**Cooling Humidifier umbrella for lung cancer and multiple sclerosis patients**

**(draft)**

Christopher R Aitken, Jaime Martinez Diaz, Jason J Schwarzwalder,

Christopher J Barry, Yizhe Wang,Yunke Chen

**Introduction**

Executive summary

Many people have respiratory difficulties that make it hard to breathe in hotter weather. We would like to alleviate this issue by creating an umbrella that has a built-in humidifier to provide cool mist to the user so they can have more freedom without worrying about possible issues breathing. The umbrella will have an integrated design that will include: A cooling system, humidity and temperature detection system, humidifier, control unit, and a power supply with possible power generation by using solar panels. Some goals are not only to keep the cost down, but to also keep the umbrella light and easy to use.

Problem statement

During the summer, many people that have respiratory difficulties cannot go outside due to the temperature and humidity. They instead are hunkered down indoors near a humidifier, unable to go anywhere. This prevents people from living their day to day lives and stops people from doing something as simple as walking their dog. Our customer would like an easy to use, portable humidifier that would provide cool air in a design of something similar to an umbrella. In the current marketplace, there is no portable cold humidifier that can reliably keep a steady humidity rating and temperature. Simple implementation of humidifiers onto umbrellas have been manufactured by previous companies but lack proper embedded humidity and temperature sensing. Our humidifier umbrella would fill in a niche market of an affordable portable respiratory aid. Target consumers would be lung cancer and multiple sclerosis patients, but broader consumers would want a device to keep them cooler during the hot summer days

**Requirements specification**

1. Input/output Requirements

(1) A button input will toggle the operation of the umbrella

(2) Battery level of the device shall be indicated using an RGB LED.

(3) The humidity level will be detected by a humidity sensor.

(4) The temperature will be detected by a temperature sensor.

2. Functional Requirements

(1) The device will be an umbrella-integrated cold humidifier.

(2) The device will shade an area with lowered temperature to 65-70 °F.

(3) The device will provide humidified air within 30% - 60% relative humidity for the user.

(4) The device will function 1~2 hours on a single battery charge.

(5) The device will provide all-direction cooling and humidifying in the area when in normal mode.

(6) The device will run a low-power mode when the battery is low.

(7) The device will provide single-direction cooling and humidifying when it is in low-power mode.

3. External Interface requirements

(1) The water tank will be easily refillable, and the refill process will take less than 1 minute for the user.

(2) External water tank should be attached to the device to provide extra water.

(3) The battery will be rechargeable using USB adapters.

4. Technology and System-Wide Requirements

(1) The umbrella will be portable and within the size of 51’’ arc and 36’’ length.

(2) The weight of the device will be within 1.5 lb, which means being light enough for people to hold for 1~2 hours.

(3) The device will be integrated with solar panels.

(4) The device will use a Raspberry Pi Zero as central processing unit.

(5) The cost of the device will be under $100.

(6) The device will be durable for 100 times or 150 hours of daily use.

(7) The device will function in an outdoor temperature from 65 °F to 70 °F (18.3 °C to 21.1 °C).

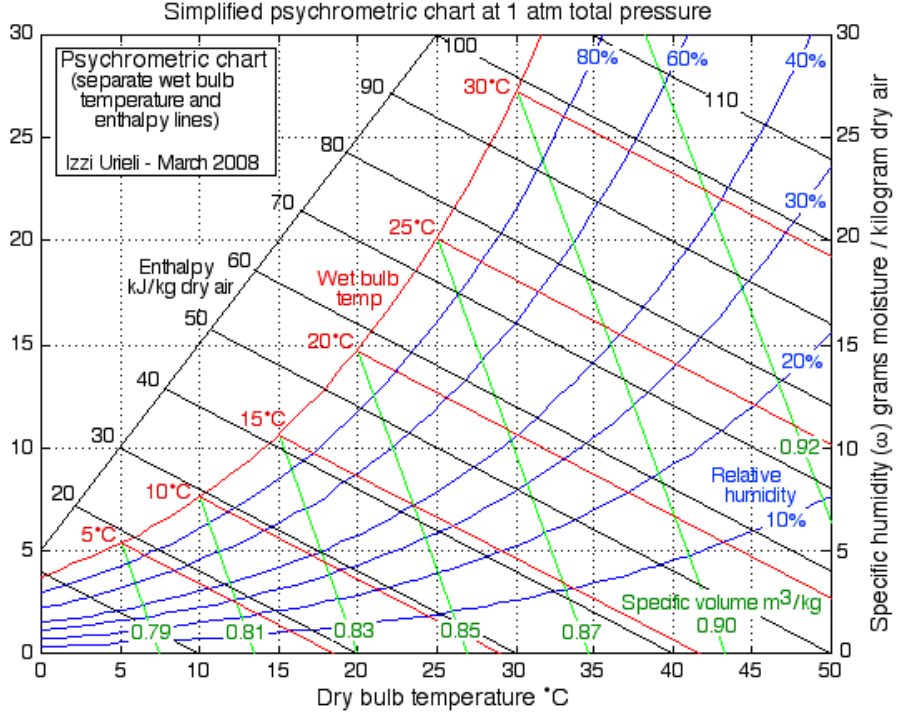
(8) The device will function in a water temperature from 32 °F to 95 °F (0 °C to 35 °C).

(9) The operation noise of the device will be less than 40 dB.

(10) The device will be integrated with an air fan.

**Background knowledge/phenomenology (math, physics, EE, etc.) used to derive a solution/design**

1.Evaporative cooling is used to lower the temperature underneath the umbrella. Evaporative cooling reduces the surrounding temperature through the evaporation of water in the system. Evaporation of water droplets lowers the surrounding temperature while increasing the relative humidity. A psychrometric chart is typically used to visually see the correlation of the relative humidity with the dry bulb temperature.

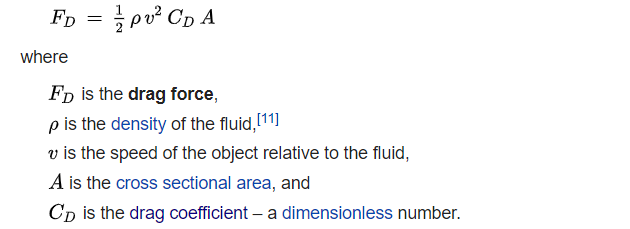


Approximation of temperature drop would depend on the dry and wet bulb temperature of the day. A typical summer day can be approximated with the following equation:

Where is the dry bulb temperature, is the wet bulb temperature, is the evaporative rate, and is the output temperature. The reduced temperature of the dry bulb will increase the relative humidity due to the reduction in temperature, as can be observed on the psychrometric chart.

2.Drag force calculation:

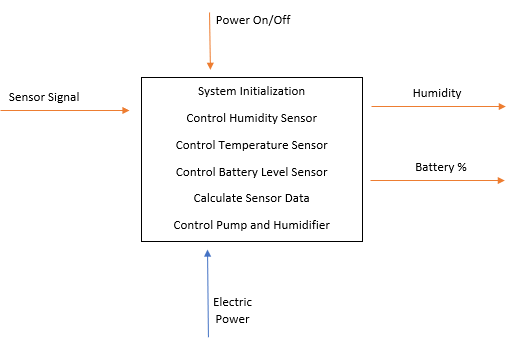
In fluid dynamics, drag (sometimes called air resistance, a type of friction, or fluid resistance, another type of friction or fluid friction) is a force acting opposite to the relative motion of any object moving with respect to a surrounding fluid.

****

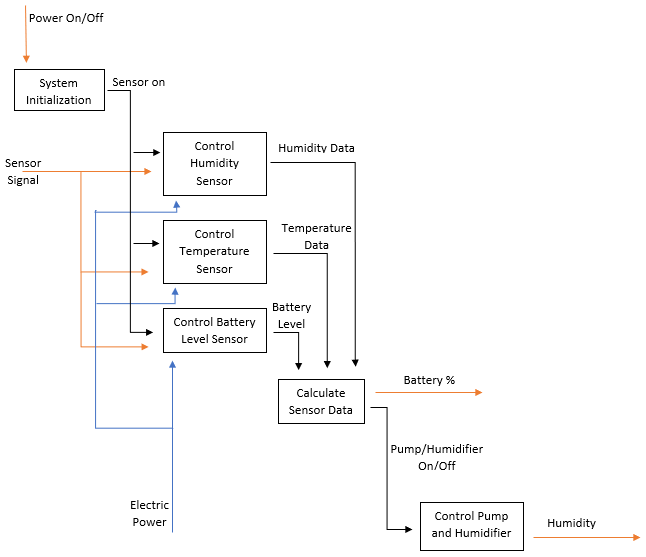
**Detailed design**

1) All levels of your design down to circuit schematics, state diagrams, and algorithm flowcharts

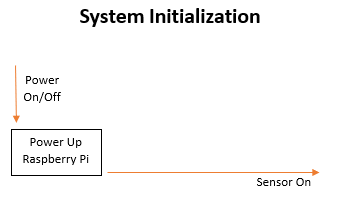
* Functional decomposition
  + Level 0

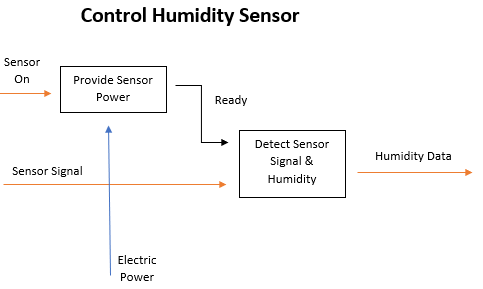


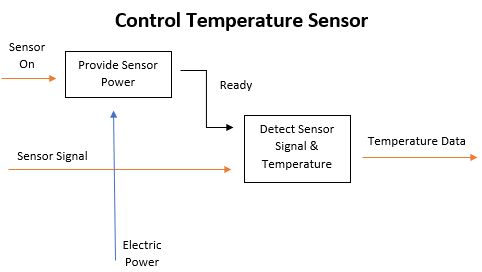
* + Level 1

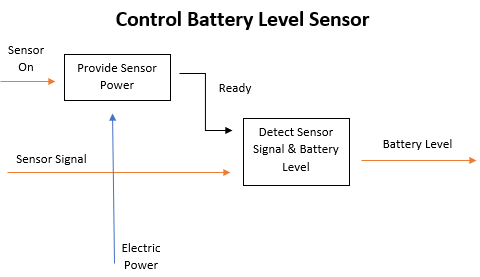


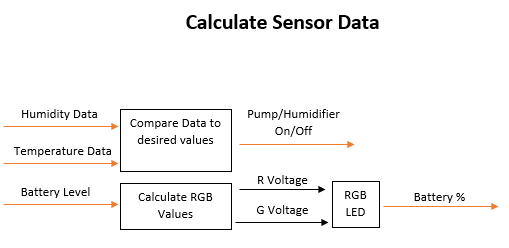
* + Level 2











2) Detailed description of components and interfaces

Humidity and temperature detection

The device needs to be able to detect the humidity and temperature of the surrounding area. The part of design requires a high sensitivity and veracity.

Approach: We will be using a DTH22 which is a temperature and humidity sensor []. This sensor can operate with a 3.3-6v DC power supply and within -40~80 °C. The sensor can only send data every 2 seconds, but this should not become an issue in the final design. The sensor costs $15 []. This sensor is both precise and cheap.

Data receiving and system control

In order to make all the desired functions work, the control system is required to receive data stably and effectively. Meanwhile, the size of the whole control system shall be small enough to be integrated into the umbrella structure.

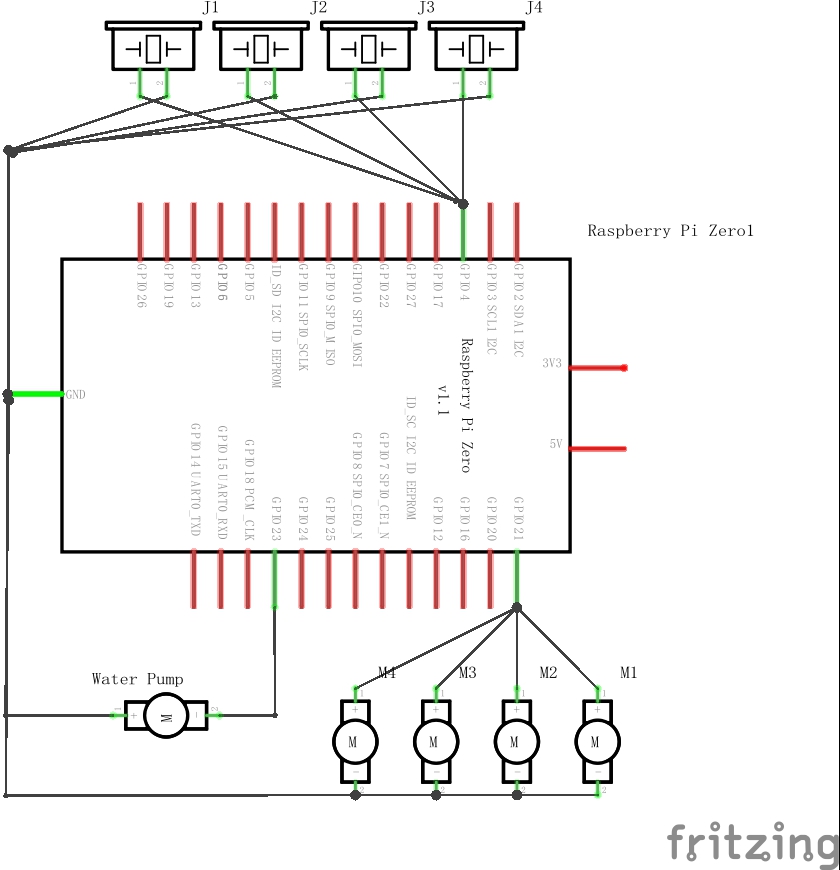
Approach: The Raspberry Pi Zero would be an ideal control board to handle the multiple functions of the umbrella []. The DTH22 is directly compatible with the Pi zero, as it only requires some libraries to be installed [].

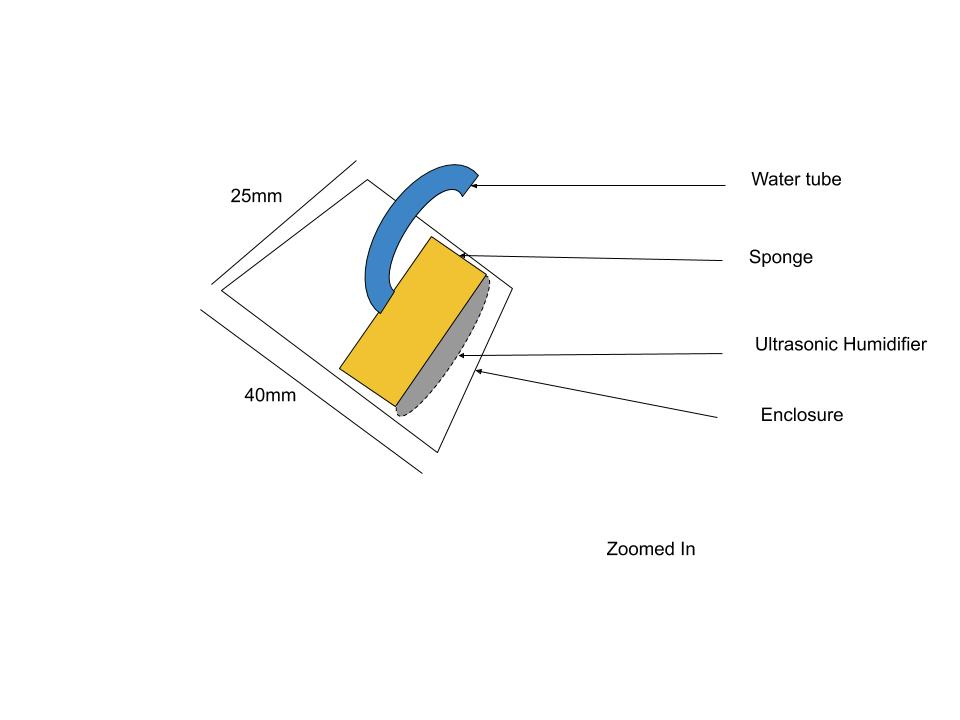
Cooling system

There is a rigid demand that the device be integrated with a cooling system. The design of the cooling system needs to meet the required temperature range that the customer specified.

Approach: The material of the umbrella cover will have reasonable defensibility of heat and ultraviolet rays. Two or more mini humidifiers will be integrated into the umbrella to spray a mist to reduce the surrounding temperature. There are two potential pumps to use. The Anself can pump 240 L/H and the Vipe can pump 120 L/H []. The Anself is the bigger and more powerful pump, but reviews state that it is notorious for burning out if it is run dry. The Vipe is reviewed better overall but it's a weaker pump []. We will be choosing the Vipe due to its lower costs and the reviews state that it is more reliable.

Cooling**:**



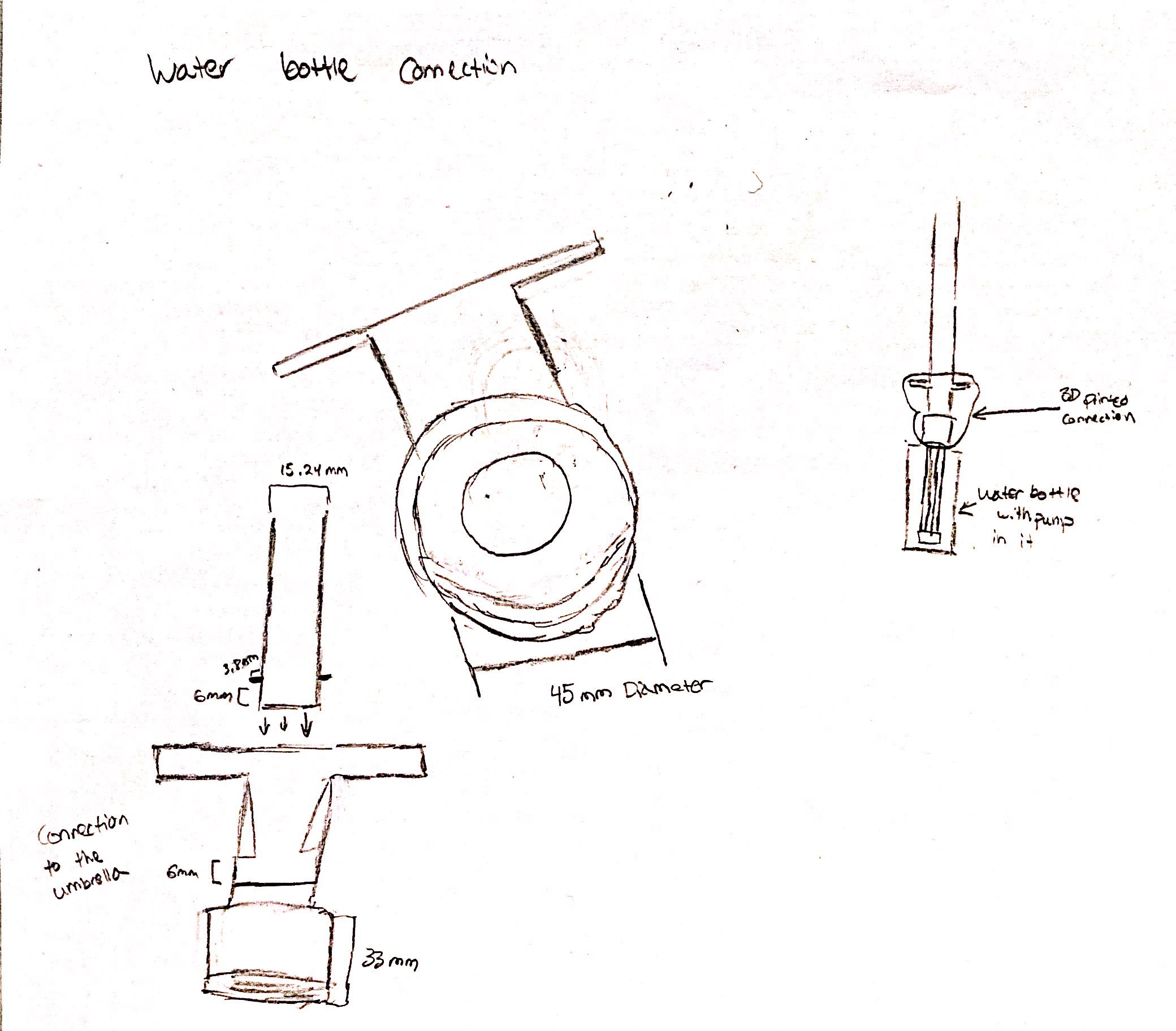


Umbrella construction

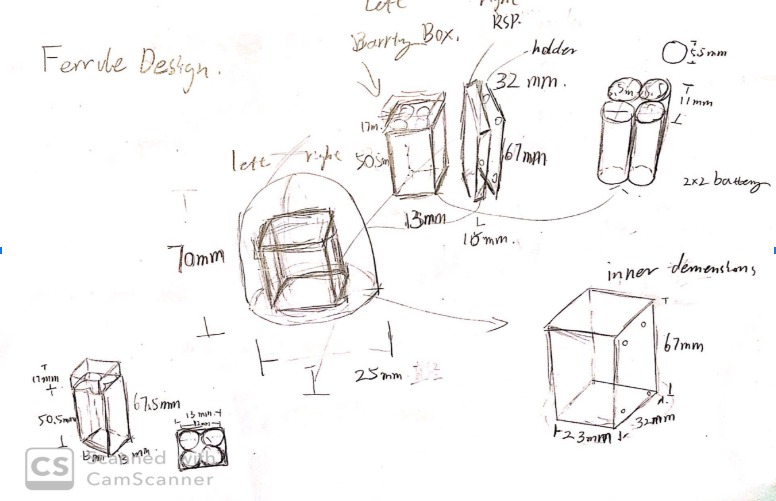
The whole umbrella will be highly integrated. The factors of weight, size and durability need to be considered.

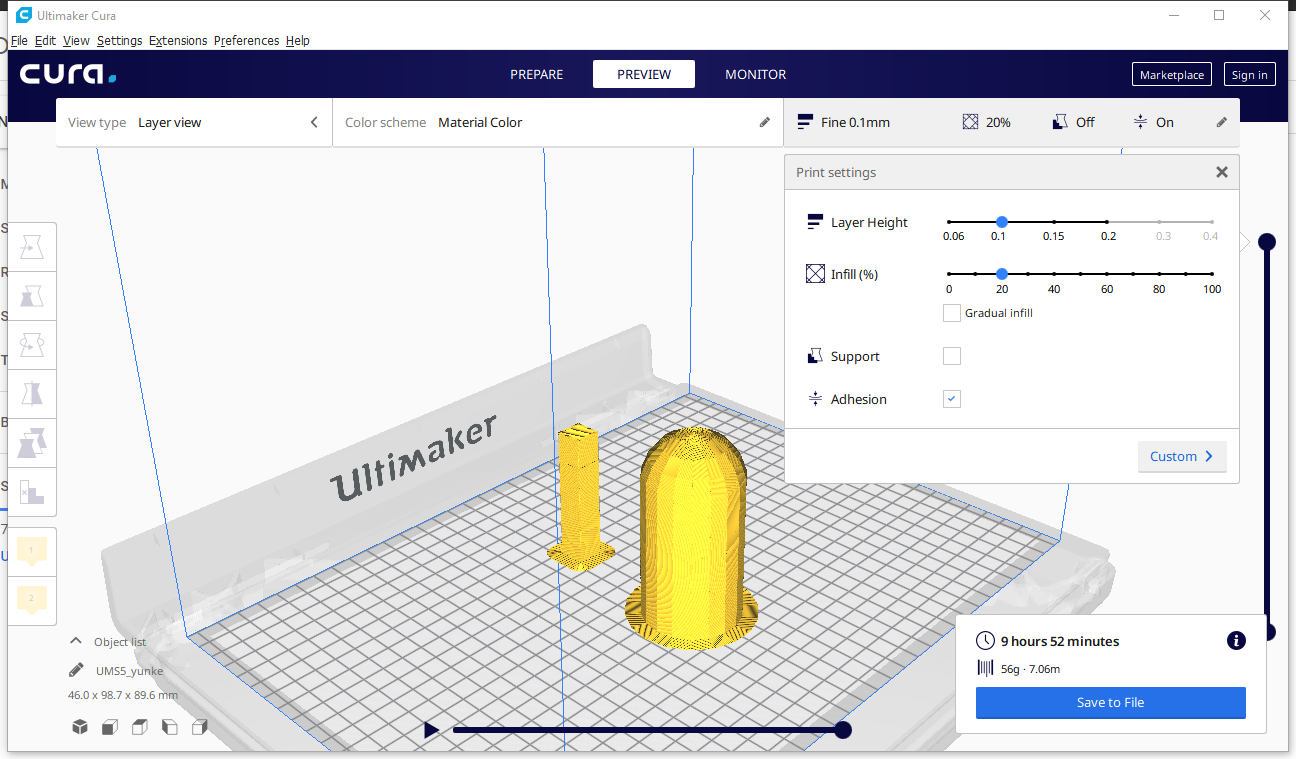
Approach: The umbrella will use bubble style, which has about a 51” canopy size and manual opening. The advantage of the bubble umbrella is that it can cover the upper body more effectively than the traditional umbrella. In addition, a bubble umbrella would be the best shape during windy conditions. It is considered as a reasonable shape to ensure the inner area of umbrella stable and let the mist spray naturally down. We will use 3D printing to build some of the structure of the umbrella as well as housing for the electrical components. The filament of the 3D printer has extrusion temperature from 210F to 356F. We may need to use a PCB design to create a waterproof housing for our circuit.

Design for the water bottle connection



Design for the Ferrule:

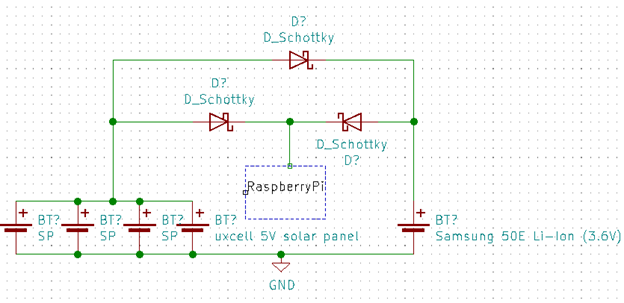


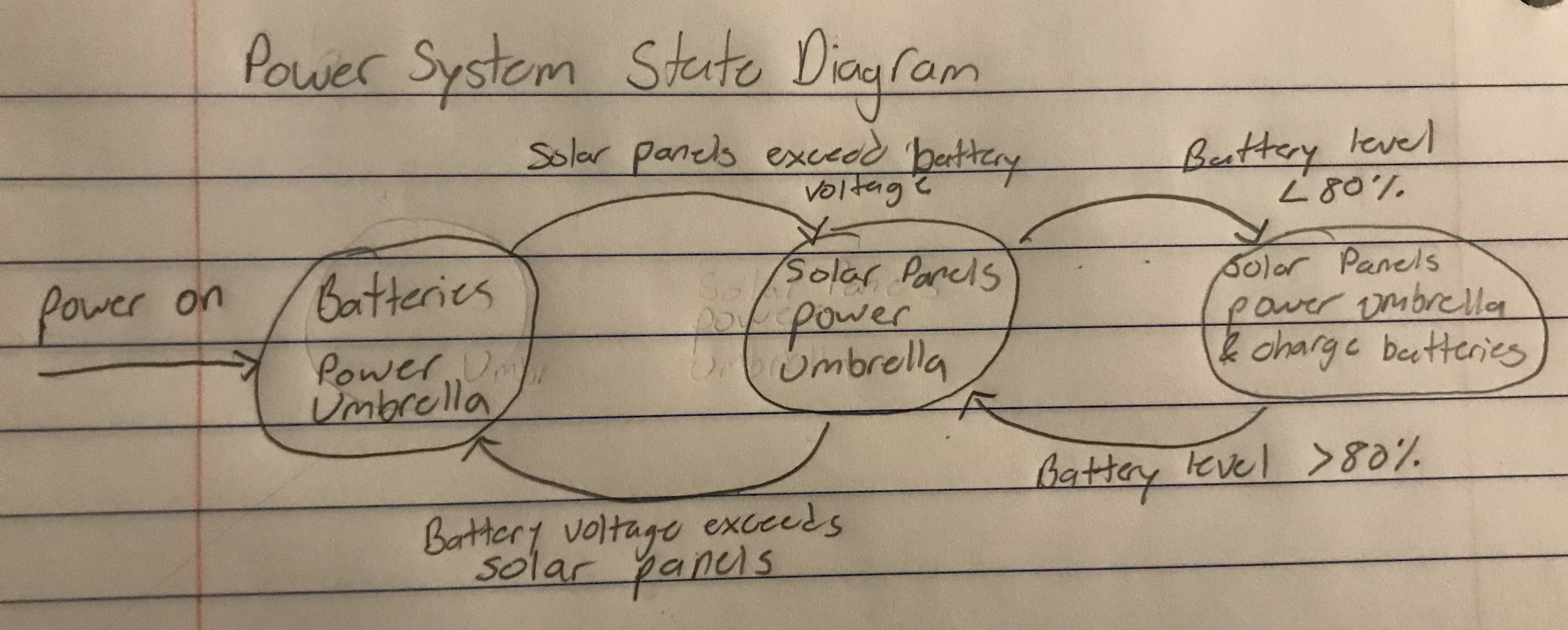


Power supply

The user specified that a solar panel should be implemented into the design.

Approach: In order to increase longevity of the power in the umbrella, solar panels will be implemented. Solar panels work by photons of light hitting the silicon atoms in the solar panels to excite electrons out of their valence shells. These free electrons are what generate electricity. Thus, the output from solar panels is directly dependent on sun exposure. The solar panels can be used to both power the umbrella as well as charge the batteries, Samsung 50E Lithium Ion Battery. A simple diode circuit shown below will be used to determine if the solar panels charge the batteries and/or power the umbrella. The bottom two diodes essentially act as a voltage comparator. Whichever has a higher voltage output will result in a voltage difference from high voltage to low, creating current to flow in that direction. The diode on the top will be used to charge the batteries as well when sufficient voltage is outputted from the solar panels. Since a microcontroller will be implemented, it can be used to detect the battery level. Based on the battery level, the microcontroller will or will not allow the solar panels to charge the batteries to prevent overcharging.





Ultrasonic humidifier

The user requires a humidifier to aid them with breathing properly in the heat.

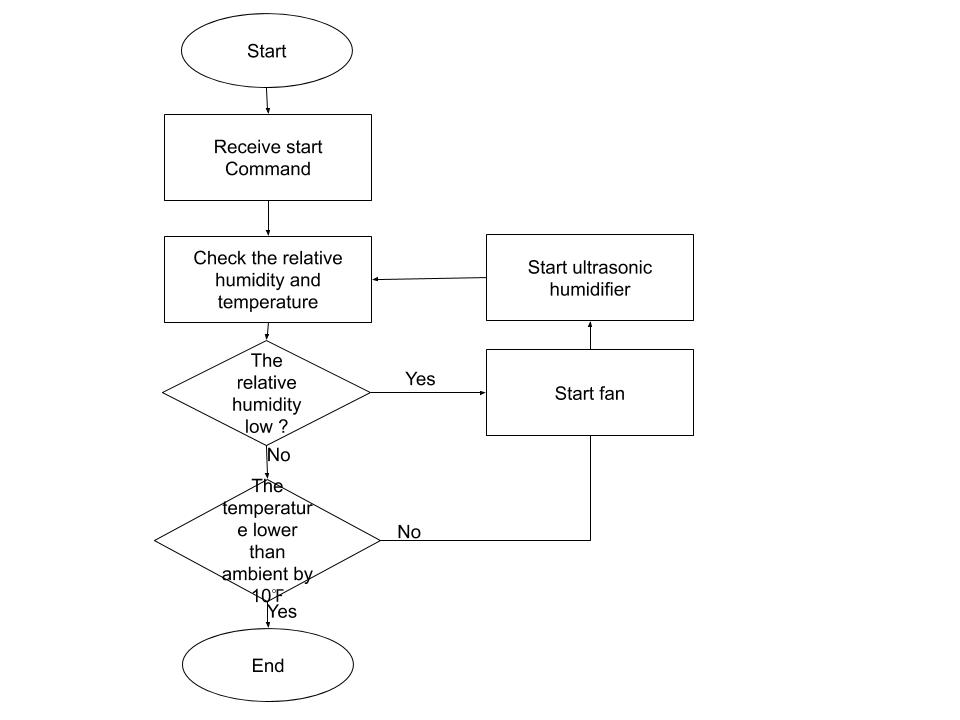
Approach: Our design will incorporate an ultrasonic humidifier, in particular the 20mm transducer fogger ceramics discs humidifier [8]. There will be four humidifiers placed on the umbrella ribs, on opposite sides of each other. Since the humidifiers are spread out there will be ample coverage to aid the users breathing. They will be controlled by the Raspberry Pi and will be adjusted by the humidifier sensors reading. Each humidifier will operate on 5V 300mA, from the solar power supply or the backup battery.

Integrated fan

We chose to use an integrated fan to accelerate evaporation and thus cool the air as is required by our consumer.

Approach: This fan will have a foldable propeller and will be placed at the upper part of the umbrella. The propeller will be able to fold when the umbrella is not being used. This type of propeller will be made using a 3D printer. For the motor of the fan, we choose to use a 1.5V to 3V DC motor . The motor can run at a continuous current of 0.09 A, consuming power as low as 0.135 W . Its size is small enough to fit in the umbrella rod; with a length of 17mm, width of 9mm, and thickness of 7mm.

3) flow diagrams with identification of subroutines and main parameters for simulation projects



4) Detailed description of software structure



The Raspberry Pi Zero will be controlling a DC motor, humidifier, temperature/humidity sensor, and a pump. The general idea for the code is that if we are outside the acceptable temperature and humidity range, we will turn the fan and humidifiers on to cool the air down. The plan is to try and keep the code as simple as possible to allow for flexibility later on. If components are added or removed the code should be able to be adjusted without too many headaches.

The humidifier turning on/off will be based on the humidity of the air.

The fan turning on/off will be based on the temperature of the air.

The water pump will require a formula based off the total time the humidifiers have been running and the rate at which they use water. The humidifiers are connected to a sponge that will store water next to the humidifier for evaporative cooling. By knowing the time the humidifiers have been on and their draw rate, we can calculate how much water will be left in the sponge. Knowing how much water a sponge can store will let us calculate how much water is left in the sponge and turn the pump on if we need to provide more water.

For the structure of the file, we will be essentially looping through a checklist of what needs to get done. On the first startup, we will need to pump some water into the sponges because they will most likely dry. Then we will call the sensor to get the temperature and humidity. Based on our accepted values we will tell the humidifier and or fan to turn on. If the sponge is getting dry, based on our calculations, we pump some water up. After some time passes we call the sensor for the temperature and humidity values again. Then we keep continuing the process until the user turns the power off.

**Prototyping progress report**

////////(Incomplete pending now )//////

-3D printing

o List of acquired components

Software, some filament

o Specify what was built, experimented with, tested

Experimented with the software and how best to 3d print certain components. In addition created one component but have not 3d printed it yet.

o Explain what you learned

Learned how to use the software more efficiently and became aware of a lot more options we have when it comes to 3d printing and the structure of the component.

**Testing plan**

o List of experiments

Cooling system:

1. Ultrasonic humidifier operation test. The ultrasonic humidifier will work under certain voltage and frequency. Since the data sheet of the ultrasonic humidifier is not provided, it is needed to test under what condition will it works.

Criteria: Success: works under 3.3V output of Raspberry Pi.

Failure: Cannot work under 3.3V output of Raspberry Pi.

1. Ultrasonic humidifier mist rate / power consumption rate test. Under the low power provided by Raspberry Pi, humidifier in different size may work differently. We need to find the most efficient one to use on our umbrella.

Criteria: Higher value of mist rate/ power consumption rate for an ultrasonic humidifier is wanted.

1. Water pump lift range test. Since different width of water pipe will result in different lift range, and based on different voltages, the lift range will also be different. We need our water pump to reach a certain lift range (larger than the length of the umbrella rod).

Criteria: Success: lift range reaches the required value using 3.3v or 5v output.

Failure: fail to reach the required lift range.

Power system:

1. Solar panel voltage test. The testing is going to focus on the solar panels. The diode circuit shown above will be created. A voltmeter will be used to test the voltage level at each node. Paper or some other material will be used to partially block the sunlight to a point where the battery voltage surpases the voltage of the batteries.

Umbrella construction:

1. 3d printed durability tests. These tests will test the usability and durability of the 3d printed parts. The parts will be used as intended to make sure they work properly without any damage to them and then if that is successful they will undergo drop tests to see what it takes to break them.

o Evaluation criteria

1. Solar panels: When the voltage of the solar panels drop below a threshold value, i.e. the voltage of the batteries, the batteries should output a constant voltage. An LED will be placed on the solar panels as well as the battery to determine where the voltage is being supplied from
2. 3d printed parts: When the 3d printed parts don't break when they are being used for the intended purpose and we redesign them as many times as needed to fix as many design flaws that can be found.

**List and description of tasks (for ECE 493)**

Stage one (one week):

1. Materials prepare and purchase

Materials test and repurchasing.

2. Soft crafting preparation

Knowledge picking up and pre-learning.

2.1 3D modeling

2.2 PCB design and assemble

Stage two(Three weeks):

3. Umbrella structure and circuit design and craft.

3.1 3D print components design and set up

3.2 umbrella cover material

3.3 3D printing modules function

4. Power system design and craft.

Solar panel placement and power supply distribution.

4.1 solar panel assembly

4.2’power bank/battery setup

4.3 circuit set up.

5. Control system design and craft.

Raspberry Pi programing and sensors connection testing.

5.1 raspberry Pi set up

5.2 sensor assembly

5.3softerware development

6. Cooling system design and craft.

Humidifier assembling and cooling level testing.

6.1 atomizer assembly

6.2 fans assembly( may be delayed one week)

6.3 tube assembly

Stage three(one week):

7. Testing#1

7.1 Experimental plans execution.

7.2 Components testing.

Stage four(one week):

8. Components integration.

Integrate all components together.

9. Testing#2

Experimental plans execution.

9.1 Experiment #1

9.2 Experiment #2

9.3 Experiment #3…..

Stage Five(three weeks):

10. Problems debugging.

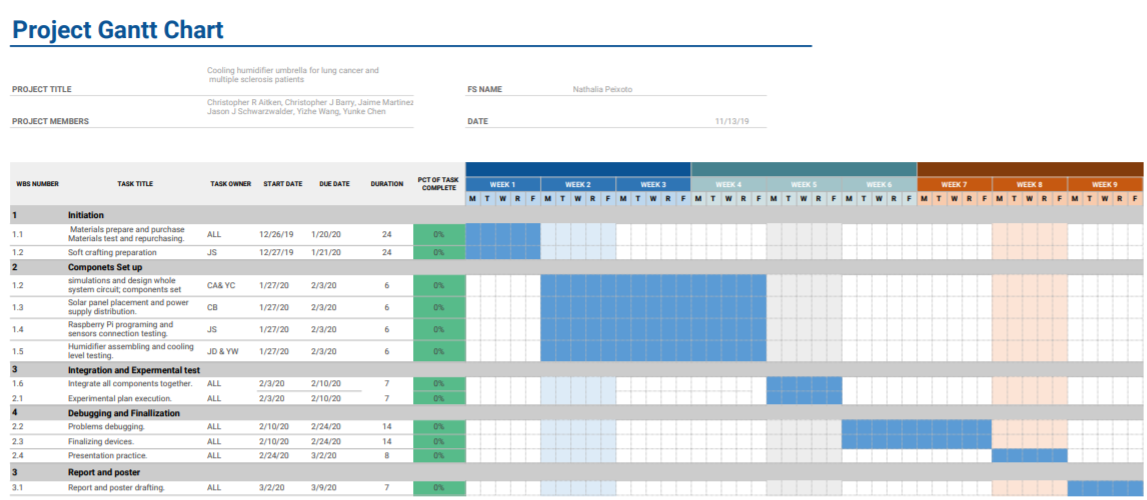
11. Finalizing devices.

Stage five (two weeks)

12. Presentation practice.

13. Report and poster drafting.

**Schedule**



Reference