



People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research
University of Echahid Hamma Lakhdar – El Oued
Faculty of Exact Sciences
Department of Informatics

MASTER DISSERTATION

Submitted in Partial Fulfillment of the Requirements
for the Academic Master's Degree

Field: IoT & Cyber Security

Secure IoT Drone-Based Weather Monitoring and Prediction System

Publicly Defended on: 07/06/ 2026

Board of Examiners

- ◆ Chairman: Dr.Mohammed Mounir Bouhamed.
- ◆ Examiner: Dr.Mostefa Kara .
- ◆ Supervisor: Dr.Abelkader Hima .

Prepared by

Hanine Benazzouz
Manar Bettayeb

Academic Year: 2025 / 2026

Acknowledgments

First and foremost, we would like to thank Almighty Allah for granting us the strength and determination to successfully complete this graduation project.

We would also like to express our sincere gratitude to our supervisor, Dr. Hima Abdelkader, for his valuable advice, continuous encouragement, guidance, and support throughout the realization of this work.

We extend our sincere thanks to the members of the jury for the honor they have given us by accepting to evaluate this work.

We are deeply grateful to our dear parents for their constant support and encouragement, and for being one of the main reasons that motivate us to persevere and overcome every difficulty.

We also thank all the professors who contributed to our education, as well as the staff members of our department.

Finally, we would like to thank everyone who helped us, directly or indirectly, in accomplishing this work.

بسم الله الرحمن الرحيم

«وَقُلْ اَعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ»

إلى من بلغ الرسالة وأدى الأمانة، إلى نبي الرحمة سيدنا محمد صلى الله عليه وسلم.

أهدي هذا العمل لنفسي التي صبرت وتحملت وتعبت وسهرت الليالي، إلى نفسي التي لم تستسلم رغم الصعوبات والتي واصلت الطريق رغم الضغط. شكراً لأنك كنت قوية لأنك لم تتوقفي، فعلاً آمنت أنك قادرة، وهذا النجاح هو ثمرة صبرك وجهدك.

إلى أمي الغالية

نبض قلبي وسر سعادتي، يا من حملتني وهنّاً على وهن وكانت دعواتها تسبق خطواتي وترافقني في كل لحظة، هذا النجاح هو حصاد تعبك. أسأل الله أن يحفظك لي وأن يطيل في عمرك ويجزيك عني خير الجزاء يا جنّتي.

إلى أبي الغالي

ذلك الجبل الذي لا يميل، لمن كان سنداً لي في كل خطوة، يا من أحمل اسمه بكل نخرة، الذي علمني معنى الصبر والاجتهاد، إلى من ضحّى براحته من أجل أن يوفر لنا الطريق الأفضل. نجاحي هذا هو جزء صغير من جميلك الكبير. أسأل الله أن يحفظك وبارك في عمرك ويدمك نفراً وسنداً.

إلى أخي وجدي رحمهما الله

أسأل الله أن يجعل قبركما نوراً وراحة وأن يجمعني بكما في جناته. غيابهما مؤلم لكن ذكراهما في القلب لا تغيب.

إلى جداتي أم السعد والزهرة

ممتنة لدعواتكن الصادقة وتربيتكن وحبكن الذي صنع مني ما أنا عليه. ربي يحفظكما ويطيل في عمركما.

إلى عائلتي الكريمة

كل باسمه و مقامه، شكراً على محبتكم ودعمكم وكلماتكم التي كانت دائماً دافعاً لي للإستمرار

إلى صديقاتي رفيفات الدرب

رفيفات الدرب وأجمل الذكريات أكيد لكم كل الشكر على اللحظات الجميلة و المساندة وقت التعب كل باسمها أحبكم

Hanine

بسم الله الرحمن الرحيم

وما توفيقني إلا بالله عليه توكلت وإليه أنيب

أما بعد، فإن الحمد لله نحمده ونستعينه ونستغفره، الذي بنعمته تم الصالحات، وبفضله تُنجز الغايات، وبكرمه تُكتب النجاحات. أهدي هذا النجاح، وهذه اللحظة التي انتظرتها طويلاً، إلى نفسي أولاً... إلى تلك الروح التي تعبت، وصبرت، وسقطت ثم قامت من جديد، ولم تجعل لليأس طريقاً إليها، فكان هذا التوفيق ثمرة إصرارها بعد توفيق الله.

ثم إلى أعلى الناس على قلبي، إلى من قال الله تعالى في حقهما: ﴿وقل ربّ ارحمهما كما ربياني صغيراً﴾

إلى أمي

إلى القلب الذي لا يعرف إلا العطاء، إلى الدعاء الذي كان يرافقني في كل خطوة دون أن أراه، إلى الحنان الذي احتواني في ضعفي، والصبر الذي علمني القوة، أمي... أنتِ الجنة في صورة إنسان، وأنتِ النور الذي لا ينطفئ في حياتي.

إلى أبي

إلى السند الذي لا يميل، إلى الرجل الذي علمني أن الكرامة تُبنى بالعلم، وأن النجاح لا يأتي صدفة، إلى من حمل همّي بصمت، وكان دعمه لي طريقاً خفياً نحو هذا الإنجاز، أبي... نغري الأول ومصدر قوتي الدائم. إلى ضلعي الثابت وأمان أيامي،

إلى أخي محمد

إلى من كان وجوده طمأنينة، وكلماته دعماً، ووقوفه بجاني سبباً في ثباتي، إلى من شاركني الطريق في لحظات القوة والضعف معاً. وإلى زوجة أخي الغالية...

إلى القلب الطيب الذي أضاف معنى جميلاً للعائلة، إلى من كانت سنداً بالكلمة الطيبة والدعاء الصادق، إلى من شعرت معها أن الدعم لا يكون فقط من الدم، بل من النقاء والنية الصافية، شكراً لوجودك الذي ترك أثراً لا يُنسى في هذه الرحلة. ولا أنسى رفقاء الدرب،

أولئك الذين شاركوني الطريق بكل ما فيه من تعب وأمل، كلُّ باسمه ومقامه، لهم في قلبي مكان لا يُنسى، وشكر لا يُحد. وفي الختام أقول:

الحمد لله حمداً كثيراً طيباً مباركاً فيه، الحمد لله الذي ما سلكنا البدايات إلا بتيسيره، وما بلغنا النهايات إلا بفضله وتوفيقه، وما هذا النجاح إلا بداية لطريق أكبر بإذن الله.

Manar

الملخص

تهدف هذه المذكرة إلى تصميم وإنجاز نظام ذكي وآمن لمراقبة الأحوال الجوية والتنبؤ بها باستخدام إنترنت الأشياء وطائرة بدون طيار. يعتمد النظام على المتحكم ESP32 ومجموعة من الحساسات لقياس درجة الحرارة والرطوبة والضغط الجوي والأمطار وشدة الإضاءة وتحديد الموقع الجغرافي. يتم إرسال البيانات بشكل آمن عبر بروتوكول MQTT فوق TLS وعرضها من خلال لوحة تحكم تفاعلية، مع تخزين نسخة احتياطية محلياً على بطاقة SD. كما يوفر النظام إمكانية التنبؤ بالحالة الجوية اعتماداً على البيانات البيئية السابقة. أظهرت النتائج فعالية النظام في جمع البيانات ونقلها وعرضها في الزمن الحقيقي، مما يجعله حلاً منخفض التكلفة وقابلاً للتنقل لمراقبة البيئة في المناطق المختلفة.

Abstract

This dissertation aims to design and implement a secure and intelligent weather monitoring and prediction system based on the Internet of Things (IoT) and a drone platform. The system relies on an ESP32 microcontroller and a set of environmental sensors to measure temperature, humidity, atmospheric pressure, rainfall, light intensity, and geographic location. The collected data are securely transmitted using the MQTT protocol over TLS and displayed through an interactive dashboard, while a local SD card provides backup storage. In addition, the system offers weather forecasting capabilities based on previously collected environmental data. The experimental results demonstrate the effectiveness of the proposed solution in collecting, transmitting, and visualizing weather data in real time, making it a low-cost and portable platform for environmental monitoring applications.

Keywords: IoT, Drone, Weather Monitoring, ESP32, MQTT, TLS, Environmental Sensors, Weather Forecasting.

Contents

Acknowledgments	i
الملخص	vi
Abstract	vii
General Introduction	xiii
Chapter 1: The Internet of Things and General Information on Meteorology	1
1.1 Introduction	2
1.2 The Internet of Things	2
1.2.1 Main Components of IoT	2
1.2.2 Workflow (Mechanism)	3
1.2.3 Key Applications	3
1.3 Meteorology	4
1.4 Basic Weather Elements	4
1.4.1 Instruments for Measuring Basic Elements	4
1.4.2 Meteorological Problems and Solutions	5
1.5 Weather Stations	5
1.6 Types of Weather Stations	5
1.7 Mobile Meteorological Stations	5
1.7.1 Features of Portable Stations	6
1.7.2 Comparison Between Portable and Fixed Weather Stations	7
Chapter 2: Drone Weather Station (Hardware & Communications)	8
2.1 Introduction	9
2.2 Overview of the Drone-Mounted Weather Station	9
2.3 The Drone as a Mobile Sensing Platform	9
2.4 Types of Unmanned Aerial Vehicles (UAVs)	10
2.4.1 Common Types of UAVs	10
2.5 Criteria for Choosing a Drone	10
2.6 Microcontroller	11

2.7	Sensors Used in the Station	12
2.7.1	DHT11 Sensor	12
2.7.2	BMP280 Atmospheric Pressure Sensor	14
2.7.3	Computation of Height Using Pressure (Barometric Altitude) . . .	16
2.7.4	Rain Sensor	16
2.7.5	Global Positioning System (GPS)	18
2.7.6	LDR (Light-Dependent Resistor)	19
2.8	Communication Modules for Wireless Transmission	22
2.8.1	Wi-Fi Communication	22
2.8.2	LoRa/LoRaWAN Communication	24
2.8.3	GSM/LTE (4G) Connectivity	26
2.9	Data Transmission Protocols	27
2.9.1	MQTT (Message Queuing Telemetry Transport) Protocol	27
Chapter 3:Conception and Implementation		31
3.1	Introduction	32
3.2	Part One: Conception	32
3.2.1	General Description of a Portable Weather Station	32
3.2.2	General Functional Diagram of the System:	33
3.2.3	Sequence Diagram Description	34
3.2.4	Circuit Diagram Description	34
3.2.5	Sensor Connections to ESP32:	36
3.3	Part two: Implementation	38
3.3.1	Development environment used	38
3.3.2	Libraries Used for Communication and Networking	39
3.3.3	Workflow of the Program	40
3.3.4	Reading from the Sensor and Data Processing	41
3.3.5	Data Transmission Using MQTT Protocol	41
3.3.6	Periodic Data Storage on SD Card	42
3.3.7	Data Formatting Using JSON	43
3.3.8	Data Security & Encryption	43
3.3.9	Dashboard Interface	43
3.3.10	Project Contribution	43
3.3.11	Machine Learning: Decision Tree	44
3.3.12	Software Platform & Project Structure (VS Code)	44
Chapter 4:Results Analysis and Discussion		48
4.1	Introduction	49
4.2	Dashbord Results	49
4.3	Temperature and Humidity Analysis	50

4.4	Weather Forecasting	50
4.5	GPS Location Tracking Result	51
4.6	Performance Evaluation	52
4.7	Discussion of Results	52
General Conclusion		54

List of Figures

1.1	Block Diagram of an IoT Network	2
1.2	Major Applications Used in the IoT	3
1.3	Basic Components of the Atmospheric Layer	4
1.4	Depicts a portable weather station	6
2.5	ESP32 Microcontroller	11
2.6	DHT11 Temperature and Humidity Sensor	13
2.7	BMP280 Atmospheric Pressure Sensor	15
2.8	Rain Sensor for Precipitation Detection	17
2.9	Global Positioning System (GPS) Module Used in Drone Systems	18
2.10	Light-Dependent Resistor (LDR) Sensor	19
2.11	SD Card Module used for local data storage	22
2.12	Wi-Fi Communication Module (ESP32) Used in Drone IoT Systems	22
2.13	LoRa Communication Module Used for Long-Range IoT Transmission	25
2.14	GSM/LTE (4G) Communication Module for IoT Connectivity	26
3.1	The system's block diagram illustrates how the sensors are connected to the ESP32 module	33
3.2	Sequence Diagram of the Proposed Weather Monitoring System	34
3.3	Diagram of the circuit showing the connections of the sensors to the ESP32	35
3.4	Li-ion/Li-Po battery used to power the ESP32-based system	38
3.5	ESP32 software libraries.	39
3.6	System Workflow Diagram	40
3.7	Reading the data from sensors	41
3.8	Storage on SD card	42
3.9	Dashboard Folder (Metadata Dashboard)	46
3.10	AI Model Folder (Prediction System)	46
4.1	Real-time monitoring dashboard displaying environmental sensor data.	49
4.2	Temperature and Humidity Analysis	50
4.3	Weather Forecast Dashboard	51
4.4	GPS location displayed on the dashboard interface	51

List of Tables

1.1	Table showing the types of weather stations	6
1.2	Comparison Between Portable and Fixed Weather Stations	7

General Introduction

Over the past few years, the world has made tremendous improvements in the IoT area due to the fact that now smart devices can gather, process, and send information about their environment through wireless networks in real-time. This trend has led to a paradigm change in many scientific and applied areas, such as meteorology that is based mainly on accurate and uninterrupted measurement of environmental data. Traditional weather stations use stationary sensors installed in particular locations, which limits their coverage area and makes it difficult for them to operate in rugged terrain, such as mountainous terrain, deserts, and areas hit by natural disasters. Thanks to advances in drone technology and wireless communication, it is now possible to create portable and compact weather stations that will be able to take atmospheric measurements from above and at different altitudes. At the same time, there are problems related to the confidentiality and safety of data sent wirelessly. With all that in mind, the purpose of this project is to create a weather station installed on a drone using IoT technologies.

Chapter 1

The Internet of Things

and General Information on Meteorology

1.1 Introduction

The purpose of this chapter is to give the reader the theoretical background needed for the understanding of the basic principles of the project by discussing the idea of the Internet of Things, including its structure and application fields, and by demonstrating how this technological approach could be applied in the area of meteorology. This chapter will discuss the instruments used in conventional meteorological stations, their drawbacks, and will show the importance of the development of mobile meteorological stations, which are the foundation of the drone-based station.

1.2 The Internet of Things

Internet of Things "IoT" is an interrelated system of smart objects and devices like vehicles, home appliances, and medical gadgets that connect to the internet to share and process data as well as automate decision-making. Internet of Things refers to a combination of individuals, processes, and technology with the help of connected sensors to achieve real-time monitoring, control, and assessment. In addition, the definition of IoT expands to encompass any object capable of having a digital identity on the Internet; for example, fridges or traffic lights and makes them "smart" objects. IoT works on four main components that function simultaneously[1].

1.2.1 Main Components of IoT

- **Sensors:** Collect data from the environment, such as temperature or motion.
- **Internet Connectivity:** Transmits data via WiFi, Bluetooth, or other networks.
- **Data Processing Software:** Analyzes data in the cloud (Cloud computing).
- **User Interface:** Displays results and allows control through applications such as mobile apps.

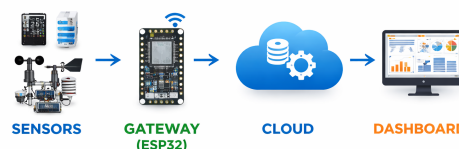


Figure 1.1: Block Diagram of an IoT Network

1.2.2 Workflow (Mechanism)

In an IoT system, a sequence of interconnected processes ensures efficient communication and operation. First, environmental parameters are measured using sensors that collect physical data from the surrounding environment. The acquired data are then transmitted through wireless communication networks to cloud platforms or remote servers. Once received, the data undergo processing and analysis using cloud computing technologies to extract meaningful information and insights. Based on the obtained results, automated actions and decisions can be executed, enabling intelligent and efficient system operation.

1.2.3 Key Applications

Applications of the Internet of Things have been seen in many fields like:

- **Smart Home:**Control of lights and air conditioners from one's mobile phone.
- **Healthcare:** Devices that monitor heart beat and give an alert to doctors.
- **Transportation:** Vehicles that can avoid accidents by information exchange.
- **Industry:** Monitoring machines for their timely maintenance to save money.
- **Agriculture:** Precise agriculture, which uses devices to check soil moisture content and automated irrigation system, besides drones to diagnose diseases[1].

The number of connected devices is expected to reach 50 billion by 2020, which enhances efficiency but also raises significant security concerns.



Figure 1.2: Major Applications Used in the IoT

1.3 Meteorology

Meteorology is the science that studies the atmosphere and all the phenomena that occur within it, such as wind, rain, storms, and temperature.

This science focuses on analyzing daily weather conditions and also studies long-term changes in the climate[2].

1.4 Basic Weather Elements

Meteorology is based on a set of key elements:

- **Temperature:** Measures how hot or cold the air is; it is a fundamental element that influences other weather phenomena.
- **Atmospheric Pressure:** The weight of the air column on the Earth's surface; it varies with altitude and temperature. Differences in pressure are the main cause of wind movement.
- **Wind:** The movement of air from areas of high pressure to areas of low pressure.
- **Humidity:** The amount of water vapor in the air, which is linked to the formation of clouds and precipitation.
- **Precipitation:** Includes rain, snow, and hail[2][3].

1.4.1 Instruments for Measuring Basic Elements

Meteorologists use several instruments, including:

- Thermometer: to measure temperature.
- Barometer: to measure atmospheric pressure.
- Hygrometer: to measure humidity.
- Anemometer: to measure wind speed.
- Rain gauge: to measure rainfall[4][5].



Figure 1.3: Basic Components of the Atmospheric Layer

1.4.2 Meteorological Problems and Solutions

Meteorological monitoring systems face several challenges that may affect their performance and reliability. These challenges include power outages, the impact of extreme weather conditions on equipment, limited data transmission capabilities in remote areas, and potential security vulnerabilities that may expose data to unauthorized access. To overcome these limitations, several modern solutions have been adopted, such as the use of solar panels to ensure continuous power supply, weather-resistant materials to improve system durability, long-range wireless communication technologies such as LoRa and GSM to enhance connectivity in remote regions, and data encryption techniques to strengthen data security and privacy.

1.5 Weather Stations

A weather station refers to facilities or locations dedicated to monitoring weather and climate conditions using various instruments. These instruments enable the measurement of factors influencing weather conditions, such as temperature, precipitation levels, wind speed, and others. Among the types of weather stations are:

- **Ben Nevis Observatory:** Precise measurements of temperature, humidity, precipitation, and atmospheric pressure.
- **Fort William Observatory:** A meteorological station used for continuous environmental monitoring and data collection.

1.6 Types of Weather Stations

Weather stations vary depending on their intended use and the level of accuracy required, and can be classified into the following types:

1.7 Mobile Meteorological Stations

Mobile meteorological stations are defined as compact, lightweight weather measurement systems that can be quickly transported and installed at any location without the need for fixed infrastructure. These stations contain a set of environmental sensors, a processing unit, and a wireless communication system that allows real-time transmission of measurements to a receiving station or cloud server.

Unlike traditional fixed stations, these systems can be deployed temporarily in multiple locations, providing great flexibility in collecting data from different areas as needed.

Table 1.1: Table showing the types of weather stations

Type	Accuracy and Measurements	Main use	Size and Portability
Household	Good (Wi-Fi, UV, Rain)	Characters and Gardens	Small, wireless
Specialized	Custom (Soil, Aviation)	Agriculture, Maritime	Miscellaneous, Industrials
Educational	Basic (Temperature, Humidity)	Education, Homes	Small, colorful
To predict	Comprehensive (pressure, radiometer)	Forecasts, Alerts	Medium, national networks
Portable – Mobile	Moderate (wind, humidity)	Fieldwork, Disasters	Small, backpack
Professionalism	Very high (wind, UV, rain)	Scientific research, courts	Large, steady

1.7.1 Features of Portable Stations

- Easy to transport and install
- No need for permanent ground fixtures
- Low power consumption
- Instant wireless connectivity
- Capable of operating in harsh conditions and rugged terrain



Figure 1.4: Depicts a portable weather station

1.7.2 Comparison Between Portable and Fixed Weather Stations

Table 1.2: Comparison Between Portable and Fixed Weather Stations

Fixed Weather Station	Portable Weather Station
Fixed location	Easily movable
Limited coverage	Wide coverage
Permanent installation	Temporary deployment
Ground measurements only	Measurements at different locations

Conclusion

Mobile stations are considered a smart and effective solution to the limitations of fixed stations, especially in emergency situations and challenging environments. Advances in the Internet of Things (IoT) and wireless communications have made these stations more accurate and faster in data transmission, paving the way for the use of drones as an ideal platform to carry them, as will be explained in the following chapters.

Chapter 2

Drone Weather Station

Hardware & Communications

2.1 Introduction

Unmanned aerial vehicles (UAVs/drones) are one of the most advanced technologies that have changed radically the ways of environmental and climatic data acquisition. UAVs are widely employed in science and engineering practices, including remote sensing, fire detection, and post-disaster damage evaluation due to the ability to gather information about the areas unreachable for measurements using traditional techniques, especially at lower altitudes. Thanks to drones, it is possible to conduct vertical profiling of the values of meteorological characteristics (temperature, humidity, atmospheric pressure, wind speed, etc.) at different altitudes and within short intervals, providing valuable opportunities for weather forecasting and analysis of the atmosphere. This chapter seeks to familiarize the reader with the idea of a weather station mounted on a drone by introducing the main software and hardware components of the proposed system and analyzing the way in which they fit into the Internet of Things architecture. The obtained data is sent from the mobile device to a server or cloud environment and is further analyzed and visualized with the help of interactive dashboards.

2.2 Overview of the Drone-Mounted Weather Station

Innovations in environmental measurements include the introduction of drones because they are now used as mobile sensor stations able to collect climate information through vertical profiling, especially within the lower part of the atmosphere called Atmospheric Boundary Layer. This kind of measuring system is scientifically referred to as Vertical Profiling, and is considered one of the most vital techniques in studying variations in atmospheric parameters with respect to altitude changes. Drones play an important role Today in climate studies, air pollution analysis, thermal layering, and the collection of meteorological information such as pressure and humidity in difficult-to-reach places.

Unlike conventional weather stations, the proposed system can collect measurements at different locations and altitudes, making it suitable for environmental monitoring, disaster assessment, and atmospheric profiling.

2.3 The Drone as a Mobile Sensing Platform

The idea behind this project is based on turning the drone into a scientific device with environmental sensors. In this case, the drone functions as a mobile measurement unit that is able to gather, analyze and send data about the current weather conditions.

The system is composed of several integrated components that ensure its proper operation. It includes a flight control structure that provides safe and stable drone operation, as well as a set of meteorological sensors used to measure environmental parameters such

as temperature, humidity, atmospheric pressure, and rainfall. A dedicated controller is responsible for maintaining flight stability and managing the overall system functions. In addition, a communication unit enables the wireless transmission of the collected data, while display and storage modules support data visualization, recording, and analysis.

This particular integration process can be called Cyber-Physical System that is widely used in the IoT world.

2.4 Types of Unmanned Aerial Vehicles (UAVs)

Drones vary in terms of their mechanical structure and functional capabilities, and these differences directly affect their suitability for carrying meteorological sensing systems and weather stations.

2.4.1 Common Types of UAVs

- **Quadcopter (four-rotor):** This type is equipped with four propellers arranged in an X configuration. It is one of the most commonly used UAVs in small and medium-scale applications. Its main advantage is high stability and ease of control in terms of altitude and direction. It is particularly suitable for short- and medium-range meteorological measurements, especially in urban and flat environments[6].
- **Hexacopter (six-rotor):** This UAV is equipped with six rotors, which provides higher payload capacity and improved stability under variable wind conditions. It is suitable for missions requiring additional equipment such as extra batteries or multiple sensors for enhanced data collection[6].
- **Fixed-wing UAV:** This type is designed for long-range missions with high energy efficiency. It is typically used for large-scale environmental monitoring and regional meteorological surveys. However, it has limited vertical take-off and landing capabilities and is generally less flexible in confined environments[7].

2.5 Criteria for Choosing a Drone

When choosing a drone that would have to accommodate a compact weather station, a variety of criteria need to be taken into account for efficient operation:

- **Weight Limitations:** The selected drone should be able to accommodate the total weight of all equipment (sensors, batteries, and the communications unit), without impacting its performance or stability.

- **Flight Duration:** The longer time spent flying increases the amount of measurements that can be taken within one session.
- **Stable Flying at Different Heights:** It is necessary to select a drone that ensures stable flying at different altitudes, which will allow accurate measurements at different atmospheric levels.
- **Control Easiness and Compatibility with Software:** Drift towards drones that use open-source software like ArduPilot or PX4 is preferable because it enhances compatibility with an IoT-based meteorological system[8].

2.6 Microcontroller

It is a small and inexpensive computer on a single chip, also called a single-chip micro-computer, and is used to control specific functions within electronic devices[9].

Its basic components, according to the same source , a microcontroller mainly consists of:

- **Central Processing Unit (CPU):** Executes instructions and performs calculations.
- **Memory:** Stores the program and data, typically including program memory and data memory.
- **Input/Output (I/O) ports:** Allow it to communicate with sensors, buttons, displays, and motors.
- **Interrupt controller:** Helps prioritize important tasks when multiple events occur simultaneously[9].



Figure 2.5: ESP32 Microcontroller

Why is ESP32 said to be a highly appropriate choice?

The ESP32 can be said to be a highly appropriate choice since it is a cheap microcontroller that has Wi-Fi and Bluetooth connectivity built into itself and adequate processing resources. Moreover, it also provides support for several communication interfaces, including GPIO, UART, SPI, I2C, and ADC interfaces. This means that ESP32 can effectively interface with many types of sensors, including DHT11, BMP280, GPS, and rain sensors.

The operation of a microcontroller involves several essential tasks. First, it acquires data from connected sensors through its input/output interfaces. The collected information is then processed according to the program stored in the microcontroller's memory. Based on the processed data, the microcontroller can make decisions and execute specific actions, such as triggering alarms or controlling connected devices. Furthermore, it supports data exchange through wireless communication technologies, including Wi-Fi and Bluetooth, enabling connectivity with external systems and remote monitoring platforms.[9].

2.7 Sensors Used in the Station

Climate sensors represent the fundamental physical component of any weather station. In this project, a set of integrated environmental sensors is used to ensure accurate, reliable, and comprehensive meteorological measurements.

2.7.1 DHT11 Sensor

The **DHT11** is an economical digital sensor that measures **temperature and relative humidity**. It is extensively utilized in embedded systems and Internet of Things applications because of its simple nature and ability to interface with microcontrollers like the Arduino and ESP32.

The sensor contains two main components inside a single package:

- A resistive humidity sensing element.
- An NTC thermistor for temperature measurement.

The DHT11 communicates using a **single-wire digital signal**, which simplifies the connection with microcontrollers without the need for analog-to-digital conversion[10].

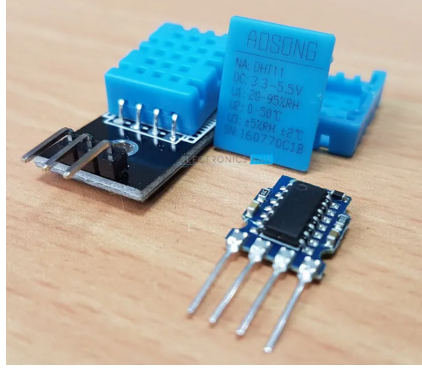


Figure 2.6: DHT11 Temperature and Humidity Sensor

Measurement Range

- Humidity: 20% to 80–90% RH
- Temperature: 0°C to 50°C

Advantages

- Low cost and suitable for educational projects
- Operates with low voltage (3.3V to 5V)
- Provides digital output (no ADC required)
- Low power consumption
- Easy to interface with Arduino and ESP32
- Compact and easy to integrate into circuits
- Stable performance for basic environmental monitoring[10].

Limitations

- Limited accuracy compared to advanced sensors
- Narrow measurement range
- Slow sampling rate (approximately one reading per second)
- Not suitable for high-precision industrial applications[10].

Working Principle

The sensor measures temperature using the NTC thermistor and humidity using the resistive humidity element. The internal circuit processes these analog values and converts them into a digital signal, which is transmitted through a single data pin to the microcontroller[10].

Technical Specifications

- Temperature range: 0 to 50°C
- Humidity accuracy: $\pm 2\%$
- Single-wire digital output
- Very low power consumption[10].

Why is it suitable for drones?

The selected sensor is particularly suitable for drone-based applications due to its lightweight design, which minimizes the impact on the drone's payload capacity. In addition, it does not require complex signal-conditioning circuits, simplifying system integration and reducing hardware complexity. The sensor is also resistant to vibrations generated during flight, ensuring reliable measurements under dynamic operating conditions. Furthermore, it provides sufficient accuracy for monitoring and studying the near-surface atmospheric layer, making it well suited for environmental and meteorological applications.[10].

2.7.2 BMP280 Atmospheric Pressure Sensor

The BMP280 is one of the highly employed sensors for measuring atmospheric pressure and temperature in high-precision digital form. This sensor was designed and manufactured by Bosch Sensortec for use in internet of things (IoT) and mobile device applications, where high accuracy, low power consumption, and compact size are required[11].



Figure 2.7: BMP280 Atmospheric Pressure Sensor

The Physical Principle of Pressure Measurement (Piezo-resistive Sensing):

The precision of the BMP280 sensor depends on the micromachined silicon diaphragm, which deforms mechanically when the atmospheric pressure changes. The change in the electrical resistance is measured by an internal circuit that generates a high-precision digital pressure value[12].

This method is called the piezo-resistive effect. Such sensors are common in professional applications because of their rapid response, accurate results, and thermal stability.

Temperature Measurement Internally:

There is an internal temperature sensor in the BMP280 that serves two primary roles, including: Giving the correct measurement of the temperature, As well as providing thermal compensation for any changes caused due to the influence of temperature on the pressure sensing of the diaphragm. This provides a highly accurate measure despite atmospheric changes in the drone's flight[11][12].

Technical Specifications:

- Pressure range: 300 to 1100 hPa (suitable for low-altitude applications)
- Pressure accuracy: up to ± 1 hPa
- Temperature accuracy: $\pm 1^\circ\text{C}$
- Very low power consumption (in the microampere range)
- Communication protocols: I2C or SPI
- Very small form factor suitable for drone integration[11].

2.7.3 Computation of Height Using Pressure (Barometric Altitude)

Some of the important factors that led us to select the BMP280 sensor for portable weather stations is its capacity to determine the estimated height without depending entirely on the GPS system. It does this through an application of a standard barometric equation used in meteorology. The equation is given as:

$$h = 44330 \left(1 - \left(\frac{P}{P_0} \right)^{0.1903} \right)$$

Where:

- h : is the altitude above sea level (m)
- P : is the atmospheric pressure measured by the BMP280 sensor (hPa)
- P_0 : is the standard atmospheric pressure at sea level (1013.25 hPa)

Significance of BMP280 in Portable Meteorology:

There are various reasons for the scientific significance of implementing such a sensor into a system using a drone. Firstly, it helps correlate measurements of atmospheric properties with altitude changes. Second, it makes possible studying the vertical profile of temperature and humidity. Third, it contributes to more precise estimates of altitude while reducing GPS measurement errors. In addition, the creation of pressure versus altitude graphs can be done through the use of such a barometer. Therefore, in many portable and scientific meteorology systems, altitudes are estimated based on a barometer rather than only using GPS[11][12].

2.7.4 Rain Sensor

Rain sensors are important for identifying rainfall while the drone is flying through the atmosphere. The rain sensors are a vital element in atmospheric field studies since they help in real-time identification of atmospheric water droplets or rainfall. The sensors play a crucial role in mobile weather stations on drones because dynamic changes in the atmosphere need to be detected immediately in different places[13].

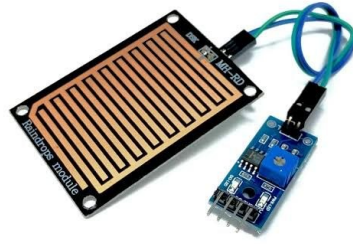


Figure 2.8: Rain Sensor for Precipitation Detection

Physical Principle of Operation:

The sensor comprises a metal plate with exposed parallel copper traces. During dry weather, the electrical resistance across the metal traces is very high. Rain falling onto the plate's surface creates a layer of water that conducts electricity, thus greatly lowering the electrical resistance across the traces. The change in the conductivity is measured and translated into either digital output (DO) or analog output (AO) by means of an onboard comparator. In other words, the principle behind operation of this device is based on changes in electrical conductivity due to water presence[13].

Technical Specifications

- Operating voltage: 3.3V – 5V
- Digital and analog output
- Very lightweight design
- Easy integration with microcontrollers
- Low cost[13].

Importance of the Sensor in Drone Applications

The integration of a rain sensor into a drone-based weather station offers several significant advantages for meteorological monitoring. It enables the detection of precipitation in areas where conventional ground-based weather stations are unavailable, thereby extending the coverage of environmental observations. In addition, rainfall events can be analyzed in relation to altitude and geographic location, providing a better understanding of spatial weather variations. Furthermore, the collected data support the study of the relationship between humidity, atmospheric pressure, and the onset of rainfall, contributing to more comprehensive weather analysis and forecasting.

2.7.5 Global Positioning System (GPS)

The Global Positioning System (GPS) is a central component of the drone-mounted meteorological station, as it enables each meteorological measurement to be accurately associated with its geographic location, altitude, and time of acquisition.

Most GPS modules used in integrated embedded systems rely on u-blox chips due to their high accuracy, reliability, and ease of integration with microcontrollers and IoT-based platforms.



Figure 2.9: Global Positioning System (GPS) Module Used in Drone Systems

The Scientific Principle Behind GPS (Trilateration)

The Global Positioning System (GPS) is based on the principle of trilateration, which relies on receiving signals from multiple satellites. Each satellite transmits its orbital position and the precise time of signal transmission.

By measuring the distance from at least four satellites, the system solves a set of equations to determine the receiver's position in terms of latitude, longitude, altitude, and UTC time.

Data Provided by the GPS Receiver

Embedded GPS modules communicate through UART serial interfaces by transmitting NMEA sentences containing essential positioning and navigation information. These data typically include latitude and longitude coordinates, altitude, speed, Coordinated Universal Time (UTC), and the number of visible satellites. The transmitted information can be directly received and processed by microcontrollers such as the ESP32 or Arduino, enabling accurate location tracking and integration with environmental monitoring applications.

Technical Specifications of u-blox Modules

- Positioning accuracy: up to 2.5 m
- Low power consumption

- Support for external antenna to improve signal reception
- Fast update rate (1 Hz to 10 Hz)
- Simple UART communication interface

Importance of GPS in a Portable Weather Station

The integration of GPS technology into a portable meteorological system offers significant scientific benefits. It enables each meteorological measurement to be associated with precise geographic coordinates, ensuring accurate spatial localization of the collected data. Furthermore, GPS information allows the visualization of measurements on interactive maps, facilitating the analysis of environmental conditions across different locations. The system can also compare the actual altitude provided by GPS with the barometric altitude measured by the BMP280 sensor, improving the reliability of altitude-related observations. In addition, GPS data support flight path tracking and the analysis of the spatial distribution of measurements. By providing accurate timestamps for all collected data, GPS technology contributes to temporal analysis and enables the creation of a comprehensive spatiotemporal meteorological database.

Integration with Other Sensors

When GPS is integrated with sensors such as the BMP280, it becomes possible to compare pressure-based altitude with GPS altitude, improve measurement accuracy, and study vertical variations in atmospheric conditions more effectively.

2.7.6 LDR (Light-Dependent Resistor)

A Light-Dependent Resistor (LDR) is used to measure the intensity of light incident on its surface. This component is typically made of cadmium sulfide (CdS), a photoconductive material whose electrical resistance varies according to the amount of light it receives.

When the light intensity increases, the resistance of the LDR decreases, and when the light intensity decreases, the resistance increases. This property makes it suitable for environmental light monitoring applications in weather stations and IoT-based systems[14].

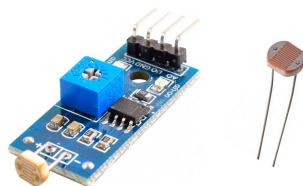


Figure 2.10: Light-Dependent Resistor (LDR) Sensor

Physical Principle (Photoconductivity)

When light strikes a cadmium sulfide (CdS) material, photons excite electrons within the material, resulting in the release of charge carriers. This increases the electrical conductivity of the material and consequently decreases its electrical resistance. This phenomenon is known as photoconductivity.

Mathematical Model of the LDR Sensor

The Light Dependent Resistor (LDR) operates based on the variation of its resistance according to light intensity. In general:

- Higher light intensity $\rightarrow$ lower resistance
- Lower light intensity $\rightarrow$ higher resistance

Voltage Divider Equation

When the LDR is connected in a voltage divider circuit with a fixed resistor, the output voltage is given by:

$$V_{out} = V_{cc} \times \frac{R_{LDR}}{R_{LDR} + R_{fixed}} [15]$$

LDR Resistance Calculation

From the previous equation, the LDR resistance can be expressed as:

$$R_{LDR} = R_{fixed} \times \left(\frac{V_{cc}}{V_{out}} - 1 \right) [15]$$

Conversion to Illuminance (Lux)

The light intensity (illuminance) can be approximately estimated using:

$$Lux = A \times (R_{LDR})^{-B} [15]$$

where A and B are empirical constants that vary depending on the type of LDR and are usually determined experimentally or taken from the sensor datasheet.

LDR Connection Method

The LDR is normally wired to a voltage divider circuit along with a constant resistor. The output voltage is measured from the analog input port of the microcontroller (Arduino and ESP32). The variation in voltage level will be proportional to the strength of the incoming light.

Technical Specifications

The sensor is characterized by several technical features that make it suitable for embedded and IoT applications. It has a lightweight design and low power consumption, which are essential requirements for portable and drone-based systems. In addition, the sensor is cost-effective, making it an affordable solution for environmental monitoring projects. It is highly responsive to variations in light intensity, allowing accurate detection of changing environmental conditions. Furthermore, the sensor can be easily interfaced with popular microcontroller platforms such as Arduino and ESP32, facilitating system integration and implementation.

Importance of the LDR in the Drone Station

Although the LDR does not provide an absolute measurement of solar radiation in W/m^2 , it is useful for:

- Providing a relative estimation of solar irradiance
- Interpreting temperature variations related to solar exposure
- Studying the effect of sunlight based on altitude and geographic location

Scientific Comparison with a Pyranometer

In professional meteorological stations, a pyranometer is used to measure solar radiation with high accuracy. The LDR, in contrast, represents a low-cost and simplified experimental alternative, mainly used for relative measurements rather than precise scientific quantification.

SD Card Module for Data Storage

The SD Card Module is regarded as a key component in the system, where it is used as a mechanism of storing data before uploading the same to the server or control dashboard. This component enables the saving of measurement data obtained by the sensor in the form of files for use in case of any failure in wireless communication.

The SD card module is interfaced with the ESP32 through the SPI communication method using MOSI, MISO, SCK, and CS pins along with VCC and GND pins for efficient data logging.

This module is widely used in IoT systems [16].

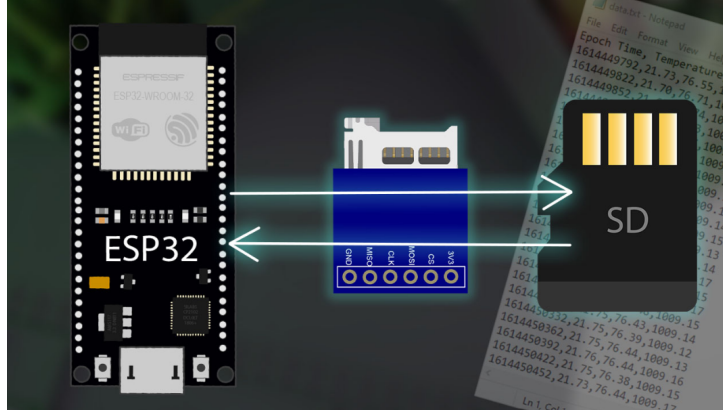


Figure 2.11: SD Card Module used for local data storage [16].

2.8 Communication Modules for Wireless Transmission

Communication modules play the role of intermediary between drones and monitoring systems installed either at ground or in the cloud. It is critical to choose the right communication technology considering several aspects like transmission range, energy consumption, data speed, and connection stability.

2.8.1 Wi-Fi Communication

Wi-Fi communication is based on the IEEE 802.11 family of standards, which define Wireless Local Area Network (WLAN) technologies. These standards enable devices to exchange data at high speeds over relatively short distances.

In drone-based systems, Wi-Fi is commonly used in conjunction with modern microcontrollers such as the ESP32, which integrates a built-in Wi-Fi module. This simplifies both hardware design and software integration, making it a practical solution for real-time data transmission in IoT-based meteorological applications[17].



Figure 2.12: Wi-Fi Communication Module (ESP32) Used in Drone IoT Systems

Why is Wi-Fi Suitable for Real-Time Streaming?

Wi-Fi offers several advantages that make it suitable for real-time data transmission in drone-based meteorological systems:

- High data transfer rates (in the Mbps range)
- Low latency communication
- Ability to establish a direct connection with a ground station (Access Point mode)

These characteristics make Wi-Fi ideal for real-time visualization of temperature, pressure, and humidity measurements during drone flights, especially in short-range field experiments[17].

Technical Analysis

Each sensor reading is transmitted as a small data packet, enabling efficient data exchange within the system. Wi-Fi technology provides sufficient bandwidth to support the transmission of a large number of packets per second while maintaining low communication latency. This capability ensures near real-time data delivery and monitoring. Furthermore, Wi-Fi is fully compatible with widely used communication protocols such as MQTT and HTTP, allowing reliable data transmission without introducing significant network bottlenecks.

Wi-Fi Operating Modes with the Drone

The ESP32 microcontroller can operate in two main modes:

- **Station Mode (STA):** The drone connects to an existing Wi-Fi router or access point.
- **Access Point Mode (AP):** The drone creates its own Wi-Fi network, allowing a computer or smartphone to connect directly.

This dual-mode capability provides high flexibility during field experiments.

Actual Range During Flight

Although Wi-Fi technology theoretically offers a communication range of up to approximately 100 meters, its actual performance during drone operations depends on several environmental and technical factors. These factors include the availability of a clear line of sight between the transmitter and receiver, the presence of electromagnetic interference, antenna orientation, and the drone's altitude. Consequently, Wi-Fi is most effective when the drone operates within the direct line of sight of the ground station, ensuring stable connectivity and reliable data transmission.

Power Consumption

Despite its numerous advantages, Wi-Fi technology presents certain limitations in drone-based applications. Compared with communication technologies such as LoRa and Bluetooth Low Energy (BLE), Wi-Fi generally consumes more power, particularly when maintaining continuous network connectivity. This increased energy demand may reduce the drone's flight time during prolonged operations. Nevertheless, power consumption can be optimized through several strategies, including reducing the sensor sampling rate, utilizing the deep sleep modes available in the ESP32 microcontroller, and transmitting data at predefined intervals rather than continuously. These techniques contribute to improving energy efficiency while preserving reliable communication performance.

Integration with IoT Protocols

Wi-Fi is highly compatible with several IoT communication protocols:

- MQTT for real-time data streaming
- HTTP for communication with local servers
- WebSocket for live dashboard visualization

When Should We Choose Wi-Fi for a Drone Project?

Wi-Fi is preferred in the following cases:

- Experiments conducted near a ground station
- Requirement of real-time data visualization
- Simple setup without complex infrastructure
- Laboratory or small-scale field environments

2.8.2 LoRa/LoRaWAN Communication

LoRa technology is a low-power, long-range wireless communication technology specifically designed for Internet of Things (IoT) applications that require the transmission of small amounts of data over long distances.

This technology is standardized and governed by the LoRa Alliance, which defines the technical specifications and ensures interoperability between LoRaWAN devices and networks[18].



Figure 2.13: LoRa Communication Module Used for Long-Range IoT Transmission

Technical Principles of LoRa

LoRa technology is based on Chirp Spread Spectrum (CSS), a modulation technique that encodes data using chirp signals whose frequency varies gradually over time. This approach makes the signal:

- Highly resistant to interference
- Capable of long-distance communication
- Effective in open and challenging environments

Advantages of LoRa in the Drone Project

- Very long communication range (approximately 2–15 km depending on environmental conditions)
- Very low power consumption
- Suitable for transmitting small sensor data packets
- Reliable operation in rural and mountainous areas[18].

Limitations

- Low data transmission rate
- Not suitable for video or large data transfers
- Requires a gateway infrastructure when using LoRaWAN[19].

Importance of LoRa in a Weather Station

In a drone-based meteorological system, LoRa technology offers several significant advantages for long-range communication. It enables data transmission even when the drone

operates at considerable distances from the ground station, thereby extending the monitoring coverage area. In addition, LoRa reduces the dependence on high-power communication links, contributing to improved energy efficiency and longer operational duration. Furthermore, it provides reliable communication performance in open and remote environments where conventional wireless technologies may experience connectivity limitations. These characteristics make LoRa a suitable solution for environmental monitoring applications requiring long-range and low-power data transmission.

2.8.3 GSM/LTE (4G) Connectivity

This technology relies on global mobile communication networks in accordance with the 3GPP LTE standard. It enables data transmission directly to the Internet through cellular networks without the need for a local gateway[20].

GSM/LTE communication is widely used in IoT systems to ensure long-range connectivity, especially in remote areas where Wi-Fi coverage is limited or unavailable[21].



Figure 2.14: GSM/LTE (4G) Communication Module for IoT Connectivity

Technical Principle

The system operates through a GSM/LTE module embedded in the drone, using a SIM card to access cellular networks. Data collected from onboard sensors is transmitted directly to a cloud server via the mobile network.

As a result, the system does not require a nearby local network infrastructure, making it suitable for wide-area deployments.

LTE/GSM Features

- Wide coverage, including remote regions
- Direct internet connectivity via cellular networks
- Suitable for real-world field applications
- Ability to connect directly to cloud databases

Limitations

- Higher power consumption compared to technologies such as LoRa and BLE
- Requires a SIM card and subscription plan
- Depends on cellular network availability
- Reduced performance in mountainous or isolated areas

Importance of LTE in the Drone Project

LTE/GSM communication provides several key advantages in the drone-based meteorological system:

- Enables real-time transmission of data to cloud platforms
- Suitable for operations beyond Wi-Fi range
- Supports professional IoT and large-scale monitoring systems

2.9 Data Transmission Protocols

2.9.1 MQTT (Message Queuing Telemetry Transport) Protocol

The MQTT protocol is one of the most widely used communication protocols in the Internet of Things (IoT), particularly in systems that rely on lightweight devices such as microcontrollers and sensors mounted on drones.

It was originally developed by IBM and later standardized as an open protocol under the OASIS organization. The main objective of MQTT is to provide a lightweight and efficient communication mechanism suitable for resource-constrained environments.

MQTT Architecture

MQTT is based on the Publisher/Subscriber model, where communication is managed through an intermediary called the Broker instead of direct device-to-device communication.

Working Principle:

- The drone (Publisher) sends sensor data (temperature, pressure, humidity, etc.)
- The data is transmitted to the MQTT Broker
- Applications or dashboards (Subscribers) receive the data from the Broker

System Flow:

Sensor → ESP32 → MQTT Broker → Dashboard / Cloud

Importance of MQTT in the Drone Project

MQTT is particularly important in drone-based systems because:

- The drone is constantly moving
- Direct communication with servers may be unstable
- The Broker ensures reliable and stable communication

Thus, MQTT solves the problem of communication interruptions during movement.

Technical Characteristics of MQTT

Lightweight Protocol: MQTT uses a very small header size compared to protocols like HTTP, which reduces:

- Data consumption
- Power consumption
- Transmission delay

Topics System: Data is organized using hierarchical topics such as:

- drone/temperature
- drone/humidity
- drone/pressure

Quality of Service (QoS): MQTT defines three levels of message delivery reliability:

- QoS 0: At most once delivery (no acknowledgment)
- QoS 1: At least once delivery (with acknowledgment)
- QoS 2: Exactly once delivery (most reliable)

Advantages in Weather Systems (Drone Weather Station)

Using MQTT in a drone-based weather station provides:

- Real-time data transmission
- Reduced data loss during movement
- Low power consumption suitable for drones
- Easy integration with cloud platforms (AWS IoT, Mosquitto)
- Support for multiple devices in the same system

Integration with ESP32 and the Drone

In this system:

- The ESP32 collects data from sensors
- Data is transmitted via Wi-Fi
- The ESP32 publishes data to the MQTT broker
- Data is visualized on a dashboard (Node-RED or web application)

Practical Applications

MQTT is widely adopted in various Internet of Things (IoT) applications due to its lightweight architecture and efficient communication capabilities. It is commonly used in smart weather stations for environmental data collection and transmission, as well as in smart city systems that require real-time monitoring and control. Furthermore, MQTT plays an important role in environmental monitoring applications, drone tracking systems, and industrial IoT solutions, where reliable and low-bandwidth communication is essential for data exchange between connected devices and cloud platforms.

Limitations

Despite its advantages, MQTT has some limitations:

- Requires a central broker
- Depends on network connectivity (Wi-Fi/LTE)
- Not suitable for large file transfers

Conclusion

This chapter highlights the importance of designing a drone-mounted weather station as a modern and effective solution for collecting climate data in various environments, particularly in areas that are difficult to access using traditional ground-based stations. The chapter discussed the general concept of the system and its main components, including the unmanned aerial vehicle (UAV), sensors, processing units, communication technologies, and data transmission protocols within the Internet of Things (IoT) framework.

It has been demonstrated that the success of such systems depends on the integration of several key elements, most notably sensor accuracy, wireless communication efficiency, and the selection of appropriate protocols such as MQTT to ensure fast and reliable data transmission to cloud or server platforms. In addition, positioning systems play a crucial

role in linking measurements to geographic location and altitude, thereby increasing the scientific value of the collected data.

Finally, the drone-mounted weather station represents a significant step toward smart monitoring systems based on IoT technologies, opening new perspectives for applications in meteorology, environmental monitoring, and smart agriculture. The next chapter will focus on the practical design and implementation of the system, as well as its integration with software platforms.

Chapter 3

Conception and Implementation

3.1 Introduction

Having described the theoretical aspect of the drone-mounted weather station concerning its use of different kinds of sensors and communication technology used for information exchange, the following chapter will focus on the practical side of the system implementation involving both hardware and software components to develop a working prototype. The aim of this chapter is to provide details about the linking of all sensors to the processor, the acquisition of the meteorological information, the local storage of the data and its wireless transfer to the control center. The chapter will also focus on the order of performance of the operations performed by the system, beginning with the collecting of environmental information, then moving on to its analysis using the controller, then to the data storage and access.

To achieve the aim of the chapter, the latter is going to be split into two sections:

- **Conception :** In this section the linking of different electronic parts of the drone will be described.
- **Implementation :** This part will be devoted to the data collection, structuring, storage, and transfer over the network.

This chapter links theory to practice in regard to the discussed weather station.

3.2 Part One: Conception

3.2.1 General Description of a Portable Weather Station

The portable weather station is a well-thought-out system intended for gathering meteorological data during the flight and under difficult conditions when other ways of doing so would not work. This system requires integration of different environmental sensors with a main processor. That means that it allows measuring different kinds of weather factors.

The system is capable of measuring a range of environmental parameters essential for meteorological analysis. These include temperature, humidity, atmospheric pressure, and ambient light intensity, as well as rainfall intensity for precipitation monitoring. In addition, the system records geographical coordinates and altitude, enabling precise spatial localization of all collected data and supporting advanced environmental and atmospheric studies.

The working principle of this system is based on operation of sensors that act as input devices and collect information about some kind of physical phenomena and transmit it in the form of electric impulses, which are further transmitted to the ESP32 processor

that analyzes it and translates it to digital data. Therefore, this system turns out to be a convenient mobile measuring tool.

3.2.2 General Functional Diagram of the System:

The design diagram displays the structure and interconnections of the different elements of the system, where the whole sensor network is interconnected with the ESP32 processor, referred to as the “brain of the system,” handling the data transfer.

Functionally, the system is comprised of the following elements:

- Sensors : sources of environmental information
- ESP32 : processor of the system
- SD-card : data storage
- Wi-Fi transmitter-receiver : wireless communication
- Dashboard : information display element

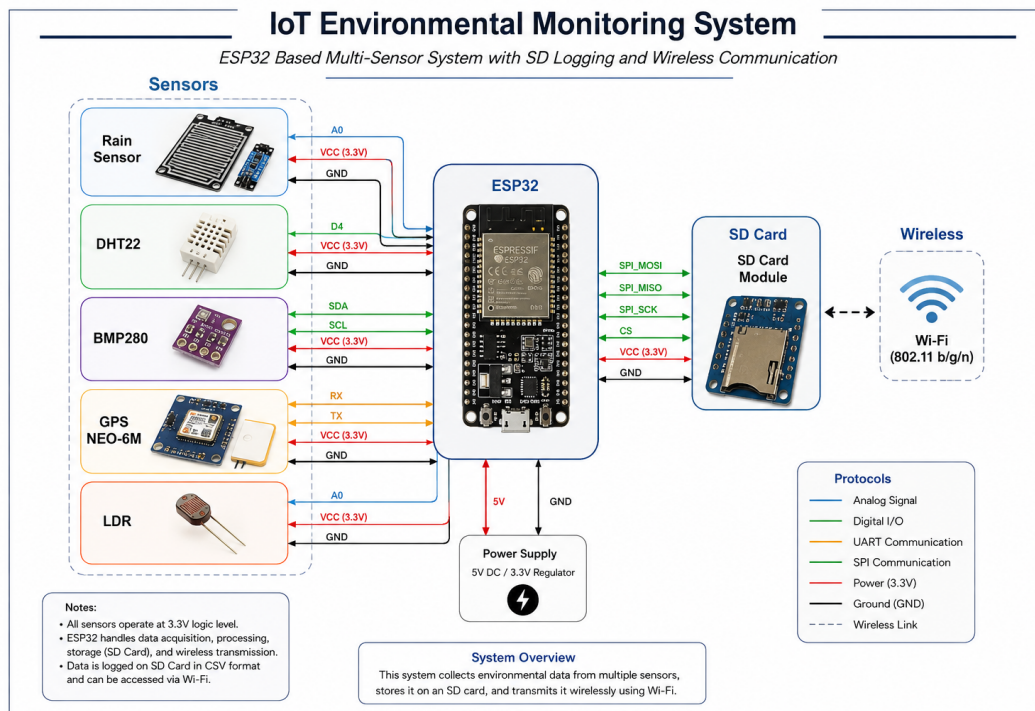


Figure 3.1: The system’s block diagram illustrates how the sensors are connected to the ESP32 module

Diagram of System: In this block diagram of the system, the connections between the sensors and the ESP32 board are shown

Interpretation of Block Diagram

In this diagram, we can see that all the sensors behave like input modules that take measurements from the atmosphere and change them into electric signals. These signals are further sent to the ESP32 where they are processed, and arranged in digital form. These pieces of information may then move on to follow either of two paths:

1. Saved as backup data in the local memory i.e., in SD Card.
2. Transmitted to the server by way of Wi-Fi using the MQTT protocol.

The arrangement ensures there will be no data loss even during a network failure situation.

3.2.3 Sequence Diagram Description

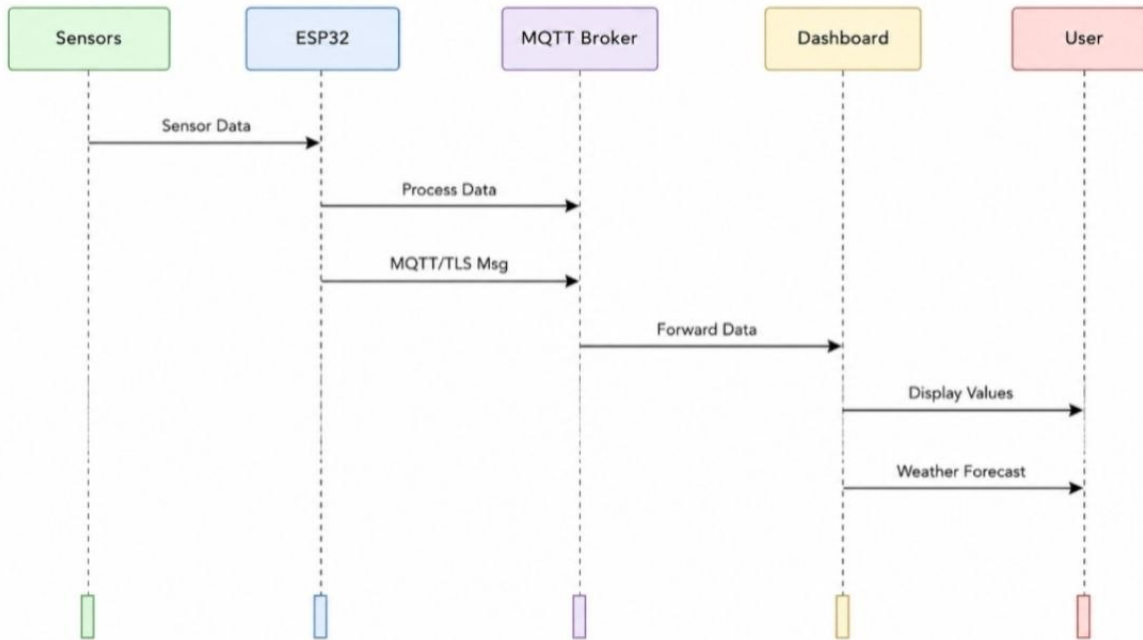


Figure 3.2: Sequence Diagram of the Proposed Weather Monitoring System

Figure 3.2 illustrates the sequence of interactions within the proposed weather monitoring system. Environmental data are collected by sensors and processed by the ESP32 microcontroller. The processed data are then transmitted securely through MQTT over TLS to the MQTT broker, which forwards them to the dashboard for visualization. Finally, the user can monitor real-time measurements and forecast information through the dashboard interface.

3.2.4 Circuit Diagram Description

The scheme illustrates how the different components of the system are physically connected to the ESP32 microcontroller through electrical signals. It also defines the type

of communication protocol used for each module, as well as the corresponding input/output ports required for proper interfacing. The system is based on several communication standards, including the I2C protocol for connecting the BMP280 atmospheric pressure sensor, the SPI protocol for interfacing the SD card storage module, GPIO and ADC interfaces for sensors such as the DHT, LDR, and rain sensors, and UART communication for the GPS module.

Purpose of the Circuit Diagram

This circuit diagram seeks to depict the connection of sensors to the processor in detail, by providing information about which pins are being used in ESP32 to connect the sensors. This also shows how the system is practically implemented.

From this diagram, it is evident that the system design was done in a manner such that:

- Port conflict issues are avoided
- The correct communication protocol is used for each type of sensor
- Signal reliability is ensured

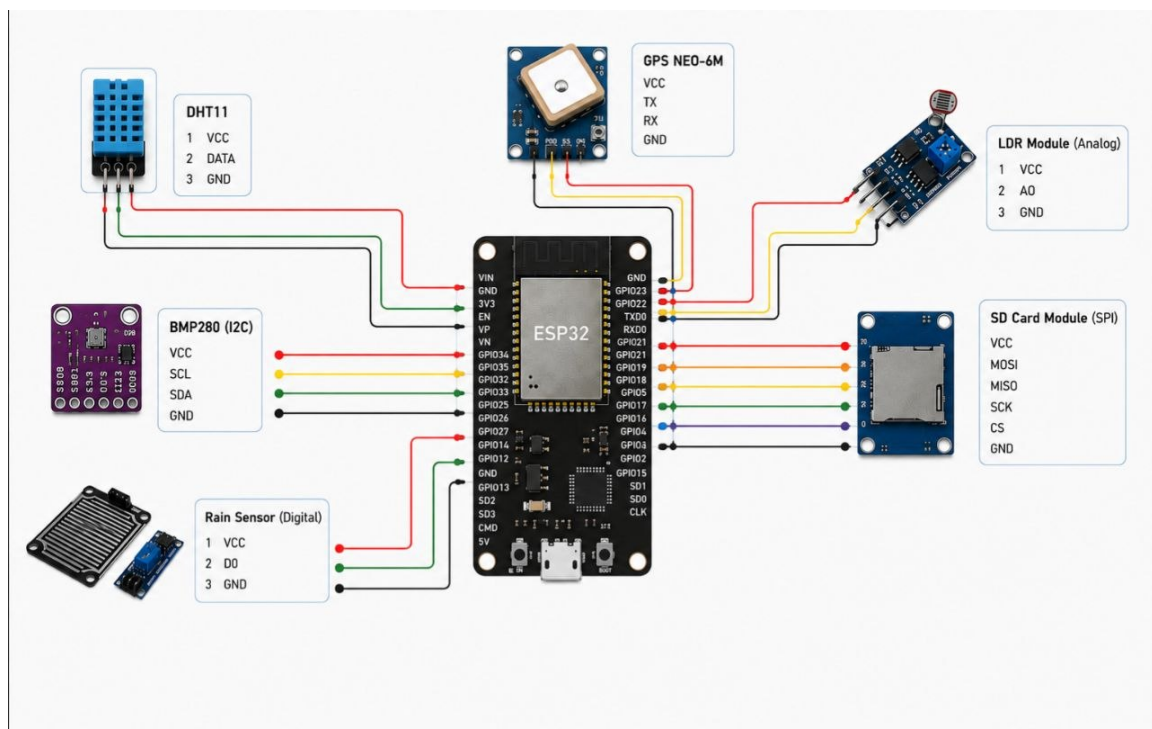


Figure 3.3: Diagram of the circuit showing the connections of the sensors to the ESP32

Every sensor was connected to its corresponding pin in the ESP32 microcontroller depending on the kind of signal it deals with, as follows:

3.2.5 Sensor Connections to ESP32:

1. DHT11 (Temperature and Humidity)

VCC → 3.3V (ESP32)

GND → GND

DATA → GPIO 4

A 4.7kΩ pull-up resistor between VCC and DATA is recommended if needed.

Functions:Sensing air temperature and humidity.

2. BMP280 (Barometric Pressure)

VCC → 3.3V

GND → GND

SDA → GPIO 21

SCL → GPIO 22

Function: Measures barometric pressure and temperature, and is also used to calculate altitude during flight via the I2C protocol.

3. Rain Sensor

VCC → 3.3V or 5V depending on the module

GND → GND

AO / DO → GPIO 35 (ADC)

Function:To detect rainfall by measuring variations in electrical resistance on the surface of the sensor..

4. LDR Sensor (Light Intensity)

Analog output → GPIO 34 (ADC1)

Connected in a voltage divider circuit with a fixed resistor.

Function:To measure ambient light intensity as an indication of variations in solar energy.

5. GPS Module (Positioning)

VCC → 3.3V

GND → GND

TX → GPIO 16

RX → GPIO 17

Function: To determine the geographical position (latitudes, longitudes) and the altitude of the drone during flights.

6. SD Card Module (Data Storage)

VCC → 3.3V / 5V

GND → GND

MOSI → GPIO 23

MISO → GPIO 19

SCK → GPIO 18

CS → GPIO 5

Function: To store information in CSV format locally to have a backup in case of wireless communication failure.

ESP32 (Main Processing Unit)

The ESP32 serves as the system's main controller, responsible for collecting data from all sensors (temperature, humidity, pressure, light, rain, and location), processing it, and converting it into digital data that can be transmitted or stored. It also stores a copy of the data on an SD card and transmits it over Wi-Fi using the MQTT protocol to the server or dashboard.

The ESP32 was chosen because it features built-in Wi-Fi and Bluetooth, multiple communication interfaces (GPIO, I2C, SPI, UART), and high performance with low power consumption. These characteristics make it highly suitable for IoT applications and drone-based embedded systems.

Power Supply and Energy Management of the ESP32

The power source of the ESP32 board is the Li-Ion/Li-Po rechargeable battery, with a nominal voltage rating of 3.7V, which works well with portable applications, like the drone. In order to maintain the steady power source, the above battery voltage is either stepped up or down, based on the ESP32 board type, considering that its voltage ranges from 4.2V in full charge and 3.0V in discharge state.



Figure 3.4: Li-ion/Li-Po battery used to power the ESP32-based system

This regulation is essential to guarantee stable operation of the microcontroller and to prevent unexpected resets or performance issues during flight.

3.3 Part two: Implementation

3.3.1 Development environment used

Arduino IDE:

The Arduino IDE is a complete integrated development environment specifically designed for writing software for microcontrollers and embedded electronic boards like Arduino, ESP32 and ESP8266. It integrates all necessary features for a developer in a single environment like editing the code, compiling the code, checking for errors, and finally uploading the compiled code through serial port directly to the target board. The ide uses mainly C / C++ programming language with a lot of ready-to-use libraries that help communicate with external devices such as sensors, communication modules, displays, and other electronic hardware. It is known for its simple and user-friendly graphical user interface[22].

Visual Studio Code (VS Code):

It is an efficient and speedy code editor that helps developers edit and write code using different languages such as HTML, CSS, JavaScript, PHP, Python, C++, Java, SQL, and other languages. Although it looks quite simple, it has the capability to provide the same services as those offered by large integrated development environments (IDEs). VS

Code is used in web and app development... It comes with AI, plug-ins, customizable user interface, among many other advantages[23].

3.3.2 Libraries Used for Communication and Networking

The following libraries are used to manage communication and networking functions within the ESP32-based system:

```
manarPro.ino
1  #include <WiFi.h>
2  #include <WiFiClientSecure.h>
3  #include <PubSubClient.h>
4  #include <Wire.h>
5  #include <Adafruit_BMP085.h>
6  #include "DHT.h"
7  #include <TinyGPSPlus.h>
8  #include "FS.h"
9  #include "SD.h"
10 #include "SPI.h"
11 #include <HardwareSerial.h>
```

Figure 3.5: ESP32 software libraries.

- **WiFi.h:** Used to connect the ESP32 to a Wi-Fi network.
- **WiFiClientSecure.h:** Enables secure encrypted communication using SSL/TLS protocols.
- **PubSubClient.h:** Used for MQTT communication to publish and receive data.
- **Wire.h:** Manages I2C communication between the ESP32 and sensors.
- **Adafruit_BMP085.h:** Used to interface with the BMP180 pressure sensor.
- **DHT.h:** Allows reading temperature and humidity values from the DHT22 sensor.
- **TinyGPSPlus.h:** Processes GPS data such as latitude and longitude.
- **FS.h:** Provides file system management functions.
- **SD.h:** Used to read and write data on the SD card.
- **SPI.h:** Manages SPI communication required for the SD card module.
- **HardwareSerial.h:** Enables serial communication with external modules such as GPS.

3.3.3 Workflow of the Program

The program works according to an organized workflow, whereby a number of processes are executed periodically and in order, in order to ensure that the environmental data is properly acquired and analyzed. First, the program is turned on and all sensors and communication modules are initialized, after which data is acquired from the sensors.

The data acquired is verified to ensure there is no error in it. After verifying the data, the program transfers the data to two locations, namely the local storage on the SD card and the wireless network through the use of the MQTT protocol.

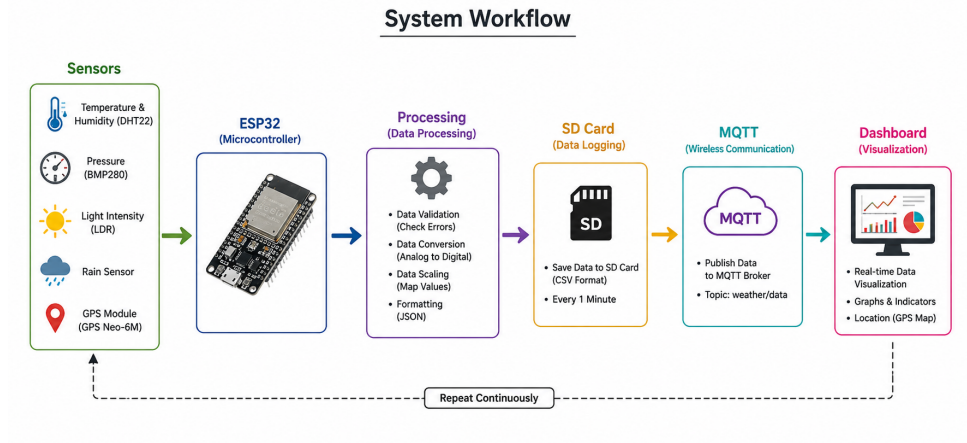
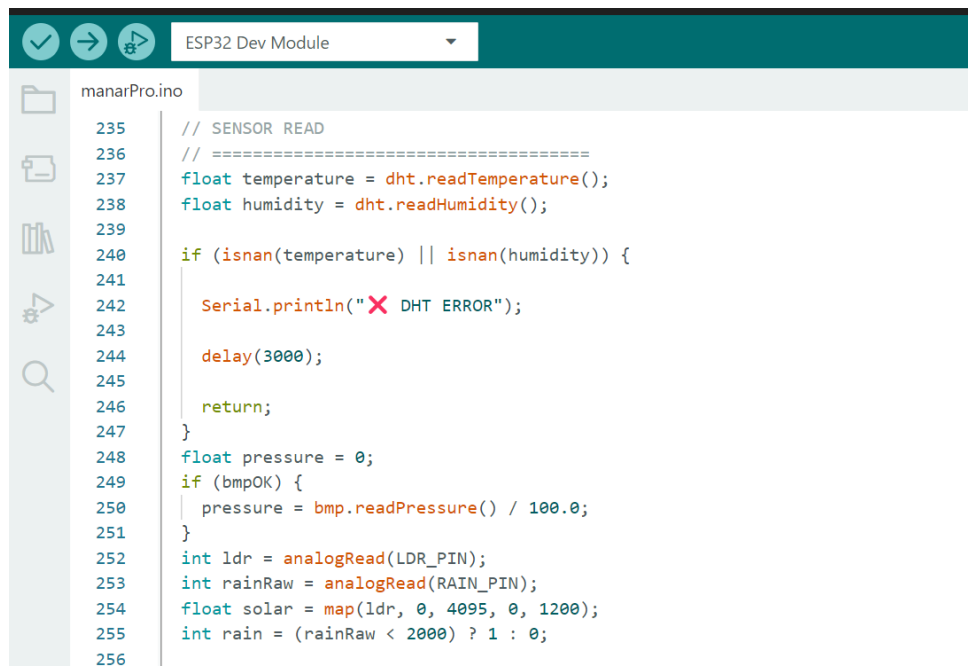


Figure 3.6: System Workflow Diagram

Figure 3.5 illustrates the overall workflow of the system. Environmental data are first collected through the integrated sensors connected to the ESP32 microcontroller. The acquired measurements are then processed and formatted into JSON structures. A local backup copy is stored on the SD card, while the same data are transmitted securely through the MQTT protocol to the broker and dashboard interface. Finally, the collected dataset can be used by the machine learning model to generate weather predictions.

3.3.4 Reading from the Sensor and Data Processing



```
235 // SENSOR READ
236 // =====
237 float temperature = dht.readTemperature();
238 float humidity = dht.readHumidity();
239
240 if (isnan(temperature) || isnan(humidity)) {
241
242     Serial.println("✖ DHT ERROR");
243
244     delay(3000);
245
246     return;
247 }
248 float pressure = 0;
249 if (bmpOK) {
250     pressure = bmp.readPressure() / 100.0;
251 }
252 int ldr = analogRead(LDR_PIN);
253 int rainRaw = analogRead(RAIN_PIN);
254 float solar = map(ldr, 0, 4095, 0, 1200);
255 int rain = (rainRaw < 2000) ? 1 : 0;
256
```

Figure 3.7: Reading the data from sensors

This part of the code illustrates how to read values coming from various sensors connected to the ESP32 module. Initially, the DHT library functions are called to read temperature and humidity. The validity of the read values is checked using the `isnan()` function to avoid errors resulting from failed readings, and in case of an error, the process is temporarily halted.

Then, the atmospheric pressure value is read from the BMP sensor once its readiness is confirmed, converting the value from Pascal to hPa by dividing it by 100.

Then, the `analogRead()` function is used to read the analog values from the light sensor (LDR) and the rain sensor. The light value is converted to an approximate solar radiation value using the `map()` function, while the analog value of the rain sensor is converted to a digital value (0 or 1) to determine the presence or absence of rain based on a specific threshold.

These steps allow the conversion of raw signals from sensors into organized and processable digital data, storage, and transmission.

3.3.5 Data Transmission Using MQTT Protocol

After processing the sensor data, the data is converted into a structured JSON object prior to data transmission over the network.

The JSON message comprises all measured variables like temperature, humidity, atmospheric pressure, rainfall, luminosity, and geographical coordinates.

Such structure of data makes possible the efficient transmission of data. The ESP32

transmits the data to the broker via the `client.publish(topic, payload)` function. The MQTT protocol is a lightweight protocol based on the Publisher/Subscriber architecture, which means that the ESP32 works as a publisher transmitting data to a broker while the dashboard works as a subscriber receiving this information. Such approach provides low latency, high efficiency, and reliability of transmission of data.

3.3.6 Periodic Data Storage on SD Card

```
// ===== SD SAVE =====  
String row =  
    String(temperature,1) + "," +  
    String(tempMin,1) + "," +  
    String(tempMax,1) + "," +  
    String(tempAvg,1) + "," +  
    String(humidity,1) + "," +  
    String(pressure,1) + "," +  
    String(rain) + "," +  
    String(solar,1) + "," +  
    String(lat,6) + "," +  
    String(lng,6);  
  
    saveToSD(row);  
  
    delay(3000);  
}
```

Figure 3.8: Storage on SD card

To ensure data persistence, the system stores all collected measurements locally on an SD card in CSV format.

The storage process is executed periodically using the `millis()` function, which checks whether a predefined time interval (approximately one minute) has elapsed since the last saving operation.

When this condition is met, a data row is created containing all sensor values, including temperature (minimum, maximum, and average), humidity, atmospheric pressure, rainfall status, light intensity, and GPS coordinates.

The `saveToSD()` function is then used to append this data to a CSV file, ensuring that each record is stored on a new line.

This mechanism guarantees data availability even in case of network failure, providing a reliable backup and historical record of environmental measurements.

3.3.7 Data Formatting Using JSON

In this system, data collected from sensors is formatted using JSON (JavaScript Object Notation) before being transmitted over the network. JSON is a lightweight and simple data format that is widely used in Internet of Things (IoT) applications because it is easy to read and process.

The ESP32 controller aggregates all measured values such as temperature, humidity, atmospheric pressure, precipitation status, and light intensity, along with geographic coordinates (latitude and longitude), into a single JSON object. This allows all data to be transmitted in a structured and unified format within a single message.

The use of JSON improves data organization and facilitates data exchange between the ESP32, the MQTT server, and the dashboard. It also makes it easier to reuse this data later for analysis or integration into artificial intelligence models.

3.3.8 Data Security & Encryption

Improve the security of transmitted data, the proposed solution employs the MQTT protocol with a connection that is encrypted via the TLS/SSL protocol by using the `WiFiClientSecure` class. Thus, such a design can help protect the transmitted data between the ESP32 and the MQTT broker.

At the moment, TLS is used by the ESP32 to establish the encryption connection. Nonetheless, the certification process is bypassed through using the `setInsecure()` function. It means that even though the transmitted data is encrypted, the validation of the certificate is performed. Such a decision was made to ease the implementation process in the test environment.

3.3.9 Dashboard Interface

The dashboard is considered the last interface of the system through which data from the environment collected by the weather station will be shown in real time. The data is then received through the MQTT protocol, as the ESP32 sends the data to the broker (server), which further transfers the data to the dashboard. Different parameters like temperature, humidity, atmospheric pressure, light level, whether there is any rain or not, as well as geographic coordinates, are displayed through graphs and indicators. The dashboard is regarded an important part of the current system because it offers real-time weather monitoring along with climatic variations analysis through this system.

3.3.10 Project Contribution

The main contribution of this project lies in the integration of IoT technologies, secure communication mechanisms, and machine learning techniques into a single portable me-

teorological platform. The proposed system combines environmental sensing, real-time monitoring, encrypted data transmission through MQTT over TLS, local backup storage using an SD card, and weather prediction capabilities. Unlike traditional fixed weather stations, the system can be deployed on a drone, allowing atmospheric measurements at different locations and altitudes.

3.3.11 Machine Learning: Decision Tree

Machine learning is one of the advanced stages of this project, where environmental data collected from the weather station is used to build a model capable of predicting future weather conditions. This model relies on analyzing relationships between various climatic variables such as temperature, humidity, atmospheric pressure, and light intensity.

Selection of the Decision Tree Algorithm

The Decision Tree algorithm was selected due to its simplicity and effectiveness in classification tasks. This algorithm works by splitting the data into a series of sequential decisions that ultimately lead to determining the appropriate category (e.g., sunny, rainy).

This algorithm is characterized by its low mathematical complexity and its ability to provide easily interpretable results, making it suitable for embedded systems and Internet of Things applications.

How the Model Works

The model-building process consists of two main stages:

- **Training Stage:** Historical data is used to teach the model the relationships between inputs (climatic values) and outputs (weather conditions).
- **Testing Stage:** The model's performance is evaluated using new data to verify its accuracy and predictive capability.

After training, the model becomes capable of predicting weather conditions based on values collected directly from sensors.

3.3.12 Software Platform & Project Structure (VS Code)

The software part of the project developed in **VS Code** is organized into two main folders. Each folder is responsible for a specific functional role within the system architecture, ensuring clarity, modularity, and ease of maintenance.

Dashboard Folder (Metadata Dashboard)

This folder contains all files related to the **web dashboard interface** used to visualize environmental data in real time. It includes the metadata dashboard components such as analytics pages, data visualization charts, and monitoring panels, which are accessed locally through a web browser using the `localhost` address.

The dashboard communicates with the system through a **local server** and the **MQTT protocol** to receive and display sensor data transmitted from the ESP32 unit.

AI Model Folder (Prediction System)

This folder is dedicated to the **artificial intelligence** component of the project, where the weather prediction model is built and executed. It contains the following elements:

- The dataset file (`data.csv`) that stores historical sensor measurements.
- A **data cleaning script** used to preprocess and prepare the data before training.
- A **server file** responsible for receiving data from the API and updating the dataset.
- The `training.py` file, which reads the stored data and trains the Machine Learning model using the **Random Forest** algorithm.

After the training process, the system generates a **trained model file** that is later used to perform predictions and determine the expected weather class based on input sensor values.

This separation into two folders ensures a clear distinction between:

- The **data visualization layer** (Dashboard),
- The **data analysis and prediction layer** (AI Model),

making the overall system architecture well-structured, modular, and easier to manage and scale.

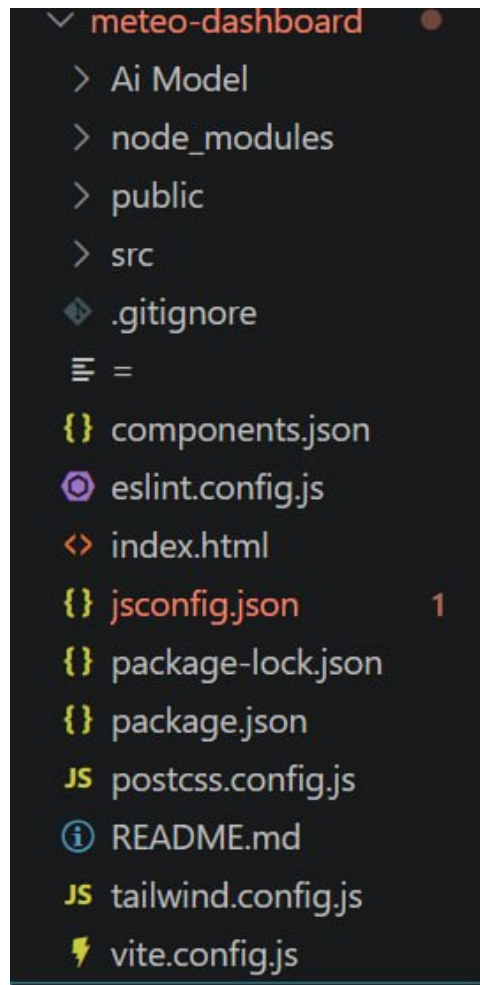


Figure 3.9: Dashboard Folder (Metadata Dashboard)

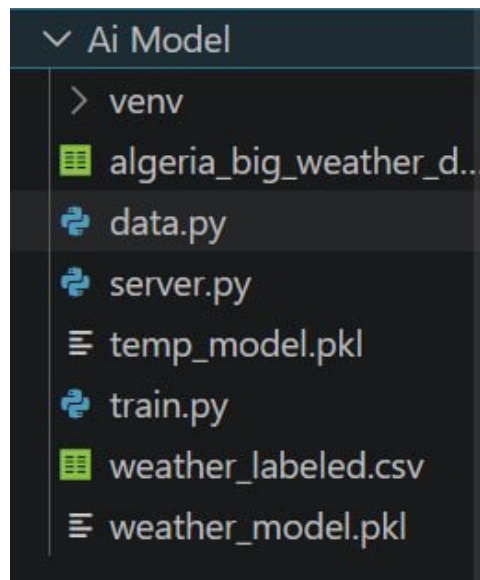


Figure 3.10: AI Model Folder (Prediction System)

Running the Dashboard Locally

To run the dashboard, a local server is launched from inside the **Dashboard folder** using the VS Code terminal. This server hosts the web interface on the local machine via a **localhost** address.

Once the server starts, a local link is generated (for example: `http://localhost:xxxx` or another available port). This link is then copied and opened in a web browser to access the dashboard interface.

At the same time:

- The **ESP32** sends sensor data using the **MQTT protocol**.
- The local server receives and processes incoming data streams.
- The dashboard updates in real time (graphs, analytics, and weather information).

This setup allows the full system workflow:

ESP32 → MQTT → Local Server → Dashboard

to run completely locally without requiring any external cloud service.

Conclusion

This chapter presented the practical implementation of the smart weather station, moving from theoretical concepts to a fully functional system. It detailed the integration of sensors with the ESP32 unit and explained the complete data flow, from acquisition and processing to local storage and transmission via the MQTT protocol.

Furthermore, the chapter described the software environment used (VS Code) and the organization of the project structure into separate modules for the dashboard and the artificial intelligence model. The integration of a Decision Tree-based machine learning model was also highlighted, enabling weather prediction based on real-time sensor data.

Overall, this chapter demonstrates the successful implementation of a complete IoT-based system that combines hardware design, software development, and machine learning techniques in a unified and scalable solution.

Chapter 4

Results

Analysis and Discussion

4.1 Introduction

This section is concerned with presenting the experimental results and evaluating the practical performance of the smart weather station system after completing the design, implementation, and integration phases. The system has been operated under real conditions to collect actual data from various sensors, allowing for the study of its field behavior and verification of its effectiveness.

The presentation is based on analyzing the accuracy of the extracted measurements, the effectiveness of local storage via the SD card, the reliability of data transmission using the encrypted MQTT protocol, in addition to evaluating the efficiency of data display and monitoring through the dashboard.

This section also includes an evaluation of the performance of the adopted machine learning model by testing its ability to predict weather conditions based on the collected real data, allowing for an assessment of the efficiency and reliability of the proposed system in Internet of Things applications for weather monitoring.

4.2 Dashbord Results

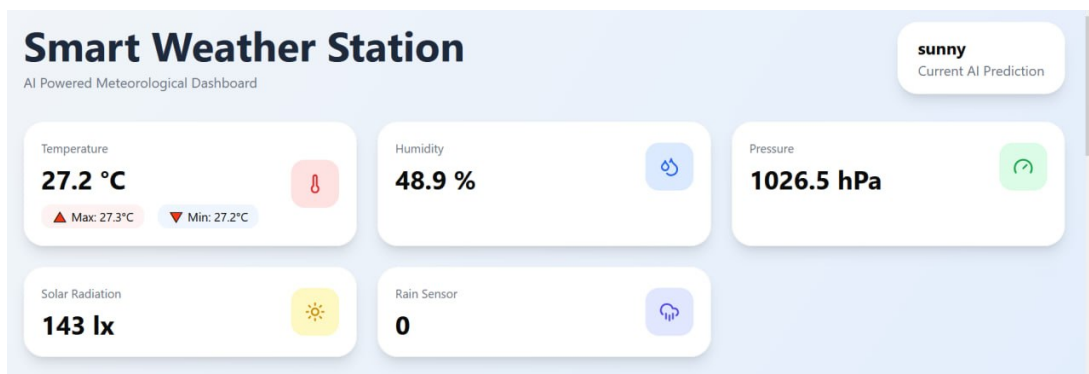


Figure 4.1: Real-time monitoring dashboard displaying environmental sensor data.

Figure (4.1) illustrates the control dashboard interface of the smart weather station, which enables real-time monitoring of environmental data. The interface displays several important variables, including:

- Temperature and humidity
- Atmospheric pressure
- light intensity
- Rain status

The dashboard was designed with a simple and clear interface to facilitate data visualization and allow direct monitoring of weather conditions in real time.

4.3 Temperature and Humidity Analysis

Figure (4.2) illustrates the graphical representation of temperature and humidity variations over time, showing the dynamic behavior of environmental conditions during the testing period:

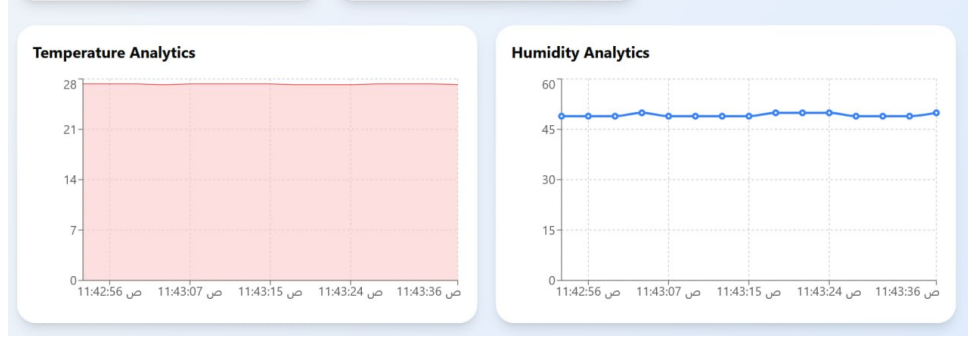


Figure 4.2: Temperature and Humidity Analysis

The temperature curve shows that the recorded values remained relatively stable around 30°C during the testing period, while the humidity values stabilized near 45%.

The analysis of the graphs highlights the following observations:

- The environmental data was continuously updated in real time.
- No significant fluctuations in temperature or humidity values were observed during the monitoring period.

These results indicate the stability of the monitoring system and the reliability of the collected sensor data under the tested environmental conditions.

4.4 Weather Forecasting

The weather prediction module was developed using historical meteorological data collected from environmental sensors. The dataset contains measurements of temperature, humidity, atmospheric pressure, light intensity, and rainfall. Before training, the data were cleaned and organized to improve prediction accuracy.

A Decision Tree machine learning model was trained to learn the relationship between sensor measurements and weather conditions. After the training phase, the model was able to analyze new sensor readings and generate predictions automatically.

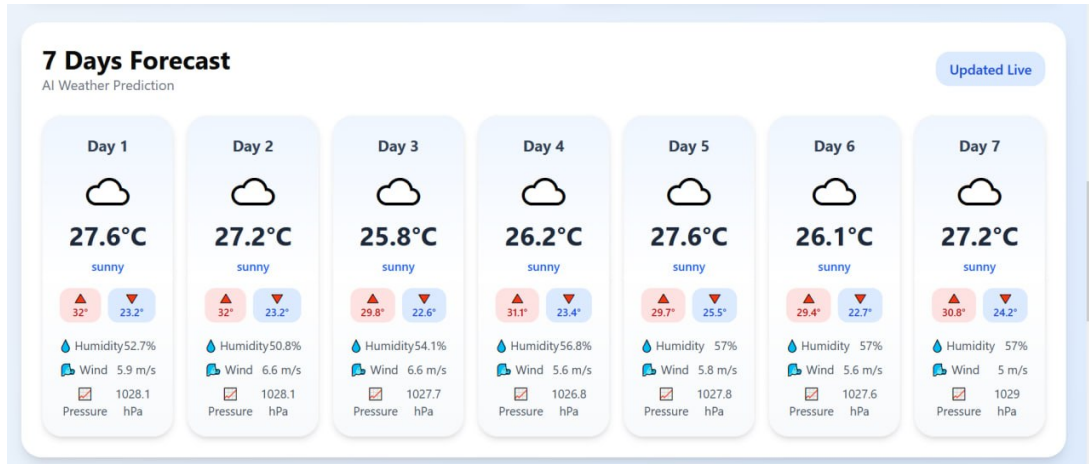


Figure 4.3: Weather Forecast Dashboard

Figure (4.3) illustrates the AI-based weather forecasting system, which predicts weather conditions for the next seven days.

The model relies on historical sensor data to forecast the following parameters:

- Future temperature
- Humidity levels
- Atmospheric pressure
- Weather condition (e.g., sunny, rainy, etc.)

This system provides a predictive view of weather conditions in a smart and relatively accurate manner, enhancing the overall monitoring capabilities of the platform.

4.5 GPS Location Tracking Result

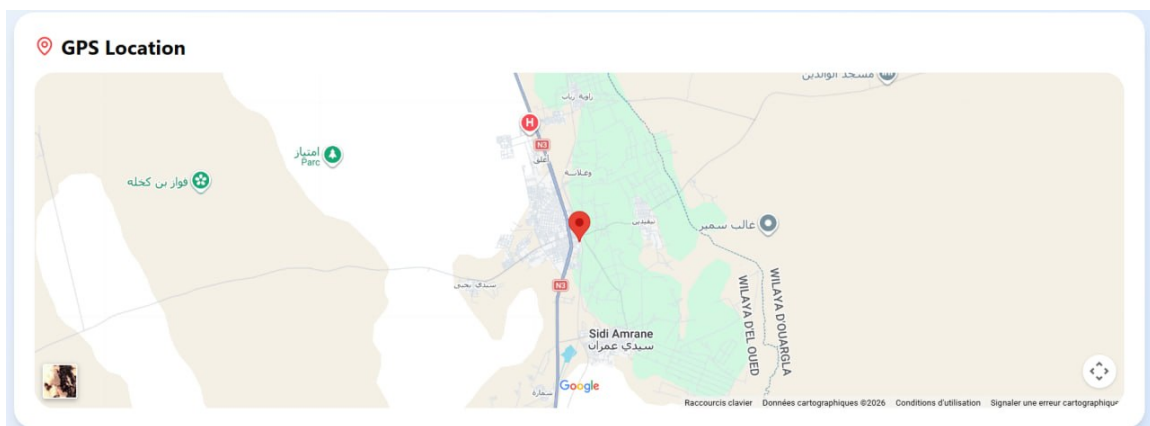


Figure 4.4: GPS location displayed on the dashboard interface

The previous image illustrates the process of displaying the current geographical location of the weather station using the integrated GPS unit with the ESP32 system. Where the

location coordinates (latitude and longitude) are received from the GPS unit and then sent to the monitoring board to be displayed directly on the map.

This feature allows for identifying the actual location of the station and tracking its position in real-time, which is considered important in drone-based applications and environmental monitoring IoT systems.

4.6 Performance Evaluation

The developed system demonstrated several advantages during practical testing. It enables real-time monitoring of environmental data, providing continuous and up-to-date measurements. The system is characterized by a lightweight and portable design, making it suitable for drone-based deployment. It also ensures stable wireless communication through the MQTT protocol, supporting reliable data transmission. In addition, the solution offers a low-cost alternative compared to professional weather monitoring systems while maintaining satisfactory performance. The system is easy to integrate with drone platforms and supports organized and secure data transmission. Furthermore, it includes capabilities for smart weather condition prediction, enhancing its overall functionality and applicability in environmental monitoring.

Flaws and limitations:

Despite the success of the system, several limitations were observed:

- Limited Wi-Fi connectivity compared to LoRa technology.
- Sensor accuracy is more suitable for educational and experimental use rather than professional-grade applications.
- Strong winds and drone movement may affect measurement stability during flight.

4.7 Discussion of Results

The results show that the system is capable of:

- Collecting data from sensors in real-time
- Displaying the data in an organized manner on the dashboard
- Providing weather forecasts using an artificial intelligence model

Although the predictions are generally accurate, minor differences were observed. These variations may be due to natural environmental changes or sensor noise.

Conclusion

This chapter presented the practical results of the Smart Weather Station system, highlighting its real-time monitoring capabilities through a control dashboard interface. The system successfully displays environmental sensor data and analyzes its variations using graphical representations.

In addition, the artificial intelligence-based weather forecasting module was introduced, which predicts future weather conditions based on historical sensor data.

Overall, the obtained results confirm that the system is capable of efficiently collecting, processing, and visualizing environmental data in a structured manner. Furthermore, it provides reliable and acceptable weather predictions, demonstrating the effectiveness of integrating IoT technologies with artificial intelligence for smart environmental monitoring.

General Conclusion

This dissertation presented the design and implementation of a secure IoT-based drone weather monitoring and prediction system. The proposed solution combines environmental sensing, wireless communication, data security, and forecasting capabilities within a single integrated platform.

The developed system successfully collects real-time meteorological data, including temperature, humidity, atmospheric pressure, rainfall, light intensity, and geographical location. These measurements are processed by the ESP32 microcontroller and transmitted securely through the MQTT protocol over TLS to a cloud-based dashboard for visualization and analysis. In addition, local data storage using an SD card ensures data availability even in the event of communication interruptions.

One of the main strengths of the proposed system is its portability. Unlike traditional fixed weather stations, the drone-based platform can operate in different locations and environments, allowing more flexible and extensive environmental monitoring. Furthermore, the integration of weather forecasting capabilities provides users with valuable insights into future weather conditions, enhancing the overall usefulness of the system.

The obtained results demonstrate the effectiveness of combining IoT technologies, secure communication mechanisms, and weather prediction techniques in a single low-cost and scalable solution. The developed platform offers a practical approach for environmental monitoring in remote, difficult-to-access, or dynamically changing areas.

Future Work:

Several improvements can be considered in future developments. Additional sensors, such as wind speed, wind direction, and air quality sensors, could be integrated to provide more comprehensive environmental monitoring. The system may also be deployed on more advanced drones capable of operating at higher altitudes, enabling atmospheric measurements over a wider vertical range. Furthermore, the adoption of long-range communication technologies such as LoRa and the integration of more advanced forecasting models could further enhance the performance, coverage, and accuracy of the proposed system.

Bibliography

- [1] Abduljabbar Hussein Al-Dhafari, *Internet of Things (IoT)*, Graduation Thesis, Ibb University, Yemen, 2020.
- [2] IBM, ما المقصود بالأرصاد الجوية؟, Available: <https://www.ibm.com/sa-ar/think/topics/meteorology>, Accessed: 2024-06-18.
- [3] FAO AGROVOC, علم الأرصاد الجوية, Available at: https://agrovoc.fao.org/browse/agrovoc/en/page/c_4783?clang=ar, Accessed: 2002-12-11.
- [4] Météo Algérie, أجهزة الرصد الجوي, Available at: https://www.meteo.dz/resources/uploadedpics/espace_educatif.pdf.
- [5] Rika Sensor, أدوات الأرصاد الجوية الأساسية التي يجب على كل خبير أرصاد جوية معرفتها, 2025, Available at: (put the URL here if available).
- [6] U.S. Army, *ATP 3-01.81: Counter-Unmanned Aircraft System (C-UAS)*, Washington, DC, 2023. Available at: <https://armypubs.army.mil>.
- [7] AUAV, *Drone Types: Multi-Rotor vs Fixed-Wing vs Single Rotor vs Hybrid VTOL*, Available at: <https://www.auav.com.au/articles/drone-types/>.
- [8] Bas Vergouw, Huub Nagel, Geert Bondt, and Bart Custers, *Drone Technology: Types, Payloads, Applications, Frequency Spectrum Issues and Future Developments*, Chapter 2, 2016. Available online (PDF).
- [9] Wonderful PCB, *What is a Microcontroller?*, Available at: <https://www.wonderfulpcb.com/what-is-a-microcontroller/>.
- [10] HWLibre, شرح الحساس ومميزاته وتطبيقاته - DHT11 Sensor, Available at: <https://ar.hwlibre.com/dht11/>, Accessed: 2026-01-23.
- [11] World Meteorological Organization. *Guide to Instruments and Methods of Observation (WMO-No. 8)*. 2024 Edition, 2024. Available at: <https://wmo.int/guide-instruments-and-methods-observation-wmo-no-8-0>

- [12] Bosch Sensortec, *BMP280 Digital Pressure Sensor*, 2014.
- [13] Wasil Zafar, *Rain Sensor: Resistive & Capacitive Water Detection*.
- [14] A. Study Resource, *Light Dependent Resistor (LDR): Working Principle and Characteristics*. Available: <https://www.studocu.com/row/document/technical-university-of-mombasa/biomedical-engineering/rohini-73227582967/95047253?origin=...>
- [15] Elektronik Kompendium, *Fotowiderstand (LDR) Formeln und Grundlagen*, Available at: <https://www.elektronik-kompendium.de/sites/formeln/fotowiderstand-ldr>.
- [16] Random Nerd Tutorials, *ESP32: Guide for MicroSD Card Module using Arduino IDE*, <https://randomnerdtutorials.com>.
- [17] Espressif Systems, *ESP32 Series Official Product Page*, Available at: <https://www.espressif.com/en/products/socs/esp32>.
- [18] LoRa Alliance, *What is LoRaWAN?*, Available at: <https://loro-alliance.org/about-lorawan-old/>.
- [19] Wikipedia, *LoRa (Long Range Wireless Communication)*, Available at: <https://en.wikipedia.org/wiki/LoRa>.
- [20] European Telecommunications Standards Institute (ETSI), *ETSI TS 136 300 - LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) and E-UTRAN; Overall description; Stage 2*, Available at: https://www.etsi.org/deliver/etsi_ts/136300_136399/136300/18.05.00_60/ts_136300v180500p.p
- [21] Manufacturer Name, *LTE/GSM Module Datasheet*. Available: [link-to-datasheet](#)
- [22] Arduino, *Arduino IDE*. Arduino Documentation. Available: <https://docs.arduino.cc/software/ide/>
- [23] Geek4Arab, *كل ما تحتاج معرفته عن برنامج Visual Studio Code (VS Code)*, Available at: <https://geek4arab.com/vs-code-guide-developers/>.