ECEN 5613 - Embedded System Design

SPRING 2023

Mining Environment Inspection Module

Vaishnavi Patekar

vaishnavi.patekar@colorado.edu

Mrunal Yadav

mrunal.yadav@colorado.edu

Final Project Report

University of Colorado, Boulder

May 5th, 2023

Table of Contents

1. Introduction

Today, the mining industry plays a very significant role in the whole country as the Geology and Mining department discovers all the major and minor minerals. Mining industries employ many mine workers for this purpose. However, many times the environment inside the mines isn't safe for humans to enter. Growing up back in India, we used to hear many news regarding the death of mine workers due to a hazardous mining environment. To avoid this, the inspection of the mine environment before the mine workers enter is very crucial. With the advent of technology these days, such problems can be resolved and the process can be made much more efficient and safer for the mine workers.

This project is inspired by this problem and we are trying our best to develop a system for the mine worker's safety.

1.1 System Overview

The system consists of a portable device that inspects the mining environment in order to decide whether the mine environment is safe for the mine workers or not. It determines the environmental conditions like the harmful gasses present, temperature, pressure, and humidity based on the sensor data. It also logs the environment data and uses this information for decision-making.

This module can be mounted over a Bluetooth-operated BOT and sent inside the mine for environmental inspection before mine workers enter the mine. Once the data is collected from the BOT module, decisions can be made regarding the mine workers' safety.

This system consists of hardware and software components. Figure 1 below shows the block diagram of the system.

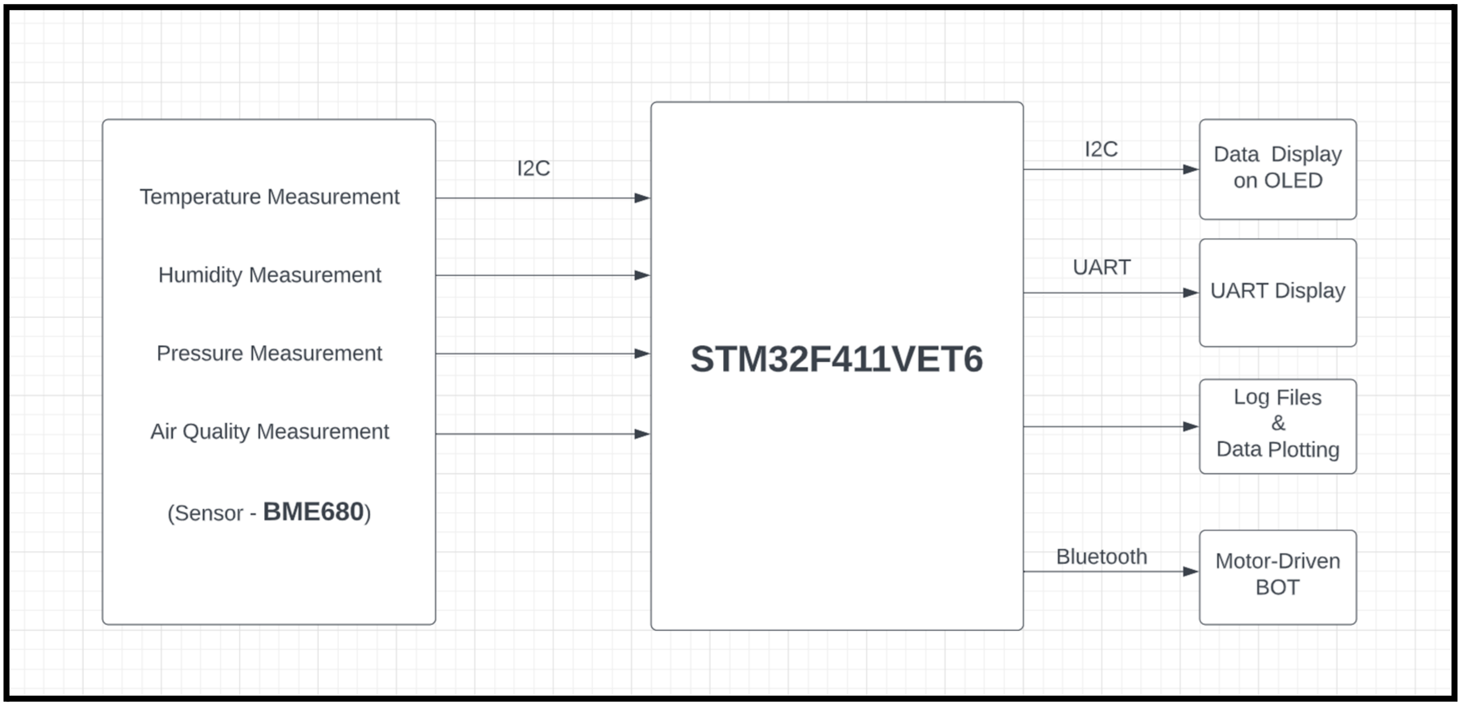


Figure 1: Block Diagram; Mining Environment Module

As shown in Figure 1 above, On the input side, it has a BME680 sensor that collects the temperature, humidity, pressure, and air quality data. STM32F411Vetx Discovery board is acting as a microcontroller and on the output side, the data is displayed over an OLED LCD screen and saved to the log files. This module has to be mounted over a BOT which will be controlled by Bluetooth.

2. Technical Description

2.1 Hardware Design

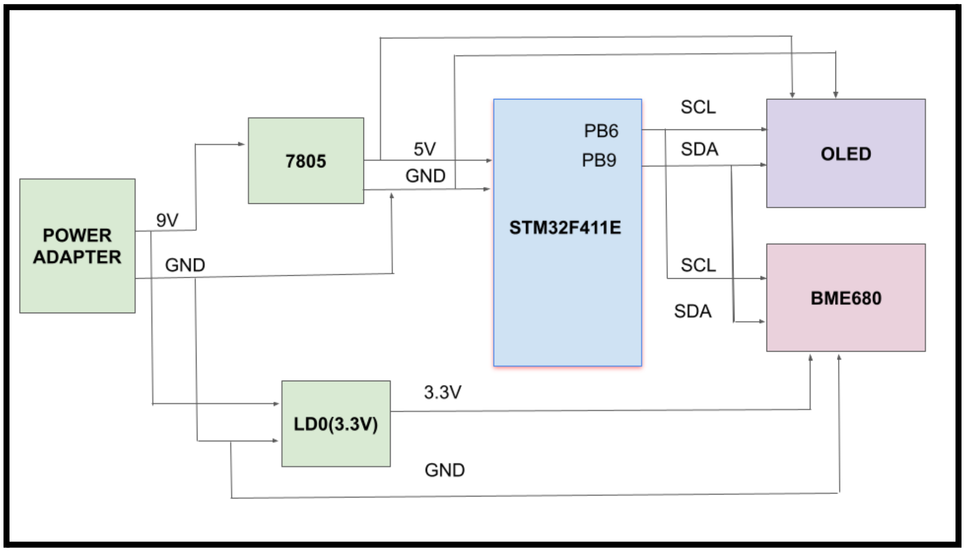


Figure 2: Hardware Design; Mining Environment Module

For powering the circuit, a power adapter of 9 V was used. As STM32 and OLED need 5V to power on, a 7805 Voltage regulator was used.

The 7805 is a type of linear voltage regulator integrated circuit that features three terminals, including an input pin, an output pin, and a ground pin. It is capable of accepting input voltages ranging from 7 to 35 volts and can regulate the output voltage to a constant 5 volts. The maximum output current of the 7805 voltage regulator is 1.5 amps that is sufficient to drive STM32 and all the sensors on the board. Moreover, BME680 requires 3.3 V to power on.

LD0 of 3.3 V was used for that purpose. An LDO 3.3V is a voltage regulator that maintains a fixed output voltage of 3.3 volts and is designed to operate with minimal voltage difference between the input and output. These regulators are commonly referred to as "low dropout" since they require very little voltage drop across the device to regulate the output voltage.

As shown in the figure, PB6 i.e. PORT B Pin 6 of STM32 was used as SCL to both OLED and BME680. Likewise, PORT B Pin 9 was used as SDA to both OLED and BME680.

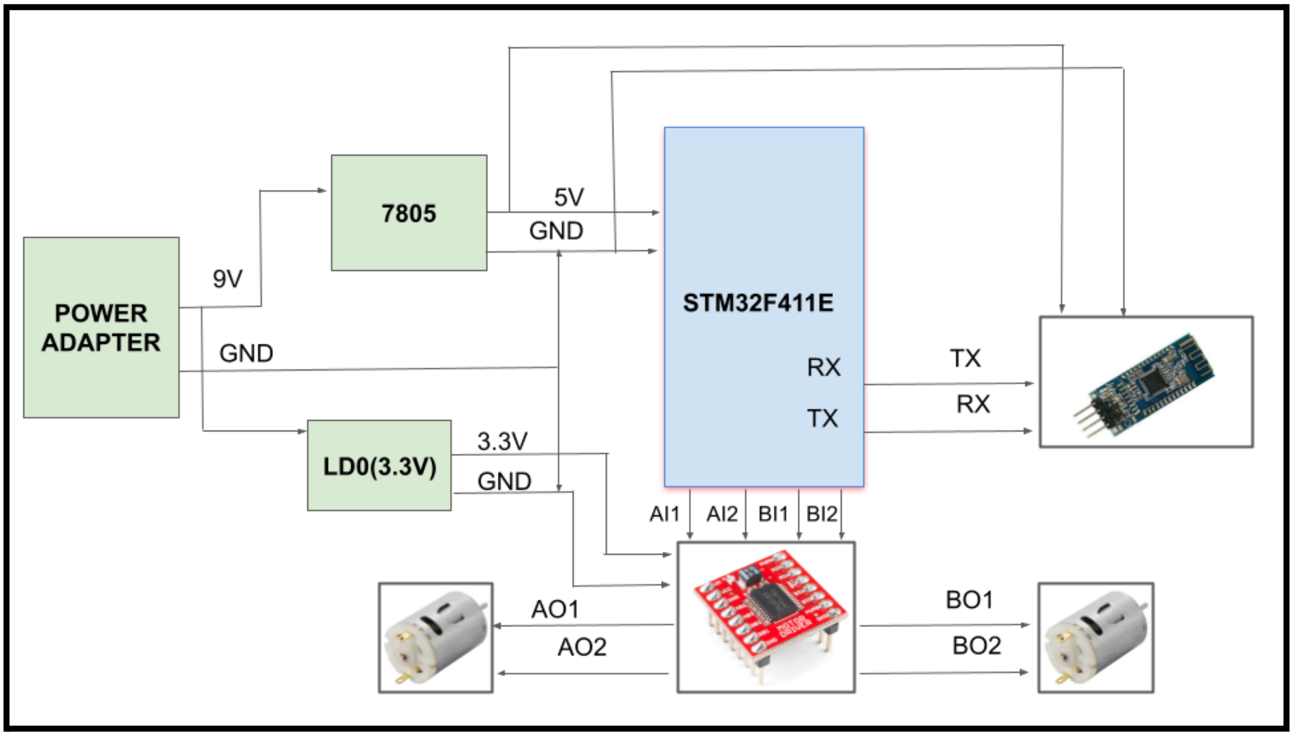


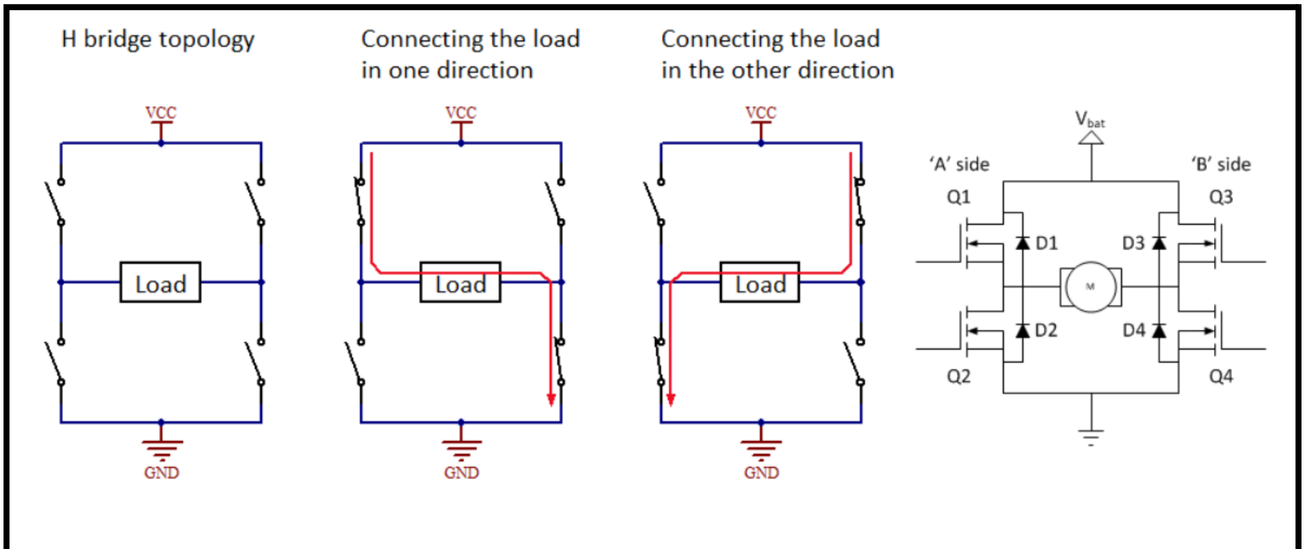
Figure 3: Hardware Design; Mining Environment Module

Similarly, for powering the bluetooth module HM-10 and STM32, a 7805 was used. The motor driver module needed 9V to drive the motors and 3.3 logic voltage was supplied. 3.3 V was required because the logical high of STM32 is 3.3 V. The logical voltage of STM32 refers to the voltage level used for digital logic signals in STM32 microcontrollers.

The STM32 microcontroller family is versatile and can support various digital logic standards, including TTL (Transistor-Transistor Logic). As per the datasheet, TTL logic uses bipolar transistors to implement logic gates, where a low voltage of approximately 0 volts represents a logical "0," and a high voltage of approximately 3.3 volts represents a logical "1."

STM32 microcontrollers offer GPIO pins that can be programmed to support multiple digital logic standards, including TTL, to connect with other digital logic devices, control external circuits, and read input signals. Apart from TTL logic, the STM32 microcontrollers also support other digital logic standards, such as CMOS (Complementary Metal-Oxide-Semiconductor) and LVCMOS (Low-Voltage Complementary Metal-Oxide-Semiconductor), which have different voltage levels and power consumption characteristics.

As shown in the figure, AI1 and AI2 are the inputs of channel A of the motor driver module and BI1 and BI2 are the inputs of channel B. One motor is connected to channel A and another to channel B. The PWM pin of the motor driver was set to high i.e 3.3 V to drive the motors. Motor driver module works on the H- bridge principle.



H-bridge logic circuit

An H-bridge is a circuit configuration used in motor drivers to regulate the speed and direction of DC motors. It comprises four switching elements arranged in the shape of an "H" and usually consist of MOSFETs or bipolar transistors. The center of the H-bridge is connected to the motor, while the four switching elements are linked to the motor and the power supply.

By switching the four elements in a specific pattern, the H-bridge circuit can apply a voltage of either polarity to the motor, resulting in a change in the direction of rotation or speed. This method allows for easy and precise control of the motor using simple control signals, such as PWM.

H-bridge motor drivers are extensively used in various fields, including industrial automation, robotics, and automotive applications. They offer several benefits, such as high efficiency, simple operation, and precise control over the motor's speed and direction.

However, H-bridge circuits come with some limitations, including the requirement for a separate power supply to drive the motor and the potential for high electrical noise and heat dissipation. To address these issues, some motor drivers may have added features like voltage regulation, overcurrent protection, and thermal protection.

Moreover, to drive the motors wirelessly, a bluetooth module HM 10 was used. Both the STM32 microcontroller and the Bluetooth module HM10 have TX (transmit) and RX (receive) pins that are utilized for serial communication. In the STM32 microcontroller, these pins are part of the USART peripheral and can be programmed for either asynchronous or synchronous serial communication.

The TX pin sends data from the microcontroller, while the RX pin receives data. The HM10 Bluetooth module also uses the TX and RX pins for serial communication via the UART protocol, with the TX pin transmitting data from the module and the RX pin receiving data. To establish serial communication between the STM32 microcontroller and the HM10 Bluetooth module, the TX pin of the microcontroller should be connected to the RX pin of the HM10 module, and the RX pin of the microcontroller should be connected to the TX pin of the HM10 module. This enabled the data transmission between the two devices.

2.1.1 Development Board (MCU) - STM32F411Vetx

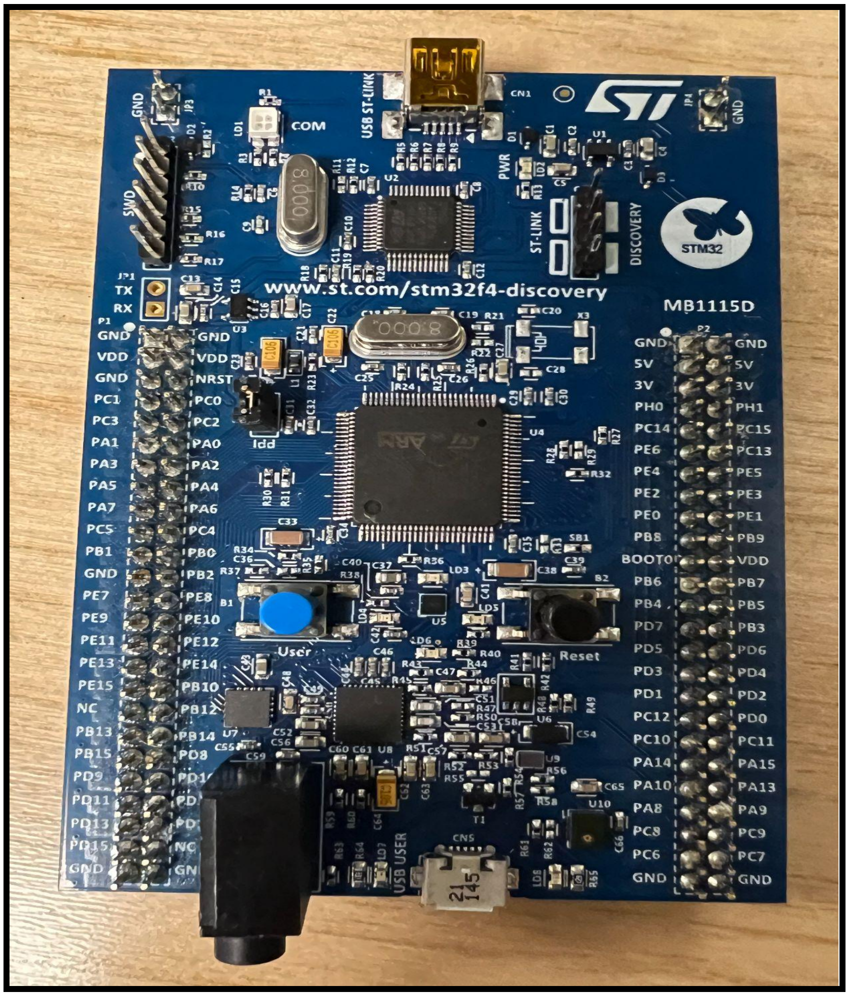


Figure 4: STM32F411VETx Board

STM32F411VETx is powered by the ARM Cortex-M4 core and has a clock speed of up to 100 MHz, 128 KB of flash memory, and 32 KB of SRAM. Additionally, it has numerous peripherals like ADCs, timers, USARTs, SPI, I2C, and USB interfaces, along with up to 100 GPIO pins. The microcontroller also has a 12-bit DAC with a maximum output voltage of 3.3V, a CRC calculation unit, watchdog timers, and a power management unit.

For this particular project, we have utilized the features like I2c communication, UART, and GPIO drive along with the BME680 sensor, OLED, and Bluetooth driver.

2.1.2 BME680 Sensor

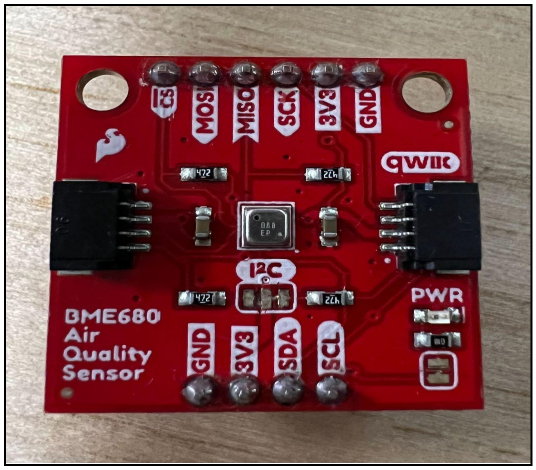


Figure 5: BME680 Sensor

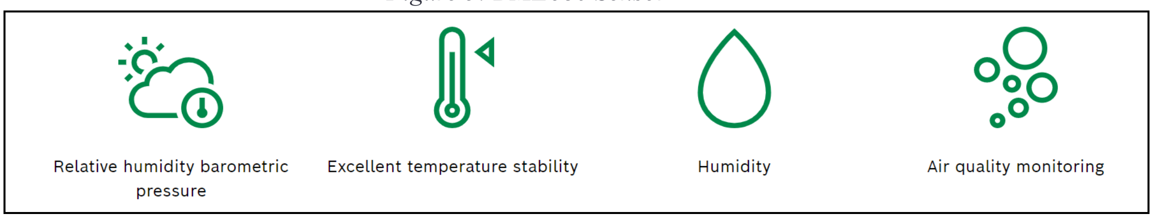


Figure 6: BME680 Sensor Capabilities

In a mining environment, to avoid harmful scenarios it is important to monitor the temperature, humidity, pressure, and the presence of the harmful gases.
In mining environment inspection, we would be mainly examining the following two scenarios:
1. Flammability
2. Harmful gases present in the environment

IAQ Index	Air Quality	Impact (long-term exposure)	Suggested action
0 – 50	Excellent	Pure air; best for well-being	No measures needed
51 – 100	Good	No irritation or impact on well-being	No measures needed
101 – 150	Lightly polluted	Reduction of well-being possible	Ventilation suggested
151 – 200	Moderately polluted	More significant irritation possible	Increase ventilation with clean air
201 – 250	Heavily polluted	Exposition might lead to effects like headache depending on type of VOCs	optimize ventilation
251 – 350	Severely polluted	More severe health issue possible if harmful VOC present	Contamination should be identified if level is reached even w/o presence of people; maximize ventilation & reduce attendance
> 351	Extremely polluted	Headaches, additional neurotoxic effects possible	Contamination needs to be identified; avoid presence in room and maximize ventilation

Table 1: Index for Air Quality Classification & Colour Coding

Referring to Table 1 above, it can be observed that when it is highly harmful, this sensor detects the scenario.
The BME680 is a digital sensor that has all these capabilities, hence we chose this sensor for our project

- Gas sensor measuring relative humidity, barometric pressure, ambient temperature, and gas (VOC, Volatile Organic COmponents)
- Operates through I2C and SPI interfaces
- It has a wide operating temperature range and provides fully calibrated and compensated sensor readings with low noise measurement capabilities. The BME680 sensor has a built-in heater for air quality measurements.
- Specially developed for mobile applications and wearables where size and low power consumption are critical requirements
- Air quality is decided by the concentration of bVOC mixture with nitrogen as carrier gas.

Molar fraction	Compound	Production tolerance	Certified accuracy
5 ppm	Ethane	20 %	5 %
10 ppm	Isoprene /2-methyl-1,3 Butadiene	20 %	5 %
10 ppm	Ethanol	20 %	5 %
50 ppm	Acetone	20 %	5 %
15 ppm	Carbon Monoxide	10 %	2 %

Table 2: bVOC Mixture with Nitrogen as Carrier Gas

Following are some of the important parameters of BME680:

Parameter	Technical data
Package dimensions	8-Pin LGA with metal 3.0 x 3.0 x 0.93 mm³
Operation range (full accuracy)	Pressure: 300...1100 hPa Humidity 0...100% Temperature: -40...85°C
Supply voltage VDDIO Supply voltage VDD	1.2 ... 3.6 V 1.71 ... 3.6 V
Interface	I²C and SPI
Average current consumption (1Hz data refresh rate)) Average current consumption in sleep mode	2.1 µA at 1 Hz humidity and temperature 3.1 µA at 1 Hz pressure and temperature 3.7 µA at 1 Hz humidity, pressure and temperature 0.09–12 mA for p/h/T/gas depending on operation mode
Gas sensor Response time (τ 33-63%) Sensor-to-sensor deviation Power consumption Output data processing	 < 1 s (for new sensors) +/- 15% +/- 15 < 0.1 mA in ultra-low power mode direct output of IAQ: Index for Air Quality
Humidity sensor Response time (τ0-63%) Accuracy tolerance Hysteresis	 8 s ± 3 % relative humidity ≤ 1.5 % relative humidity
Pressure sensor RMS Noise Sensitivity Error Temperature coefficient offset	 0.12 Pa (equiv. to 1.7 cm) ± 0.25 % (equiv. to 1 m at 400 m height change) ±1.3 Pa/K (equiv. to ±10.9 cm at 1°C temperature change)

We have interfaced BME680 with microcontroller STM32F411Vetx, via I2C, with pins PB6 → SCL and PB9 → SDA.

2.1.3 OLED - SSD 1306

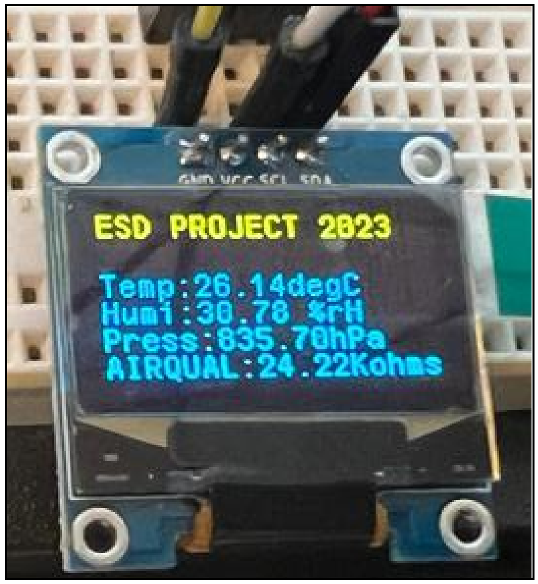


Figure 7: OLED, SSD1306

The SSD1306 is a well-known OLED display controller that has the ability to support a variety of OLED display sizes and interfaces, including both I2C and SPI. It can be used to display text, images, and graphics on a monochrome OLED screen, and includes features like low power consumption, a built-in charge pump for driving OLED pixels, and a 16-level grayscale display. The controller has an internal RAM buffer that allows for display data storage and can be used to display both text and graphics.

When working with an STM32 microcontroller and the SSD1306 OLED controller, it is necessary to connect the OLED display to the microcontroller using either I2C or SPI. Additionally, software must be written to control the display and send display data to the SSD1306 controller.

In our application, the temperature, humidity, pressure, and air quality are displayed and updated on the LCD screen every 10s.

2.1.4 Bluetooth Driver - HM10

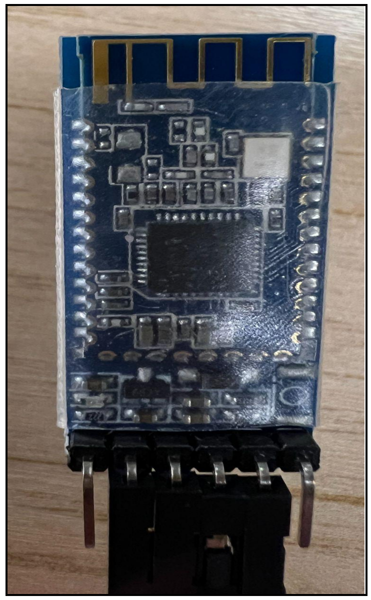
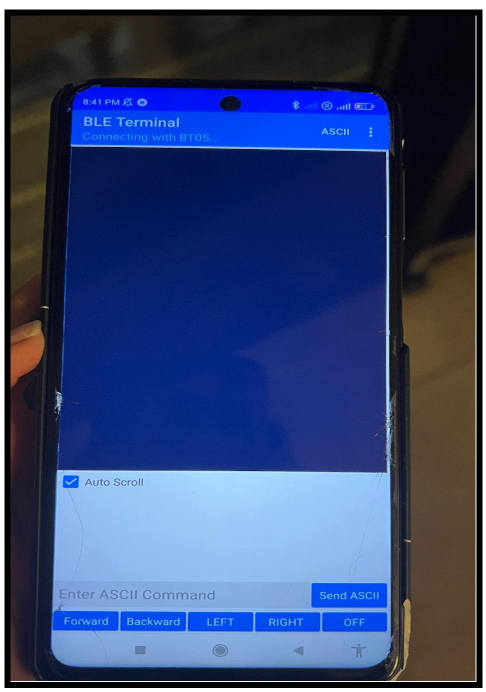


Figure 8: Bluetooth Module, HM-10

The HM-10 is a small and low-power Bluetooth Low-Energy (BLE) module that can be easily integrated into various electronic devices to enable wireless communication via Bluetooth using the BLE 4.0 protocol. With a UART interface, the module can be easily connected to a microcontroller or other electronic device. It supports configurable transmission power levels and a simple command-based interface for configuration and control. B;E Terminal mobile application was used for wireless communication with HM 10.



2.1.5 Motor Driver - TB6612FNG

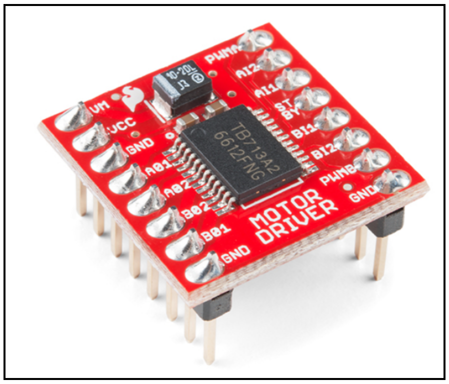


Figure 9: Motor Driver, TB6612FNG

The Motor Driver - Dual TB6612FNG is a motor control board designed for use in robotics and other motor control applications. It is small, inexpensive, and can control two DC motors or a single stepper motor with a maximum current of 1.2A per channel. The board features low standby current consumption, an input voltage range of 2.7V to 5.5V, and built-in thermal shutdown and overcurrent protection. The motor driver can be controlled through a microcontroller or other control device using the I2C or PWM interface. Due to its ease of use, small size, and affordability, the SparkFun Motor Driver - Dual TB6612FNG is popular among hobbyists and makers and is suitable for various applications, including robotics and remote control systems.

2.2 PCB Board Design

2.2.1 Schematic

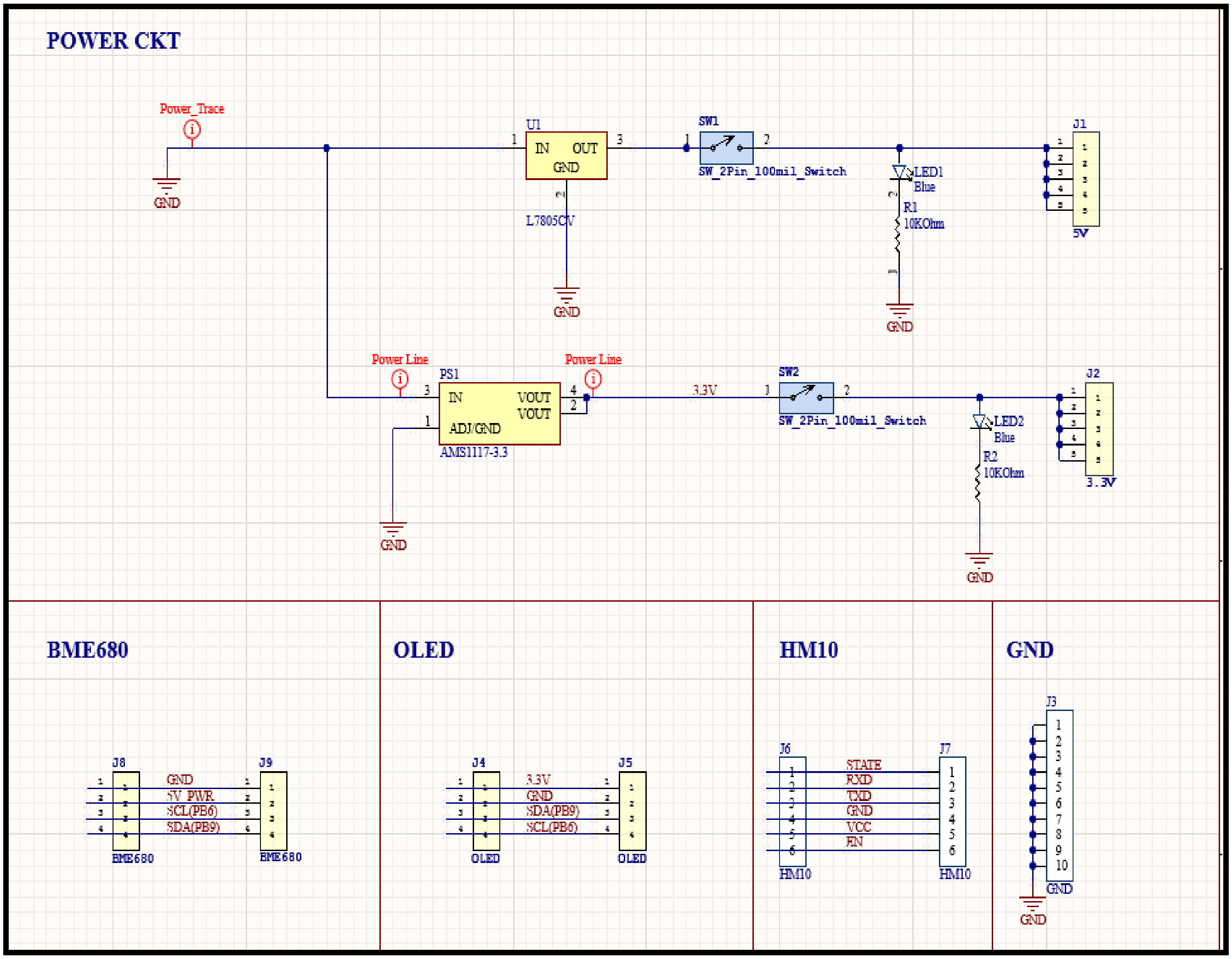


Figure 10: Altium Schematic of the module

2.3 Solderless Breadboard Version of the Project Module

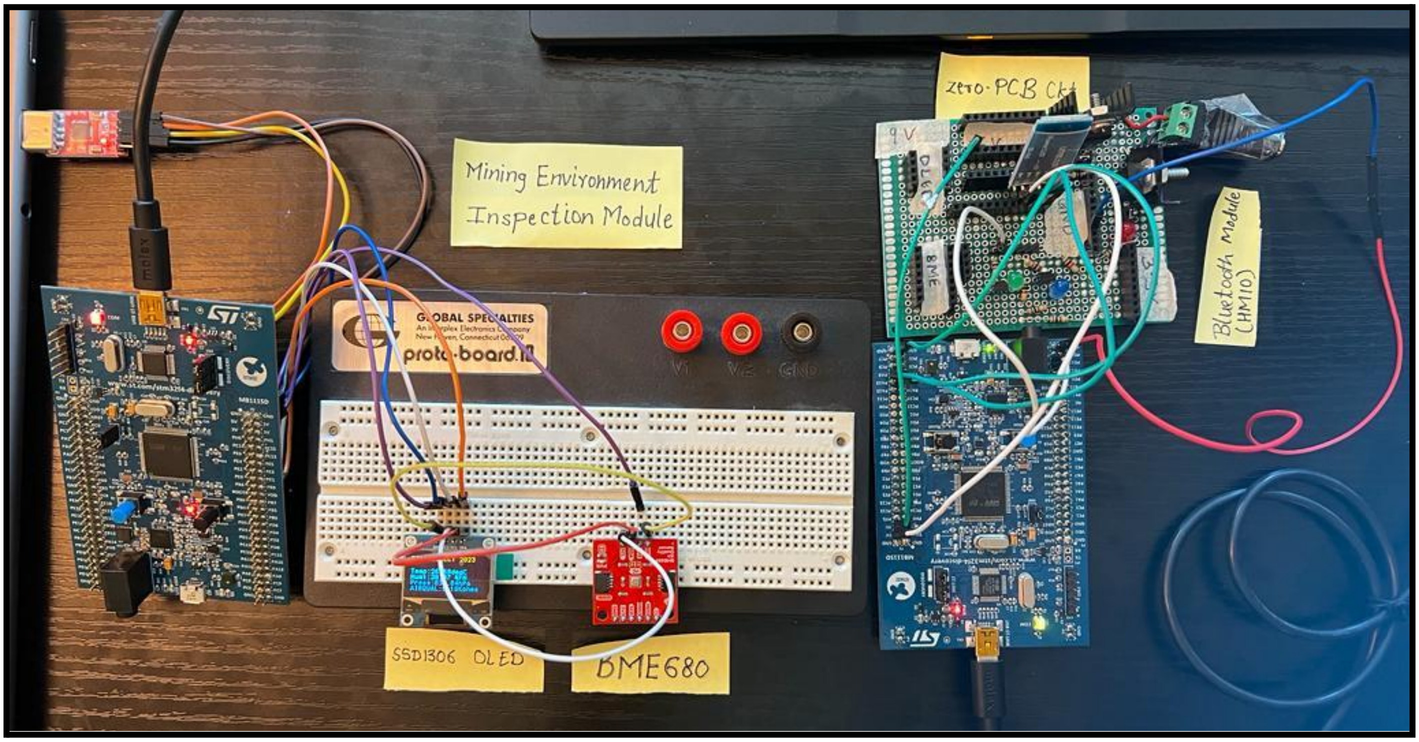
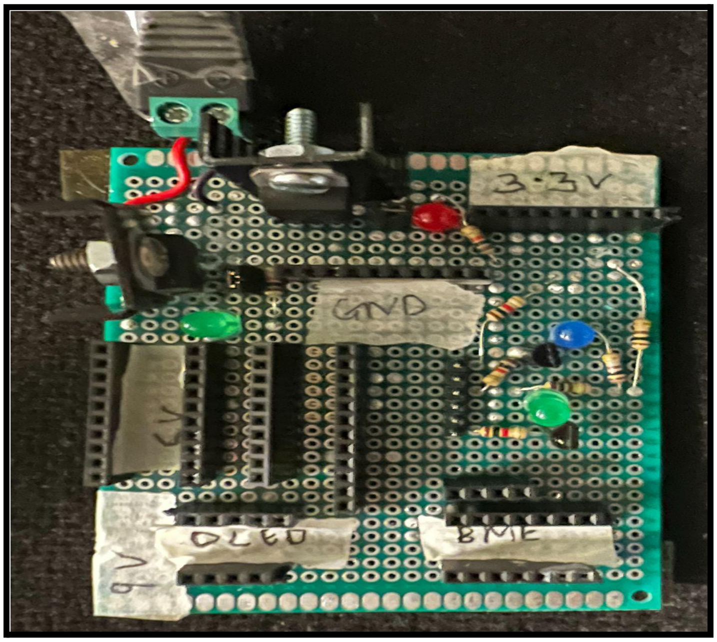


Figure 12: Solderless breadboard Version of Mining Environment Inspection Module

Figure 2.3 above shows the solderless breadboard version of the Mining Environment Inspection module, where STM32F411Vetx microcontroller is driving the BME680 sensor, OLED and the Bluetooth driver HM10.

After testing the solderless breadboard version, the module was developed and tested over a zero-PCB board as shown in the figure below:

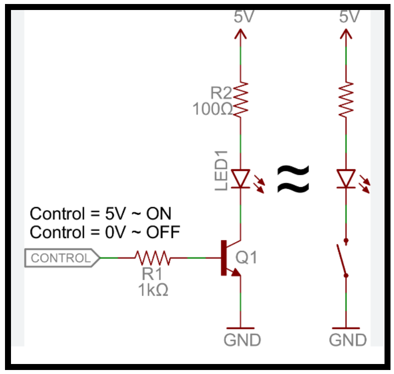


As we can see in the image, a 2 Pin power connector(green) was used to get battery or adapter input voltage. Red wire represents positive voltage and black as ground. 7805 and LDO(3.3V) were soldered and heat sinks were connected. A heatsink is a cooling device that does not require power to function and is used to transfer heat away from a hot surface or component. These devices are usually made of thermally conductive materials like copper or aluminum and their primary purpose is to increase the surface area available for heat transfer to the surrounding air.

To isolate the power circuitry with the signal 2 pin jumper is used. After connecting power, indicator LEDs were soldered along with 1 K current limiting resistor. LEDs are commonly used in electronic circuits as indicators to provide visual feedback to the user. They are low-power devices and emit light when a current passes through them in the forward direction.

A resistor was connected to the cathode (negative) of an LED (Light Emitting Diode) in a circuit to limit the amount of current flowing through the LED. This is done to protect the LED from damage due to excessive current, as LEDs have a maximum current rating. We used a 1K resistor to limit the current.

After soldering all the female headers for all the respective voltages and sensors, bluetooth module and OLED testing was done whether there were no shorts between VCC and ground. On the right hand side of the board we can see transistors, leds and resistors are soldered. This was done to test whether the bluetooth module can be used as a transistor as a switch to drive the leds. The ability to control the on/off state of a transistor makes it useful in a variety of applications, such as controlling the brightness of an LED or turning on and off a motor. To use a transistor as a switch, a resistor is often connected between the base and the control signal to limit the amount of current flowing through the transistor and protect it from damage.



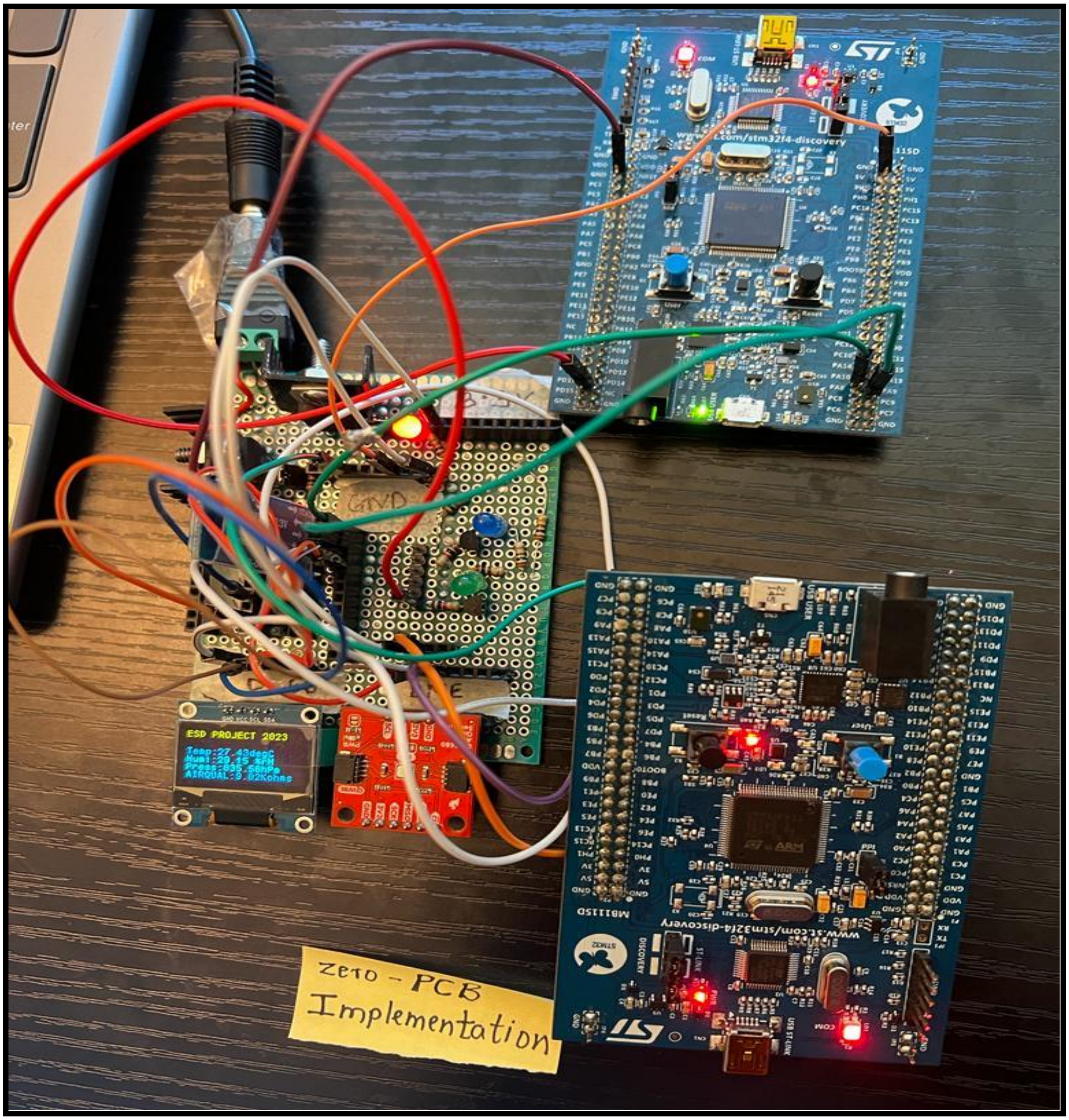


Figure 13: Zero-PCB Version of Mining Environment Inspection Module

2.4 Data Flow Design

Figure 14 below shows the data flow diagram for the system.

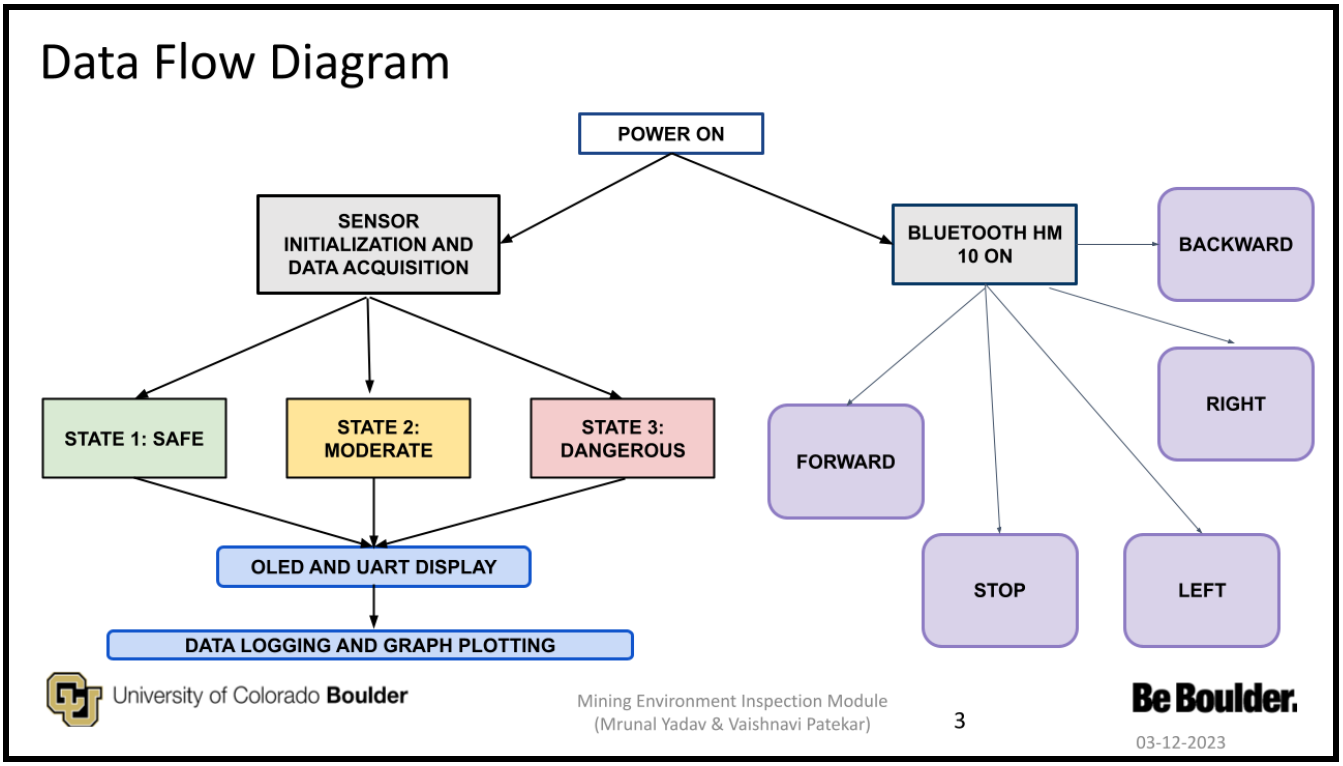
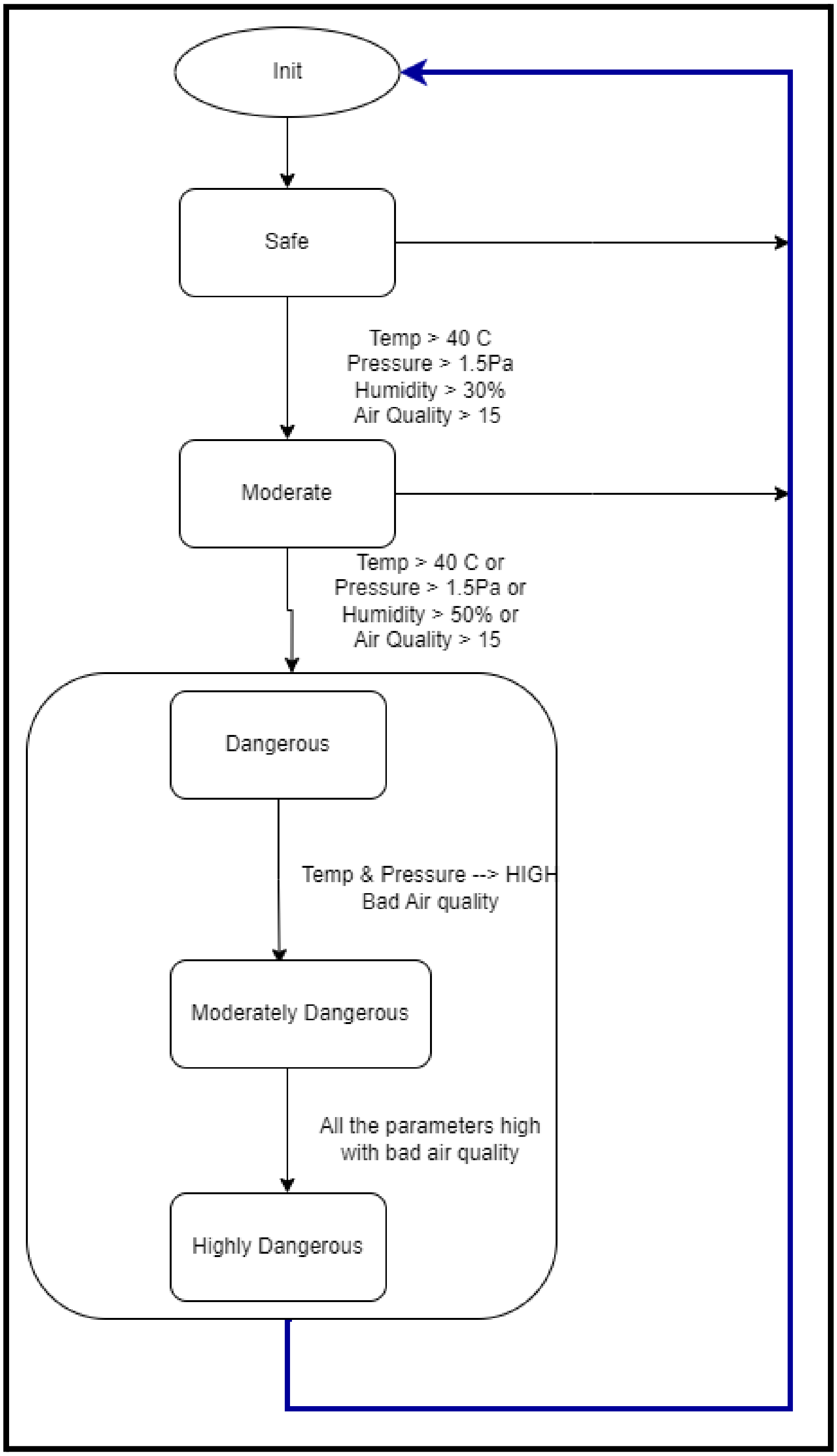


Figure 14: Data flow diagram for the system of the module

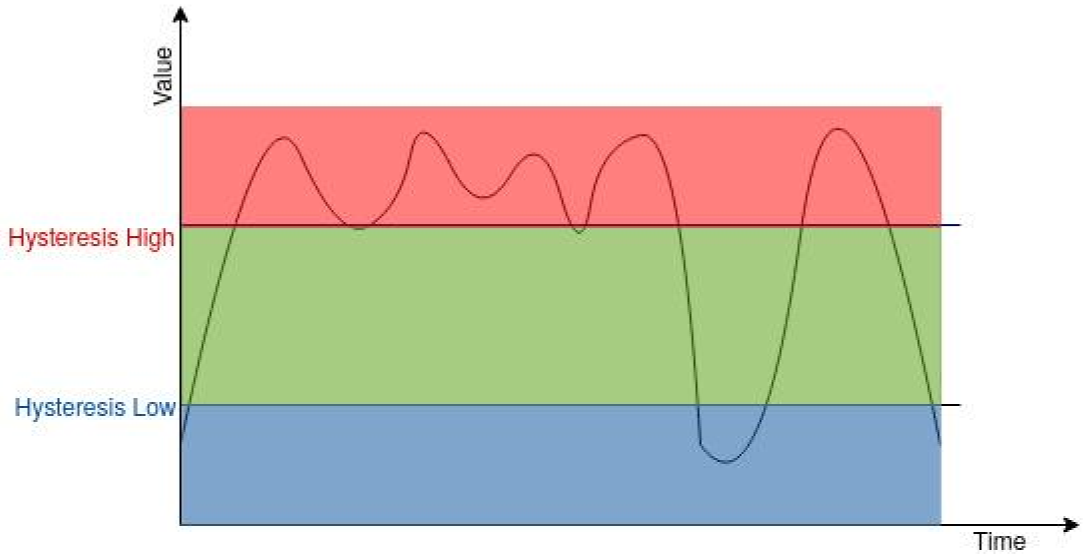
As shown in Figure 14, all the sensors and modules are initialized after the power is on. BME680 sensor starts reading the data for every 1 second. BME680 sensor displays the temperature, pressure, humidity, and air quality data over an OLED display. Based on the acquired data, we have divided the scenarios into 5 states as follows:

the state machine will enter a specific state as explained in section 2.2.1 below. Depending on the state, the UART console and LCD will display safe/moderate/dangerous messages as well as the data will be stored in log files after a buffer is full with 1000 readings.

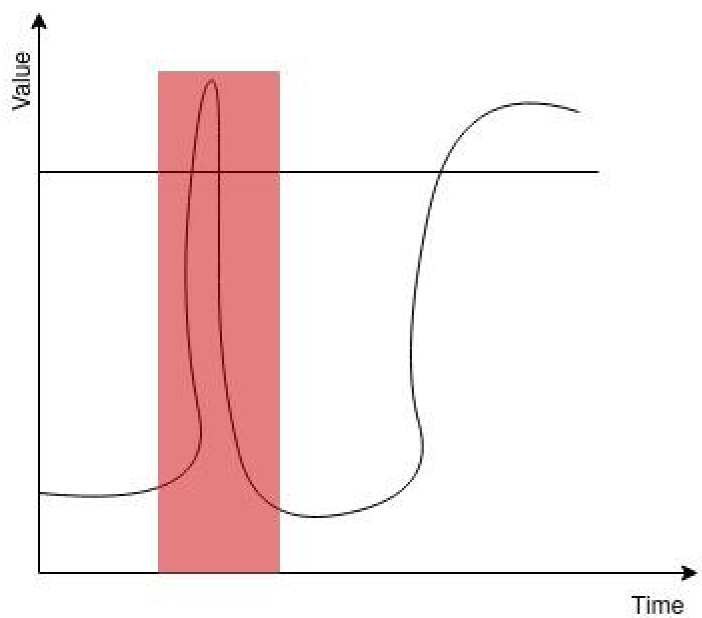
2.5 State Machine Design



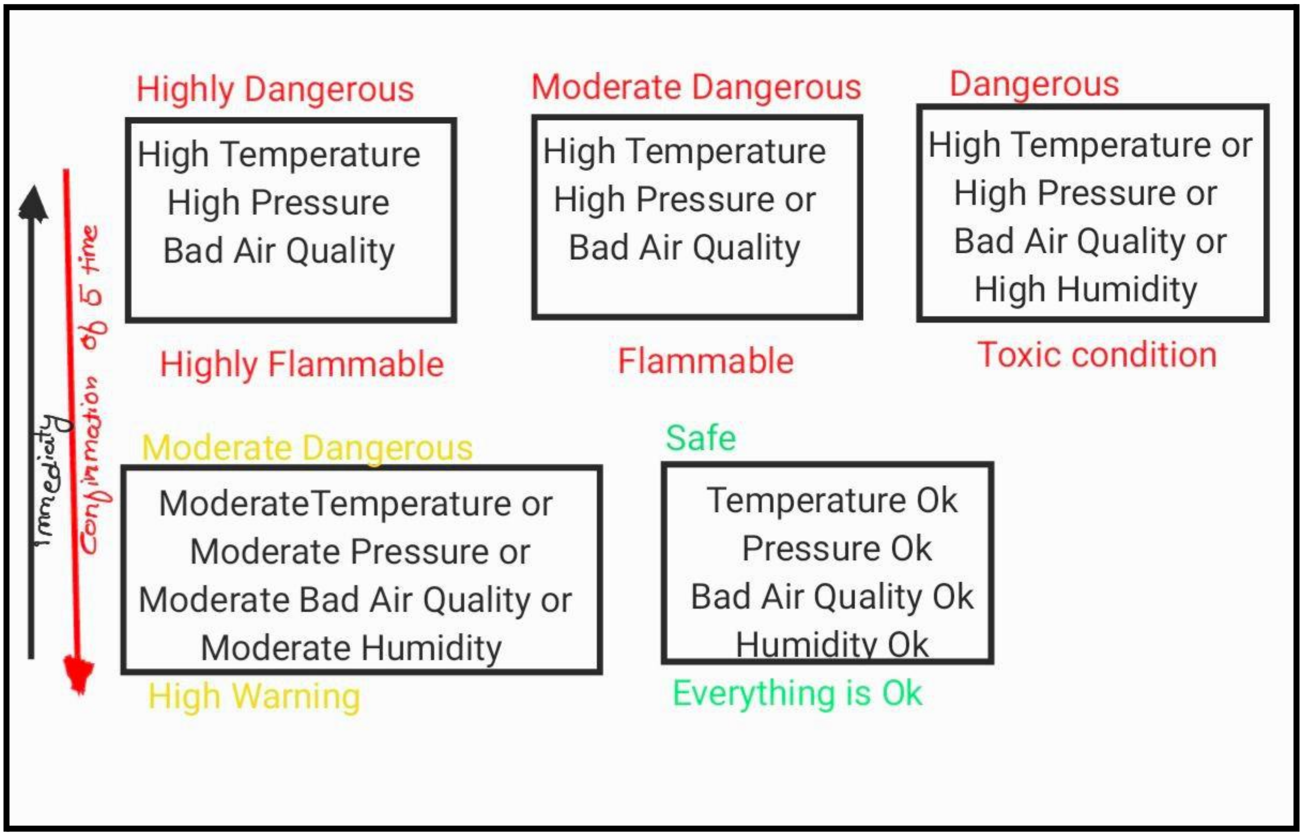
2.5.1 Hysteresis Function



Hysteresis is used to avoid state change in case of value fluctuation between the high Threshold value. So the value of Hysteresis will only change back when it moves below Low(High-Delta)[Blue Region]or above High[Red Region]. In between which is the green Region the value would not change but retain the value of the past value.



This is the is the case while there a glitch and value goes above a threshold only for a moment.To resolve the same it's take care the it have a value above a threshold for at least for certain consecutive time steps(considering the threshold of hysteresis)



2.5 Firmware Design

Firmware of this project includes device drivers for the sensor BME680, OLED, and the Bluetooth module HM10. We have implemented a state machine for processing the data and for decision-making. In addition, we have developed a Python script that saves the data to the log files and also shows the graphical representation of the data.

STM32CUBE IDE was used for the development of the firmware of BME280, OLED, and the Bluetooth module, whereas Python IDE was used to develop the data logging script.

The source code can be found in Appendix 2.3.

2.5.1 BME680 Driver

The BME680 driver, bme680.h contains the routines for BME680 sensor operations that are listed in the table below

Function Name	Description
bme680_init	Initializes the sensor i.e. reads the chip-id and calibration data from the sensor

bme680_set_regs	reads the data from the given register address of the sensor
bme680_soft_reset	performs the soft reset of the sensor.
bme680_set_sensor_mode	To set the power mode of the sensor
bme680_get_sensor_mode	To get the power mode of the sensor
bme680_set_profile_dur	To set the profile duration of the sensor
bme680_get_profile_dur	To get the profile duration of the sensor
bme680_get_sensor_data	reads the pressure, temperature and humidity and gas data from the sensor, compensates the data and store it in the bme680_data structure instance passed by the user
bme680_set_sensor_settings	To set the oversampling, filter and T, P, H, gas selection settings in the sensor
bme680_get_sensor_settings	To get the oversampling, filter and T, P, H, gas selection settings in the sensor

Table 2.3: BME680 Driver Functions Header (bme680.h)

BME480.c

Function Name	Description
get_calib_data	to read the calibrated data from the sensor
set_gas_config	to set the gas configuration of the sensor
get_gas_config	to get the gas configuration of the sensor
calc_heater_dur	to calculate the Heat duration value
calc_temperature	to calculate the temperature value
calc_pressure	to calculate the pressure value
calc_humidity	to calculate the humidity value
calc_gas_resistance	to calculate the Gas Resistance value
calc_heater_res	to calculate the Heat Resistance value

read_field_data	to calculate the field data of sensor
set_mem_page	to set the memory page based on register address
get_mem_page	To get the memory page based on register address
null_ptr_check	to validate the device pointer for null conditions
boundary_check	to check the boundary conditions

Table 2.4: BME680 Driver Functions (bme680.c)

2.5.2 OLED - SSD1306 Driver

Ssd1306.h

Ssd1306.h contains the macros for ssd address, width, and height along with the following function:

Function Name	Description
SSD1306_Init	Initializes SSD1306 LCD
SSD1306_UpdateScreen	Updates buffer from internal RAM to LCD
SSD1306_ToggleInvert	Toggles pixels invertion inside internal RAM
SSD1306_Fill	Fills entire LCD with desired color
SSD1306_DrawPixel	Draws pixel at desired location
SSD1306_GotoXY	Sets cursor pointer to the desired location for strings
SSD1306_Putc	Puts character to internal RAM
SSD1306_Puts	Puts string to internal RAM
ssd1306_I2C_Init	Initializes SSD1306 LCD
ssd1306_I2C_Write	Writes a single byte to slave

SSD1306_Clear	clear the display

Ssd1306.c

SSD1306_ScrollRight	To scroll to the right address on the LCD
SSD1306_ScrollLeft	To scroll to the left address on the LCD
SSD1306_Scrolldiagright	To scroll to the diagonal right address on the LCD
SSD1306_Scrolldiagleft	To scroll to the diagonal left address on the LCD

2.5.3 Statemachine.c

Statemachine.c contains the basic decision-making logic of this project based on the temperature, pressure, humidity, and air quality data.
It also contains the macros that define the threshold and offset values of the temperature, pressure, and humidity parameters.
As explained in section 2.2, it contains 5 states: Safe, moderate, dangerous, moderately dangerous, and highly dangerous.
Depending on the state, it turns on the specific LED as explained in the algorithm. The code snippet for it can be referred from Annexure 2.3.3.

2.5.4 Bluetooth main.c

Main.c contains a UART handle for tx rx communication and logic to drive the LEDs and motor driver module.

2.4 Data Display & Data Logging

Ambient Temperature, pressure, humidity, and air quality data acquired by the sensor and the decision made by the module is displayed over a serial UART, and the data is saved into a log file.
We have written a Python script to automate the data collection from a serial UART to save it to the log files. The data is also live plotted in the form of a graph as shown in figure 2.4 below.

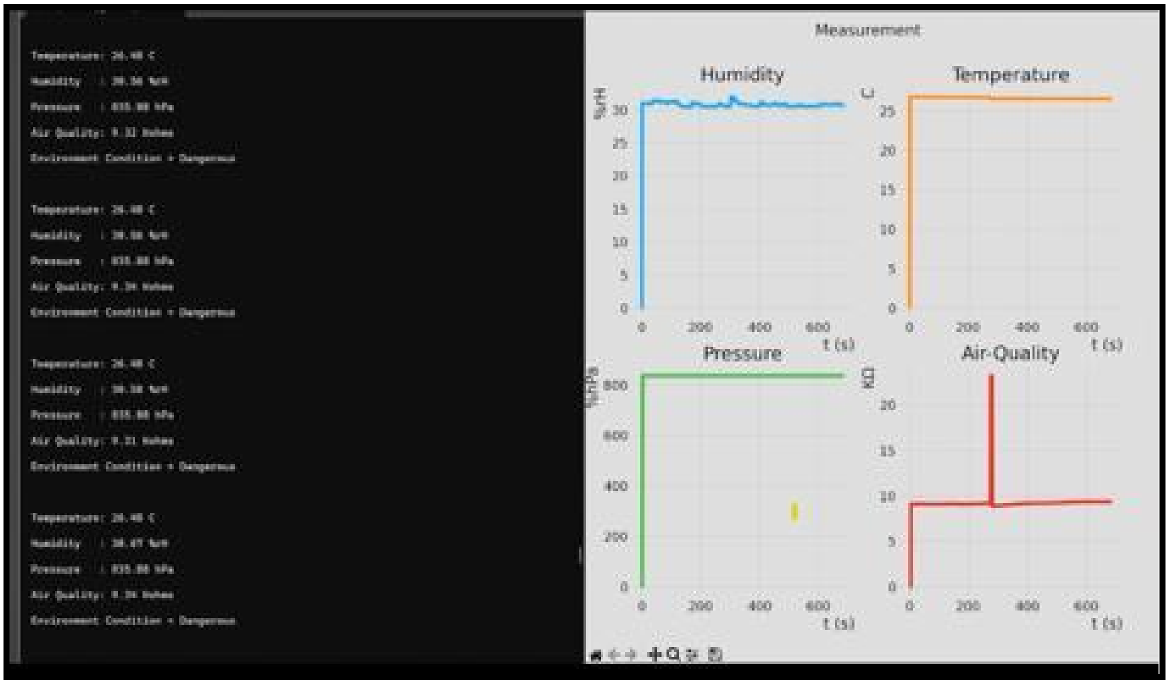


Figure 2.5: Serial data display and the data plot

The log files get automatically created after every 1000 samples collection. These log files can be referred from Appendix 4.

MISTAKES & LESSONS LEARNT

- We dedicated less time for PCB board development and testing, as a result, we fell short of time when our 1st PCB board did not work as expected and due to the long delivery timeline from the PCB vendor, we could not fix and re-order our PCB board. Hence, the lesson learnt here was to always estimate more time for testing and considering the timeline for re-ordering the components.
- Even though our SD card implementation did not work due to an SD card reader hardware issue, we could thoroughly understand the SD card configuration as well as the FATFS file system.
- Initially, we tried to drive motors using L298 IC but to make logical high transistors were used by connecting collector 5 V to motor inputs which was increasing the hardware. Hence, we decided to use the motor driver module.
- Both Sensor module, OLED can be integrated with Bluetooth on a single STM32. Due to time constraint, integration testing was not done.
- Along with the many software, hardware, and firmware concepts, We learnt the product development process right from the proposal, requirement analysis, and project planning to the integration testing.
- More time should be dedicated for the integration testing and enhancement of the features after the integration testing.
- Significant amount of time should be dedicated for the documentation of the project such as report writing.

CONCLUSION

As electrical engineers and firmware enthusiasts, this project gave us a complete zest of the firmware development process. While working on this project, we explored the software and hardware concepts related to the project. Along with the software, and hardware development, we learnt communication protocols like UART, i2c, SPI. Also, we learnt the importance of unit testing and integration testing.

In summary, we were successfully able to implement the ideas proposed in the PDR. With some more enhancements and features, our project can be turned into an actual product for Mining environment inspection. Also, our current implementation can be used for home ambient temperature, pressure, humidity, and air quality measurement.

FUTURE SCOPE/DEVELOPMENT IDEAS

Most of our proposed ideas were implemented within the allocated project timeline. However, there is still a lot of future scope that can be covered. Because of a faulty PCB, we had to implement our idea with the help of zero PCB board.

In the future, we would like to design and complete the PCB of our molecule and mount it on a Bluetooth-controlled BOT. Within the given project timeline, we were able to develop a bare BOT with wheels and motors controlled by a Bluetooth driver. In the future, we would like to develop a complete BOT that can be actually used as a product for the mining environment inspection.

Due to the limitation of the test environment, we could not use NOX, or CO sensors to test the harmful gasses present in the environment. However, it could be one of the parameters of the future scope.

ACKNOWLEDGEMENT

We would like to thank Professor Linden McClure for the well-designed course structure, content, and all the firmware concepts teaching.

We would like to thank our TAs Maanas, Sanish, and Jordi for their support and assistance in debugging.

Finally, we would like to acknowledge and thank the authors of the various sources cited in the references below.

REFERENCES

[1] https://everythingcivic.com/a-complete-guide-to-make-mining-inspection-process-easy/

[2] https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme680-ds001.pdf

[3] https://www.digikey.com/htmldatasheets/production/2047793/0/0/1/ssd1306.html

[4]https://www.st.com/content/ccc/resource/technical/document/datasheet/b3/a5/46/3b/b4/e5/4c/85/DM00115249.pdf/files/DM00115249.pdf/jcr:content/translations/en.DM00115249.pdf

[5] https://www.youtube.com/watch?v=ukdR-yoFcMI&ab_channel=MicrocontrollersLab

[6]https://www.digikey.com/en/maker/projects/getting-started-with-stm32-i2c-example/ba8c2bfef2024654b5dd10012425fa23

[7] https://deepbluembedded.com/stm32-hc-05-bluetooth-module-examples/

[8]https://www.sparkfun.com/products/14450?gclid=Cj0KCQjwu-KiBhCsARIsAPztUF0lVSoxmHxAfibLobLU8Tlg6KhSIymuOcoUYP3htjYJgpOzzITdur0aAn_4EALw_wcB

[9]https://circuitdigest.com/microcontroller-projects/how-to-use-arduino-and-hm-10-ble-module-to-control-led-with-android-app

[10]https://www.hnhcart.com/blogs/learn/2n2222-transistor-and-its-applications

APPENDICES

All the topics mentioned in the Appendix can be found in following GitHub link:
https://github.com/vaishnavi-p/ESD-Final-Project

Demo Video Link

https://drive.google.com/file/d/1UTJGVD3aQWLQJhB3bxff05D4mffDJv_Y/view?usp=sharing

Bill of Materials

Log Files

Software Souce Code

Data Sheets and Application Notes

Demo Video Link

HONOR CODE PLEDGE

ECEN 5613 Final Project

By signing this sheet, students certify that the work they submit is their own, and that they have clearly identified any code, schematics, design details, documentation, pictures, or other work obtained from another source. Each student on a project team is expected to certify their work by signing this pledge.

Honor Code Pledge: "On my honor, as a University of Colorado student, I have neither given nor received unauthorized assistance on this work. I have clearly acknowledged work that is not my own."

Student 1 Name (printed): Vaishnavi Patekar

Student 1 Signature:

Date: 05/05/2023

Student 2 Name (printed): Mrunal Yadav

Student 2 Signature:

Date: 05/05/2023