidity data. Wi-Fi (2.4 GHz), electromagnetic waves travels nearly fast as speed of light, thus for 6 meters, it takes about 20 ns to reach from the humidity sensor to the humidifier, but it's such fast compared to desirable humidifier turn on/off update frequency, the actual data transfer speed can be mitigated and only it's enough to only verify that the humidifier receives all the data from three humidity sensors every X seconds/minutes.

iii. ESP32 should open the DC water valve when the bottom water level sensor suggests the water tank needs to be filled and close the DC water valve when the top water level sensor suggests the water level is at maximum water tank capacity. These functionalities can be verified by filling up the tank and seeing if the ESP32 on the humidifier board successfully receives signals from both sensors.

iv. ESP32 should turn on the humidifier when the average (from three sensors combined) room humidity level falls under 35% humidity and turn off the humidifier when the average room humidity level rises above 50% humidity. These functionalities can be verified if ESP32 on the humidifier board can correctly control the fan motor when it gets corresponding data from the sensor ESPs.

v. This subsystem will contribute to the overall design dictated by the high-level requirements by acting as the central subsystem that will control the main microcontroller, along with the 12V motors that drive the water supply valve and the water evaporation fan. Then the readings from the humidity sensor subsystem will communicate through 2.4 GHz Wi-Fi to our main microcontroller to tell the fan to turn on or not based on an ideal range of between 35% and 50% humidity as mentioned in the requirements. Then the microcontroller will send the data to moisture the air, with approximately 83 mililiters per hour. The water sensor will also control our water filling of up to 4.16 liters to communicate to the microcontroller whether we have reached max capacity or not.

RV-Table – Humidifier Subsystem

Requirements	Verification
• **Provide 3.3V +/- 0.5% from an input voltage of 12V +/- 0.10% DC source** • **Thermal stability maintains below 120’C**	• Use multimeter to test and take average voltage output over 1-hour to test stability • Monitor heat dissipation with thermal laser gun over an average of 1-hour use time, expected +/- 5’C under load
• **ESP32 communication with remote sensor ESP32 utilizing 2.4GHz Wi-Fi**	• Verify functionality of ESP32 Wi-Fi component by creating an access point, and accessing access point to remotely turn on LED lights • Verify communication between two ESP32 boards by sending command to turn on LED light from one ESP32 board • Verify thermal performance of ESP32 chip during operation by probing with a laser thermal gun
• **ESP32 control of activation and deactivation of water valve** • **ESP32 control of activation and deactivation of fan**	• Verify functionality of ESP32 communication with DC motor controls by sending activation signal to water valve/fan DC motor controller, and when signal is received, DC motor controllers output a lit LED light • Probe the output voltage of the DC motor controllers to make sure it is 12V +/- 0.5%
• **ESP32 detects signals from two water-level sensors**	• Contact/No Contact water to the water-level sensors and check through Arduino IDE to see if ESP32 receives contact/no contact signal from the two water-level sensors • Water level sensor lights up LED when in contact with water, and LED is not lit when not in contact with water

Humidity Sensor Subsystem

i. Circuit must be powered by a 12V input source, which is then regulated at 3.3V for respective components. We will use an oscilloscope to verify the voltage that it's the desirable number with +/- 0.5%.
ii. ESP32 board should be able to communicate with SHT45 humidity sensor with I2C. Communication between the humidity sensor and the ESP32 on the same PCB through I2C is fast enough compared to how frequently the humidity sensor measures the humidity, which varies from 1.3 to 8.3 ms depending on the noise and energy consumption. It's enough to verify that the ESP32 in the humidity sensor receives all the data from its sensor every X seconds/minutes.
iii. ESP32 board should be able to communicate with ESP32 in humidifier, relaying the SHT45's data to it.
iv. ESP 32 will communicate with the humidity sensor as same frequency.
v. These functionalities can be verified by giving different humidity conditions to the humidity sensor.
vi. This subsystem will be connected to the overall design through measuring the correct humidity in a certain area, and with 3 different humidity sensors, there would be a range that is created that the central microcontroller would read because the data is being sent from each individual microcontroller per sensor through Wi-Fi, with an expected response time of 44ns through communication of the I2C protocol.

RV-Table – Humidity Sensor Subsystem

Requirements	Verification
• **Provide 3.3V +/- 0.5% from an input voltage of 12V +/- 0.10% DC source** • **Thermal stability maintains below 120’C**	• Use multimeter to test and take average voltage output over 1-hour to test stability • Monitor heat dissipation with thermal laser gun over an average of 1-hour use time, expected +/- 5’C under load
• **ESP32 communication with humidity sensor receiving humidity data at least once in a minute** • **Humidity sensor measuring humidity data at least between 25% - 55%**	• Verify communication between ESP32 and humidity sensor Arduino IDE by checking humidity data ESP32 receives • Compare the humidity data received by ESP32 to the commercial humidity sensor and check if the humidity is within +/- 3% • Make the air dry/moist to see if the sensor can measure humidity between 25% - 55% • Compare humidity reading with reference humidity reader

Tolerance Analysis / Potential Error

A significant aspect of our design that could pose a major risk to our project is the strength in the Wi-Fi signals being used to communicate between the sensors. Our project majorly depends on multiple microcontrollers in multiple sensors across many different areas in a room using Wi-Fi to connect to the central microcontroller. If any of these signals were interrupted, it would cause an impactful change in the way our system functions. Our completion of the project would be deemed nearly impossible without a proper wireless communication protocol, whether it be Bluetooth or Wi-Fi. For example, Wi-Fi is essentially a EM wave that operates through the medium of air containing the data we need, and can be more accurately defined using sets of Maxwell’s equations, most notably the Helmholtz equation that gives us the equation:

$$E\nabla^2 + \frac{E(k^2)}{n^2} = 0$$

which are the electric fields mapped based on the angular wavenumber k and the material index n (Hecht). Below is a sample model taken from a reference site, Freefem, that will model what the signals will look like at a certain point in the area.

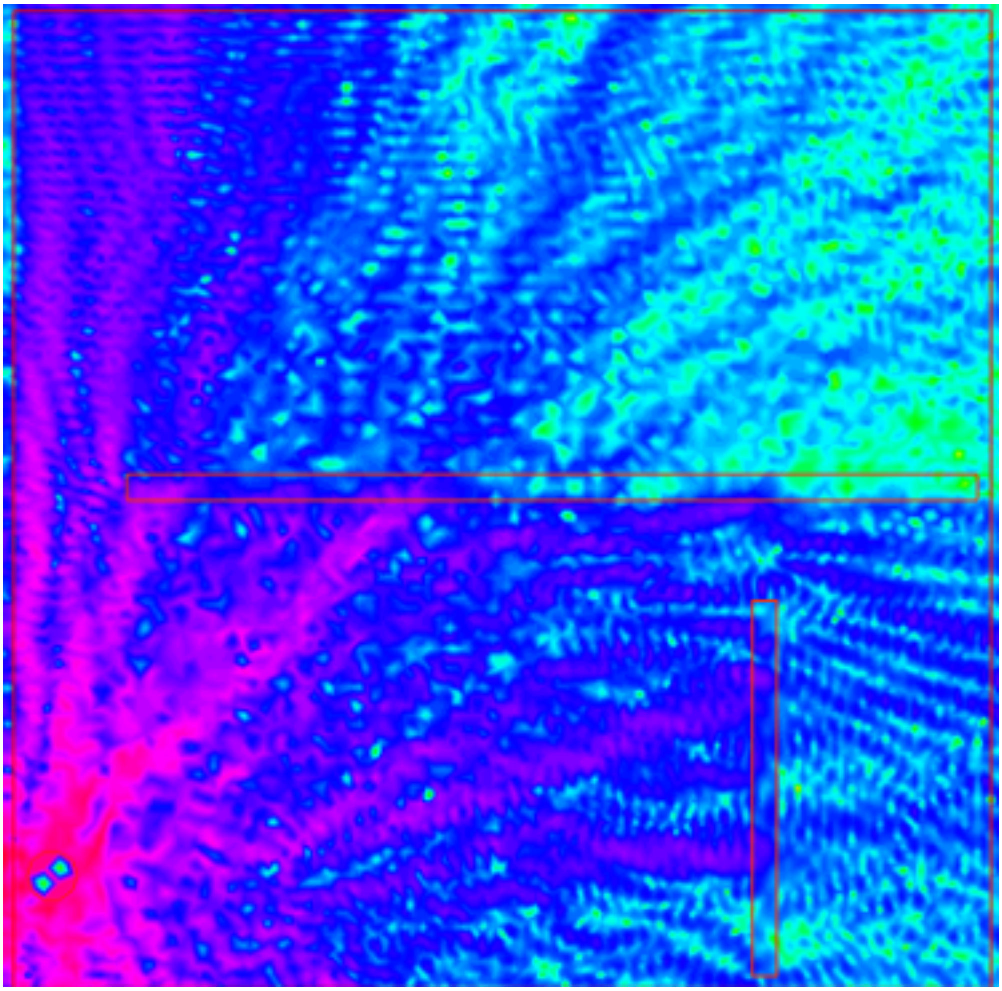

Figure 6: Signal Strength Plot

In *figure 6*, it shows a point where the electromagnetic signal is being read and the relative strength of it compared to the waves across the rest of the room. This will be how our sensors operate, with the signals being the strongest at receiving/interacting ends and having the most possible interference in between transmissions. In our respective setup, we will use example calculations to model this. For example, wavelengths of our WiFi signals which will be used to communicate are approximately 12 centimeters give or take, given a 2.4 GHz WiFi default in our system. So the angular wavenumber:

$$k = 2\frac{\pi}{\lambda}$$

which given our lambda wavelength value, will give us $0.52\ \frac{rad}{cm}$. As for material index n, the value should just be the material index of air, which is the constant of 1. Given this, we have taken E as an average of $50\ \frac{V}{m}$ for an electric field value inside a home. So our electric field gradient value will be equal to:

$$50 \times\ 0.52^2 = 13.52\ \frac{v}{m^3}$$

This value will indicate the electric field strength in a specific area, which matches our signal strength plot example above. The field will obviously be different at random areas in the room due to interference from various radio sources, such as other electronics or even your own body as a source. To mitigate this issue, we will test if signals are possible to fully transmit all necessary data given multiple sensor points. If information is not fully available because of the strength or interference of other signals, we will act to measure where exactly the field is weak and create optimal positions for our sensors to generate the most effective waves. Another solution to this problem could be examining the materials that have interference potential such as metal or concrete, and either removing or replacing those materials in highly affected areas. Trade-offs to our solutions would be that optimal positions for signal strength might not be optimal positions for humidifier strength. For example, if our sensors were positioned at 30, 45 and 180 degrees in a circular pattern in a rectangular room, there would be nearly no sensing of the humidity in the 270 degrees area. This would be sub-optimal because we would essentially have 2 sensors instead of 3 near evenly spaced ones, despite the connection being very reliable.

Cost and Schedule

Cost Analysis

Parts and Materials

In our project, we have an estimated breakdown of the parts and materials needed in our final product. The part name, part number, number required per system, and the cost per system is provided. Each system consists of one humidifier subsystem, as well as three humidity sensor subsystems.

	Description	Part Number	Unit Price	#/System	$/System
1	ESP32-S3-WROOM-1-N8	ESP32-S3-WROOM-1-N8	$3.20	4	$12.80
2	Optomax Liquid Level Sensor	LLC200D3SH-LLPK1	$24.95	2	$49.90
3	Humidity Sensor	SHT45-AD1B-R3	$7.18	3	$21.54
4	Barrel Jack	PJ-202AH	$0.69	4	$2.76
5	AC to DC Converter	WSU120-2000	$15.12	4	$60.48
6	Solenoid Valve, DC 12V 1/2" Water Flow	Walfront36atn8w5ph	$18.44	1	$18.44
7	Humidifier Filter	B07V4T5477	$15.06	1	$15.06
8	10uF Decoupling Capacitor 0805 SMD	GMC21X7R106K10NT	$0.03	18	$0.45
9	0.1uF Decoupling Capacitor 0805 SMD	0805B104K101CC	$0.05	18	$0.90
10	120 Ohm Resistor 0805 SMD	738-RMCF0805FT120RCT-ND	$0.62	2	$1.24
11	500 Ohm Resistor 0805 SMD	CRCW08050000Z0EA	$0.10	1	$0.10
12	200 Ohm Resistor 0805 SMD	AC0805FR-07200RL	$0.10	1	$0.10
13	Adjustable Voltage Regulator	LD1117SC-R	$0.78	2	$1.56
14	1x3 Horizontally Mounted Socket (F)	SSW-103-02-G-S-RA	$0.96	2	$1.92
15	1x2 Horizontally Mounted Socket (F)	SSW-102-02-G-S-RA	$0.71	2	$1.42
16	Green Diode SMD 0805	350-2044-1-ND	$0.10	4	$0.40
17	Thermal Stable Solder Paste	TS391AX-ND	$15.95	1	$15.95
18	Click Button	2223-TS02-66-60-BK-100-LCR-D-ND	$0.10	1	$0.10
				Total =	$205.12

Table 1 – Estimated Product Cost

Hours of Development and Labor

Our project consists of multiple roles that need to be completed. We have an estimated 373.5 hours total between three group members on the project. Below is a breakdown of our estimated hours per person on each category.

	Member		
Category	Andrew	Woojin	Jalen
Circuit Design	21	15	12
PCB Design	25	10	14
Quality Check	6	1.5	12
Software Engineering	10	32	2
Soldering	1	2	10
Design Troubleshoot	12	12	8
Documentation	20	20	20
Meetings	35	30	30
Miscellaneous	5	5	3
Total Hours	135	127.5	111

Table 2 – Estimated Hours

With mixed electrical engineers and computer engineers in our group, we all have different hourly wages. Using the average UIUC ECE computer engineering published salary in 2021-2022 ($109,176.00), using 2080 work hours in a work year, we can expect an average hourly salary of $52.49.

$$\frac{\$109{,}176.00}{2080\ hours} = \$52.49\ per\ hour$$

$$(135 hours + 127.5 hours + 111 hours) \times\ \$52.49 = \$19{,}605.02$$

Summing up the total hours in the table above and multiplying it by the average salary, we get the total cost for labor to be: $19,605.02 for this project.

Aside from our own hours of labor, the machine shop estimates 7 hours of work to modify our humidifier with some parts included in *Table 1*. The machine-shop technician's hourly cost is not public, so we will not include the cost in the total cost of development above.

Schedule

1. Week 6 (02/19)
    a. Design Document
    b. Proposal Revision
    c. I2C communication between ESP32 and Humidifier
    d. 2.4GHz Wi-Fi communication between two ESP32 boards
    e. Give humidifier to machine shop
    f. Begin PCB Design for humidity sensor subsystem
2. Week 7 (02/26)
    a. Design PCB for humidity sensor subsystem
    b. Debug potential issues with PCB
    c. Interface water-level sensor with humidifier ESP32
    d. Control 12V DC water valve with ESP32
    e. Control humidifier fan with ESP32
    f. Potentially place first PCB order (individual)
3. Week 8 (03/04)
    a. Order PCB for humidity sensor subsystem (with class)
    b. Order electronic water valve
    c. Order second Adafruit water level sensor
    d. Design PCB for humidifier subsystem
    e. Debug potential issues with PCB
    f. Potentially send humidity sensor subsystem PCB to machine shop
    g. Potentially solder SMD components on humidity sensor PCB
4. Week 9 (03/11)
    a. Finish humidifier subsystem PCB design
5. Week 10 (03/18)
    a. Solder PCB for humidity sensor PCB
    b. Send humidity sensor PCB to machine shop
    c. Order PCB for humidifier
6. Week 11 (03/25)
    a. Reorder revised PCBs
    b. Solder PCB for humidifier subsystem
    c. Potentially put together humidifier
7. Week 12 (04/01)
    a. Begin writing final paper
    b. Finalize PCBs
    c. Send remaining updates to machine shop
8. Week 13 (04/08)
    a. Finalize product for presentation
    b. Begin writing final paper
    c. Last minute debugging
9. Week 14 (04/15)
    a. Team contract fulfillment
    b. Mock demonstration
    c. Finalize product for presentation
    d. Continue writing final paper
10. Week 15 (04/22)
    a. Final Demonstration
    b. Mock Presentation
    c. Final Presentation Preparation

    d. Continue writing final paper

11. Week 16 (04/29)
    a. Final Presentation Preparation
    b. Final Presentation
    c. Final Papers
    d. Lab Notebook
    e. Lab Checkout

Ethics and Safety

For our initial examination of ethics, we were able to justify our project using the IEEE code of ethics. As students and soon-to-be workers, we know that whatever we do will have an impact on the world, primarily through the products and services we work on. Therefore, we must make sure to set some ethical guidelines that match with the code.

Ethics

i. To uphold the highest standards of integrity, responsible behavior, and ethical conduct in professional activities. (IEEE, 2020)

   a. We will keep an open-minded state to each other's opinions. Utilizing everyone's opinions, we will try to find the most ideal solution to our problems.

   b. We will keep the integrity of our research to the highest degree. All data collected will be the truthful original and will not be "tuned" in any way.

   c. We will keep the safety of our members and those working around us to the utmost highest standard.

   d. We will adhere to the rules and regulations of the University of Illinois and ECE445's lab conduct rules.

   e. Constructive advice and criticisms will be accepted, and any forms of bribery will not be tolerated.

ii. To treat all persons fairly and with respect, to not engage in harassment or discrimination, and to avoid injuring others. (IEEE, 2020)

   a. We will treat all members within and outside our group with respect on physical and non-physical standpoints.

   b. Sexual harassment is not tolerated and will be reported to appropriate parties.

iii. To strive to ensure this code is upheld by colleagues and co-workers. (IEEE, 2020)

   a. We will strive to uphold this code of conduct provided from IEEE and in our team contract. Any behavior outside the rules will be reported to the correct authorities.

iv. To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, to be honest and realistic in stating claims or estimates based on available data … (IEEE, 2020)

   a. A central ethical concern includes directly impacting humans as a first resort without being able to test otherwise. For example, for most scientific products, humans are the final stage of a trial in a product because all previous stages have been cleared or given approval through uses of inanimate objects or lab animals. Our problem is generally a comfort-based one, and so the general ethical concern would be the first stage exposure to humans since there would not be a test trial. If our main issue is comfort, we will never know how it feels until actual humans have tested it. Therefore, as one of the IEEE Code of Ethics indicates, we will seek and accept any technical criticisms from TA, instructors, and machine shop, be realistic and responsible to revise the problem until we reach agreement. On top of that we will measure the humidity carefully and regulate with the correct amounts of moisture, making sure that we only deviate from the average quantity only once we have received indication that it is comfortable.

Safety

In this section, we consider every type of the most common situations that could occur when using our automatic humidifier product.

i. A safety concern would be the wall plug power. This is 120 V AC power typically, and so to prevent electrocution and shock issues, we would need to be aware of our surroundings and to connect the plug into the wall without any external connections to the outlet interrupting it in any way, such as holding loose wires or the metal section of the plug when you are plugging it into the wall.

ii. A safety concern is the issue of potential bacteria and mold. Our automatic humidifier should get rid of this, but this issue could severely impact the humidifier if there turns out to be traces of something contagious that can be widespread with the device. The dangers of something airborne that could get into your lungs would be catastrophic if not regulated correctly.

iii. Another possible safety issue could be the moving parts of a fan that could injure you if you try to touch the mechanical components. The fan's motor will likely be insulated from external touch, but there is always a chance that something breaks and another chance that the moving part accidentally comes in contact with someone's finger or something easily mistouched, leading to a higher chance of injury.

iv. A common matter that arises as well is if there is a connected cable that meets water. Our humidifier will likely be powered through a wired connection, and any water that comes into contact either from the humidifier or an external water source will damage the system, or even worse, cause physical harm to someone. This safety issue would need to be fixed through epoxy or some type of non-porous material insulating the device and its connections.

v. Dangerous chemicals are a final hazard that could be caused intentionally or by accident. Chemicals could be an issue if put inside the supply which normally contains water. If a dangerous fluid was put in the supply on purpose, then the humidifier might act as normal, sensing the humidity levels and then dispersing a life-threatening liquid instead of water. This would cause great harm and even possibly death if misused, so it is of utmost importance that the person using the device is careful and is sensible.

Otherwise, the designers are not subject to responsibility and take no claim in the misuse of the product, which in that case, the user should refer to the instructions provided.

References

[1] Hecht, F. (n.d.). WiFi Propagation . WIFI propagation. https://doc.freefem.org/tutorials/wifiPropagation.html

[2] Marwell, M. (2018, August 9). Issues with the I^2C (inter-ic) bus and how to solve them. *DigiKey*. https://www.digikey.com/en/articles/issues-with-the-i2c-bus-and-how-to-solve-them

[3] Sensirion. (2023). SHT4x 4th Generation, High-Accuracy, Ultra-Low-Power, 16-bit Relative Humidity and Temperature Sensor. In *Sensirion Official Documentation* (Issue 6). Sensirion.

[4] United States Environmental Protection Agency. (2014, August 28). *Care for your air: A guide to indoor air quality*. US EPA. https://www.epa.gov/indoor-air-quality-iaq/care-your-air-guide-indoor-air-quality

[5] Espressif. (2023). *ESP32-S3-WROOM-1 ESP32-S3-WROOM-1U Datasheet*. Espressif. https://www.espressif.com/sites/default/files/documentation/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf

[6] Metageek. (2024). *Wi-Fi and non wi-fi interference*. MetaGeek. https://www.metageek.com/training/resources/wifi-and-non-wifi-interference/

[7] Arundel, A. V., Sterling, E. M., Biggin, J. H., & Sterling, T. D. (1986). Indirect health effects of relative humidity in indoor environments. *Environmental Health Perspectives*, *65*, 351–361. https://doi.org/10.1289/ehp.8665351

[8] Perrone, V., Veronesi, C., Giacomini, E., Citraro, R., Dell'Orco, S., Lena, F., Paciello, A., Resta, A. M., Nica, M., Ritrovato, D., & Degli Esposti, L. (2022). The epidemiology, treatment patterns and economic burden of different phenotypes of multiple sclerosis in Italy: Relapsing-Remitting multiple sclerosis and secondary progressive multiple sclerosis. *Clinical Epidemiology*, *14*, 1327–1337. https://doi.org/10.2147/CLEP.S376005

[9] IEEE. (2020). *IEEE code of ethics*. IEEE Code of Ethics. https://www.ieee.org/about/corporate/governance/p7-8.html

[10] SENSIRION. (2023, January). 4th Generation, High-Accuracy, Ultra-Low-Power, 16-bit Relative Humidity and Temperature Sensor. Cdn-Shop.Adafruit.Com. https://cdn-shop.adafruit.com/product-files/5665/5665_Datasheet_SHT4x.pdf

[11] University of Illinois. (2022). *Salary averages*. UIUC. https://ece.illinois.edu/admissions/why-ece/salary-averages