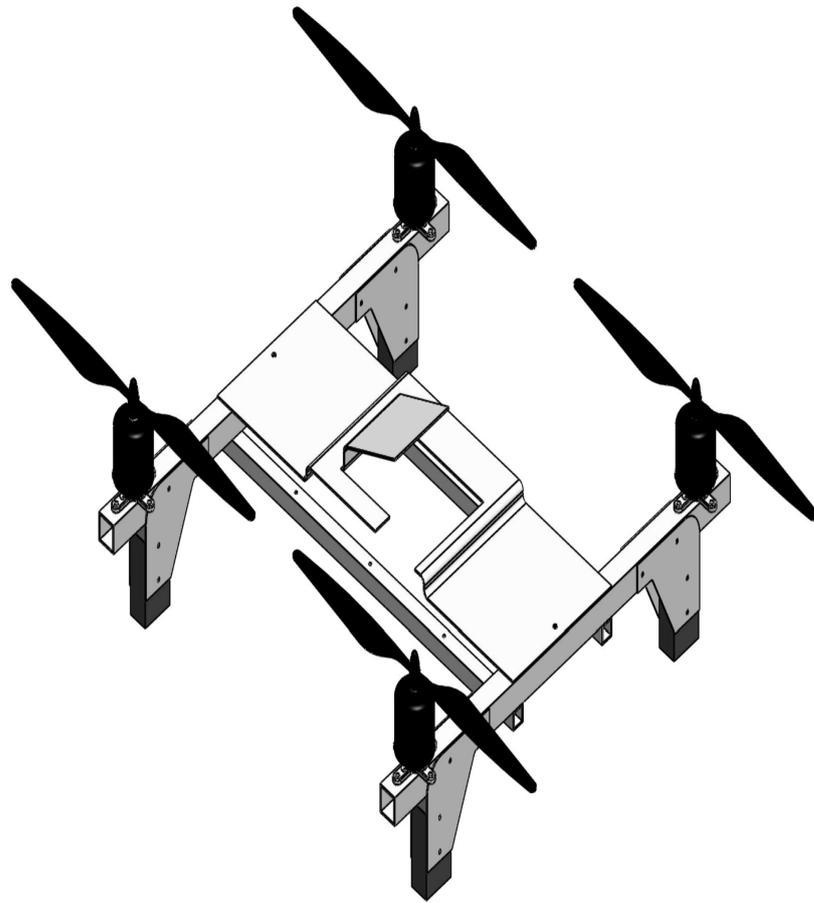


# Development of Secure Framework for Data Harvesting and Monitoring System for Large Scale Farms

A Research Sponsored by Nigerian Communications Commission



Principal Investigation Officer  
Dr OLAKANMI Olufemi Oladayo  
Office 6, Faculty of Technology Building  
University of Ibadan  
+2348068730815  
©NCC September, 2023

## ABSTRACT

Precision farming requires the collection and utilization of data for efficient farm management. This makes the application of sensors and wireless technology indispensable. However, wireless sensors have low transmission coverage, low bandwidth, and low computation power, which affect their ability to process and relay data across large geographical distances. Besides from this, most farms are located outside the network coverage, which limits the effectiveness of the available telecommunication infrastructure for digitizing farm monitoring. Hence, a cybertronic unmanned aerial vehicle (UAV) is needed to cover the required distance in data collection from different farm wireless sensor network (FWSN) gateways and relay it to the server\base station. Therefore, the development of a novel payload-carrying UAV is presented in this report.

On this UAV, is a custom data acquisition system called onboard gateway which collect data from the on-ground FWSNs gateways making the UAV acts as an airborne mobile base station. A separable data aggregation scheme for the FWSN gateway is also proposed to aggregate data from sensors in the same FWSN before transmitting them to the onboard gateway. Thus, reducing the number of data traffics from farms to the server\base station. The result of this work is a uniquely development of UAV system that can semi-autonomously traverse a widely separated farmlands to harvest data from their FWSNs. Performance analysis has been carried out on this system and its subcomponents. Our findings show that the data aggregation and security scheme reduces traffic and increases the turnaround speed of farm data while the UAV increases the data transmission coverage.

## ACKNOWLEDGEMENTS

We are most thankful to Nigerian Communications Commission for not only sponsoring the work but also provides the necessary guidance and directions from the beginning to the end.

The contributions of Mrs Fatima Kyari are also appreciated. She is actually play the role of an indefatigable project manageress who guided the project from the beginning to the end.

We are also indebted to the contributions of all the technical and non-technical members of the Nigerian Communications Commission during the final presentation.

Also, our profound gratituted goes to the University of Ibadan and Department of Electrical and Electronic Engineering for providing enabling environments.

# Contents

<b>Abstract</b>	<b>2</b>
<b>Acknowledgements</b>	<b>3</b>
<b>List of Figures</b>	<b>6</b>
<b>List of Tables</b>	<b>8</b>
<b>1 Introduction</b>	<b>9</b>
1.1 Problem statement . . . . .	10
1.2 Aim and objectives . . . . .	10
1.3 Scope and limitations . . . . .	11
<b>2 Literature review</b>	<b>12</b>
2.1 Unmanned Aerial Vehicle . . . . .	12
2.2 Farm Adhoc Wireless Sensor Network . . . . .	14
2.3 Security and Data Aggregation . . . . .	18
<b>3 Methodology</b>	<b>21</b>
3.1 Methodology for the UAV . . . . .	21
3.1.1 Hardware: Flight Control Circuit . . . . .	24
3.2 Methodology for the Security and Data Aggregation Scheme for FWSN Gateway . . . . .	31
3.2.1 Registration and Key Management . . . . .	33
3.2.2 Data Aggregation . . . . .	34
3.2.3 Data De-aggregation . . . . .	35
3.3 Methodology for the Farm Wireless Sensor Network System . . . . .	36
3.3.1 Architecture and Implementation of FWSN . . . . .	39
3.3.2 Data Acquisition system . . . . .	43
<b>4 Test and Performance Analysis</b>	<b>49</b>
4.1 Tests and Performance Analysis on UAV . . . . .	49

4.1.1	Airworthiness and Hovering Capability of the UAV . . . . .	49
4.1.2	Payload-carrying Test and Analysis . . . . .	49
4.2	Tests and Performance Analysis on FWSN System . . . . .	50
4.2.1	Tests and performance evaluation of FWSN . . . . .	54
4.3	Performance Analysis on Security and Aggregation Scheme . . . . .	56
4.3.1	Performance Evaluation of Data Aggregation . . . . .	56
4.3.2	Results and Discussion . . . . .	57
<b>5</b>	<b>Overview of System, Conclusion, and Futuristic Road-maps</b>	<b>63</b>
5.1	Overview of System . . . . .	63
5.2	Conclusion . . . . .	63
5.3	Futuristic Road-maps . . . . .	65
	<b>References</b>	<b>66</b>

# LIST OF FIGURES

3.1	Arrangement and rotational direction of UAV motor (The direction of each motor spin is manually configured while the its speed is controlled by the ESC by means PWM which in turn generates the trusts – $T_1$ , $T_2$ , $T_3$ , and $T_4$ ) . . . . .	23
3.2	Flight controller circuit diagram . . . . .	25
3.3	Physical implementation of the flight controller (In the center of the controller is the Arduino Nano board, which incorporates the Atmel AT-MEGA328 microcontroller chip. The foamy mounting of the IMU serves to reduce gyroscopic error) . . . . .	26
3.4	Complete UAV System . . . . .	27
3.5	CAD generated 3D layout of the quadcopter airframe . . . . .	28
3.6	CAD generated 2D layouts of the quadcopter airframe . . . . .	29
3.7	CAD generated layouts of the left-to-right arms (15 mm × 15 mm pipe) . . . . .	30
3.8	CAD generated layouts of the front-to-rear connecting chassis (10 mm × 10 mm pipe) . . . . .	30
3.9	CAD generated layouts of the UAV lander (legs) . . . . .	31
3.10	A system model of the FWSNs . . . . .	33
3.11	Circuit diagram describing NRF24L01 radio transceiver module to Arduino board connection . . . . .	38
3.12	Constructed NRF24L01 transceiver module . . . . .	38
3.13	Interconnections and relationships among the different components of FWSN . . . . .	40
3.14	Schematic set-up of the FWSN . . . . .	40
3.15	Schematic circuit of power unit . . . . .	41
3.16	Schematic circuit of sensors’ unit . . . . .	42
3.17	Schematic circuit of sensors’ unit . . . . .	43
3.18	Complete package of the FWSN . . . . .	44
3.19	Practical demonstration of farm data acquisition at high turnaround speed of farm data using the FWSN system . . . . .	47

3.20	Visualization of sensors data from a single node in the FWSN . . .	48
4.1	Testing of the data harvester in a simulated 50 m × 100 m FWSN plantation . . . . .	50
4.2	Caption of UAV’s ascension to high altitude . . . . .	51
4.3	Caption of UAV’s flight at high altitude . . . . .	52
4.4	Plots of attitudes versus time to evaluate the hovering capability of the UAV boarded with various sizes of payloads . . . . .	53
4.5	Comparison in the average round trip time (i.e., time latency) between FWSN and Google cloud with and without data harvester and aggregation for different sensors . . . . .	54
4.6	Display showing the real-time harvested data on the display of the FWSN . . . . .	56
4.7	Comparison of the effects of the number of sensors in FWSN on the security cost . . . . .	60
4.8	Estimated encryption and signature costs for a sensor . . . . .	60
4.9	Comparison of the effects of number of sensors oin FWSN on the aggregation cost . . . . .	61
4.10	Aggregation cost percentages for the developed scheme, and the schemes in [1], [2], [3], and [4] . . . . .	61
4.11	Comparison of access delays . . . . .	62
4.12	Communication overheads . . . . .	62

# LIST OF TABLES

3.1	Parameters for radio transmission of data packages using the NRF24L01 transceiver module . . . . .	37
4.1	Round trip time (ms) of a sensor's 32-byte data from farm to Google cloud with and without data harvester and aggregation . .	54

““

# Chapter 1

## Introduction

Precision farming involves the collection, transmission, processing, and archiving of farm-borne data and information for efficient farm management and decision making. This is based on the effective integration of smart sensors and communication technologies, such as wireless sensor networks (WSNs). The WSNs make it possible to monitor and specifically target each farm activity at a low cost, regardless of the farm's geographic area and location. However, an obstacle to precision agriculture is the limitation of readily available WSN technology, as most farms are located in regions with little or no coverage of the telecommunication network. Although the adoption of fixed base stations (i.e., repeaters) may solve these problems. However, this would increase the overhead cost of the project. Overall, WSNs produce large amounts of sensitive data at regular intervals, which are not only susceptible to cyber attacks but also require high bandwidth channels.

To solve these problems, we developed an edge-embedded unmanned aerial vehicle (UAV) system (with radio\Internet capability), a secure data aggregation scheme for effective data acquisition, and a mobile application for the remote access of the farm data. The UAV acts as an airborne mobile base station with integrated cyber-physical and electronic control capabilities that extends its data communication coverage at a minimal cost, thus essentially making it an airborne cybertronic system. Each farm WSN comprises several sensor nodes that

in turn consist of a radio-enabled microcontroller that aggregates data from heterogeneous sensors. These low-level sensors could be adapted to measure farm variables such as soil fertility\moisture\pH, temperature, humidity, egg counts, or animals' health status and location. The proposed system could allow farm managers to monitor and respond to different situations on the farm in real-time through a remote server\base station with analytical front-end mobile applications to interpret sensor data and make appropriate decisions for actors (i.e., actuation mechanisms) such as an irrigation system or other farm mechanization for effective and precision farming.

## **1.1 Problem statement**

Nigerian youths have developed apathy towards farming believing that farms operations especially monitoring are too tedious and dirty. Therefore, there is need to encourage them through the introduction of a technology that will ease the monitoring of farms. However, the major difficulty lies on the bandwidth and coverage limitness of WSN which is the only network that can be used to harvest the essential farm data. Also, most of the farms are located in a remote places where there is no existing telecommunication infrastructures, thus making the transmission of WSN harvested data to the farmers impossible.

## **1.2 Aim and objectives**

The aim of this research work is to develop a prototype of farm monitoring system capable of collecting the essential farms' data, through an adhoc wireless sensor network, and relaying it to the farmers' mobile devices equipped with a front end mobile application. To achieve this, the following objectives must be carried out to:

1. develop an edge based unmanned aerial vehicle (UAV) that serves as mobile base station to increase the wireless sensor networks coverage.

2. develop a wireless sensor networks that harvest the concentration of farm air pollutants and other farm environmental parameters in real-time.
3. develop a security and aggregation scheme to secure the harvested farm data.
4. develop a mobile application for the end users to access farm parameters.
5. test and evaluate the performance of the entire monitoring system.

### **1.3 Scope and limitations**

The project involves an unmanned aerial vehicle capable of flying over farm wireless sensor network to harvest farms' data collected through the WSN. Onboard this UAV is a custom data acquisition system and a separable data aggregation scheme for the FWSN gateway. The system can only cover a distance of 1km and hovers for 5 minutes before returning to the source. The developed FWSN is using a rechargeable 9V battery can last for a 5 days. Meanwhile, the UAV uses a LIPO battery which must be recharged after every flight of about a kilometer.

# Chapter 2

## Literature review

In this chapter, the existing work on the three major units of the farm monitoring system: UAV, ad-hoc farm wireless sensor network, and data security and aggregation are reviewed under different sub-sections of this chapter. The project takes care of different research gaps. For example, most of the existing projects do not take care of the security and privacy of the harvested data, meanwhile the developed project is able to achieve the security and privacy through the development of security and aggregation scheme purposely for the project. Also, high volume of traffic on the limited bandwidth of wireless sensor network is another research gap which the project take care of by using an aggregation scheme. The aggregation scheme aggregates all the data harvested from all the monitoring sensors and sends them a single traffic, thus reducing the volume and number of times of transmission from the farm wireless sensor network to the farmer. In addition, most of the existing UAVs are highly expensive for local farmers making their usage unrealistic for farm monitoring.

### **2.1 Unmanned Aerial Vehicle**

The UAV is not only capable of providing ideal remote monitoring platforms but is also capable of solving compatibility and reliability problems in precision agriculture. It provides small ground sampling distances, coverage on-demand, and

fast turnaround of information to farmers [5]. This was demonstrated in [6], where an autonomous low-weight and low-cost UAV was developed using an Android-based flight controller. This system also uses the Google Maps application for localization, trajectory planning and navigation.

To enhance UAV performance, Daniel *et al.* [7] proposed two approaches for mathematical modeling of UAV dynamics and kinematics. Their dynamics model was based on Lagrangian mechanics and Denavit-Hartenberg formulas, whereas their kinematic mathematical model was derived from classical mechanics equations. They applied their model to control the one-axis motion of a UAV. In [8], Runfeng and Xi introduced a novel solution for UAV inclination detection, which involved coupling a special robot vision camera with an inclinometer to analyze and formulate intelligent behavior for UAVs. Their solution aided a prototype UAV to achieve and maintain a level attitude for sustained periods (a crucial feat for the deployment of UAVs as airborne mobile base stations). Most of these UAVs have low battery life, poor power-material balance, or are too expensive for economic use, and are therefore not suitable for use as mobile base stations for widely separated FWSNs as required for large-scale precision farming.

Meanwhile, Bayerlein *et al.* in [9] proposed a multi-UAV Path Planning scheme for wireless data harvesting with deep reinforcement learning that can be adapted to various parametric changes of a data harvesting mission, such as the number of deployed UAVs, number, geospatial factors, maximum flight time, etc., without the need to perform expensive recomputations or relearn control policies. With this, several approaches have already been proposed to improve the observability of precision farming by improving data communication in FWSN systems [10, 11, 12, 13, 14, 15, 16]. However, this favors the adoption of multipath as the best approach to developing WSN protocols for reliable data communication in FWSNs. Consistent with this, Olakanmi and Adama [10] proposed a secure multipath routing protocol based on sectorization and best-neighboring node selection models. According to them, this could satisfy the performance

requirements of FWSNs in precision agriculture. In general, multipath solutions are capable of increasing the reliability of FWSN systems, although they could incur high computational and communication costs. Therefore, they are excellent choices for precision farming. In essence, this suggests a dual approach involving the improvement of both the design and model of the UAV and FWSN systems, respectively, for effective data harvesting and precision farming.

## **2.2 Farm Adhoc Wireless Sensor Network**

There is a growing demand for effective air quality monitoring system in farm to provide real-time data on air pollution levels and enable prompt response to potential risks, over the years researchers have taken multiple approaches to air quality monitoring and pollution sensing, using different sensors. Sensor-based air quality monitoring involves the use of specialized sensors interfaced with microcontrollers to detect and measure levels of air pollution in real-time.

Sensor-based air quality monitoring systems use various types of sensors to detect and measure air pollutants, these sensors are often combined with microcontrollers, wireless communication technologies, and data analysis tools to create a comprehensive air quality monitoring system. Researchers have worked on various sensor-based air quality monitoring systems making use of different microcontrollers, and sensors. For example, Al Ahasan et al. in [17] describes an Arduino-based real-time air quality and pollution monitoring system. The hardware comprised an Arduino UNO microcontroller, an MQ-135 air quality sensor capable of detecting carbon dioxide, Benzene, Alcohol, smoke, nitrogen dioxide and ammonia, and an LCD display to display the collected data. The system uses self-developed customised software running on the Arduino Integrated Development Environment (IDE) for real-time data gathering from the hardware system. The system is tested within different air-polluted conditions, using cigarette smoke, mosquito coil burning smoke and motor vehicle smoke, providing the concentration of pollutants in ppm (parts per million) within a

3-meter range at 2-second intervals. The system has the following limitations, it is unable to measure particulate matter, ozone, carbon monoxide, and sulphur dioxide, which are part of the WHO major pollutants, additionally, the collated pollution data is only available on the LCD display of the system, and finally, the alarm system is limited to a visual message on the LCD display of the system.

Similarly, Kaur et al. in [18] developed an air quality monitor based on the Arduino microcontroller that monitors environmental parameters such as temperature and humidity and concentration of pollution including carbon monoxide, carbon dioxide, smoke, alcohol, and LPG. The hardware consists of an Arduino Uno microcontroller, and gas sensors: MQ-135 for carbon dioxide (CO<sub>2</sub>), MQ-2 for Liquefied Petroleum Gas (LPG), alcohol and smoke, MQ-7 for Carbon monoxide (CO), a Zigbee S2 module, a GSM module, and a Bluetooth module. The software consists of C language code running on the Arduino IDE and X-CTU simulation environment serving as the base station receiving information from the Zigbee module. The data in the air are acquired by sensors and processed by the microcontroller, the Zigbee S2 acts as a gateway for the communication between the microcontroller and the base station.

In [19] an IOT-based air pollution monitoring system that monitors air quality over a Web server using the Internet was also developed. The system will monitor the following pollutants: carbon dioxide, smoke, alcohol, benzene, ammonia, nitrogen oxides and Liquefied Petroleum Gases (LPG), and temperature and humidity. The hardware consists of an Arduino Uno microcontroller, gas sensors MQ6 for LPG and MQ135 for the other pollutants to obtain the concentration of the pollutants in the air, an LM35 temperature sensor, SY-HS-220 humidity sensor, an ESP8266 Wi-Fi module to give the system access to the internet, a GSM module to enable communication with a GSM system and a buzzer to signal when the concentration of pollutants goes beyond a certain level. The software would consist of code written in C running on the Arduino IDE. The system would display the concentration of the pollutants in parts per million (ppm), along with a

message on the LCD and web page indicating if the air is fresh or not along, and in the case of poor air quality the buzzer begins beeping and an alert message is sent to a phone through GSM.

Also Jha in [20] presented an advanced real-time air quality reporting system supported by Internet of Things (IoT) architecture, that measures the concentration of PM2.5, carbon monoxide and nitrogen dioxide in the air and integrates the real-time data into a custom designed mobile application. The hardware consists of an Arduino Uno microcontroller, gas sensors: MQ135, MQ7 and dust sensor GP2Y1010AU0F, a Wi-Fi module ESP8266 and a buzzer.

Due to Raspberry Pi versatility, Kumar and Jasuja in [21] presented a real-time standalone air quality monitoring system using ARM based microcomputer Raspberry Pi and a sensor to measure PM 2.5, carbon monoxide, carbon dioxide, temperature, humidity, and air pressure. The hardware consists of a raspberry pi as the major control node, the pollutants sensors including DSM501A used for measuring particulate matter, MHT22 for temperature and humidity, BMP180 for pressure, MQ9 and MQ135 are connected to an Arduino Uno board which is interfaced with the Raspberry pi through a USB cable. The device however makes no provision for an alarm to alert to high pollution levels. Similarly, a low-cost pollution control and air quality monitoring system using Raspberry Pi for Internet of Things was also developed in [22]. The hardware consists of a Raspberry Pi single board computer, used as a sensor node and gateway node, an ESP8266 Wi-fi module for adding Wi-Fi functionality to the Pi, an MQ5 gas sensor, a temperature and humidity sensor, a soil moisture sensor and a GrovePi device to interface the sensors with the main board controller. The system is proposed to be installed in several places of preferably higher population density to observe the parameters over a period, this data will be highly useful for understanding the levels of pollution in a place.

Meanwhile, Marinov et al. in [23] presented Outsense using PIC18F87K22 for measuring relevant environmental parameters, based on a scalable sensor array

with integrated amperometric and infrared gas sensors. The hardware consists of microcontroller PIC18F87K22, a GPS module based on the MTK3339 chipset, ESP8266 chip as a Wi-Fi module, gas sensors: COAF (for carbon monoxide), NO<sub>2</sub>-A42F (for nitrogen dioxide), OX-A421 (for ozone) and O<sub>2</sub>-A2 (for oxygen), and sensors for physical parameters: MPXA6115A (for pressure), HDC1050 (for humidity), and MCP9808 (for temperature). In line with this, Gunawan et al in [24] designed and implemented a portable outdoor air quality measurement system using Arduino capable of measuring the concentration of carbon monoxide, groundlevel ozone, and particulate matter (PM<sub>10</sub> and PM<sub>2.5</sub>) in the air and converting the readings to Air Pollutant Index (API) value. The hardware consists of an Arduino Uno microcontroller, MQ-9 CO sensor, MQ-131 O<sub>3</sub> sensor, GP2Y1010AUF PM<sub>10</sub> sensor, Shinyei PPD42NS PM<sub>2.5</sub> sensor, 16x2 LCD display and an SD card reader for data logging. Also, Abiduzzaman et al. in [25] developed a real-time outdoor air quality monitoring system based on the ESP32 microcontroller that measures the concentration of carbon monoxide (CO) and nitrogen dioxide (NO<sub>2</sub>) in the air and enables access to the Air Pollution Index (API) and location data through a mobile application. The hardware consists of the gas sensors: MQ-7 for CO and MQ-135 for NO<sub>2</sub> and a GPS module which are connected to an ESP32 microprocessor.

Meanwhile, it was observed that most of these work have limitations that make the accuracy of sensors used to vary widely, and they are not reliable for monitoring air pollutants in a critical facilities such as farms. Also, the sensors node in most of these projects have a limited lifespan due to the battery life. Apart from these, most of the air quality monitors have a limited range and may not be able to cover large areas. Therefore, unsuitable for monitoring the air quality in large farmland.

## 2.3 Security and Data Aggregation

Several cryptographic based security solutions had been developed to protect the data collected from the IoT and WSNs to reduce the volume of traffic on the network. Examples of these are in [26, 27, 28, 4, 29]. Lu et al. [28] proposed a lightweight privacy-preserving data aggregation scheme for fog-based IoT. They used Paillier homomorphic encryption, Chinese remainder theorem, and one-way hash chain techniques to aggregate IoT devices' data, and filter injected data. The scheme is capable of aggregating data from various heterogeneous IoT devices, however, the scheme is malleable, susceptible to unauthorized aggregation, and has a high computational cost and limited aggregation functionalities. Similarly, Oladayo and Adedamola [30] proposed an efficient framework and aggregation scheme to secure and manage data in the IoT. Their framework consists of a cascaded look ahead fog devices, which effectively solve problems associated with data locality and information management in IoT networks. The scheme frequently keeps used data in the closer and computationally cheaper fog levels while rarely used data are pushed to the cloud data center. The scheme is capable of aggregating different devices' data, however, it could not detect false data injection during in and out-of-network accesses.

To ensure confidentiality in IoT or WSNs networks, an effective lightweight mutual authentication is needed. Meanwhile, a few of the existing authentication schemes rely on per-packet signature and per-signature verification which increases computation and communication costs making them unsuitable for IoT. To address this, Li et al. [31] proposed an authentication scheme for data aggregation in the smart grid through signature aggregation, batch verification, and signature amortization to lower the communication overhead, reduce numbers of signing and verification operations, and provide fault tolerance. However, their work only focused on authentication, it could not reduce the volume of traffic on the network.

Meanwhile, Tan et al. [27] also proposed an end-to-end security and aggrega-

tion scheme using elliptic curve based chameleon hashing to provide data integrity and authenticity for smart grid networks. Their model includes a concentrator which aggregates the smart meter readings and generates a chameleon hash value of the aggregated metering. This is sent to the Meter Data Management System (MDMS) for verification. The smart meter computes and sends a commitment value to the MDMS in a way that the chameleon hash value is equivalent to the previous hash value sent by the concentrator. However, the scheme can only aggregate the monolithic metering of a consumer at a different time interval. Therefore, can not be used to aggregate the metering of different consumers. Li et. al. [32] also presented a distributed incremental data aggregation approach for smart meter in the smart grid networks. They adopted an aggregation tree and homomorphic encryption to secure the data along the route in which data aggregation is performed by all the smart meters along the routing path. However, data sniffing attack can be successfully launched on the scheme. That is, an attacker may masquerade itself as the neighbor of the targeted smart meter to sniff any out-going data from such a meter. Also, their approach cannot be adopted for the client-server model which is being used by most of the IoT networks. Another aggregation scheme was proposed in [33] to facilitate bidirectional demand-response communication between individual users and the energy provider without compromising the users' private information. To achieve this, the authors developed an efficient and privacy-preserving power requirement and distribution aggregation scheme (EPPRD) based on a hierarchical communication architecture. However, the communication overhead was high, therefore unsuitable for IoT.

Wireless sensor network (WSN) an integral of IoT exhibits the same characteristics with IoT especially being resource constraint. To reduce traffic in WSN for multi-application environments, Lin et al. [2] proposed aggregation scheme for multi-application environments. The scheme adopted homomorphic public encryption to aggregate ciphertext and count to provide secure counting for a

single application case. In their scheme, the base station obtains the aggregated result and its count which is the number of the nodes involved in the aggregation. The count is used to detect unauthorized aggregations launched by a malicious. Similarly, Shim et al. [3] proposed an aggregation scheme based on additive homomorphic encryption, an identity-based signature, and a batch verification technique with an algorithm for filtering injected false data. They adopted additive homomorphic encryption such that only a BS can decrypt encrypted data aggregated by the cluster head received from member nodes in each cluster. Also, Chien et al. [1] proposed recoverable concealed data for WSN for homogeneous and heterogeneous WSN. In their scheme, each signature and data are mapped to group elements before being aggregated.

A security solution for preventing unauthorized data injection was proposed in [4]. Zhong et al. [4] proposed a scheme to solve the weaknesses of a homomorphic encryption-based aggregation scheme. They proposed a secure data aggregation scheme by combining the homomorphic encryption technique with signature to prevent in-of-network false data injection, and ensure authorized aggregation. The scheme allows the base station to identify the origin and validity of the messages received so that the base station can recover the original sensing data, and perform arbitrary aggregation operations. However, the adoption of homomorphic encryption gives it high computational cost.

Asides all these aforementioned drawbacks in some of the existing aggregating schemes, only a few of them exhibit recoverability, an essential characteristic of a good aggregation scheme. Also, all of these schemes except [1] and [2] could only perform aggregation among homogeneous nodes. Therefore, there is a need for a recoverable aggregation scheme for heterogeneous IoT devices with an effective authentication technique at a low cost.

# Chapter 3

## Methodology

In this chapter, the methodologies adopted for the development of the UAV, farm WSNs, and the security and aggregation scheme are discussed as follows:

### 3.1 Methodology for the UAV

The present work builds on the work reviewed in Section 2.1, but with great emphasis on the physical engineering of the UAV system and its subcomponents, such as the flight controller and the on-board data acquisition system. Unlike related work, the present work involves the design of a completely new UAV, a type of UAV that flies with four rotor-driven propellers. The reason for this is that many commercially available UAV systems are general-purpose UAVs that are not fully optimized to perform routine flight operations, especially in semi-autonomous mode, over farmland. This technical limitation is evident in [6, 34], where factors such as battery life, payload carrying capacity, and structural integrity of the aircraft material limited the overall performance of the UAV system. As a consequence, the present work draws on several technical know-how that includes the expertise of a mechanical designer, material scientist, cybernetician, computer scientist, and electrical engineer. More importantly, the integration and control of knowledge systems were very crucial in harmonizing the various expertise. As a result, the UAV prototype developed, along with external platforms

such as the base station and the FWSNs that communicate with it, constitutes a complex multi-technical system, in which moving parts are activated by electronic devices under software verification and control.

Computer-aided design (CAD) approaches were crucial during the implementation of this project. This involves the use of software platforms such as SolidWorks (for the design and simulation of mechanical subsystems), KiCAD (for the design of custom circuits), and the Python language for data analysis. The FWSN aspect of this work involves the formulation of various data communication and security schemes for efficient and secure signaling of sensory and control data. Details of these developments are sequentially discussed within three major sections of this chapter: the UAV system in Section 3.1, the data aggregation scheme for FWSN gateway in Section 3.2, and the farm wireless sensor network system in Section 3.3. These entail relevant analysis of the airworthiness, dynamic stability, and geospatial telemetric efficiency of the FWSN system\data aggregation scheme.

$$Y(s) = D(s)U(s) = D(s)C(s)E(s), \quad (3.1)$$

$$E(s) = Y_R(s) - Y_M(s) = Y_R(s) - M(s)Y(s). \quad (3.2)$$

By substituting Equations 3.2 into 3.1, we derive the transfer function of the entire control system as:

$$H(s) = \frac{Y(s)}{Y_R(s)} = \frac{D(s)C(s)}{1 + C(s)D(s)M(s)}, \quad (3.3)$$

where  $D(s)$  is the transfer function of the ESC block that controls a brushless DC motor of the UAV. For the automatic orientation of the UAV in space, the IMU, with the transfer function  $M(s)$ , measures the actual attitude of the UAV, which is then compared with the desired equivalent. The flight controller uses subroutines to implement altitude control. This involves the generation of PWM signals to alter the speed of the motors through the ESC (represented by  $D(s)$ ),

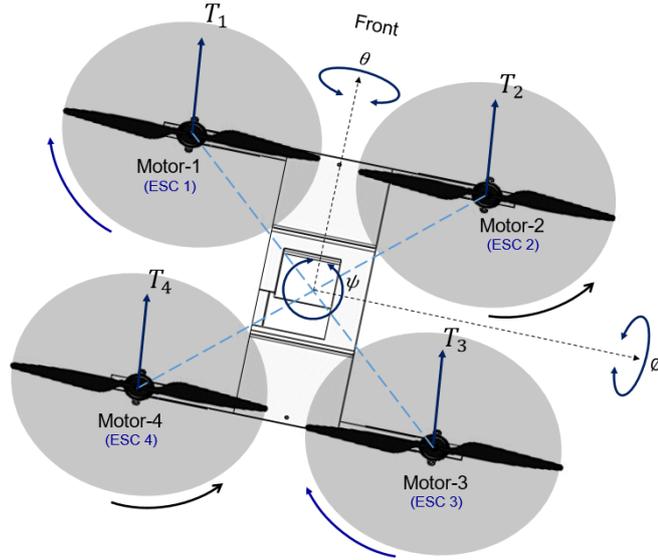


Figure 3.1: Arrangement and rotational direction of UAV motor (The direction of each motor spin is manually configured while the its speed is controlled by the ESC by means PWM which in turn generates the trusts –  $T_1$ ,  $T_2$ ,  $T_3$ , and  $T_4$ )

which in turn alters the orientation of the UAV. This described how the flight controller regulates the PWM signals fed to the four ESCs in connection with their respective motors to control the orientation of the UAV. Figure 3.1 conceptualizes the motor arrangement and rotation direction of the UAV. Motor 1 and motor 3 rotate in the clockwise direction, while motor 2 and motor 4 rotate in the counterclockwise direction. The motor 1–2 pair and motor 3–4 pair control the pitch angle of the UAV, the motor 1–3 pair and the motor 2–4 pair control the yaw angle of the UAV, while the motor 1–3 pair and the motor 2–4 pair control the yaw of the UAV.

As the flight controller repeats these loops, the pitch, roll, and yaw angle errors diminish as the UAV is gradually oriented to the desired attitude in the 3D space. When the desired pitch, roll, and yaw angles of the desired orientation were programmed in the flight controller microcontroller to all be zero degrees, the flight controller PID algorithms made alterations to the UAV orientation until it attained a completely horizontal hovering profile in 3D space. Also, this model is sufficiently described in algorithm 1. By varying the desired values for pitch, roll, and yaw angles, we can effectively control the flight trajectory of our UAV in any direction in 3D space. By electronically varying the desired attitude,  $Y_R(s)$  of

the UAV, we can control its movement and direction during flight operation. The hardware technology with which this is implemented is discussed in the following.

---

**Algorithm 1** PID control algorithm for all Euler angles of UAV attitude (Note:  $U \rightarrow$  PID output,  $\tau \rightarrow$  throttle)

---

```

1: function PITCH_CONTROL
2:    $e_\phi(k) = \phi_{desire} - \phi_{actual}$ 
3:    $P_\phi(k) = Kp_\phi * e_\phi(k)$ 
4:    $I_\phi(k) = I_\phi(k-1) + Ki_\phi * e_\phi(k)$ 
5:    $D_\phi(k) = Kd_\phi * (e_\phi(k) - e_\phi(k-1))$ 
6:    $U_\phi = P_\phi(k) + I_\phi(k) + D_\phi(k)$ 
7: end function
8: function ROLL_CONTROL
9:    $e_\theta(k) = \theta_{desire} - \theta_{actual}$ 
10:   $P_\theta(k) = Kp_\theta * e_\theta(k)$ 
11:   $I_\theta(k) = I_\theta(k-1) + Ki_\theta * e_\theta(k)$ 
12:   $D_\theta(k) = Kd_\theta * (e_\theta(k) - e_\theta(k-1))$ 
13:   $U_\theta = P_\theta(k) + I_\theta(k) + D_\theta(k)$ 
14: end function
15: function YAW_CONTROL
16:   $e_\psi(k) = \psi_{desire} - \psi_{actual}$ 
17:   $P_\psi(k) = Kp_\psi * e_\psi(k)$ 
18:   $I_\psi(k) = I_\psi(k-1) + Ki_\psi * e_\psi(k)$ 
19:   $D_\psi(k) = Kd_\psi * (e_\psi(k) - e_\psi(k-1))$ 
20:   $U_\psi = P_\psi(k) + I_\psi(k) + D_\psi(k)$ 
21: end function
22: function ESC_SIGNALS
23:   $PWM_1 = \tau - U_\phi - U_\theta + U_\psi$ 
24:   $PWM_2 = \tau - U_\phi + U_\theta - U_\psi$ 
25:   $PWM_3 = \tau + U_\phi + U_\theta + U_\psi$ 
26:   $PWM_4 = \tau + U_\phi - U_\theta - U_\psi$ 
27: end function

```

---

### 3.1.1 Hardware: Flight Control Circuit

The control hardware of our UAV is developed using an Atmel ATMEGA328 microcontroller (i.e., Arduino Nano) and contains other components that include the MPU6050-IMU, screw-terminal connection to four electronic speed controller (ESC), radiofrequency receiver (RF) (with five control channels) and any serial device. This microcontroller board serves as the central control and processing unit (CCPU) of the UAV flight system. The IMU contains a three-axis gyroscope and an accelerometer sensor on a single chip. This is used to measure the attitude

(that is, Euler angles:  $\phi$ ,  $\theta$ , and  $\psi$ ) of the UAV in flight. The integration of all these components constitutes the flight control system (or controller), the schematics and circuits of which are shown in Figure 3.2. Here, all the peripherals are grounded by connecting the GND (Ground) pins of all mentioned parts to the GND pin of the Arduino board in the flight controller.

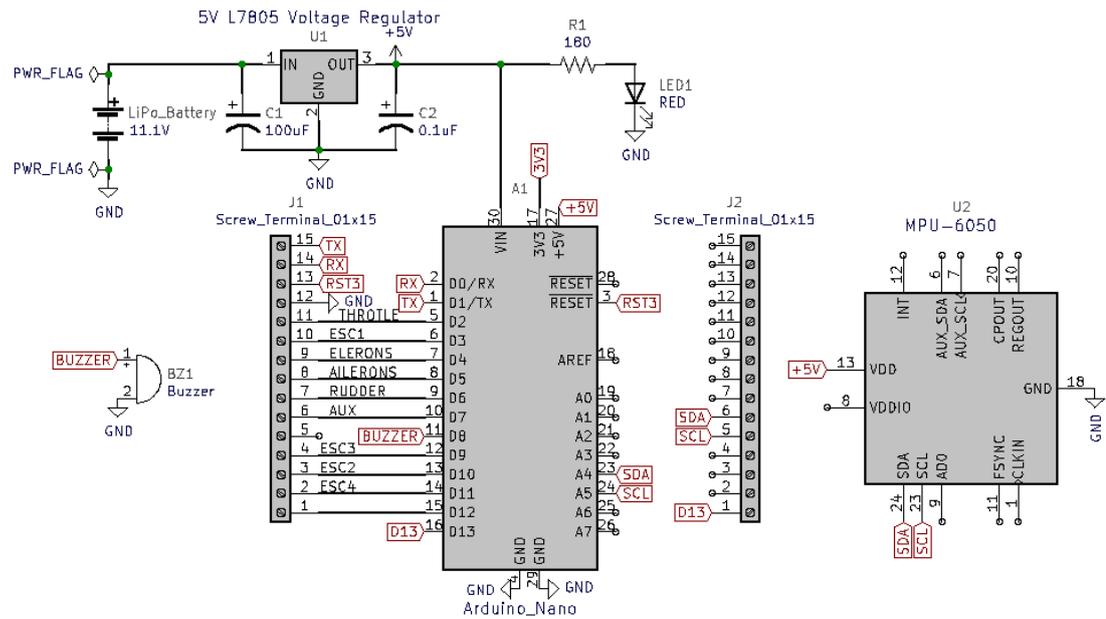


Figure 3.2: Flight controller circuit diagram

The physical implementation of the actual flight controller is shown in Figure 3.3. Due to structural and cost considerations, this was implemented on a transparent plastic board and includes an additional  $1 \times 15$  screw terminal to accommodate the remaining navigational sensors. The plastic mounting board provides basal insulation to the circuit components while the IMU is fastened through the foam to provide a damping effect on the IMU, thus reducing noisy signals resulting from the rotor-dynamic vibration of the airframe during flight; thus, preserving the gyroscopic integrity of the IMU. The RF receiver\additional sensors (such as the GPS and other aforementioned sensors) can be attached to the relevant screw terminals as external peripherals. Our flight algorithms were encoded in the Arduino compatible C/C++ programming language and embedded into the developed flight controller for controlling the flight dynamics of the UAV. Using this electro-software device and the selected duralumin alloy, a proto-

type of the UAV system was manufactured. The complete UAV system is shown in Figure 3.4. Embedded in the top cover of this UAV is the data acquisition system, the function of which is described in Section 3.3.

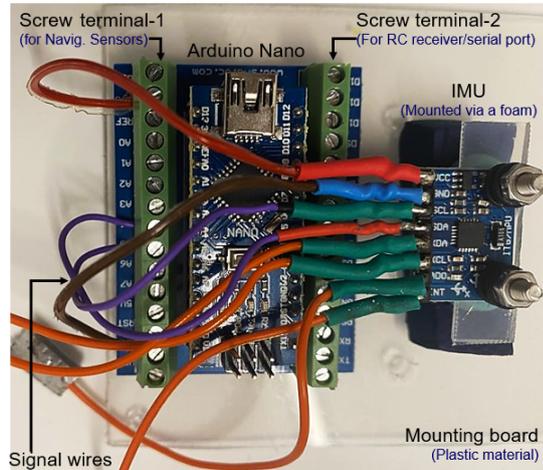


Figure 3.3: Physical implementation of the flight controller (In the center of the controller is the Arduino Nano board, which incorporates the Atmel ATMEGA328 microcontroller chip. The foamy mounting of the IMU serves to reduce gyroscopic error)

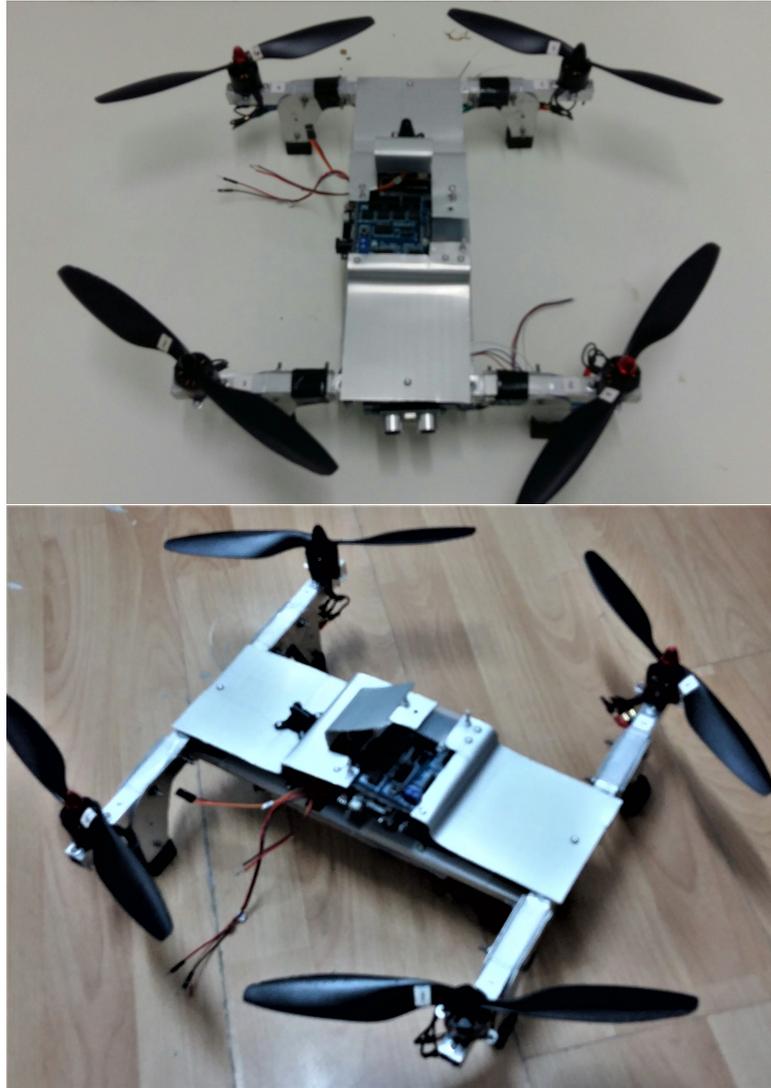


Figure 3.4: Complete UAV system (attached to the top cover of the UAV is the data acquisition system. This aluminum cover also serves to protect the flight controller and its peripherals underneath it)

Supplementary information on UAV design geometries and test flights are shown in Figures [4.2](#), [4.3](#), [3.7](#), [3.8](#), and [3.9](#).

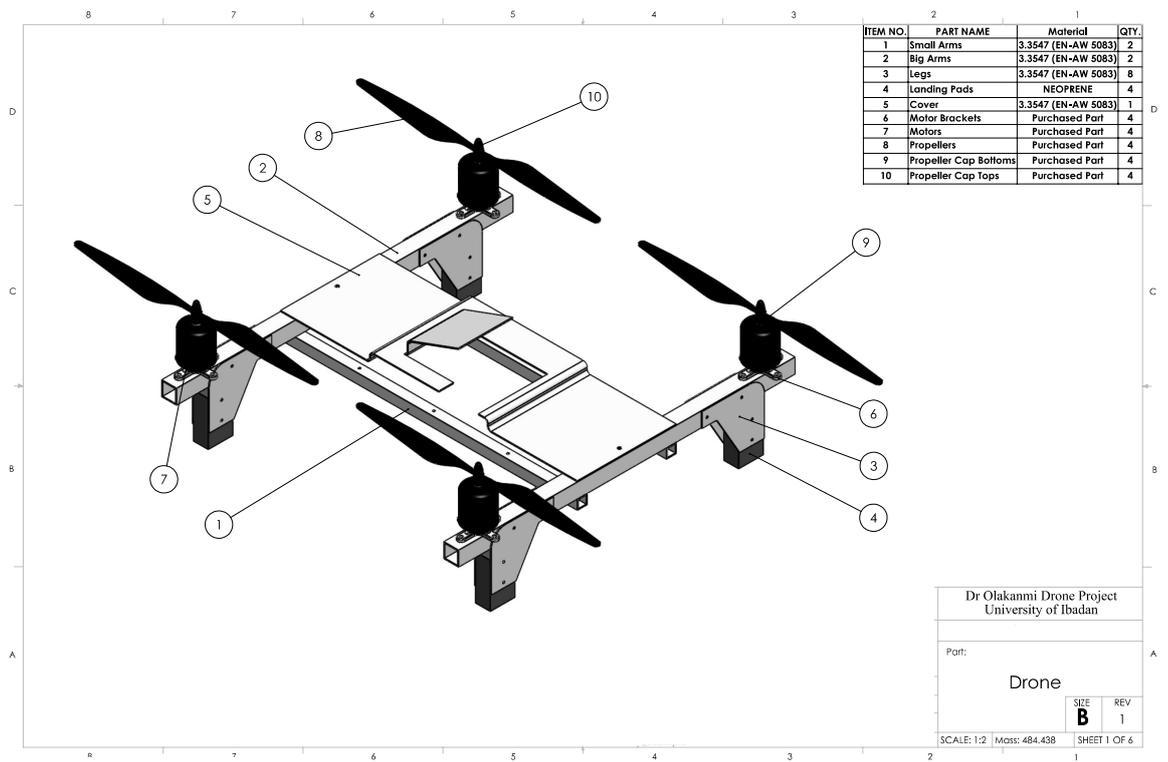


Figure 3.5: CAD generated 3D layout of the quadcopter airframe

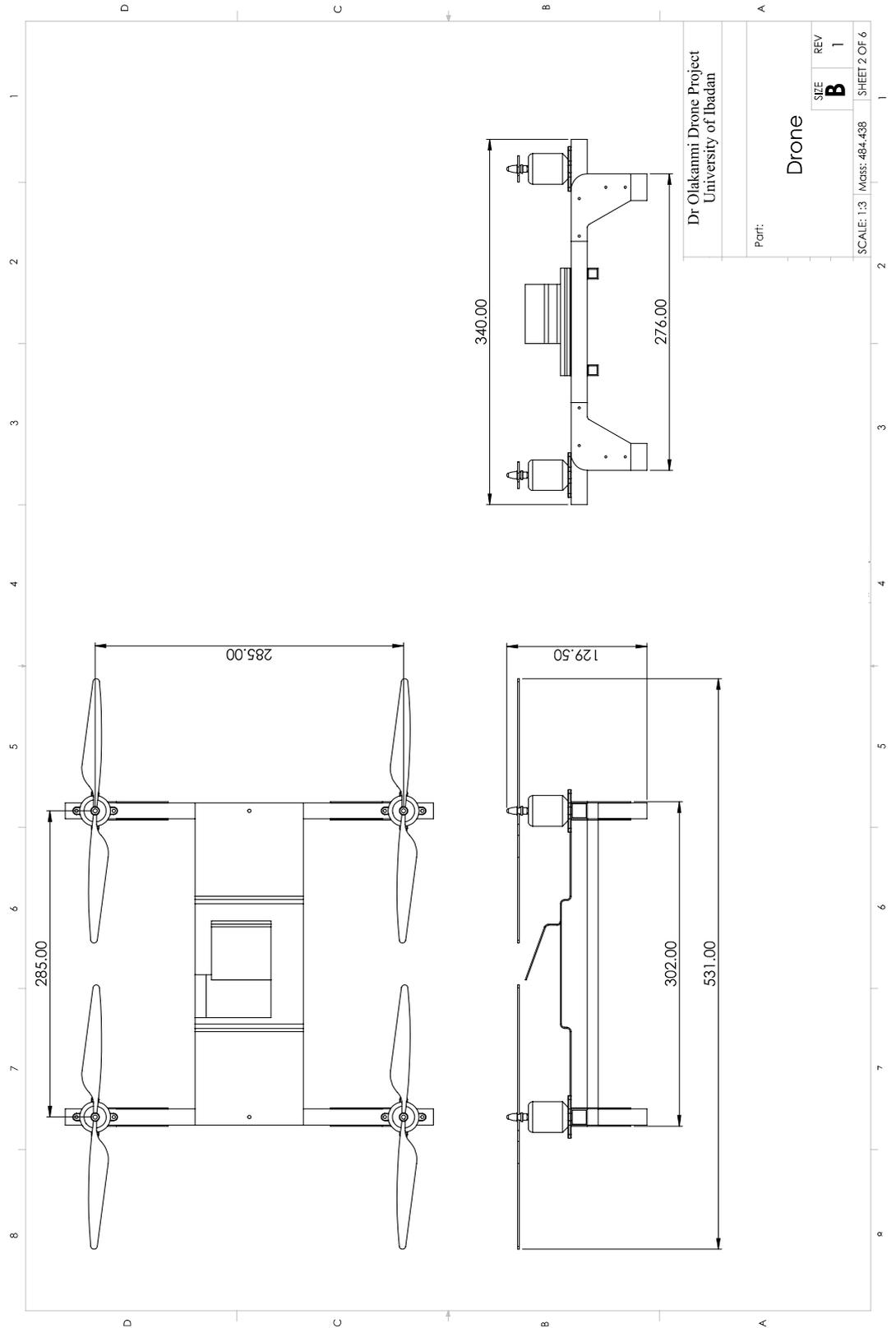


Figure 3.6: CAD generated 2D layouts of the quadcopter airframe

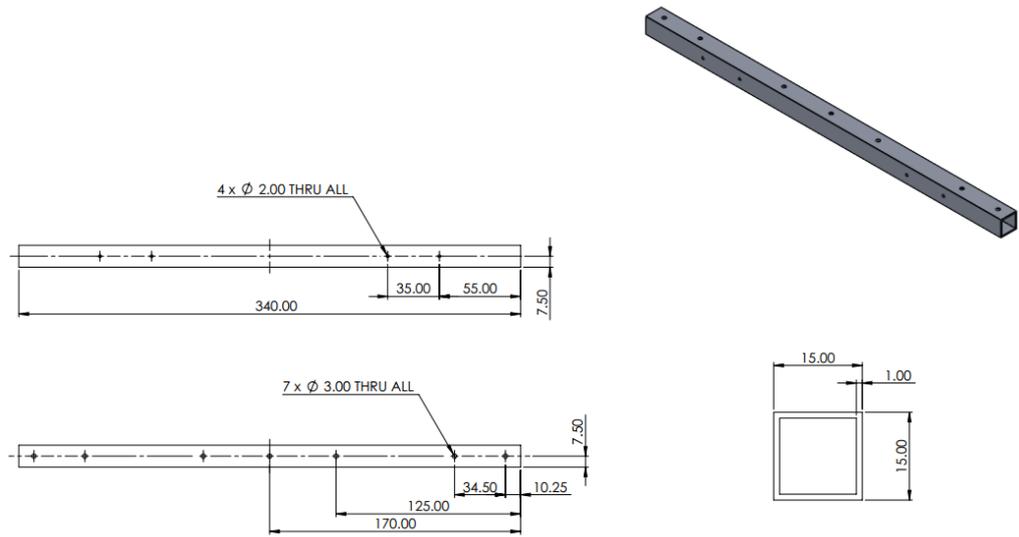


Figure 3.7: CAD generated layouts of the left-to-right arms (15 mm  $\times$  15 mm pipe)

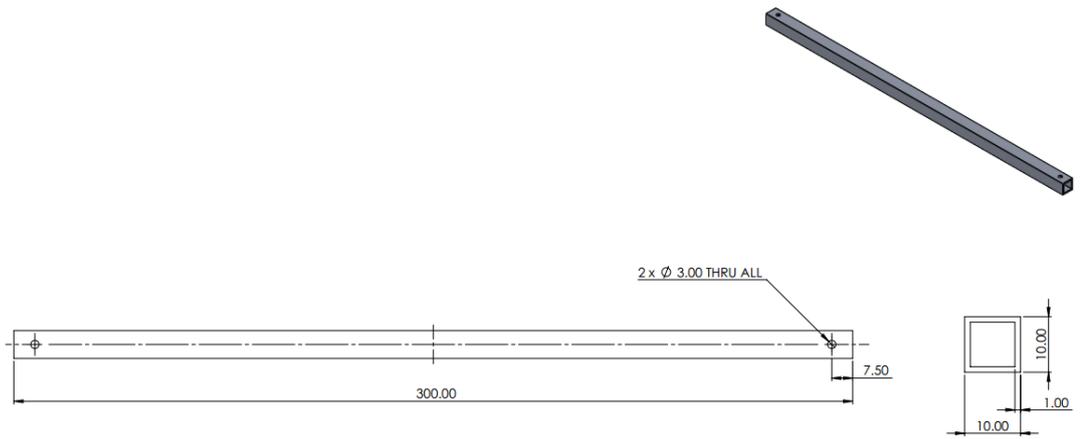


Figure 3.8: CAD generated layouts of the front-to-rear connecting chassis (10 mm  $\times$  10 mm pipe)

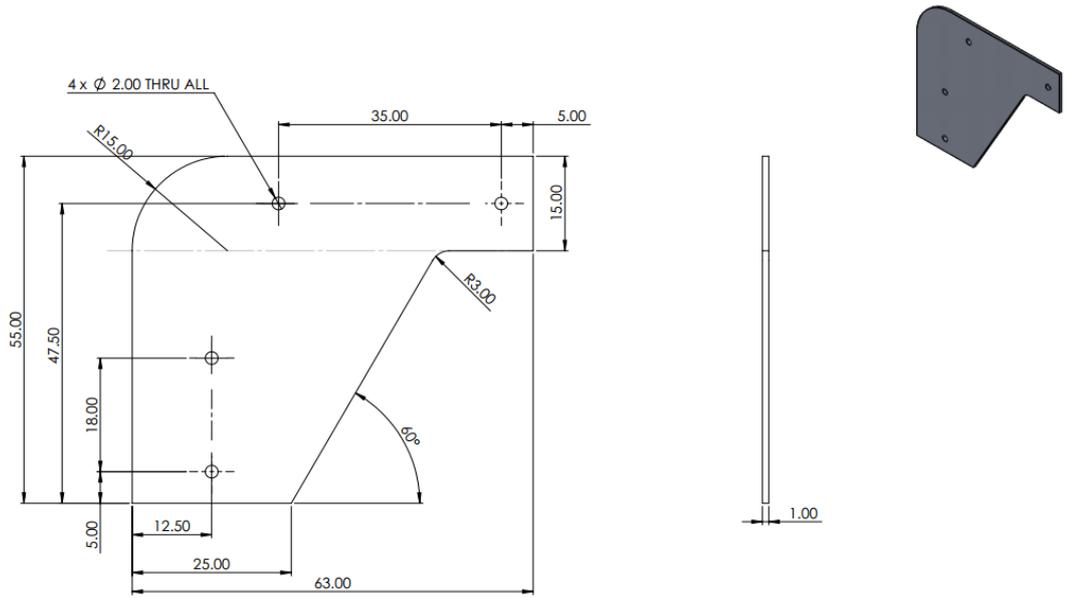


Figure 3.9: CAD generated layouts of the UAV lander (legs)

### 3.2 Methodology for the Security and Data Aggregation Scheme for FWSN Gateway

As shown in Figure 3.10, security and aggregation scheme model is grouped into different FWSNs. The core of each FWSN is the gateway which consists of a controller with a transceiver. The wireless transceiver serves as an access point through which data enter or exit a FWSN. Each gateway aggregates the readings of its sensors and sends them to the UAV once it reaches the range ( $\approx 100$  m) during data collection. The UAV is equipped with a wireless transceiver to collect data from a specific FWSN, which consists of several sensor nodes. The UAV semi-autonomously flies to the farm manager server(FMS) or mobile device to relay the harvested data, through radio telemetry. In this section, we described the methodology of the developed data aggregation scheme that is embedded in the FWSN gateway. The scheme not only secure the harvested data but also reduces the number of transmission sessions between the gateway and the UAV. The scheme takes care of different research gaps in the existing system. For example, most of the existing systems do not take care of the security and privacy of the harvested data. Meanwhile, the developed project, through the security and

aggregation scheme, is able to achieve the security and privacy of the harvested data and farm. It also reduces volume of traffic on the limited bandwidth of wireless sensor network through the aggregation scheme. The aggregation scheme aggregates all the data harvested from all the monitoring sensors and sends them a single traffic, thus reducing the volume and number of times of transmission from the farm wireless sensor network to the farmer.

The following cryptography primitives are used in the proposed scheme.

1. **Hash Function:** A key based hash function  $H(.)$  is a mathematical operator that maps a message  $M$  of arbitrary length to strings of fixed length bits ( $n$  bits) to give strings of fixed length bits ( $l$  bits). That is,  $H: \{0, 1\}^* \times \{0, 1\}^n \rightarrow \{0, 1\}^l$ . It is used to generate a message digest to confirm that the content of the message has not changed. A hash function is easy to compute, has public functionality, pre-image, and collision resistance.
  
2. **Elliptic Curve Cryptography:** Elliptic curve is a mathematical group that can be represented as:  $E_q(a, b) : y^2 = x^3 + ax + b \pmod{p}$  over a prime finite field  $F_q$ , where  $q > 3$ ,  $a, b \in F_q$ , and  $4a^3 + 27b^2 \pmod{p} \neq 0$ . It has the following properties; with a group member  $P$  and  $\kappa \in Z_p^*$ , the point  $Q = \kappa_1 * P$  where  $Q$  is a point on the elliptic curve, such that  $Q$  can be computed given  $\kappa_1$  and  $P$  (scalar multiplication), while finding  $\kappa_1$  given  $Q$  and  $P$  is infeasible. This infeasibility is known as the Elliptic Curve Discrete Logarithm Problem. Also, the addition of any points of  $E_q(a, b)$  with  $R \in Z_p^*$  gives a value that is not an element of  $E_q(a, b)$  but an element of  $Z^*$ .

The formulated security and data aggregation scheme contains three algorithms; registration and key management, separable data aggregation, and de-aggregation algorithms.

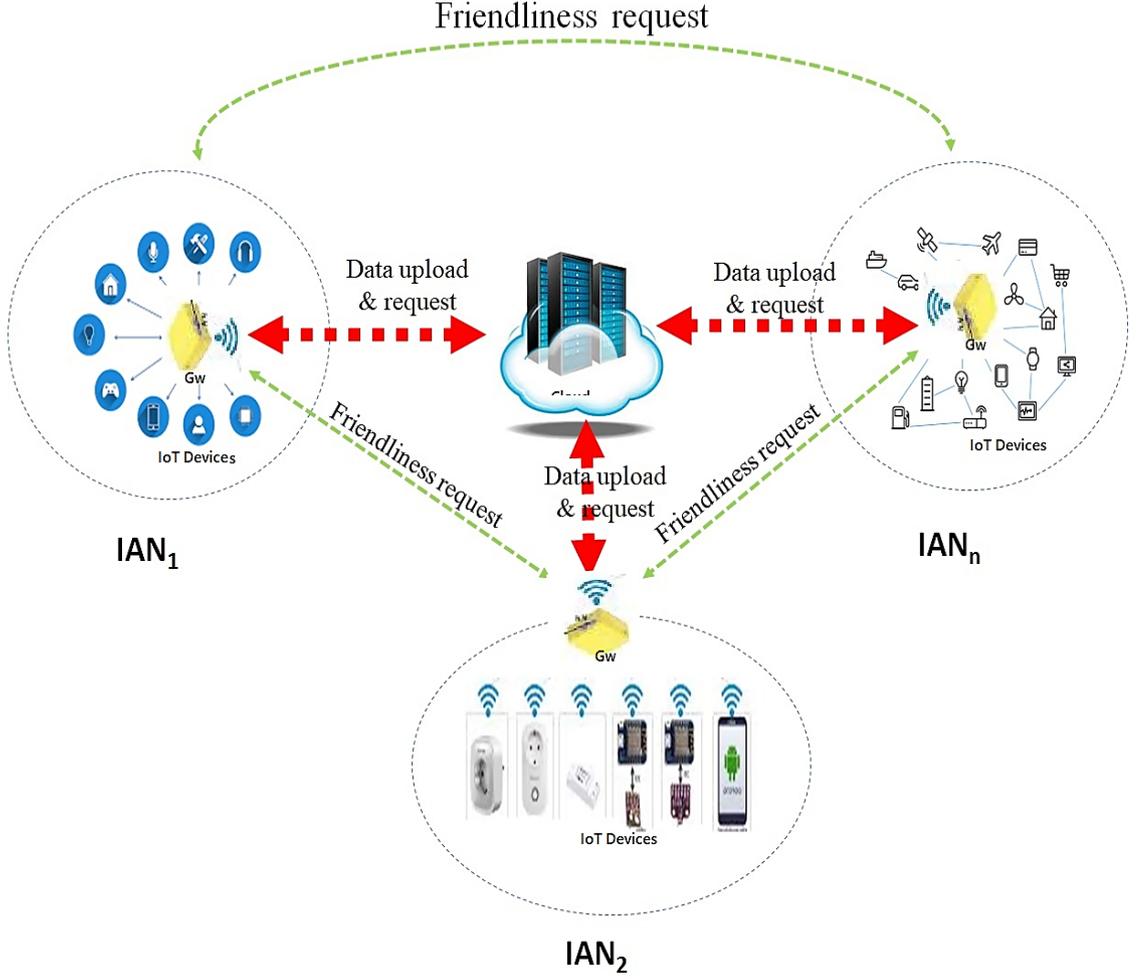


Figure 3.10: A system model of the FWSNs

### 3.2.1 Registration and Key Management

Registration and key management algorithm involves the generation of cryptographic key parameters for each FWS gateway (BS), Farm Manager Server (FMS) and sensors in the data harvesting system. The FMS registers each of the gateways and sensors in the FWSNs through the following steps:

- **Step 1:** The FMS randomly generates  $b_k \in Z_p^*$  as master secret key for each BS  $k$  and randomly generates  $a_{i,k} \in Z_p^*$  for each of the sensors  $i$  under the BS  $k$  to compute:

$$A_{i,k} = (b_k + a_{i,k})P; gp_k = b_kP,$$

where  $A_{i,k}$  is the secret key of the sensor and  $gp_k$  is the public key of the BS.

- **Step 2:** FMS keeps a triple  $\{A_{i,k}, b_k, gp_k, \kappa\}$  then publishes  $gp_k$  to the BS and securely sends a  $A_{i,k}$  as  $B_{i,k} = A_{i,k} \oplus \kappa$  and  $Y_k = b_k \oplus \kappa$  to BS  $k$ , where session  $\kappa$  is the mutually generated session key between the FMS and BS generated as follows:
  - BS of each cluster randomly generates  $x_1$ , computes, and publishes  $w_1 = x_1P$  and  $q_1 = w_1P$ .
  - FMS randomly generates  $x_2$ , computes, and publishes both  $w_2 = x_2P$  and  $q_2 = w_2P$ .
  - Then, the BS computes its session key with the FMS as  $\kappa = x_1q_2$ .
  - FMS also computes its session key with BS as  $\kappa = x_2q_1$ .
- **Step 3:** BS decrypts  $B_{i,k}$  and  $Y_k$  as:  $A_{i,k} = B_{i,k} \oplus \kappa$ ,  $b_k = Y_k \oplus \kappa$  and keeps a copy of  $\{A_{i,k}, b_k, gp_k\}$ .

### 3.2.2 Data Aggregation

A simple but efficient data aggregation approach is proposed for each BS to aggregate and secure their sensors data. To secure data and reduce traffic, at each synchronized interval, each sensor forwards its data  $M_{i,k}$  as  $m_{i,k} = M_{i,k} \oplus A_{i,k}$  to its BS, which generates the sensor aggregation parameters  $D_k, \Delta_k, \tau_k, \varphi_k, \omega_k$  as follows:

- **Step 1:** BS  $k$  randomly selects  $\gamma_k, \beta_k \in \mathbb{Z}_q^*$  and time stamp  $t_s$  and computes its aggregation parameters as:
 
$$\varphi_k = (b_k \oplus \gamma_k)$$

$$D_k = \gamma_k \oplus \beta_k$$

$$\Delta_k = H(gp_k \parallel D_k \parallel b_k)$$

$$\tau_k = D_k * \gamma_k - \beta_k$$

$$\omega_k = (b_k \oplus \beta_k \oplus t_s).$$
- **Step 3:** BS then publishes  $\varphi_k, D_k$  of each sensor to FMS, who will later use it for data de-aggregation.

- **Step 5:** BS aggregates all sensor data  $m_{i,k}$  in the synchronized interval as 
$$\rho_k = \sum_{i=1}^n m_{i,k}$$
- **Step 6:** Signs the aggregated data as  $\sigma_k = (H_{b_k}(\varphi_k \parallel \omega_k \parallel \beta_k \parallel t_s \parallel \tau_k \parallel \rho_k) \parallel \Delta_k)$
- **Step 7:** For each sensor data  $m_{i,k}$  BS computes the aggregation ratio  $\bar{h}_{i,k}$  as  $\bar{h}_{i,k} = \frac{m_{i,k}}{\min(m_{1,k}, \dots, m_{n,k})}$  and computes the signature ratio  $\eta$  as  $\eta_{i,k} = \sigma_k * \bar{h}_{i,k}$
- **Step 8:** BS uploads  $\rho_k \parallel \sigma_k \parallel (\eta_{1,k} \parallel \dots \parallel \eta_{n,k}) \parallel \varphi_k \parallel D_k$  to the harvester as their aggregated data, who then transfers it to FMS.

### 3.2.3 Data De-aggregation

The major problem with most existing aggregation schemes is how the receiver separates the individual data from the aggregated data received. To obtain individual sensor data from the aggregated data received, we developed the following separation procedure for the aggregation scheme as follows.

- **Step 1:** FMS recomputes  $\tau'_k, \omega'_k, \Delta_k, t_s$  as:
  - $\gamma'_k = \varphi_k \oplus b_k$
  - $\beta'_k = D_k \oplus \gamma'_k$
  - $t'_s = \omega_k \oplus t_s \oplus b_k \oplus \beta'_k$
  - $\tau'_k = D_k * \gamma'_k - \beta'_k$
  - $\Delta'_k = H(gp_k \parallel D_k \parallel b_k) \sigma'_k = (H_{b_k}(\varphi'_k \parallel \omega'_k \parallel \beta'_k \parallel t'_s \parallel \tau'_k \parallel \rho'_k) \parallel \Delta'_k)$
- **Step 2:** FMS checks if the computed  $\sigma'_k \stackrel{?}{=} \sigma_k$ , accepts the aggregated data  $\rho_k$  and proceeds to step 3, otherwise rejects  $\rho_k$ .
- **Step 3:** To deaggregate the aggregated data  $\rho_k$ , for each sensor, FMS extracts the signature ratio  $\eta_{i,k}$  from the FWSN data  $\rho_k \parallel \sigma_{i,k} \parallel (\eta_{1,k} \parallel \dots \parallel \eta_{n,k})$

to calculate the aggregation ratio  $\tilde{h}_{i,k}$  as  $\tilde{h}'_{i,k} = \frac{\eta_{i,k}}{\sigma'_{i,k}}$ , and generates the sensor data  $m_{i,k} = \frac{\eta'_{i,k} * \min(m_{1,k}, m_{2,k}, \dots, m_{n,k})}{\sigma_k}$ .

- **Step 4:** Decrypts  $m_{i,k}$  to obtain  $M_{i,k} = m_{i,k} \oplus A_{i,k}$ .

FMS repeats steps 1-4 for each of the sensors in each cluster to obtain their data. FMS securely sends the de-aggregated data to the cloud server for analytical requests from actors. Possible technological processes by which these data packages could be communicated between the various FWSN gateways and the data acquisition system, and between the latter and the cloud server\base station are explained in Section 3.3.

### 3.3 Methodology for the Farm Wireless Sensor Network System

For efficient wireless transmission\acquisition of information in FWSN systems, the combination of radio, serial Bluetooth, and WiFi communications is used. The radio communication medium is operated within the industrial scientific medical (ISM) range of bands. This is a band of radio and microwave frequencies clustered around 2.4GHz, reserved and designated for industrial, scientific and medical equipment that use RF. As a result of this, the developed UAV would not violate the policies and regulation especially the aviation policies and regulation. The FWSN consists of two parts; the WSN on the farm and the communication unit. The communication unit of the FWSN consists of onboard NRF24L01 based gateway (transceiver) on the UAV and NRF24L01 based module on the FWSN.

The hardware implementation of the onboard NRF24L01 based gateway comprises a central microcontroller that is digitally interfaced with an NRF24L01 transceiver module, as shown in Figures 3.11 and 3.12. The operational parameters of this system are given in Table 3.1. This is a *RF* front-end device that can be programmed to function as a transmitter or receiver, or a combination of both. In addition, it enables the interconnection of several similar transceiver modules,

Table 3.1: Parameters for radio transmission of data packages using the NRF24L01 transceiver module

S/N	Parameter	Value
1	Frequency range	2.4 - 2.5 GHz
2	Radiation power	8 dBm
3	Transmission rate	250 kbps - 2 Mbps
4	Effective range	$\approx 100$ m
5	Available no. of channels	125
6	Possible addresses/channel	6

which could be used to set up a WSN cybertronics and cyber-physical system. In the current work, it is configured to harvest sensors' data from a given FWSN and transmit them as a single data package to the data acquisition system on the UAV.

Meanwhile, the Wireless Sensor Network part is classified into two different network; Soil and Farm quality monitoring WSN (FWSN 1) and farm air quality monitoring WSN (FWSN 2). The architectures of the two classes are the same. The only difference is the types of sensors used as the nodes of each of the FWSN. For example, FWSN 1 comprise sensors such as a temperature sensor for measuring the temperature of a farmland soil. The soil temperature would be a good indicator to inform a farmer of how much irrigation his farmland would require. A soil moisture sensor is also incorporated to estimate the percentage of moisture contained in farmland soil. This enables a farmer to determine when the soil is saturated with water and when to stop irrigation. A leaf moisture sensor can also be used to estimate the average volume of moisture on the surface of a leaf, which, in conjunction with soil temperature and moisture sensors, is a pointer to the level and timing of required irrigation. Air temperature and relative humidity sensors are sensors that could help a farmer correctly estimate the current weather season of the year. Meanwhile, the farm air quality monitoring WSN (FWSN 2) comprise of the following sensors which are used for monitoring the farm air quality: BME280 (Temperature and Humidity), PMS5003 (PM2.5 and PM10), MQ-7 (Carbon monoxide (CO)), MiCS-6814 (Nitrogen dioxide (NO<sub>2</sub>)).

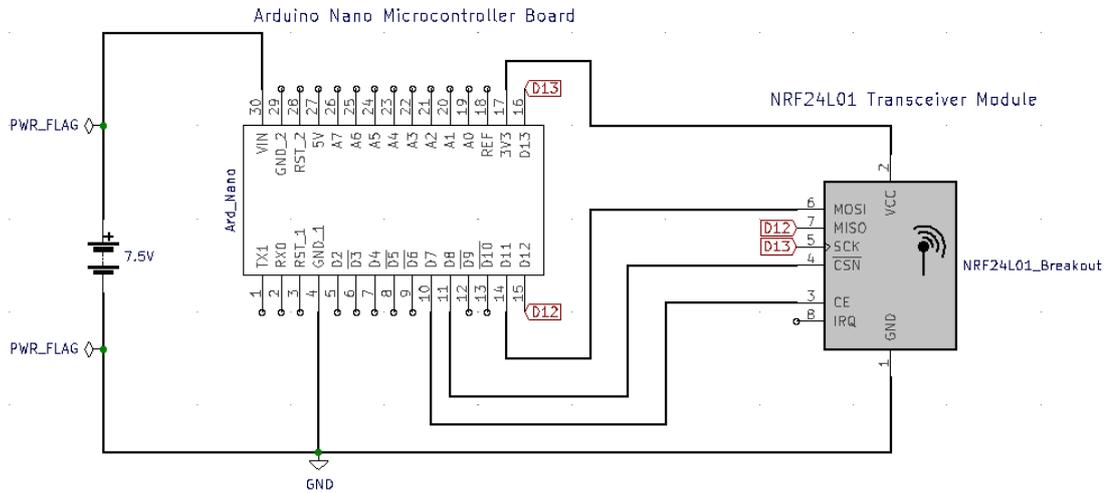


Figure 3.11: Circuit diagram of microcontroller based NRF24L01 transceiver system

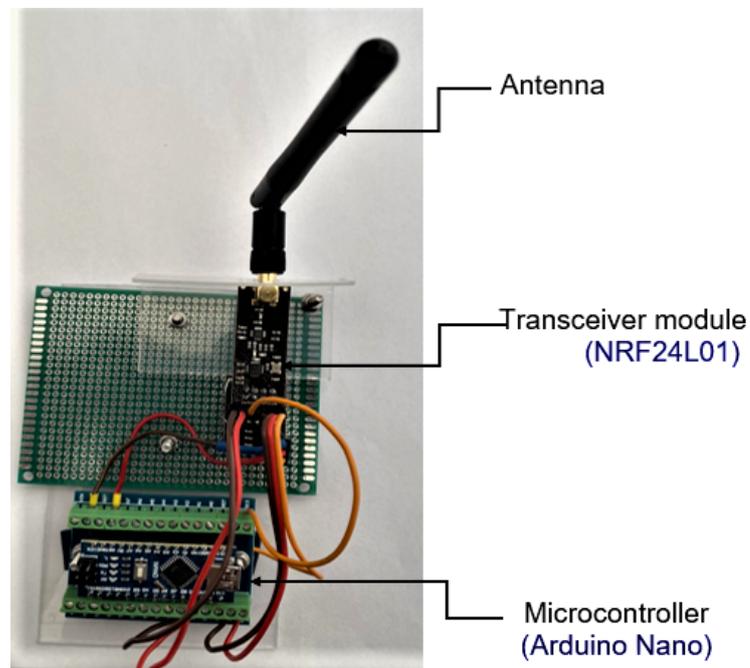


Figure 3.12: Constructed NRF24L01 transceiver module

### 3.3.1 Architecture and Implementation of FWSN

This subsection explains the methodology of the design and implementation of the IoT based FWSN, hardware and software requirements, and the schematic setup and the algorithm used . The block diagram in Figure .1 visually illustrates the interconnections and relationships among the different components comprising the system, effectively portraying the functional structure and dependencies of the system's architecture. The hardware unit of the FWSN includes sensors for measuring temperature, air and soil quantities, a microcontroller for data processing, and output devices such as displays. It enables real-time data collection, analysis, and visualization for effective air quality monitoring. Figure .2 depicts the schematic setup of the FWSN, showcasing the interconnection of sensors, microcontrollers, and output devices for comprehensive monitoring and analysis. The hardware unit of the FWSN is divided into 4 sections each encompassing a function. These sections are listed below.

- Power Unit
- Sensor Unit
- Microcontroller Unit
- Output Unit

#### **FWSN: Power Unit**

The power section shown in Figure is responsible for providing a stable and appropriate power supply to the components of the monitoring station. It consists of a pair of 3.7V rechargeable lithium-ion (Li-ion) batteries and a voltage regulator circuit. The Li-ion battery is used because of its high energy density, low self-discharge rate, and long cycle life.

A voltage regulator circuit is used to regulate the Li-ion battery voltage to a stable 3.3V or 5V, which is required by the Arduino Nano and ESP32, respec-

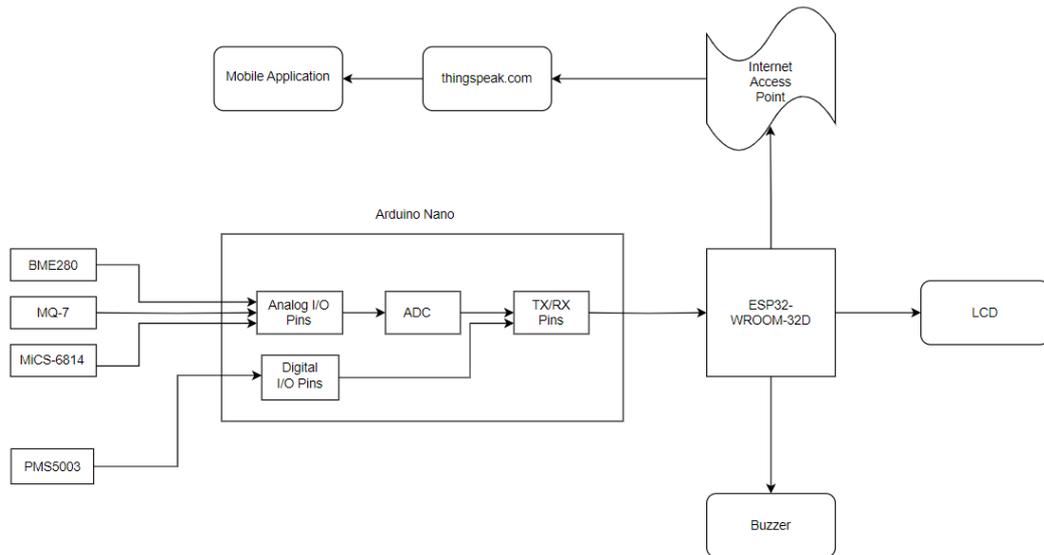


Figure 3.13: Interconnections and relationships among the different components of FWSN

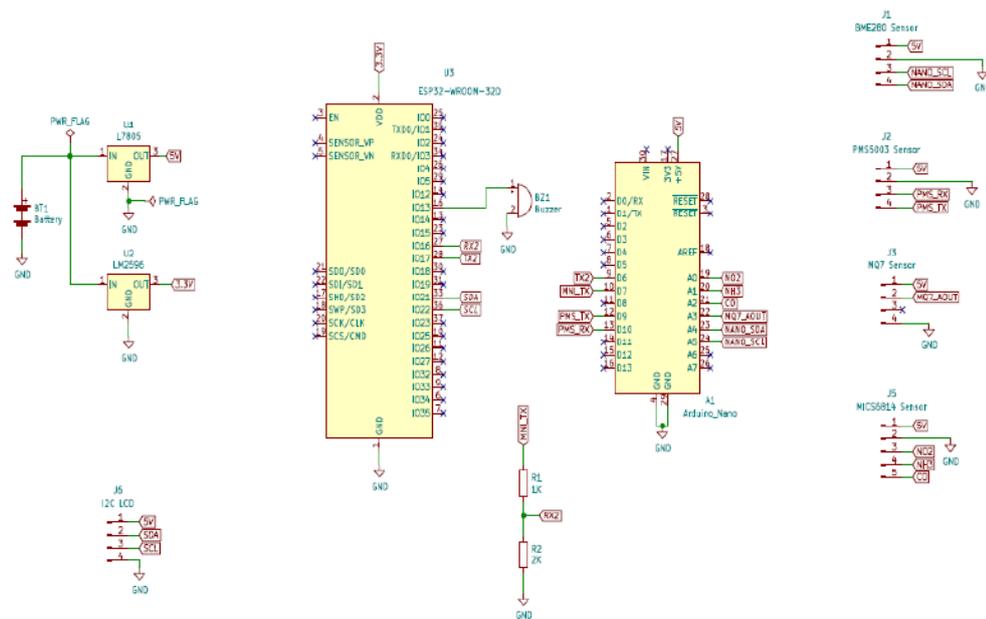


Figure 3.14: Schematic set-up of the FWSN

tively. For the Arduino Nano, The L7805 voltage regulator is used, the L7805 is a three-terminal linear voltage regulator that provides a fixed output voltage of 5V from an input voltage of up to 35V. The L7805 is connected to the input voltage source, ground, and output load, which is the Arduino Nano. For the ESP32 connection, the LM2596 voltage regulator is used, The LM2596 is a popular step-down voltage regulator that can be used to convert a higher input voltage

to a lower output voltage with high efficiency. It accepts the input voltage and outputs a fixed voltage of 3.3V, the operating voltage of the ESP32. The LM2596 is connected to the input voltage source, ground, and output load.

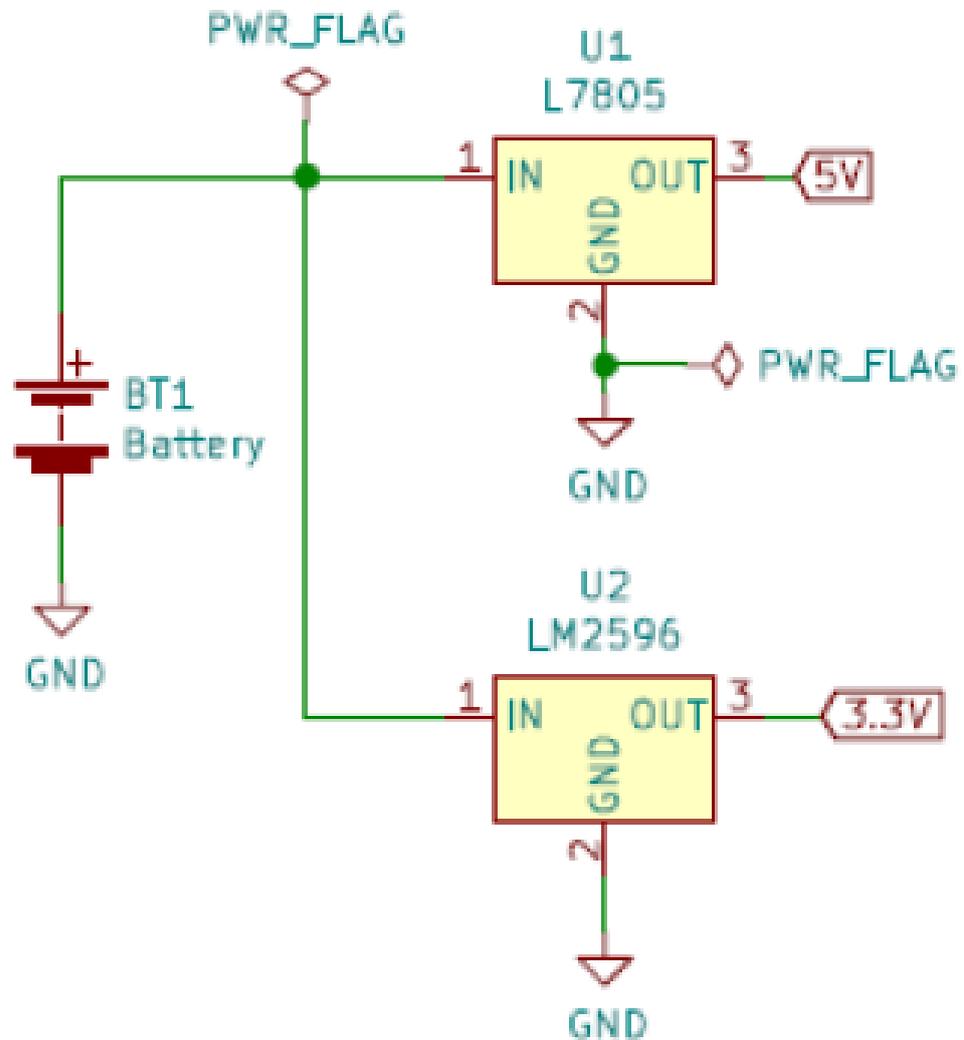


Figure 3.15: Schematic circuit of power unit

### FWSN: Sensors' Unit

The sensor unit, as shown in Figure .4 of the hardware unit consists of the sensors used to measure the concentration of the pollutants and the Arduino Nano microcontroller, which acts as the Analog-to-Digital converter. The following sensors are used for the air quality monitoring FWSN: BME280 (Temperature and Humidity), PMS5003 (PM2.5 and PM10), MQ-7 (Carbon monoxide (CO)),

MiCS-6814 (Nitrogen dioxide (NO<sub>2</sub>)).

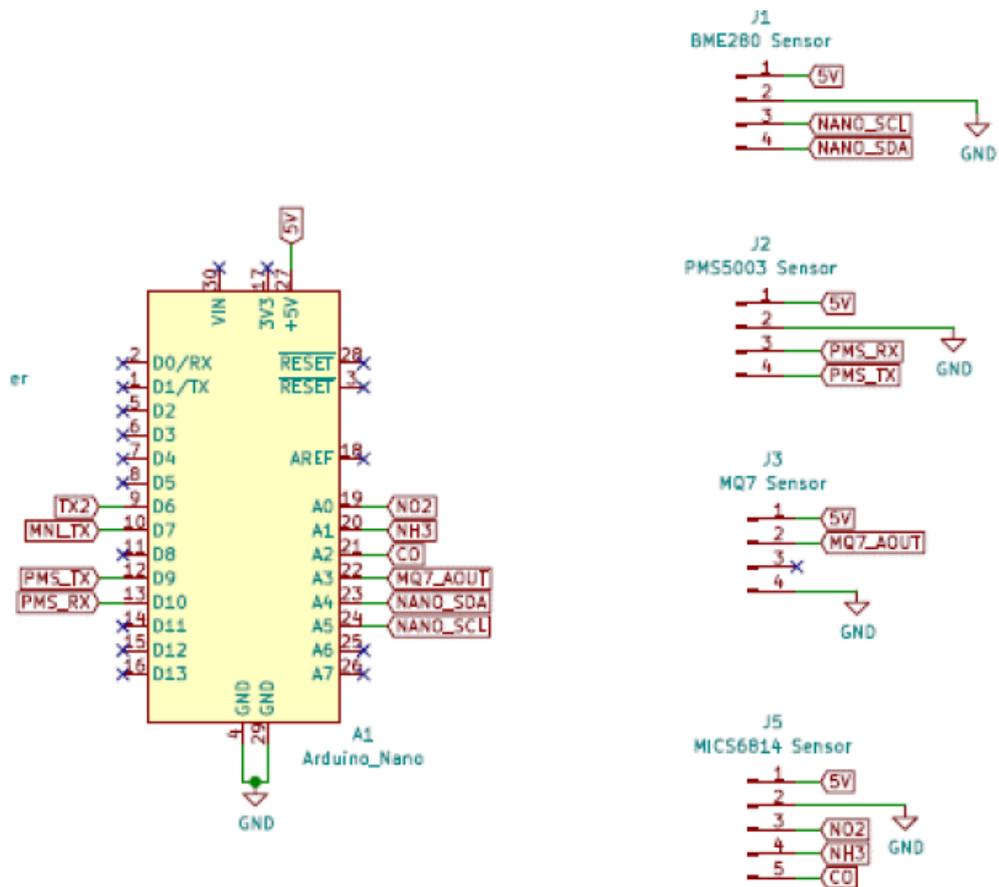


Figure 3.16: Schematic circuit of sensors' unit

## FWSN: Controller Unit

The microcontroller section consists of an ESP32-WROOM-32D development board acting as the main controller of the system and the Arduino Nano board. In this setup, the ESP32 is used to run the application code, communicate with other devices over Wi-Fi or Bluetooth, and provide a user interface. The Arduino Nano, with its onboard analog-to-digital converter (ADC), can be used to read analog signals from sensors such as gas sensors and send the data to the ESP32 for processing and analysis. The transmit signal from the Arduino Nano must be stepped down from 5V to 3.3V as the ESP32 is at a 3.3V logic level while the Arduino Nano is at a 5V logic level. A voltage divider circuit is employed for the step-down operation. Figure .5 shows the schematic of ESP32-WROOM-32D and Arduino Nano board controllers.

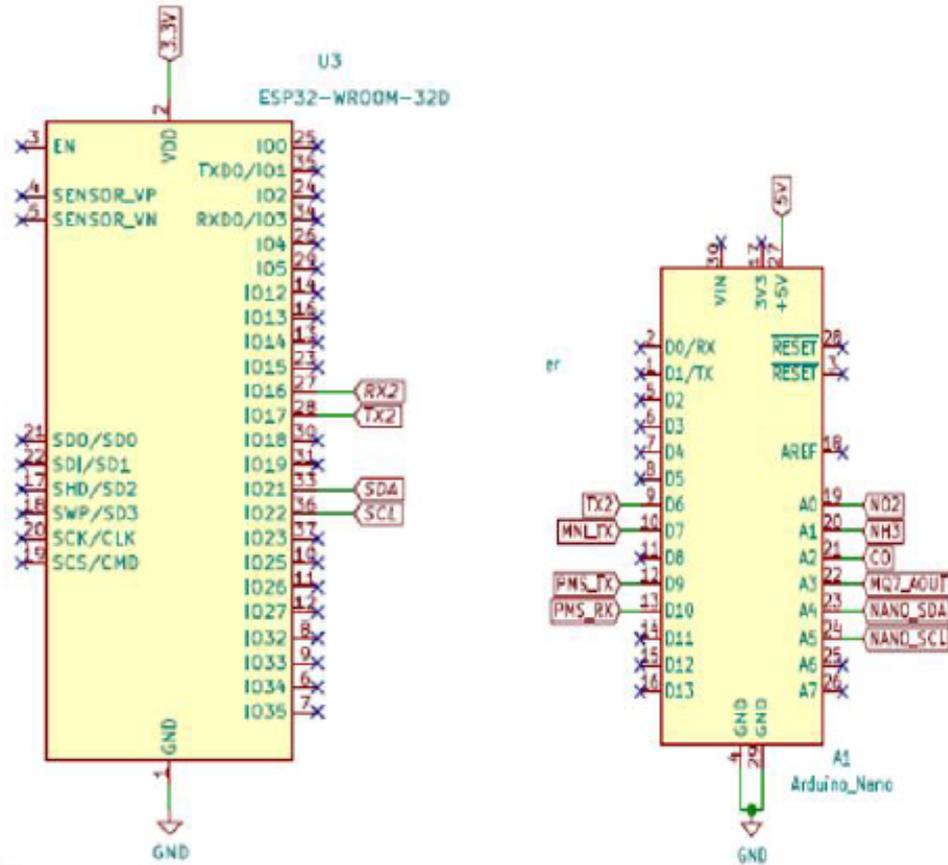


Figure 3.17: Schematic circuit of sensors' unit

## FWSN:Output Unit

The output section of the system comprises devices that provide visual and auditory feedback to the user. These devices include an I2C LCD module, that uses the I2C communication protocol to interface with a microcontroller and is used for displaying sensor data. The LCD is connected to the ESP32 microcontroller as follows: GND to GND, VCC to 3.3V, Serial data (SDA) to GPIO21 pin, Serial clock (SCL) to GPIO22 pin.

### 3.3.2 Data Acquisition system

This is the cybertronic communication system that is onboard the UAV. This constitutes the augmentation of the NRF24L01 transceiver system in Figure 3.11 with the WiFi ESP32 WiFi module. Its function is to receive data packages



Figure 3.18: Complete package of the FWSN

from the FWSN transmitter and then relay them to the base station and a cloud server via the RF communication channel and WiFi connectivity, respectively. The WiFi ESP32 WiFi module is adopted for the latter to effectively extend the capability of the RF communication medium and transmit the same data package to a remote\ beyond the LoS cloud server via the Internet. Described in Section 3.2 are the security schemes and algorithms that are used to develop the software components of the WSN system.

The adopted data aggregation\ deaggregation (including intrinsic security features) is a core part of the software layer of the WSN system. In pseudocode, this task is sufficiently described as Algorithms 2 and 3 for the transmitter (ground-based) and receiver (airborne) ends, respectively, of the FWSN system. The practical demonstration of this FWSN system is captured in Figure 3.19, the visualization of which is shown in Figure 3.20. This shows the applicability of the system for the collection, transmission, classification, visualization, and increase in response speed of farm data. This telemetric method was applied in the field evaluation of the complete UAV\FWSN system. Several clusters of FWSN were sampled. The details and results of these are given in Section 3.2 and Section 4.3.1.

---

**Algorithm 2** Data aggregation and transmission

---

**Input:** *moist\_sensor, temp\_sensor, Channel\_Address,*

**Input:** *pH\_sensor, gas\_sensor*

**Output:** *data\_package*

```
1: Activate RADIO_PROTOCOL
2: Assign port (7,8)  $\leftarrow$  (CE, CSN)
3: function SETUP_FUNC ▷ Sets device as transmitter
4:   Open RADIO_WRITE
5:   Stop LISTENING_MODE
6: end function
7: repeat ▷ Transmission iteration
8:   function AGGREGATION(data_package, base_parameters)
9:      $\rho = \text{humi} + \text{temp} + \text{pH} + \text{conc}$ 
10:     $\text{temp\_ratio} = \frac{\text{temp}}{\min(\text{humi}, \text{temp}, \text{pH}, \text{conc})}$ 
11:     $\text{humi\_ratio} = \frac{\text{humi}}{\min(\text{humi}, \text{temp}, \text{pH}, \text{conc})}$ 
12:     $\text{conc\_ratio} = \frac{\text{conc}}{\min(\text{humi}, \text{temp}, \text{pH}, \text{conc})}$ 
13:     $\text{pH\_ratio} = \frac{\text{pH}}{\min(\text{humi}, \text{temp}, \text{pH}, \text{conc})}$ 
14:     $\sigma_k = (H_{b_k}(\varphi_k \parallel \omega_k \parallel \beta_k \parallel t_s \parallel \tau_k \parallel \rho_k) \parallel \Delta_k)$ 
15:     $\text{temp\_sign\_ratio} = \text{temp\_ratio} * \sigma_k$ 
16:     $\text{conc\_sign\_ratio} = \text{conc\_ratio} * \sigma_k$ 
17:     $\text{humi\_sign\_ratio} = \text{humi\_ratio} * \sigma_k$ 
18:     $\text{pH\_sign\_ratio} = \text{pH\_ratio} * \sigma_k$ 
19:     $\text{data\_package} = \rho_k \parallel \sigma_k \parallel (\text{temp\_sign\_ratio} \parallel \text{temp\_sign\_ratio} \parallel$ 
20:  $\text{humi\_sign\_ratio} \parallel \text{conc\_sign\_ratio} \parallel \text{pH\_sign\_ratio} \parallel \varphi_k \parallel D_k)$ 
21:   end function ▷ Transmits data package to receiver
22:   while using Channel_Address do
23:     Bulk_Transmit data_package
24:   end while
25:   wait for 1.0 s
26: until (false) ▷ Iteration logic
```

---

---

**Algorithm 3** De-aggregation and re-transmission

---

**Input:** *data\_package*, *Channel\_Address*, *IP\_Address*, *Port\_Address*

**Output:** *serial\_data*, *data\_package*

```
1: Activate RADIO_PROTOCOL
2: Activate WIFI_PROTOCOL
3: Activate SERIAL_COM                                ▷ For Bluetooth\USB com.
4: Assign port (7,8) ← (CE, CSN)
5: function SETUP_FUNC                                ▷ Sets device as receiver
6:   Open RADIO_READ
7:   Open WIFI_GATEWAY
8:   Open SERIAL_PORT
9:   Start LISTENING_MODE
10: end function
11: repeat                                            ▷ Acquisition iteration
12:   if Radio_Signal AVAILABLE then
13:     while using Channel_Address do
14:       Fetch data_package                            ▷ From transmitter
15:     end while
16:     while using Port_Address do
17:       Serial_Transmit (data_package)
18:       function DEAGGREGATION(data_package, base_parameters)
19:          $temp = \frac{temp\_sign\_ratio * \min(humi, temp, conc, pH)}{\sigma_k}$ 
20:          $conc = \frac{conc\_sign\_ratio * \min(humi, temp, conc, pH)}{\sigma_k}$ 
21:          $humi = \frac{humi\_sign\_ratio * \min(humi, temp, conc, pH)}{\sigma_k}$ 
22:          $pH\_ratio = \frac{pH\_sign\_ratio * \min(humi, temp, conc, pH)}{\sigma_k}$ 
23:         deaggregate_data_package = temp, humi, conc, pH
24:       end function
25:     end while
26:     while using IP_Address do
27:       Bulk_Transmit data_package
28:     end while
29:   end if
30: until (false)                                    ▷ Iteration logic
31: wait for 1.0 s
32:
```

---

n,	soil_humidity,	soil_temperature,	soil_pH,	voc_concentration
0,	0.00,	0.00,	0.00,	0.00
1,	53.00,	21.00,	7.00,	151.00
2,	57.00,	34.00,	7.00,	177.00
3,	51.00,	31.00,	6.00,	192.00
4,	58.00,	28.00,	7.00,	175.00
5,	52.00,	32.00,	7.00,	185.00
6,	55.00,	28.00,	7.00,	159.00
7,	51.00,	25.00,	7.00,	169.00
8,	50.00,	27.00,	6.00,	184.00
9,	51.00,	30.00,	6.00,	156.00
10,	55.00,	32.00,	6.00,	165.00
11,	56.00,	28.00,	6.00,	169.00
12,	53.00,	34.00,	7.00,	186.00
13,	58.00,	34.00,	6.00,	162.00
14,	53.00,	21.00,	7.00,	186.00
15,	56.00,	33.00,	7.00,	173.00
16,	51.00,	26.00,	6.00,	150.00
17,	53.00,	33.00,	7.00,	165.00
18,	56.00,	30.00,	7.00,	164.00
19,	50.00,	21.00,	6.00,	184.00
20,	52.00,	33.00,	7.00,	178.00
21,	59.00,	22.00,	6.00,	168.00
22,	54.00,	33.00,	6.00,	181.00
23,	56.00,	23.00,	6.00,	153.00
24,	57.00,	29.00,	7.00,	189.00
25,	50.00,	34.00,	6.00,	189.00
26,	59.00,	24.00,	6.00,	179.00
27,	56.00,	22.00,	7.00,	182.00
28,	59.00,	31.00,	7.00,	188.00
29,	50.00,	29.00,	7.00,	162.00
30,	57.00,	27.00,	7.00,	160.00

Figure 3.19: Practical demonstration of farm data acquisition at high turnaround speed of farm data using the FWSN system

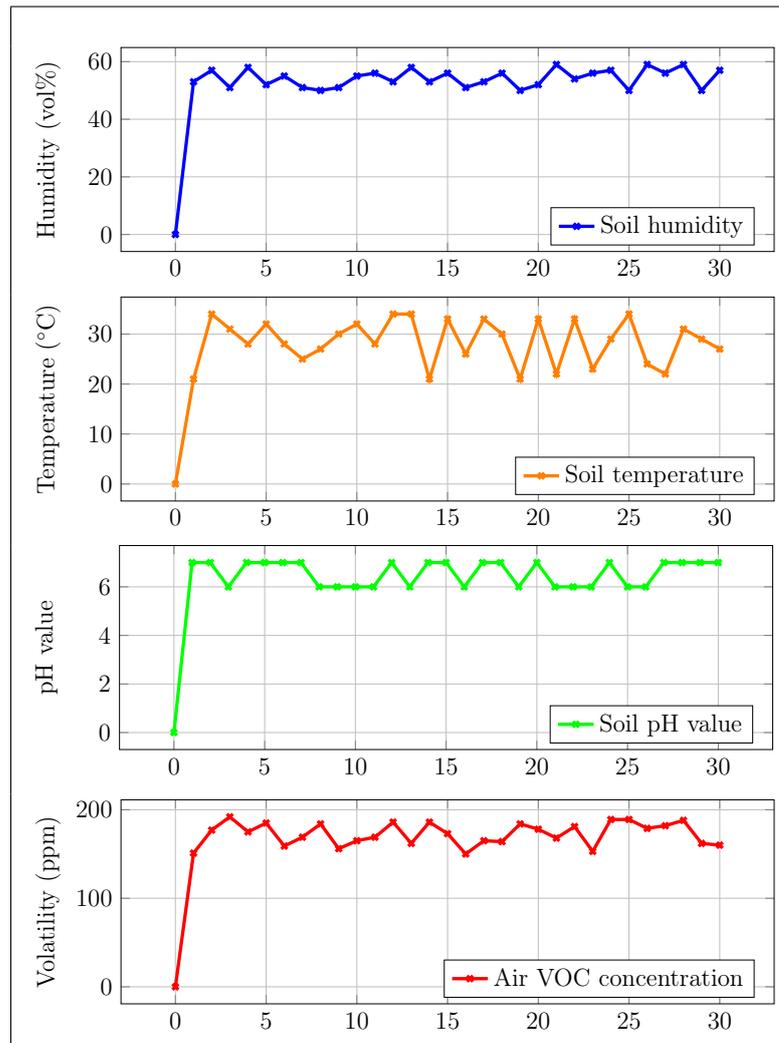


Figure 3.20: Visualization of sensors data from a single node in the FWSN

# Chapter 4

## Test and Performance Analysis

### 4.1 Tests and Performance Analysis on UAV

#### 4.1.1 Airworthiness and Hovering Capability of the UAV

To evaluate the dynamics of the UAV, several test flights are performed through several flight tests with an on-board data acquisition system on farmland, as shown in Figure 4.1. The PID-based thrust vectoring function in the algorithm 1 was applied to the four ESC motor propulsion systems of the UAV and was flown to the test location so that it would hover or move at a fixed height above ground level. Furthermore, the high-altitude UAV flight, as captured, can be viewed in Figures 4.3 and 4.3. This demonstrates its propulsion power and maneuverability. Therefore, attitude measurements from the UAV's IMU were streamed to the base station and plotted against its time stamps to visualize the gyroscopic orientation of the UAV (note: yaw angle,  $\psi$  was not considered in this analysis since only angles  $\phi$  and  $\theta$  are used for gyrostabilization of the UAV).

#### 4.1.2 Payload-carrying Test and Analysis

In this subsection, we verified the UAV's payload-carrying capacity. Here, flyable payloads of various weights, 10 N to 30 N (in steps of 5N), were consecutively boarded on the UAV, which is then flown to hover over a specific geographical

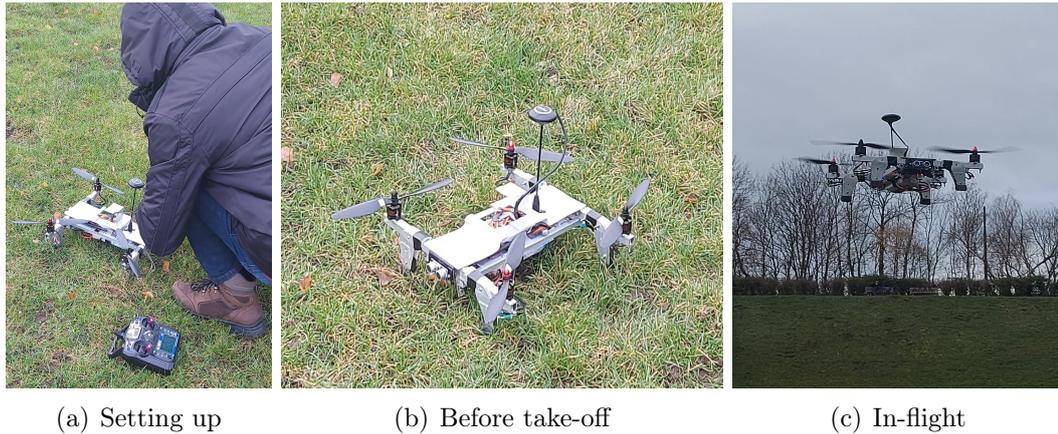


Figure 4.1: Testing of the data harvester in a simulated  $50 \text{ m} \times 100 \text{ m}$  FWSN plantation

location. Its attitude measurements are telemetered to the base station and used to visualize the gyroscopic behavior of the UAV. Important results are shown in Figure 4.4.

This enabled us to understand the aerodynamic stability of the UAV and adjust the control parameters of its flight algorithm to fine-tune its dynamic responses to external forces such as gravity and air drag until the required rotor-dynamic gyro-stability is attainable. The simulation results in Figure 4.4(a-c) show that the the rotor-dynamic vibration of the UAV increase as its load profile is continually increased from 10 N to 30 N, due to the induced misalignment of thrust vectors. In particular, Figure 4.4(b) is the result of the test carried out with a payload of 20 N (which corresponds to approximately 2.04 kg). This approximately supports the estimated weight-carrying capacity (that is, 2.3 kg). Overall, this analysis indicates that the UAV with a payload within 2.04 kg, as a mobile base station, is suitable and safe for aerial harvesting of data from sensor that are embedded as part of WSNs in a spatial farm.

## 4.2 Tests and Performance Analysis on FWSN System

To test the performance of the FWSN system (incorporating the FWSN gateways, the data acquisition system, and the aggregation schemes), we used three



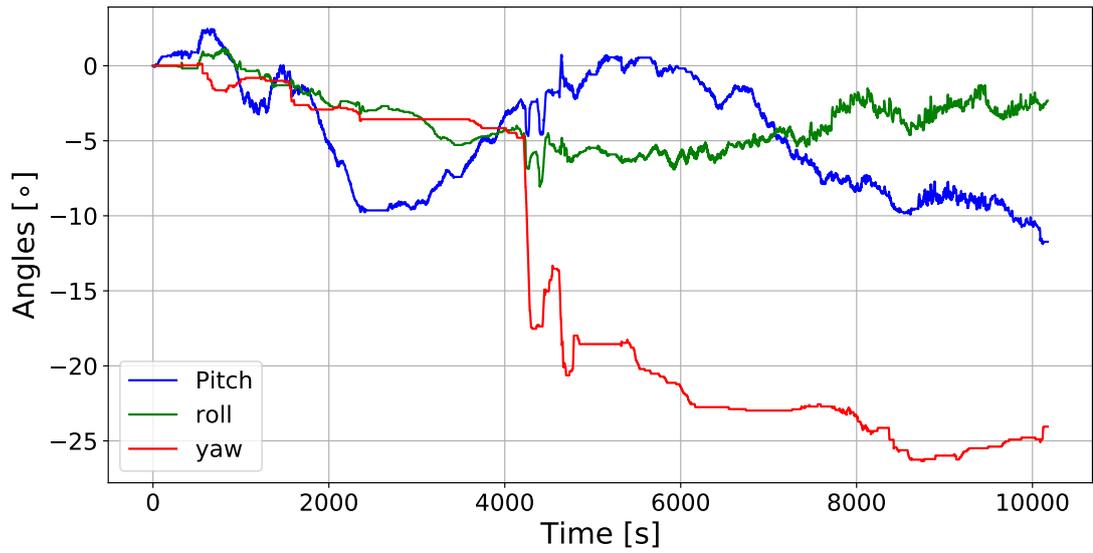
Figure 4.2: Caption of UAV's ascension to high altitude

50 m by 100 m farmlands. Also, to evaluate the effect of the data harvester and aggregation scheme on the turnaround of the sensors' data, we measured the average round trip time of the sensors to two cloud servers with and without the aggregation scheme. Tables 4.1 and Figure 4.5 show the average round trip time for varied numbers of different 32-byte size sensors' data harvested, aggregated,

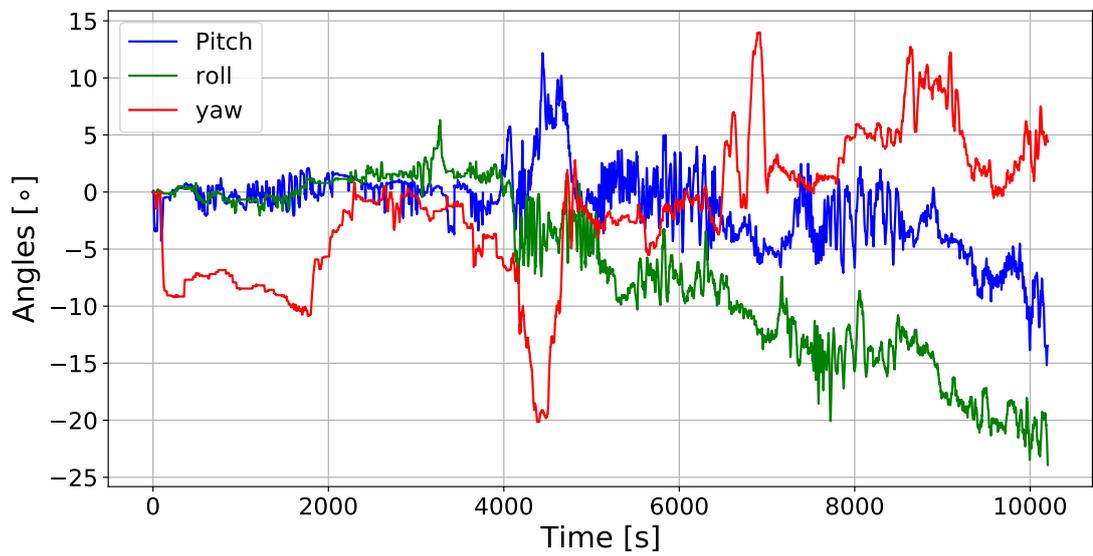


Figure 4.3: Caption of UAV's flight at high altitude

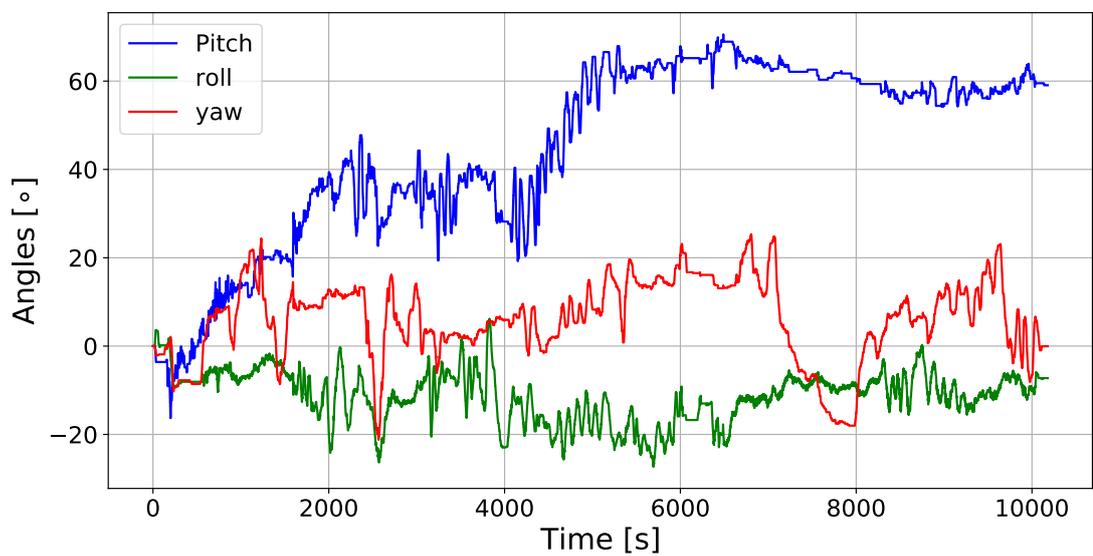
and uploaded to Google (Note that  $t_1$  denotes time latency for data transmission with data harvester\aggregation scheme while  $t_2$  denotes data without two entities). These analyzes reveal that the data aggregation scheme reduces the average round-trip time, regardless of the number of sensors on the farm compared to when the data aggregation scheme is not engaged. This indicates that the use



(a) Attitudes plot for hovering test with 10 N payload



(b) Attitudes plot for hovering test with 20 N payload



(c) Attitudes plot for hovering test with 30 N payload

53  
Figure 4.4: Plots of attitudes versus time to evaluate the hovering capability of the UAV boarded with various sizes of payloads

Table 4.1: Round trip time (ms) of a sensor’s 32-byte data from farm to Google cloud with and without data harvester and aggregation

No. of sensors	$t_1$ (ms)	$t_2$ (ms)
1	387	392
2	558	786
3	357	1250
4	376	1740
5	402	2002
7	457	2461
8	465	2934
9	521	3275
10	554	4821

of the developed data aggregation scheme improves the data harvester in terms of latency and traffic.

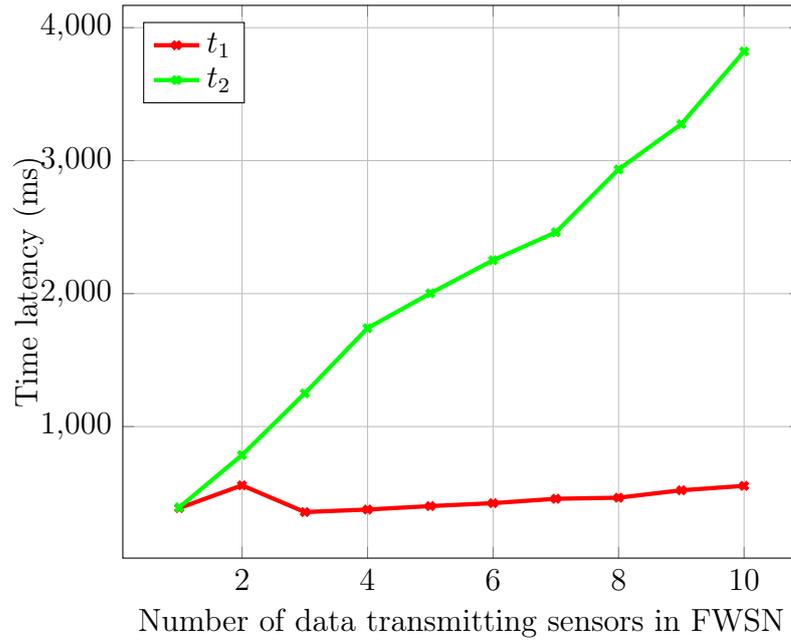


Figure 4.5: Comparison in the average round trip time (i.e., time latency) between FWSN and Google cloud with and without data harvester and aggregation for different sensors

#### 4.2.1 Tests and performance evaluation of FWSN

This subsection interprets and discusses the data collected during the implementation and testing phases of the FWSNs. The FWSN is subjected to two forms of testing:

- **Operational Testing:** In which the functionality of the system is evaluated against the objectives proposed towards achieving the aim of the project.
- **Performance Testing:** The ability of the system to capture changes in air quality parameters is evaluated.

## **Operational Testing**

The system is subjected to evaluation to assess its ability to meet the objectives of the project and to verify its overall functionality and performance alignment with the objective of the project. Three key functionalities must be present in the system to indicate the successful accomplishment of the project objectives; real time measurement, and remote access.

## **Real Time Measurement**

The system enables real-time monitoring of quantities such as temperature, soil humidity, PH, PM2.5, PM10, CO, and NO2. This functionality allows the data acquisition or gateway on the UAV to conveniently access the collected data.



Figure 4.6: Display showing the real-time harvested data on the display of the FWSN

## Remote Access

Authorized farmers can easily retrieve the system’s data through the mobile application developed on their mobile devices or the MQTT dashboard on ThingsSpeak as shown in Figure 4.6

## 4.3 Performance Analysis on Security and Aggregation Scheme

### 4.3.1 Performance Evaluation of Data Aggregation

We analyze the efficiency of the data aggregation scheme in this section by evaluating the aggregation cost at the FWSN side, communication overhead, in-of-network, and out-of-network access delay for a different number of the IoT devices. We also compare our scheme with four state-of-the-art schemes proposed

in [1], [2], [3], and [4] based on functionalities and aggregation cost. These schemes are selected based on their similarities with the proposed scheme in terms of the problems they solve and their approaches. For example, they all solve security and traffic issues in a heterogeneous wireless sensor network and adopted cryptographic primitives to evolve aggregation solutions. Access delays of our scheme at different time intervals for a different number of in and out-of-IAN accesses and different IoT devices were observed. We also compared the communication overhead of our scheme with a scheme based on basic Paillier encryption (AggBPE) and another scheme in [35].

## Experiment setup

We adopt a Type A curve in the pairing-based cryptographic library. The implementation uses a 160-bit elliptic curve group based on the supersingular curve  $y^2 = x^3 + x$  over a 512-bit finite field on Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz processor as the gateway and 20 Arduino Uno @ 12MHz based IoT devices for each of the three IANs. This is to obtain the running time of cryptographic operations in all schemes, such as the running time of a pairing  $T_{bp}$ , the running time of a modular exponent operation  $T_{exp}$ , the running time of scalar multiplications  $T_{sm}$ , the running time of one-point addition  $T_{pa}$ , the running time of an exclusive or, and the running time of a hash function  $T_{hash}$ . The efficiency of the cryptographic procedures in terms of data aggregation, signature generation, and verification, and in and out-of-network access of all the schemes are analyzed under the same situation.

### 4.3.2 Results and Discussion

The results obtained from all performance analyzes are presented as follows:

## Functionalities

For a security scheme to be suitable for IoT, it must have the following security characteristics; recoverability, ensure data integrity, authorized aggregation, end-to-end confidentiality, end-to-end integrity, and anonymity. A perfect recovery of individual data from the aggregated data should be achieved. Most of the IoT data are sensitive; therefore, a security scheme must ensure confidentiality by preventing unauthorized access to the data. Integrity involves the ability of the scheme to deliver original data to the authorized user. IoT privacy is one of the main considerations required of any IoT scheme. It involves the protection of the information of individuals from exposure in the IoT environment. Table 2 shows the features of the proposed scheme compared to other schemes. It shows that the proposed scheme not only has more features but also has more security functionalities that are more relevant to IoT networks.

## Data security and aggregation costs

Data security cost involves the cost of encryption or perturbation, and cost of signature generation and verification operations. For the proposed scheme, a sensor requires  $3T_{hs}+2T_{xor}$  operations for generating the perturbation-value and signature. Meanwhile, the gateway requires  $nT_{sm}$  operations to aggregate the data of  $n$  sensors of the FWSN. The scheme in [1] involves  $4T_{sm}+1T_{pa}+T_{hs}$  operations for signature generation operations, while the gateway requires  $(2n-2)T_{pa}$  operations for homomorphic-based aggregation. The scheme in [2] requires  $2T_{sm}+T_{pa}$  operations for the encryption and signature operation, and  $(n+1)T_{sm}+nT_{pa}$  operations to perform the homomorphic aggregation. Furthermore, the scheme in [3] requires  $4T_{sm}+T_{pa}+T_{hs}$  operations for the encryption and signature operation, while  $(2n+2)T_{sm}+(2n-2)T_{pa}+T_{hs}$  operation is required for aggregation. The scheme in [4] requires  $T_{sm}+T_{hs}$  for the encryption and signature operation, and  $(n+3)T_{sm}+T_{hs}$  for the aggregation.

Using the cost of each primitive and operation, we generated the security and

aggregation cost for each scheme. For security and aggregation schemes to be effective, the number of sensors involved in aggregation should not have much effect on the overall cost of aggregation. The results, as depicted in Figure 4.12, indicate that the proposed scheme not only requires the lowest aggregation cost but also the number of sensors involved does not increase the total aggregation cost compared to other schemes.

Figure 4.8 shows that the proposed scheme has the lowest encryption or perturbation and signature cost of 0.168ms followed by the scheme in [4] of 13.46ms compared to other schemes. According to Figures 4.9 and 4.9, the number of sensors does not increase the aggregation cost of the proposed scheme. It further shows that other schemes have higher aggregation cost than the proposed scheme.

The percentage aggregation costs of four different sizes of FWSN, as shown in Figure 4.10, also corroborates the independence of the developed scheme's aggregation cost on the size of the FWSN. Meanwhile, the aggregation cost of other schemes increases as the size of the FWSN increases. This affirms that out of all the schemes, it is only the developed scheme that meets up the required security goals at the lowest computation cost.

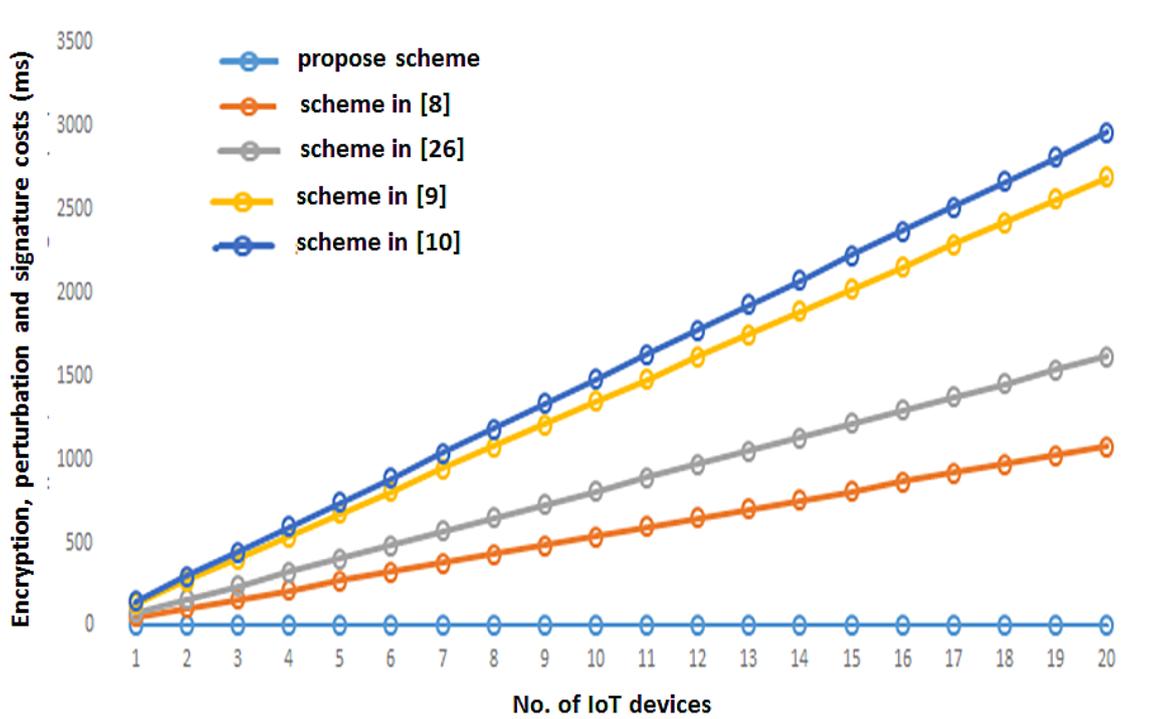


Figure 4.7: Comparison of the effects of the number of sensors in FWSN on the security cost

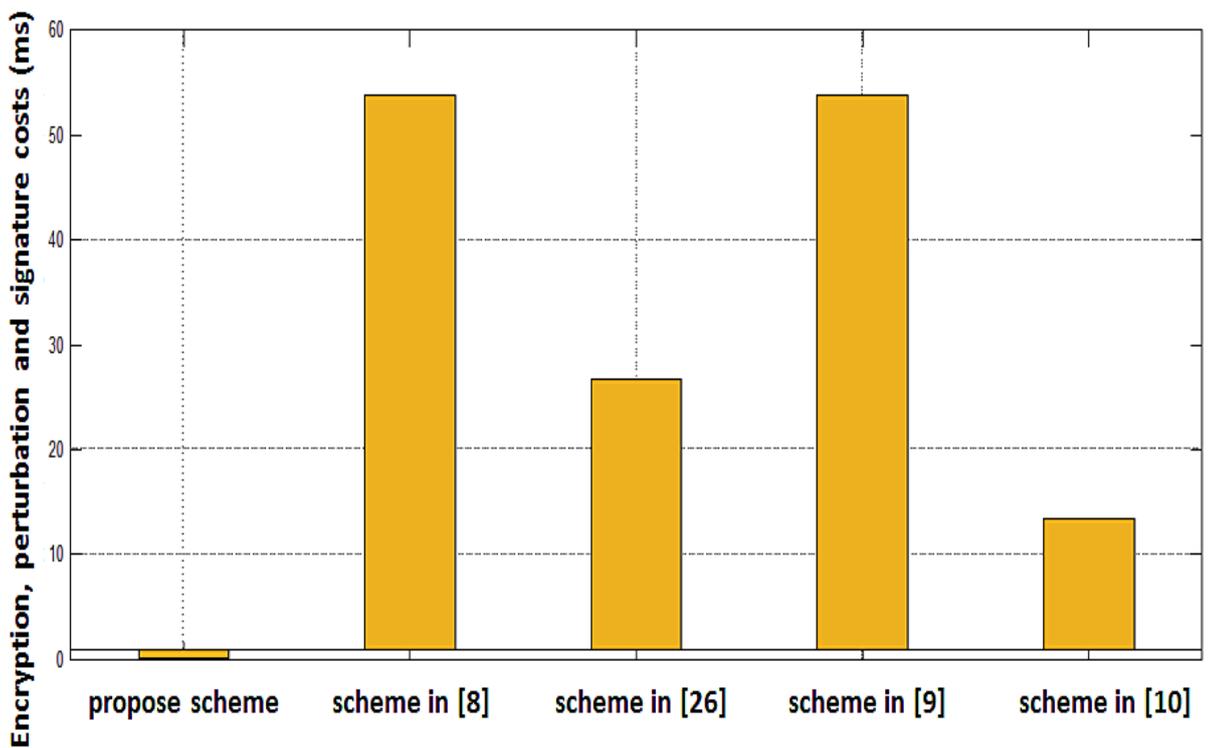


Figure 4.8: Estimated encryption and signature costs for a sensor

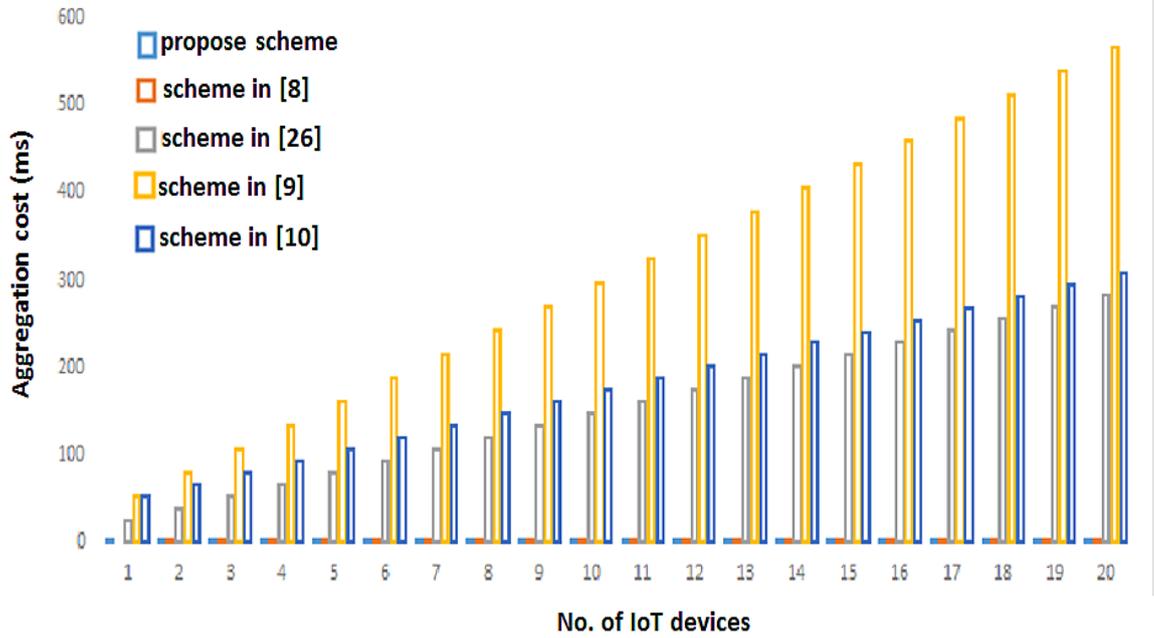


Figure 4.9: Comparison of the effects of number of sensors on FWSN on the aggregation cost

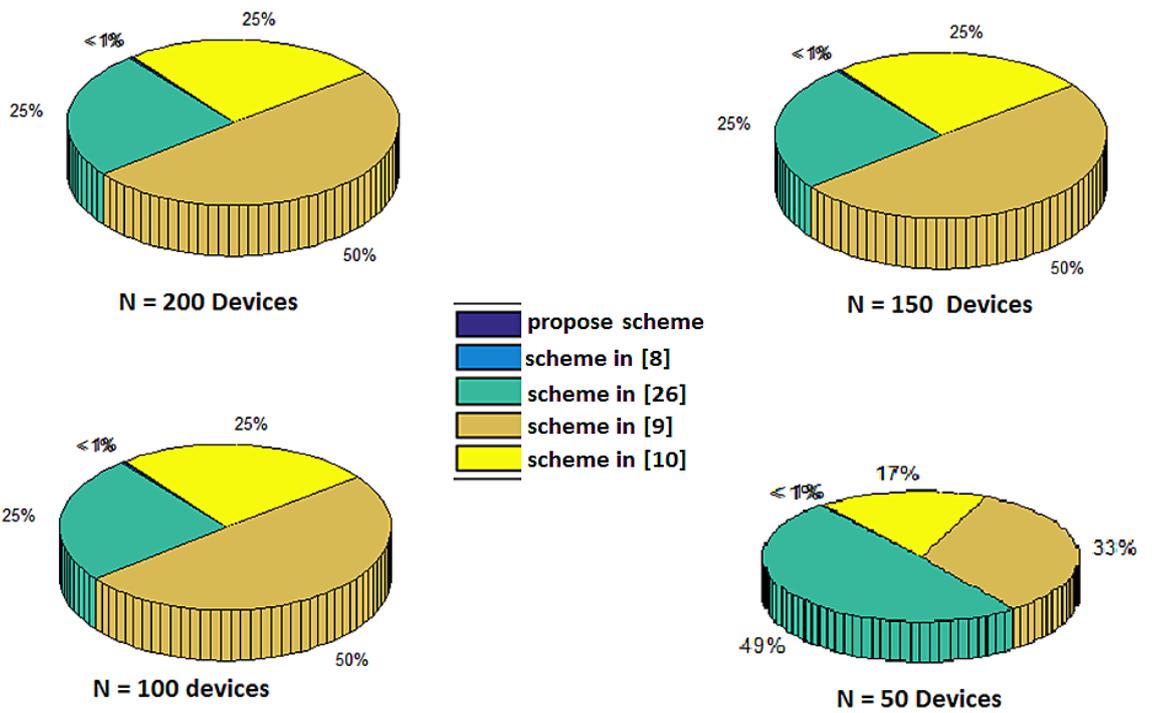


Figure 4.10: Aggregation cost percentages for the developed scheme, and the schemes in [1], [2], [3], and [4]

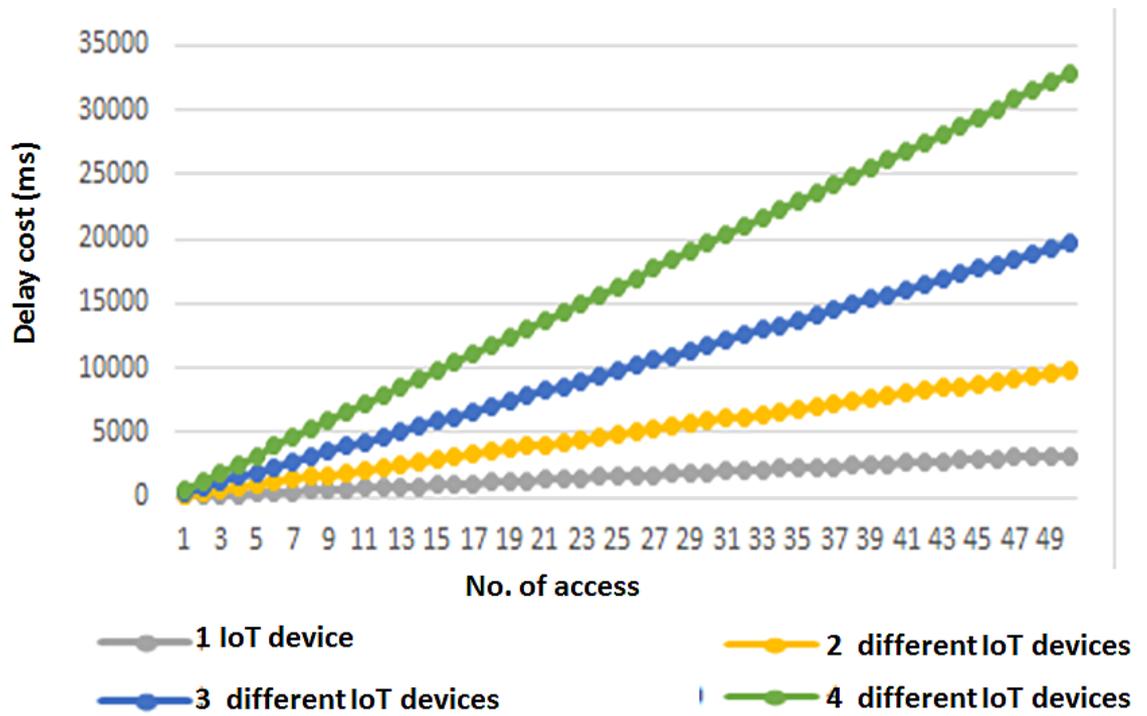


Figure 4.11: Comparison of access delays

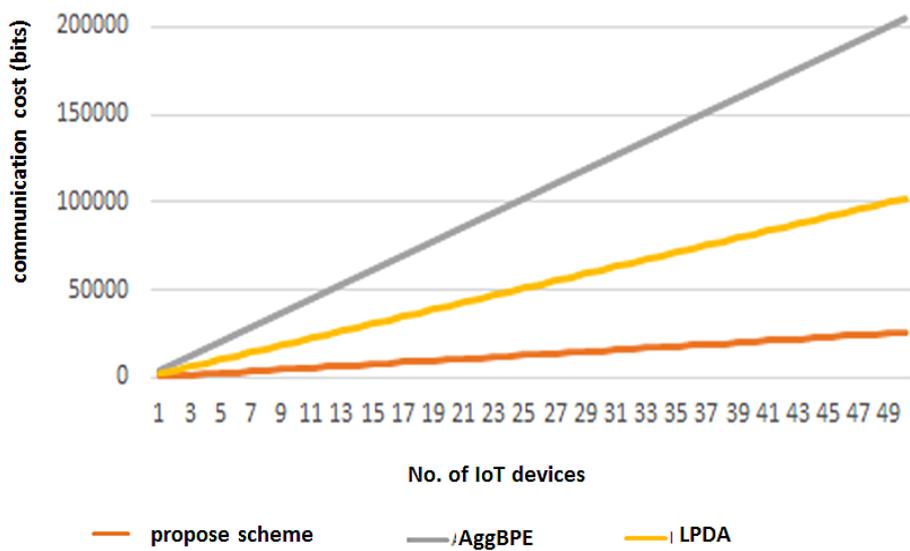


Figure 4.12: Communication overheads

# Chapter 5

## Overview of System, Conclusion, and Futuristic Road-maps

### 5.1 Overview of System

Our work presents technological capabilities that could be used to monitor and improve agriculture, especially in developing countries like Nigeria. These include the engineering and technical know-how to develop UAVs for precision agriculture, modular multisensor data acquisition systems (for collecting agricultural data), reliable mid-range and long-range communication technologies (for relaying data to base-stations and data storage servers), and security schemes for a secure data exchange, algorithms and software (for processing information both at the collection point, mid-way or the end-target).

### 5.2 Conclusion

In this work, we developed a cybertronic data harvester and aggregation scheme for efficient precision farming. These also include the development of a UAV for deploying the onboard data harvesting electronic communication system to relevant farm locations. Experiments were performed to ascertain the applicability of the system to modern agriculture. Preliminary results show that the developed

system is robust and effective for real-time precision agriculture. In addition, flight tests show that the developed UAV has effective hovering capability over a particular location while the FWSN gateway uploads the necessary data. Meanwhile, the aggregation scheme and the associated edge computing\communication system have high reliability to secure the telemetry of farm data while reducing communication traffic between the farm and the cloud server. In the future, we will look at how this can be extended to other applications of telemetry, remote sensing, and fish farming. Therefore, we hope that the present work incites interest in the systems developed and the general applications of electronics and computers in agriculture. In view of this, we envisage that this work would also improve scientific collaborations between scientists, engineers, and farmers to fully exploit the capabilities of the developed systems.

The prototype has a complete CAD design that can be mass produced for the readily available market especially the youth that want to farm but believe that some of the farm activities are dirty and not lucrative. Moreover, the project greatly encourages local content in terms of the manpower and materials used in building the project. In terms of materials, the edge (Unmanned aerial vehicle), air quality and farm monitoring systems are constructed using local and readily available materials such as aluminum and used plastic materials. Also, in terms of manpower, the project was built as part of outcome-based education for both undergraduate and postgraduate students. Mass production of the project will bring down the cost to =N=20,000. This cost reduction is due to the fact that most of the materials used can be sourced locally. The Nigerian Communications Commission can facilitate a meeting point between we the innovator and the investors who are ready to help in the commercialization of the product. Also, NCC can facilitate a workable and profit oriented business model that will make the mass production and commercialization a reality. The success rate of the project is about 85%, however, the only thing that can make it fails is the absence of a good UAV pilot.

### 5.3 Futuristic Road-maps

Palpably, the present work suggests the need for a new research company that would be dedicated to solving technical problems in the area of telecommunications, cybersecurity, cybertronics, and control of technical systems (such as farm machinery) to facilitate research and production of various commercial end products. These include but are not limited to:

1. Re-programmable real-time plug-and-play transceiver module that can be procured by scientists, geographers, agriculturists, etc. for various data communications applications. For instance, the sensors node in Figure ?? is a market-ready product that can be used to set up a weather station and telemetry for collecting agricultural data. Also, the telematic function of these devices can be used to develop various telerobotic and remote control systems for scientific and industrial applications.
2. Unmanned aerial vehicles that can be adopted for various applications such as security drones for environmental and industrial monitoring.
3. Agricultural robots and remotely controlled machinery that can be engaged for various farming operations.
4. Software solutions and training for adapting the developed systems and products to productive applications.

# REFERENCES

- [1] Y.-C. L. Chien-Ming Chen, Yue-Hsun Lin and H.-M. Sun, “Rcda: Recoverable concealed data aggregation for data integrity in wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, p. 727–734, 2012.
- [2] S. C. Y. Lin and H. Sun, “Cdama: Concealed data aggregation scheme for multiple applications in wireless sensor networks,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 7, pp. 1471–1483, 2013.
- [3] K. Shim and C. Park, “A secure data aggregation scheme based on appropriate cryptographic primitives in heterogeneous wireless sensor networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 8, pp. 2128–2139, 2015.
- [4] S. L.-C. J. Zhong, H. and Y. Xu, “An efficient and secure recoverable data aggregation scheme for heterogeneous wireless sensor networks,” *Journal of Parallel and Distributed Computing*, pp. 1–12, 2018.
- [5] E. R. H. Jr. and C. S. T. Daughtry, “What good are unmanned aircraft systems for agricultural remote sensing and precision agriculture?” *International Journal of Remote Sensing*, vol. 39, no. 15-16, pp. 5345–5376, 2018.
- [6] M. R. Haque, M. Muhammad, D. Swarnaker, and M. Arifuzzaman, “Autonomous quadcopter for product home delivery,” *2014 International Conference on Electrical Engineering and Information & Communication Technology*, pp. 1–5, 2014.
- [7] D. Gheorghita, I. Vintu, L. Mirea, and C. Braescu, “Quadcopter control system,” *2015 19th International Conference on System Theory, Control and Computing (ICSTCC)*, pp. 421–426, 2015.
- [8] R. Yang and X. Wang, “Vision control system for quadcopter,” in *2013 IEEE Third International Conference on Information Science and Technology (ICIST)*, 2013, pp. 261–264.

- [9] H. Bayerlein, M. Theile, M. Caccamo, and D. Gesbert, “Multi-uav path planning for wireless data harvesting with deep reinforcement learning,” *IEEE Open Journal of the Communications Society*, vol. 2, pp. 1171–1187, 2021.
- [10] O. O. Olakanmi and P. Adama, “An efficient multipath routing protocol for decentralized wireless sensor networks for mission and safety-critical systems,” *International Journal of Sensors, Wireless Communications and Control 2020*, vol. 10, 2020, <https://doi.org/10.2174/2210327909666190531113558>.
- [11] W. Gong, M. Zhang, X. Yang, J. Li, N. Zhang, and K. Long, “Waras: An adaptive wsn multipath selection model inspired by metabolism behaviors of escherichia coli,” in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, 2015, pp. 1–6.
- [12] M. Saikia, U. K. Das, and M. Hussain, “Secure energy aware multi-path routing with key management in wireless sensor network,” *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, pp. 310–315, 2017.
- [13] S. Kim, H. Cho, T. Yang, C. Kim, and S. Kim, “Low-cost multipath routing protocol by adapting opportunistic routing in wireless sensor networks,” in *2017 IEEE Wireless Communications and Networking Conference, WCNC 2017, San Francisco, CA, USA, March 19-22, 2017*. IEEE, 2017, pp. 1–6. [Online]. Available: <https://doi.org/10.1109/WCNC.2017.7925730>
- [14] Y. Lu, G. Wang, W. Jia, and S. Peng, “Multipath-based segment-by-segment routing protocol in manets,” in *2008 The 9th International Conference for Young Computer Scientists*, 2008, pp. 527–532.
- [15] M. Kim, E. Jeong, Y.-C. Bang, S. Hwang, and B. Kim, “Multipath energy-aware routing protocol in wireless sensor networks,” in *2008 5th International Conference on Networked Sensing Systems*, 2008, pp. 127–130.
- [16] S. K. Gupta, P. Kuila, and P. K. Jana, “Energy efficient multipath routing for wireless sensor networks: A genetic algorithm approach,” in *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2016, pp. 1735–1740.
- [17] R.-S. S. A. H. M. A. R. Al Ahasan, M. and M. Hossain, “Arduino-based real time air quality and pollution monitoring system,” *International Journal of Innovative Research in Computer Science Technology (IJIRCST)*, vol. 6, no. 4, 2018.

- [18] M. R.-B. D. . S. P. G. Kaur, N., “Air quality monitoring system based on arduino microcontroller,” *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 5, no. 6, pp. 9635–9646, 2016.
- [19] K.-Z. M. A. A. M. M. S. A. Shah, H. N. and P. Rane, “Iot based air pollution monitoring system,” *International Journal of Scientific Engineering Research*, vol. 9, no. 2, pp. 62–66, 2018.
- [20] J. R. K., “Air quality sensing and reporting system using iot,” in *2020 Second international conference on inventive research in computing applications (ICIRCA)*, 2020, pp. 790–793.
- [21] . J.-A. Kumar, S., “Air quality monitoring system based on iot using raspberry pi,” in *2017 International conference on computing, communication and automation (ICCCA)*, 2017, pp. 1341–1346.
- [22] R. Kiruthika and A. Umamakeswari, “Low cost pollution control and air quality monitoring system using raspberry pi for internet of things,” in *2017 International conference on energy, communication, data analytics and soft computing (ICECDS)*, 2017, pp. 2319–2326.
- [23] T.-I. G. E. Marinov, M. B. and G. Nikolov, “Air quality monitoring in urban environments,” in *2016 39th International Spring Seminar on Electronics Technology (ISSE)*, 2016, pp. 443–448.
- [24] M.-Y. M. S. K. M. Gunawan, T. S. and H. Mansor, “Design and implementation of portable outdoor air quality measurement system using arduino,” *International Journal of Electrical Computer Engineering*, vol. 8, no. 1, pp. 2088–8708, 2018.
- [25] M.-H. G. T. S. Abiduzzaman, S. M. and R. Ahmad, “Real-time outdoor air quality monitoring system,” in *2021 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA)*, 2021, pp. 140–145.
- [26] O. Olakanmi, “Secure and privacy-oriented obfuscation scheme for smart metering in smart grid via dynamic aggregation and lightweight perturbation,” *International Journal of Information Privacy, Security and Integrity (IJIPSI)*, vol. 3, no. 1, 2017.
- [27] L.-K. K. S. L. T. Z. L. D. . S. C. S. Tan, H. C., “Chameleon: A blind double trapdoor hash function for securing ami data aggregation,” in *IEEE 4th World Forum on Internet of Things (WF-IoT)*, 2018.
- [28] H. K.-L. A. H. Lu, R. and A. A. Ghorbani, “A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot,” *IEEE Access*, vol. 5, p. 3302–3312, 2017.

- [29] Z. X.-G. W. L. X. Yang, Y. and V. Chang, "Privacy-preserving smart iot-based healthcare big data storage and self-adaptive access control system," *Information Sciences*, vol. 11, no. 10, 2018.
- [30] O. Olakanmi and A. Dada, "Felas: Fog enhanced look ahead secure framework with separable data aggregation scheme for efficient information management in internet of things networks," *Journal of Applied Security Research*, 2019.
- [31] A. Z.-W. J. R. Li, D. and A. Sanchez, "Efficient authentication scheme for data aggregation in smart grid with fault tolerance and fault diagnosis," in *2012 IEEE PES Innovative Smart Grid Technologies (ISGT)*, 2012.
- [32] L. B. Li, F. and P. Liu, "Secure information aggregation for smart grids using homomorphic encryption," in *2010 First IEEE International Conference on Smart Grid Communications*, 2010.
- [33] L. Zhang and J. Zhang, "Epprd: An efficient privacy-preserving power requirement and distribution aggregation scheme for a smart grid sensors," *Sensors (Basel)*, vol. 17, no. 8, 2017.
- [34] M. Benyeogor, A. Olatunbosun, and S. Kumar, "Airborne system for pipeline surveillance using an unmanned aerial vehicle," *European Journal of Engineering and Technology Research*, vol. 5, no. 2, pp. 178–182, Feb. 2020. [Online]. Available: <https://www.ejers.org/index.php/ejers/article/view/1761>
- [35] H. K.-L. A. H. Lu, R. and A. A. Ghorbani, "A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced iot," *IEEE Access*, vol. 5, p. 3302–3312, 2017.