CONCEPTUALIZATION, PLANNING AND PROGRAMMING OF AN ENVIRONMENTAL MONITORING STATION USING ARDUINO IDE AND NEXTION EDITOR

A THESIS SUBMITTED TO
THE FACULTY OF ARCHITECTURE AND ENGINEERING
OF
EPOKA UNIVERSITY

BY

DORJAN DAFKU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRONICS AND COMMUNICATION ENGINEERING

JUNE, 2024

Approval sheet of the Thesis

This is to certify that we have read this thesis entitled **"Conceptualization, Planning and Programming of an Environmental Monitoring Station using Arduino IDE and Nextion Editor"** and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

____

Assoc. Prof. Dr. Arban Uka
Head of Department
Date: June, 28, 2024

Examining Committee Members:

Prof. Dr. Gezim Karapici (Computer Engineering) ________________

Prof. Dr. Betim Çiço (Computer Engineering) ________________

Assoc. Prof. Dr. Arban Uka (Computer Engineering) ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name Surname: Dorjan Dafku

Signature: ______________

ABSTRACT

CONCEPTUALIZATION, PLANNING AND PROGRAMMING OF AN ENVIRONMENTAL MONITORING STATION USING ARDUINO IDE AND NEXTION EDITOR

Dafku, Dorjan

M.Sc., Department of Computer Engineering

Supervisor: Prof.Dr. Betim Çiço

This paper explores the transformative role of microcontrollers and sensors in modern electronic systems, with a focus on environmental air quality measurement. By leveraging the programmability and integration capabilities of components like the Arduino Pro Mini, coupled with various air quality sensors (e.g., PM2.5, CO2, humidity, temperature), we illustrate how these technologies enhance efficiency, accuracy, and flexibility in data collection and analysis. Through a systematic examination of over 21 case studies and research papers, we demonstrate the practical applications and benefits of these systems in diverse fields such as automation, healthcare, and environmental monitoring. Our goal is to provide a comprehensive understanding of the interplay between microcontrollers and sensors, highlighting recent advancements in IoT protocols that enable remote monitoring and control. Ultimately, we aim to develop a low-cost, high-precision, and scalable air quality monitoring system that can effectively measure and record pollutant concentrations, offering valuable insights for improved air quality management.

Keywords: *Microcontrollers, Sensors, Arduino Pro Mini, Environmental Monitoring, Air Quality Measurement, Data Integration*

ABSTRAKT

KONCEPTIMI, PLANIFIKIMI DHE PROGRAMIMI I NJË STACIONI MONITORIMI MJEDISOR PËRDORUR IDE-në ARDUINO DHE EDITORIN NEXTION

Dafku, Dorjan

Master Shkencor, Departamenti i Inxhinierisë Kompjuterike

Udhëheqësi: Prof.Dr. Betim Çiço

Ky punim eksploron rolin transformues të mikrokontrollorëve dhe sensorëve në sistemet elektronike moderne, me fokus në matjen e cilësisë së ajrit mjedisor. Duke shfrytëzuar aftësitë e programueshmërisë dhe integrimit të përbërësve si Arduino Pro Mini, të kombinuara me sensorë të ndryshëm të cilësisë së ajrit (p.sh., PM2.5, CO2, lagështi, temperaturë), ne ilustrojmë se si këto teknologji rrisin efikasitetin, saktësinë dhe fleksibilitetin në mbledhjen dhe analizën e të dhënave. Përmes një shqyrtimi sistematik të mbi 21 studimeve të rasteve dhe punimeve kërkimore, ne demonstrojmë aplikimet praktike dhe përfitimet e këtyre sistemeve në fusha të ndryshme si automatizimi, kujdesi shëndetësor dhe monitorimi mjedisor. Qëllimi ynë është të ofrojmë një kuptim gjithëpërfshirës të ndërveprimit midis mikrokontrollorëve dhe sensorëve, duke theksuar përparimet e fundit në protokollet IoT që mundësojnë monitorimin dhe kontrollin në distancë. Në fund të fundit, synojmë të zhvillojmë një sistem monitorimi të cilësisë së ajrit me kosto të ulët, precizitet të lartë dhe të shkallëzueshëm që mund të masë dhe regjistrojë me efektivitet përqendrimet e ndotësve, duke ofruar njohuri të vlefshme për menaxhimin e përmirësuar të cilësisë së ajrit.

Fjalët kyçe: *Mikrokontrollorë, Sensorë, Arduino Pro Mini, Monitorim Mjedisor, Matja e Cilësisë së Ajrit, Integrimi i të Dhënave*

This thesis is dedicated to my family, whose unwavering support and encouragement have been the foundation of my academic journey. To my mentors and professors, whose guidance and wisdom have inspired my passion for research and innovation in the field of environmental monitoring. And to all the pioneers of technology and science whose groundbreaking work has paved the way for the advancements in microcontrollers and sensors. Your contributions have not only influenced this work but have also sparked my enthusiasm for exploring the boundless possibilities of modern electronic systems. Thank you for believing in me and for helping me reach this milestone.

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to all those who have contributed to the successful completion of this thesis.

First and foremost, I extend my heartfelt thanks to my advisor, Prof. Dr. Betim Çiço, for their invaluable guidance, support, and encouragement throughout this research journey. Your expertise and insightful feedback have been instrumental in shaping this work.

I am also grateful to the faculty and staff of Epoka University for providing a stimulating academic environment and the necessary resources to conduct this research. Special thanks to the members of my thesis committee for their time, constructive criticism, and suggestions that greatly enhanced the quality of this work.

I owe a great deal of appreciation to my colleagues and friends, especially Nasho Themeli, for their constant support, collaboration, and the stimulating discussions that enriched my research experience. Your companionship made this journey enjoyable and memorable.

A special note of thanks to AGNA S.A. for the practical experience and professional insights that significantly contributed to the real-world applicability of this research. The opportunity to work with Nextion displays and Arduino Pro Mini has been invaluable.

I am eternally grateful to my family for their unwavering love, patience, and understanding. Your constant support and belief in my abilities have been my greatest motivation.

Lastly, I dedicate this work to all the pioneers and researchers in the field of environmental monitoring. Your groundbreaking work has inspired this research and will continue to inspire future innovations.

Thank you all for being a part of this journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Problem Statement

The electronics world has been through a transformative proevolution over the recent years based on the countless innovations and technological breakthroughts as a result of the microcontrollers and sensors.These components and many more have redefined the spectrum of the electronic measurements deriving in every plentiful area of integration possible.Therefore this academic exploration aims to expose the interplay between microcontrollers, sensors and other integrating devices as the heart of the revolution meantioned above, emphasising and displaying their major role in enhacing efficiency, accuracy and flexibility in our desired measurements, in our case environmental air quality.

The combination of the integrated circuits, microcontrollers with the power processing unit and the access of programmability surely have served as the backbone of the nowadays modern electronic systems because of their ability to execute and perform certain tasks reaching remarkable outputs and successful measurements conducted as well. On one hand by orchestrating a specified centralized control unit our microcontrollers are able to execute the integration of various sensors producing real-time data which are processed and acquired before.On the other hand sensors are obviosly important since they convert the present physical phenomena of a predefined range area into electical signals, giving life as the sensory part or organs of the electronic systems.The diversity of sensors is truly remarkable, ranging from temperature, humidity, matter, acceleration, CO_2, VOC and beyond imagination. In our case we are only focused on the air quality sensors and their crucial role in quantifying, capturing, taking input environmental variables and serving to a matrix of applications which range from automation, industial usage, daily lifestyle up to healthcare diagnostics or even simply as for informational unit.

This academic work derives into the usage of Arduino Mini Pro microcontroller which is programmed carefully by us and different sensor technologies regarding air quality like PM2.5, CO2, humidity, temperature and more, illustrating the fundamental applications of their functions in our environment. After we went thorught a systematic examinaton of more than 21 case studies, research papers and observing their practical usefulness conducted in similar matterns we aim here to illustrate that how we can collaborate them to address the challanges that we face or encounter in everyday life, having a better scope of our air pollution while also specifying the recent improvements or advancements such IoT protocols which enable fancy remote controlling and monitoring of our systems by the third parties.

As we further progress on this academic work journey, our truly goal is to provide a better understanding on the landscape of the electronic world to other enthusiasts, providing explanation of the relationship between arduino and other electronic devices, highlighting their impact on monitoring different phenomena using measurement metrics like reliabilty, precision, accuracy, scalability, showing their complexities, their implementation risks on the board in order to navigate through our objectives and also have a bird eye view of our air quality, on how each air parameter affects our human being, while also trying to group them under one display for a cycle of 24 hours.

1.2 Motivation

The matter of diving into this rapidly challenging domain from our critical needs introduces the transformative potential of these pivotal components propelling innovation and adaptability in measurements while this realm complexity continues to escalade.On one hand microcontrollers as they provide powerful computational capabilities, they offer a pathway of executing or extracting subtle sensitive data from sensors while on the other hand harnessing this capability is immersive for the future of industries like automation, health, medical, environmental etc.Furthermore this investigation is justified by the another major metric which is efficiency in real time information and the motivation to the reach the maximum of its optimization. Microcontrollers operating as intelligent coordinators they investigate the real time execution or function of several sensors which expedite the input data collected during their sensing process while also minimizing errors associated or noise interference with manual intervention of humans, resulting in speeding up the measuring process.The desire to unlock the full potential or possibility of this throughput, basically efficiency accelerates the procedures and reliability in different meaurement applications, implicates the need to realize the potential of this synergy. Secondly the universality afforded by microcontrollers and sensors is essential for solving the demands of the modern technology because of the upcoming trends coming from the Internet of Things technology denoted by (IoT), therefore it exists an increasing demand for electronic devices that can interact easily with interconnected networks. Microcontrollers, which act as the brains of Internet of Things devices, allow for remote monitoring, flexible control, and easily data transmission. This inspires more investigation into the integration of microcontrollers and sensors to profit the most out of this power of IoT, rushing in a new era of interconnected electronic measures.

As conclusion our motivation to dive and explore microcontrollers and sensors in electronics technology measurements arises non only from the nowadays trends but also from the urgency to navigate and solve complexities of modern systems by studying these components, combining, connecting them together, testing and programming them so that we can unlock the new dimensions of the metrices meantioned above while also navigating through uncharted territories of innovation and future possible works. Therefore we aim to realize a low-budget cost, performing in high-precision, portable and reasonably scale-efficient monitoring system device that is more than capable to measure and record the concentration of our main focused gases and matters or pollutants in the air, displaying and interconnecting the output data wherever we need them to be seen and analyzed for further procedures.

1.3 Objectives

Understanding the architecture of a microcontroller and in our case our Arduino Pro Mini:

Providing a detailed sense of our microcontroller architecture, explaining its key components included such as the power processing unit, memory, input/output connections and also how it communicates with other components. Emphasising the huge role of the microcontroller in executing our main task which is monitoring while also managing real-time operations recorded by the real time clock device in our desired measurement system.

A detailed analysis of our sensor technologies :

Examing the diverse range of the sensors that we are using, including temperature and humidity sensors, particles sensor, VOC sensor, and others, getting a better view of the underlying principles of their operation and how they are connected in our system. Exploring advancements in sensor technologies, and their direct implications for improved precision, power efficiency and also sensitivity in electronic measurement systems.

The programmig techniques for microcontroller and dispaly:

Diving into the programming language and technologies or functions suitable for microcontroller-based systems programming such as Arduino Ide for our microcontroller, with our main goal to optimize and achieve optimal performance. Programming our display using Nextion Editor, in order to provide us with detailed data of our recorded measurements from all the sensors connected in our system parallel with the 24 hour cycle of the real time clock device.

Implementation strategies for our sensors and microcontroller:

Analyzing different strategies for optimized integration of the microcontroller and sensors in our system, based on major risky factors such as communication protocols for the devices, data synchronization between devices, and furthermore the calibration of them. Exploring the associated difficulties and challanges appearing with interfacing and intercommunication of multiple sensors to a microcontroller, enabling a bird eye view understanding of the system integration and configuration.

Applications and Case Studies:

Appearing all the possible practical real-life applications of microcontrollers combined with sensors in diverse fields despite our case air monitoring, including healthcare, environmental physical phenomena monitoring, technological and industrial automation, furthermore consumer electronics. Highlighting the impact of using pre-desired and well-calculated microcontrollers and sensors on improving the measurement accuracy and efficiency of a system.

Internet of Things (IoT) Integration:

Investigating and better understanding the further integration of microcontrollers and sensors into IoT-enabled electronic measurement systems as a flow towards nowadays trend.Exploring the role of microcontrollers in providing connectivity, remote monitoring accessibility, and data transmission into different devices, emphasizing the potential for smaller scalability and flexible adaptability.

The evaluation of the performance Metrics:

Defining key performance metrics that we will use for our electronic measurement systems incorporating microcontrollers and sensors on which we rely our changes and improvements, including accuracy and error, precision in configuration, data response time, and energy/power efficiency.Benchmarking the effectiveness of different microcontroller and sensor combination/ calibration in our applications.

Future and emerging trends:

Discuss emerging trends and technologies in microcontrollers and sensors technology, displaying their potential impact on the future of electronic measurements based on our system.Motivating a forward-looking optimistic perspective, providing more possibilities.

1.4 Organization of the thesis

This thesis is divided in 7 chapters. The organization is done as follows:

In Chapter 1, the problem statement, motivation and objectives of works is presented. Chapter 2, includes the literature review, summary of resolved probles, unsolved problems, future work, research gaps, hypotheses and research aim. Chapter 3, consists of the methodology followed in this study. In Chapter 4, the system design along with every sensor and circuit element included. In Chapter 5, the software implementation. In Chapter 6 testing and performance is included.In Chapter 7 conclusions and recommendations for further research are stated.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The literature review presented in our work is composed by the listed different papers investigated during our case and related to the latest work done in the field of microcontrollers and sensors technology. They are organized in the table as shown below, where basically the table consists of the paper title, the authors, the year of publishing and also some basic review or summarization based on the problems that they treated, how do they proposed the solution, the recommended future works and more on. Furthermore we have also analyzed these papers which corresponds mostly from the last four years and group them into 4 categories regarding the similarities of their contents.

2.2 Summary of Literature Review (Resolved Problems)

As a result of studying and critically analyzing the listed papers above, we have organized them into four groups including an optimized summary of their resolved problems provided while also mentioning their methods or ways of solutions. This categorization is based on their similarities, for instance the microcontroller that they used or proposed their application intentions, the technology integrated on them and more on.

Environmental Phenomena Detecting and Monitoring (Papers from 1 to 5):

These 5 papers listed above focus on developing monitoring and controlling systems for detecting and measuring physical parameters such as temperature by DHTT1, humidity, air quality, and pollution levels based on their specific sensors.

They mostly are based on microcontrollers like Arduino Pro or Mega to provide optimized smart solutions and innovations for indoor or outdoor environments. Furthermore, some of them include the integration of wireless sensor networks and the usage of IoT interactivity for data collection and analysis in real time which we will regroup below in category 4.

Health and Medical Systems Monitoring (Papers from 6 to 10):

The second group of 5 papers focuses on the advancements done in remote health monitoring systems directly for medical purposes, introducing a variety of methods used such as wireless body temperature monitors, infant temperature monitoring systems, feasible models, and wearable detectors. Furthermore, technologies mentioned there enable real-time transmission of data between different devices, improving patient care and accessibility to medical services. They appear cost-friendly and power effective, providing hopeful results as well.

Sensor Technology and Applied Applications (Papers from 11 to 15):

These papers included in group 3 explore truly remarkable trending innovations and new impacts in sensor technology, rising from analog front ends, and circuits for various applications mostly industrial, similar to our work. They highlight model developments and electronics challenges in programmable elements, sensors, evaluate their performance to reach high precision integration, while also finding path to enhancing their performance matrices in diverse spectrum of ways.

IoT and Remote connectivity (Papers from 16-21):

This group of papers showcases modern research cases done in IoT applications and connectivity solutions between devices using microcontroller-based systems like Arduino on which we are more interested in our work. Papers here feature weather monitoring systems, home warning systems, and low-cost sensor assessments. They mostly emphasize the role of IoT in enhancing performance, connectivity, automation across different domains, flexibility of the data transmission, creating positive results and networks.

2.3 Summary of Literature Review (Future Work)

In this section we have conducted a summary of the future work proposed by each of the four specific groups mentioned above, based on which we seek the motivation to build our work progress. We have pointed out the most key point directions including some detailed descriptions of their proposed work.

Group 1

Energy Efficient Controlling Systems using Arduino:

Future work proposed by this group of papers focuses on integrating better sensors and devices, practically updating the elements while also optimizing energy consumption plan based on real-time environment conditions and trials.

Microcontroller Based Detection and Evacuation System:

Furthermore improvements suggested by these papers include optimizing sensor sensitivity and reliability, integrating cloud monitoring for remote alerts and data logging, and executing advanced algorithms for early detection and prevention when these phenomena appear dangerous for our system.

Framework for Real-Time Monitoring:

Future research directions include developing betetr algorithms for prediction in pollution modeling, miniaturization of sensor nodes scalability for increased flexibility, and addressing security challenges in IoT networks to reduce the risk of errors.

Group 2

Design and Implementation of Monitors with Alert System:

Future enhancements include expanding sensor integration into monitoring additional integrating AI for real-time analyses, and developing applications for data management, more accurate alert and self-awareness systems.

IoT Based Monitoring System:

Potential future work involves integrating additional medical sensors for comprehensive health monitoring, security protocols to protect sensitive health information, and conducting more trials for system validation and optimization.

Group 3

Analog Front End in a Microcontroller for Industrial Pupose Sensing:

Future research directions include optimizing hardware implementations for improved system efficiency, conducting a greater number of experimental evaluations to validate sensor performance, and exploring novel applications in industrial automation and control.

Wireless Temperature Sensor:

Further developments could focus on improving sensor sensitivity and stability in more environmental situations, integrating wireless communication modes for remote data communication, and improving applications in healthcare and environmental monitoring electronic systems.

Group 4

Arduino Based Ambient Air Pollution Sensing System:

Future work involves developing new models based on real-time sensor data, implementing low power techniques, and addressing data integration challenges in IoT modules as well.

2.4 Summary of Literature Review (Unsolved Problems)

In this section I have conducted a summary of the unsolved problems proposed by each group of the specific papers. These problems appear because of their testing or their performance evaluation during the performance of their experimental conducting, while some of these papers prefer to specify their problems, the rest have shown small indices of their challenges. Based on studying this section, we have been able to construct the research gaps and the enthusiasm to operate our work.

Sensor Technology:

Enhancing sensor accuracy and reliability, especially in detecting physical environments. Improving sensor calibration techniques for optimal performance. Addressing compatibility, scalability issues and providing noise immunity.

Communication and Connectivity:

Developing better communication protocols for better data transmission. Increasing wireless communication range for remote applications.

Control and Automation:

Optimizing power consumption for prolonged battery life and low power consumption. System responsiveness and faster real-time response capabilities.

Safety and Efficiency:

Improving the algorithms to operate early detection and prediction for the monitoring process. Focusing in privacy concerns associated with sensitive data transmission. Optimizing energy efficiency and reliability.

2.5 Research Gaps

Regardless the research solutions provided and done in these papers, there are also some missing parts that need to be resolved and critically be analyzed. We have displayed below the listed main research gaps for every group of papers:

Group 1

Energy Efficient Air Quality Monitoring System using Arduino: Integration of real-time data organization to optimize energy consumption dynamically.Standardized protocols for data communication and control across different systems.Evaluation of long-term performance and maintenance needs in diverse environmental conditions and increasing trials performed.

Microcontroller Based Detection and Prevention System: Improvements in sensor sensitivity and longer range to detect accurately. Development of fail-safe protection mechanisms and alert systems wherever needed.Longer time reliability and careful calibration of sensors in varying conditions.

Framework for Real-Time Air Monitoring:

Security challenges in IoT networks.Design and implementation of the system for large-scale deployment.Integration of predictive analytics for proactive pollution management.

Group 2

The process including design, implementation and calibration of a monitoring with alert system:

Exploration of sensor capabilities to monitor signs accurately.Data privacy and security in wireless transmission of health data.

IoT Based Monitoring System:

Integration of diverse sensors.Addressing communication issues between different IoT devices and platforms.Validation of system reliability and accuracy in real-world.

Group 3

High-Precision Microcontroller for Industrial Sensing:

Hardware optimization to minimize noise and enhance precision in industrial applications.Development of cost-effective solutions for production and deployment.Ensuring compatibility and integration with existing industrial control systems and standards.

Wireless Temperature Sensor:

Integration of wireless communication modules without compromising sensor performance.Exploration of new application areas and customization for specific industrial or medical needs.

Group 4

Ambient Air Pollution Sensing System based on the Arduino Architecture:

Development of reliable low-power techniques for extended sensor operation.Standardization of data formats and communication protocols for seamless integration with other systems.Addressing data issues in large sensor networks.

Design of Micro-Climate Data Monitoring System :

Improving data analytics to provide actionable insights.Developing affordable, easy controlled interfaces to interact with and utilize the system effectively.

In our research work we will be focused in the first group of papers analyzed, moreover on the microcontroller based detection and prevention system gaps.

2.6 Hypotheses and Research Questions

Hypothesis 1: Environmental Phenomena Detecting and Monitoring

Hypothesis: Implementing real-time data integration recorded from multiple environmental sensors will significantly optimize energy consumption in systems and improve accuracy.

Research Questions:

1. How does the integration of real-time data impact the energy efficiency of our systems?
2. What improvements in sensor technology can enhance the sensitivity and specificity of our monitoring systems?
3. What are the long-term reliability and maintenance requirements needed for integrated environmental monitoring systems in diverse conditions?

Hypothesis 2: Health and Medical Systems Monitoring

Hypothesis: Maximazing the capabilities of wireless health monitoring systems to include multiple signs will enhance their performance matrices, especially in diverse and resource-limited settings.

Research Questions:

1. How can multi-sensor integration improve the accuracy and reliability of wireless health monitoring systems?
2. How does ensuring data privacy and security impact the effectiveness of wireless health monitoring systems in real-world settings?

Hypothesis 3: Sensor Technology and Applied Applications

Hypothesis: Improving the formats and communication protocols of the data recorded, will enhance the reliability and scalability of IoT-based air pollution sensing systems, facilitating their large-scale deployment and interconnection with other IoT systems.

Research Questions:

1. What are the most effective low-power techniques to extend the operation of IoT electronic sensor technology nodes in air pollution monitoring systems?

2. How does the standardization of data formats and communication protocols impact the integration and performance of large-scale IoT sensor networks?

2.7 Research Aim

This research work aims to investigate and optimize the integration and performance of multi-sensor environmental. This includes enhancing energy efficiency, accuracy, and usability of systems, as well as addressing the challenges of data standardization, security, and scalability. Specifically, the research seeks to:

1. Environmental Phenomena Detecting and Monitoring:

Improve the energy efficiency and functionality of air quality monitoring systems through the integration of real-time environmental data and advanced sensor technologies.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This particular section concludes the mechanisms that are going to be utilized to process, operate, investigate the materials and papers related to our aim, and therefore in this thesis the aim is to design and test the results of a air monitoring system in order to be effective, accurate, and accessible. The design of this operation will be associated with schematics and circuit diagrams, plenty of measurements will be performed in different environments for testing and calibration of the devices, graphs will be displayed for each sensor operating on a 24 hours cycle, programming of the devices will be conducted, more specifically for our Nextion Display and Arduino Pro Mini, while also the result analysis of the measured parameters will be illustrated, and a report of procedures will be conducted. Therefore, there will be a normative studies work presented as quantitative research using the experimental method and qualitative as well.

3.2 Quantitative Research

As it is meantioned above the work of this research will serve to determine our indication to the according field of study.The results of our modified work organized in schematic and mathematical way will be used to test the effects predicted by our hypotheses while also comparing to the existing systems to show which one provides better performance.

3.3 Qualitative Research

As we mentioned above, in this work qualitative research will be included. It will point out the significance of this research and also give a kind of background about this reserach area. There will be done also a research about the designs used before in order to improve, always by being concentrated in the aspects that will be studied.

CHAPTER 4

SYSTEM DESIGN

4.1 Introduction

The air quality monitor that we have proposed in our work is composed of two major parts that are the software and the hardware, furthermore in this particurlar section we will be presenting and describing the hardware of this system firstly. Basically we will show below all the elements included in our circuit with detailed explanation in a step by step manner, presenting the design flowchart containing the input/output parts of the system, the circuit schematics with noted pins and labels and also the diagram in 3D which later can be used as a future work in printing the PCB platform of this work.

4.2 Design Specification

As shown below in **Table 1** our monitoring system is constructed based on twelve components. Basically we have 8 sensing units ranging from: particle or matter sensor PMS5003, MH-Z19 CO2, MP503 VOC and MQ-131 Ozone, DHT22 temperature and humidity, LDR4589 lighting, UV 8521 UV and also BMP180 pressure unit as well. Furthermore as for the power unit supporter we will use a mini usb female connector that supports 5V to the circuit enough to make it run properly. The main component or the controlling unit is Arduino Pro Mini microcontroller since it serves as the sensor interface to collect all the possible needed data.It also performs the data acquisition and processing meaning that using its own ADC converter,this device does the transition of signals,it stores them temporarily until further communication with output devices.

On the other hand the output unit is the Nextion display which also helps interfacing with microcontroller and allows for us users the remote interaction while displaying data. Moreover, it helps in data visualization like charts, graphs, gauges, real-time monitoring, alerts and notifications based on exceeding certain thresholds, user input and control through its touch screen.

Table 1. Design Components of the System

	The List of all our Components	
Nr.	**Component**	**Operation**
1.	PMS5003	PM Sensing unit
2.	MH-Z19	CO2 Sensing unit
3.	MQ-131	Ozone Sensing unit
4.	MP503	VOC Sensing unit
5.	DHT22	Temp & Hum Sensing unit
6.	Nextion NX3224T028	2.8″ Display Output unit
7.	Arduino Pro Mini	Microcontroller unit
8.	DS3231	Real Time Clock
9.	Mini USB Connector	Power unit 5V
10.	LDR4589	Light Sensing unit
11.	UV 8521	UV Sensing unit
12.	BMP180	Pressure Sensing unit

4.3 Design Schematics and Flowchart

Figure 1 depicts our proposed design concept schematics which are performed using KiCad software. It includes all the devices used in our work with each individual pin and the way they are connected within the system, whereas **Figure 2** shows the flowchart of the system communication between the input block containing all our sensors connected in serial connection with their respected valuables of the display output, signal processing and the display, and on the other hand we have also included the block diagram and the operational description highlighted as follows.

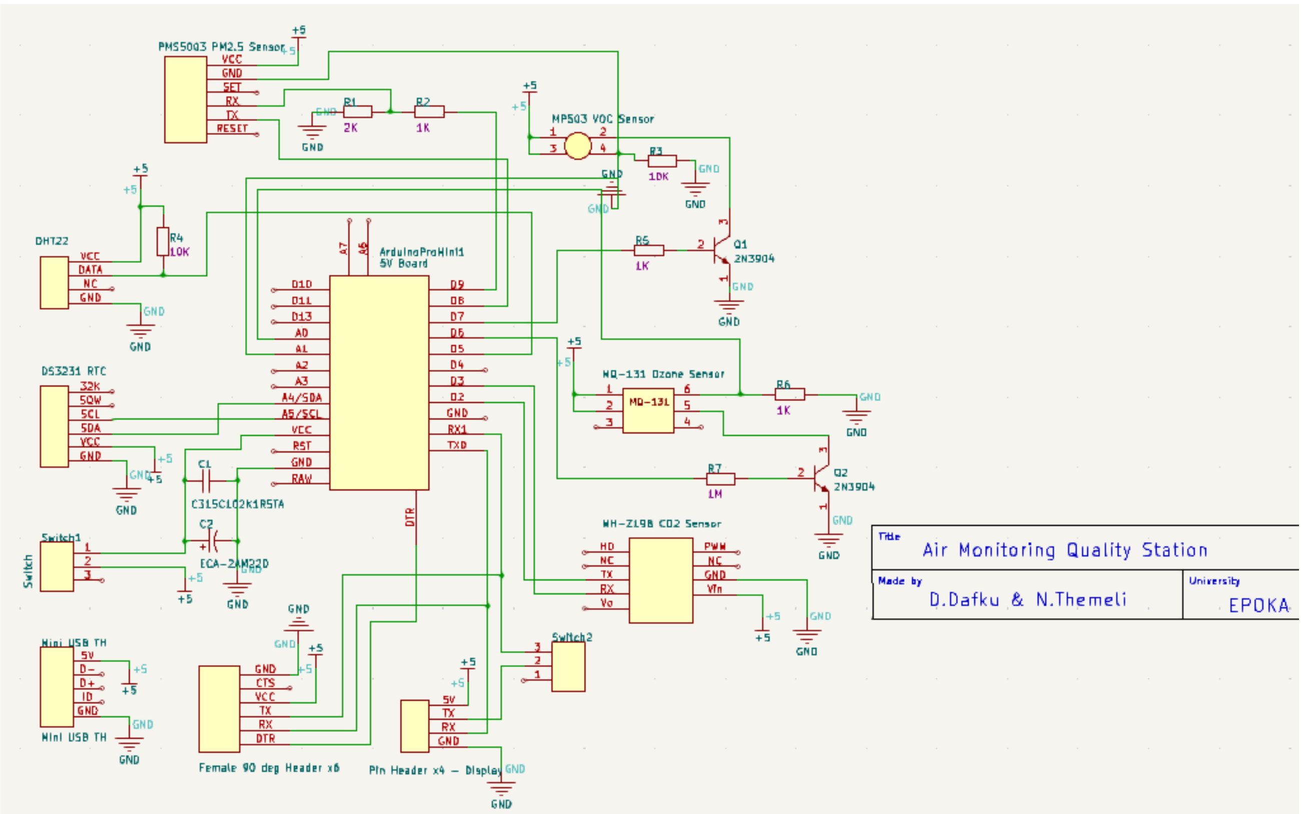

Figure 1. The Schematics design of the system using KiCad

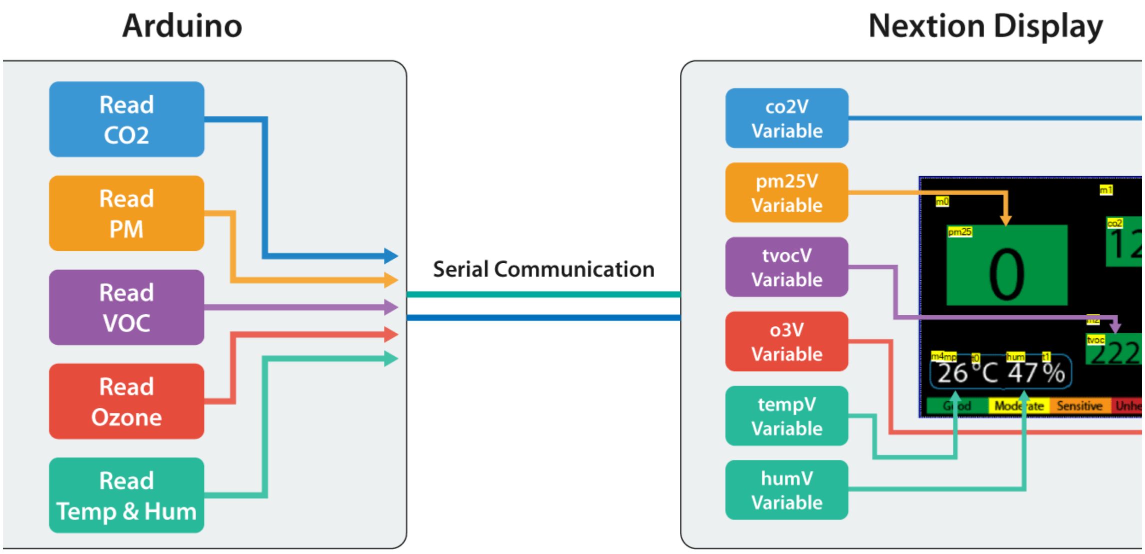

Figure 2. The Circuit Flowchart of the System between Input / Output

4.4 Design Block-diagram

Figure 3 displays the circuit diagram and every component of this circuit is connected as explained below:

Starting from the PM2.5 sensor which is responsible for sensing particle matter pollution we make the connection to Arduino using an interface in serial matter. This device operates at 5Volt for itself but the RX logic level of its components inside requires 3.3V therefore the urgent need for a voltage divider pops up.On the other hand also the CO2 sensor same as the previous one communicates with the Nextion display using serial connection.After that in order to read the data from the two other sensors which are VOC and Ozone we have to use the analog input ports, other than that the left senson DHT22 responsible for temperature and humidity uses digital connection.

The sensors that we are using have heaters inside which will require the help of transistors to activate them.In order to keep track of the time and the measurements recorded or done during our testing, furthermore saving temporarily we will use a real time clock device which does the communication process using I2C.The whole system is powered by 5V mini usb port as the feeder provided.

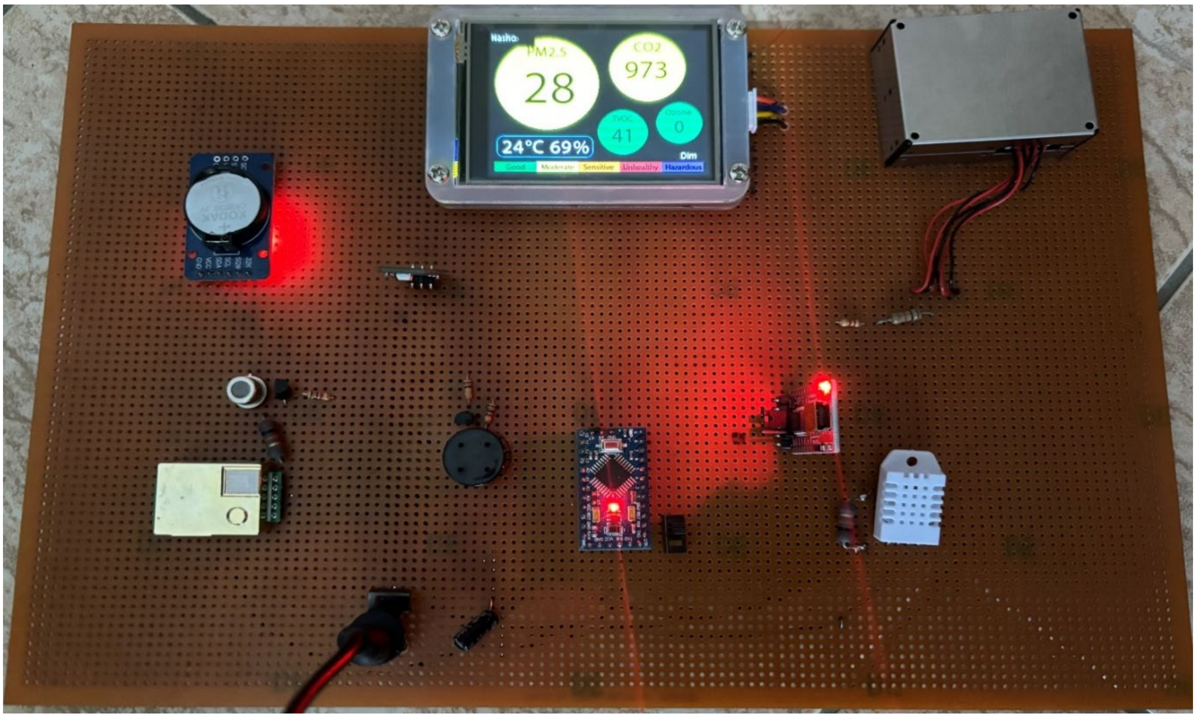

Figure 3. Real Time Circuit Implementation of the System

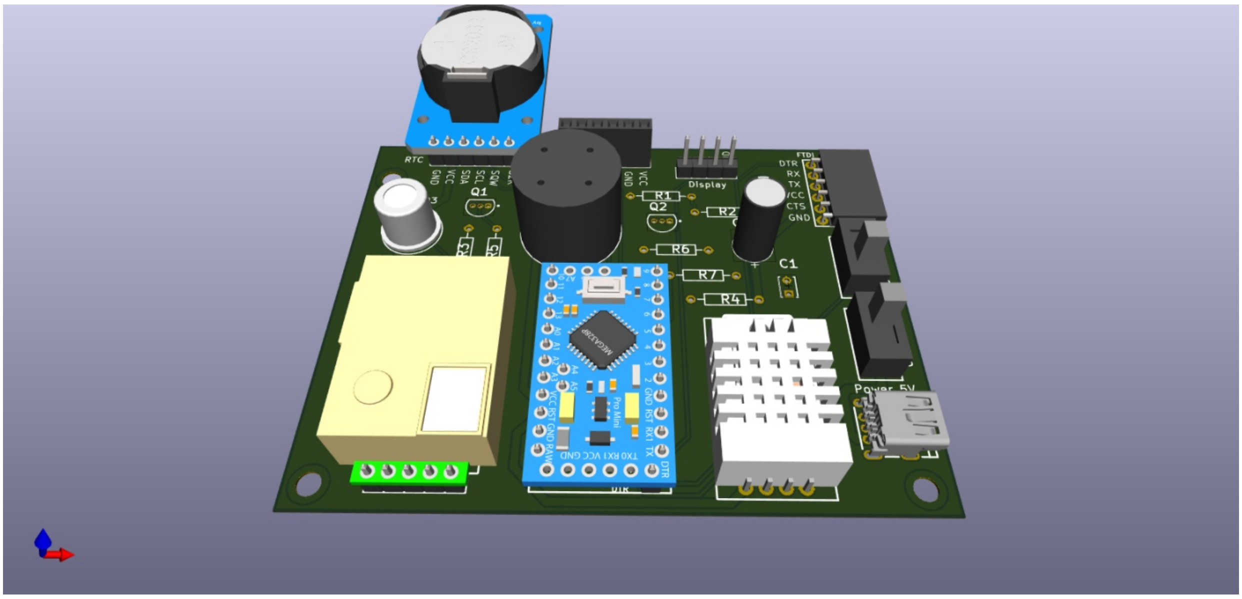

Figure 4. The functional Block Diagram of the system in 3D using KiCad

4.5 The PM2.5 Sensor – PMS5003

This component works as a digital particle concentration sensor which calculates the intensity of pollutiants in the air and then it produces an output in the form of digital interface for our system and provide correct data in time.It has four main components and measures matters in the air of around 2.5 microns which by the way are considered truly harmful for humans hence they penetrate deep into our system causing different health problems.

The block diagram of this sensor is directly taken from its own datasheet and we will show below in **Figure 5** while its major configuration features are listed in **Table 2.**

Table 2. *The PM2.5 Sensor Parameters*

Feature	Definition	Measurement Unit
Component Name	PMS5003	
Operation Range	0.3-1.0;1.0-2.5;2.5-10	(μ m)
Efficiency in Counting	50% for 0.3μ m 98% for 0.5μ m	
The Range of effictiveness	0 up to 500	μ g/m3
The Maximum Range Covered	1000	μ g/m3
Error as Consistency	10% for 100 up to 500μ g/m3 10μ g/m3 for 0~100μ g/m3	
The Power Supply In DC	5.0 Min & 4.5 Max: 5.5	(V)
Current in Active Mode	100	(mA)
Current Standby mode	200	(μ A)
The Temperature Range of Operation	10 up to 60	°C

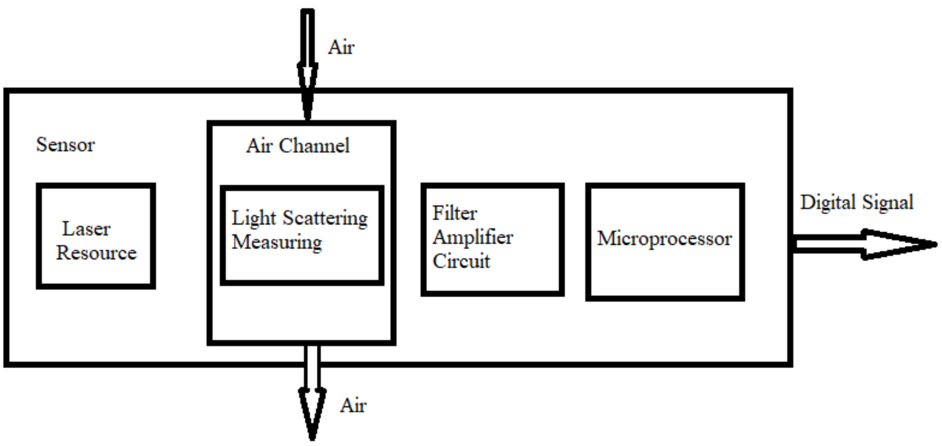

Figure 5. The Flowchart of the PM2.5 Functional Sensor Mechanism

PM2.5 Installation:

The installation process of this sensor needed the DV 5V power supply as the fan inside the component should be driven by that voltage, and since the level of the data pin was 3.3V we used conversion whenever the MCU is 5V. Regarding the pins, the set and reset pins should not connected including pin 7 and 8 as well.Furthermore we left the air flow way of the sensor free of and shield or structure.

Table 3. PM2.5 Sensor Pin Definition

Pin Coordination	Definition	Measurement Unit
PIN1	VCC	5V
PIN2	GND	Negative
PIN3	SET	Set pin 3.3V
PIN4	RX	Serial
PIN5	TX	Serial
PIN6	RESET	Module reset signal
PIN7/8	NC	

Figure 6. PM2.5 Sensor

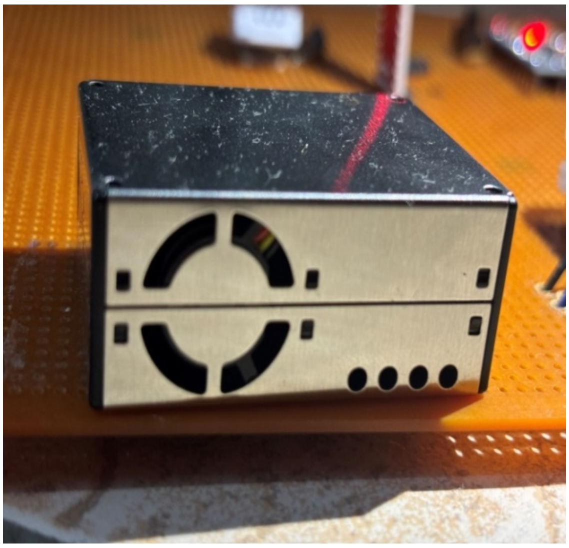

Figure 7. Real Time Implementation in Board

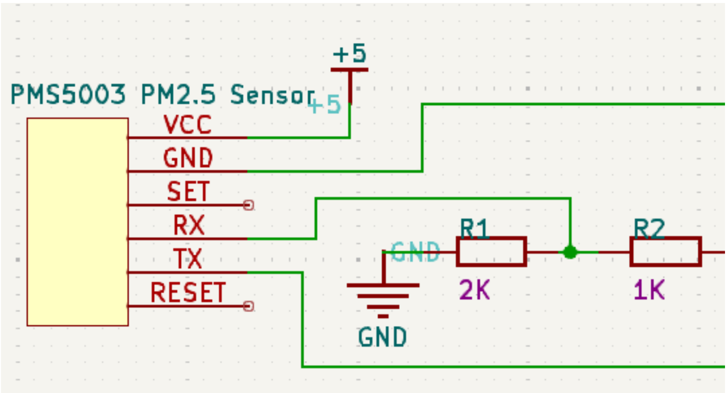

Figure 8. PM2.5 Schematics

4.6 The CO2 Sensor – MH-Z19

The next sensor that we are going to be using is the MH-Z19 which is responsible for recording the CO2 in the air. As humans breath and exits carbon dioxide while respiration we can agree that the indoor concentration of this gas can easily reach high levels resulting in being dangerous not only for its density but also for its effects on us like tiredness, sleepiness and more on. This sensor uses infrared principles for measuring this gas in air, basically an infrared source of the component directs or shots light through a tube filled with the air where our device is operating, the CO2 molecules present absorb a specific band of the IR light directed from the source while letting some other wavelengths to pass through. The reasons that we chose this sensor are because of its good selectivity, long lifespan, accuracy, does not depend on oxygen, low power consumption while also its built-in temperature sensor can do the compensation. Below we represent its main features in **Table 4** and how it looks in **Figure 9**.

Table 4. The CO2 Sensor

Component	**Measurement Unit**
The Component Name	MH-Z19
Operational Voltage	From 3.6 up to 5.5 V
Operational Current	18 mA
Covered Range	From 0 up 0.5%
The Operational Working Temperature	From 0 up to50 °C
Pin Nr.6	Vin
Pin Nr.7	GND
Pin Nr.1	Vout (3.3V, 10mA)

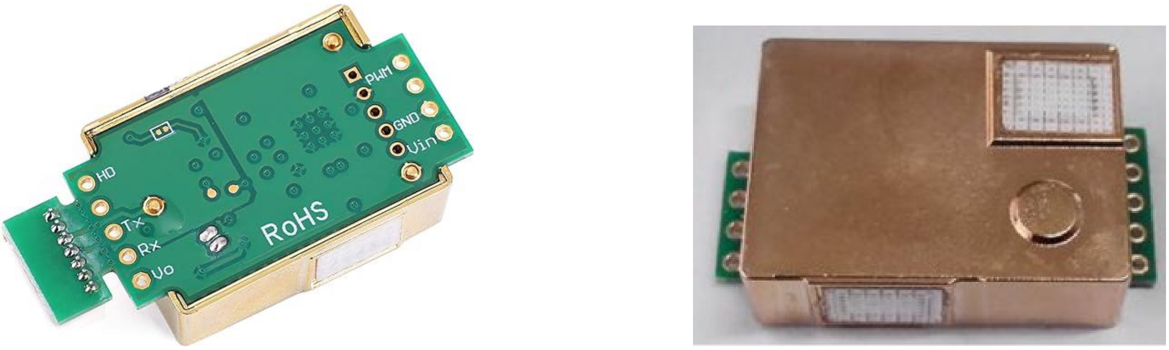

Figure 9 MH-Z19

The operation of this sensor is based on the connection of the pins where Vin-5V power, RXD connect TXD, TXD connect RXD, on the other hand we could read gas concentration without calculations.Wiring connections:MH-Z19 VCC: Connect to Arduino Pro Mini VCC (5V); MH-Z19 GND: Connect to Arduino Pro Mini GND; MH-Z19 TX: Connect to Arduino Pro Mini RX (Pin 2); MH-Z19 RX: Connect to Arduino Pro Mini TX (Pin 3).

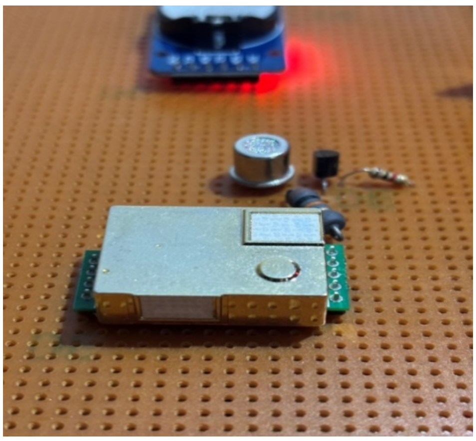

Figure 10. Real Time Implemtation in Board

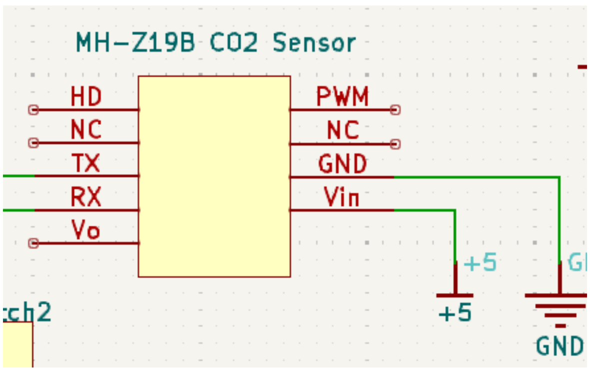

Figure 11. *MH-Z19 Schematics*

4.7 The VOC and Ozone Sensors – MP503 and MQ-131

In this section we have two more gas sensors that we are using in our work for measuring Ozone **Figure 12** which is basically an indoor normal household gas generated by different devices like steamers, lamps that use ultraviolet light or even air purifiers, while on the other hand we have the other sensor which measures gases like smoke, alcohol, methanol, butane and more on. These sensors are basically semiconductor heated metal oxide devices based on detecting change in their resistance of the aimed gases as mentioned above. The amount or density of the gases present at our area of operation changes the resistance as a specific electrical current pass through the metal substrate constructing these components.

Our MQ131 gas sensor has high conductivity in clean air and high sensitivity to ozone which means that when ozone gas exists and rises, the conductivity of the sensor will get lower while converting this change to output signal for our system though its circuit. On the other hand, the MP503 sensor **Figure 15** is more about the air quality comparing to the other sensor but the working principle is almost the same.

Table 5. The VOC sensor

Component	**Measurement Unit**
Component Name	MP503
Operational Voltage Inside the Loop	Less than 24V DC
The Voltage in Heating	From 5.0V up to 0.1V AC/DC
The Range of Detection	From 10 up to1000ppm
Time needed to Heat Up	48 hours
Pin Nr. 1&2	Electrode Heating
Pin Nr. 3&4	Electrode Measuring

Table 6. The Ozone sensor

Component	**Measuring Unit**
Component Name	MQ131
Operational Voltage Inside the Loop	From 5.0V up to 0.1V DC
The Voltage in Heating	From 5.0V up to 0.1V AC or DC
The Range of Detection	From 10 up to 1000ppb
Time needed to Heat Up	48 hours
Pin Nr. 1&2	Electrode Measuring
Pin Nr. 3&4	Electrode Heating

Figure 12. MQ-131

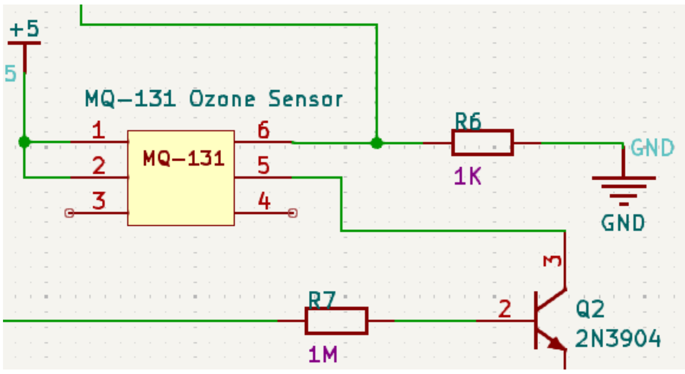

Figure 13. MQ-131 Schematics

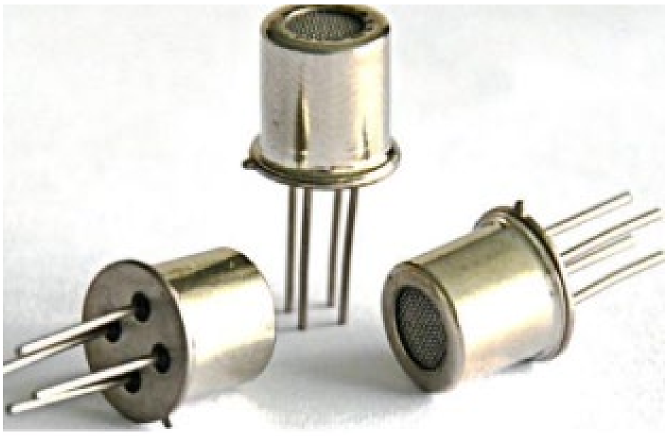

Figure 14. *MP503*

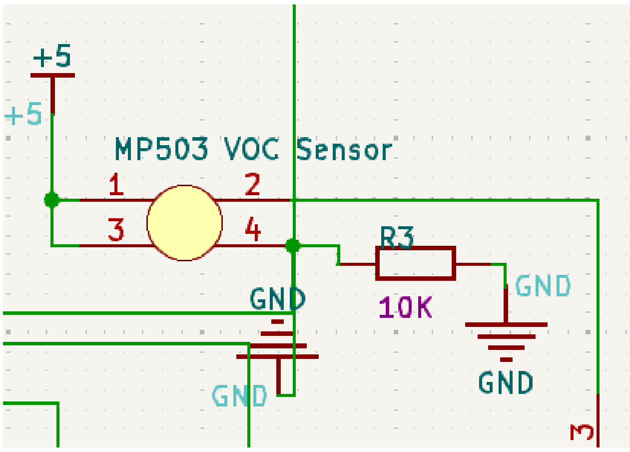

Figure 15. MP503 Schematics

The MP503 sensor requires two voltage inputs which are the heater voltage and the circuit voltage.The first one is used to supply the working temperature of the sensor while it can adopt DC or AC power and on the other hand VC supplies the detect voltage to load resistance RL.

Wiring Connections MP503:Power Connections: MP503 VCC: Connect to Arduino 5V pin. MP503 GND: Connect to Arduino GND pin. Analog Output Connection: MP503 OUT: Connect to Arduino analog input pin (A1). This pin will read the analog voltage output from the sensor. Heater Control : MP503 Heater Control: This to a digital output pin on the Arduino (D7).

Wiring Connections MQ131:Power Connections: MQ131 VCC: Connect to Arduino 5V pin. MQ131 GND: Connect to Arduino GND pin. Analog Output Connection: MQ131 OUT: Connect to Arduino analog input pin (A0). This pin will read the analog voltage output from the sensor. Heater Control : MQ131 Heater Control: Connect this to a digital output pin on the Arduino (D6).

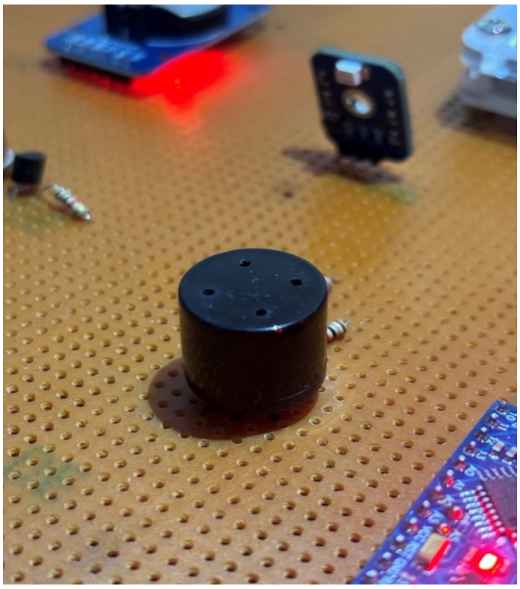

Figure 17. MQ131 in Real Time Circuit Implementation

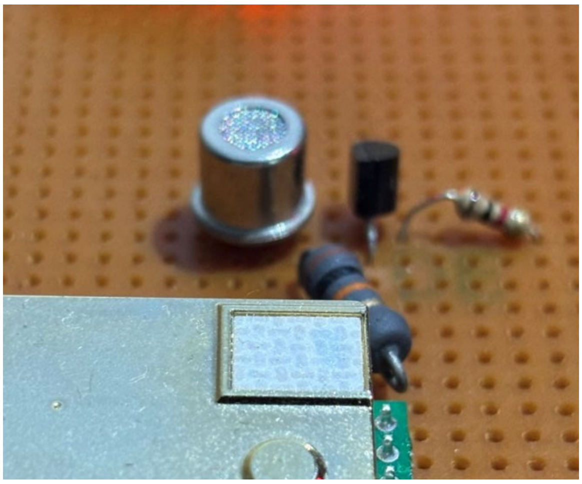

Figure 16. MP503 in Real Time Circuit Implementation

4.8 DHT22 – Temperature and Humidity sensor

The next sensor that we will be using is DHT22 which produces as output calibrated digital signal that later is connected and showed in our display.This component utilizes a technique that collects digital signals and its sensing elements are connected with 8-bit chip computer.Furthermore its advantages as reliability or stability, small size and low consumption are the reasons why we prefered to implement this sensor.On the other hand these specifications **Table 7** make it suitable for almost all kinds of occasions.

Table 7. The Temperature and Humidity Sensor

Component	**Measuring Unit & Definition**
Component Name	DHT22
The supplied Power	From 3.3 up to 6V DC
The Signal of Output	Digital
Level of Sensitivity	Hum. 0.1%RH; Temp. 0.1Celsius
The range of the operation	Hum. 0-100%RH; Temp. 4080Celsius
Pin Nr.1	VDD
Pin Nr.2	DATA
Pin Nr.3	NULL
Pin Nr.4	GND

Wiring Diagram Installation:

DHT22 Sensor Pinout: VCC: Connect to Arduino 5V pin. GND: Connect to Arduino GND pin.OUT: Connect to Arduino digital pin (D5).

Pull-up resistor (10kΩ) between the VCC and OUT pins of the DHT22 sensor to stabilize the data line.

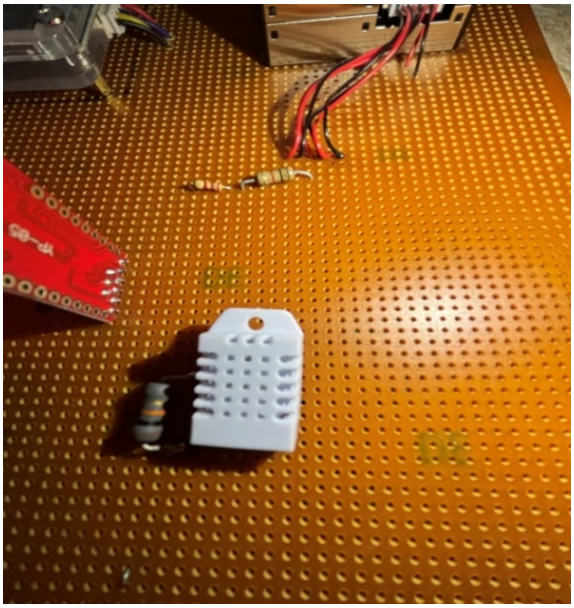

Figure 18. Real Time Implementation in Board

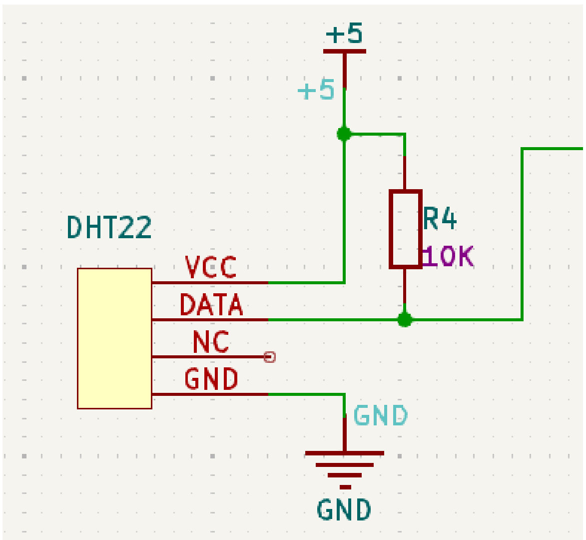

Figure 19. DHT22 Schematics

4.9 DS3231 RTC- Real Time Clock

Next, we will be using a RTC component to keep track for a period of 24 hours of the values recorded for each sensor included in our work. We preferred to work with DS3231 RTC since it can adapt to many applications like telematics, GPS, power meters and so on. It can count seconds up to days of week and year with 400 kHz I2C interface.

Table 8. RTC- Real Time Clock

Parameter	**Index**
Product Model	DS3231
Supply voltage	From 2.3V up to 5.5V
Active supply current	From 200 up to 300 µA
Standby supply current	From 110 up to 170 µA
Active Battery Current	From 70 up to 150 µA
Timekeeping Battery Current	From 0.84 up to 3.5 µA

DS3231 RTC Installation:

VCC: Connect to Arduino 5V pin.GND: Connect to Arduino GND pin.SDA: Connect to Arduino A4 (or SDA pin on Arduino Pro Mini).SCL: Connect to Arduino A5 (or SCL pin on Arduino Pro Mini).

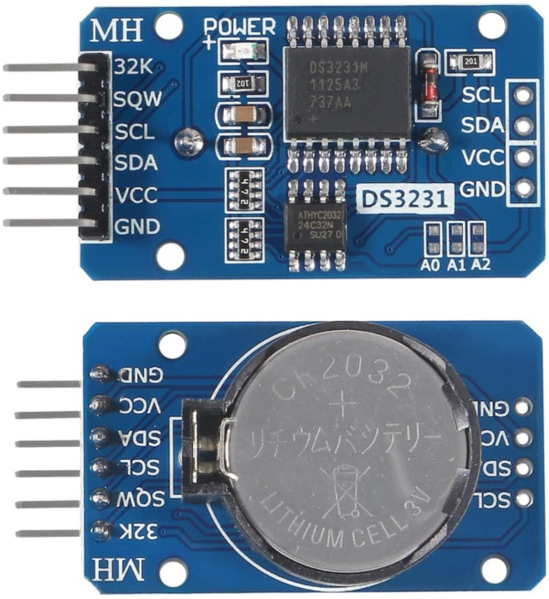

Figure 20. DS3231 RTC

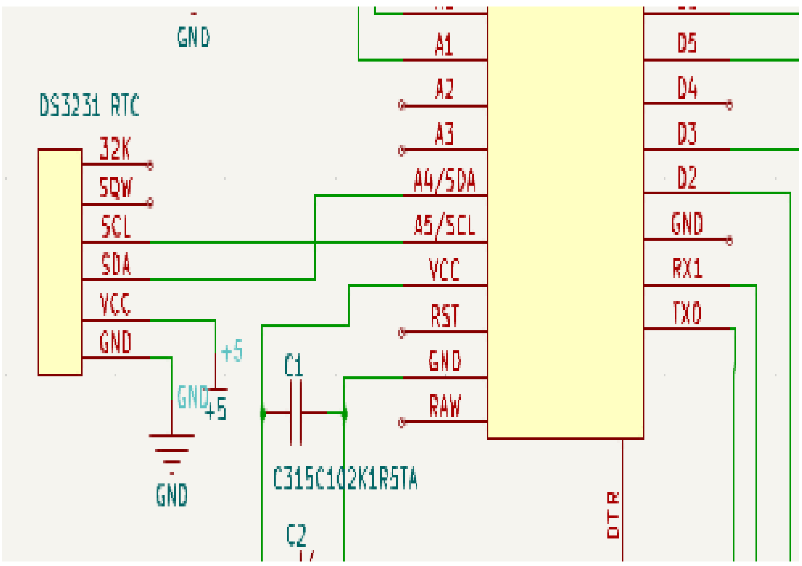

Figure 21. DS3231 RTC Schematics

4.10 NX3224T028 – Nextion Display

In order to monitor our recorded data, we will need the help of a display and in our case we have the NX3224T028 Nextion display which provides a control by humans and visualization interface as well, while also coming as a substitution for the traditional LCD or LED.

Table 9. The Nextion Display

Parameter	**Index**
Product Model	NX3224T028
Operating voltage	From 4.75V up to 7V
Operating current	65 mA
The recommended supplied power：5V, 500mA, DC	

Wiring Diagram Installation:

Nextion Display Pinout: 5V: Connect to Arduino 5V pin. GND: Connect to Arduino GND pin. TX: Connect to Arduino RX pin (for serial communication). RX: Connect to Arduino TX pin (for serial communication).

Make sure to connect the TX pin of the Nextion display to the RX pin of the Arduino Pro Mini, and vice versa.

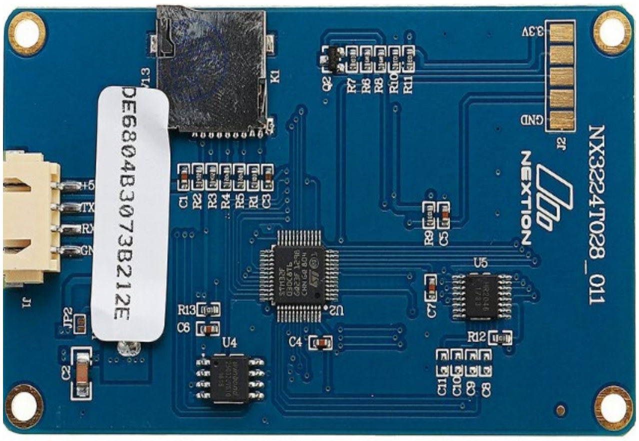

Figure 22.Nextion Display

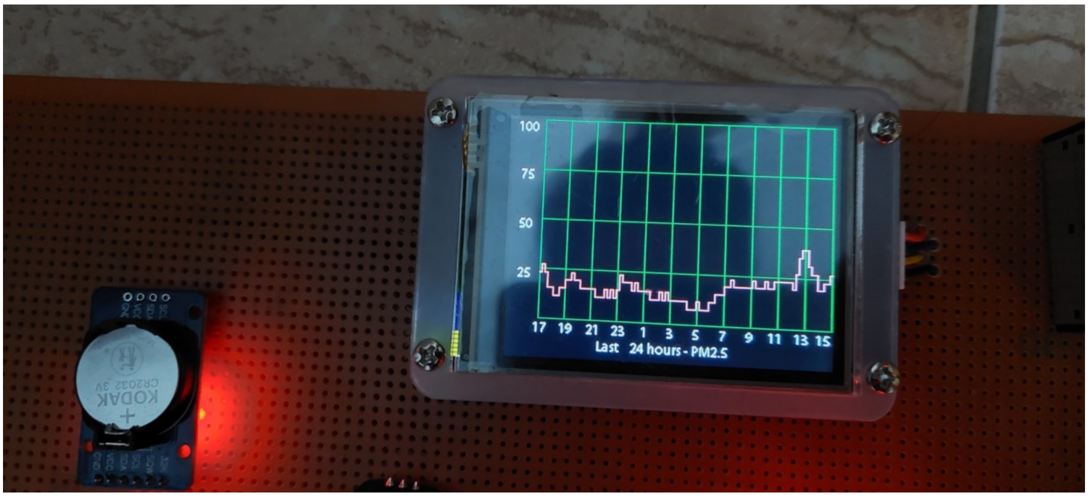

Figure 23. Real Time Graph Display

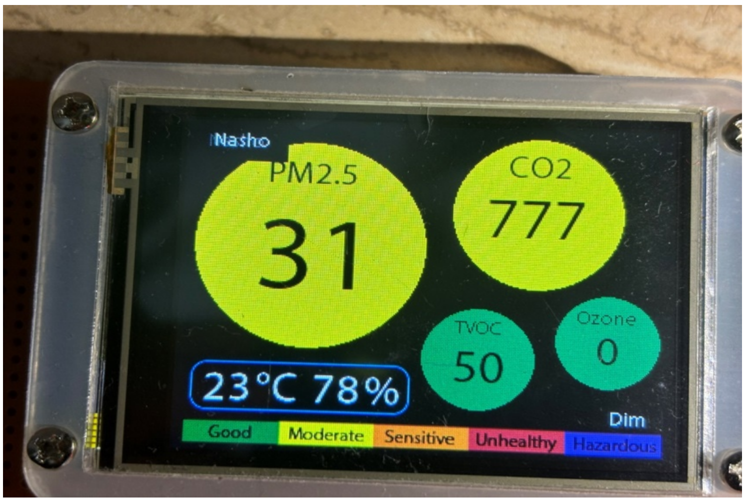

Figure 24. Real Time Implementation in Board

4.11 Arduino Pro Mini – Microcontroller

Our microcontroller choice was Arduino Pro mini which executes as the heart or brain of our system at 5V and 16 MHz, and the reason is since it is best for reading our sensors detecting harmful gases and matters in our environment.This component has 14 digital pins while also provides 6 analog ones, a regulator accepting voltages up to 12VDC, a memory, a reset button, a header that can be connected to a cable to provide power or communication.

Table 10. Arduino Pro Mini

Component	**Definition**
Name	ATmega328P
Supply power	(5V model)
Operating Voltage	5V
Clock	16 MHz
Memory	32KB

Arduino Installation:

Powering the Board: Connect a power source to the RAW pin (6-12V) or directly to the VCC pin (regulated 5V).

Connecting an FTDI Adapter: Connect the FTDI adapter to the GND, VCC, RX, and TX pins on the Pro Mini.Ensure that the voltage levels match (5V) between the FTDI adapter and the Pro Mini.

Uploading Code: Select "Arduino Pro or Pro Mini" in the Arduino IDE under Tools -> Board.Choose the correct processor (ATmega328P 5V 16MHz).Select the correct COM port.Upload the sketch.

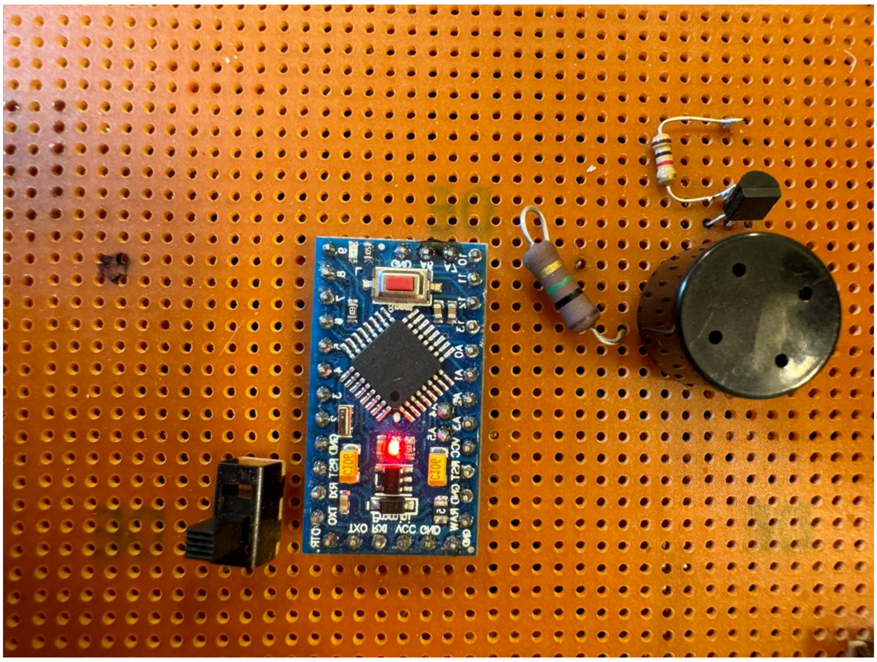

Figure 25. Arduino Real Time Implementation in Board

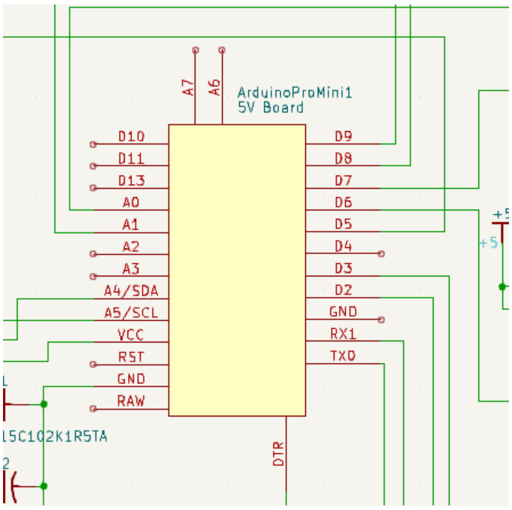

Figure 26. Arduino Schematics

4.12 LDR4589

This specific sensor that we use detects light density or darkness around the area which we are investigating, proving adjustable digital or analog output trigger level, suitable for different systems such as lighting, alarm etc.

Table 11. LDR

Component	**Definition**
Product Model	LDR4589
Operating voltage	From 3.3V to 5V DC
Operating current	15 mA
Output digital	0V to 5V
Pin Nr.1	AO
Pin Nr.2	DO
Pin Nr.3	GND
Pin Nr.4	VCC

Installation:

Connect one end of the LDR to the 5V pin of the Arduino.Connect the other end of the LDR to one end of the fixed resistor.Connect the junction of the LDR and the fixed resistor to an analog input pin of the Arduino (A0).Connect the free end of the fixed resistor to the GND pin of the Arduino.

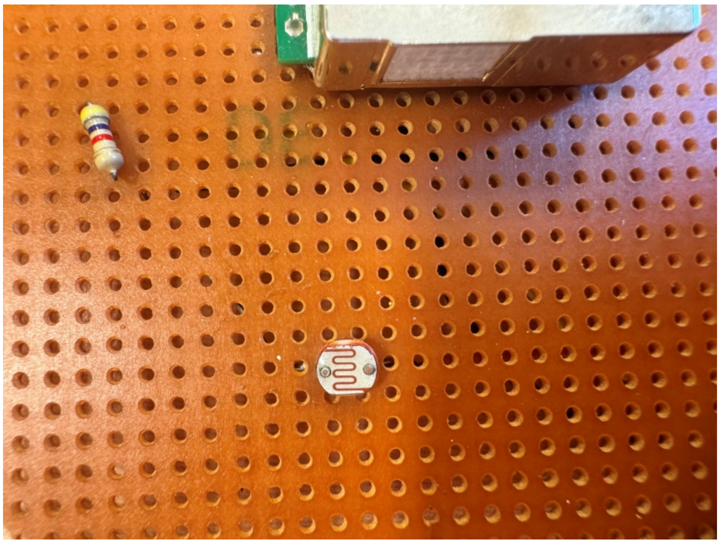

Figure 27. Real Time LDR Impelemtation

4.13 UV detection sensor

The last sensor that we will use is a UV detection sensor which is essential for monitoring ultraviolet radiation and truly crucial in industrial processes, consumer electronics etc. They provide accurate data for ensuring safety and healthy environments when they are exposed to radiation. Those sensors operations rely on converting the radiation or light into electrical signal basically using a photodiode made from truly sensitive materials to light. The operation results in a current which is proportional to the intensity of the light which is generated by electron-hole pairs of the device system.

Installation:

VCC: Connect to the 5V pin of the Arduino. GND: Connect to the GND pin of the Arduino. Analog Output (AOUT): Connect to an analog input pin of the Arduino.

Table 12. UV Sensor

Component	**Definition**
Product Model	UV sensor 8521
Operating voltage	From 3.3 to 5 DC
Current	From 0.06 to 0.1 mA
Output	0V to 1V DC
Wavelength	From 200 up to 370nm
Pin Nr. 1	OUT
Pin Nr. 2	VCC
Pin Nr. 3	GND

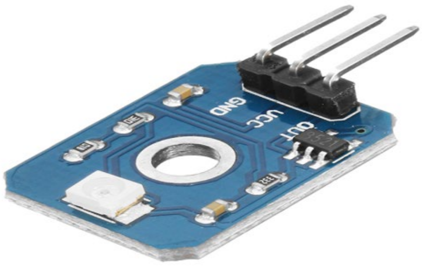

Figure 28. UV Sensor

Figure 29. Real Time Implementation

CHAPTER 5

SOFTWARE IMPLEMENTATION

5.1 Introduction

The software implementation of our system is divided into two parts, first part is implementen in Arduino Ide which will we useful to execute all the neccessary commands and functions to connect our microcontroller with all the other parts of the system, while on the second part we will use Nextion Editor to implement the code for the display.

5.2 Arduino Ide Code Implementation

The code for this part of the system will be included into the appaendix part.

5.3 Nextion Editor Code Implementation

The Nextion display simulation and design is implemented using Nextion Editor starting from the background, including the gadges, colours and all the parameter variables.We will be inlcuding how our display or monitor screen look while also inlcuding each respective code implemented.

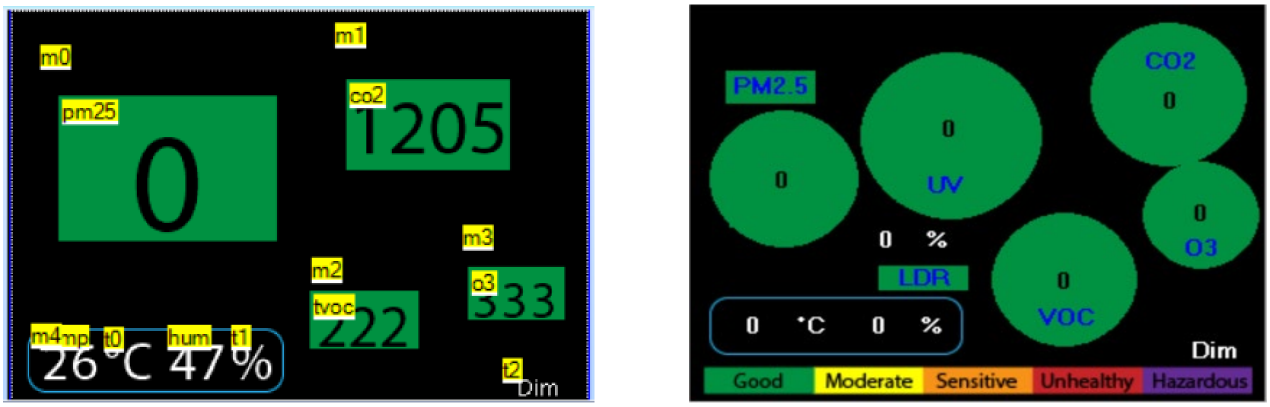

Figure 30. Display Design

Here we have listed all the attributes included in page0:

page0(Page); pm25(Number); co2 (Number); o3(Number); tvoc(Number); temp(Number); hum(Number); m1 (Hotspot); m0(Hotspot); pm25V(Variable); co2V(Variable); tvocV(Variable); o3V(Variable); tempV(Variable); humV(Variable); tmO(Timer); m2(Hotspot); m3 (Hotspot); pageSwitch (Variable); t0(Text); t1(Text); m4(Hotspot); t2(Text); dim1 (Variable); pm25C(Variable); pm25CurColor(Variable); pm25PrevColor(Variable); co2C(Variable); co2CurColor(Variable); co2PrevColor(Variable); tvocCurColor(Variable); tvocPrevColor(Variable); tvocC(Variable); o3C(Variable); o3CurColor(Variable); o3PrevColor(Variable); warmUp(Variable); uv(Variable); uvC(Variable); uvCurColor(Variable); uvPrevColor(Variable); uv(Number); m5(Hotspot); IdrV(Variable); Id(Number); t3(Text); m6(Hotspot).

Here we have listed all the attributes included in page1:

page1(Page); s0(Waveform); t12(Text); t13(Text); n0 (Number); n1(Number); n2(Number); n3(Number); n4 (Number); n5(Number); n6(Number); n7(Number); n8(Number); n9(Number); n10(Number); n 11(Number); y0(Number); y1 (Number); y2(Number); y3(Number); t0(Text); m0(Hotspot); tm0(Timer)

Here we will display the code used for Nextion Display Touch Events:

PM2.5:

```
prints "0",1
page1.t0.txt="PM2.5"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait“
```

CO2:

```
prints "1",1
page1.t0.txt="CO2"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait"
```

Temperature and Humidity:

```
prints "4",1
page1.t0.txt="Temp and Hum"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait
```

VOC:

```
prints "2",1
page1.t0.txt="TVOC"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait"
```

Ozone:

```
prints "3",1
page1.t0.txt="Ozone"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait“
```

UV:

```
prints "5",1
page1.t0.txt="UV"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait“
```

LDR:

```
prints "6",1
page1.t0.txt="LDR"
xstr 100,8,100,30,0,WHITE,RED,1,1,1,"Please Wait"
```

Here we will display the code used for Nextion Display Timer Events:

```
//Update the values
pm25.val=pm25V.val
co2.val=co2V.val
tvoc.val=tvocV.val
o3.val=o3V.val
temp.val=tempV.val
hum.val=humV.val
uv.val=uvV.val
ldr.val=ldrV.val
if(warmUp.val==0)//sensors warming up
{
  xstr 30,8,250,20,0,WHITE,RED,1,1,1,"Warming up sensors, please wait :)"
  delay=20000 //20seconds
  fill 30,8,250,20,BLACK
  warmUp.val=1//warming up in progress
}
if(pageSwitch.val==1)//switch to page 1
{
  page 1
}
```

Here we will display the code used for Nextion Display Post Initialize Events:

```
// Updates the buttons texts with the current variables values
pm25.val=pm25V.val
co2.val=co2V.val
tvoc.val=tvocV.val
o3.val=o3V.val
temp.val=tempV.val
hum.val=humV.val
uv.val=uvV.val
ldr.val=ldrV.val
cirs 52,105,41,1160
cirs 266,54,43,1160
cirs 208,166,40,1160
cirs 284,127,32,1160
cirs 145,79,50,1160
xstr 20,40,50,20,0,BLUE,1160,1,1,1,"PM2.5"
xstr 115,100,55,20,0,BLUE,1160,1,1,1,"UV"
xstr 245,25,45,20,0,BLUE,1160,1,1,1,"CO2"
xstr 275,138,20,20,0,BLUE,1160,1,1,1,"O3"
xstr 190,180,40,20,0,BLUE,1160,1,1,1,"VOC"
xstr 105,158,50,15,0,BLUE,1160,1,1,1,"LDR"
```

PM2.5 TIMER EVENT:

```
// PM2.5 Sensor
  if(pm25V.val<=12) // Good
  {
   pm25.bco=1160 // Green color
   pm25CurColor.val=1160
   if(pm25C.val==0)
   {
    cirs 52,105,41,1160
    xstr 20,40,50,20,0,BLUE,1160,1,1,1,"PM2.5"
    pm25C.val=1
   }
  }else if(pm25V.val>12&&pm25V.val<=35) // Moderate
  {
   pm25.bco=65504 // Yellow color
   pm25CurColor.val=65504
   if(pm25C.val==0)
   {
    cirs 52,105,41,65504
    xstr 20,40,50,20,0,BLUE,65504,1,1,1,"PM2.5"
    pm25C.val=1
   }
  }else if(pm25V.val>35&&pm25V.val<=55) // Sensitive
  {
   pm25.bco=62595 // Orange color
   pm25CurColor.val=62595
   if(pm25C.val==0)
   {
    cirs 52,105,41,62595
    xstr 20,40,50,20,0,BLUE,62595,1,1,1,"PM2.5"
    pm25C.val=1
   }
  }else if(pm25V.val>55&&pm25V.val<=150) // Unhealty
  {
   pm25.bco=49445 // Red  color
   pm25CurColor.val=49445
   if(pm25C.val==0)
   {
```

```
    cirs 52,105,41,49445
    xstr 20,40,50,20,0,BLUE,49445,1,1,1,"PM2.5"
    pm25C.val=1
  }
}else if(pm25V.val>150) // Hazardous
{
  pm25.bco=24946 // Purple  color
  pm25CurColor.val=24946
  if(pm25C.val==0)
  {
    cirs 52,105,41,24946
    xstr 20,40,50,20,0,BLUE,24946,1,1,1,"PM2.5"
    pm25C.val=1
  }
}
// Checking whether the color has changed
if(pm25CurColor.val!=pm25PrevColor.val)
{
  pm25PrevColor.val=pm25CurColor.val
  pm25C.val=0  //reset
}
```

CO2 TIMER EVENT:

```
/ CO2 Sensor
if(co2V.val<=700) // Good
{
  co2.bco=1160 // Green color
  co2CurColor.val=1160
  if(co2C.val==0)
  {
    cirs 266,54,43,1160
    xstr 245,25,45,20,0,BLUE,1160,1,1,1,"CO2"
    co2C.val=1
  }
```

```
}else if(co2V.val>700&&co2V.val<=1000) // Moderate
{
 co2.bco=65504 // Yellow color
 co2CurColor.val=65504
 if(co2C.val==0)
 {
  cirs 266,54,43,65504
  xstr 245,25,45,20,0,BLUE,65504,1,1,1,"CO2"
  co2C.val=1
 }
}else if(co2V.val>1000&&co2V.val<=1500) // Sensitive
{
 co2.bco=62595 // Orange color
 co2CurColor.val=62595
 if(co2C.val==0)
 {
  cirs 266,54,43,62595
  xstr 245,25,45,20,0,BLUE,62595,1,1,1,"CO2"
  co2C.val=1
 }
}else if(co2V.val>1500&&co2V.val<=2500) // Unhealty
{
 co2.bco=49445 // Red  color
 co2CurColor.val=49445
 if(co2C.val==0)
 {
  cirs 266,54,43,49445
  xstr 245,25,45,20,0,BLUE,49445,1,1,1,"CO2"
  co2C.val=1
 }
}else if(co2V.val>2500) // Hazardous
{
 co2.bco=24946 // Purple  color
 co2CurColor.val=24946
 if(co2C.val==0)
 {
  cirs 266,54,43,24946
  xstr 245,25,45,20,0,BLUE,24946,1,1,1,"CO2"
  co2C.val=1
```

```
  }
}
// Checking whether the color has changed
if(co2CurColor.val!=co2PrevColor.val)
{
  co2PrevColor.val=co2CurColor.val
  co2C.val=0
}
```

TVOC TIMER EVENT:

```
// TVOC Sensor
if(tvocV.val<=200) // Good
{
  tvoc.bco=1160 // Green color
  tvocCurColor.val=1160
  if(tvocC.val==0)
  {
    cirs 208,166,40,1160
    xstr 190,180,40,20,0,BLUE,1160,1,1,1,"VOC"
    tvocC.val=1
  }
}else if(tvocV.val>200&&tvocV.val<=400) // Moderate
{
  tvoc.bco=65504 // Yellow color
  tvocCurColor.val=65504
  if(tvocC.val==0)
  {
    cirs 208,166,40,65504
    xstr 190,180,40,20,0,BLUE,65504,1,1,1,"VOC"
    tvocC.val=1
  }
}else if(tvocV.val>400&&tvocV.val<=600) // Sensitive
```

```
{
 tvoc.bco=62595 // Orange color
 tvocCurColor.val=62595
 if(tvocC.val==0)
 {
  cirs 208,166,40,62595
  xstr 190,180,40,20,0,BLUE,62595,1,1,1,"VOC"
  tvocC.val=1
 }
}else if(tvocV.val>600&&tvocV.val<=800) // Unhealty
{
 tvoc.bco=49445 // Red  color
 tvocCurColor.val=49445
 if(tvocC.val==0)
 {
  cirs 208,166,40,49445
  xstr 190,180,40,20,0,BLUE,49445,1,1,1,"VOC"
  tvocC.val=1
 }
}else if(tvocV.val>800) // Hazardous
{
 tvoc.bco=24946 // Purple  color
 tvocCurColor.val=24946
 if(tvocC.val==0)
 {
  cirs 208,166,40,24946
  xstr 190,180,40,20,0,BLUE,24946,1,1,1,"VOC"
  tvocC.val=1
 }
}
// Checking whether the color has changed
if(tvocCurColor.val!=tvocPrevColor.val)
{
 tvocPrevColor.val=tvocCurColor.val
 tvocC.val=0
}
```

OZONE TIMER EVENT:

```
/ Ozone Sensor
if(o3V.val<=50) // Good
{
  o3.bco=1160 // Green color
  o3CurColor.val=1160
  if(o3C.val==0)
  {
    cirs 284,127,32,1160
    xstr 275,138,20,20,0,BLUE,1160,1,1,1,"O3"
    o3C.val=1
  }
}else if(o3V.val>50&&o3V.val<=100) // Moderate
{
  o3.bco=65504 // Yellow color
  o3CurColor.val=65504
  if(o3C.val==0)
  {
    cirs 284,127,32,65504
    xstr 275,138,20,20,0,BLUE,65504,1,1,1,"O3"
    o3C.val=1
  }
}else if(o3V.val>100&&o3V.val<=150) // Sensitive
{
  o3.bco=62595 // Orange color
  o3CurColor.val=62595
  if(o3C.val==0)
  {
    cirs 284,127,32,62595
    xstr 275,138,20,20,0,BLUE,62595,1,1,1,"O3"
    o3C.val=1
  }
}else if(o3V.val>150&&o3V.val<=200) // Unhealty
{
  o3.bco=49445 // Red  color
  o3CurColor.val=49445

  if(o3C.val==0)
```

```
 {
  cirs 284,127,32,49445
  xstr 275,138,20,20,0,BLUE,49445,1,1,1,"O3"
  o3C.val=1
 }
}else if(o3V.val>200) // Hazardous
{
 o3.bco=24946 // Purple  color
 o3CurColor.val=24946
 if(o3C.val==0)
 {
  cirs 284,127,32,24946
  xstr 275,138,20,20,0,BLUE,24946,1,1,1,"O3"
  o3C.val=1
 }
}
// Checking whether the color has changed
if(o3CurColor.val!=o3PrevColor.val)
{
 o3PrevColor.val=o3CurColor.val
 o3C.val=0
}
```

The same code follows also the other sensors.

CHAPTER 6

TESTING PERFORMANCE

6.1 Momental Testing

As we can see ***Figure 35*** for that time period our sensors have detected that: the purity of the air is quite acceptable at 29 PM2.5, the temperature was 22, humidity at 42%, VOV at 84, Ozone 42 inactive since it need 48 hours to turn on, CO2 is extremely normal at 541. In the display it is also included the range of colors with additional descriptions to better understand the environment status even by looking only at the gadges.

6.2 Testing and Results for a cycle of 24 hours

Below we will display and observe the recordings done of these environmental monitoring system components during a 24 hour shift at a stationary position.

As we can see from the graph the PM2.5 or the particle sensor monitoring have changed quite a lot during the 24 hour interval strarting from 25 and ending at a higher level.During this measurement we observed that during the night the sensor has recorded the lowest values approximately 15 while during the day it has reached values up to 40.

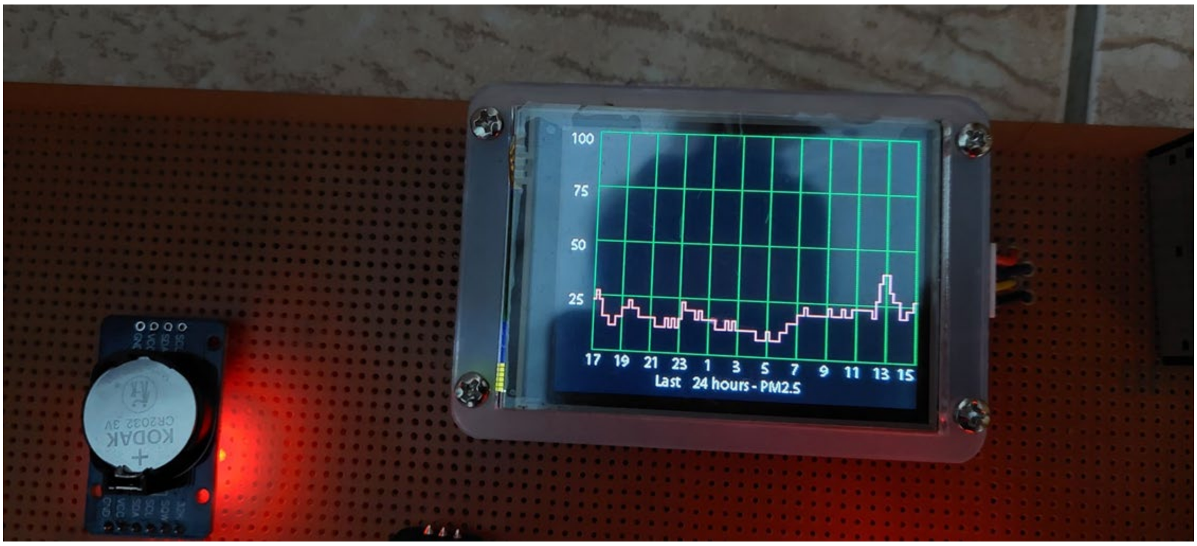

Figure 31. PM2.5 Graph Representation of 24 Hours Measurement

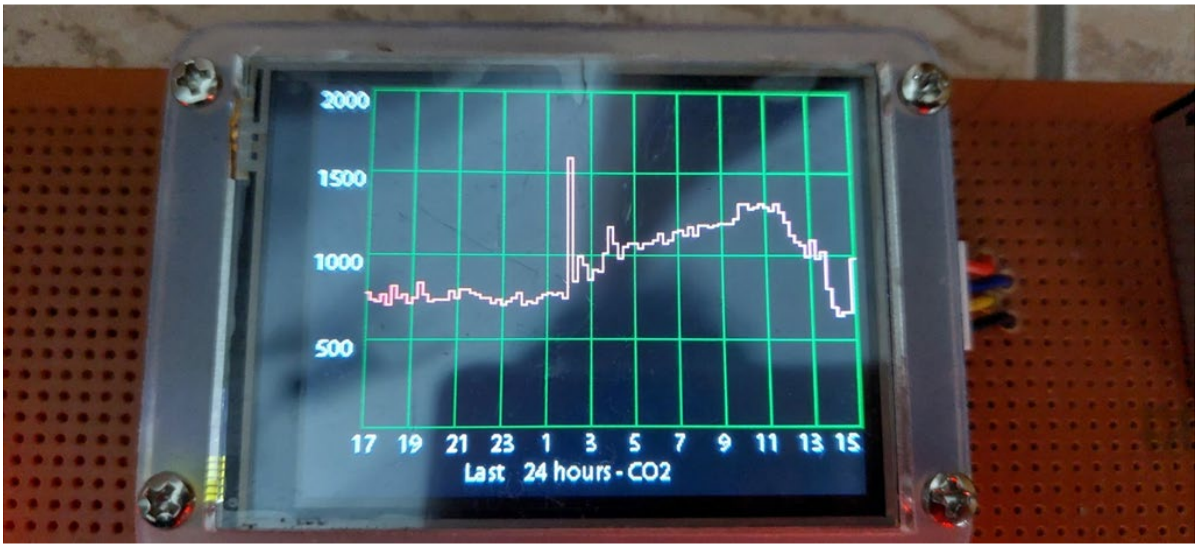

Figure 32. CO2 Graph Representation of 24 Hours Measurement

As shown in the graph, the values of the CO2 sensor exhibit a wide distribution over time. The highest number observed during this experiment, which was conducted under harsh conditions, was over 2000 ppm. In our stationary case, however, the maximum reading was around 1600 ppm, with a low of 600 ppm. This large range of data implies that CO2 concentrations vary significantly, most likely due to variations in environmental conditions and human activities. Furthermore, the sensor's continuous fluctuation within the measured range indicates that it is extremely responsive to variations in CO2 levels, making it a dependable tool for monitoring air quality. The graph also indicates periods of stability and spikes, which could be attributed to certain events or conditions during the measurement period.

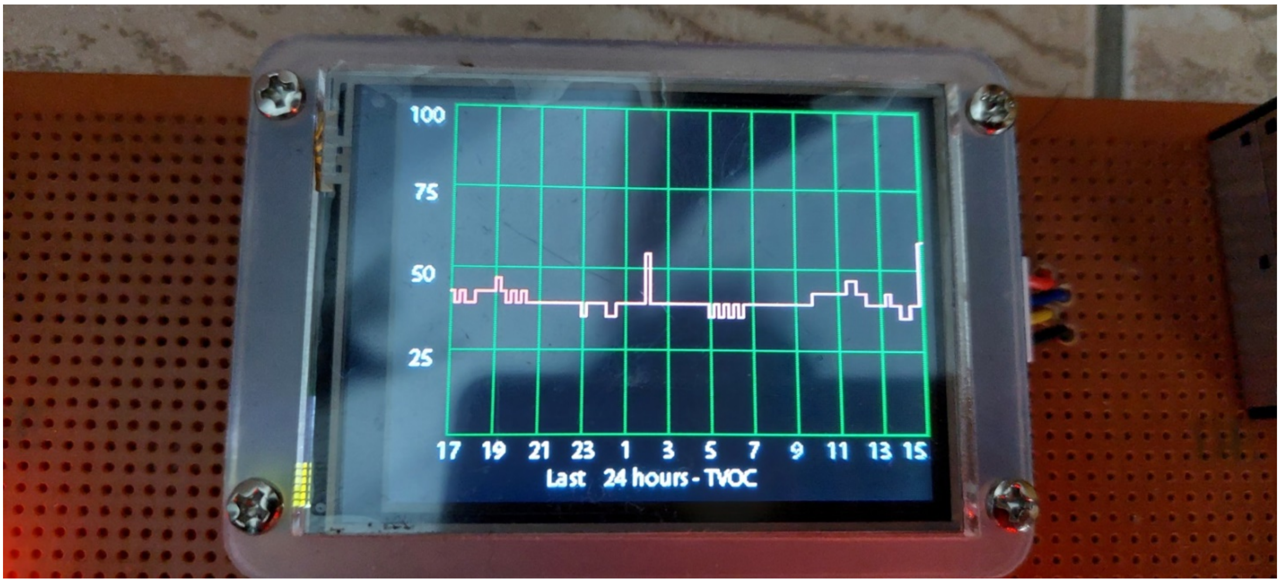

Figure 33. TVOC Graph Representation of 24 Hours Measurement

The result of this measurement is different compared to the other sensors since as we can distinguish the graph distribution is smoother and the range or difference between the values of this measurement conducted is very close to each other.

The minimum value of the VOC sensor is around 35 while the maximum value is around 55. This small range and smooth distribution indicate that VOC levels remain generally steady over time, with few variations.

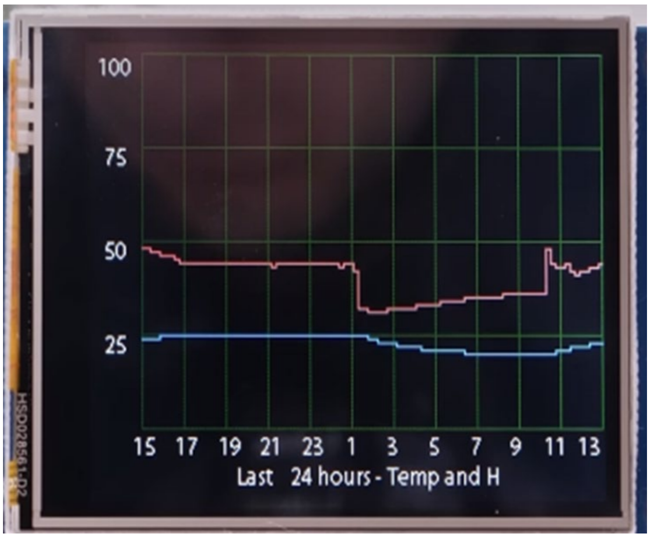

Figure 34. Temperature and Humidity 24 Hours Measurement

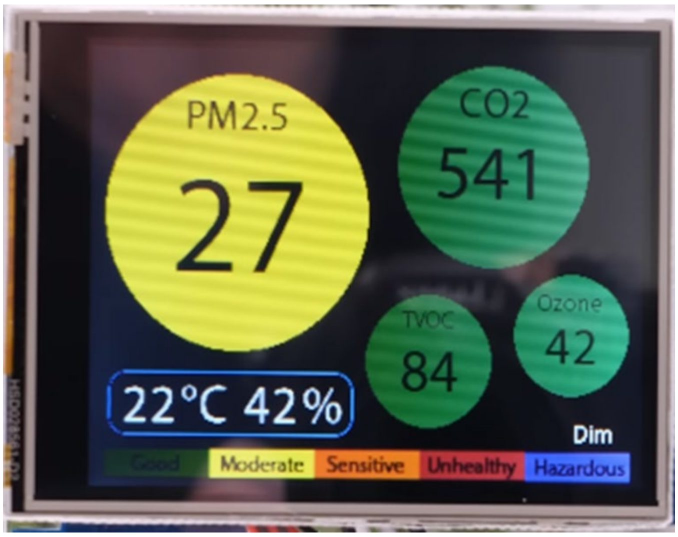

Figure 35. Display Color Range and Parameters Testing

CHAPTER 7

CONCLUSIONS

7.1 Conclusions

During the operation of this project we were able to sketch detailed schematics and the circuit and we were able to test it in different places.

The programming of the devices, was performed using the integration of the Nextion Display and Arduino Pro Mini in Arduino Ide and Nextion Editor.

The successful implementation of this environmental monitoring station demonstrates its potential as a reliable tool for air quality assessment, paving the way for broader applications in environmental monitoring and public health initiatives.

Table 13. Power Consumption and Current Needed

Component	Voltage (V)	Current (mA)	Power (mW)
PMS5003	5	100	500
MH-Z19	5	60	300
MQ-131	5	150	750
MP503	5	150	750
DHT22	5	2.5	12.5
DS3231	5	0.3	1.5
Nextion NX3224T028	5	65	325
BMP180	3.3	0.003	0.0099
LDR4589	5	15	75
UV 8521	5	0.1	0.5

Total Current Needed: 543 mA

Total Power Consumption: 2.715 W

Table 14. Total Price

Component	Price ($)
PMS5003	30
MH-Z19	40
MQ-131	20
MP503	15
DHT22	5
DS3231	10
Nextion NX3224T028	50
BMP180	5
LDR4589	2
UV 8521	10
Arduino Pro Mini	10

Total Price Estimate: $197 but including the maintaing and the process cost it goes approximately $400.

Table 15. Frequency and Data Acquisition Rate

Component	Data Acquisition Rate (Hz)
PMS5003	1
MH-Z19	1
MQ-131	0.008 (1 per 2 min)
MP503	0.008 (1 per 2 min)
DHT22	0.5
DS3231	400 kHz (I2C)
Nextion NX3224T028	As needed
BMP180	7.5
LDR4589	Continuous analog output
UV 8521	Continuous analog output

Data Recording Frequency: 1 Hz (once per second)

Table 16. Calibration

Component	Calibration Status
PMS5003	Factory calibrated; requires periodic field calibration after assembly
MH-Z19	Factory calibrated; zero and span calibration recommended after installation
MQ-131	Requires calibration in clean air and target gas environment post-assembly
MP503	Requires baseline and target gas calibration post-assembly
DHT22	Factory calibrated; no additional calibration needed after installation
DS3231	Factory calibrated; no additional calibration needed
Nextion NX3224T028	Not applicable
BMP180	Factory calibrated; no additional calibration needed after installation
LDR4589	Requires calibration for ambient light levels after installation
UV 8521	Requires calibration for UV index after installation

7.2 Recommendations for future research

As for the future work we are planning on adding more sensors to this circuit, in this manner it can be furthermore completed and sophisticated.In addition to our experimental trials we will be conductioning more measurements in different places of our university environment, in that way our data processed and displayed in our monitor could bring awereness to people of the status of their area.

Secondly by connecting the system to Internet of Things platforms, users could obtain real-time data analysis and remote monitoring of the air purity using online or mobile applications from any location. Deeper insights into the elements impacting air quality and aiding in the early detection of pollution events can be gained by implementing machine learning algorithms to forecast trends in air quality and identify anomalies.

The value of the monitoring station will be improved by adding more sensors to the array to measure other environmental characteristics like noise levels, nitrogen dioxide (NO_2), and sulfur dioxide (SO_2). The system can be made more sustainable and appropriate for remote deployments by optimizing power usage through the use of energy-efficient components, introducing sleep modes, and investigating the use of renewable energy sources like solar panels.

Ensuring the long-term precision and reliability of the monitoring data will require developing automatic calibration algorithms and investigating the use of more accurate sensors. Enhancing the Nextion Display's user interface and adding additional interactive features, like adjustable settings, alerts, and historical data visualization, would improve the user experience.

Adding the monitoring data to policy frameworks in partnership with legislators and environmental agencies would help to further promote data-driven decision-making and regulatory compliance. Broader accessibility will also be ensured by looking into ways to lower the cost of the monitoring station and make it scalable for larger deployment, such as employing affordable components and production economies of scale.

.

REFERENCES

[1] K. N. M. M. Taamté Jacob, "Low-cost air quality monitoring system design," *Energy and Environmental Engineering ,* vol. 12, p. 873–884, 2021.

[2] E. P. W. Rusimamto, "Implementation of arduino pro mini and ESP32," *Electrical Engineering and Computer Science,* vol. 23, pp. 1366-1375, 2021.

[3] T. I. N. Azhari, "“Design of Monitoring System Temperature And Humidity using DHTT22 Sensor and NRF24L01"," *Science and Technology,* vol. 2421, pp. 4-6, 2023.

[4] A. E. Fatemeh Ghassemi, ""Design and Implementation of Wireless Body Temperature Monitor with warning system via SMS"," *Signal Processing and Intelligent Systems ,* vol. 6th, pp. 2-4, 2020.

[5] A. B. Fakhri, ""Accurate Infrants Remote Temperature Monitoring System Based on Contactless Sensors"," *Developments in eSystems Engineering (DeSE),* vol. 13th, pp. 1-4, 2020.

[6] X. Q. Gurmu M. Debele, ""Automatic Room Temperature Control Sensor Using Arduino Uno R3 and DHT11 Sensor"," *Wavelet Active Media Technology ,* vol. 7th, pp. 1-3, 2020.

[7] C. P. Y. Lai, ""A microcontroller based gas leakage detection and evacuation system"," *Materials Science and Engineering,* vol. 2nd, pp. 3-6, 2020.

[8] P. Khoenkaw, ""An Implementation of Programmable Oscillator for Inductive-Loop Vehicle Sensor using Microcontroller"," *Computer and Telecommunications Engineering (NCON),* vol. 4th, pp. 1-3, 2019.

[9] A. H. Kelechi, ""Design of a Low-Cost Air Quality Monitoring System Using Arduino"," *Computers, Materials & Continua,* vol. 4th, pp. 3-10, 2022.

[10] J. Diz-Bugarin, ""Arduino-compatible microcontroller module for electronic practices and environmental monitoring"," in *IEEE*, 2020.

APPENDIX

Nr.	Paper title	Authors	Year	Source	What is it about?	Solved Problems	Unsolved Problems	Future work
1.	“The Design of Monitoring System Temperature And Humidity Using DHT22 Sensor and NRF24L01 Based on Arduino” [3]	Azhari , T I Nasution , S H Sinaga , Sudiat	2023	IEEE	The monitoring system that is proposed here consists of 2 nodes: the base and the client.The focus in on the base where there appear two hardware components: Arduino Uno as the microcontroller and the nRF24L01 modul operating in wireless.	It uses the NRF24L01 module so that the data received is in order with the data sent by testing the module in different rooms and the same range.Secondly the buzzer work better in 32℃ range. Furthermore because of the ratio between DHT22 and the thermohygro error is 1.4.	No unsolved proplems appeared in this paper.	No future work is proposed.
2.	“The Design and Implementation of Wireless Body Temperature Monitor with warning system via SMS” [4]	Fatemeh Ghassemi, Mohammad Saleh Hoseinzadeh, Ali Ekhlasi	2020	IEEE	This paper builts a thermometry with assisted online monitoring and SMS system as alert by the help of two circuits that communicate with each other by the radio receiver and transmitter module interconnection.	The solution is provided by financing the right amount of budget and their hardware design by increasing the range to measure temperature of the patient.	No unsolved proplems appeared in this paper.	Better application can be designed and be available to users easily according to the hardware of this system. Secondly better alert systems or warning signals can be developed.

3.	"Accurate Infants Temperature Monitoring System based on Contactless Temperature Sensor and GSM Network" [5]	Ahmed Bashar Fakhri, Sadik Kamel Ghargha n, and Salah L. Zubaidi	2020	IEEE	This study proposes a remote temperature monitoring system in order to stay away from medical centers. temperature to stay.Their model consists of an Arduino microcontrolle r (ATmega 328p) type and also the short message service (SMS) for emergency cases.	The mobility of the platform was provided by consulting specialists of the field. The proposed system reached effective performance with an accuracy about 99.	No unsolved proplems appeared in this paper.	Since the IRTM platform finds itself useful in plenty applications, in this matter for respiratory rate or heasrt rate as well, it will be needing to upgrade their elements and furthermore better algorithms to support its longevity.
4.	"Automatic Room Temperature Control System Using Arduino Uno R3 and DHT11 sensor" [6]	GURM U M. DEBEL E , XIAO QIAN	2020	IEEE	is about designing a system that performs the automation of the room temperature control using a DHT11 sensor.It also talks about the fan control as well.	The solution is provided by setting up a threshold value , based on which the microcontroller controls the heater.On the other case when the temperature is too high the fan is triggered.	No unsolved proplems appeared in this paper.	No future work proposed.
5.	" Street Light Controlling System using Arduino Microcontrol ler in Comparison with 8051 Microcontrol ler"	G. Chandra shekar , K. Vijayala kshmi	2022	IEEE	This paper aims to construct a street light system which is controlled by two types of microcontrolle rs: Arduino and 8051 microcontrolle r.	This model outsmarts the traditional street system because of their system with a significance value of 0.001.	No unsolved proplems appeared in this paper.	This paper proposes the construction of their model while also the need to be tested in order to verify its statements.

6.	"The Study on Power Converters Control in Hardware System Using Low Cost Microcontroller "	-Trias Andromeda -Ismoyo Haryanto -Joga Setiawan -Betantya Nugroho	2022	IEEE	It proposes a real time inverter monitor that produces a small power output of 5 where the inverters are controlled using proportional integral control with closed loop feedback.	The proposed solution charges the battery in its initial stage and turned into full charge for only 30 minutes at 14 Volt range.	No unsolved proplems appeared in this paper.	Furthermore improvement is needed as the converters used are cheap, therefore better converters implicate better system architecture and better performance..
7.	"A High-Precision Analog Front End Integrated in a 32bit Microcontroller for Industrial Sensing Applications only"	Koji Yoichi, Sugako Otani, Kazutoshi Tsuda, Naoya Tokimoto,	2020	IEEE	This paper focuses on the application of a microcontroller based on the combination of an AFE and a 32bit controller for industrial uses.	Solution is provided by employing PGA with the techniques of auto-zeroing and chip implementation.In that manner AFE reaches high precision and with optimal gain and offset drift in order to meet the requirements.	Solution is needed in improving gate driver and the cooling system so that the switching frequency can be increased resulting in improved performance.	Simplification, the reduction of the components required, space saving and wide range providing for the MCU are needed to force sensing and optimal performance of the system.
8.	" Gas leakage detection and evacuation System based on a microcontroller" [7]	-Cindy Poh Yuan Lai -Kah Haw Law -King Hann Lim	2020	IEEE	This model proposes a detection and prevention or evacuation system for the gas measurement recording in order to ensure safety.This will be realized using SMSs.	The solution is provided by the evacuator fans which are trigerred on a threshold value depending on the concentration of the gas.this smart system has many benefits in ensuring safety because of its early detection stages of detection.	No unsolved proplems appeared in this paper.	Hardware improvements are neeeded to ensure maximum of efficiency and performance.Better matching capacitors and values are also required.

9	"A Microcontroller-Based Interface Circuit for Three-Wire Connected Resistive Sensors system"	-Ferran Reverter	2022	IEEE	This paper proposes a microcontrolled based interface ,where the circuit reads three wire resistive for applications as: sensors.measure, for instance, temperature.	The solution consists on using embedded digital timmers for the process of charching and discharging correspondinf to four RC circuits and the sensor resistance.	No unsolved proplems appeared in this paper.	No further work is proposed.
10.	"An Implementation of Programmable Oscillator for Inductive-Loop Vehicle Sensor Using Low-Cost Microcontroller architecture" [8]	Paween Khoenkaw	2019	IEEE	The implementation of an automatic oscillator is provided tuning for vehicle sensor technology.	The solution is composed of a series elements like oscillator, resonance circuit, microcontroller to capture the output and the combination of the wires.The experiment was conducted on 24 combinations and simulated in real life scenarios.	No unsolved proplems appeared in this paper.	No future work is proposed.
11.	"Wireless Graphene Temperature Sensor"	Andrey Somov, Evgeniya Kovalska, Anna Baldycheva	2020	IEEE	The development of a sensitive portable and wearable temperature sensor is suggested with the usage of flexible polyvinyl substrate and graphene electrodes.	The solution is provided by enabling the temperature sensor to operate and combine with wireless communication features.	No unsolved proplems appeared in this paper.	No future work is proposed.

12	“Design of a Low-Cost Air Quality Monitoring System Using Arduino and ThingSpeak technology” [9]	Anabi Hilary Kelechi Mohammed H. Alsharif	2022	TSP	The system proposes the usage of different sensors for monitoring pollutants using Arduino Nano microcontroller development board. development board equipped with a Wi-Fi module and real-time display of air quality.	Solution is based on a threshold value in order for the sensors and the microcontroller to trigger the alarm system and provide notifications to the user.	No unsolved proplems appeared in this paper.	No future work is proposed.
14	“Arduino-compatible microcontroller module for electronics practices and environmental monitoring purposes” [10]	Javier Diz-Bugarín Rafael Rodríguez-Paz	2020	IEEE	This paper consists of an Arduino microcontroller board designed that can be assembled by the students as it combines hardware and software implementation.	This sosution is basically an update of the previous work done in Atmel AT89 microcontrollers but keeps compatibility with modules and is compatible with many commercial modules specifically for the Arduino.	No unsolved proplems appeared in this paper.	No future work is proposed.
15.	“Microcontroller Unit-Based Wireless Sensor Network Nodes: A state of art Review”	Ala’ Khalifeh, Felix Mazunga , Action Nechibvute and Benny Munyaradzi Nyambo	2022	IEEE	This paper presens detailed review of microcontroller unit (MCU)-based wireless sensor node platforms developed over the years.	This paper purpose is to provide a clear picture of the challenges and trendts towards the design of autonomous wireless sensor nodes to avoid redundancy in research by industry and academia.	The limitation of this research results as the absence of field power consumption data only for some sensor nodes.	Ultra-low power techniques that are present in wireless sensor nodes should continue to be explored. Miniaturization of the sensor nodes is also a future research topic as it directs into better scalability. Furthermore security issues must be adressed.

16.	“Design and Development of Microcontroller Based Wireless Humidity Monitor Architecture”	Jahedul Islam , Umme Habiba , Humayun Kabir	2019	IEEE	This research work aims to design and development low cost, small portable microcontroller based on wireless and humidity monitor sensing.	Solution is provideed as the sensors operate in capacitive principle meaning that they changes in relative humidity of the surrounding while also the module converts this capacitance change of values into analog voltage impulses.	No unsolved proplems appeared in this paper.	No future work is proposed.
17.	“Smart Embedded Framework using Arduino and IoT for Real-Time Noise and Air Pollution Monitoring and Alert system”	Janeera. D.A Poovizhi .H Sheik Haseena .S.S	2021	IEEE	This paper provides a solution for noise and air pollution level using wireless embedded computing system. All devices in the system inclusive of ESP8266, Xmega 2560, sound sensor, dust, gas, humidity and temperature sensor as well as Wi-Fi are connected via Internet of Things technology.	Firstly, the Arduino Uno processor is used for noise and air pollution monitoring system. Secondly Port 9600 is used for providing a 5V DC power supply to the Arduino board while on the other hand sound and gas sensors are used for monitoring the change in levels of these measurements.	No unsolved proplems appeared in this paper.	Future work is focused towards installing multiple components prototypes and testing the efficiency of the system by increasing the trials as well as commercializing this project as a final real world product.

18.	"Design and Development of Arduino Based Portable Air Quality Monitoring System"	Shubhajit Vishwas Shyam S Kundu Amitabha Nath	2021	IEEE	In this paper, an Arduino microcontroller-based air quality monitoring system has been developed for measuring the various atmospheric parameters viz. carbon-monoxide, temperature, humidity, and methane using the respective sensors.	In this paper, a low cost and less complex Arduino based air quality monitoring system was designed. The sensors can measure data in an ambient environment. It will be helpful in monitoring the air pollution of a particular location/environment.	No unsolved proplems appeared in this paper.	No future work is proposed.
19.	"Electronic System for Real-Time Indoor Air Quality Monitoring"	Felix-Constantin Adochiei , Erban – Teodor Nicolescu , Loana-Raluca Adochiei	2020	IEEE	A system capable of performing real-time measurements of a wide parameters such as: CO, EtOH, H2, H2S, NO2, O3, PM1.0, PM2.5, PM4.0, PM10, RH/T, Rn, SO2, typical PM size, (TPS) proposed and constructed.	The system is represented by an electronic noise through which the ambient air parameters are measured in real-time and recorded for further investigation.	No unsolved proplems appeared in this paper.	Future directions include collecting data for different time intervals and analyzing the variation of the monitored parameters.

20.	"Development of Arduino based air quality monitoring systems for measuring gas sensor"	Aimudo O.S Usikalu M.R Ayara W.A	2023	IEEE	It is proposed here an air quality monitoring system that is portable to give real time updates on the air quality around. The system contains an air quality sensor, which measures the air quality.	The solutions is conducted and used to ensure that the device shows a level of being accurate and reliable. This was achieved by comparing the results from the device with those of standard results.	No unsolved proplems appeared in this paper.	It can be improved by implementing a temperature and humidity sensor, a wireless network card to this monitoring system to save values from sensors while also another feature of particle matter measurement can be added as well.
21.	"IoT-Enabled Air Pollution Monitoring Systems: Technologies, Solutions, and Challenges in trending technologies"	Surleen Kaura, Sandeep Sharma	2023	IEEE	This paper discusses all the key technologies which support IoT-enabled systems for air pollution sensing and monitoring.	Solutions such as the suggesting alternate routes to drivers, have been propose making path to the wide range of enabling technologies offered by IoT.	Regarding the selection of sensors and the accuracy of the dataset. Deployment of sensor nodes is another concern; supply and can sense pollution data accurately and correctly.	Firstly, future work is to validate the credibility of the data gathered by the sensors, secondly the maintenance, in the long run, needs to be addressed. Furthermore, there is high dependability on cloud platforms for storage and computational purposes.

The Code Implemented in Arduino IDE to run our circuit.

```
#include "SoftwareSerial.h"
#include "MHZ19.h"
#include "PMS.h"
#include "MQ131.h"
#include "dht.h"
#include "DS3231.h"
#define led 13
#define tvocPin 7
#define dht22 5
dht DHT;
DS3231 rtc(SDA, SCL);
Time t;
MHZ19 myMHZ19;
SoftwareSerial co2Serial(2, 3);
SoftwareSerial pmsSerial(8, 9);
PMS pms(pmsSerial);
PMS::DATA data;
unsigned long dataTimer = 0;
unsigned long dataTimer3 = 0;
unsigned long dataTimer4 = 0;
int readDHT, temp, hum;
int CO2;
int o3;
int tvoc;
int pm25;
int hours, minutes;
int previousMinutes = 1;
String timeString;
String receivedData = "Z";
uint8_t tempData[96] = {};
uint8_t humData[96] = {};
uint8_t tvocData[96] = {};
uint8_t co2Data[96] = {};
uint8_t pm25Data[96] = {};
uint8_t o3Data[96] = {};
int8_t last24Hours[12] = {};
int yAxisValues[4] = {};
int maxV = 0;
int8_t r = 99;
void setup() {
  Serial.begin(9600);
  pinMode(6, OUTPUT);
  pinMode(tvocPin, OUTPUT);
  digitalWrite(6, HIGH);
  digitalWrite(tvocPin, HIGH);
  delay(20 * 1000);
  digitalWrite(6, LOW);
  digitalWrite(tvocPin, LOW);
  rtc.begin();
  co2Serial.begin(9600);
  pmsSerial.begin(9600);
  myMHZ19.begin(co2Serial);
  myMHZ19.autoCalibration(false);
  MQ131.begin(6, A0,
LOW_CONCENTRATION,
1000000);
  MQ131.setTimeToRead(20);
```

```
  MQ131.setR0(9000);
}
void loop() {
  readDHT = DHT.read22(dht22);
  temp = DHT.temperature;
  hum = DHT.humidity;
  digitalWrite(tvocPin, HIGH);
  delay(5000);
  tvoc = analogRead(A1);
  digitalWrite(tvocPin, LOW);
  checkForIncomingData();
  co2Serial.listen();
  dataTimer = millis();
  while (millis() - dataTimer <= 3000)
{
    CO2 = myMHZ19.getCO2();
  }
  checkForIncomingData();
  pmsSerial.listen();
  dataTimer3 = millis();
  while (millis() - dataTimer3 <=
1000) {
    pms.readUntil(data);
    pm25 = data.PM_AE_UG_2_5;
  }
  checkForIncomingData();
  MQ131.sample();
  o3 = MQ131.getO3(PPB);
  checkForIncomingData();
  t = rtc.getTime();
  hours = t.hour;
  minutes = t.min;
  storeData();
  dataTimer4 = millis();
  while (millis() - dataTimer4 <= 200)
{
    Serial.print("co2V.val=");
    Serial.print(CO2);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("pm25V.val=");
    Serial.print(pm25);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("o3V.val=");
    Serial.print(o3);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("tempV.val=");
    Serial.print(temp);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("humV.val=");
    Serial.print(hum);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.print("tvocV.val=");
    Serial.print(tvoc);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
  }
}
```

```
void checkForIncomingData() {
  if (Serial.available() > 0) {
    receivedData = Serial.readString();
    delay(30);
    if (receivedData == "0") {
      r = 0;
    }
    if (receivedData == "1") {
      r = 1;
    }
    if (receivedData == "2") {
      r = 2;
    }
    if (receivedData == "3") {
      r = 3;
    }
    if (receivedData == "4") {
      r = 4;
    }
  }
  if (r == 0 || r == 1 || r == 2 || r == 3 || r
== 4) {
    delay(200);
    dataTimer3 = millis();
    while (millis() - dataTimer3 <=
200) {
      Serial.print("pageSwitch.val=");
      Serial.print(1);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
    }
    delay(100);
    getLast24Hours();
    getYAxisValues();
    sendDataToWaveform();
    r = 99;
  }
}
void storeData() {
  if ((minutes - previousMinutes) >=
15) {
    memmove(tempData,
&tempData[1], sizeof(tempData));
    tempData[sizeof(tempData) - 1] =
temp;
    memmove(humData, &humData[1],
sizeof(humData));
    humData[sizeof(humData) - 1] =
hum;
    memmove(tvocData, &tvocData[1],
sizeof(tvocData));
    tvocData[sizeof(tvocData) - 1] =
map(tvoc, 0, 1000, 0, 255);
    memmove(co2Data, &co2Data[1],
sizeof(co2Data));
    co2Data[sizeof(co2Data) - 1] =
map(CO2, 0, 3000, 0, 255);
    memmove(pm25Data,
&pm25Data[1], sizeof(pm25Data));
    pm25Data[sizeof(pm25Data) - 1] =
map(pm25, 0, 1000, 0, 255);
    memmove(o3Data, &o3Data[1],
sizeof(o3Data));
    o3Data[sizeof(o3Data) - 1] =
map(o3, 0, 1000, 0, 255);
    previousMinutes = minutes;
  } else if ((minutes - previousMinutes)
== -45) {
    memmove(tempData,
&tempData[1], sizeof(tempData));
    tempData[sizeof(tempData) - 1] =
temp;
```

```
    memmove(humData, &humData[1], sizeof(humData));
    humData[sizeof(humData) - 1] = hum;
    memmove(tvocData, &tvocData[1], sizeof(tvocData));
    tvocData[sizeof(tvocData) - 1] = map(tvoc, 0, 1000, 0, 255);
    memmove(co2Data, &co2Data[1], sizeof(co2Data));
    co2Data[sizeof(co2Data) - 1] = map(CO2, 0, 3000, 0, 255);
    memmove(pm25Data, &pm25Data[1], sizeof(pm25Data));
    pm25Data[sizeof(pm25Data) - 1] = map(pm25, 0, 1000, 0, 255);
    memmove(o3Data, &o3Data[1], sizeof(o3Data));
    o3Data[sizeof(o3Data) - 1] = map(o3, 0, 1000, 0, 255);
    previousMinutes = minutes;
  }
}

void getLast24Hours() {
  for (int i = 11; i >= 0; i--) {
    last24Hours[11] = hours;
    last24Hours[i - 1] = last24Hours[i] - 2;
    if (last24Hours[i - 1] < 0) {
      for (int k = -0; k > -11; k--) {
        if (last24Hours[i - 1] == k) {
          last24Hours[i - 1] = 24 + k;
        }
      }
    }
  }
  for (int i = 0; i < 12; i++) {
    String last24 = ("n") + String(i) + String(".val=") + String(last24Hours[i]);
    Serial.print(last24);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    delay(20);
  }
  for (int i = 0; i < 12; i++) {
    String last24 = ("n") + String(i) + String(".val=") + String(last24Hours[i]);
    Serial.print(last24);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    delay(20);
  }
}
void getYAxisValues() {
  maxV = 0;
  switch (r) {
    case 0:
      for (int i = 0; i < sizeof(tempData); i++) {
        if (tempData[i] > maxV) {
          maxV = tempData[i];
        }
      }
      break;
    case 1:
      for (int i = 0; i < sizeof(humData); i++) {
        if (humData[i] > maxV) {
```

```
        maxV = humData[i];
      }
    }
    break;
  case 2:
    for (int i = 0; i < sizeof(tvocData);
i++) {
      if (tvocData[i] > maxV) {
        maxV = tvocData[i];
      }
    }
    break;
  case 3:
    for (int i = 0; i < sizeof(co2Data);
i++) {
      if (co2Data[i] > maxV) {
        maxV = co2Data[i];
      }
    }
    break;
  case 4:
    for (int i = 0; i <
sizeof(pm25Data); i++) {
      if (pm25Data[i] > maxV) {
        maxV = pm25Data[i];
      }
    }
    break;
  case 5:
    for (int i = 0; i < sizeof(o3Data);
i++) {
      if (o3Data[i] > maxV) {
        maxV = o3Data[i];
      }
    }
    break;
  }
  int step = maxV / 4;
  for (int i = 0; i < 4; i++) {
    yAxisValues[i] = step + step * i;
  }
  for (int i = 0; i < 4; i++) {
    String yVal = ("y") + String(i) +
String(".val=") +
String(yAxisValues[i]);
    Serial.print(yVal);
    Serial.write(0xff);
    Serial.write(0xff);
    Serial.write(0xff);
    delay(20);
  }
}
void sendDataToWaveform() {
  String set;
  switch (r) {
    case 0:
      for (int i = 0; i < sizeof(tempData);
i++) {
        set = "add 1,0," +
String(tempData[i]);
        Serial.print(set);
        Serial.write(0xff);
        Serial.write(0xff);
        Serial.write(0xff);
        delay(20);
      }
      break;
    case 1:
      for (int i = 0; i < sizeof(humData);
i++) {
```

```
      set = "add 1,0," + String(humData[i]);
      Serial.print(set);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      delay(20);
    }
    break;
  case 2:
    for (int i = 0; i < sizeof(tvocData); i++) {
      set = "add 1,0," + String(tvocData[i]);
      Serial.print(set);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      delay(20);
    }
    break;
  case 3:
    for (int i = 0; i < sizeof(co2Data); i++) {
      set = "add 1,0," + String(co2Data[i]);
      Serial.print(set);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      delay(20);
    }
    break;
  case 4:
    for (int i = 0; i < sizeof(pm25Data); i++) {
      set = "add 1,0," + String(pm25Data[i]);
      Serial.print(set);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      delay(20);
    }
    break;
  case 5:
    for (int i = 0; i < sizeof(o3Data); i++) {
      set = "add 1,0," + String(o3Data[i]);
      Serial.print(set);
      Serial.write(0xff);
      Serial.write(0xff);
      Serial.write(0xff);
      delay(20);
    }
    break;
  }
}
```