SMART GREENHOUSE SYSTEM

Senior Design Project II

Naci Gökberk Tandoğan

Samet Refik Derbentli

2023

MEF UNIVERSITY
FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

SMART GREENHOUSE CARBON DIOXIDE TRACKER

Senior Design Project II

Naci Gökberk Tandoğan

Samet Refik Derbentli

Advisor: Dr. Tuba Ayhan

2023

MEF UNIVERSITY
FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

Project Title	: Smart Greenhouse Carbon Dioxide Tracker
Student(s) Name	: Naci Gökberk Tandoğan
	Samet Refik Derbentli
Date	: 15/06/2023

I hereby state that the design project prepared by Tuba Ayhan has been completed under my supervision. I accept this work as a "Senior Design Project."

15/06/2023
Tuba Ayhan

I hereby state that I have examined this senior design project by Serap Kırbız-Şimşek. This work is acceptable as a "Senior Design Project."

15/06/2023
Serap Kırbız-Şimşek

Head of the Department of
Electrical and Electronics Engineering

ACADEMIC HONESTY PLEDGE

In keeping with MEF University Student Code of Conduct, I pledge that this work is my own and that I have not received inappropriate assistance in its preparation. I further declare that all resources are explicitly cited.

NAME	DATE	SIGNATURE
Naci Gökberk Tandoğan	15/06/2023	
Samet Refik Derbentli	15/06/2023	

ABSTRACT

SMART GREENHOUSE SYSTEM

Naci Gökberk Tandoğan
Samet Refik Derbentli

MEF UNIVERSITY
Faculty of Engineering
Department of Electrical and Electronics Engineering

Advisor: Tuba Ayhan

JUNE, 2023

The main idea of the project is to transmit carbon dioxide, humidity and temperature data in greenhouses to the user in a cheap, easy and fast way. In addition, it will be prevented from exceeding the upper limit of the optimal carbon dioxide level determined for plant growth. Blocking will be done with the help of a fan. The fan will activate when the maximum amount of carbon dioxide required for plant growth in the greenhouse is exceeded. Two separate systems will work on the project. The first system, which is the agent, will collect the data received from the greenhouse and send it to the host system. At the same time, it will decide when the fan will be activated and transmit the status of the fan to the host. The host part of the project will upload the data to a server. And finally, there will be an app and app that will display the data to the user. The app will be easy to understand and use. Our aim to produce cheaply will be achieved by imitating the data we receive from the high-precision infrared sensor through electro-chemical sensors and by choosing the cheapest of the other components used, which will provide the minimum expectation. Electro-chemical sensors to be used are cheaper than infrared sensors. The project will enable the greenhouse producer to monitor the carbon dioxide level regularly and uninterruptedly through the application and the website. Due to its cheapness, it is aimed to be easily accessible by every greenhouse producer and it is aimed to increase greenhouse efficiency throughout the country. The host part of the project will send sensor data to the agent part over Lo-Ra every 2-3 minutes. During this communication, data will be encrypted to prevent parasitic data.

Keywords: Greenhouse, Gas Sensor, Microcontroller, Arduino IoT Cloud, Lo-Ra

ÖZET

AKILLI SERA SİSTEMİ

Naci Gökberk Tandoğan
Samet Refik Derbentli

MEF ÜNİVERSİTESİ
Mühendislik Fakültesi
Elektrik-Elektronik Mühendisliği Bölümü

Tez Danışmanı: Tuba Ayhan

HAZİRAN, 2023

Projenin ana fikri, seralardaki karbondioksit, nem ve sıcaklık verilerini ucuz, kolay ve hızlı bir şekilde kullanıcıya iletmektir. Bunların yanı sıra, bitki gelişimi için belirlenen optimal karbondioksit seviyesinin üst sınırını geçmesi engellenecektir. Engelleme işlemi bir fan yardımı ile yapılacaktır. Fan, sera içinde bitki büyümesi için gerekli olan maksimum karbondioksit miktarı geçilince aktive olacak. Projede 2 ayrı sistem çalışacaktır. Agent olan ilk sistem, seradan alınan verileri toplayacak ve host sistemine gönderecektir. Aynı zamanda fanın ne zaman devreye gireceğine karar verip fanın durumunu da host kısmına iletecektir. Projenin host kısmı, verileri bir sunucuya yükleyecektir. Ve son olarak, verileri kullanıcıya gösterecek bir uygulama ve aplikasyon olacak. Uygulamanın anlaşılması ve kullanılması kolay olacaktır. Ucuza üretme amacımız, yüksek hassasiyetli kızılötesi sensörden aldığımız verilerin elektro-kimyasal sensörler aracılığıyla taklit edilmesiyle ve kullanılan diğer komponentlerin minimum beklentiyi sağlayacak en ucuz olanlarından seçilerek sağlanacaktır. Kullanılacak elektro-kimyasal sensörler, kızılötesi sensörlere göre daha ucuzdur. Projede uygulama ve web sitesi üzerinden sera üreticisine karbondioksit seviyesinin düzenli ve kesintisiz olarak takip edilebilmesini sağlayacaktır. Bu amaçla kolay bir takip sistemi kullanmak ve üretim verimliliğini optimum seviyede tutmak hedeflenmektedir. Projenin host kısmı, her 2-3 dakikada bir Lo-Ra üzerinden agent kısmına sensör verilerini gönderecektir. Bu iletişim sırasında veriler şifrelenerek parazit veriler engellenecektir.

Anahtar Kelimeler: Sera, Gaz Sensörü, Mikrokontrolcü, Arduino IoT Cloud, Lo-Ra

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

PPM	Parts Per Million
PWM	Pulse Width Modulation
Lo-Ra	Long Range
Wi-Fi	Wireless Fidelity
CO_2	Carbon Dioxide
APP	Application
UART	Universal Asynchronous Receiver Transmitter
dBm	Decibel Milliwatt
ISM	The Industrial, Scientific, and Medical
RF	Radio Frequency
SPI	Serial Peripheral Interface
GPIO	General-Purpose Input/Output
USB	Universal Serial Bus
ADC	Analog-to-Digital Converter
MISO	Master In Slave Out
MOSI	Master Out Slave

INTRODUCTION

A greenhouse is an isolated structure to provide suitable environmental conditions for growing plants. Temperature, humidity, light and carbon dioxide and irrigation level are important factors for the efficiency of a greenhouse [1]. Today, monitoring and management systems are becoming widespread in agriculture or greenhouse areas due to the developments in new sensors and electronic communication systems [2] [3]. By using computer-aided monitoring systems in agricultural fields or greenhouse systems, soil and cultivated soils of agricultural products can be tracked [2]. Thanks to these systems, the need for labour and production time can be optimized by increasing the productivity of the soil by providing light, temperature, humidity, carbon dioxide (CO_2), plant nutrients and irrigation at the required time intervals to ensure optimum efficiency [1]. Greenhouse farmers cannot accurately determine the amount of carbon dioxide in the greenhouse. They try to understand the situation inside the greenhouse only manually and by feeling themselves. Experiences play a larger role in their day-to-day operations [4]. To facilitate these daily operations, we aim to make an inexpensive, accessible and highly accurate greenhouse monitoring system. The importance and original value of the subject and the research question are given below with reference to the literature.

Importance of the Subject: Keeping the amount of carbon dioxide at *1200 ppm* is very important for maximizing photosynthetic reactions [5]. It has been observed that the growth rate of plants can increase by *40%* when the amount of carbon dioxide is kept at *1200 ppm*. Therefore, it is necessary to constantly monitor the amount of carbon dioxide [5].

Unique Value of Research Proposal: Existing systems are not affordable to greenhouse producers of all levels, and costly to administer [3]. On the other hand, we want to create a product that can be easily assembled, parts are often found and inexpensive.

Research Question/Hypothesis: Developing an affordable and readily deployable system with a user-friendly interface.

1.1. Motivation

Greenhouse owners predominantly rely on manual observations and assessments to understand the prevailing conditions inside their greenhouses. They heavily rely on their experience and expertise gained over time to make informed decisions and adjustments in their day-to-day operations [4]. However, this manual approach can be time-consuming, subjective, and prone to human error.

To overcome these challenges and streamline daily operations, we have developed an innovative greenhouse monitoring system. This system provides a cost-effective solution accessible to greenhouse owners of varying scale and resources. Leveraging sensors and microcontroller technologies, our monitoring system provides accurate and real-time information on key environmental factors inside the greenhouse, including temperature, humidity, carbon dioxide. It also offers the chance to easily monitor this information from any location.

Overall, our inexpensive, accessible, and accurate greenhouse monitoring system empowers greenhouse owners to make data-driven decisions, optimize resource utilization, and ultimately enhance the productivity and yield of their greenhouse operations. By reducing dependency on manual observations and leveraging technology, we aim to advance IoT technology in the way greenhouse monitoring is conducted, providing a more efficient and reliable approach to greenhouse management.

1.2. Broad Impact

It is aimed to offer a cheaper and simpler system to the farmers or greenhouse owners, as the cost of our project will be cheaper than the previous examples. After the hardware and software part of our project is completed, it will be tested in a small greenhouse environment and improvements will be made according to the results obtained. Our goal in the project is to ensure that the results can be easily monitored over the phone app via internet, that the user can monitor their data continuously, and that this situation is sustainable. In this way, it is aimed to increase the efficiency of greenhouse production and to keep the CO_2 level at the optimum level.

1.2.1. Public Health, Safety, and Welfare

The project contains electronic materials. Thus, we have to be aware of IEEE electronic disposal standards. There is a human factor in this project. This means we do have to pay attention to the human life regulations standards of Turkey. There are regulations about electricity usage because the product will use city electricity. We have to pay attention to the electricity usage of our product also. These standards are given below.

ISO 14001 is a standard that provides guidelines for environmental management systems, including the disposal of electronic waste. It promotes responsible and sustainable practices for managing the environmental impacts of electronic waste throughout its lifecycle [15].

The Waste Electrical and Electronic Equipment Directive (WEEE Directive) is a European Union directive that sets the guidelines for the disposal and recycling of electronic waste in EU member states. It outlines the responsibilities of producers, distributors, and consumers in managing electronic waste in an environmentally friendly manner [16].

The Basel Convention on the Control of Transboundary Movements of Hazardous Wastes and Their Disposal is an international treaty aimed at minimizing the generation and transboundary movement of hazardous waste, including electronic waste. It establishes guidelines for the environmentally sound management of electronic waste and promotes the reduction, recycling, and safe disposal of such waste [17].

The e-Stewards certification program provides a set of standards for responsible electronic waste recycling and disposal. It ensures that certified recyclers adhere to strict environmental and social criteria, including the prohibition of exporting hazardous electronic waste to developing countries [18].

LoRa uses chirp spread spectrum modulation to enable long-range communication with low power consumption. It can operate in different frequency bands, including 433 MHz, 868 MHz, and 915 MHz, depending on the region and regulatory requirements. In Asia, the commonly used frequency bands for LoRa communication are 433 MHz and 920-925 MHz [11].

1.2.2. Global, Cultural, Social, Environmental, and Economic Impact

The global impact of implementing a smart greenhouse CO_2 tracker is significant, particularly in the realm of climate change mitigation. This innovative technology plays a crucial role in contributing to global efforts aimed at combating climate change. By effectively monitoring and managing greenhouse gas emissions in agricultural practices, the smart greenhouse CO_2 tracker helps reduce the overall carbon footprint associated with greenhouse operations. This, in turn, leads to a more sustainable and environmentally friendly approach to food production. By actively monitoring and tracking CO_2 levels within greenhouses, farmers and agricultural experts gain valuable insights into the environmental impact of their practices, allowing them to make informed decisions and adopt more sustainable methods. Ultimately, the adoption of smart greenhouse CO_2 trackers on a global scale contributes to the reduction of greenhouse gas emissions, thereby playing a vital role in mitigating climate change.

The cultural impact of the smart greenhouse CO_2 tracker is significant, particularly in the realm of food production. This innovative technology has the potential to enhance agricultural productivity, leading to a consistent and high-quality food supply. By improving food security, the smart tracker contributes to the preservation of cultural practices and traditional food systems associated with agriculture. It empowers farmers to continue practicing their cultural agricultural methods while adopting more sustainable and efficient approaches.

The social impact of greenhouse CO_2 is its impact on working conditions by accurately controlling CO_2 levels, fostering a safer and healthier environment for greenhouse workers, and reducing the risks of high CO2 exposure, thereby promoting worker well-being and safety. Moreover, this technology has the potential to improve hybrid operating conditions. Additionally, the project contributes to knowledge sharing and skill development among agricultural communities, empowering farmers and greenhouse operators to adopt sustainable practices and enhancing their expertise in the field.

The environmental impact of the greenhouse CO_2 tracker includes reduced energy consumption and resource conservation. By accurately monitoring and controlling CO_2 levels, the technology optimizes temperature and humidity within greenhouses, resulting in decreased energy usage. This leads to lower greenhouse gas emissions and reduces the overall environmental footprint of agricultural operations.

The economic impact of the smart greenhouse CO_2 tracker is that the optimization of CO_2 levels in greenhouses can enhance crop yield, accelerate growth cycles, and improve the overall quality of greenhouse produce. With the efficient utilization of resources and reduced energy consumption, the project generates cost savings. This translates into increased profitability for farmers and greenhouses owners.

2. PROJECT DEFINITION

The components to be used in the project are Arduino Uno R3, ESP32 Wemos D1 R32, MQ-135, Lo-Ra Ra-02, 5V Dc Fan, DHT-22 Humidity & Temperature sensor and TIP31c NPN transistor. Components are carefully selected for their lifetime, precision, and ease of use, while special attention will be paid to making it as budget friendly as possible within the limits of our target in terms of cost. In addition, the data display on the app has to be clear and simple. Thus, every user can easily understand and be easily interpreted. The system measures the carbon dioxide level at more than one point test and transmits its average to the user via the app. The sensitivity of the sensors we will use will be at least 100 ppm. This is important for observing the stable change of data. In our interface on the phone, the carbon dioxide average of the instant sensors, the carbon dioxide average of the daily sensors, and also the carbon dioxide average of the weekly sensors will be displayed. The sensitivity that is aimed is acceptable for monitoring plant growth [5].

The reason for this is 100 ppm is an accessible and testable value, since the sensitivity of the inexpensive electro-chemical sensors we will use is 100 ppm, and the sensitivity of the infrared sensor we will use is 50 ppm. Sensor cards will be designed to send measurements to the motherboard at least every 10 minutes. Figure 1. presents an exemplary graph showcasing the fluctuation in CO_2 levels, which will be prominently featured in our application.

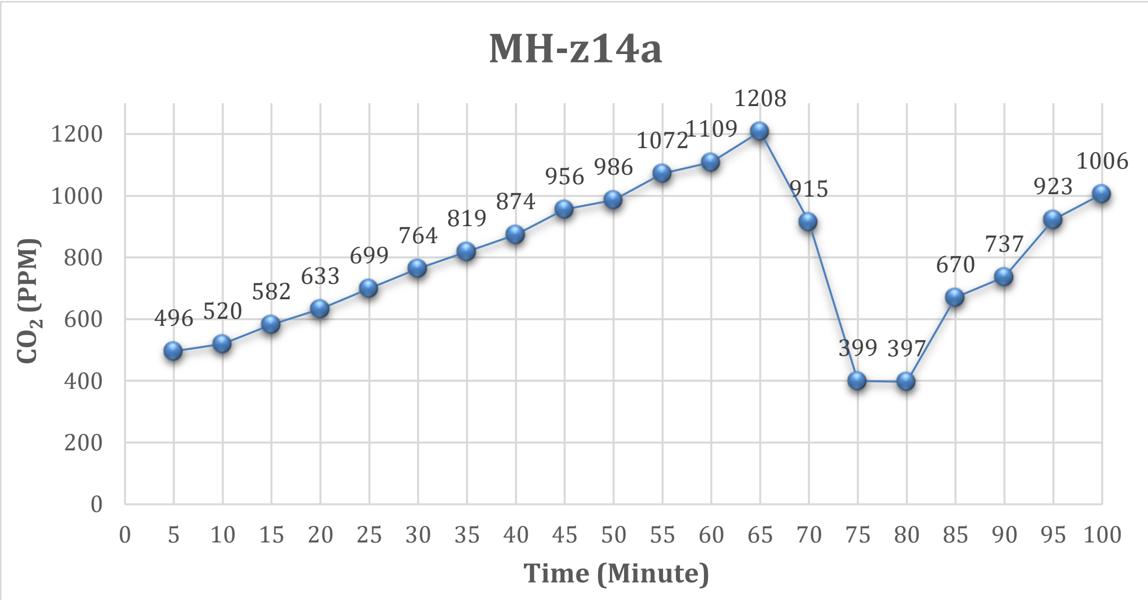

Figure 1. MH-Z14a Change Carbon Dioxide Graph

3. THEORETICAL BACKGROUND

3.1. Literature Survey

Selecting the components is the most important part of the project. A survey about greenhouses and the relationship between plants and CO_2 is a guide of component selection stage. As a gas sensor, we have turned to analog gas sensors that are both budget-friendly and have a long working life. Such gas sensors transmit information analogously to the microcontroller. Based on this, we decided that the microcontroller we will choose should be a microcontroller that we can also use analog data.

When the literature related to the interface was scanned, it was seen that the CO_2 and humidity values in greenhouses were monitored through the Android Application in previous projects [19]. Android APP function module mainly includes real-time display of greenhouse environmental parameter data, dynamic real-time control of the greenhouse environmental enforcement agency, static display of related technical web pages, and other functions. The user can check the greenhouse environment parameters at any time and control the greenhouse environment to keep the greenhouse environment at its best. Interface used in project Figure 2. is also shown.

Optimal carbon dioxide values for plant development 77% of the emissions that cause an increase in carbon dioxide concentration are of fossil fuel origin, and 23% are due to the destruction of large, forested areas. When this rate drops to around 150 ppm, plants become unable to photosynthesize and begin to fade. When this rate decreases further and reaches around 100 ppm, photosynthesis stops completely, and plants die. the level of 100 ppm is the lowest limit for plant life. For optimal photosynthesis, the amount of CO2 should be 1200 ppm [4]. The amount of CO_2 in a plant-grown environment can be increased to 1200-1500 ppm [5]. As result, a fan was placed in the smart greenhouse system, and it was deducted from running the fan when the CO_2 PPM exceeded 1500. With this method, it was aimed to keep the CO_2 level close to the optimum. Detailed CO2 limit values are shown in detail in Table 1.

Figure 2. Example Android Application UI [19]

Table 1. Carbon dioxide levels and possible effects for plants [5]

Carbon Dioxide limit values (ppm)	Possible effects on the plant
10000	Plants die.
1200	Plants provide the best development.
350	The natural value in the air, there is no effect.
150	Photosynthesis stops.
100	Plants die.

3.1.1 Sensor Survey

Infrared sensor sensitivity and stability are high, but the price is high. There is no electrochemical sensor that reacts only to carbon dioxide. Electrochemical gas sensors are instruments that produce current by performing reduction or oxidation reactions as a result of contact with an electrode located in a target gas and measure the concentration of gas thanks to this generated current [6].

We aim to simulate a sensor with high sensitivity and accuracy, such as an infrared sensor, under greenhouse conditions with several chemical sensors. To do this, a greenhouse system will be installed. Infrared sensor and chemical sensors data will be collected into the installed greenhouse system. These collected data will be monitored by connecting to a computer via Analog Discovery.

The infrared sensor will send PWM, analog value will be received from other sensors. Planned to be used in electrochemical sensors*: MQ-4* natural gas and methane gas metering sensor, *MQ-7* carbon monoxide measurement sensor *MQ-135* air quality sensor measuring; if the infrared sensor is used which is planned to be *MH-Z14a* Sensor carbon dioxide [7].

3.1.2 Data Transmission Selection

Table 2. Comparison technologies used for agricultural communication connection [9]

Parameters	Data Rate	Frequency Band	Transmission Range	Energy Consumption
Bluetooth	1–24 Mb/s	24 GHz	8–10 m	Very low
LoRa	0.3–50 kb/s	868/900 MHz	<30 km	Very low
RFID	40 to 160 kb/s	860–960 MHz	1–5 m	Low
ZigBee	20–250 kb/s	2.4 GHz	10–20 m	Low
Mobile	200 kb/s (3G) 0.1–1 Gb/s (4G)	865 MHz, 2.4 GHz	Entire Cellular Area	Low
Wi-Fi	1 Mb/s–7 Gb/s	5 GHz–60 GHz	20–100 m	High

Long ranged and low energy consuming data transfer method is needed to complete the aim for the project. So, different types of communication methods are searched. After comparing these types of communication methods, Lo-Ra for the communication method to be used in the project.

Lo-Ra's transmission range is the highest compared to other technologies (except mobile). The fact that the energy consumption rate is extremely low has also been an important factor for us, as we aim to keep the cost low. The detailed comparison can be seen in Table 2.

3.1.3 Application & User Interface Selection

In the application & UI Selection, our pursuit involved exploring options for transferring the data acquired through the microcontroller to the designated server and subsequently transmitting it to the application. Considering the modest volume of data intended for transmission and storage, we have judiciously concluded that Arduino IoT Cloud is the optimal choice [10].

Arduino IoT Cloud is a platform developed by Arduino that presents a swift and effortless solution for Internet of Things (IoT) endeavors. This cloud-based service facilitates the connection and management of Arduino-based devices on the Internet. Arduino IoT Cloud delivers a user-friendly interface that expedites the seamless configuration and monitoring of devices. The platform adeptly facilitates the collection, analysis, and dissemination of data from sensors, actuators, and other electronic components.

Accessible through a web-based user interface and a dedicated mobile application, Arduino IoT Cloud empowers users to vigilantly oversee the status of Arduino devices, continuously monitor real-time sensor data, and exert remote control over their devices. Moreover, users are afforded the opportunity to establish rules and triggers, ensuring the receipt of notifications via mediums such as emails or instant alerts, thereby optimizing the operational efficiency of their devices with Arduino IoT Cloud.

With its flexible infrastructure, Arduino IoT Cloud accommodates diverse devices via an array of protocols and communication channels. This inclusivity streamlines the integration process for various sensors, microcontrollers, and supplementary components. Furthermore, the platform provides application programming interfaces (APIs) that facilitate harmonious integration with cloud-based services and other IoT platforms.

The tenets of data security are of paramount importance to Arduino IoT Cloud, thereby fostering a secure environment for users. Stringent measures, including data encryption and the utilization of secure communication protocols, fortify the protection and confidentiality of users' devices and data.

In summation, Arduino IoT Cloud offers a user-centric, secure, and versatile solution for IoT undertakings. By means of this platform, Arduino devices can be effortlessly connected to the internet, granting users the capacity to diligently monitor data, exert remote control over devices, and seamlessly integrate with diverse IoT platforms. For these reasons, the use of the Arduino IoT Cloud platform in the project.

4. DESIGN

Our project consists of 1x Arduino Uno R3, 1x MQ-135 sensor, 1x Temperature & Humidity Sensor, 1x 5V DC Fan, 2x Lo-Ra Ra-02 and 1x ESP32 Wemos D1 R32.

The system consists of 2 parts as Transmitter and Receiver. In the transmitter part, MQ-135 Air quality sensor, Temperature and Humidity Sensor and Fan are connected to Arduino Uno. The aim is to measure the amount of carbon dioxide in the greenhouse environment and keep it at the optimum level, and in cases where the amount of carbon dioxide increases too much *(When CO_2 PPM >1500)*, the fan is operated autonomously, and the carbon dioxide level is returned to the optimum level by increasing the amount of ventilation.

Temperature & Humidity sensor is placed to provide additional information to the user and to provide the most suitable humidity and temperature conditions.

The data received from Arduino UNO communicates with the ESP32 Wemos D1 R32 via Lo-Ra, and the data is transferred to the Arduino IoT Cloud by establishing a Wi-Fi connection with the Wi-Fi module in the ESP32. Project Diagram can be seen in detail in Figure 3.

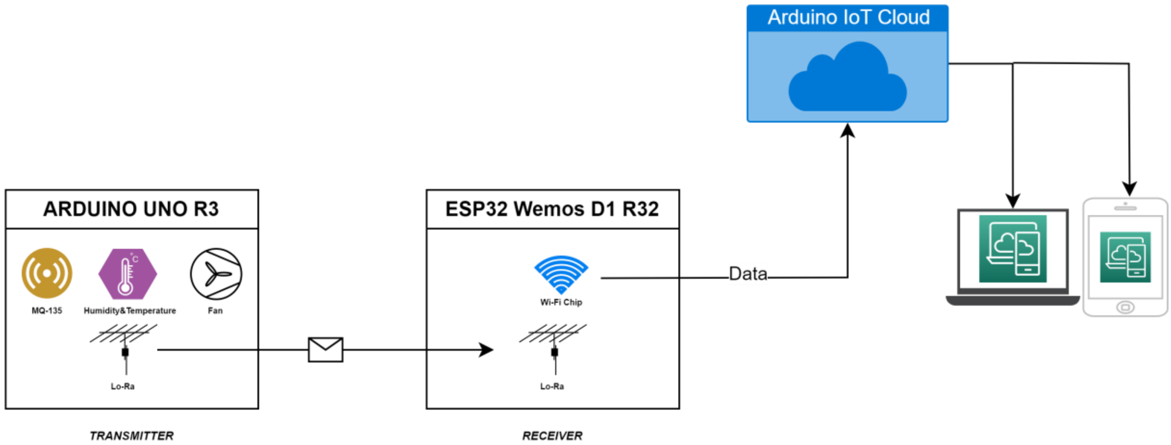

Figure 3. Project Diagram

4.1 COMPONENTS

4.1.1 Lo-Ra

The SX1278 Lo-Ra transceiver module is a versatile wireless communication device designed for long-range, low-power applications. It operates in the ISM bands, such as 433MHz, 868MHz, or 915MHz, depending on the region. Lo-Ra offers programmable RF output power levels, typically ranging from 5dBm to 20dBm, and high receiver sensitivity, typically around -137dBm to -148dBm [7]. The module supports various data rates, allowing flexible communication based on bandwidth and range requirements. It utilizes standard interfaces like SPI for seamless integration with microcontrollers or host devices. The SX1278 is known for its low power consumption, making it suitable for battery-powered applications. With line-of-sight ranges of several kilometers achievable in ideal conditions. Lo-Ra SX1278 module can be seen in Figure 4.

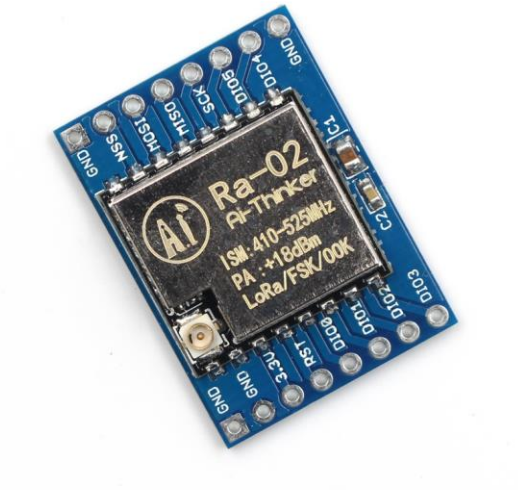

Figure 4. Lo-Ra SX1278 Module [7]

4.1.2 Wemos D1 R32

The Wemos D1 R32 is a development board based on the ESP32 microcontroller. The board features the ESP32 module, which is powered by a dual-core Tensilica LX6 processor running at up to 240MHz [13]. It includes built-in Wi-Fi and Bluetooth connectivity, enabling seamless wireless communication. The Wemos D1 R32 offers a variety of interfaces and peripheral options, including GPIO pins, I2C, SPI, UART, and PWM, allowing for versatile sensor and device integration. It also provides onboard USB-to-serial conversion for programming and debugging. Additionally, the Wemos D1 R32 supports a wide range of libraries and has excellent community support, making it a popular choice for IoT projects requiring connectivity, processing power, and a compact form factor. Wemos D1 R32 module can be seen in Figure 5.

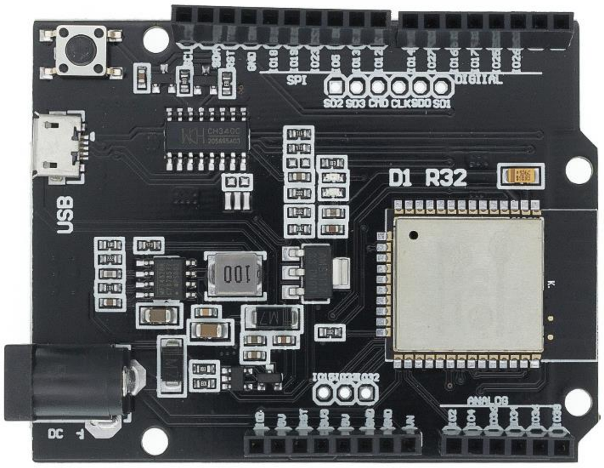

Figure 5. Wemos D1 R32 [7]

4.1.3 Arduino Uno R3

The Arduino Uno R3 is a microcontroller board based on the ATmega328P microcontroller. The board features 14 digital input/output (I/O) pins, among which 6 can be used as PWM (Pulse Width Modulation) outputs. It also includes 6 analog input pins for reading sensor data. The Arduino Uno R3 supports a clock speed of 16MHz and offers 32KB of flash memory for storing program code. It comes with built-in USB connectivity for easy programming and communication with a computer. The Arduino Uno R3 can be powered via USB or an external power source and supports a wide range of shields, allowing for easy expansion and customization. Arduino UNO R3 can be seen in Figure 6.

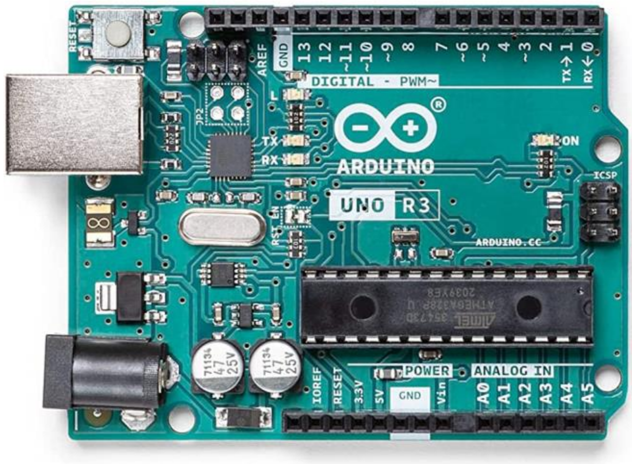

Figure 6. Arduino Uno R3 [7]

4.1.4 DHT-22

The DHT22 is a temperature and humidity sensor module. It offers a temperature measurement range of -40°C to 80°C with a typical accuracy of ±0.5°C and a humidity measurement range of 0% to 100% with a typical accuracy of ±2-5%. It communicates using a proprietary 1-wire protocol, which can be easily implemented with a microcontroller [14]. DHT-22 can be seen in Figure 7.

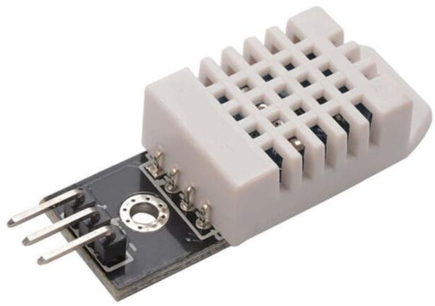

Figure 7. DHT-22 [7]

4.1.5 MQ-135

The MQ-135 is a gas sensor module widely used for detecting various harmful gases in the air. The sensor operates based on the principle of the chemical reaction between the target gas and the sensing material on the sensor's surface [7]. It provides an analog output voltage that can be read and interpreted by a microcontroller or ADC.

MQ-135 can be seen in Figure 8.

Figure 8. MQ-135 [7]

4.2 HARDWARE

In our project, one Transmitter Arduino UNO R3 and one Receiver ESP32 Wemos D1 R32 are used [7]. We also use a Lo-Ra Module to send the data we receive to the receiving Arduino.

The agent microcontroller collects data from sensors. It sends this data by LoRa communication to the Host microcontroller. The data is confirmed by the security process. If the data is valid, it is uploaded to the server.

The other side of the flow works like a fuse. If the PPM level exceeds the maximum value of the optimal PPM level (which is 1500 PPM), the fan starts blowing. It can be seen in Flowchart Figure 9.

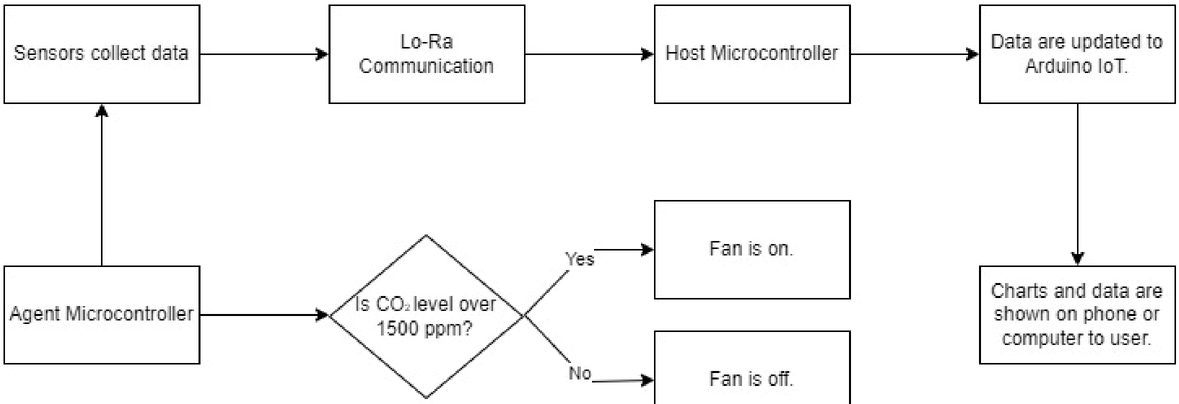

Figure 9. Project Flow Chart

4.2.1 HOST

The host part consists of 1x ESP32 Wemos D1 R32 and 1x SX1278 Lo-Ra module. Lora module's Reset, MISO, MOSI, SLCK, and NSS outputs are connected to RST, IO19, IO23, IO18, and IO05 pins of ESP32 Wemos D1 R32, respectively. Feeding and grounding of the Lora module is also provided via ESP32 Wemos D1 R32.

4.2.2 AGENT

The agent part consists of 1x Arduino Uno R3, 1x SX1278 Lo-Ra module, 1x MQ-135, 1x DHT-22, and 1x 5V Dc Fan. Lora module's Reset, MISO, MOSI, SLCK, and NSS outputs are connected to RESET, 12, 11, 13, and 10 pins of Arduino Uno R3, respectively. The supply of Lora module is also provided via Arduino Uno R3. DHT-22 and MQ-135 are fed from additional power units. They are connected to the D2 and A1 pins of the Arduino. The 5V fan is powered by the Arduino Uno R3 and is controlled via an NPN transistor.

4.3 SOFTWARE

The analog data received from the MQ-135 sensor is processed in the equation resulting from the sensor tests. Humidity & Temperature sensor data is received. These 3 data are encrypted with a 4-digit pin and sent from the host microcontroller to the agent microcontroller via Lo-Ra. The Agent microcontroller checks the passwords of the incoming data, and if the passwords match, it decrypts the data and sends it to the Arduino IoT Cloud. The most important thing is to prevent the data from mixing with each other by encrypting each data especially and also to prevent parasite infiltration. Software flow chart can be seen in Figure 10. Agent and host microcontrollers codes can be seen in appendix B.

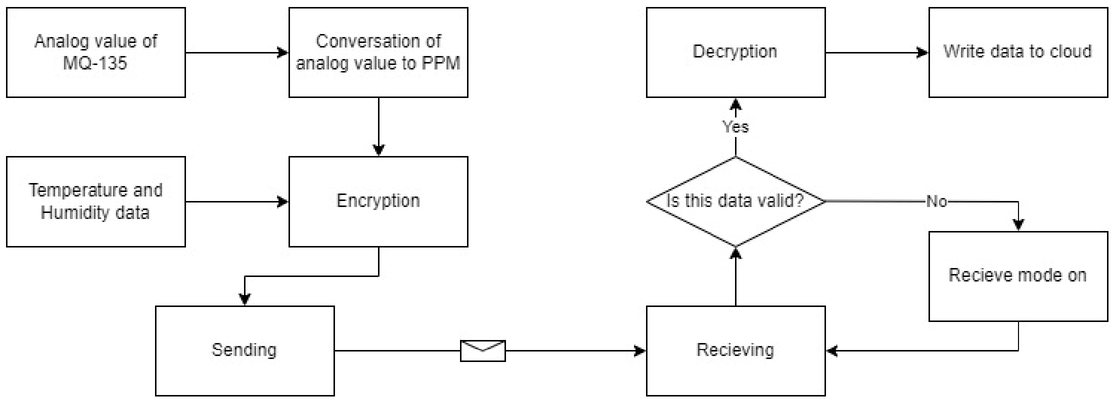

Figure 10. Software Flow Chart

4.4 USER INTERFACE

Arduino IoT Cloud was preferred for User Interface. The main reasons for this preference are that it can be watched instantly both on the website and through the application, as well as working in harmony with the Wemos D1 R32. In the first column, carbon dioxide graph and instant carbon dioxide data are seen. The second column shows the humidity graph and instant humidity data. In the third column, the temperature graph and instantaneous temperature data are displayed. At the bottom, there is an ON/OFF light showing whether the fan is working or not, it can be seen as Figure 11. The data is sent by being encrypted with 4 separate packages at 1 second intervals, and the incoming data is updated every 20 seconds in the IoT Cloud interface and transferred to the graph.

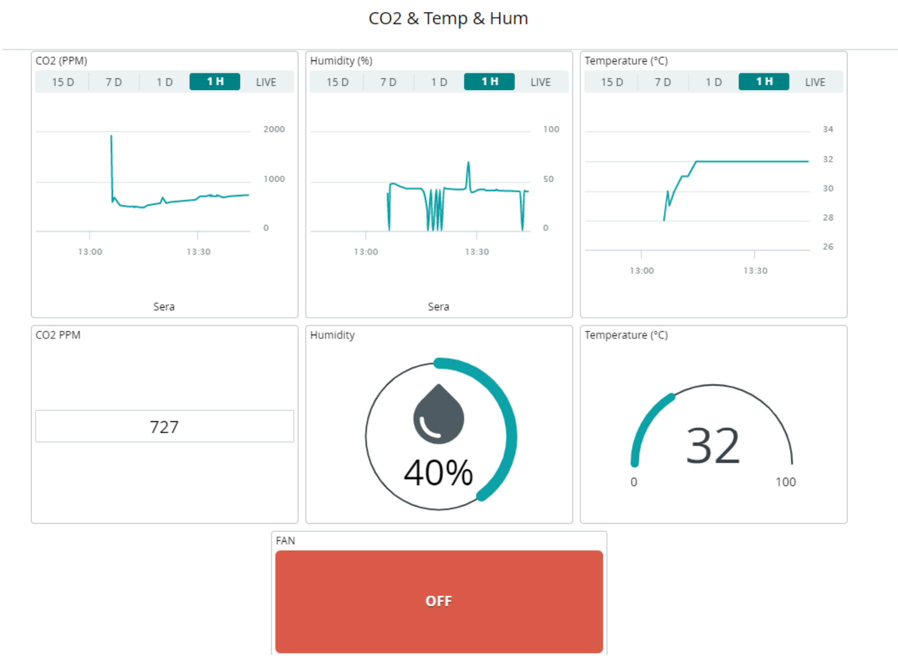

Figure 11. User Interface Web Screen

5. IMPLEMENTATION AND TESTING

5.1. Implementation

We implemented a test circuit to measure the response similarities and differences between the electrochemical sensors and the infrared sensor. In this test circuit, while the electrochemical sensors are connected to the analog output of the Arduino, the infrared sensor is connected to the PWM output. In this way, analog outputs from electrochemical sensors and PWM outputs from infrared sensors were compared.

A test circuit was applied to measure the response similarities and differences between the electrochemical sensors and the infrared sensor. In this test circuit, while the electrochemical sensors are connected to the analog output of Arduino, the infrared sensor is connected to the PWM output. In this way, analog outputs from electrochemical sensors and PWM outputs from infrared sensors are compared.

As a result of the data obtained as a result of the test circuit (explained in detail in the test part), it was given to use only the MQ-135 sensor for CO_2 measurement in our project. In addition, temperature & humidity sensor and fan were added to the project.

5.1.2. Testing

The test part consists of 2 parts. These parts are divided into two as sensor testing and system testing.

5.1.2.1 Sensor Testing

Some tests were carried out to test the stability of the outputs of the MH-Z14a infrared sensor. UART, PWM, Analog outputs of the infrared sensor were tested in the same environment. As a result of the test, UART and Analog output Peaks and irregular, instantaneous drops were seen. PWM output, a more stabler graphic appeared compared to the others. As shown in Figure 12, the green line represents PWM, the red line represents UART, and the blue line represents Analog Output. For this reason, we preferred to use PWM output during the testing process.

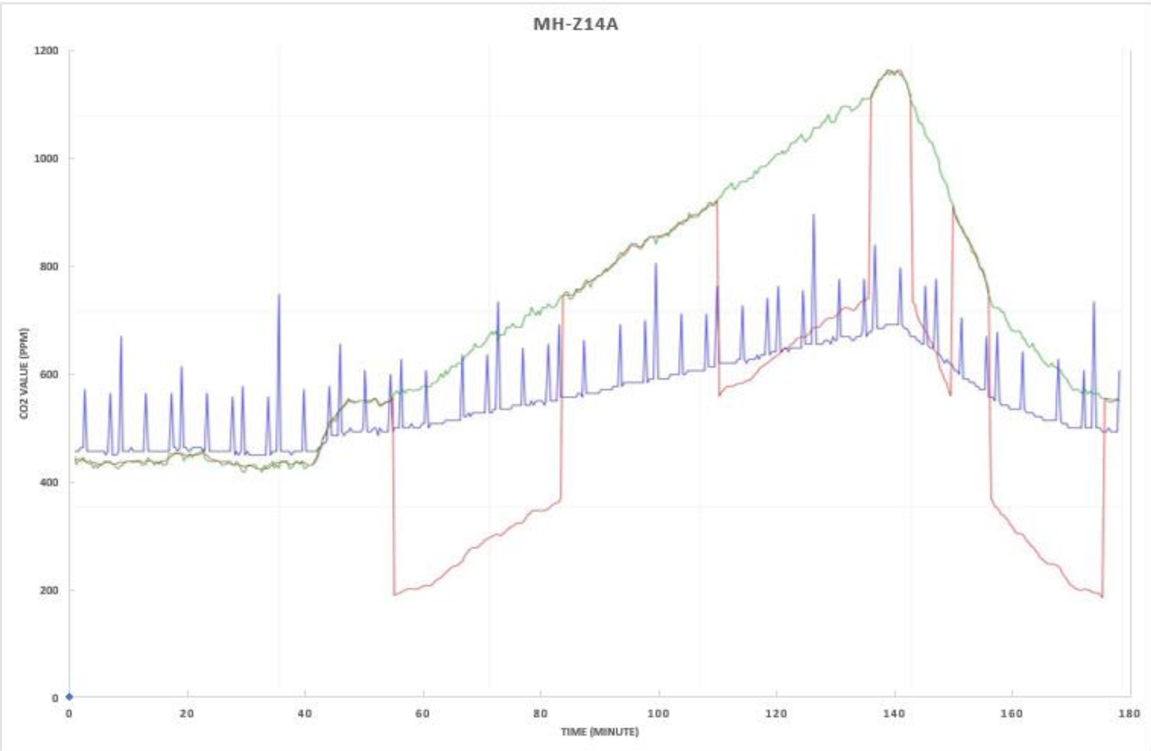

Figure 12. Outputs of MH-Z14a Infrared Sensor

The second test scenario was realized with the implemented test circuit. There are three electro-chemical sensors (MQ-135, MQ-4, MQ-7) and Lo-Ra Module connected to the Transmitter Arduino [7]. One of our Infrared CO_2 sensors was connected to our Transmitter Arduino. The data received from the Infrared Sensor was used to control the results of the data received from the electro-chemical sensor, as well as to monitor their response to CO_2 increases and decreases. When our project is finalized, the infrared sensor sender will be removed from the Arduino. The sensors were left inside a room with the doors and windows closed. Measurements of the room, in which 2 people were present, were made at 5 minutes intervals for 5 hours. After 3 hours, it was observed that the CO_2 output increased regularly. At the end of the 3rd hour, when the window was left open for a sufficient time, it was observed that the CO_2 output data returned to the initial state of the measurement and the window was closed again. During the remaining time, the sensors gave the same response as the 3 hours at the beginning, and the amount of CO_2 increased steadily. The graph of data obtained as a result of the tests is seen in Figure 13. Analog outputs obtained from electrochemical sensors are not as smooth graphic as an infrared sensor graphic (explained in detail in the Testing section). An attempt was made to decipher the data changes and find the connections between them through this graph. Electrochemical and infrared sensors' reactions can also be seen in Figure 13.

As expected, analog outputs were obtained from electrochemical sensors, so we could not get a stable graphic as an infrared sensor PWM outputs.

Figure 13, in which MQ-7 and MQ-4 sensors do not react directly to carbon dioxide gas seems to be. For this reason, we only graphed the outputs of MQ-135 and MH-Z14a sensors in Figure 14.

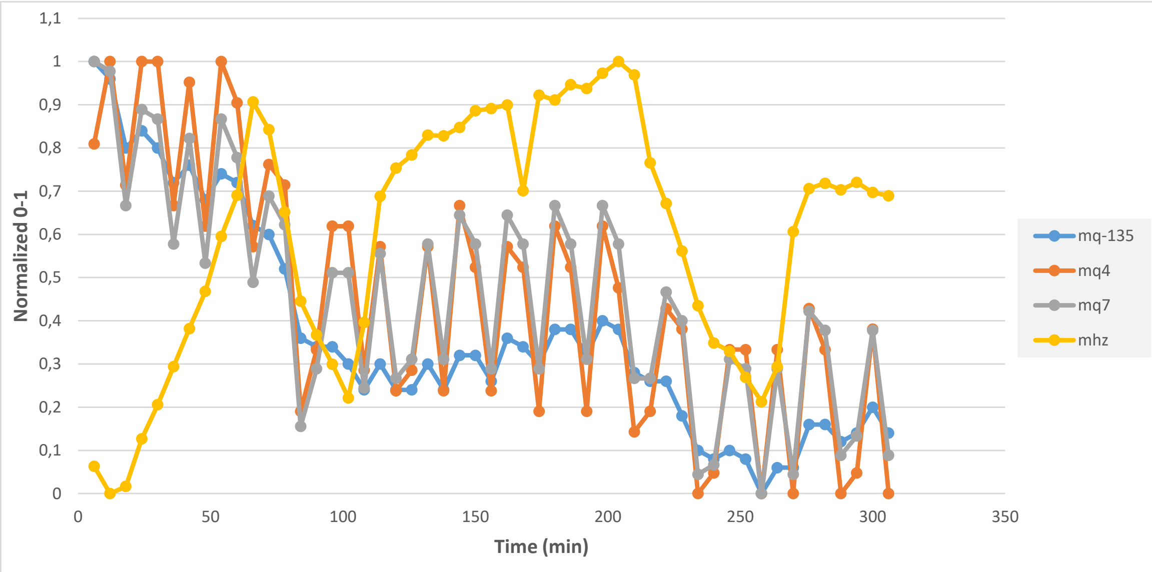

Figure 13. Sensor results of the 5-hours measurement test

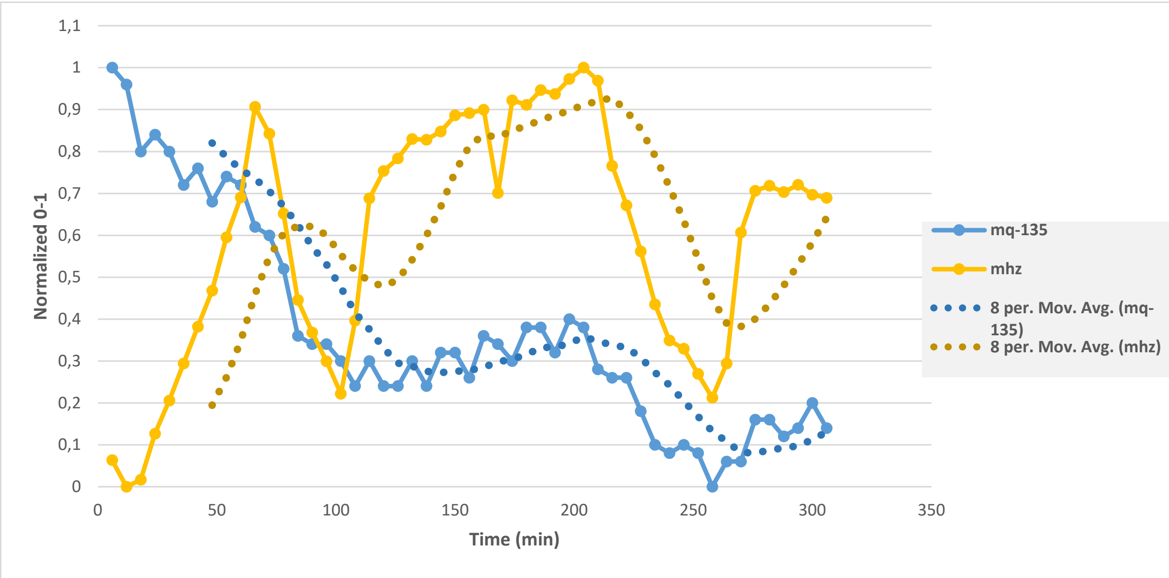

Figure 14. Comparison MHZ-14a and MQ-135 sensors with trendlines

We have taken into account that some peaks may have been experienced because the measurement times are in short intervals such as 3 minutes and especially because the MQ-135 sensor is an analog output. For this reason, we have drawn 8 per Moving Average Trendline. In this way, small peaks, sudden rises, and falls in the outputs are reduced and we have obtained a more fluent line. When Trendlines were examined, the different responses of the sensors during the calibration process during the first hour of measurement were not taken into account. It has been observed that the MQ-135 sensor can respond in the same direction for about 4 hours after the 1-hour calibration process, although it is not as sensitive as MHZ-14a. This is preliminary evidence that targeted sensor calibration can be achieved.

The MHZ-14a and MQ-135 sensors were calibrated in the same environment for 1 day, and the data were collected again for 4 days. This calibration method is intended to keep the sensors in a well-ventilated environment for 24 hours, to match the natural PPM value in the air which is 400 PPM. Based on the collected data, the MHZ-14a sensor was used to create an equation using the Curve Fitting Tool. And finally, 3 different equations were obtained.

The coefficient of determinations of the resulting equations were examined, as you can see in Figure 15. We chose the second order polynomial equations (red one in graph) which is *equation (1)* because it has the highest coefficient of determination.

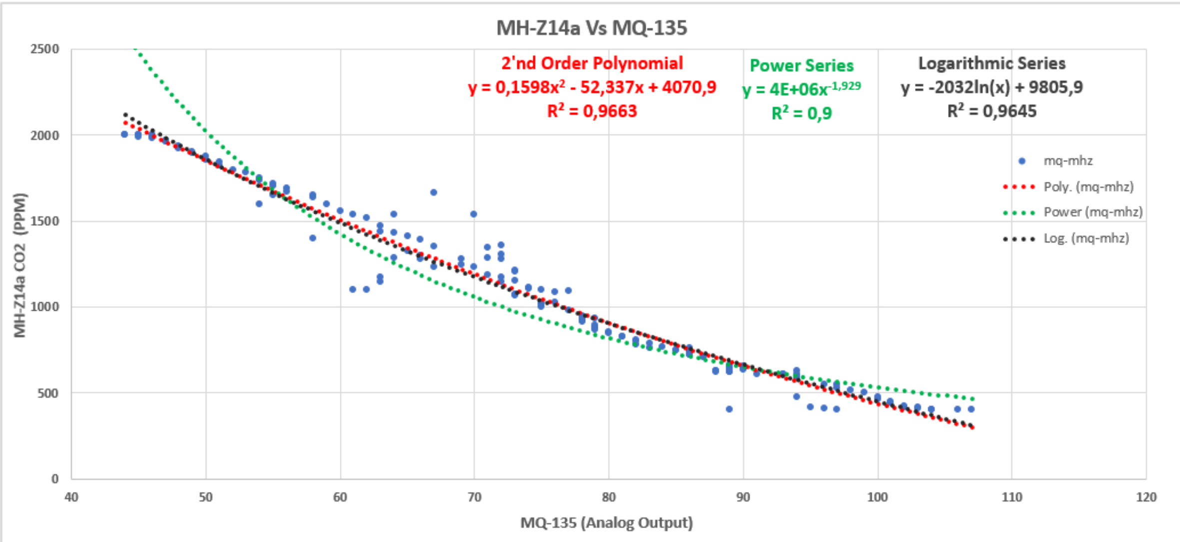

Figure 15. Sensor Fitting Chart

$$y = 0,1598x^2 - 52,337x \quad R^2 = 0,9663 \tag{1}$$

$$y = 4E + 06x^{-1,929} \quad R^2 = 0,9 \tag{2}$$

$$y = -2032ln(x) + 9805,9 \quad R^2 = 0,9645 \tag{3}$$

5.2.3. Cost Analysis

A number of components were taken for the test process and will not be used in the product, but the products that are planned to be used are 1x Arduino Uno R3, 1x ESP32 Wemos D1 R32, 2x Lo-Ra Module, 5V Dc Fan, MQ-135, 1x Humidity & Temperature sensor and 1x TIP31C BJT Transistor. When the unit cost and total cost of the purchased products are examined, it can be said that the product is cheap, detailed prices can be seen in Table 3.

Table 3. Cost Analysis

Component	Count	Unit Price
Arduino UNO R3	1	₺ 605
ESP32 Wemos D1 R32	1	₺ 143
Lo-Ra Module	2	₺ 180
MQ-135 (Air Quality Module)	1	₺ 45
DC216 5V Dc Fan	1	₺ 75
DHT-22 Temperature & Humidity Sensor	1	₺ 45
TIP31C Transistör BJT TO220 - NPN	1	₺ 4,8
TOTAL	**8**	**₺1.277,84**

6. RESULTS

The project was successfully completed, the project's demos can be seen in Figure 16. The sensors are located near the top of the greenhouse, while the fan system is attached to the ventilation section above the greenhouse. When the fan is turned on, oxygen circulation will increase from outside to inside.

Figure 16. Greenhouse Demonstration

The User Interface was successfully designed completed, it can be seen in Figure 17. The sensor seen on the left in the picture is the DHT-22, the one on the right is the MQ-135 sensor, with the fan at the top. In the first column, the graphs of CO_2, humidity and temperature can be seen. Instant data and all data up to 15 days ago can be seen through this chart. In the second column, instantaneous data of CO_2, instantaneous data of humidity and instantaneous data of temperature can be seen. At the bottom, whether the fan is on or not is indicated by the ON/OFF led. Data can be easily monitored from the Arduino IoT Cloud website or with the Remote IoT Cloud app to download.

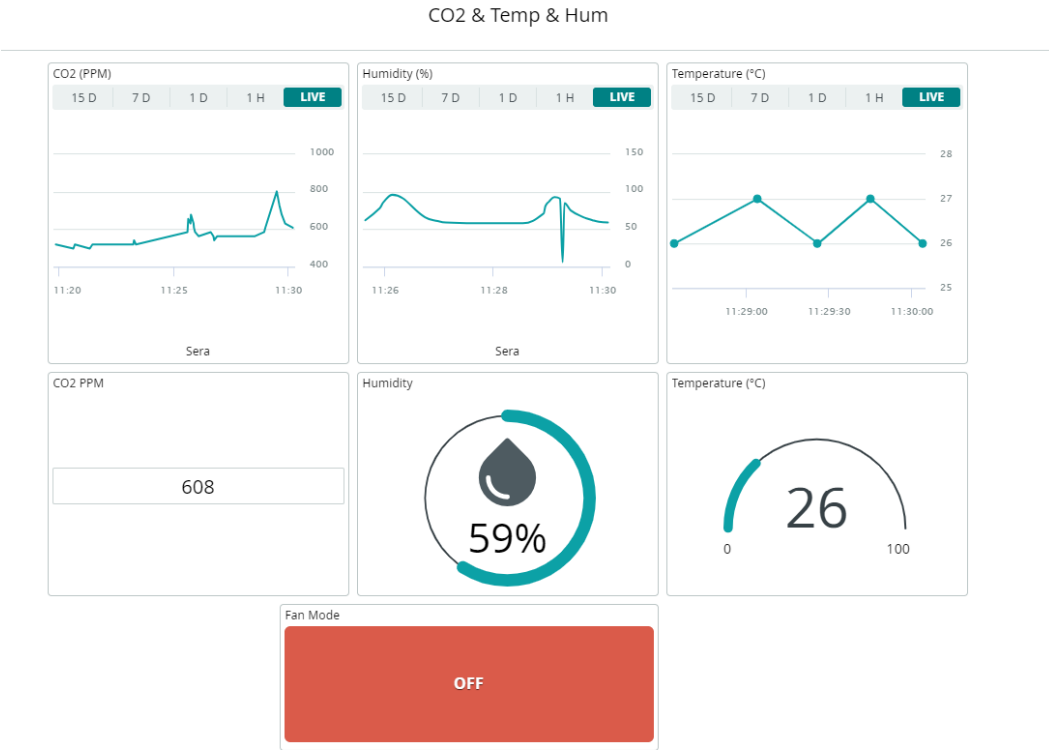

Figure 17. User Interface

The MQ-135 sensor was adapted to the CO_2 change via equation. The data update time targeted in the project was reduced from 15 minutes to 2 minutes. In addition, humidity and temperature sensors were added to the project. Data sent with LoRa is encrypted. In this way, data security was ensured. A Fan was added to keep the CO_2 level at the optimum. With these add-ons, the project has exceeded expectations and at the same time has shown that it can develop further.

It cost about 50% (fifty percent) less than similar products, products can be seen in Figure 18. The similarities between our project and the products are that the CO_2 level can be monitored. The advantages of our product are that temperature and humidity levels can be monitored along with the CO_2 level, it is cheaper and can be monitored remotely via APP/Web. The disadvantages of our product are that it does not have an LCD screen and is not portable. The comparison chart can be seen in Table 4.

Figure 18. CO_2 Monitoring Products

Table 4. Product Comparison

Products	Cost	Monitoring	Mobil/Plugged
The Greenhouse System	1277	APP and Web Site	Plugged in
Product 1*	2448	LCD Screen	Plugged in
Product 2**	4400	LCD Screen	Portable

*Lixada Air Quality Monitor CO_2 and Smoke Detector [20]

**Lixada Rechargeable Battery Carbon Dioxide Detector Portable [21]

7. CONCLUSION

7.1. Learning Strategies

Initially, an extensive literature review was conducted on the impacts of CO_2 on plants, greenhouses, gas sensors, communication technologies, and android applications, utilizing resources available at the MEF Library. Subsequently, the gathered information was synthesized, and conclusions were drawn regarding our product. Multiple project drafts were then generated, and a comprehensive analysis was conducted, comparing them with existing products in the market. Through this rigorous evaluation process, the most optimal draft was identified and refined by aligning it with the materials intended to be utilized.

7.2. Professional and Ethical Responsibilities of Engineers

By providing an affordable solution for greenhouse owners, the opportunity to monitor their greenhouses will be extended to a wider range of users. Utilizing this opportunity effectively will lead to an increase in productivity within greenhouses, even those operated by non-professionals. Consequently, a greater abundance of vegetables and fruits will be available in the market. This project has the potential to contribute positively to addressing potential food scarcity resulting from population growth.

In addition, it is very important to carefully select the communication systems to be used in the coming period in order to preserve the naturalness of the products produced and to be affected by radiation very little.

7.3. Further Discussion

It may be possible to keep the amount of carbon dioxide in the greenhouse at a certain level without human touch by adding a control system to this system with the development of artificial intelligence in the future. To achieve this, sophisticated autonomous systems can be implemented to regulate and maintain optimal levels of carbon dioxide and oxygen within the greenhouse. It can be improved by adding a CO_2 increase system such as CO_2 tubes.

Our project can be specifically designed and applied in soilless agriculture. Given the importance of climatic conditions in hydroponic farming, and in particular increasing plant stress using fans, our solution has the potential to significantly increase crop yields.

Communication can be secured by means of standard crypto algorithms, data can be transferred more securely, and a two-way security system can be established. Cost can be reduced by employing an 8-bit microprocessor provided that Wi-Fi.

APPENDIX A

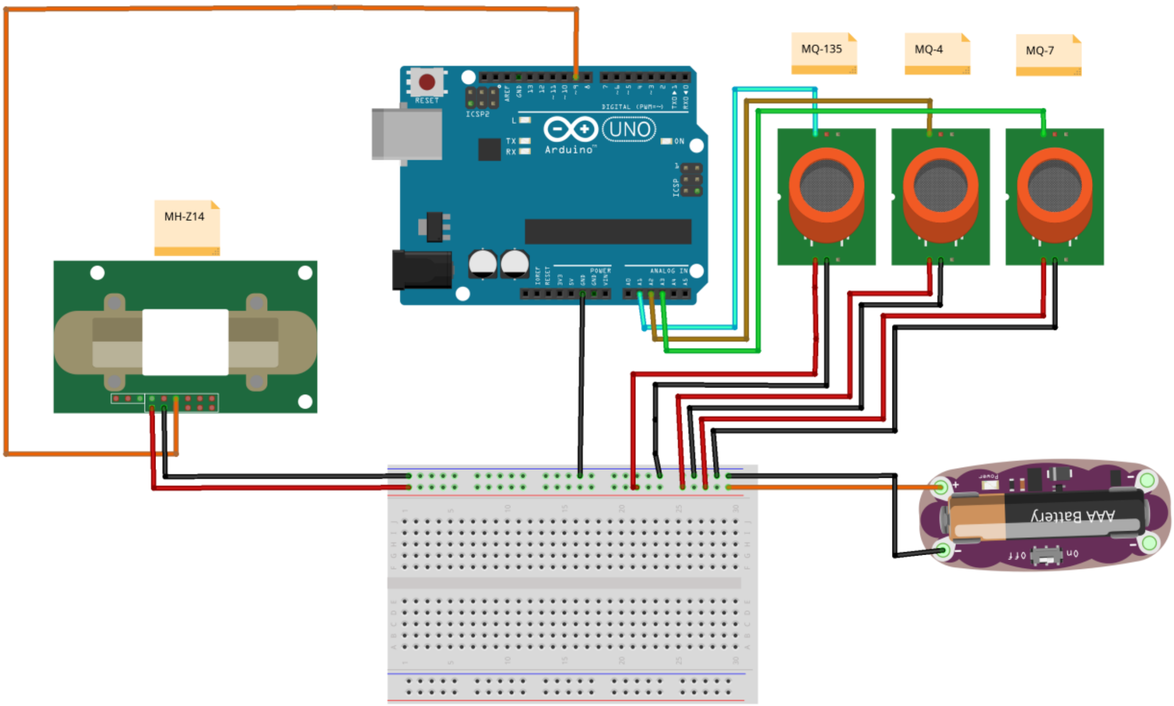

Figure 19. Testing Circuit

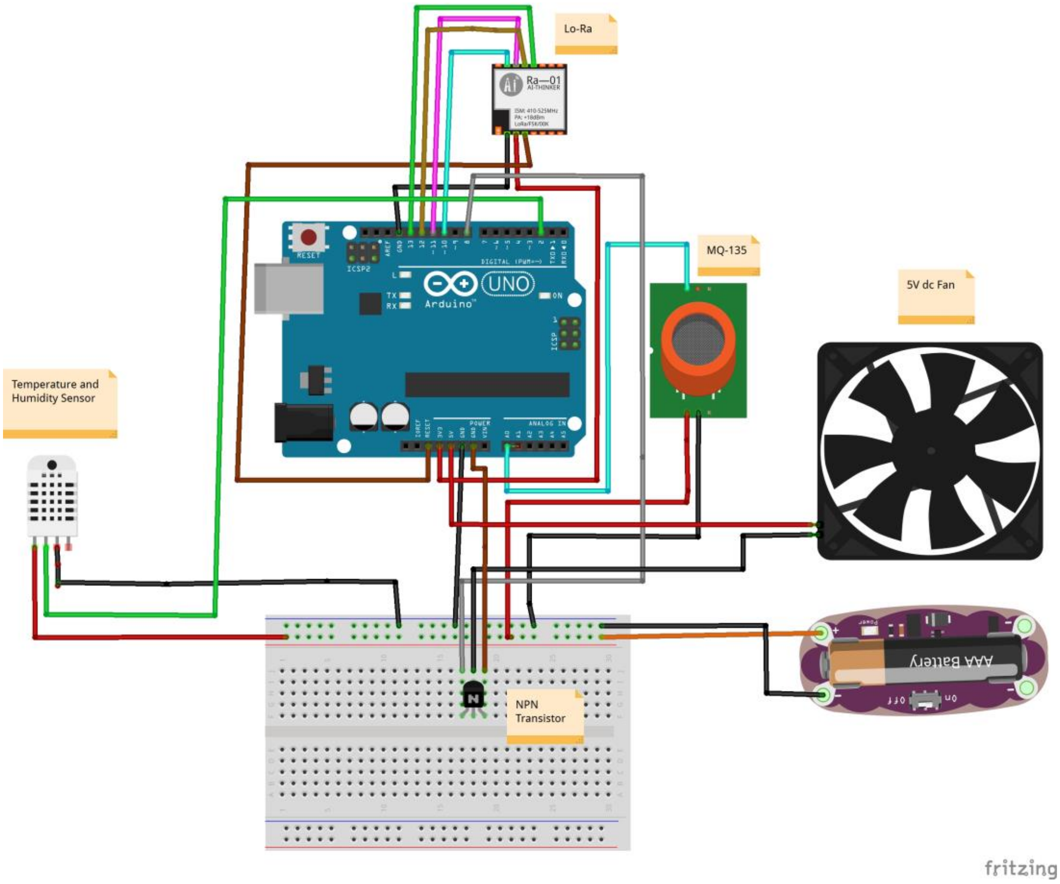

Figure 20. Agent Circuit

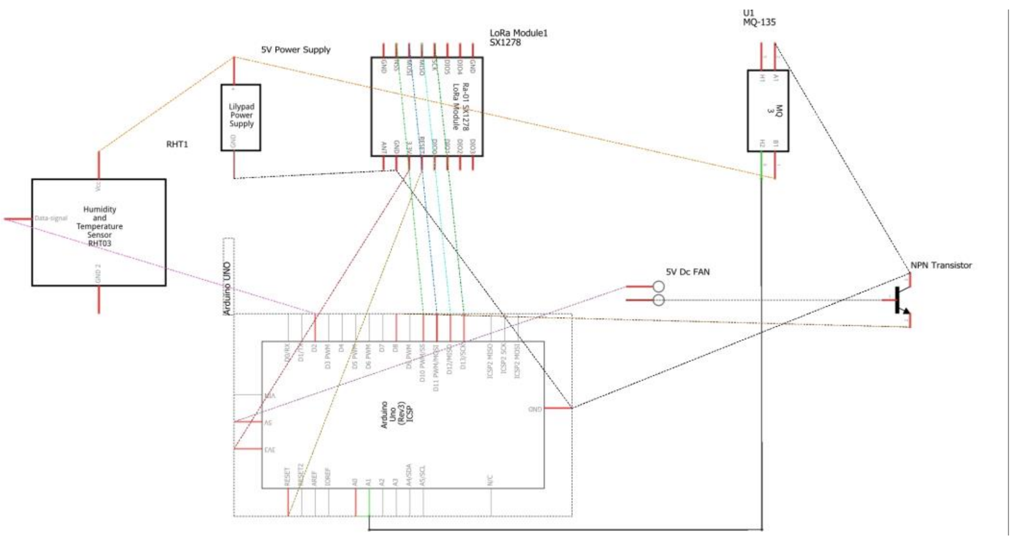

Figure 21. Agent Circuit Schematic

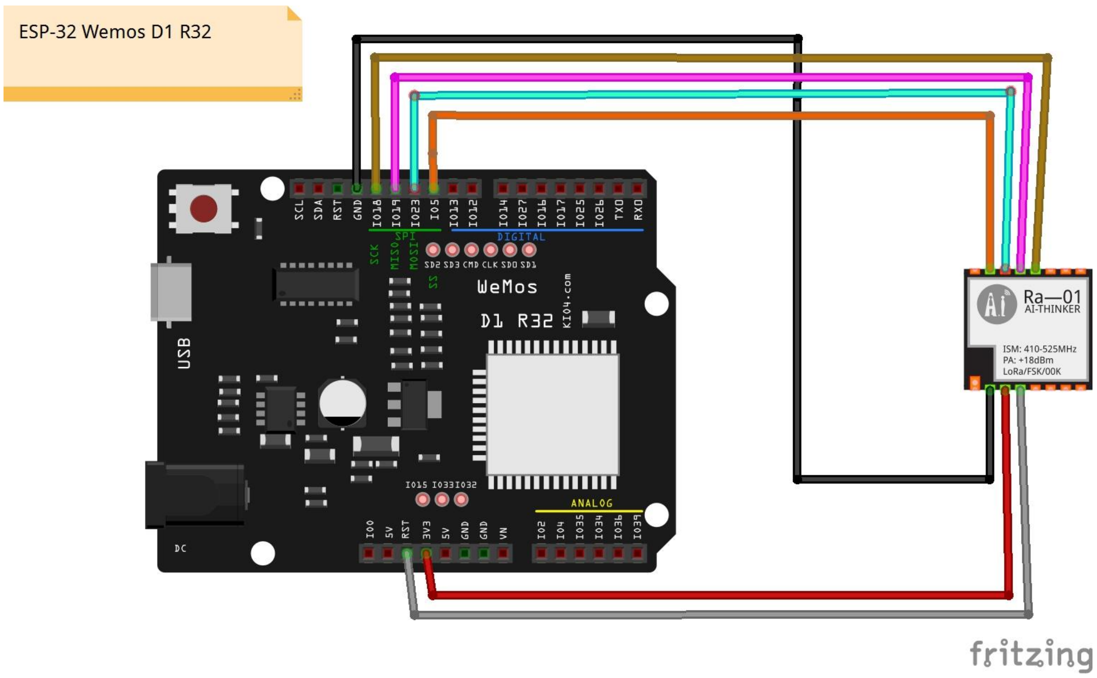

Figure 22. Host Circuit

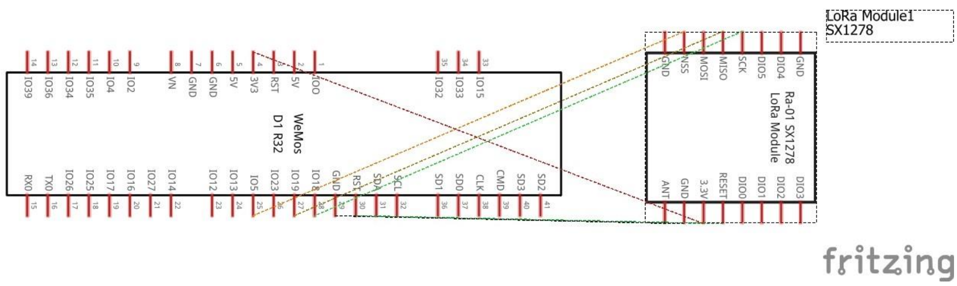

Figure 23. Host Circuit Schematic

APPENDIX B

Arduino UNO R3 Code (Agent Microcontroller):

```
/*
Define Pins Lo-Ra
    Vcc          ->   3.3V
    MISO         ->   D12
    MOSI         ->   D11
    SLCK         ->   D13
    Nss          ->   D10
    GND          ->   GND
    Reset        ->   6
    HMD Sensor   ->   4
 */
#include <SPI.h>
#include <LoRa.h>
#include <dht.h>
#define pin 4 //DHT Pin
dht DHT;
//Passwords for each sensor data
int passwordCo2 = 5221;
int passwordTemp = 5223;
int passwordHum = 5225;



void setup() {
  pinMode(8, OUTPUT); // Fan Pin
  Serial.begin(9600);

  //LoRa initializing
  while (!Serial)
    ;
  Serial.println("LoRa Sender");

  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed!");
    while (1)
      ;
  }
}

void loop() {

  // reading MQ-135
  int c = analogRead(A0);
```

```
//MQ-135's analog data converted to PPM level of CO2
int y = ((0.1598 * c * c) - (52.337 * c) + 4071);  //MQ-135 Sensor

//reading humidity and temperature data
int readData = DHT.read22(pin);

int isi = DHT.temperature;

int nem = DHT.humidity;

Serial.println(c);  //Controlling A0
Serial.print("CO2 PPM:"); //Controlling Values

Serial.println(y);

Serial.print(isi);

Serial.print(" C");

Serial.print("\t");

Serial.print("%");

Serial.println(nem);

// Starting to send sensor data by LoRa packages
LoRa.beginPacket();

LoRa.print("");
LoRa.endPacket(); //Empty packet is sent, lora memory is cleared.
delay(200);

LoRa.beginPacket();
LoRa.print(passwordCo2); //Sending CO2 password
LoRa.print(y);          // Sending CO2
LoRa.endPacket();
delay(200);

// LoRa.print(",");
LoRa.beginPacket();
LoRa.print(passwordTemp); //Sending Temp password
LoRa.print(isi);        // Sending temp
LoRa.endPacket();
delay(200);
// LoRa.print(",");

LoRa.beginPacket();
```

```
  LoRa.print(passwordHum);  //Sending Humidity password
  LoRa.print(nem);          //Sending Humidity
  LoRa.endPacket();

  delay(200);

  if (y > 1500) {          // Controlling CO2 value for opening fan
    digitalWrite(8, HIGH);
  }

  else {
    digitalWrite(8, LOW);
  }
  delay(100);
}
```

ESP32 Wemos D1 R32 Code (Host Microcontroller):

```
// Arduino Uno WiFi Dev Ed Library - Version: Latest
//#include <UnoWiFiDevEd.h>

/*
  Sketch generated by the Arduino IoT Cloud Thing "Untitled"
  https://create.arduino.cc/cloud/things/16b07abb-8466-4376-bb14-
110c0b0e33d0

  Arduino IoT Cloud Variables description

  The following variables are automatically generated and updated when
changes are made to the Thing

  int cO2;
  int humidity;
  int temperature;
  bool fan;

  Variables which are marked as READ/WRITE in the Cloud Thing will also have
functions
  which are called when their values are changed from the Dashboard.
  These functions are generated with the Thing and added at the end of this
sketch.
*/

#include "thingProperties.h"
#include "WiFi.h"
#include "LoRa.h"
#include "SPI.h"

#define ss 5 // NSS Pin
#define rst 14 // RESET Pin
#define dio0 16 //DIO0 Pin
//Passwords for each sensor data
int passwordCo2 = 5221;
int passwordTemp = 5223;
int passwordHum = 5225;
int fancheck = 0;

void setup() {
  // Initialize serial and wait for port to open:
  Serial.begin(115200);
  // This delay gives the chance to wait for a Serial Monitor without
blocking if none is found
```

```
  delay(1500);

  //WiFi initializing
  initWiFi();
  // Connect to Arduino IoT Cloud
  initProperties();
  ArduinoCloud.begin(ArduinoIoTPreferredConnection);
  setDebugMessageLevel(2);
  ArduinoCloud.printDebugInfo();

  while (!Serial);
  Serial.println("LoRa Receiver");

  //setup LoRa transceiver module
  LoRa.setPins(ss, rst, dio0);

  //LoRa initializing
  while (!LoRa.begin(433E6)) {
    Serial.println(".");
    delay(500);
  }
  Serial.println("LoRa Initializing OK!");


}

int a = 1;
void loop() {
  //Arduino IoT Cloud Updater
  ArduinoCloud.update();

  // try to parse packet
  int packetSize = LoRa.parsePacket();

  if (packetSize) {
    // received a packet
    Serial.print("Received packet '");

    // read packet
    while (LoRa.available()) {

      String LoRaData = LoRa.readString();
      Serial.print(LoRaData);

      //The password stored in first four bytes
      //So the incoming data are splitted and controlled by the system
      String ctrlPassword =  LoRaData.substring(0,4);
```

```
    int intPassw = ctrlPassword.toInt();
    //Each sensor data have their own password. So, this section is
deciding
    //if the data valid by encrypting it.
    //If the data is valid, they are stored in variables.
     if (intPassw == passwordCo2){
      int data=LoRaData.toInt();
      if (data > 99999999){
        data = data - (passwordCo2*100000);
        String a = String(data);
        // data is encyripted by using substring method
        String co2data = a.substring(0,4);
        String fandata = a.substring(4,6);
        cO2 = co2data.toInt();
        fancheck = fandata.toInt();
      }
      else{
        data = data - (passwordCo2*10000);
        String a = String(data);
        String co2data = a.substring(0,3);
        String fandata = a.substring(3,5);
        cO2 = co2data.toInt();
        fancheck = fandata.toInt();
      }
      if (fancheck == 1){
        fan = true;
      }
      else if (fancheck == 0){
        fan = false;
      }
      Serial.println("Co2:");
      Serial.print(cO2);
      Serial.println("fandata:");
      Serial.print(fancheck);
    }
    else if (intPassw == passwordTemp){
      int data=LoRaData.toInt();
      data = data - (passwordTemp*100);
      temperature=data;
      Serial.println("temp:");
      Serial.print(temperature);
    }
    else if (intPassw == passwordHum){
      int data=LoRaData.toInt();
      data = data - (passwordHum*100);
      humidity =data;
      Serial.println("humidity");
```

```
        Serial.print(humidity);
      }
      //If the incoming data can not pass the password control, data won't
be proccesed
      else {
        Serial.println("This message is not for me!");
      }
     delay(200);
    }

    // print RSSI of packet
    Serial.print("' with RSSI ");
    Serial.println(LoRa.packetRssi());
    delay(1000);
  }




}



/*
This is the WiFi initialize method
*/


void initWiFi() {
  WiFi.mode(WIFI_STA);
  WiFi.begin("GokS22", "gok12345");
  Serial.print("Connecting to WiFi ..");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print("Wifi_Not Connected!");
    delay(1000);
  }
  Serial.println(WiFi.localIP());
}




/*
This method section writed by Arduino IoT Cloud automatically
*/
```

```
void onCO2Change()  {
}

void onTemperatureChange()  {
}

void onHumidityChange()  {
}

void onFanChange()  {
}
```

ACKNOWLEDGEMENTS

This work is supported by TÜBİTAK 2209-A University Students Research Projects Support Program 2023.

We are grateful to our dearest instructor and advisor Tuba AYHAN for support.

REFERENCES

[1] «Unsal, Emre & Yelkuvan, Ahmet Fırat & Taşbaşı, M & Sönmez, Ö. (2018). A New Greenhouse Monitoring System Based on Internet of Things Technology.».

[2] «https://www.alibaba.com/product-detail/BECC-CO2-Temp-Humidity-Controller_1600580339235.html?spm=a2700.7724857.0.0.7f932c5ahwZuiK&s=,» [Çevrimiçi].

[3] «https://www.alibaba.com/product-detail/Plant-Grow-Tent-Climate-Mushroom-Room_1600656805759.html?spm=a2700.7724857.0.0.7f932c5ahwZuiK,» [Çevrimiçi].

[4] «Hanggoro, Aji & Putra, Mahesa & Reynaldo, Rizki & Sari, Riri. (2013). Green house monitoring and controlling using Android mobile application. 79-85. 10.1109/QiR.2013.6632541».

[5] «Tezcan, Ahmet & Atilgan, Atilgan & Öz, Hasan. (2011). Seralarda Karbondioksit Düzeyi, Karbondioksit Gübrelemesi ve Olası Etkileri. Süleyman Demirel Üniversitesi Ziraat Fakültesi Dergisi. 6. 44-51.».

[6] «https://www.karfandscoot.com/elektrokimyasal-gaz-sensoru-nedir-nasil-calisir/,» [Çevrimiçi].

[7] «https://www.direnc.net/,» [Çevrimiçi].

[8] «Qiao Xiaohui, Du Shangfeng, He Yaofeng, Liang Meihui,(2018). Development and design of mobile terminal APP for greenhouse environment control. https://doi.org/10.1016/j.ifacol.2018.08.093.».

[9] «Bersani, Chiara & Ruggiero, Carmelina & Sacile, Roberto & Soussi, Abdellatif & Zero, Enrico. (2022). Internet of Things Approaches for Monitoring and Control of Smart Greenhouses in Industry 4.0. Energies. 15. 3834. 10.3390/en15103834.».

[10] Arduino, 2023. [Çevrimiçi]. Available: https://cloud.arduino.cc/.

[11] «SX1278,» [Çevrimiçi]. Available: https://www.semtech.com/products/wireless-rf/lora-connect/sx1278.

[12] «I2C vs SPI vs UART – Introduction and Comparison of their Similarities and Differences,» [Çevrimiçi]. Available: https://www.totalphase.com/blog/2021/12/i2c-vs-spi-vs-uart-introduction-and-comparison-similarities-differences/.

[13] «ESP32 Wemos D1 R32,» [Çevrimiçi]. Available: https://www.halloweenfreak.de/arduino/pdfs/D1_R32_ENG.pdf.

[14] «DHT-22,» [Çevrimiçi]. Available: https://cdn-learn.adafruit.com/downloads/pdf/dht.pdf.

[15] R. Howe, "ISO 14001: the green standard," in Computer, vol. 30, no. 11, pp. 133-134, Nov. 1997, doi: 10.1109/2.634868.

[16] K. Hieronymi, "Implementing the WEEE Directive," Proceedings of the 2001 IEEE International Symposium on Electronics and the Environment. 2001 IEEE ISEE (Cat. No.01CH37190), Denver, CO, USA, 2001, pp. 217-222, doi: 10.1109/ISEE.2001.924529.

[17] J. C. Bullock, "Environmentally sound management of electronic scrap and the Basel Convention on control of transboundary movements of hazardous wastes and their disposal," Proceedings of the 1995 IEEE International Symposium on Electronics and the Environment ISEE (Cat. No.95CH35718), Orlando, FL, USA, 1995, pp. 192-197, doi: 10.1109/ISEE.1995.514974.

[18] "IEEE Standard for Environmental Assessment of Imaging Equipment-Amendment 1," in IEEE Std 1680.2a-2017 (Amendment to IEEE Std 1680.2-2012) , vol., no., pp.1-19, 22 Dec. 2017, doi: 10.1109/IEEESTD.2017.8237223.

[19] Qiao Xiaohui, Du Shangfeng, He Yaofeng, Liang Meihui,(2018). Development and design of mobile terminal APP for greenhouse environment control. https://doi.org/10.1016/j.ifacol.2018.08.093.

[20] https://www.hepsiburada.com/lixada-hava-kalitesi-monitoru-co2-ve-duman-dedektoru-lcd-ekranli-yurt-disindan-pm-HBC00000LRZ41. [Online]

[21] https://www.hepsiburada.com/lixada-sarj-edilebilir-pilli-karbon-dioksit-dedektoru-tasinabilir-siradan-surum-yurt-disindan-pm-HBC00000LRZ2L [Online]