

ECE 445
Spring 2026

Mobile Hive Checker Design Document

Team #1

Fiona Cashin (fcashin2)

Olivia Guido (ojguido2)

Rawda Abdeltawab (rawdaka2)

TA: Hossein Atae

Professor: Arne Fliflet

Contents

1. Introduction.....	3
1.1 Problem.....	3
1.2 Solution.....	3
1.3 Visual Aid.....	4
1.4 High Level Requirements.....	4
2. Design.....	5
2.1 Physical Design.....	5
2.2 Block Diagram.....	6
2.3 Functional Overview & Block Diagram Requirements.....	6
2.3.1 Sensor Subsystem.....	6
2.3.2 Power Subsystem.....	7
2.3.3 Microcontroller Subsystem.....	8
2.4 Hardware Design.....	9
2.4.1 Operating Voltage & Regulation.....	9
2.5 Software Design.....	11
2.5.1 Finite State Machine(FSM) Logic.....	11
2.5.2 Temperature and Humidity Sensor Interaction.....	12
2.5.3 CO ₂ Sensor Interaction.....	14
2.5.4 Threshold Comparison.....	15
2.5.5 Display Control Logic.....	16
2.6 Commercial Component Selection.....	16
2.6.1 Microcontroller- ESP32-C3-WROOM-02.....	16
2.6.2 Temperature and Humidity Sensor - AM2301B.....	17
2.6.3 Display - DIYables 20x4.....	17
2.6.4 CO ₂ Sensor SKU:SEN0159.....	18
2.6.5 Switch Rocker SPST 10A 125V.....	18
2.7 Tolerance Analysis.....	18
2.7.1 Measurement Uncertainty.....	18
2.7.2 Power Feasibility Analysis.....	19
2.8 Cost Analysis.....	20
2.8.1 Cost of Labor.....	20
2.8.2 Cost of Components.....	21
2.9 Schedule.....	23
2.10 Risk Analysis.....	25
3. Ethics and Safety.....	26
4. References.....	27

1. Introduction

1.1 Problem

Beekeepers must routinely monitor hive conditions to maintain healthy colonies and prevent colony loss. Critical environmental parameters such as temperature, humidity, and carbon dioxide (CO₂) concentration directly impact brood development, honey production, and overall colony health. However, traditional monitoring requires physically opening the hive, which stresses the bees, disrupts internal thermal regulation, and can negatively affect productivity and behavior when performed frequently.

Conversely, insufficient monitoring increases the risk of swarming, starvation, disease, or freezing, any of which can result in the loss of an entire colony. Each lost hive can cost a beekeeper approximately \$100–\$200 in replacement costs alone, not including lost honey production or pollination value. On a broader scale, honeybee decline is associated with habitat loss, pesticide exposure, climate change, and improper apiary management (Mull et al., 2022). Because pollination supports global agriculture and ecosystem stability, hive loss represents both an economic and environmental concern.

While commercial hive monitoring systems exist, many require permanent installation and cost approximately \$100 per hive, making them impractical for hobbyist and small-scale beekeepers. Additionally, many products are manufactured overseas, limiting accessibility and domestic technical support. These factors motivate the need for a low-cost, portable, and minimally invasive hive health assessment device.

1.2 Solution

The proposed solution is a portable Mobile Hive Checker device that allows beekeepers to assess colony health without opening the hive. Two slim sensors attached via cables are inserted through the hive entrance, while the main electronics enclosure remains external. This minimizes disruption to the bees while enabling measurement of internal hive conditions.

The sensors measure temperature, relative humidity, and carbon dioxide concentration. An ESP32 microcontroller processes sensor data and displays readings on an integrated screen. The system compares measured values against predefined healthy operating ranges and provides a clear “PASSED” or “FAILED” result. This allows beekeepers to quickly determine whether intervention is necessary while avoiding unnecessary hive disturbance.

The system is battery-powered and portable, allowing it to be used across multiple hives without permanent installation.

1.3 Visual Aid



Figure 1. *Visual Aid of Design*

Note. Images used: *What are the dimensions of a medium bee box* (HonestBee, n.d.); *4046* (DigiKey, n.d.-a); *DC-58P Heavy-Duty Electronics Enclosure* (Polycase, n.d.-a); *DHT21 AM2301 Capacitive Digital Temperature Humidity Sensor* (HiLetgo, n.d.); *3349839-left-hand-outstretched.jpg* (Kara98, 2014, CC BY-SA 4.0); *LP-31P Plastic Electronics Box* (Polycase, n.d.-b); *Arduino Pro Micro* (Pk0001, 2019, CC BY-SA 4.0); *PTS645SL43-2 LFS* (DigiKey, n.d.-b); *SEN0159* (DigiKey, n.d.-c); *LCD Display Module 16x02 LCD Screen Module* (Hosyond, n.d).

1.4 High Level Requirements

The following is a list of 3 quantitative characteristics this project should exhibit to solve the problem.

1. **Measurement Accuracy** : The device will measure hive temperature with an accuracy of ± 0.54 °F (± 0.3 °C), relative humidity with an accuracy of ± 2 % RH, and CO₂ concentration with an accuracy of $\pm(50$ ppm + 5 % of reading).
2. **User Interaction** : The device will respond to a push button input. Pressing the ON/OFF button shall power the display and show a Start Screen, followed by a Testing Screen during data acquisition. Pressing the Reset button shall return the device to the Start Screen. A manual boot button shall allow firmware recovery if necessary.
3. **User Output** : The device will display measured temperature, humidity, and CO₂ values. The device will display “PASSED” if all measurements are within acceptable ranges (70–97 °F, 50–60 % RH, CO₂ < 80,000 ppm). If any measurement falls outside these ranges, the device shall display “FAILED.”

2. Design

2.1 Physical Design

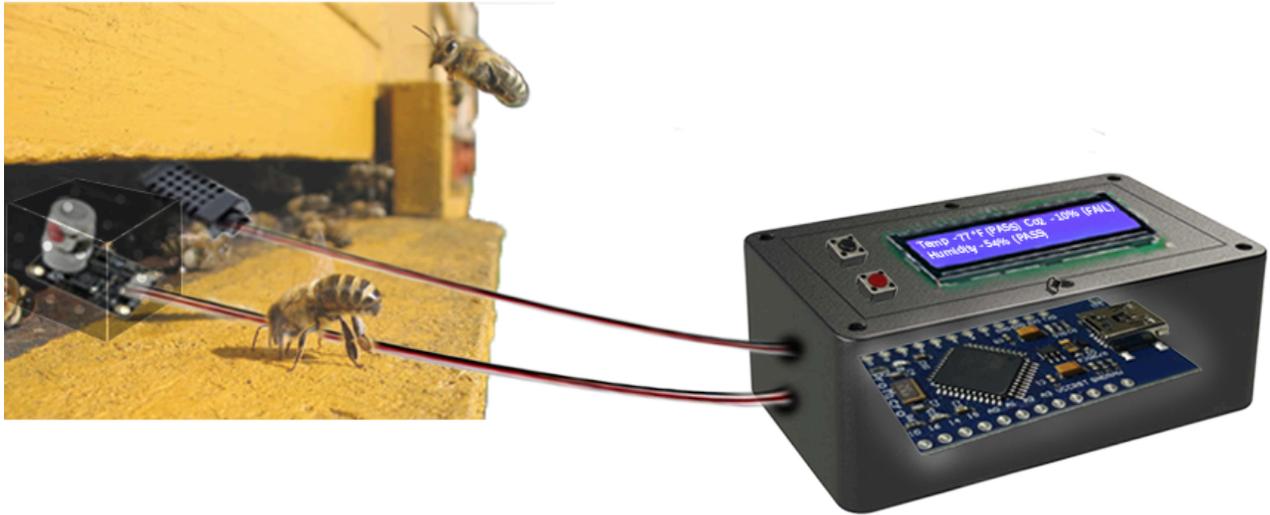


Figure 2. *Physical Design of Design*

Note. Images used: *A Guide to Beehive Entrance Management* (Perfect Bee, 2025); 4046 (DigiKey, n.d.-a); DC-58P Heavy-Duty Electronics Enclosure (Polycase, n.d.-a); DHT21 AM2301 Capacitive Digital Temperature Humidity Sensor (HiLetgo, n.d.); Arduino Pro Micro (Pk0001, 2019, CC BY-SA 4.0); PTS645SL43-2 LFS (DigiKey, n.d.-b); SEN0159 (DigiKey, n.d.-c); LCD Display Module 16x02 LCD Screen Module (Hosyond, n.d).

Unlike the visual aid, the physical design reflects realistic handheld dimensions. The sensor housings are sized to fit within a typical hive entrance (approximately 0.5 in height). Ventilation features for the PCB are not depicted; wiring enters one side of the enclosure while ventilation holes are located on the opposite side of the PCB box.

The Boot, Reset, and Power switches are the primary user inputs. The cable-mounted sensors allow mobility between hives while minimizing bee disturbance.

2.2 Block Diagram

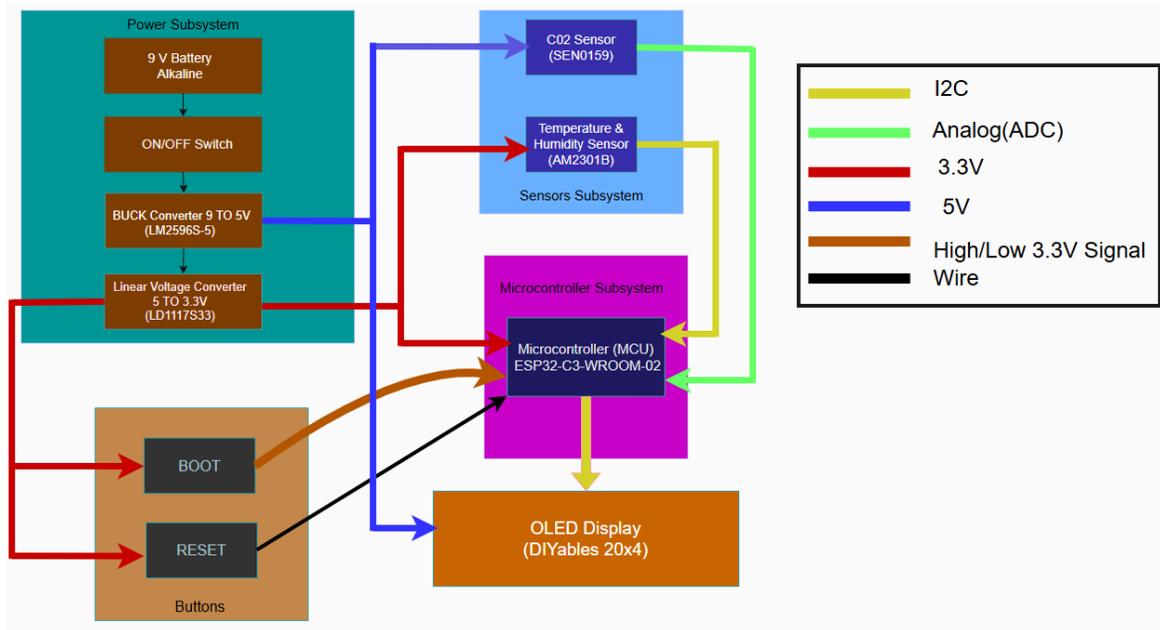


Figure 3. Block Diagram of Design

2.3 Functional Overview & Block Diagram Requirements

2.3.1 Sensor Subsystem

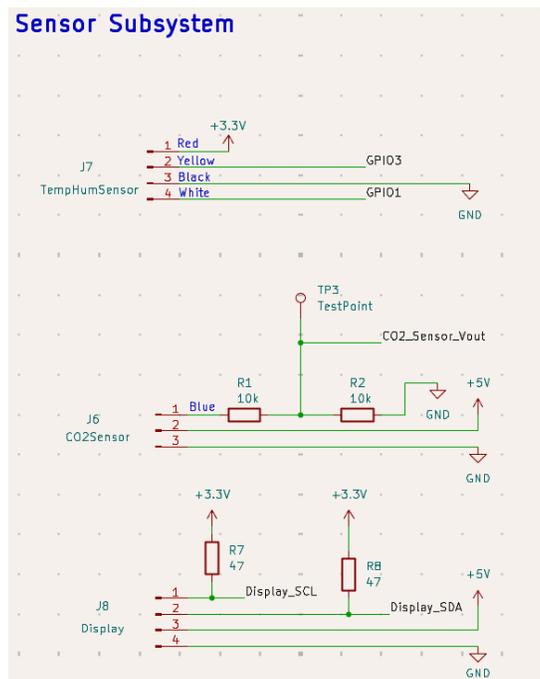


Figure 4. Sensor Subsystem Schematic

This subsystem measures internal hive temperature, humidity, and CO₂ concentration. The temperature/humidity sensor and CO₂ sensor are mounted on extended cables, allowing insertion into the hive while keeping the main enclosure external. Sensor outputs interface with the ESP32 microcontroller for processing.

Maintaining temperature between 70–97 °F is critical for brood survival. The selected temperature sensor provides ±0.54 °F accuracy, sufficient to detect meaningful deviations.

Humidity must remain between 50–60 % RH for proper nectar dehydration and brood health. The selected humidity sensor provides ±2 % RH accuracy.

CO₂ concentrations above 8 % (80,000 ppm) may indicate overcrowding or poor ventilation. The selected CO₂ sensor measures 400–100,000 ppm with ±(50 ppm + 5 % of reading) accuracy.

Requirements	Verification
<ul style="list-style-type: none"> The Temperature and Humidity Sensor shall operate within ±5 % of its rated +3.3V supply voltage. 	<ul style="list-style-type: none"> Measure Temperature and Humidity Sensor +3.3V supply voltage using a multimeter.
<ul style="list-style-type: none"> The CO₂ Sensor shall operate within ±5 % of its rated +5V supply voltage. 	<ul style="list-style-type: none"> Measure CO₂ Sensor +5V supply voltage using a multimeter.
<ul style="list-style-type: none"> The Temperature and Humidity Sensor shall provide temperature accuracy of ±0.54 °F and humidity accuracy of ±2 % RH. The CO₂ sensor shall provide CO₂ accuracy of ±(50 ppm + 5 % reading) 	<ul style="list-style-type: none"> Compare sensor readings to known environmental references or calibrated instruments.

2.3.2 Power Subsystem

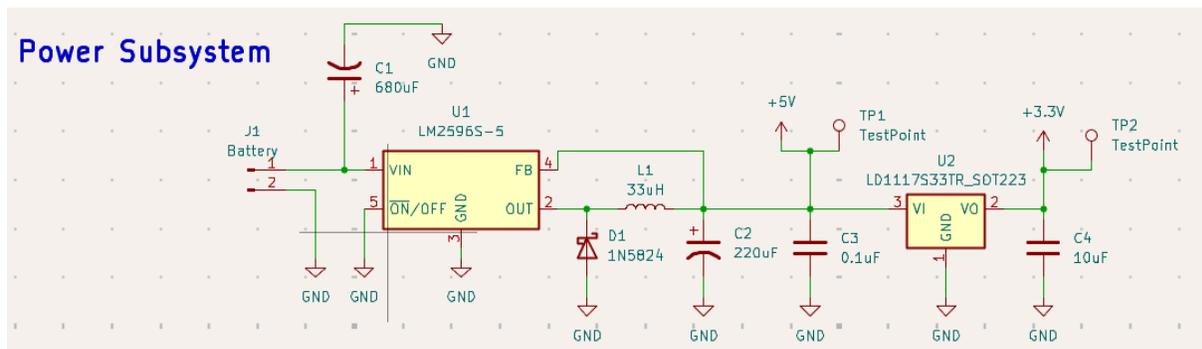


Figure 5. Power Subsystem Schematic

The system is powered by a 12V battery. The battery voltage is stepped down to 5V using a DC-DC buck converter to improve efficiency. The 5V rail powers the CO₂ sensor and the Display. A 3.3V low-dropout (LDO) regulator steps the 5V rail down to 3.3V to power the ESP32 microcontroller and Temperature and Humidity Sensor. The voltage rails are labeled and distributed according to subsystem requirements.

Requirements	Verification
<ul style="list-style-type: none"> The battery shall supply 12V DC. Buck Converter shall regulate 12V to 5V $\pm 5\%$ under full load. The Linear Voltage Regulator shall regulate 5V to 3.3V $\pm 3\%$. 	<ul style="list-style-type: none"> Measure output voltages under load conditions using a multimeter
<ul style="list-style-type: none"> The power subsystem shall supply at least 1A peak current without voltage droop exceeding 5%. 	<ul style="list-style-type: none"> Apply maximum expected load and verify voltage stability.

2.3.3 Microcontroller Subsystem

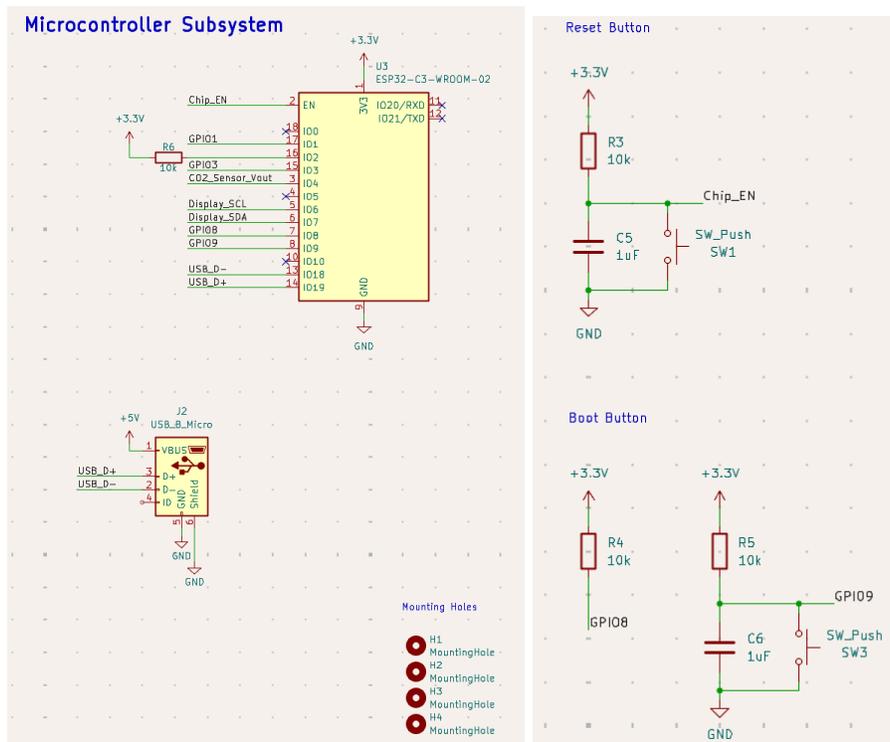


Figure 6. Microcontroller Subsystem Schematic

The ESP32-C3-WROOM-02 microcontroller collects data from all sensors and processes the results. Temperature and humidity data are transmitted via I²C. The CO₂ sensor provides an analog voltage output (0–5V), which is scaled using a voltage divider to ensure the ESP32 ADC input does not exceed 3.3V.

The ESP32 processes measurements and compares them against defined thresholds. Results are displayed via I²C to the LCD.

The ESP32 operates at 3.3V and has a typical active current significantly lower than 345 mA (that value represents peak transmission current under Wi-Fi usage). Since we do not require the wireless features, the average system current will be substantially lower.

Requirements	Verification
<ul style="list-style-type: none"> ● The microcontroller successfully receives data from peripheral sensors when data is available. Data will be received from: <ol style="list-style-type: none"> 1. Temperature and Humidity Sensor 2. CO₂ Sensor 	<ul style="list-style-type: none"> ● Observe output data on serial monitor to confirm successful communication ● Ensure data is received after push button is pressed
<ul style="list-style-type: none"> ● The microcontroller successfully boots the system with the USB-C mode 	<ul style="list-style-type: none"> ● Observe the booting sequence on serial monitor ● Could push button manually to boot it as fail-safe option
<ul style="list-style-type: none"> ● The microcontroller successfully delivers information to the display ● The display successfully shows the correct data from the microcontroller conveniently to the user 	<ul style="list-style-type: none"> ● Ensure correct output is displayed by comparing output data from the serial monitor to the display

2.4 Hardware Design

2.4.1 Operating Voltage & Regulation

The power subsystem needs to power the sensors, the display, and the MCU. The current draw of these components vary. For example, the display's brightness can be tuned using a potentiometer that is on the board already. Because of these variations, we needed a power subsystem that can be used with a larger battery. To convert our battery's 12V to 5V, I used a simple switcher Step-Down Regulator.

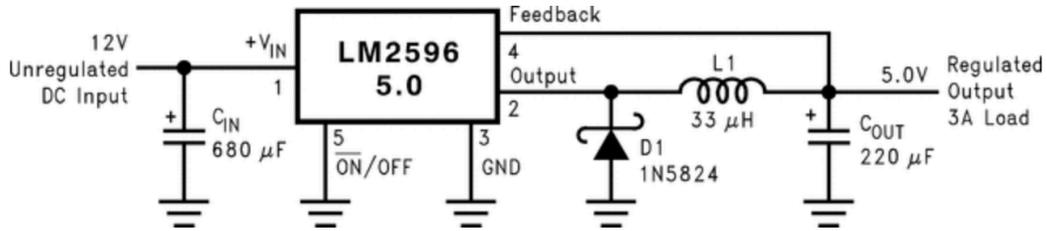


Figure 7. *LM2596 Typical Application*

As shown, a 12V battery is typical to use for this component and the output current is more than enough as calculated in our Power Feasibility Analysis. This circuit is then directly connected to the low drop voltage regulator to decrease the voltage even more for the microcontroller and Temperature and Humidity Sensor.

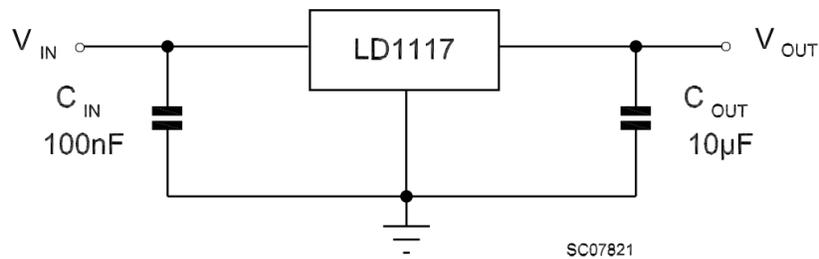


Figure 8. *LD1117 Typical Application*

Since we do have a large current draw, specifically the CO₂ sensor, we made sure to pick these devices that are very power efficient.

2.4.2 Sensitive Components Separation

To reduce any electromagnetic interference, we made sure to place components that are very susceptible to noise far away from noisy components. The sensors and antenna are on the opposite corner of the PCB than the battery.

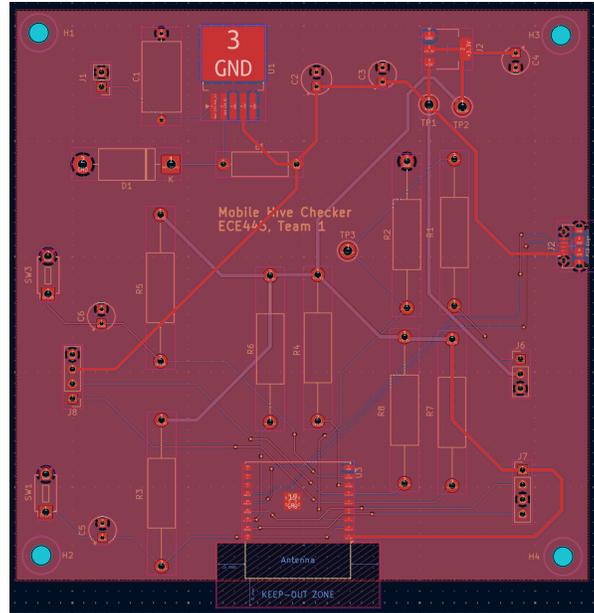


Figure 9. *PCB Layout*

2.5 Software Design

2.5.1 Finite State Machine(FSM) Logic

The microcontroller’s software is the central decision-making engine of our system. It’s responsible for continuously acquiring data from all onboard sensors, processing the measurements, and updating the display with the latest results. The system operates using a finite state machine (FSM) built around a simple polling-based control loop that achieves periodic sensor reads, evaluates system conditions, and transitions between states as needed to maintain a safe and reliable operation. The firmware is developed using Arduino IDE 2.0 and flashed on a microcontroller via USB. The main control flow states are:

- **START/IDLE:** system initialization where display shows a start screen with project name and information about buttons after the ON/OFF button is pressed. All sensors are powered but not actively measuring.
- **TESTING:** this is triggered once the system validates that sensors are ready for operation. It could be calibration wait or warming up phase for sensors. Then the MCU polls the temperature and humidity sensor over I²C and reads the analog output of the CO₂ sensor via its ADC pin. The display will show a “testing phase”.
- **RESULT:** Display all three measured values (CO₂ in ppm, temperature in °F, and relative humidity in %RH) along with an PASS/FAIL status overall if one of measurement is over the threshold values.
- **RESET:** this is triggered by pressing the Reset button at any time where it returns the firmware to the START/IDLE state and clears all stored measurement values.

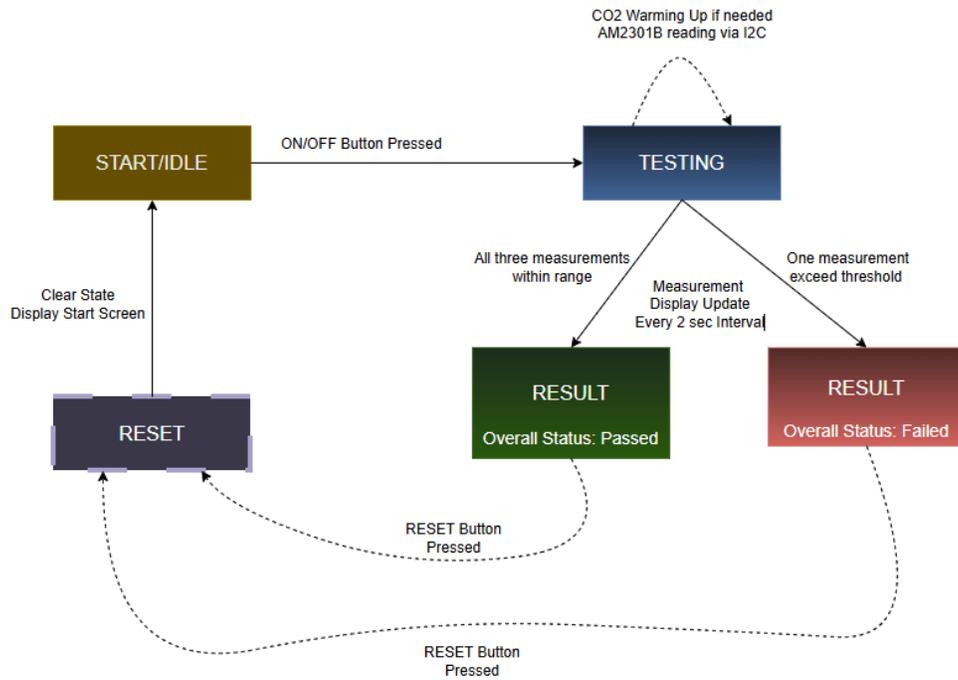


Figure 10. *Finite State Machine(FSM) Logic*

2.5.2 Temperature and Humidity Sensor Interaction

The AM2301B is a sensor that measures humidity and temperature. It communicates to the MCU using either standard or Fast I²C protocol mode at a 7-bit device address of 0x38 on its own bus compared to the display. There are two processes for the I²C communication protocol for sensors: Initialization and Measurement (Adafruit Industries, n.d.).

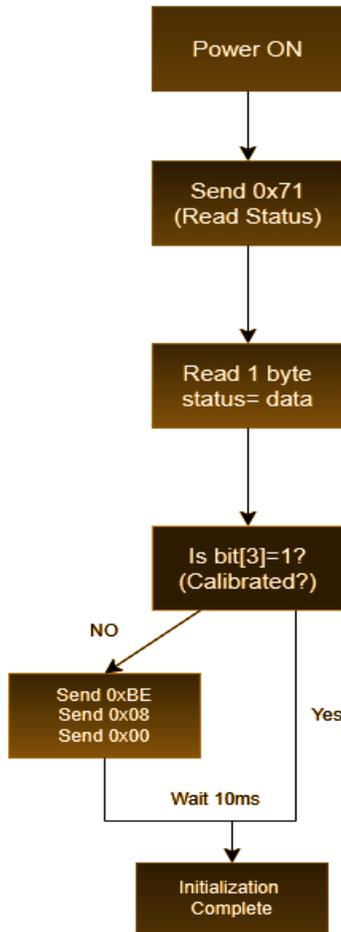


Figure 11a. *Initialization Sequence*

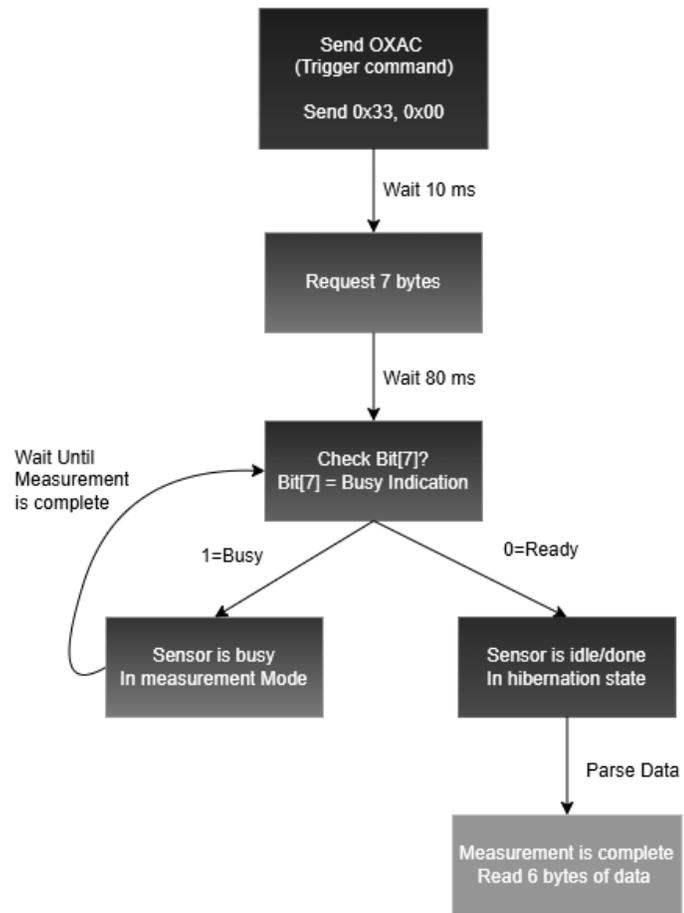


Figure 11. *Measurement Sequence*

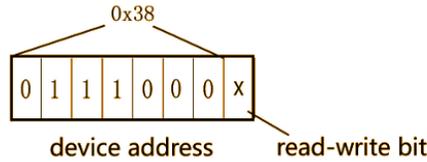
NOTE : We do not read CRC check data as there is no need for it.

SIGNAL CONVERSION FORMULAS : temperature and humidity are computed from the raw 20-bit values using the formulas defined in the datasheet after the measurement is complete:

$$T[{}^{\circ}\text{C}] = (S_T/2^{20}) \times (200 - 50) ; S_T \text{ is raw temperature}$$

$$RH[\%] = (S_{RH}/2^{20}) \times 100\% ; S_{SH} \text{ is raw relative humidity}$$

Also, temperature is displayed in ${}^{\circ}\text{F}$ which is derived by applying the standard Celsius to Fahrenheit conversion. According to the datasheet, it recommends sampling no more than once every 2 seconds to avoid self-heating of the sensor so the firmware enforces a 2-second minimum measurement interval.



Bits	Significance	Description
Bit[7]	Busy indication	1-Equipment is busy, in measurement mode 0- Equipment is idle, in hibernation state
Bit[6:5]	Retain	Retain
Bit[4]	Retain	Retain
Bit[3]	CAL Enable	1 - Calibrated 0 - Uncalibrated
Bit[2:0]	Retain	Retain

Figure 12. Status Bit Description
(Adafruit Industries, n.d.)

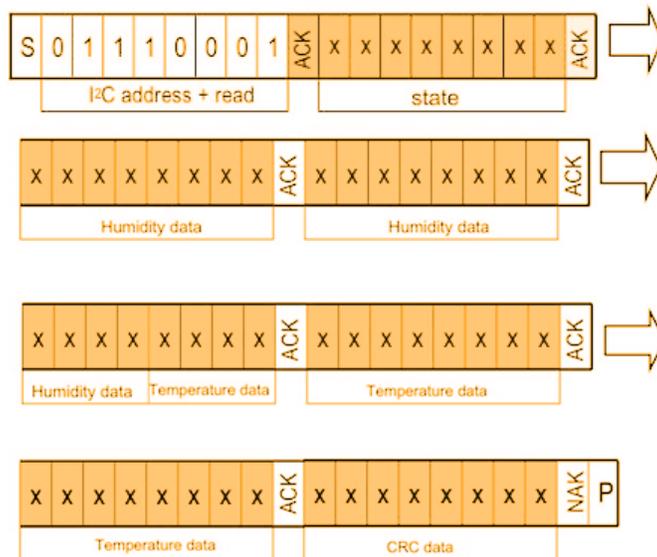


Figure 13. Read Temperature and Humidity data after Trigger Command
(Adafruit Industries, n.d.)

2.5.3 CO₂ Sensor Interaction

The SEN1059 CO₂ sensor produces an analog output voltage that decreases as CO₂ concentration increases. The sensor requires a minimum warm-up period of 1 hour (especially after extended storage time) to output stable reading. The MCU 12-bit ADC pin reads the voltage from the analog output pin. Since the sensor outputs 0-5V signal and MCU input pins tolerate 3.3V, a voltage divider (2- 10KΩ resistors) used to scale the signal to 0-2.5 V range before it reaches the MCU ADC pin.

$$V_{MCU,ADC} = 5V * \frac{10K\Omega}{10K\Omega+10K\Omega} = 3.3V$$

The code will sample the ADC 10 times with 50ms time interval between each sample then averages the result to reduce noise as shown in the datasheet.

The averaged raw ADC value is converted to a voltage value at the ADC pin then scaled back to reconstruct the CO₂ original output voltage signal. This voltage is divided by onboard amplifier gain (DC_GAIN=8.5) to get voltage value compared to volts of maximum and minimum concentration of CO₂. The maximum (MAX_POINT_VOLTAGE) and minimum (ZERO_POINT_VOLTAGE) have default calibrated values but we could update them after measurement using a CO₂ meter. The CO₂ connotation [ppm] is computed using a two-point logarithmic calibration curve as shown in Figure 14.

```

/***** MQGetPercentage *****/
*****
Input:  volts  - SEN-000007 output measured in volts
        pcurve - pointer to the curve of the target gas
Output: ppm of the target gas
Remarks: By using the slope and a point of the line. The x(logarithmic value
of ppm)
        of the line could be derived if y(MG-811 output) is provided. As it
is a
        logarithmic coordinate, power of 10 is used to convert the result t
o non-logarithmic
        value.
*****/
int MGGetPercentage(float volts, float *pcurve) {
    volts = volts / DC_GAIN;
    if (volts > ZERO_POINT_VOLTAGE || volts < MAX_POINT_VOLTAGE ) {
        return -1;
    } else {
        return pow(10, (volts - pcurve[1]) / pcurve[2] + pcurve[0]);
        volts = 0;
    }
}

```

Figure 14. Calculation of CO₂ sensor in ppm (DFRobot Wiki, n.d.)

2.5.4 Threshold Comparison

The microcontroller evaluated each sensor measurement against the predefined healthy beehive operating ranges after the sensor's data had been collected. The comparison logic is as follows:

Measurement	Healthy Range	Sensor Accuracy	Result
CO ₂ Concentration	< 80,000 ppm	±(50 ppm + 5 % reading)	PASS/FAIL
Temperature	70-97 °F	±0.54 °F (±0.3 °C)	PASS/FAIL
Relative Humidity	50-60 % RH	±2 % RH	PASS/FAIL

An overall result of “PASSED” is displayed only when all three individual measurements are within their healthy ranges. If any measurement falls outside its range, it will display “FAILED” as the overall result. This ensures the beekeeper is promptly alerted to address the issue.

2.5.5 Display Control Logic

The 20x4 DIYables display communicates to the microcontroller via I²C communication protocol and uses an edited version of LiquidCrystal_I2C library as the display is on its own bus compared to the temperature and humidity sensor. Since the display drives SDA/SCL at 5V, we need to adjust it by pull-up resistors to bring those lines up to a safe 3.3V that the MCU’s input pins can tolerate. The display has 0-19 cols and 0-3 rows to show the data. During the START screen, the display will show the name of the project and information about reset and power ON/OFF button. During the TESTING state, the display will show "SENSORS WARMING UP AND TESTING..." to activate processing. The display brightness can be reduced via the potentiometer on the display onboard to manage the current draw. During the RESULT state, the display will show four organized rows.



Figure 15. *RESULT State Display*

2.6 Commercial Component Selection

2.6.1 Microcontroller- ESP32-C3-WROOM-02

The ESP-32-C3-WROOM-02 (Espressif Systems) is chosen as the central microcontroller (MCU) for this project with operating voltage of 3.3V. It's a compact, low-power, RISC-V based single core module operating at up to 160 MHz with 4 MB embedded Flash and 400 KB SRAM. The rationale behind this component (Espressif Systems, n.d.):

- ADC Module: it provides a 12-bit SAR ASC with up to 6 channels which's sufficient to read the analog voltage output of the CO₂ sensor. The output of the sensor will be scaled down to 0-2.5v via external voltage divider and MCU can tolerate up to 3.3V for input pins.
- GPIOs: It has enough digital I/O pins to support Reset button, manual Boot button, SDA/SCL lines.

- Power Consumption: Active CPU current is around 28 mA which is substantially lower than 345 mA peak WiFi. Since Wi-Fi is not used in design yet, the actual MCU current is within range of the power system.
- Antenna: It has an integrated onboard antenna that can be leveraged later to add more features to the project if time permits. It has Wi-Fi 2.4 GHz and Bluetooth 5.0.
- USB Serial/JTAG controller: it's integrated in the ESP32-C3 which makes it easier to boot via USB without the need of USB-to-UART bridge and automatically boot the system.
- I²C support: it integrates an I²C controller to enable direct integration with both the temperature and humidity sensor and display on shared or separate I²C bus at their respective addresses.
- It has a 40MHz crystal oscillator integrated into the module so we can save space on the PCB.

2.6.2 Temperature and Humidity Sensor - AM2301B

The AM2301B was selected for its communication protocol over I²C, its accuracy, and wide supply voltage range from 2.2-5.5 V. Also, it has a long term stability and enclosed housing to allow sensor to be inserted into the beehive environment without risking damage to the sensing elements. The sensor's datasheet was significantly more detailed and informative than those of other possibilities we evaluated and this sensor was offered at lower overall cost.

More rationale behind this component (Adafruit Industries, n.d.):

- Accuracy: The sensor archives $\pm 2\%$ RH for relative humidity and $\pm 0.3\text{ }^{\circ}\text{C}$ for temperature which meet the project high level requirements of beehive normal operation.
- Supply Voltage: it's directly compatible with 3.3 V of the MCU so no level shifting will be required.
- I²C Interface: it communicates at address 0x38 using the standard I²C protocol(Standard Mode up to 100 KHz and Fast mode up to 400 KHz).
- Power Consumption: It uses 980 μA furin active measurement and 250 nA in deep sleep
- Response time: < 8 seconds (63% step)
- Cable mounting: the sensor includes a lead wire that connects to the PCB while the enclosed housing at the opposite end is inserted into the hive entrance so the PCB can remain outside the hive.

2.6.3 Display - DIYables 20x4

- The opposite end is inserted into the hive entrance so the PCB can remain outside the hive.

The DIYables 20x4 display communicates over I²C protocol where we can use up to 19 cols and 3 rows. The display requires a 5V supply voltage which comes from the power subsystem. It includes a 40 mA backlight current that can be adjustable via onboard potentiometer to reduce power consumption. It has sufficient character capacity to simultaneously display all three sensor

readings with an overall PASSED/FAILED status. It's compact form will make it fit perfectly on the box will make at so it can be portable system

2.6.4 CO₂ Sensor SKU:SEN0159

This CO₂ sensor was chosen for its ppm rating. The max humidity of a healthy beehive is 8% which correlates to 80,000 ppm. This sensor has a max ppm rating of 100,000 ppm making this well within the range that we need. Additionally the sensors had to be small enough to fit into the beehive entrance. The average height of a beehive is 0.5 inches and length of 14 inches. The main concern is the height. This CO₂ sensor was the best option for accuracy and ppm range however it is not the best for physically putting into the hive because of its 1 in height. To mitigate this, we plan to put the sensor on its side to slide it into the hive entrance.

2.6.5 Switch Rocker SPST 10A 125V

This switch was chosen for its high current capacity. In case we need to use a larger battery with more current, this single pole single throw switch will work. We also need a switch that could be connected to the PCB without actually being clipped into the PCB. The switch will be connected to the PCB board by soldering a wire to the button and then that wire is soldered to the PCB. It has a snap in panel mounting style for easy installation. Another important aspect that we wanted in our On Off button was to see if the button was in the on or off position. This button has the on off symbols on the button making sure that the user can easily identify how to turn the device on.

2.7 Tolerance Analysis

2.7.1 Measurement Uncertainty

A major risk to the Mobile Hive Checker is measurement uncertainty near the pass/fail thresholds. Because hive health is determined by comparing measured temperature, humidity, and CO₂ values to acceptable ranges, sensor tolerances may cause values near these limits to be misclassified.

The acceptable temperature range is 70–97 °F, and the temperature sensor has an accuracy of ± 0.54 °F. At the lower limit of 70 °F, measurements may range from 69.5 °F to 70.5 °F, meaning a healthy hive could be incorrectly marked as FAILED. However, the total temperature range is 27 °F, so the sensor uncertainty represents only about 2 % of the range, which is acceptable for detecting meaningful temperature changes.

- **Temperature :**
 - *Range = 27 °F (70–97 °F)*
 - *Uncertainty = ± 0.54 °F (~2 % of total range)*
 - *Impact: Minimal risk of incorrect classification.*

The acceptable humidity range is 50–60 % RH, with a sensor accuracy of ± 2 % RH. At 50 % RH, readings may range from 48 % to 52 %, meaning that readings near the boundary could occasionally fall outside the acceptable range. This uncertainty represents about 20 % of the total 10% acceptable range, which is relatively large. However, humidity in a hive changes gradually, and values near the limits already indicate emerging suboptimal conditions. Therefore, occasional boundary misclassification is acceptable for this application.

- **Humidity :**
 - *Range = 10 % RH (50–60 %)*
 - *Uncertainty = ± 2 % RH (20 % of acceptable band width)*
 - *Impact: Higher relative uncertainty. However, gradual environmental change reduces the likelihood of repeated misclassification.*

The CO₂ threshold is set at 80,000 ppm, and the sensor accuracy is $\pm(50 \text{ ppm} + 5 \%$ of the reading). At this threshold, measurements may range from approximately 75,950 ppm to 84,050 ppm. Although this introduces uncertainty near the limit, CO₂ levels approaching this value already indicate dangerous hive conditions, making the tolerance acceptable for a diagnostic tool.

- **CO₂ at 80,000 ppm threshold :**
 - *Range = $> 80,000 \text{ ppm}$*
 - *Uncertainty = $\pm(50 + 0.05 \times 80,000) = \pm(50 + 4000) = \pm 4050 \text{ ppm}$*
 - *Impact = Thus actual reading could range from $\sim 75,950 \text{ ppm}$ to $84,050 \text{ ppm}$. Because 80,000 ppm represents extreme hive conditions, this tolerance is acceptable for diagnostic purposes.*

While sensor inaccuracies may affect measurements near decision boundaries, the selected sensors provide sufficient accuracy for the intended purpose to prompt the beekeeper to action when the hive starts to reach those unhealthy boundaries.

2.7.2 Power Feasibility Analysis

Another important consideration in our design is the ability of the battery to supply sufficient current to all components during operation. Because the device is intended to be portable and used across multiple hives, accurate estimation of current draw is necessary to ensure reliable operation and acceptable runtime.

A standard 9V alkaline battery can typically supply approximately 300–600 mA for short durations, though sustained operation near the upper limit may cause voltage sag and reduced battery life. Therefore, the worst case current draw must be carefully evaluated.

$$28 \text{ mA} + 0.00098 \text{ mA} + 500 \text{ mA} + 40 \text{ mA} = 568.001 \text{ mA}$$

This results in an approximate total current draw of 568 mA, which is close to the upper limit of a typical 9V battery's current capability. Because the estimated draw approaches the maximum supply capability, the LCD backlight brightness will be reduced to lower display current.

Although the calculated draw is within the 600 mA upper range, sustained operation at this level may reduce battery life significantly and may cause voltage droop under load. Because of this calculation, we ordered a 12V rechargeable Li-ion pack instead of a 9V Duracell battery.

Component	Operating Voltage	Current Draw
MCU	3.0 - 3.6 V	345 mA (peak with WiFi) 28 mA (Active CPU (no radio))
Temperature and Humidity Sensor	3.3V-5.2V (recommended supply is 5V)	0.00098 mA (measuring) 250 μ A (dormant)
CO ₂ Sensor	0 - 5V (recommended supply is 5V)	500 mA (absolute max)
LCD Display	2.8-5.5V	40 mA (lowered by decreasing backlight via a potentiometer already on display)

2.8 Cost Analysis

2.8.1 Cost of Labor

On average thus far in the semester, each of us have been contributing approximately 15 hours a week to the research and construction of our design. We multiplied this by 2.5 to adjust our numbers to a reasonable salary. A few weeks have been closer to 30 hours per week while others are closer to 10, thus 15 hours is a fair future weekly estimate. A lot less time could be spent in the initial construction of our design if we didn't keep meticulous design and measurement notes along each step of the process. As a group we agree on not being careless throughout this semester as this can lead to damaged parts or errors due to disorganization, which can cost us all time and money that we might not have by the end of the semester. From the creation of the idea to preparing for our demo, without including time for school breaks, mock demos, or the final demo, is 12 weeks. According to the University of Illinois' ECE Department, the most recent statistic for employment comes from 2023-2024, with an annual salary of \$90,115 for electrical engineering graduates, and \$103,222 for computer engineering graduates (University of Illinois Urbana-Champaign, n.d.). Our team is composed of one computer engineering student and two electrical engineering students. In our labor calculations, we thus will use \$43/hour and \$50/hour

respectively in our team cost estimation. This was seen as fair by our team as different skills correlate to different salaries in industry.

Electrical engineering members:

$$\frac{\$43}{\text{hour}} \times 2.5 \times \frac{15 \text{ hours}}{\text{week}} \times 12 \text{ weeks} = \$19,350 \times 2 \text{ members} = \$38,700$$

Computer engineering member:

$$\frac{\$50}{\text{hour}} \times 2.5 \times \frac{15 \text{ hours}}{\text{week}} \times 12 \text{ weeks} = \$22,500 \times 1 \text{ member} = \$22,500$$

For the machine shop, we only needed the construction of two boxes. One box will serve as a protective casing for our CO₂ sensor, as the one we ordered for the design didn't come with one. This box comes equipped with custom ventilation holes to allow our sensor to function properly. The other box is a large plastic housing that will hold every other component that makes up our monitoring system. The shop will have to make precise and custom holes to allow the Boot and Reset buttons to be usable from the lid of the box, as well as integrate our LCD display within the lid of the box. The estimated cost per hour for use of the machine shop services outside of this class is \$70 per hour.

Machine Shop:

$$\frac{\$70}{\text{hour}} \times \frac{3 \text{ hours}}{\text{week}} \times 2 \text{ weeks} = \$420$$

This brings our total for labor using starting salaries and estimated hours to be \$61,620.

2.8.2 Cost of Components

Description	Part Number	Datasheet	Manufacturer	Vendor	Cost Per (\$)	# Ordered	Total Cost (\$)
MCU	ESP32-C3-WR OOM-02	Datasheet	Espressif Systems	Digi Key	3.28	5	16.4
USB-C Connector	USB4085-GF-A GCT	Datasheet	GCT	DigiKey	0.87	1	0.87
Temperature and Humidity Sensor	adafruit Industries 5181 Temperature Sensors	Datasheet	Adafruit	Adafruit	14.39	1	14.39

CO ₂ Sensor	SEN0159	Datasheet	DFRobot	Amazon	79.82	1	79.82
Display	LCD I2C 20x4 Display with Blue Backlight with ESP32	Datasheet	DIYables 20x4	Amazon	7.99	1	7.99
12V Battery	KBT 12V 1200mAh Rechargeable Li-ion Battery	N/A	Shenzhen (KBT) Keep Better Tech Electronics Co., Ltd	Amazon	15.99	1	15.99
Buck Converter(9-5)	LM2596S-5.0/NOPB	Datasheet	Texas Instruments	DigiKey	6.97	2	13.94
Linear Voltage Regulator	LD1117S33	Datasheet	STM	DigiKey	0.32	3	0.96
On/Off Button	RA1113112R	Datasheet	E-Switch	DigiKey	0.64	2	1.28
RST Button	PS1024ARED	Datasheet	E-Switch	DigiKey	1.87	2	2.74
Boot Button	PS1024ABLK	Datasheet	E-Switch	DigiKey	1.55	2	2.10
Resistor	4.7k	N/A	Stackpole Electronics Inc	ECEB E-Shop	0	4	0
Resistor	10k	N/A	Stackpole Electronics Inc	ECEB E-Shop	0	12	0
Resistor	5.1k	N/A	Stackpole Electronics Inc	ECEB E-Shop	0	4	0
Capacitor	680μF	Datasheet	Rubycon	ECEB E-Shop	0	2	0
Capacitor	220μF	N/A	Rubycon	ECEB E-Shop	0	2	0
Capacitor	0.1μF	N/A	Rubycon	ECEB E-Shop	0	2	0
Capacitor	10μF	N/A	Rubycon	ECEB E-Shop	0	2	0
Capacitor	1μF	N/A	Rubycon	ECEB E-Shop	0	4	0
Inductor	33μH	Datasheet	EPCOS - TDK Electronics	DigiKey	0.27	4	1.08
Diode	1N5824	Datasheet	Vishay General Semiconductor - Diodes Division	DigiKey	0.62	4	2.48
2-Pin Connector	M20-9990246	N/A	Harwin Inc.	ECEB E-Shop	0	5	0

3-Pin Connector	<u>M20-9990346</u>	N/A	Harwin Inc.	ECEB E-Shop	0	5	0
4-Pin Connector	<u>M20-9990446</u>	N/A	Harwin Inc.	ECEB E-Shop	0	5	0
COMPOENTS							\$160.04
LABOR							\$61,620
TOTAL SUM							\$61,780.04

2.9 Schedule

Week	Team's Goals	Person	Deadlines
Jan 19 - Jan 25	Getting organized in the class and communicating as a group.	Whole group	Initial Web Board Post (Jan 22)
Jan 26 - Feb 1	Each member is in charge of creating 2-3 well thought out project ideas _____ Attend a group meeting to discuss pros and cons to each design _____ Keep track of ideas on the Web Board _____ Begin working on the RFA document	Whole group _____ Whole group _____ Whole group _____ Whole group	Early Project Approval (Jan 29)
Feb 2 - Feb 8	Begin project proposal _____ Gather all old lab materials to have a sense of what the group has _____ Members start sourcing possible sensors, displays, and MCUs to use	Whole group _____ Whole group _____ Whole group	Project Approval (Feb 5)
Feb 9 - Feb 15	MCU Pin, Sensors and USB Micro B layout _____ PCB Design KiCad _____ Finish Project Proposal	Rawda Abdeltawab _____ Fiona Cashin _____ Whole group	First Meeting with TA (Feb 10th) _____ Project Proposal (Feb 13th)
Feb 16 - Feb 22	Finish PCB Design _____ Test Functionality of MCU with blinking LED _____ Test MCU with display _____ Finish team contract	Whole group _____ Rawda Abdeltawab _____ Rawda Abdeltawab _____ Whole group	PCB review (Feb 20th) _____ Team Contract (Feb 20th)
Feb 23 - Mar 1	Test temperature and humidity sensor with and and without display. _____	Rawda Abdeltawab Olivia Guido _____	Design Review Sign-up (Feb 27th)

	<p>Test Temperature and humidity sensor with Separate I2C Bus.</p> <hr/> <p>Test design ideas for 5V to 3.3V SDA/SCL From display to MCU</p> <hr/> <p>Assemble MCU+Display + temperature and humidity sensor on breadboard.</p> <hr/> <p>Finish Design Document</p>	<p>Rawda Abdeltawab</p> <hr/> <p>Rawda Abdeltawab</p> <hr/> <p>Rawda Abdeltawab Olivia Guido</p> <hr/> <p>Whole group</p>	<p>Design Document (Feb 27th)</p>
<p>Mar 2 - Mar 8</p>	<p>Test the power across connected components. Use both the power supply from the ECE445 lab and a 12V battery.</p> <hr/> <p>CO₂ Sensor Code Development</p> <hr/> <p>If there are any issues with the PCB, a new design is required by March 5th, 4:45pm</p> <hr/> <p>If no issues with PCB and it arrives by this week, begin soldering</p>	<p>Olivia Guido</p> <hr/> <p>Rawda Abdeltawab</p> <hr/> <p>Fiona Cashin Rawda Abdeltawab</p> <hr/> <p>Olivia Guido</p>	<p>Second Round PCBway Orders (Mar 5th)</p>
<p>Mar 9 - Mar 15</p>	<p>CO₂ & temperature and humidity Testing in Nano Tech building (with TA), and in ECEB 2070</p> <hr/> <p>Compare first round PCB to design (look for any imperfections)</p> <hr/> <p>Continue or start soldering on PCB (depending on if waiting for new PCB or have a correct one)</p>	<p>Olivia Guido Rawda Abdeltawab</p> <hr/> <p>Fiona Cashin</p> <hr/> <p>Olivia Guido</p>	<p>Third Round PCBway Orders (Mar 8th)</p> <hr/> <p>Last day for revisions to the machine shop (Mar 9th)</p>
<p>SPRING BREAK</p>	<p><i>Enjoy Spring Break while keeping track of any components that are being delivered during this time!</i></p>	<p>Whole group</p>	<p>.....</p>
<p>Mar 23 - Mar 29</p>	<p>Continue with testing</p> <hr/> <p>Adjust the code development as needed</p> <hr/> <p>Continue with soldering (or start depending on if waiting for new PCB)</p>	<p>Whole group</p> <hr/> <p>Rawda Abdeltawab</p> <hr/> <p>Olivia Guido</p>	<p>Fourth Round PCBway Orders (Mar 26th)</p>
<p>Mar 30 - Apr 5</p>	<p>Everyone should finalize their own individual progress reports in addition to other group duties</p> <hr/> <p>Schedule Demo with UIUC Bee Club</p> <hr/> <p>Continue working on PCB functionality / working out any issues with design</p> <hr/> <p>Add extra software features if time & hardware permits</p>	<p>Whole group</p> <hr/> <p>Olivia Guido</p> <hr/> <p>Whole group</p> <hr/> <p>Rawda Abdeltawab</p>	<p>Individual progress reports (April 1st)</p>
<p>Apr 6 - Apr 12</p>	<p>Everyone should individually complete Team Contract assessment in Canvas</p> <hr/> <p>Film demo with UIUC Bee Club (weather dependent)</p> <hr/> <p>Continue working on PCB functionality / working out any issues with design</p>	<p>Whole group</p> <hr/> <p>Olivia Guido</p> <hr/> <p>Whole group</p>	<p>Team Contract Assessment (April 10th)</p>

	Add extra software features if time & hardware permits	Rawda Abdeltawab	
Apr 13 - Apr 19	Film demo with UIUC Bee Club (if not done yet due to weather) Code cleaned up and commented	Olivia Guido Rawda Abdeltawab
Apr 20 - Apr 26	Before mock demo practice as a team Continue to work on Final Report	Whole group Whole group	Mock Demo (April 22nd, during weekly TA meeting) Mock Presentation (April 23th-April 24st)
Apr 27 - May 3	Continue to work on Final Report Prepare for the Final Demo	Whole group Whole group	Final Demos (April 27th - 29th) Final Presentation (April 30th-May 1st)
May 4 - May 7	Finalize and turn in Final Report Return LCD display to the lab	Whole group Whole group	Final Report Due (May 6th) Lab Checkout (May 7th) Lab Notebook Due (May 7th)

2.10 Risk Analysis

With beekeeping, there are many risks such as bee stings, burns from the use of smokers, and strain from handling heavy equipment. Since this is a handheld product, wearing protective equipment around the hives is necessary, as the sensors will need to be individually placed in the opening of the hive. While this is a product designed for the outdoors, it is not designed for standard commercial use in a variety of weather conditions. While each main component is protected by a casing, there are many ventilation holes and cutouts that would allow pollen, dust, or water to collect.

Beyond environmental hazards, since this is a powered product, there is always a risk of shock if the wires, PCB casing, or sensor casings are damaged. Additionally, with our selected rechargeable Li-ion battery, there are many manageable risks. If the battery overheats, is overcharged, or is damaged, it then poses a fire, explosion, and gas hazard. Our team reduces the risk of battery damage by securing the battery in place within the outer, handheld box.

3. Ethics and Safety

Humans need honeybees. The pollination helps increase the type and amount of plants and crops. This product will help society as it will allow hobby beekeepers continual success for keeping bees alive all year round. Increasing the presence of bees will increase pollination, which will thus increase the prosperity of plants and crops. This is especially important in current times as the number of honeybees are decreasing at a rapid rate due to “habitat loss, improper apiary management, pesticide usage, [and] climate change” (Mull et al., 2022). While we can’t control many of these environmental factors, our design aims to improve apiary care through a monitored hive (IEEE CoE 3).

This project was evaluated for ethical and safety concerns during development and potential misuse. The design aligns with the IEEE and ACM Codes of Ethics by prioritizing public welfare, honesty, competence, and harm prevention (IEEE CoE 1).

Ethically, the device aims to reduce stress on bee colonies by minimizing hive intrusion, benefiting the environment through improved pollination and honey production. The system does not collect personal data and presents minimal risk of misuse. Ethical risks are further reduced through accurate performance claims, clear documentation, and disclosure of system limitations (IEEE CoE 5).

Safety is a major focus of the project. All electronic components are properly insulated and enclosed to prevent electrical hazards, and materials are chosen for safe outdoor and agricultural use. Additionally, our design includes appropriate covers to further protect all components against harsher weather conditions. While the components are not claimed to be waterproof or water-resistant, there should be ample protection against damages from the hive’s internal environment.

Key safety practices include avoiding battery short circuits, keeping liquids away from electronics, preventing battery overload, using correctly sized wiring, monitoring heat from voltage regulators, ensuring correct battery polarity, and regularly inspecting for exposed wires. All components are housed in a protective enclosure to reduce shock risk. Additionally, all elements of the PCB in this design have been further inspected by a member in the University of Illinois’s ECE445 Staff.

All team members completed required laboratory safety training, including electrical safety. Overall, the project follows the IEEE Code of Ethics by prioritizing safety, accurately representing performance, working within the team’s expertise, and supporting respectful collaboration (IEEE CoE 6).

4. References

ACM Code of Ethics and Professional Conduct, ACM, 2018. OnlineOnlineOnline Available: <https://www.acm.org/code-of-ethics>

Adafruit Industries. (n.d.). *AM2301B - wired enclosed AHT20 - temperature and humidity sensor*. https://cdn-shop.adafruit.com/product-files/5181/5181_AM2301B.pdf

Beehive Monitoring, *Beehivemonitoring.com*, 2016. OnlineOnlineOnline Available: <https://beehivemonitoring.com/>

DFRobot Wiki. (n.d.). *CO2 Sensor SKU SEN0159*. https://wiki.dfrobot.com/CO2_Sensor_SKU_SEN0159

DigiKey. (n.d.-a). *4046* [Photograph]. <https://www.digikey.com/short/nfj42m9h>

DigiKey. (n.d.-b). *PTS645SL43-2 LFS* [Photograph]. <https://www.digikey.com/short/hrq8d0dp>

DigiKey. (n.d.-c). *SEN0159* [Photograph]. <https://www.digikey.com/short/8n857nj4>

Espressif Systems (n.d.). *ESP32-C3-WROOM-02 & ESP32-C3-WROOM-02U Datasheet*. https://documentation.espressif.com/esp32-c3-wroom-02_datasheet_en.html

HiLetgo. (n.d.). *DHT21 AM2301 Capacitive Digital Temperature Humidity Sensor* [Photograph]. <https://a.co/d/04i0vYrn>

HonestBee. (n.d.). *What are the dimensions of a medium bee box?* [Photograph]. <https://honestbeeltd.com/faqs/what-are-the-dimensions-of-a-medium-bee-box>

Hosyond. (n.d). *LCD Display Module 16x02 LCD Screen Module* [Photograph].
<https://www.amazon.com/Hosyond-Display-Module-Arduino-Raspberry/dp/B0BWTFN9WF?th=1>

IEEE. (2020). *IEEE Code of Ethics*. <http://www.ieee.org/about/corporate/governance/p7-8.html>

Kara98. (2014). *3349839-left-hand-outstretched.jpg* [Photograph] Wikimedia Commons.
<https://commons.wikimedia.org/wiki/File:3349839-left-hand-outstretched.jpg>

Mull, A., Gunnell, J., Hansen, S., Ramirez, R., Walker, A., Zesiger, C., & Spears, L. (2022, February). *Factors contributing to bee decline*. Utah State University Extension.
<https://extension.usu.edu/planthealth/research/factors-contributing-to-bee-decline>

Perfect Bee (2025). *A Guide to Beehive Entrance Management* [Photograph]
<https://www.perfectbee.com/your-beehive/starting-your-beehive/a-guide-to-beehive-entrance-management>.

Pk0001. (2019). *Arduino Pro Micro* [Photograph]. Wikimedia Commons.
https://commons.wikimedia.org/wiki/File:Arduino_Pro_Micro.jpg

Polycase. (n.d.-a). *DC-58P Heavy-Duty Electronics Enclosure* [Photograph].
<https://www.polycase.com/dc-58p>

Polycase. (n.d.-b). *LP-31P Plastic Electronics Box* [Photograph].
<https://www.polycase.com/lp-31p>

University of Illinois Urbana-Champaign. (n.d.). *Salary averages*.
<https://ece.illinois.edu/admissions/why-ece/salary-averages>