



**INTEGRATION OF SENSORS AND TELEMETRY SYSTEM FOR
AN RC AIRSHIP USING CAD AND EMBEDDED SYSTEM**



**BACHELOR OF MECHANICAL ENGINEERING TECHNOLOGY
(AUTOMOTIVE TECHNOLOGY) WITH HONOURS**

2024



Faculty of Mechanical Technology and Engineering



**INTEGRATION OF SENSORS AND TELEMETRY SYSTEM FOR AN
RC AIRSHIP USING CAD AND EMBEDDED SYSTEM**

SHARIZAN BIN YUNUS

**Bachelor of Mechanical Engineering Technology (Automotive Technology) with
Honours**

2024

**INTEGRATION OF SENSORS AND TELEMETRY SYSTEM FOR AN RC
AIRSHIP USING CAD AND EMBEDDED SYSTEM**

SHARIZAN BIN YUNUS

**A thesis submitted
in fulfillment of the requirements for the degree of
Bachelor of Mechanical Engineering Technology (Automotive Technology) with
Honours**



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2024

BORANG PENGESAHAN STATUS LAPORAN PROJEK SARJANA MUDA

TAJUK: INTEGRATION OF SENSORS AND TELEMETRY SYSTEM FOR AN RC AIRSHIP USING CAD AND EMBEDDED SYSTEM

SESI PENGAJIAN: 2023-2024 Semester 1

Saya **SHARIZAN BIN YUNUS** mengaku membenarkan tesis ini disimpan di Perpustakaan Universiti Teknikal Malaysia Melaka (UTeM) dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hak milik Universiti Teknikal Malaysia Melaka dan penulis.
2. Perpustakaan Universiti Teknikal Malaysia Melaka dibenarkan membuat salinan untuk tujuan pengajian sahaja dengan izin penulis.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

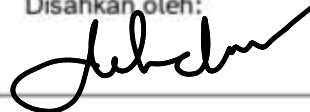
TERHAD (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia sebagaimana yang termaktub dalam AKTA RAHSIA RASMI 1972)

SULIT (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

TIDAK TERHAD



Disahkan oleh:



Alamat Tetap:

KAMPUNG MASAK, KENINGAU, 89000, SABAH

Cop Rasmi:

TS. MUHAMMED NOOR B. HJ HASHIM
Jurutera Pengajar Kanan
Universiti Teknikal Malaysia Melaka

Tarikh: 12/01/2024

Tarikh: 9/2/2024

** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan PSM ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this Choose an item. entitled “ Integration of Sensors and Telemetry Systems for an RC Airship using CAD and Embedded Systems ” is the result of my own research except as cited in the references. The Choose an item. has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature

:

Name

:

SHARIZAN BIN YUNUS

Date

:

12/01/2024



UTeM

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

APPROVAL

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the Bachelor of Mechanical Engineering Technology (Automotive Technology) with Honours.

Signature :



Supervisor Name :

TS. MUHAMMED NOOR B. HJ HASHIM
Jurutera Pengajar Kanan
Universiti Teknikal Malaysia Melaka

Date :

9/2/2024



اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATION

With humility, I dedicate this with sincere appreciation to my family, friends, and supervisor, for being the driving force behind my achievements. To my beloved family, friends, and supervisor, Ts. Muhammad Noor Bin Hashim, With heartfelt gratitude, I would like to express my appreciation and thanks to all of you in my project report on the Integration of Sensors and Telemetry Systems for an RC Airship using CAD and Embedded Systems. To my dear family, thank you for your unwavering support, endless encouragement, and unwavering belief throughout this journey. You have been my source of inspiration and strength at every step of the way. Thank you for always being by my side and providing invaluable moral support. To my loyal friends, thank you for your unparalleled friendship. Each one of you has provided the spirit and positive energy that propelled me forward. I appreciate every sharing, support, and joyful moments we have had together throughout this time. To my supervisor, Ts. Muhammad Noor Bin Hashim, thank you for your guidance, advice, and invaluable knowledge that you have shared throughout this project. I greatly appreciate your commitment and patience in helping me overcome challenges and achieve the goals of this project. Your influence as a supervisor has had a significant impact on my professional development. All the success and accomplishments in this project are the result of the cooperation, support, and guidance from all of you. Without your presence, this project would not have reached such heights. Thank you for being unwavering supporters and believing in my potential.

ABSTRACT

This study aims to enhance the capabilities of remote-controlled (RC) airships through the integration of advanced sensors and telemetry systems. The primary objective is to develop a custom sensor board and telemetry module that can significantly improve the airship's situational awareness and data collection abilities during flight operations. To accomplish this, a comprehensive research process was conducted to understand the existing market for sensors and telemetry systems in the field of RC airships. After careful consideration of various factors such as performance, compatibility, and data collection capabilities, a combination of sensors and microcontrollers was chosen. The custom sensor board consists of an accelerometer sensor and an altimeter sensor, which provide crucial data on the airship's acceleration, orientation, and altitude. These sensors enable the airship to gather real-time information about its flight dynamics and positioning in the airspace. The telemetry module is based on the ESP32 microcontroller, known for its reliable communication capabilities. It allows for remote access and control of the airship, as well as seamless transmission of sensor data to a ground station or control center. By implementing the proposed custom sensor board and telemetry system, RC airship operators and researchers will benefit from enhanced control, communication, and monitoring capabilities. This technology empowers operators to remotely access and control the airship, facilitating safer and more efficient flight operations. Additionally, the improved situational awareness and data collection capabilities enable researchers to gather valuable data for analysis, experimentation, and further advancements in the field of RC airships.

ABSTRAK

Kajian ini bertujuan untuk meningkatkan keupayaan kapal udara terkawal jarak jauh (RC) melalui integrasi sensor dan sistem telemetri canggih. Objektif utama adalah untuk membangunkan papan sensor khusus dan modul telemetri yang dapat meningkatkan kesedaran keadaan dan keupayaan pengumpulan data kapal udara semasa operasi penerbangan. Untuk mencapai ini, satu proses penyelidikan menyeluruh dijalankan untuk memahami pasaran sensor dan sistem telemetri sedia ada dalam bidang kapal udara RC. Setelah mempertimbangkan pelbagai faktor seperti prestasi, keserasian, dan keupayaan pengumpulan data, gabungan sensor dan mikropemproses dipilih. Papan sensor khusus terdiri daripada sensor pemercepat dan sensor altimeter, yang menyediakan data penting tentang pecutan, orientasi, dan ketinggian kapal udara. Sensor-sensor ini membolehkan kapal udara mengumpul maklumat secara masa nyata tentang dinamik penerbangan dan kedudukannya di ruang udara. Modul telemetri berdasarkan mikropemproses ESP32, yang dikenali dengan kebolehpercayaan komunikasinya. Ia membolehkan akses dan kawalan kapal udara secara jauh, serta pemindahan data sensor yang lancar ke stesen darat atau pusat kawalan. Dengan melaksanakan papan sensor khusus dan sistem telemetri yang dicadangkan, operator dan penyelidik kapal udara RC akan mendapat manfaat daripada peningkatan dalam kawalan, komunikasi, dan keupayaan pemantauan. Teknologi ini memberi kuasa kepada operator untuk mengakses dan mengawal kapal udara secara jauh, memudahkan operasi penerbangan yang lebih selamat dan cekap. Selain itu, kesedaran keadaan yang ditingkatkan dan keupayaan pengumpulan data membolehkan penyelidik mengumpul data berharga untuk analisis, eksperimen, dan kemajuan lanjut dalam bidang *kapal udara RC*.

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Gracious, the Most Merciful

First and foremost, I would like to thank and praise Allah the Almighty, my Creator, my Sustainer, for everything I received since the beginning of my life. I would like to extend my appreciation to the Universiti Teknikal Malaysia Melaka (UTeM) for providing the research platform. Thank you also to the Malaysian Ministry of Higher Education (MOHE) for the financial assistance.

My utmost appreciation goes to my supervisor, Ts. Muhammad Noor Bin Hashim for all his support, advice and inspiration. His constant patience for guiding and providing priceless

insights will forever be remembered.

Last but not least, from the bottom of my heart a gratitude to my beloved parents for the encouragements and who have been the pillar of strength in all my endeavors. My eternal love also to all my friends for their patience and understanding. Finally, thank you to all the individual(s) who had provided me the assistance, support and inspiration to embark on my study.

TABLE OF CONTENTS

	PAGE
DECLARATION	
APPROVAL	
DEDICATION	
ABSTRACT	i
ABSTRAK	ii
ACKNOWLEDGEMENTS	iii
TABLE OF CONTENTS	iv
LIST OF TABLES	vi
LIST OF FIGURES	vii
LIST OF SYMBOLS AND ABBREVIATIONS	x
LIST OF APPENDICES	xii
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	1
1.3 Research Objective	2
1.4 Scope of Research	2
CHAPTER 2 LITERATURE REVIEW	3
2.1 Introduction	3
2.2 Importance of Sensor Integration in RC Airships / Components	4
2.3 Telemetry Systems for RC Airships	6
2.3.1 Introduction to telemetry systems and their significance in remote data transmission	6
2.3.2 Examination of different telemetry technologies (e.g., Wi-Fi, Bluetooth, RF) and their suitability for RC airships	7
2.3.3 Analysis of existing telemetry systems used in RC Airships and their performance	8
2.4 Design and Development of Custom Sensor Boards	9
2.4.1 Overview of the design process for custom sensor boards using CAD software	9
2.4.2 The considerations for sensor selection, placement, and integration on the board	10
2.4.3 Analysis of different PCB design techniques and best practices for sensor board development	11

2.5	Embedded Systems for Sensor Integration	13
2.5.1	Explanation of embedded systems and their role in data acquisition and processing	13
2.5.2	Overview of microcontrollers and single-board computers suitable for RC airship applications	13
2.5.3	Exploration of programming languages, frameworks, and development tools for embedded systems	14
2.6	Integration Techniques for Sensor Boards and Telemetry Modules	16
2.7	Performance Evaluation of Integrated Systems	19
2.8	Summary of Research Gap	24
CHAPTER 3 METHODOLOGY		25
3.1	Introduction	25
3.2	Background Research	26
3.3	Design Development	27
3.3.1	Design Concept	27
3.3.2	Component Design	28
3.3.3	Sensors	30
3.3.4	ESP32	35
3.3.5	Wiring	36
3.3.6	Program Code	38
3.3.7	Prototype	40
3.4	Product Testing	40
3.5	Summary	42
CHAPTER 4 RESULTS AND DISCUSSION		43
4.1	Introduction	43
4.2	Result and Discussion	43
4.2.1	Actual Data at Kolej Kediaman Satria (Kasuri)	43
4.3	Experimental Data at Kolej Kediaman Satria (Kasturi)	44
4.3.1	Data collection using direct USB cable connection	44
4.3.2	Blynk IoT	63
4.3.3	Data collection using IoT	66
4.4	Summary	86
CHAPTER 5 CONCLUSION AND RECOMMENDATIONS		89
5.1	Conclusion	89
5.2	Recommendations	89
REFERENCES		91
APPENDICES		93

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1	Main open-source SBMs (Single-Board Microcontroller) in the market. (Álvarez et al., 2021)	11
Table 2.2	GPS recording for the third test flight	21
Table 2.3	Language execute speed	22
Table 3.1	SN-ADXL335-CY Details	31
Table 3.2	GY-68 Details	34
Table 3.3	Wiring Connectivity	37
Table 4.1	Dataset for Altitude and Temperature with direct USB cable connection at rest condition	45
Table 4.2	Dataset for Tilt Angle with direct USB cable connection at rest condition	51
Table 4.3	Dataset for Altitude and Temperature with direct USB cable connection at moving condition	54
Table 4.4	Dataset for Tilt Angle with direct USB cable connection at moving condition	59
Table 4.5	Dataset for Altitude and Temperature with Blynk IoT at rest condition	67
Table 4.6	Dataset for Tilt Angle with Blynk IoT at rest condition	72
Table 4.7	Dataset for Altitude and Temperature with Blynk IoT at moving condition	76
Table 4.8	Dataset for Tilt Angle with Blynk IoT at moving condition	83

LIST OF FIGURES

FIGURE	TITLE	PAGE
Figure 2.1	Surface Surveillance Sensors and Applications Enabled with Multi-Sensor Data Processing (Fitch et al., 2002)	6
Figure 2.2	Data Flow Diagram (Biju & Pant, 2017)	9
Figure 2.3	Single Board Architectures classification and main platforms.	14
Figure 2.4	The MakeCode and CODAL Architecture (Devine et al., 2018)	15
Figure 2.5	Basic structure diagram of calibration system (Hua et al., 2021)	19
Figure 3.1	Flow chart of the development of the custom sensors and telemetry systems	26
Figure 3.2	Flow Chart of the system work flow	29
Figure 3.3	SN-ADXL355-CY Accelerometer Sensor	30
Figure 3.4	GY-68 BMP180 Barometric Pressure Sensor	33
Figure 3.5	ESP32	35
Figure 3.6	Accelerometer sensor to the ESP32	36
Figure 3.7	Barometric sensor to the ESP32	37
Figure 3.8	Program Code	39
Figure 3.9	Program Code Continuation	39
Figure 3.10	Custom Sensor Prototype	40
Figure 3.11	Product Testing Result	41
Figure 4.1	Altitude actual reading at Kolej Kediaman Satria (Kasturi)	44
Figure 4.2	Test 1 Direct Connection for Altitude at rest condition	46
Figure 4.3	Test 1 Direct Connection for Temperature at rest condition	47

Figure 4.4 Test 2 Direct Connection for Altitude at rest condition	48
Figure 4.5 Test 2 Direct Connection for Temperature at rest condition	48
Figure 4.6 Test 3 Direct Connection for Altitude at rest condition	49
Figure 4.7 Test 3 Direct Connection for Temperature at rest condition	50
Figure 4.8 Test 1 Tilt Angle at rest condition for Direct Connection	52
Figure 4.9 Test 2 Tilt Angle at rest condition for Direct Connection	53
Figure 4.10 Test 3 Tilt Angle at rest condition for Direct Connection	53
Figure 4.11 Test 1 Direct Connection for Altitude at moving condition	55
Figure 4.12 Test 1 Direct Connection for Temperature at moving condition	56
Figure 4.13 Test 2 Direct Connection for Altitude at moving condition	56
Figure 4.14 Test 2 Direct Connection for Temperature at moving condition	57
Figure 4.15 Test 3 Direct Connection for Altitude at moving condition	58
Figure 4.16 Test 1 Direct Connection for Temperature at moving condition	59
Figure 4.17 Test 1 Tilt Angle at moving condition for Direct Connection	61
Figure 4.18 Test 2 Tilt Angle at moving condition for Direct Connection	62
Figure 4.19 Test 3 Tilt Angle at moving condition for Direct Connection	63
Figure 4.20 Blynk Datasreams Setup	64
Figure 4.21 Blynk Dashboard Setup	64
Figure 4.22 Blynk Additional Function	65
Figure 4.23 Blynk Wireless Notifications	65
Figure 4.24 Test 1 Blynk IoT Altitude at rest condition	68
Figure 4.25 Test 1 Blynk IoT Temperature at rest condition	68
Figure 4.26 Test 2 Blynk IoT Altitude at rest condition	69
Figure 4.27 Test 2 Blynk IoT Temperature at rest condition	70

Figure 4.28 Test 3 Blynk IoT Altitude at rest condition	70
Figure 4.29 Test 3 Blynk IoT Temperature at rest condition	71
Figure 4.30 Test 1 Tilt Angle at rest condition for Blynk IoT	73
Figure 4.31 Test 2 Tilt Angle at rest condition for Blynk IoT	74
Figure 4.32 Test 3 Tilt Angle at rest condition for Blynk IoT	75
Figure 4.33 Test 1 Blynk IoT Altitude at moving condition	77
Figure 4.34 Test 1 Blynk IoT Temperature at moving condition	78
Figure 4.35 Test 2 Blynk IoT Altitude at moving condition	79
Figure 4.36 Test 2 Blynk IoT Temperature at moving condition	80
Figure 4.37 Test 3 Blynk IoT Altitude at moving condition	81
Figure 4.38 Test 3 Blynk IoT Temperature at moving condition	82
Figure 4.39 Test 1 Tilt Angle at driving condition for Blynk IoT	84
Figure 4.40 Test 2 Tilt Angle at driving condition for Blynk IoT	85
Figure 4.41 Test 3 Tilt Angle at driving condition for Blynk IoT	86

LIST OF SYMBOLS AND ABBREVIATIONS

RC	-	Remotely Controlled
CAD	-	Computer Aided Design
NAS	-	National Airspace System
ADSB	-	Automatic Dependent Surveillance Broadcast
FAA	-	Federal Aviation Administration
GPS	-	Global Positioning System
OSC	-	Open Sound Control data
PD	-	Pure Data
PCB	-	Printed Circuit Board
INS	-	Inertial Navigation System
SBA	-	Single-Board Architectures
IDE	-	Integrated Development Environment
DAL	-	Data acquisition systems
WSN	-	Wireless Sensor Networks
OS	-	Operating System
IoT	-	Internet of Things
MCU	-	Microcontroller Units
USB	-	Universal Serial Bus
UAS	-	Unmanned Aerial System
UAV	-	Unmanned Aerial Vehicles
UART	-	Universal Asynchronous Receiver-Transmitter
I2C	-	Inter-Integrated Circuit
SPI	-	Serial Peripheral Interface
UART	-	Universal Asynchronous Receiver-Transmitter
SCLK	-	Clock Line
CS	-	Chip Select Lines
MOSI	-	Master Output Slave Input
MISO	-	Master Input Slave Output
IIoT	-	Industrial Internet of Things

- VM - Virtual Machine
RF - Radio Frequency



LIST OF APPENDICES

APPENDIX	TITLE	PAGE
APPENDIX A	List of distribution network parameters.	93



CHAPTER 1

INTRODUCTION

1.1 Background

The use of remotely controlled (RC) airships has gained popularity in recent years for a variety of applications, such as research, surveillance, and environmental monitoring. These airships can be equipped with various sensors to collect data and transmit it back to a ground station in real-time. However, the accuracy and range of the sensors and telemetry systems are critical for effective data collection and analysis. This project aims to improve the situational awareness and data collection capabilities of an RC airship by integrating custom-designed sensor boards and telemetry modules with the airship's control system. This will enable the airship to collect and transmit more accurate and detailed data for research or surveillance purposes. The integration will involve the use of CAD software and embedded systems design, and the performance of the system will be tested in different flight scenarios to ensure its functionality and reliability.

1.2 Problem Statement

The current challenge faced by existing RC airships utilized in research roles lies in their inability to provide adequate situational awareness and collect data effectively, limiting their performance across various flight scenarios. To address this deficiency and enhance pilot awareness, there is an urgent requirement to integrate advanced sensors and telemetry systems into the airship's control system. However, the integration process, which necessitates specialized skills in Computer Aided Design (CAD) and embedded systems,

may pose accessibility issues for users. This project aims to address these challenges by developing a custom sensor board and telemetry module and incorporating them seamlessly into the RC airship's control system prior to installation. The primary objective is to significantly enhance pilot awareness during operations.

1.3 Research Objective

The main objective of this research is to improve its situational awareness and data collection capabilities. Specifically, the objectives are as follows:

- a) Design and develop a custom sensor board and telemetry module that can be integrated with an RC airship's control system to improve its situational awareness and data collection capabilities.
- b) Create a telemetry module that facilitates real-time data transmission from the sensor board to the ground control station.
- c) Test and evaluate the performance of the integrated system in different scenarios to determine its effectiveness in enhancing the airship's situational awareness and data collection capabilities.

1.4 Scope of Research

The scope of this research are as follows:

- Development of the sensor board and telemetry module using CAD software and embedded systems.
- Develop a telemetry module capable of facilitating real-time data transmission from the sensor board to the ground control station.
- Perform testing to validate performance in various temperatures and altitudes.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

According to (Motiwala *et al.*, 2013), applications needing a robust, affordable, long-lasting platform with minimum maintenance are ideal for airships, noise and vibration levels. Airships are frequently used for aerial surveillance and can serve as a platform for placing sensors for research projects like gathering samples of air or gas, agriculture uses including forecasting the weather and applying pesticides. For some specific missions, such as prolonged aerial observation of any activity occurring on the ground (such as aerial coverage of sporting events or security patrolling), long endurance loiter for product promotion, or precise dropping of payload (such as showering of flowers during religious functions or inauguration/closing ceremonies), a remotely controlled (RC) airship may be much more suitable than a remotely controlled aircraft. It may be used everywhere since, unlike aeroplanes, it doesn't need a runway to take off and land.

The five basic parts of an RC airship are the gondola, the propulsion system, the stabiliser and fins, the remote control system, and the envelope. Based on the performance and operational criteria supplied by the user, the approach determines the size and configuration of an RC airship. The design of RC airships has several characteristics with that of RC aircraft, such as the gondola's resemblance to the fuselage of a low speed RC aircraft. But when creating other components, there are certain distinctions and more alternatives. For instance, the propulsion system of RC airships can be identical to that of

RC aeroplanes, with propellers providing thrust and being powered by either IC engines or DC motors with onboard batteries (Motiwala *et al.*, 2013).

(Motiwala *et al.*, 2013) stated that, one of the key parts of the Airship is the gondola. It is essentially a box-shaped structure that bears the payload and houses all of the electrical machinery. In comparison to the drag of the envelope, the drag produced by the gondola is relatively small and is thought to be negligible. More versatility in terms of weight, size, storage capacity, and structure is provided by the custom-designed gondola. Despite the fact that curved elements are more difficult to fabricate than straight parts, curved surfaces provide airships a smoother, more aerodynamic finish. Therefore, it is essential to create a system that can be simply made and has a superior aesthetic appearance. The ratio of the suspended mass of the gondola to the total mass of the airship rises with airship size, making the gondola more of a burden than a support. Gondolas are divided into two categories based on their structures: open and closed.

2.2 Importance of Sensor Integration in RC Airships / Components

Different kinds of surveillance resources are employed for tracking as aircraft move through the route, terminal, and surface stages of flight. Historically, radar has been utilised to give air traffic controllers access to surveillance and situational awareness. Recent years have seen the introduction of new surveillance technologies into the National Airspace System (NAS), including multilateration and automatic dependent surveillance broadcast (ADS-B). Currently, multisensor data processing or sensor fusion techniques are being used to integrate these new surveillance sources with radar systems. Data from every sensor is combined during multisensor data processing to create a single location estimate for each aircraft. While enabling new air and surface safety, capacity and efficiency applications for

airports, airlines and pilot, a multi-sensor data processing architecture gives controllers highly accurate air and surface situational awareness (Fitch, Southwick and Morganti, 2002).

A multisensor data processor combines short-range air surveillance radar, surface movement radar, multilateration, and ADSB for surface monitoring. The Federal Aviation Administration (FAA) is putting this design into practise through two programmes: the FAA Safe Flight 21 Programme and the Airport Surface Detection Equipment Model -X (ASDE-X) Programme for air traffic control. (Fitch, Southwick and Morganti, 2002) stated that the architecture for multisensor data processing provides the opportunity to:

- air traffic controllers with highly accurate surveillance in all weather conditions.
- pilots and vehicle drivers with situational awareness via Traffic Information System - Broadcast (TIS-B) services.

airports and airlines with customize data sets, for automating operational applications including gate management, landing and parking billing, noise monitoring and data recording for process analysis and procedural review via a secure communication gateway.

Figure 2.1 depicts the surface surveillance sensor sources input to the multi-sensor data processor and end user applications.

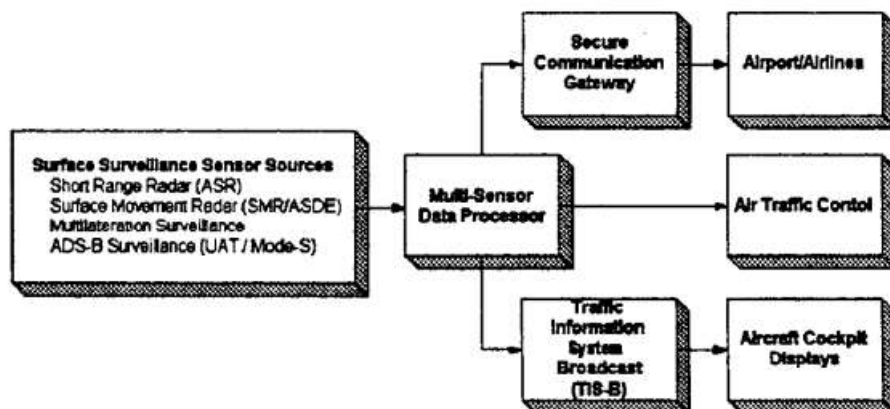


Figure 2.1 Surface Surveillance Sensors and Applications Enabled with Multi-Sensor Data Processing (Fitch et al., 2002)

Meanwhile, (Biju and Pant, 2017) used infrared sensors In order to identify walls, floors, ceilings, and any other obstacles that the airship may run into during its autonomous flight, six proximity sensors were connected to the airship envelope, one on each side (viz., front, back, top, bottom, left, and right). (Biju and Pant, 2017) employed ultrasound sensors, which produced an analogue voltage signal that was then processed by integrated circuitry on each sensor module.

Compass, accelerometers, inclinometers, and gyroscopes are among the inertial navigation sensors utilised for flight path execution together with a GPS (global positioning system) receiver in the internal sensor suite. The operator on the ground can use the cameras mounted on the airship's gondola to transmit aerial photos and for visual navigation based on topographic features of the terrain (Elfes *et al.*, 1998).

2.3 Telemetry Systems for RC Airships

2.3.1 Introduction to telemetry systems and their significance in remote data transmission

According to (Vilkov *et al.*, 2018), the purpose of the modern automated control system for spacecraft is to guarantee the appropriate operation of the onboard systems for the duration of the spacecraft's operation. The automated control system for spacecrafts consists of a number of onboard and ground-based technological process control facilities with the required software, such as:

- the apparatus for regulating the channels of an onboard radio complex, which is part of the onboard control complex.
- the ground control software.

In such a system, it is urgently necessary to discover technological process violations as rapidly as is humanly possible, from a simple fuse blowing to the early detection of onboard equipment breakdown using telemetry data analysis. The creation of technical complexes for the automation of telemetry data receiving, processing, and analysis seems to be feasible, but, as people are unable to properly analyse an infinite amount of information. High levels of automation and the system will enable operators to work more effectively while spending less time gathering the information they need to keep the spacecraft operating steadily.

2.3.2 Examination of different telemetry technologies (e.g., Wi-Fi, Bluetooth, RF) and their suitability for RC airships

Typically, an integrated circuit or a microcontroller is utilised to process the proximity data and control the motor speeds to prevent collisions. In order to accommodate any future growth of any kind, sufficient processing capacity has to be built in to account for the different potential uses of the airship. The Raspberry Pi single-board computer was utilised for this, together with an Arduino Uno controller (Biju and Pant, 2017).

Raspberry Pi, this is a board that serves as a standalone computer, complete with a graphical user interface and a Linux-based operating system. It enables the concurrent execution of intricate computations, analyses, and synthesis of control data on custom software or on any programming language with an editor that can operate on a Linux computer. Due to its Wi-Fi capabilities, the Raspberry Pi 3 Model B can serve as a receiver for any control device with a comparable Wi-Fi connection. A high-end, portable camera can be interfaced with this board. Both the analogue voltage input from the ultrasonic sensors

and the pulse-width-modulated signal for two of the three ESCs can be directly provided by the board's own GPIO ports. However, there aren't many IO pins and not enough PWM pins on the board. Another board, the Arduino Uno, based on an 8-bit, 16-MHz Atmel ATmega328P microcontroller, was utilised to address the problem (Biju and Pant, 2017).

A single-board microcontroller called the Arduino Uno has up to 14 IO pins, six PWM output pins, and six analogue input pins. This gives greater "space" for any potential future expansion and can connect to the Raspberry Pi.

2.3.3 Analysis of existing telemetry systems used in RC Airships and their performance

Based on (Biju and Pant, 2017) research on Design and Development of an Indoor Autonomous Airship, two boards, the Arduino Uno and Raspberry Pi make up the control system. The Arduino Uno is used to read sensor values, transfer them to the Raspberry Pi, and write PWM signals to the ESCs. The Arduino Uno provides sensor values to the Raspberry Pi, which then receives control values from any iOS device, determines the speeds of the three motors, and sends the information back.

The airship's control algorithms still need to be developed in order for it to fly itself. The used handheld device is an iPhone, however any device that is able to send Open Sound Control data (OSC) via a wireless network can be used in its place. TouchOSC9 is the programme used to accomplish this. The Raspberry Pi then receives the OSC data using the visual scripting language Pure Data (PD). The PD programme (patch) then determines the necessary speeds and delivers them to the Arduino Uno via a COMPORT. The ESCs use a PWM signal from the Uno to control the speed of the BLDC motors.

Figure 2.2 provides a visual representation of how the control system operates and how different components are interconnected.

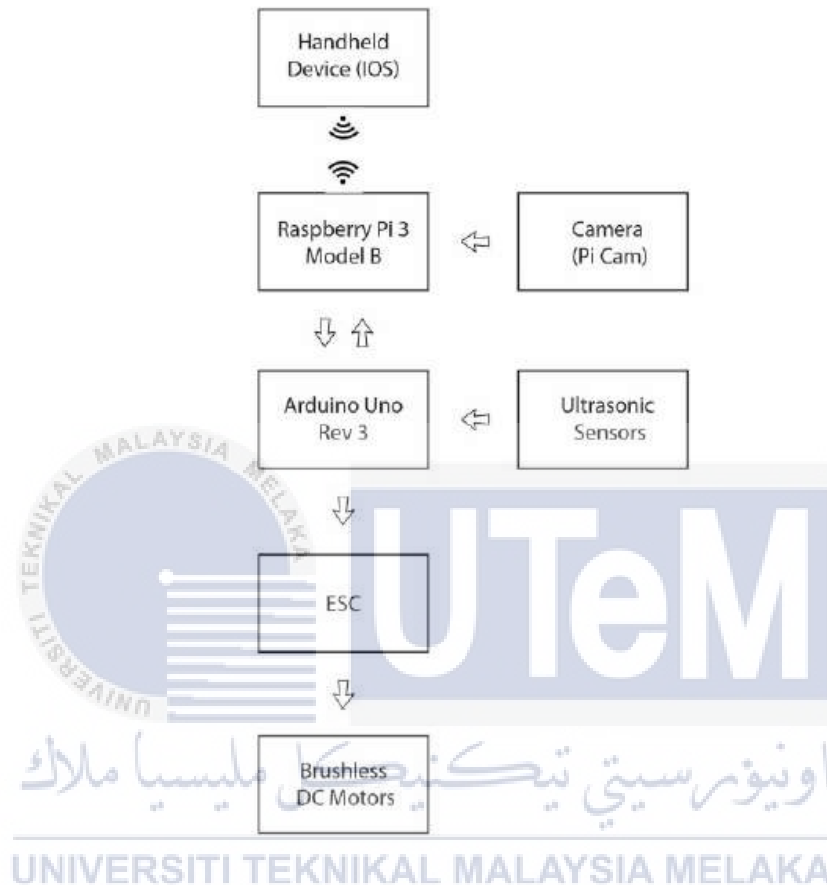


Figure 2.2 Data Flow Diagram (Biju & Pant, 2017)

2.4 Design and Development of Custom Sensor Boards

2.4.1 Overview of the design process for custom sensor boards using CAD software

A new sensor board requires time and work to design. When producing in small quantities, the cost of the hardware components is outweighed by the cost of the design. For instance, circuit design and CAD (Computer-Aided Design) artwork can take weeks to complete in a research project or a pilot project. To create tens of sensor boards with material costs of a few dollars each, thousands of dollars are then spent on PCB (Printed Circuit

Board), metal mask, and assembly. Different sensor boards share some features. The design effort for a new sensor board can be considerably decreased by modularizing sensor boards. There is only one module that must be specifically constructed for each sensor board (Kang *et al.*, 2012).

2.4.2 The considerations for sensor selection, placement, and integration on the board

Custom sensor boards for RC airships must carefully take into account a variety of factors, including sensor placement, integration, and choice. This section includes a summary of pertinent research that focuses on the design and development aspects of custom sensor boards.

Selection of sensors is essential for gathering accurate and reliable data. Numerous studies have looked into the application of particular sensors in RC airships. For example, (Yu, Hu and Huang, 2018) investigated the integration of GPS and INS (Inertial Navigation System) to provide the necessary information for data fusion such vehicle position, heading angle and yaw rate. They mention that to achieve favourable performance, GPS is often integrated with some other positioning technologies. (Barón and Barón, 2014) present the advantages of smart sensors, among which stands out the SHT71 temperature/humidity sensor.

Proper sensor placement and integration are essential to optimize data collection and minimize interference. The optimisation of sensor placement and number for network provisioning was covered in research by (Dhillon and Chakrabarty, 2003) , which also supported such minimalistic sensor networks. Finding the bare minimum of sensors and

where to put them so that each grid point is covered with a minimum degree of confidence is the aim of the sensor placement algorithms.

2.4.3 Analysis of different PCB design techniques and best practices for sensor board development

The development kit is made up of the PCB, connection cables, and development software tools. These boards typically also have expansion sockets, connections, and pins to link the system to additional devices. While it varies depending on the SBA (Single-Board Architectures) manufacturer, there are a number of more or less common peripherals that enable increasing the capacity of these systems. There are different extension boards designed fit into the pins and connectors of each development board and platform. As a result, expansion boards are unique to a particular development kit and are frequently given distinctive names to set them apart from others. For instance, Raspberry Pi extension boards are referred to as "hat" and Arduino expansion boards as "shield" (Álvarez, Mozo and Durán, 2021).

Table 2.1 Main open-source SBMs (Single-Board Microcontroller) in the market. (Álvarez et al., 2021)

	Wiring [85]	Adafruit [38]	Arduino/Genuino [42]		Teensy [77]
Processor	AVR8	Tensilica L106	AVR8	ARM Cortex-M0+	ARM Cortex-M
Architecture/Bits	AVR atmega/8	RISC/32	AVR atmega/8/32	Atmel SAMD21/32	MK20DX/32
Europe Prize (€) [44,49,54,72]	-	10	10-35	20-40	10-30
IDE	Wiring	Arduino IDE MicroPython	Arduino IDE	Arduino IDE	Teensyduino
Open-Source HW	Yes	Yes	Yes	Yes	Yes
Open-Source SW	Yes	Yes	Yes	Yes	Yes
Versions	3	1	10	11	8
IST	No	No	Yes	Yes	No
Most Popular	Wiring V1.1 Wiring Mini V1.0 Wiring S	Huzzah ESP8266	UNO Rev. 3 Mega 2560 Leonardo Nano Every Micro	MKR1000 MKR Zero Zero	Teensy LC Teensy 3.2 Teensy 3.6 Teensy 4.0 Teensy 4.1

Based on Table 2.1, the most widely used SBM platform globally and one of the free source SBMs is Arduino, which has become nearly a standard (Álvarez, Mozo and Durán, 2021). Due to the fact that anyone can alter or improve the original design, there are numerous third-party Arduino compatible SBMs with the same pinout, size, shape, or other features. Additionally, there is a sizable developer community that has produced a vast array of libraries and resources that may be used for any project. This is also true of the several shield-boards, which are existing extension boards.

(Álvarez, Mozo and Durán, 2021) also mention that, Arduino boards' hardware consists of a PCB containing an Atmel AVR microcontroller (ATmega8, Atmega168, Atmega328, or Atmega1280, depending on the model), whose input-output ports are pin-accessible, and a minimal number of auxiliary components. The boards can be purchased fully assembled or without any additional components, but because their technical data are freely available online, they can also be changed. On the other hand, the software is a free to download IDE (Integrated Development Environment) built on Processing. The processor is programmed using Wiring, a C-based programming language, and its reference data is regularly debugged and annotated by a sizable developer community. If an interactive autonomous object is created, the Arduino projects can function without being connected to a computer. To run as an auxiliary object in a large, complete project, Arduino can also be connected to software such as Processing, Max/MSP, Pure Data, Java, JavaScript, and others.

2.5 Embedded Systems for Sensor Integration

2.5.1 Explanation of embedded systems and their role in data acquisition and processing

Data acquisition systems (DAL) are essential for monitoring, analysing, and studying specific occurrences and processes in many research domains. Depending on the DAL architectural solution that is chosen, its installation time and associated costs will vary. The wireless sensor networks (WSN)-based real-time, distributed DAL is being utilised extensively in a variety of applications that involve the environmental, medical, agro-food, and industrial areas. (Grunde, 2013) stated that, different types of sensors can be interfaced to data recording devices using a dedicated programmable interface device. The seamless link between end-user and smart sensor systems must be created using a sensor driver and programmable input-output interface.

2.5.2 Overview of microcontrollers and single-board computers suitable for RC airship applications

Microcontrollers and single-board computers have gained significant interest in various domains, including RC airship applications. Their integration with sensors and communication technologies provides a powerful platform for control, monitoring, and data acquisition. By evaluating factors such as cost, processing capacity, open-source availability, power consumption, reliability, programming flexibility, support availability, and electronics utilities, researchers and enthusiasts can choose the most suitable platform for their specific RC airship requirements. Future research may focus on advancements in microcontrollers and single-board computers, exploring their potential for further enhancing RC airship performance, autonomy, and functionality (Álvarez, Mozo and Durán, 2021).

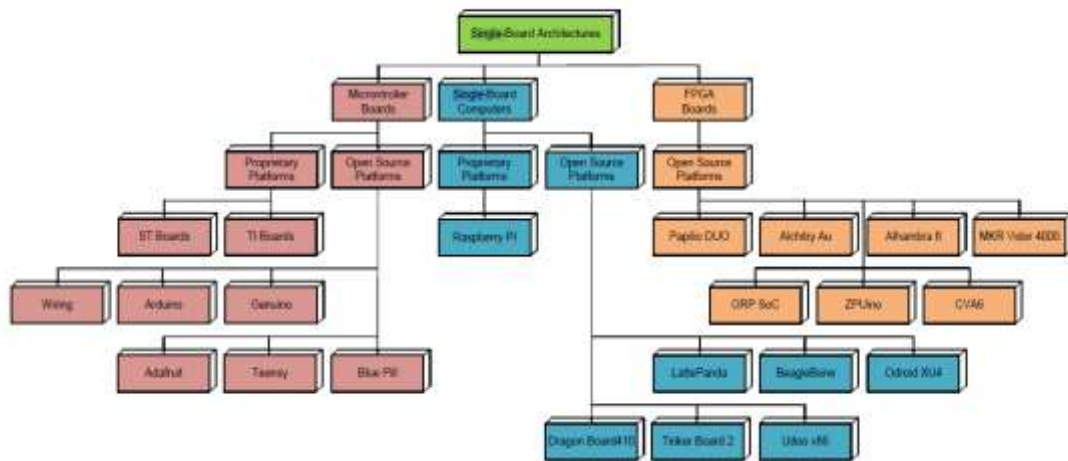


Figure 2.3 Single Board Architectures classification and main platforms.

According to (Álvarez, Mozo and Durán, 2021) by referring to Figure 2.3, single-board computers are more adaptable and dependable than single-board microcontrollers. They often run an OS (Operating System) like Linux or Windows and have a substantially better computational power. The latter, in contrast, are more electronics-focused and geared towards input/output port management. As an illustration, with IoT (Internet of Things) devices, the hardware that is connected to sensors for data collection should be controlled by microcontrollers, whereas the hardware that processes the volume of input information will be a microcomputer.

2.5.3 Exploration of programming languages, frameworks, and development tools for embedded systems

Microcontroller units (MCUs) programming languages have not kept up with hardware advancements. The C/C++ languages continue to be the industry standard for embedded systems despite ongoing research in the area because they offer a well-known imperative programming model, compilers that generate extremely efficient code, and low level access to hardware features when required (Devine *et al.*, 2018).

(Devine *et al.*, 2018) also mention that installing code editors, specialised device drivers, compiler toolchains, and even additional programming hardware (such a JLink programmer) is frequently necessary to set up an embedded system's development environment. This offers a significant adoption barrier for many, notably in the sphere of education, as in many schools, customised hardware and software is simply not allowed by policy or access to the necessary technical support is not present. The development of embedded software must therefore be completely transparent, platform independent, and require zero installation for a solution to be effective.

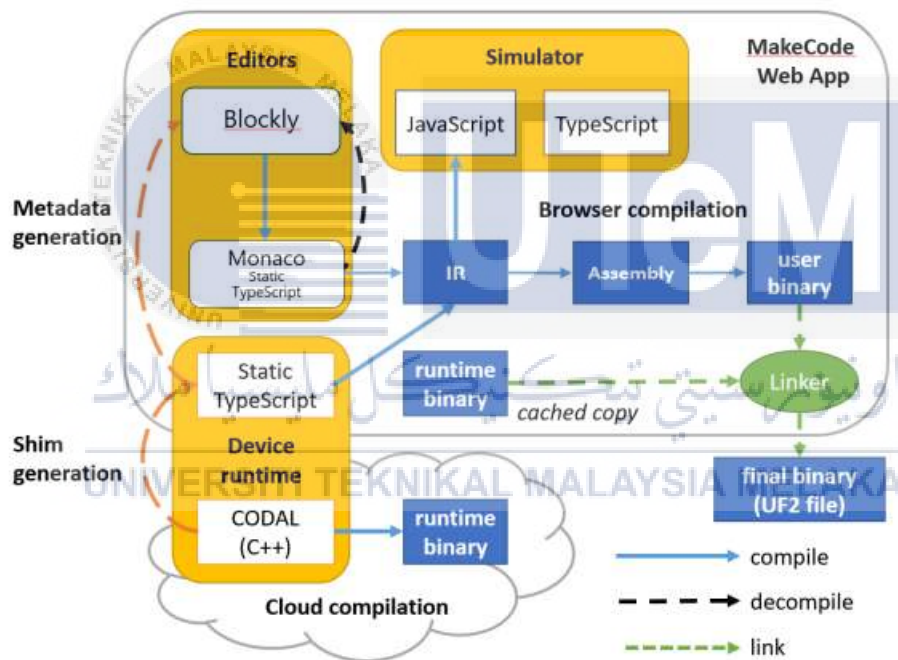


Figure 2.4 The MakeCode and CODAL Architecture (Devine et al., 2018)

(Devine *et al.*, 2018) platform's architecture is shown in Figure 2.4. The main point of entry for the user is the MakeCode web application. MakeCode offers editors for visual blocks and the text-based TypeScript2 language to facilitate the streamlined programming of MCUs. The CODAL runtime environment for C++ bridges the semantic gap between

higher-level languages like TypeScript and the hardware (bottom left of the Figure). It is component-oriented, event-driven, and fiber-based.

UF2, a new file format and bootloader for the streamlined transfer of binaries to the device through USB (Universal Serial Bus) (bottom-right), enables the flashing of the microcontroller.

Any current web browser can access MakeCode, which is also cached locally for completely offline use. The open-source Blockly3 and Monaco4 editors, an in-browser device simulator for testing programmes before transferring them to the actual device, in-browser compilation of TypeScript to machine code, and linking against the pre-compiled CODAL C++ runtime are all included in the MakeCode web app. When connected to a computer, MakeCode devices show up as USB pen drives courtesy to UF2. The compiled binary is "downloaded" locally to the user's computer (lower-right) after the user has done creating the programme, and it is then transferred (flashed) to the MCU (Microcontroller Units) via a straightforward file copy procedure. Any operating system (MacOS, Windows, Linux, etc.) with built-in support for USB pen drives can use this out of the box (Devine *et al.*, 2018).

2.6 Integration Techniques for Sensor Boards and Telemetry Modules

(Olejnik, 2020) study shows that the integration of all electronics put in a tiny aeroplane is a critically important issue during the creation of a new unmanned aerial system (UAS). This integration must be completed at the following three levels, mechanical assembly, electric connectivity, and establishing readiness for operation of each subsystem in order to guarantee full system complexity and, consequently, required functionality. The mini-UAV (Unmanned Aerial Vehicles) described here presented more generally was

created as a demonstrating product of some research and development projects carried out to prove the usability and effectiveness of a small flying platform designed and constructed for the purposes of video monitoring and inspecting in close proximity and at low altitude. The main objective was to design, build, and test a few mini-UAV technology demonstrators that were required to have the following components or systems:

- electric propulsion, with an electric engine pulling a propeller and controller;
- communication subsystem to ensure transmission of telemetry and picture data
- recovery system with autonomously deployed parachute
- power supply system with electrical sources and wiring. Camera gimballed head to enable day and night landscape detection (in TV or thermal mode).

Sensor boards are essential components for collecting and processing information from physical occurrences. It has been suggested that several integration methods can be used to successfully integrate sensor boards into telemetry systems. Standard communication protocols like I2C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface), and UART (Universal Asynchronous Receiver-Transmitter) are used frequently as one common strategy. These protocols provide efficient data transfer and synchronisation by facilitating seamless communication between sensor boards and telemetry modules.

I2C, a popular synchronous serial communication protocol, provides a simple bus topology that enables the connection of numerous sensors over just two wires (SCL and SDA). I2C's addressing method makes it simple to integrate a variety of sensors and allows for effective data transport at modest speeds. The protocol's ability to support several masters makes it appropriate for complicated systems that require simultaneous communication between numerous embedded devices and sensors. I2C, however, may present problems in applications that need high-speed data capture or operate over greater distances because to its restricted range and significantly slower data transfer rate compared to other protocols.

SPI is a synchronous full-duplex serial communication protocol that gives embedded systems and sensors a high-speed, full-duplex interface. In addition to a clock line (SCLK) and chip select lines (CS) for each sensor, it uses separate lines for data transmission MOSI (Master Output Slave Input) and reception MISO (Master Input Slave Output). For applications demanding quick data transmission rates or real-time data gathering, SPI provides great throughput. SPI is more suited to scenarios with fewer sensors because of its bus layout, which restricts the number of devices that may be connected directly to the embedded system.

The asynchronous serial communication protocol known as UART (Universal Asynchronous Receiver-Transmitter) is frequently used for close proximity communication between embedded devices and sensors. Two data lines, RX for receiver and TX for transmitter, as well as a shared ground connection are used to power it. Since UART is so versatile, it is very easy to implement and integrate many types of sensors. UART is frequently used for one-to-one sensor transmission or in applications with minimal sensor needs due to its simple point-to-point connection. UART can be scaled, but because of its slower data transfer rate and absence of an intrinsic addressing scheme, it may not be appropriate for systems that need to acquire data quickly or from many sensors.

The primary responsibility of the sensor network is to provide the secure transfer of measured data from the sensors to the control block, where it can be processed further. These data must be real in order to confirm their legitimacy because of the nature of the information. In the IoT (Internet of Things) space, communication security is also crucial. The Industrial Internet of Things (IIoT) in the industrial sector includes embedded systems, wireless sensor networks, and low-cost sensors. Industry 4.0's deployment of these systems is built on the IIoT. IIoT may securely transmit sensor readings to cloud-based data centres

in order to continuously gather information from a variety of sensors and update associated parameters in attached control subsystems (Dudak *et al.*, 2019).

For the integration of sensors into embedded systems to be successful, it is essential to choose the right communication protocol (Dudak *et al.*, 2019). I2C, SPI, and UART are just a few of the protocols that each have their own advantages and things to think about. I2C offers simplicity and support for many masters, but it has a smaller operating window and slower data transfer rates. SPI shines in real-time applications and high-speed data transfer, but it limits the variety of devices that can be directly connected. Although UART is straightforward and adaptable, it has lower data transmission rates and no built-in addressing mechanism. The best protocol depends on the particular needs of the application, including data transfer rate, sensor count, distance factors, and system complexity.

2.7 Performance Evaluation of Integrated Systems

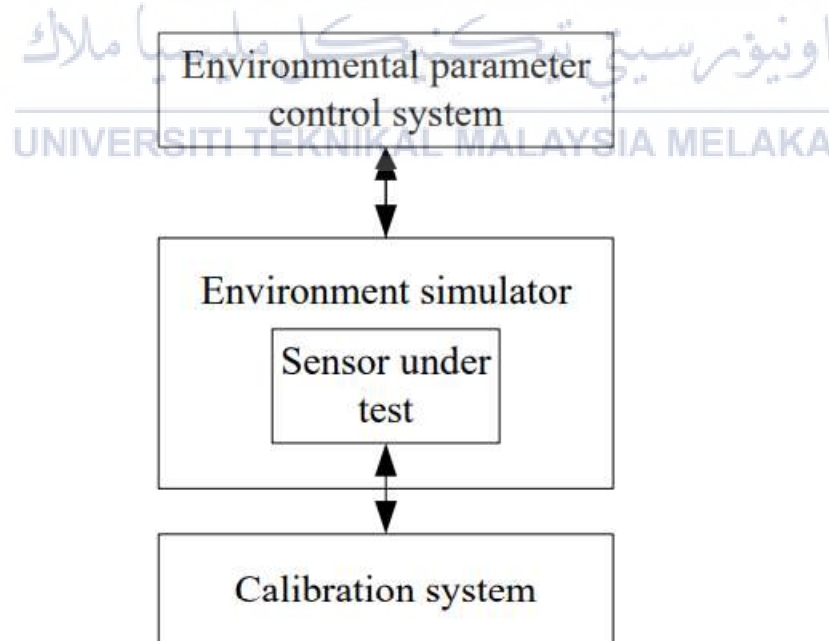


Figure 2.5 Basic structure diagram of calibration system (Hua *et al.*, 2021)

As observed from (Hua, Zhang and Feng, 2021) study, the calibration performance of the sensor in the actual working environment differs from the typical calibration in the laboratory when environmental conditions, such as temperature, humidity, air pressure, vibration, etc., have a major impact on the sensor's performance. The simulation of environmental effect variables is required in addition to the standard laboratory calibration. The simulation is accomplished by integrating environmental simulation technology with measurement and calibration technologies. creation of a calibrating tool for a global working environment. As shown in Figure 2.5, an environment simulation device, an environmental parameter management system, and a sensor calibration system make up a typical calibration system for mimicking a real working environment.

Due to their current calibration principles, technologies, software, and methods, some measurement systems and calibration devices no longer have good adaptability and measurement accuracy in the involved changing environment or limit environment. This is because the development of sensor calibration devices in actual working environments has not yet covered all sensor parameters. The calibration tool is either challenging to use or not suited for mimicking the severe environment. Because of these constraints, only a limited number of sensors, characteristics, and environmental factors may be evaluated for performance in a sensor calibration device that simulates the real working environment (Hua, Zhang and Feng, 2021).

Meanwhile, (Gawale *et al.*, 2009) had conducted a flight test on RC Airship, the first flight's objectives were to determine whether the airship could fly without stabilisers and whether the chosen engine could fulfil the requirements. The combination of an envelope and a gondola was found to be unstable. The highest speed attained was reportedly around 7 m/s.

These conclusions led to the second flying test, which had a flight time of more than two hours. At times, the airship was flown up to 60 metres above land. Even at slow cruising speeds, the control system was shown to have a very good response. To test the airship's ability to turn, it was guided in every direction. Furthermore, it was seen that the created angle of attack was much greater than necessary, resulting in the production of extra lift. This issue arose during the third field testing, during which the gondola was moved to preserve dynamic stability.

A GPS was installed in the gondola during one of the field experiments, and it was used to log flight information including altitude, speed, duration, track, etc. The test was successful, and Table 2.2 contains a tabulation of the GPS data that were taken during the flight.

Table 2.2 GPS recording for the third test flight

Total trip	1.9 km
Flight duration	10:12 min
Max. Speed	24.5 kmph
Avg. Speed	11.2 kmph
Max. Altitude	75 m
Cruising altitude	69 m
Track	0.3 km

Making use of native C++ as our benchmark, (Devine *et al.*, 2018) conduct a comparative analysis of MakeCode against two cutting-edge solutions used by educators in the classroom in order to put its performance into perspective. Espruino, a JavaScript implementation for MCUs, and MicroPython, a Python implementation for MCUs, serve as

the two points of comparison. CircuitPython, a clone of MicroPython, was employed for the CPX. Espruino and MicroPython both employ virtual machine (VM) strategies.

Table 2.3 compares the speeds at which the following languages execute, native C++ with CODAL, MakeCode compiled to native machine code; MakeCode compiled to AVR VM, MicroPython, and Espruino. The C++ time is listed on the first line, while successive lines show times that are slower than the C++ time. The 5x improved code density of MakeCode VM makes up for the 6.4x slower speed of native MakeCode on the Uno.

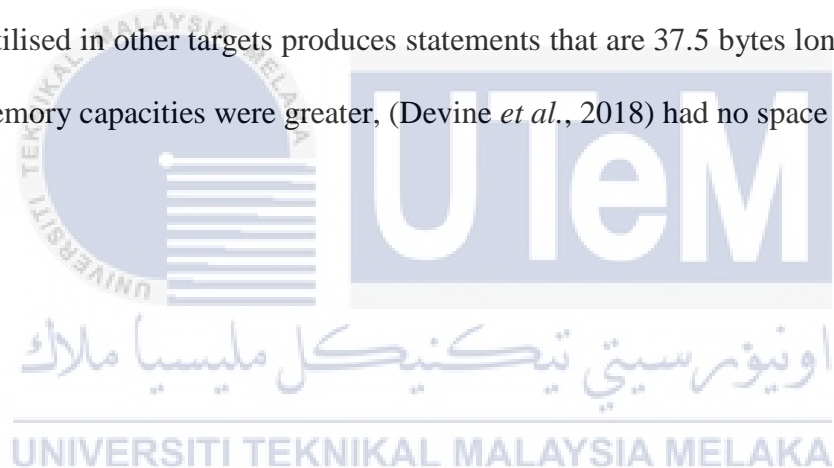
Table 2.3 Language execute speed

	UNO	micro:bit	CPX
CODAL	171ms	102ms	31ms
MakeCode	2.4x	2.1x	7.3x
MakeCode VM	15.3x	-	-
MicroPython	-	101x	183x
Espruino	-	1139x	-

(Devine *et al.*, 2018) created a straightforward programme that counts from 0 to 100,000 in a close loop in the language of each solution to provide an indicative general case execution time cost for each solution; the results are displayed in Table 2.3. (Devine *et al.*, 2018) scale up the results on AVR after counting up to 25,000 (to fit within a 16 bit int). The run is two or more orders of magnitude slower for MicroPython and Espruino on the micro:bit than a native CODAL programme. MakeCode just operates 2 times slower. The slowness is due to our STS compiler's basic code generation. It should be noted that MakeCode for the CPX employs the tagged technique, causing a further 3x slower but allowing for seamless runtime transition to floating point numbers. MakeCode runs at least

an order of magnitude better than the VM-based solutions of MicroPython and Espruino on both devices.

The Uno's flash and RAM size restrictions prevent the use of MicroPython and comparable environments. (Devine *et al.*, 2018) encountered these restrictions as well, and as a result, two compilation modes for AVR were created. The other (MakeCode VM) creates density-optimized byte code for a tiny (500 bytes of code) interpreter while the first translates STS to AVR machine code. The native strategy manages to fit 150 lines of STS user code in the code density of roughly 60.8 bytes per statement. At 12.3 bytes per statement, the VM can handle around 800 lines. For comparison, the ARM Thumb code generator utilised in other targets produces statements that are 37.5 bytes long, but because the flash memory capacities were greater, (Devine *et al.*, 2018) had no space problems.



2.8 Summary of Research Gap

Aerial surveillance, research endeavours, agriculture, and particular missions requiring continuous observation or accurate payload drops are just a few of the uses for RC airships, which are adaptable and reasonably priced platforms. They have advantages over conventional aircraft since they can take off and land without a runway. The gondola, propulsion system, stabiliser and fins, remote control system, and envelope are the five fundamental parts of an RC airship. The cargo and electrical equipment are housed in the gondola, which should be designed with simplicity, aesthetics, and structural strength in mind. For RC airships to perform tasks like surveillance, situational awareness, and data collection, sensor integration is essential. For obstacle identification, navigation, and monitoring, a variety of sensors including infrared, ultrasonic, compass, accelerometers, and gyroscopes are utilised. In RC airships, telemetry systems are crucial for transmitting data remotely, assuring appropriate operation, and enabling the early identification of equipment failures. For telemetry, Wi-Fi, Bluetooth, and RF (Radio Frequency) technologies are frequently employed. In order to maximise data gathering and reduce interference, designing custom sensor boards requires taking factors like sensor placement, integration, and selection into account. Modularization and PCB design strategies help speed up the development process. For data collection, processing, and control in RC airships, embedded systems, such as microcontrollers and single-board computers like Arduino and Raspberry Pi, are employed. These systems provide real-time monitoring and analysis while facilitating the seamless integration of sensors.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The capabilities of RC (Radio Controlled) airships can currently be improved in a number of ways, particularly when sensors and telemetry systems are included. These developments have shown to be helpful for improving situational awareness and data collection, making them suitable for use in surveillance and research projects. Engineers are constantly exploring new ideas and improving existing systems since each one has unique advantages and disadvantages. This project uses computer-aided design (CAD) and embedded technologies to integrate sensors and telemetry systems for an RC airship. The main objective is to create a custom sensor board and telemetry module that can be smoothly integrated into the control system of the airship. The project intends to enhance the airship's situational awareness and data collection capabilities by creating a custom solution. This chapter outlines the method used to create a unique sensor board and telemetry system. Figure 3.1 illustrates the results of extensive research into sensor board and telemetry system technology that was done for integrating them into the airship's control system for the present task. Every stage is further discussed in this chapter to meet the objective of the present task.

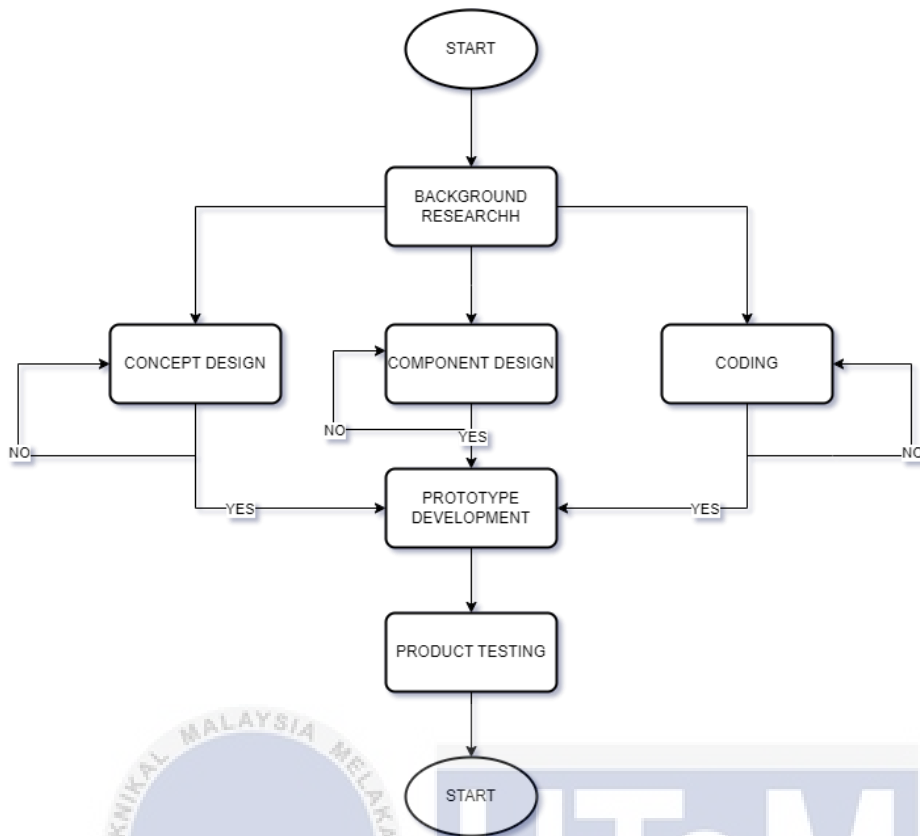


Figure 3.1 Flow chart of the development of the custom sensors and telemetry systems

3.2 Background Research

Before creating a new invention or technology, researchers should thoroughly examine the structure of the sensors and telemetry systems that are currently on the market. Additionally, investigation into the efficacy and operation of the system is necessary to address situation awareness and data collection capabilities. Each product's advantages and disadvantages have been reviewed in order to find areas for future research that could be improved even more.

The most efficient sensors and telemetry systems are chosen based on the background research. However, this depends on the main goal of this project as well as the

system's capabilities for data collection and situation awareness. The best technology can be produced after taking all relevant elements into account. The huge amount of knowledge gathered in this field of research leads to a broader perspective, which helps in choosing the ideal alternative for the new system. To demonstrate that it can achieve the objectives of the project, the developed system will be put to the test in a number of scenarios.

3.3 Design Development

The development of the custom sensor board and telemetry systems is the primary focus of this chapter. The criteria include the necessary market research that results in product development to ensure the best end product that reduces product shortcoming. The use of every component to build one system in the best way possible is an essential stage in achieving that. As a result, the design development process needs to be divided into numerous key components. There are three primary parts in this step which are component design, concept design, and coding. All components are in perfect alignment with one another.

3.3.1 Design Concept

When paired with the RC Airship, the product design concept is a crucial component of the whole system's architecture. The design concept of this product is to be both flexible and simple, incorporating modern technologies that simplifies installation, disassembly, and maintenance. In the meantime, the flexible design concept refers to a new system that is extremely compatible with any RC Airship, meaning that any kind of RC device that requires a specific sensor and telemetry system can use it in addition to luxury RC Airships. In order to achieve the goals, some of the component in the system has been set able to be adjusted

on both software and hardware based on the size of a particular RC Airship. First, the programming code of the control unit can be set up for the range of the ultrasonic sensor detection area. Second, the temperature sensor position can be adjusted on the body of the RC Airship. Both of these functions are to make sure that the device sensing range will cover up the specific area that the user need to sense.

3.3.2 Component Design

It is necessary to initially understand the system's flow in order to determine the parts required for a custom sensor and telemetry system. Figure 3.3 shows the operation of the system.



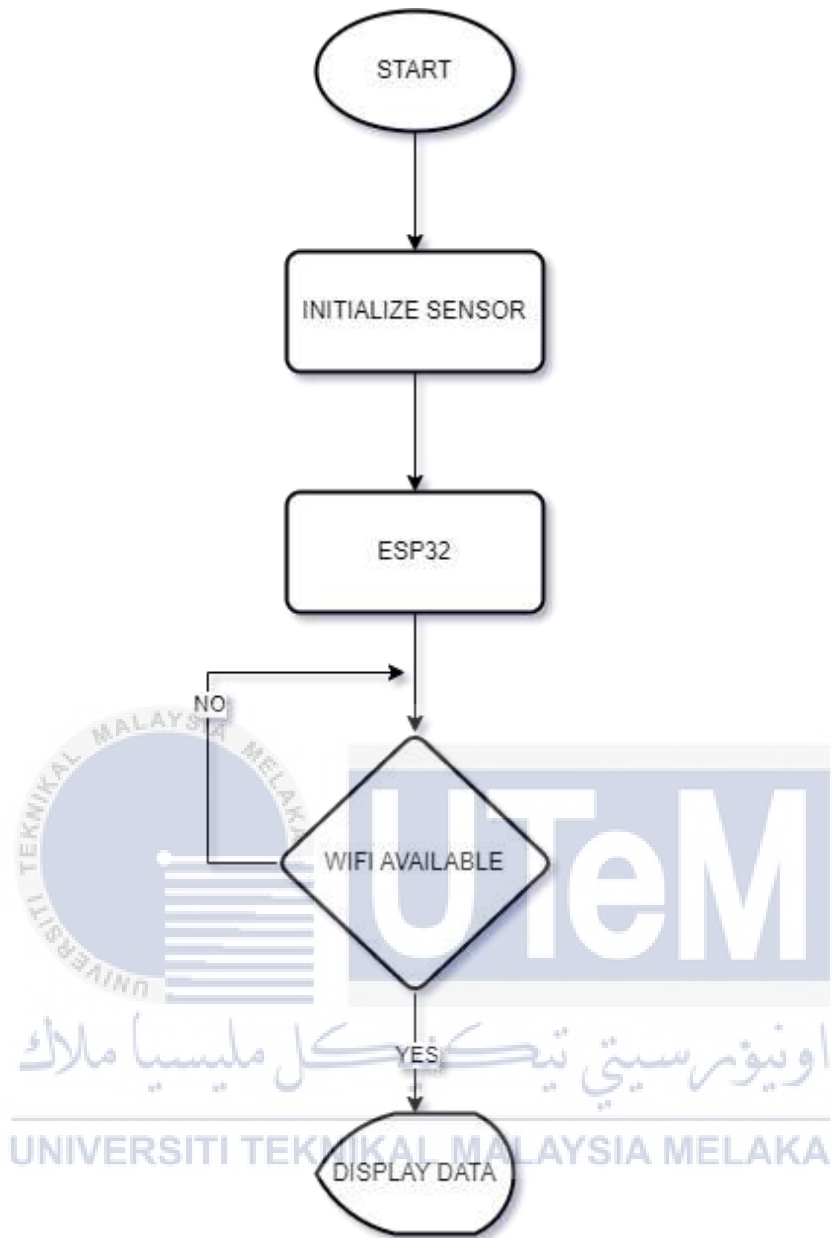


Figure 3.2 Flow Chart of the system work flow

3.3.3 Sensors

3.3.3.1 Accelerometer Sensor

First, an accelerometer sensor that is compatible with the ESP32 was chosen to serve as the system's sensing component (see Figure 3.3). An accelerometer in the customized system measures the tilt angle in tilt-sensing applications as well as the dynamic acceleration



Figure 3.3 SN-ADXL355-CY Accelerometer Sensor

The SN-ADXL355-CY sensor is comprised of seven pins: "3v3" for the 3.3V power supply pin, "5V" for the regulated 5V output from the regulator on the board, "X-Out" for the analogue voltage proportional to acceleration along the X axis, "Y-Out" for the analogue voltage proportional to acceleration along the Y axis, "Z-Out" for the analogue voltage proportional to acceleration along the Z axis, "Ground (GND)" for connecting the ground, and "ST(Self-Test)" for controlling the self-test feature that lets you test the sensor's performance in the final application. This sensor is a highly reliable and well established. With only 320 A of current drawn, it promises excellent performance with less noise and power consumption. The breakout board has an integrated 3.3V voltage regulator that enables it to function between a voltage range of 2.5V and 6V. Its adaptability and versatility

with different microcontrollers and systems are improved by this capability. The board also includes 0.1uF capacitors that enable bandwidth adjustment for each axis, setting it to a frequency of 50Hz. This sensor is perfect for applications needing precise and accurate acceleration measurements in a variety of domains, including robotics, motion tracking, tilt sensing, and vibration analysis. Its small dimensions of 19.5mm x 22mm. Table 3.1 shows the technical specification of this sensor.

Table 3.1 SN-ADXL335-CY Details

Module Name: SN-ADXL335-CY	
Operating Voltage	2.5V – 6V
Operating Current	350µA (typical)
Sensing Range	±3g (Full Scale)
Temperature Range	-40 to +85°C
Sensing axis	3 axis
Sensitivity	270 to 330mV/g (Ratiometric)
Shock Resistance	Up to 10,000g
Dimension	19.5mm x 22mm (LxH)

One of the reasons this sensor being chosen for the development system is its capability in sensing range of $\pm 3g$ allows it to simultaneously record acceleration in three axes. Capacitive structures embedded on a silicon-based substrate are used to power the sensor. Both stationary plates and moving masses make up these structures. Between the stationary plates and moving masses, there is an initial capacitance created when the sensor is at rest. The movable masses, however, move when acceleration is applied because of inertial forces. The capacitance is altered as a result of the displacement, which alters the distance between the stationary plates and moving masses. Utilising internal circuitry, the sensor transforms this change in capacitance into the relevant voltage levels. These voltage outputs are proportional with the applied acceleration along the sensor's X, Y, and Z axes. The sensor derives actual acceleration values by processing and analyzing these voltage outputs, providing precise and reliable data for applications requiring three-axis acceleration measurements.

3.3.3.2 Altimeter Sensor

Another sensing component chosen for system is BMP180 barometric pressure sensor shown in Figure 3.4 which is also compatible with the ESP32. In the custom system, this sensor exhibits notable capabilities in measuring temperature with accuracy and reliability.

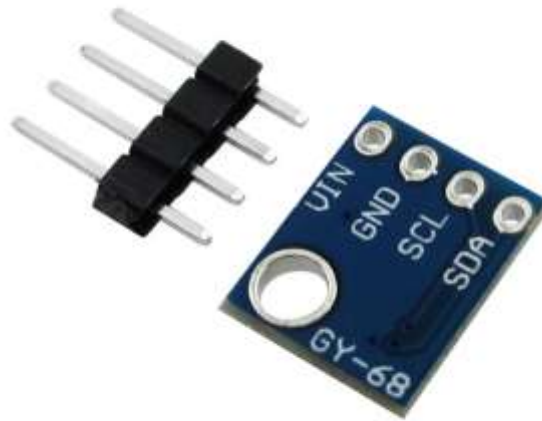


Figure 3.4 GY-68 BMP180 Barometric Pressure Sensor

The GY-68 BMP180 barometric pressure sensor has 4 pins which are 'VIN' for connecting power supply, 'Ground (GND)' for connecting to ground, 'SDA' Serial Data pin (I2C interface) and 'SCL' Serial Clock pin (I2C interface). For a variety of applications, the sensor delivers excellent performance capabilities. With a 1.3V to 3.6V operational voltage range, it offers a variety of power supply options. The BMP180 module, which houses the sensor, is compatible with a variety of microcontrollers or systems since it accepts input voltages ranging from 3.3V to 5.5V. The sensor's highest current consumption while active operation is 1000uA, demonstrating its power needs. However, it only uses 0.1uA when in standby mode, ensuring effective power management and longer battery life. The maximum voltage at the SDA and SCL lines is $V_{IN} + 0.3V$, guarding against possible sensor harm. Additionally, the BMP180 sensor is appropriate for a variety of climatic circumstances because it can function dependably throughout a wide temperature range, from $-40^{\circ}C$ to $+80^{\circ}C$. The BMP180 is a great option for applications requiring accurate altitude measurement, temperature monitoring, and weather analysis because of its performance characteristics. Table 3.2 shows the technical specification of this sensor.

Table 3.2 GY-68 Details

Module Name: GY-68			
Item	Min	Typical	Max
Voltage	3	5	5.5
Current	1.1	/	20
Pressure Range	300	/	1100
Faster I2C data transfer	/	/	3.4
Dimension	40.1x20.2x9.7		mm

One of the reasons this sensor being chosen for the development system is, it enables the airship to accurately measure the atmospheric pressure, which is crucial for determining altitude. In order to determine the airship's elevation, the sensor makes use of the relationship between atmospheric pressure and altitude. The atmospheric pressure reacts to the airship's ascent or descent, and the sensor picks up these changes. The sensor uses its barometric pressure detecting capabilities to measure the current air pressure in order to determine altitude. The reference pressure, which is normally measured at ground level or an altitude known, is then compared to this pressure value. The sensor can determine the airship's relative height by examining the difference between the measured pressure and the reference pressure. As temperature affects air pressure, it is crucial to account for temperature fluctuations to improve accuracy. The temperature sensor included into the BMP180 sensor enables it to take temperature fluctuations into account when calculating altitude.

Furthermore, accurate altitude measurements are made possible by the sensor's excellent resolution and sensitivity, even for minute variations in pressure. This ability is essential for keeping the airship at the desired altitude or performing precise maneuvers based on altitude. The RC airship's control system can use the BMP180 sensor's measured altitude data for altitude hold, altitude tracking, and altitude-based navigation. It enables the airship to maintain a particular height or adhere to a predetermined altitude profile by allowing it to modify its buoyancy, manage engine thrust, or initiate other control activities.

3.3.4 ESP32

The third component is ESP32 is shown in Figure 3.5, which can be effectively used to improve the capability and communication of an RC airship.

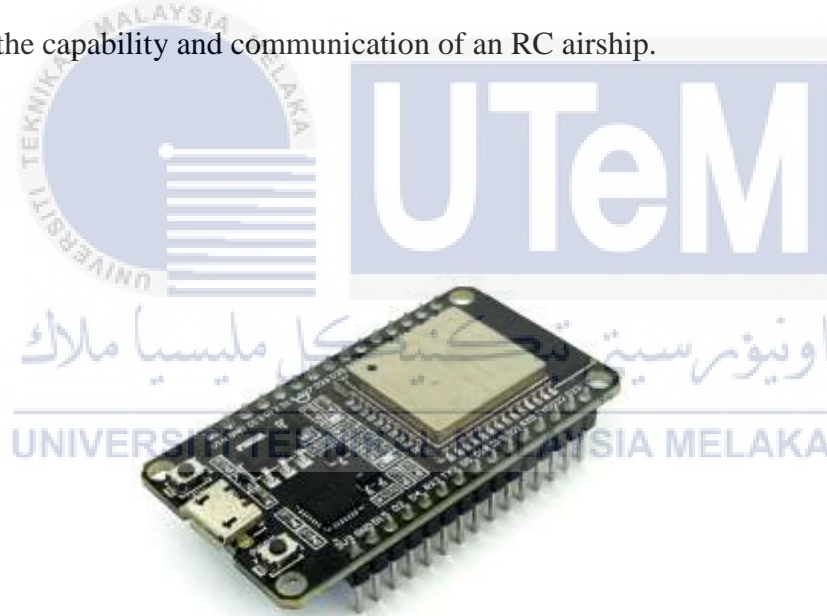


Figure 3.5 ESP32

An RC airship can take advantage of the ESP32's robust computing powers, as well as its integrated Wi-Fi and Bluetooth characteristics, to enable a variety of features and control mechanisms. For instance, the ESP32's dual-core Xtensa LX6 CPU is capable of handling challenging tasks relating to real-time communication, sensor data processing, and

flight control algorithms. It contributes to a more stable and responsive flight experience by being able to process computations and control orders quickly. With the ESP32's Wi-Fi capabilities, you may connect the airship wirelessly to a mobile device or a ground station. This makes it possible for you to remotely check and manage the airship's many settings, including altitude, tilt angle, heading, and battery life. Additionally, the ESP32's Bluetooth capabilities can be used to remotely operate the blimp from a mobile device or other electronic device. It offers a practical and dependable wireless communication route for transmitting instructions and receiving telemetry information.

3.3.5 Wiring

Wiring is essential to the proper operation of every component and must be completed properly in relation to the program code that will be uploaded into the ESP32. To make additional maintenance easier, the wiring arrangement must also be nice and orderly. The connecting of the accelerometer sensor to the ESP32 and the GY-86 to the ESP32 are shown in Figure 3.6 and Figure 3.7. Table 3.3 provides a summary of the process of uploading program code to the control unit system.

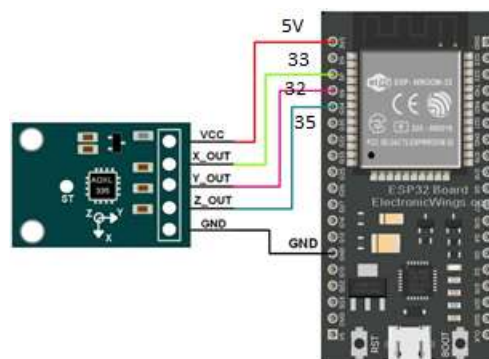


Figure 3.6 Accelerometer sensor to the ESP32

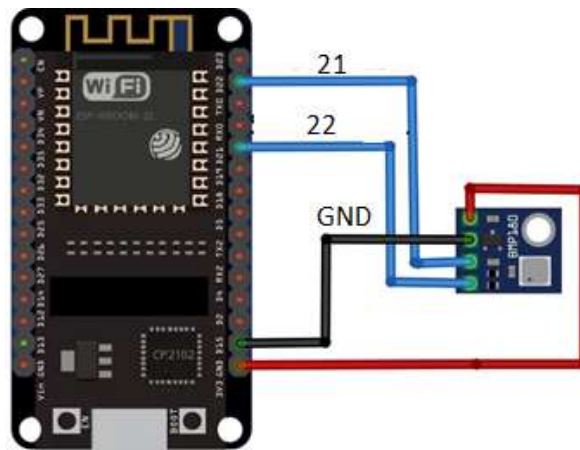


Figure 3.7 Barometric sensor to the ESP32

Table 3.3 Wiring Connectivity

PIN/TERMINAL	ADXL355	GY-68
VCC	PIN 5V	PIN 5V
X_Out	33	-
Y_Out	32	-
Z_Out	35	-
GND	PIN GND	PIN GND
SCL	-	22
SDA	-	21

3.3.6 Program Code

The inside design of this smart system is just as important as the external appearance. Since ESP32 uses C/C++ as its programming language, C++ programming code is utilized. In order to identify the warning behaviors for the driver's alert, this will measure the parameters that need to be managed while recognizing the threat. This stage of data processing is when the reading for temperature, altitude and axis angle is constantly transmitted. Additionally, the 'void loop' function is present to guarantee that the system operates nonstop. Figure 3.8 and Figure 3.9 below displays the custom system's program code.



```

#include <Adafruit_BMP085.h>
#include <math.h>

const int x=25, y=26, z=27;
int xv=0, yv=0, zv=0;
Adafruit_BMP085 bmp;

void setup() {
  Serial.begin(115200);
  if (!bmp.begin()) {
    Serial.println("Could not find a valid BMP085 sensor, check wiring!");
    while (1) {}
  }
}

void loop() {
  Serial.print("Temperature = ");
  Serial.print(bmp.readTemperature());
  Serial.println(" °C");

  Serial.print("Pressure = ");
  Serial.print(bmp.readPressure());
  Serial.println(" Pa");

  // Calculate altitude assuming 'standard' barometric
  // Pressure of 1013.25 millibar = 101325 Pascal
  Serial.print("Altitude = ");
  Serial.print(bmp.readAltitude());
  Serial.println(" meters");

  Serial.print("Pressure at sealevel (calculated) = ");
  Serial.print(bmp.readSealevelPressure());
  Serial.println(" Pa");

  Serial.print("Real altitude = ");
  Serial.print(bmp.readAltitude(101500));
  Serial.println(" meters");

  adxl();
  delay(1000);
}

void adxl()
{
  xv = analogRead(x);
  yv = analogRead(y);
  zv = analogRead(z);
  Serial.print("x= ");

```

Figure 3.8 Program Code

```

Serial.print(xv);
Serial.print(" y= ");
Serial.print(yv);
Serial.print(" z= ");
Serial.println(zv);

if((xv<2400) || (xv>3800) || (yv<2400) || (yv>3700))
{
  Serial.println("Overtumed");
}
else
{
  Serial.println("Normal");
}
}

```

Figure 3.9 Program Code Continuation

3.3.7 Prototype

At the conclusion of this project, a fully functional prototype of the custom sensor and telemetry system was constructed using all the methods that were involved. At last, the prototype depicted in Figure 3.10 is prepared for real-world experimentation.

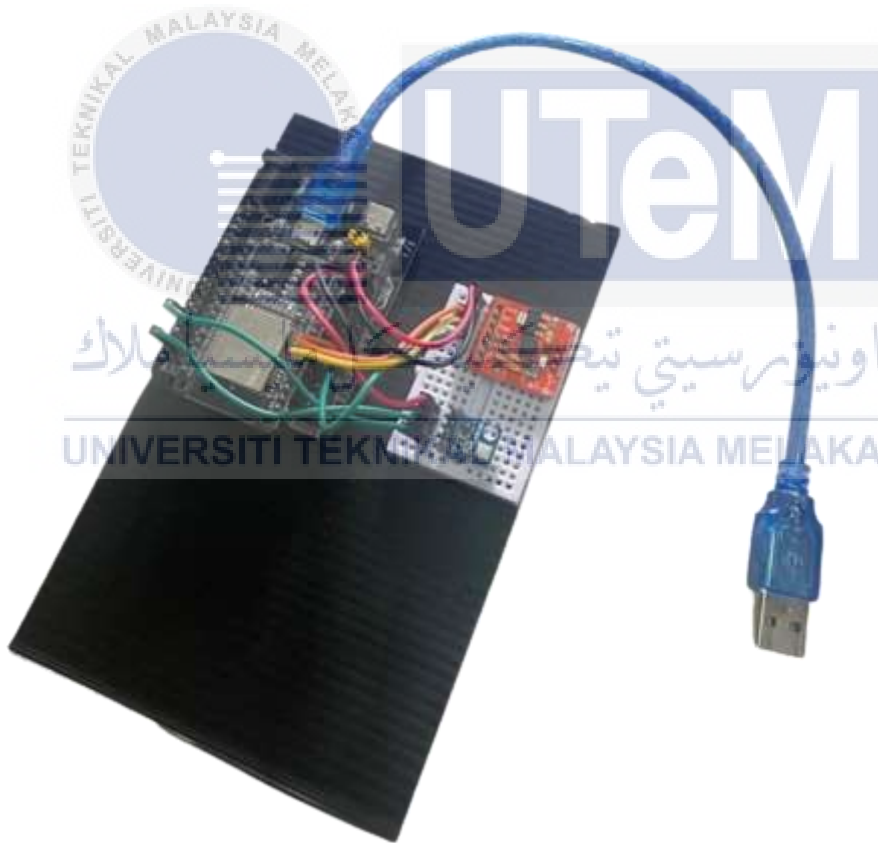


Figure 3.10 Custom Sensor Prototype

3.4 Product Testing

```
Output Serial Monitor X
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')
15:24:36.616 -> real altitude = 69.00 meters
15:24:36.921 -> x= 2928 y= 2864 z= 3487
15:24:36.921 -> Overturned event not triggered
15:24:37.513 -> Temperature = 29.50 *C
15:24:37.513 -> Pressure = 100665 Pa
15:24:37.546 -> Altitude = 54.68 meters
15:24:37.585 -> Pressure at sealevel (calculated) = 100673 Pa
15:24:37.618 -> Real altitude = 69.55 meters
15:24:37.928 -> x= 2863 y= 2786 z= 3436
15:24:37.928 -> Overturned event not triggered
15:24:38.498 -> Temperature = 29.50 *C
15:24:38.498 -> Pressure = 100673 Pa
15:24:38.543 -> Altitude = 54.51 meters
15:24:38.578 -> Pressure at sealevel (calculated) = 100669 Pa
15:24:38.610 -> Real altitude = 68.71 meters
15:24:38.922 -> x= 2823 y= 2783 z= 3440
15:24:38.922 -> Overturned event not triggered
```

Figure 3.11 Product Testing Result

Figure 3.11 presents information gleaned from product evaluation. A stable testing environment is ensured by the constant temperature indicated in the picture. This is especially important for the barometer sensor, which shows promise in gathering atmospheric pressure data.

The x, y, and z axes' accelerometer sensor values all show dynamic variations at the same time. Specifically, the x-axis values ranged between 2928 and 2823, the y-axis between 2864 and 2783, and the z-axis between 3487 and 3440. These variations provide information about the dynamic behaviour of the product by indicating changes in acceleration, movement, or vibrations during the testing procedure.

In addition, the text specifies an altitude above sea level with a range of 69.55 to 68.71 metres. In some situations, this parameter is quite important since it tells you how well the product performs as the altitude varies. The product's stated range suggests that it was evaluated within this altitude range, which adds to our understanding of how it behaves in various environmental settings.

To summarize the product's response to variations in height above sea level within the designated range, dynamic fluctuations in accelerometer readings, and the stable operation of the barometer sensor under constant temperature circumstances.

3.5 Summary

In conclusion, RC airships have been improved by the application of an organized, research-based approach that incorporates cutting-edge sensors and telemetry systems. To choose effective technology, a comprehensive background investigation is the first step in the process. Flexibility and simplicity are given top priority during the design development phase to ensure a smooth integration into a variety of RC airships. Important parts are added to improve capabilities, like the ESP32. For optimal performance, the wiring is prioritized, and the temperature, altitude, and axis angle are continuously monitored by the C/C++ computer code. A fully working prototype is built, and product testing shows dynamic variations in accelerometer readings and reliable functioning under constant settings. To improve RC airships with a custom sensor and telemetry system, a comprehensive strategy comprising research, design, and testing is reflected in the technique.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

It is now essential to incorporate cutting edge sensor technologies and telemetry systems with remotely controlled (RC) airships to maximise their potential for research, surveillance, and environmental monitoring, among other uses. The goal of this research is to overcome the present constraints that RC airships have, particularly their poor situational awareness and ineffective data collection capabilities. This project attempts to smoothly integrate a custom sensor board and telemetry module with the control system of the remote-control airship. This study's methodology focuses on the experimentation and analysis required to determine the sensor's surroundings' temperature, tilt angle, and altitude. The findings will enhance the knowledge of tailored sensor for enhancing data collectability and situational awareness.

4.2 Result and Discussion

4.2.1 Actual Data at Kolej Kediaman Satria (Kasuri)

By using the website (<https://en-gb.topographic-map.com/map-nq18/Malaysia/>), it enables to actual altitude data collection.

Based on Figure 4.1, the actual altitude at Kolej Kediaman Satria (Kasturi) based on sea level is 71m.

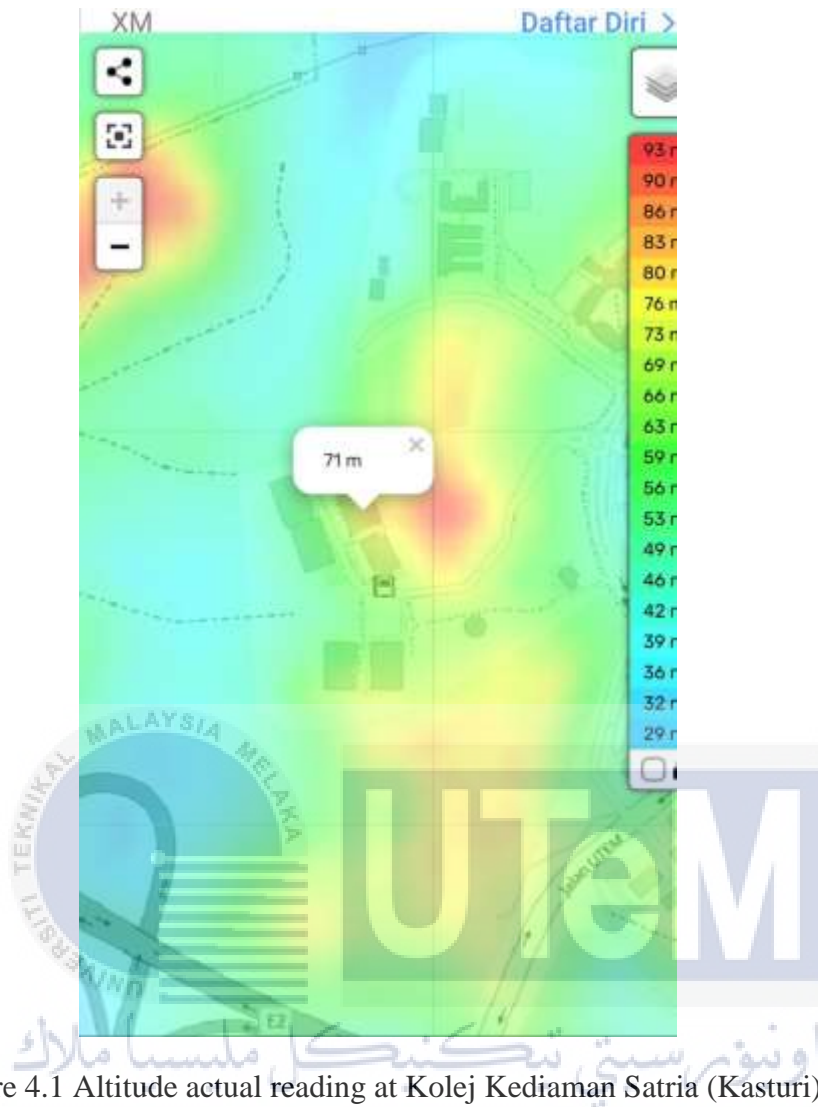


Figure 4.1 Altitude actual reading at Kolej Kediaman Satria (Kasturi)

For the temperature data, the actual data used is 27 °C which is based on the room temperature.

4.3 Experimental Data at Kolej Kediaman Satria (Kasturi)

4.3.1 Data collection using direct USB cable connection

4.3.1.1 Test at rest condition

With reference to Table 4.1, the GY-68 barometric sensor was the subject of the first test. Throughout all data points, the recorded temperature values remained constant at 27.3°C. Concurrently, the GY-68's altitude measurements showed a limited range from

91.24 to 91.91. Ten different data collection processes were used throughout the process. The second test continued to use only the GY-68 barometer sensor, maintaining consistent temperature readings at 27.3°C. The GY68's reported altitude measurements showed a range of 100.55 to 101.64. In the third test, which used only the GY-68 barometer sensor, the temperature was consistently recorded at 26.6°C. The GY68 also measured altitude, and its results ranged from 115.33 to 116.43. In these testing, the GY-68 barometric sensor was the only one used, and it continuously provided stable and accurate temperature readings. This sensor proved that it could remain accurate in a variety of test conditions. The sensor system's direct USB cable connection to the laptop allowed for effective data transfer for real-time monitoring.

Based on Table 4.1 The average altitude based on sea level taken on test no 1 is 91.425 m, by subtracting it the actual data with the data taken, the initial altitude value is 20.425 m.

Table 4.1 Dataset for Altitude and Temperature with direct USB cable connection at rest condition

No of Test No of Data	1		2		3	
	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)
1	91.74	27.00	100.64	27.30	116.43	26.60
2	91.24	27.00	101.22	27.30	115.84	26.60
3	91.83	27.00	101.22	27.30	115.33	26.60
4	91.33	27.00	101.22	27.30	115.67	26.60
5	91.91	27.00	101.22	27.30	115.75	26.50
6	91.49	27.00	101.39	27.30	116.01	26.50
7	90.99	27.00	101.06	27.30	115.75	26.50
8	91.41	27.00	100.55	27.30	115.67	26.50
9	91.49	27.00	101.06	27.30	115.33	26.50

10	90.82	27.00	101.64	27.30	115.92	25.60
Average	91.425	27.00	101.122	27.30	115.77	26.45

Based on Figure 4.2, when at rest, the GY-68 barometer sensor's altitude graph shows readings between 90.82 and 91.91. This very small operating range implies that the sensor can continue to monitor altitude reliably even when stationary. The little variations in this range show how accurately the sensor records changes in atmospheric pressure while it is not in use.

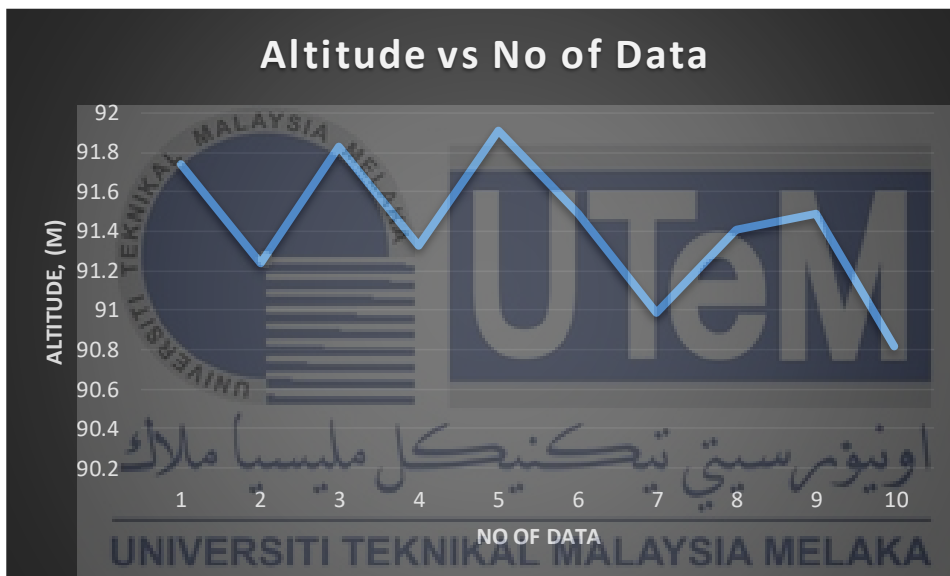


Figure 4.2 Test 1 Direct Connection for Altitude at rest condition

Based on Figure 4.3, for each of the ten data points, the temperature values, shown on the Y-axis, stay constant at 27°C. This uniformity in temperature measurements points to a continuous state of the external factors or experimental settings affecting temperature during the data gathering process.

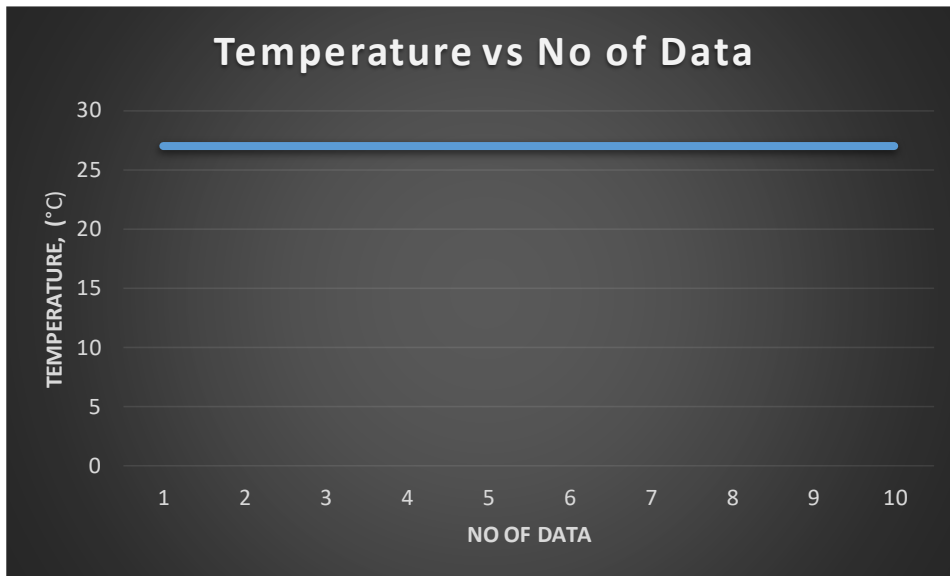


Figure 4.3 Test 1 Direct Connection for Temperature at rest condition

Figure 4.4 representation of the altitude data shows a generally stable pattern, with the numbers circling a core region. There is very little variation in the reported values of altitude; the lowest is 115.33, and the highest is 116.43. This narrow range implies that the Altitude is constant with only little variations. There are no notable peaks or dips on the graph, suggesting that the recorded data points show a constant trend in altitude. With the observed variances most likely falling within the predicted range of natural fluctuations, the overall pattern points to a reasonably stable altitude.

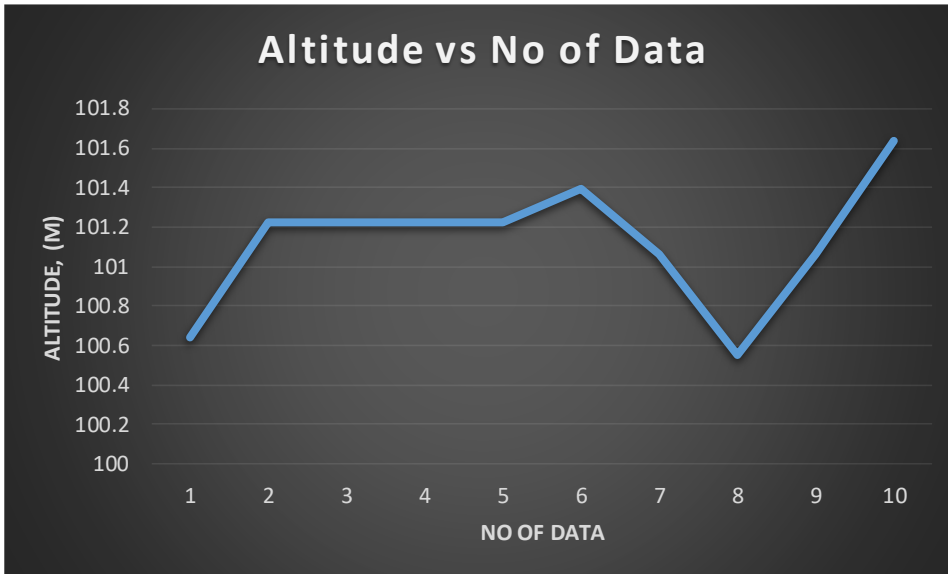


Figure 4.4 Test 2 Direct Connection for Altitude at rest condition

Figure 4.5 shows the temperature is depicted on the graph as a constant, horizontal line at the height of 27.3. This shows that the temperature did not vary during the recorded period and stayed at 27.3 for the duration of the dataset.

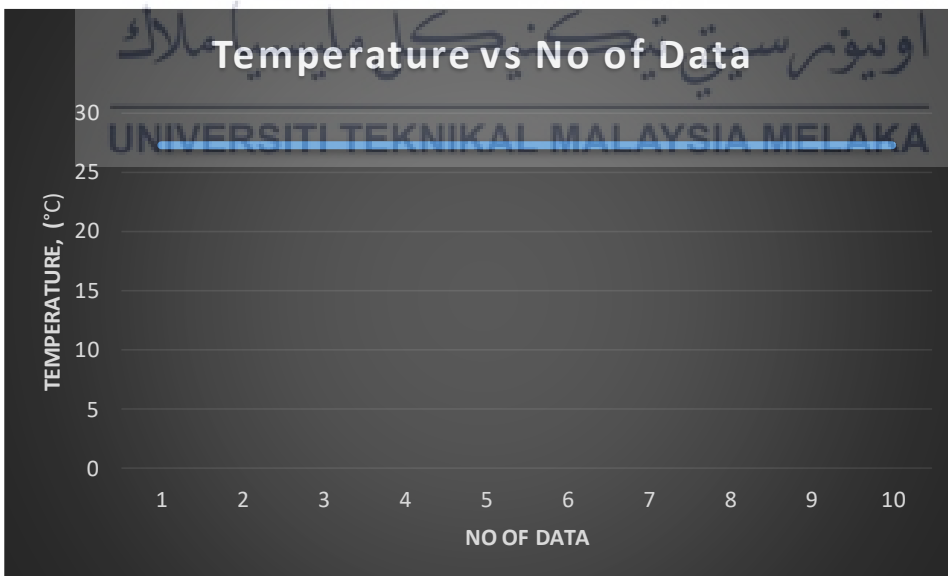


Figure 4.5 Test 2 Direct Connection for Temperature at rest condition

From Figure 4.6, it appears that the values in the collection correspond to measurements of altitude taken at various times. The average Altitude can be computed to give an overview. For this exact dataset, the average Altitude is roughly 115.77. This average provides a central point of reference and information about the overall trend of the dataset's Altitude values.

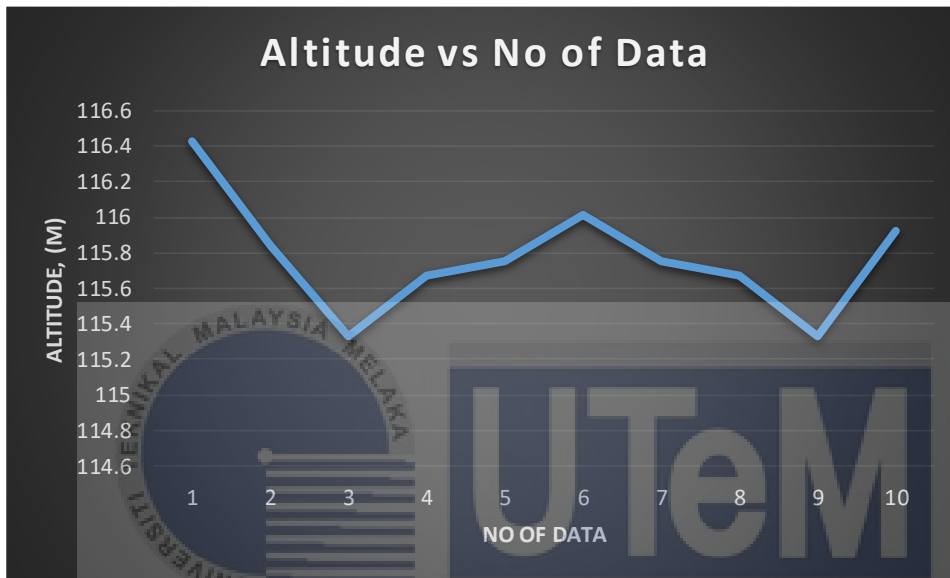


Figure 4.6 Test 3 Direct Connection for Altitude at rest condition

Figure 4.7 shows a steady and regular trend can be seen in the graphs, with an average value of 26.45 derived from the 10 data points. The observations show a very constant temperature over the course of the measured period, clustering closely around this mean. These variations seem to be within the predicted range around the centre value, notwithstanding a few slight swings, such a brief fall from 26.6 to 26.5 and a more noticeable decrease to 25.6. All things considered, the evidence points to a general trend in temperature that is centred around the determined average of 26.45.

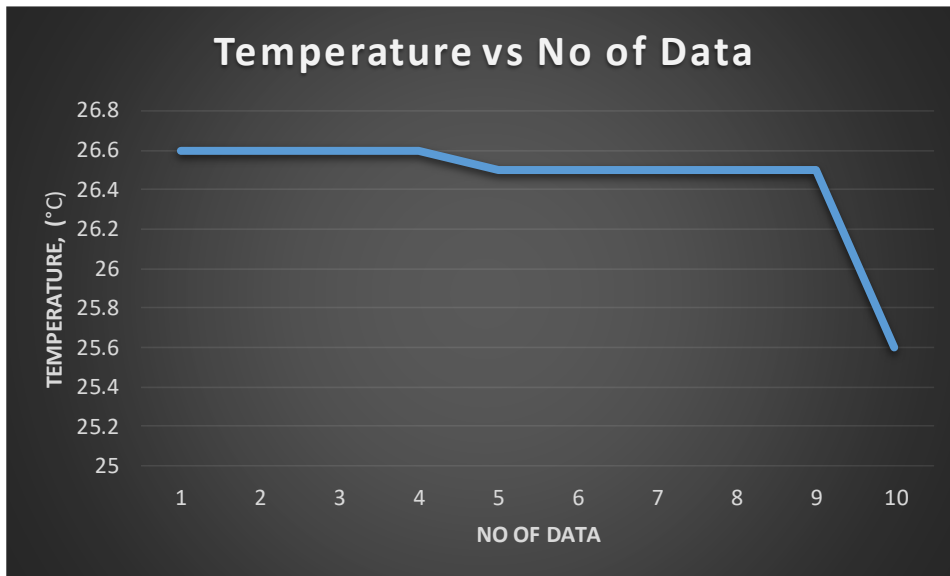


Figure 4.7 Test 3 Direct Connection for Temperature at rest condition

Based on Table 4.2, the X and Y coordinates in Test 1 vary from 2960 to 2992 and 2971 to 2983, respectively. There is a steady but modest change in both the X and Y values across measurements, suggesting that the tilt axis data is somewhat variable. The close proximity of X and Y values shows a probable link between the two axes.

Comparable features may be seen in Test 2, where X values span from 2960 to 2992 and Y values oscillate between 2971 and 2983. This variation suggests that the tilt axis mechanism is sensitive or responsive. Over the course of ten measurements, the system remains stable, with tiny deviations due to either intrinsic system nuances or minor external perturbations.

Test 3 records coordinates in the ranges 2946 to 2998 for X and 2911 to 2918 for Y. The two axes exhibit minor oscillations, which are consistent and suggest that the tilt axis system's response is under control. The connection between X and Y values is a remarkable observation as it implies synchronised movement or dependency between the two axes.

Ultimately, the examination of these experiments indicates a tilt axis system that is stable, sensitive, and may exhibit an X-Y coordinate connection. Additional investigation using statistical analysis or graphical displays may provide more light on the underlying trends or patterns in the tilt axis data. The foundation for a thorough comprehension of the nuances of the tilt axis system as it is represented in the dataset is laid by this summary.

Table 4.2 Dataset for Tilt Angle with direct USB cable connection at rest condition

No of Test	1		2		3	
Tilt Axis Data No	X (°)	Y (°)	X (°)	Y (°)	X (°)	Y (°)
1	2971	2977	3015	2951	2992	2914
2	2977	2983	3019	2950	2990	2912
3	2969	2977	3013	2955	2993	2911
4	2966	2975	3016	2951	2998	2918
5	2960	2971	3018	2950	2994	2917
6	2966	2975	3019	2953	2993	2912
7	2971	2978	3008	2947	2989	2911
8	2992	2977	3029	2950	2992	2911
9	2966	2978	3014	2952	2997	2914
10	2965	2976	3014	2955	2946	2914

Based on Figure 4.8, in a perfect state of rest, one would expect the sensor to remain in the same orientation. The tilting angle variations, however, imply that there may be some errors in the measured values even when the sensor is at rest. Environmental considerations, calibration complexities, and sensor sensitivity are a few examples of the variables that may affect these differences.

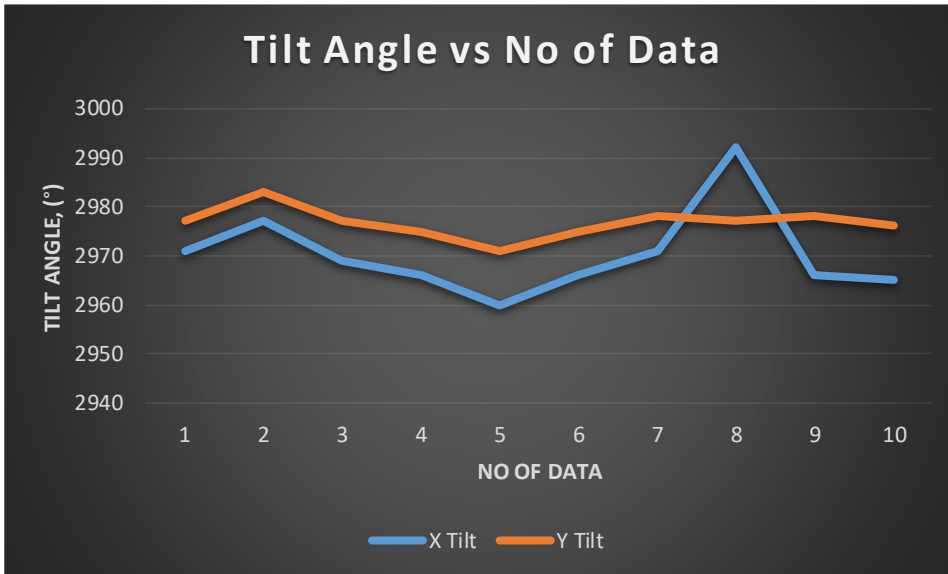


Figure 4.8 Test 1 Tilt Angle at rest condition for Direct Connection

Based on Figure 4.9, which shows tilting degrees as pairs of X and Y values, a steady and constant orientation is suggested. The data's repeating patterns and sparse fluctuations show that the sensor is holding its position steadily without seeing any appreciable tilt or orientation changes. Any little variations in the values could be ascribed to intrinsic noise or small variations in the sensor readings while the sensor is at rest. The repeating pairings of X and Y values most likely represent the sensor's stationary state.

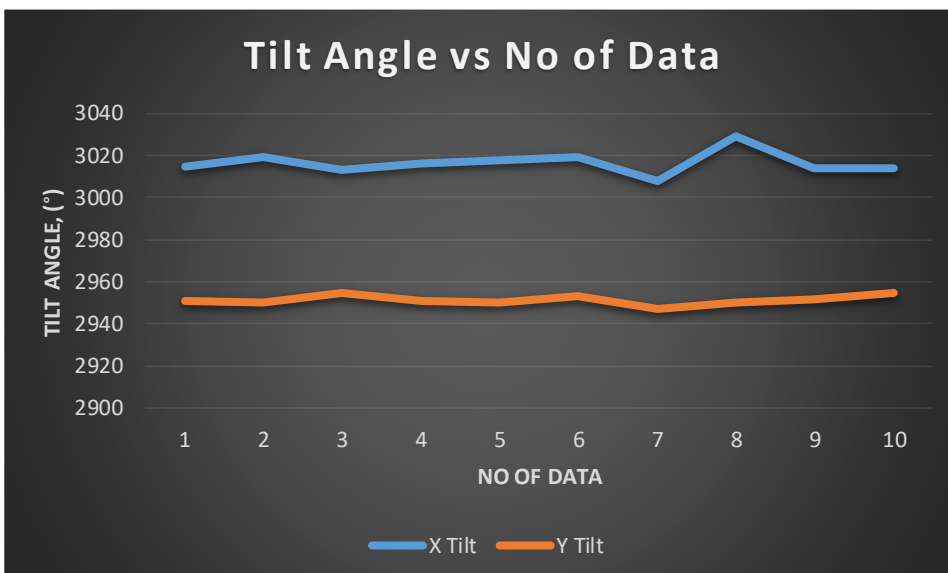


Figure 4.9 Test 2 Tilt Angle at rest condition for Direct Connection

Based on Figure 4.10 observation, the oscillations in the given data imply that there are small fluctuations in the recorded measurements even in a stationary state. The variations may result from a number of things, including intrinsic noise in the sensor, the sensor's susceptibility to outside variables, or precision constraints. It is possible that the sensor is picking up on minute alterations that add to the apparent differences in the tilting angles even though they are not indicative of actual physical movement.

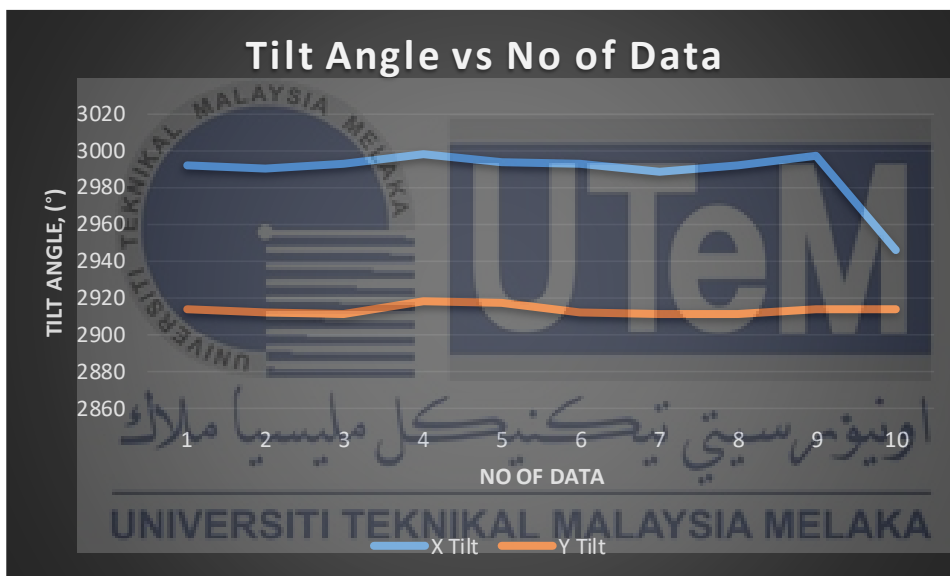


Figure 4.10 Test 3 Tilt Angle at rest condition for Direct Connection

4.3.1.2 Test at moving condition for direct USB cable connection

With reference to Table 4.3, the GY-68 barometric sensor was the subject of the first test. Throughout all data points, the recorded temperature values remained constant at 27.2°C with. Concurrently, the GY-68's altitude measurements showed a limited range from 91.24 to 91.91. Ten different data collection processes were used throughout the process.

The second test continued to use only the GY-68 barometer sensor, maintaining consistent temperature readings at 27.3°C. The GY68's reported altitude measurements showed a range of 100.55 to 101.64.

In the third test, which used only the GY-68 barometer sensor, the temperature was consistently recorded at 26.6°C. The GY68 also measured altitude, and its results ranged from 115.33 to 116.43.

In these testing, the GY-68 barometric sensor was the only one used, and it continuously provided stable and accurate temperature readings. This sensor proved that it could remain accurate in a variety of test conditions. The sensor system's direct USB cable connection to the laptop allowed for effective data transfer for real-time monitoring.

Based on Table 4.3 The average altitude based on sea level taken on test no 1 is 91.436 m, by subtracting it the actual data with the data taken, the initial altitude value is 20.436 m.

Table 4.3 Dataset for Altitude and Temperature with direct USB cable connection at moving condition

No of Test No of Data	1		2		3	
	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)
1	91.49	27.20	100.64	27.30	116.43	26.60
2	91.49	27.20	101.22	27.30	115.84	26.60
3	91.49	27.20	101.22	27.30	115.33	26.60
4	91.33	27.20	101.22	27.30	115.67	26.60
5	91.58	27.20	101.22	27.30	115.75	26.50
6	91.66	27.20	101.39	27.30	116.01	26.50
7	92.00	27.20	101.06	27.30	115.75	26.50

8	91.16	27.20	100.55	27.30	115.67	26.50
9	91.33	27.20	101.06	27.30	115.33	26.50
10	90.83	27.20	101.64	27.30	115.92	25.60
Average	91.436	27.20	101.122	27.30	115.77	26.45

The fluctuations in Altitude values corresponding to each measurement are shown in the Figure 4.11. The altitude measurements show a rather limited range, ranging from 90.82 to 91.74. The clustering of data points within this range of altitude highlights the accuracy with which the sensor measures altitude and bolsters its capacity to provide accurate data under a variety of conditions.

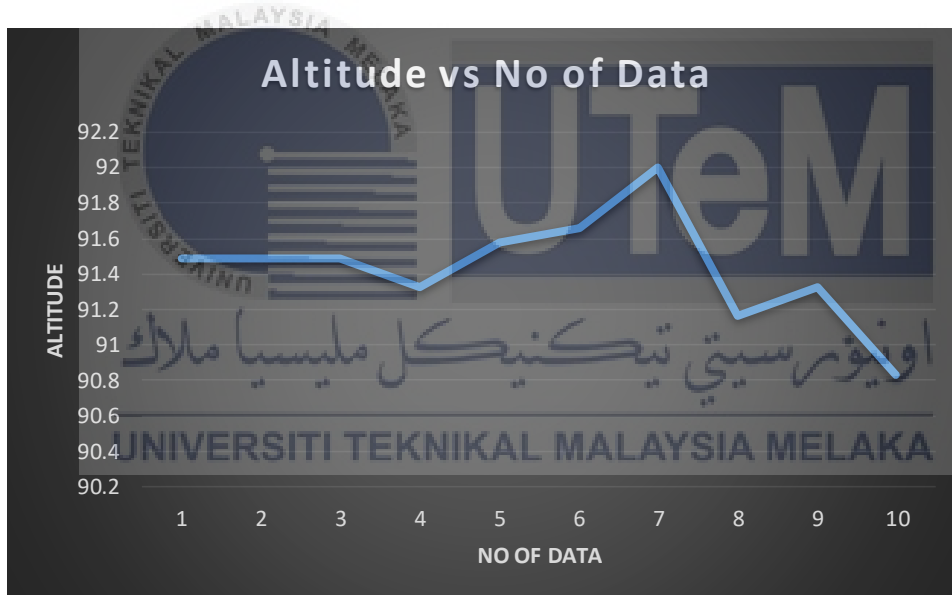


Figure 4.11 Test 1 Direct Connection for Altitude at moving condition

Figure 4.12 shows that the value is 27°C, which is a constant and stable temperature throughout all readings. A high degree of accuracy and stability in temperature measurements is indicated by the repeated use of the same temperature value, which each data point on the graph represents as a unique observation. The temperature stays at 27.2°C despite fluctuations in altitude.

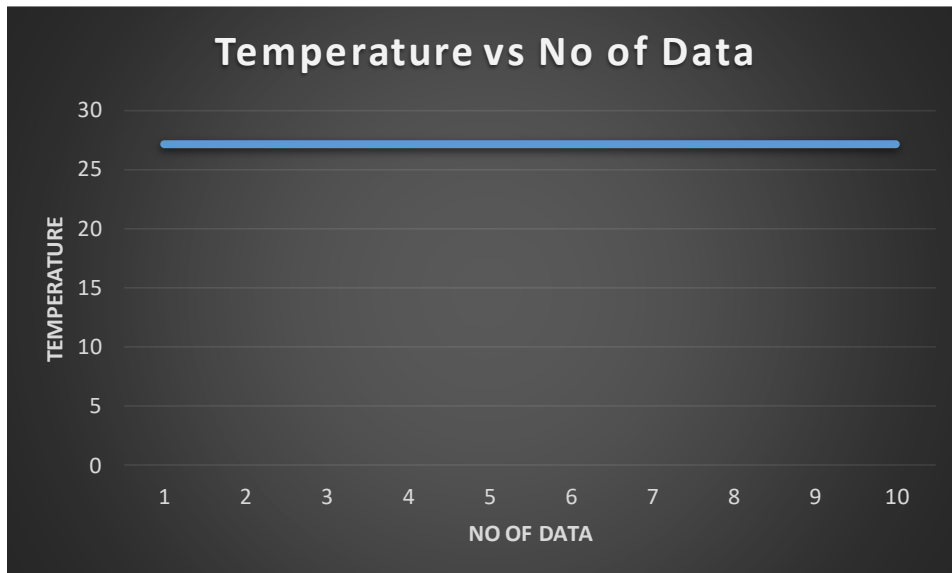


Figure 4.12 Test 1 Direct Connection for Temperature at moving condition

The fluctuations in Altitude values corresponding to each measurement are displayed on the Figure 4.13. The Altitude data show a rather tight range, ranging from 100.64 to 101.64. A consistent Altitude measurement across the recorded values is suggested by the close proximity of the data points on the graph, each of which represents a single measurement.

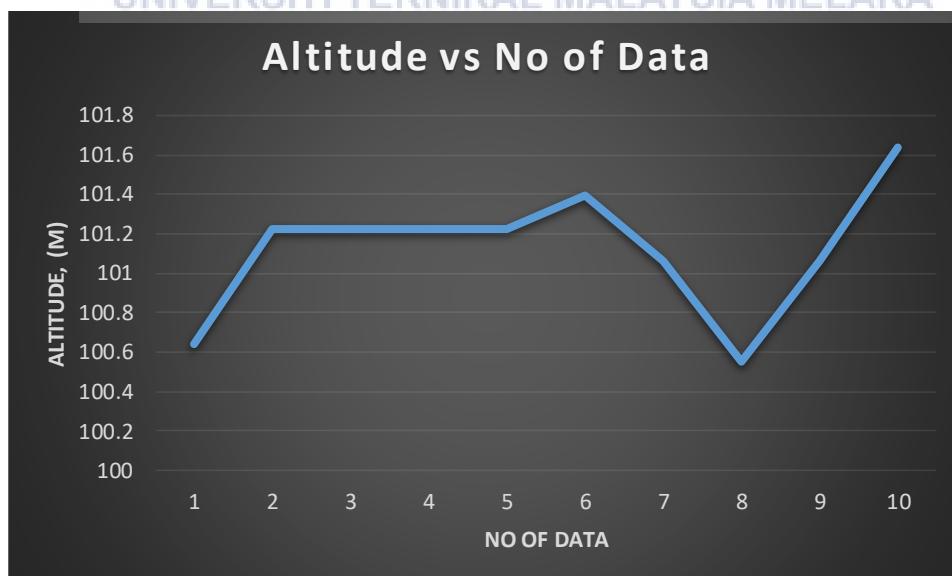


Figure 4.13 Test 2 Direct Connection for Altitude at moving condition

The constancy of the temperature readings over the course of the measurements is shown by the Figure 4.14. The reported temperature values are consistent at 27.3°C for the data. The same temperature figure occurring denotes a high degree of accuracy and consistency in temperature readings.

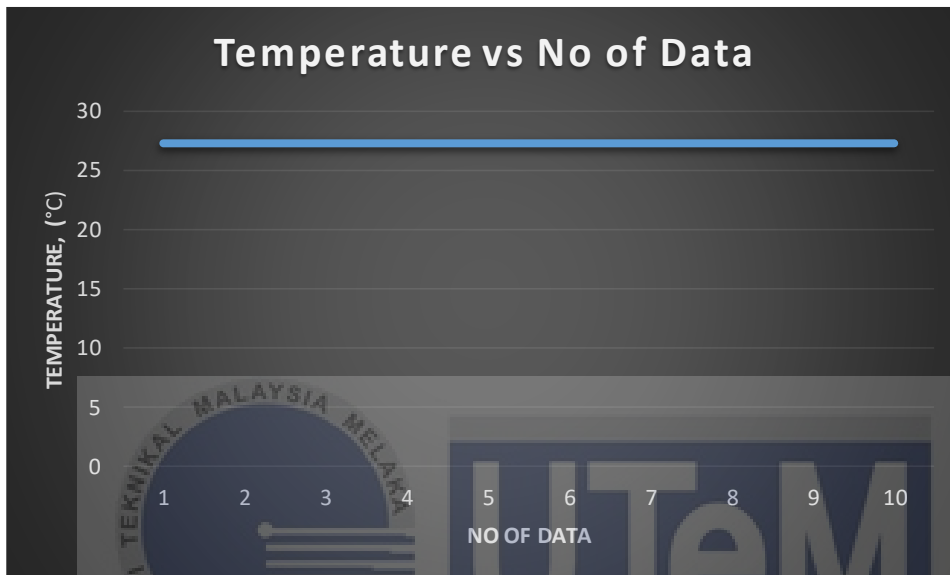


Figure 4.14 Test 2 Direct Connection for Temperature at moving condition

A distinct pattern can be seen in the Figure 4.15, where the numbers are constantly in the 115.25–115.92 range. This stability shows that the GY-68 barometric sensor can produce accurate and consistent readings of altitude, indicating that it is a capable device. The low variations in Altitude values highlight how reliably the sensor measures this parameter over the course of several observations.

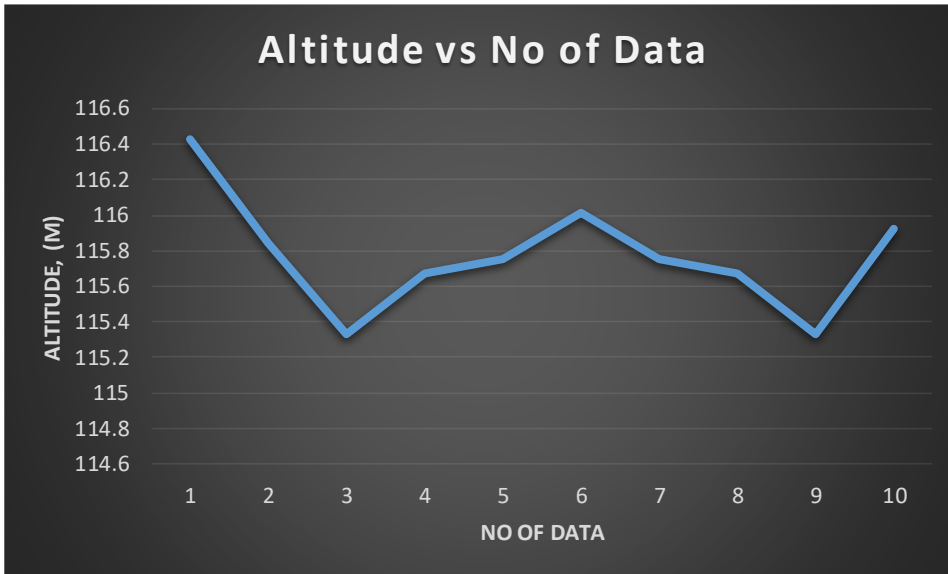


Figure 4.15 Test 3 Direct Connection for Altitude at moving condition

Figure 4.16 exhibits a noteworthy degree of constancy, as most readings remain consistent at 26.8°C. The temperature measurements are consistent, highlighting the sensor's accuracy in sensing temperature and proving its dependability at different altitudes. The two occasions in which the temperature momentarily lowers to 26.7°C show a slight variance, but they nevertheless support the sensor's general capacity to provide accurate and consistent temperature measurements.

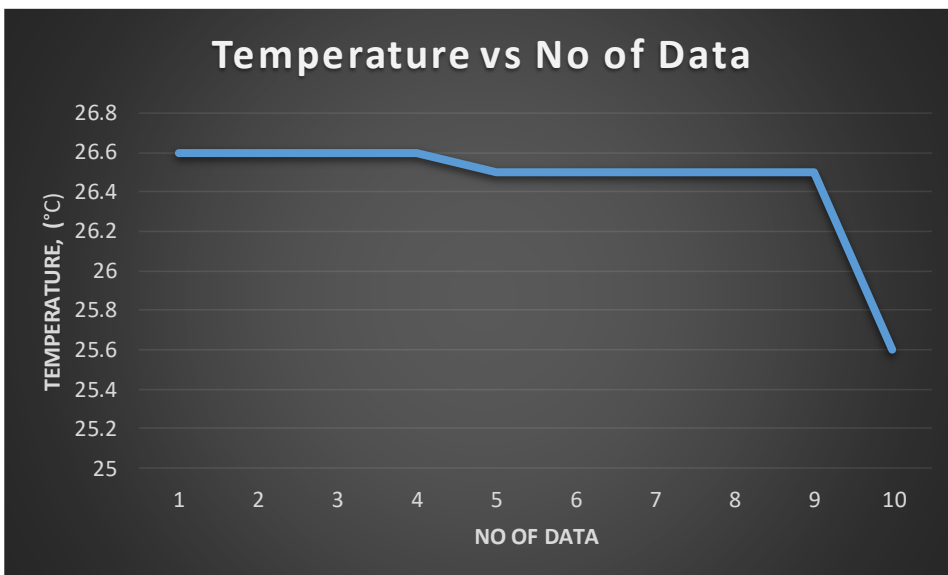


Figure 4.16 Test 1 Direct Connection for Temperature at moving condition

Based on Table 4.4, the X and Y coordinates for three tests performed on a tilt axis system are listed in the table. The X and Y coordinates in Test 1 vary from 2946 to 2992 and from 2919 to 2971, respectively. A constant but modest fluctuation is seen in both the X and Y values across observations, indicating some degree of variability in the tilt axis data. A possible association between the two axes is implied by the near proximity of the X and Y values.

Test 2 exhibits comparable trends, with Y values varying between 2908 and 3237 and X values spanning from 2960 to 2996. This variation suggests that the tilt axis mechanism is sensitive or responsive. Over the course of ten measurements, the system remains stable, with tiny deviations due to either intrinsic system nuances or minor external perturbations.

In Test 3, coordinates from 2946 to 2993 and from 2905 to 2928 are recorded for X and Y, respectively. The two axes exhibit minor oscillations, which are consistent and suggest that the tilt axis system's response is under control. The connection between X and Y values is a remarkable observation as it implies synchronised movement or dependency between the two axes. The information presented in the table is consistent with the features mentioned in the reference text, demonstrating a tilt axis system with stability, responsiveness, and possible correlation between X and Y coordinates.

Table 4.4 Dataset for Tilt Angle with direct USB cable connection at moving condition

No of Test	1	2	3
------------	---	---	---

Tilt Axis Data No	X (°)	Y (°)	X (°)	Y (°)	X (°)	Y (°)
1	2981	2971	2946	2928	2992	2919
2	2981	3237	3216	3187	2996	2908
3	2970	3430	3274	2977	2993	2921
4	2971	3659	3453	2965	2610	2920
5	2974	3792	3500	2903	2266	2920
6	2976	3810	3782	2897	2155	2912
7	2975	3829	3954	2896	2124	2905
8	2978	3834	3966	2934	2104	2906
9	2973	3771	3943	2921	2812	2928
10	2955	3679	3283	2909	2951	2867

Based on Figure 4.17, the dataset that is provided is made up of pairs of values that correspond to X and Y coordinates. It is assumed that values for Y that are greater than 3500 signify a tilt to the right. The first data point in the analysis, (2981, 2971), indicates a comparatively neutral or slightly slanted attitude. But the data points that come after (2981, 3237), (2970, 3430), and (2971, 3659) show a steady trend of rising Y values above the 3500 mark. This trend suggests a steady and progressive tilt to the right. The idea of a continuous rightward tilt is supported by the data sequence's continuation with Y values continuously over 3500. For the ninth data point (2973, 3771), there is a modest reduction in the Y value, which points to a possible small adjustment in the rightward tilt. Even so, the last data point (2955, 3679) continues to exhibit a Y value above 3500, supporting the general pattern of a tilt to the right. To sum up, the dataset shows how a tilt sensor can detect movements of tilting to the right when a threshold is reached; changes in the X and Y coordinates may be used to record the tilt locations as they change.

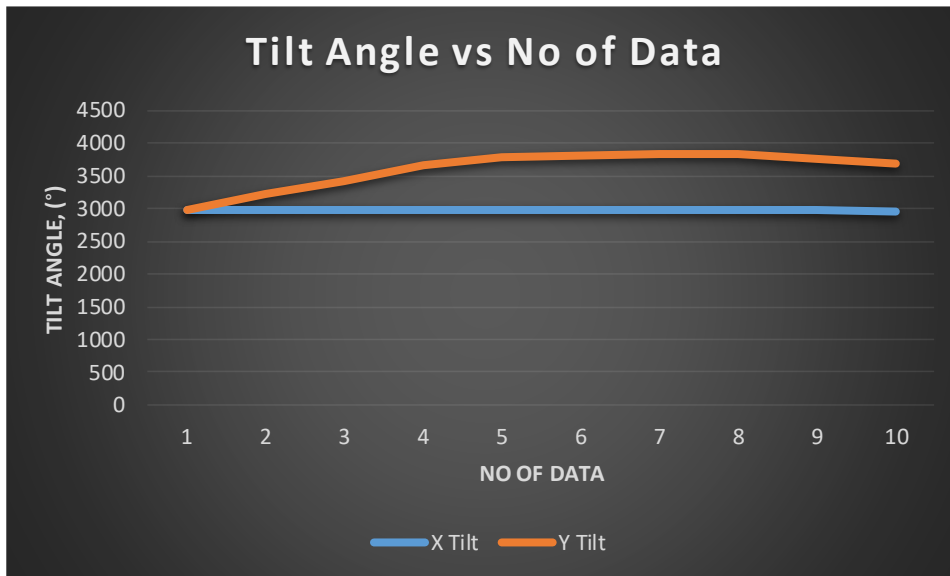


Figure 4.17 Test 1 Tilt Angle at moving condition for Direct Connection

Based on Figure 4.18, the data point (2946, 2928), which represents an initial position, is where the dataset starts. Following data points show a progressive increase in X values. The X value exceeds the designated threshold of 3500 when we get to (3500, 2903), suggesting an upward tilt in the sensor. On the other hand, "over-tilted" suggests a possible excessive tilt that goes beyond a reasonable bound. The sensor seems to keep an upward inclination as we go through the dataset, with X values like 3782, 3954, and 3966 staying over 3500. Although the Y values are not constant, the general pattern points to a persistent tilt. The data point (3283, 2909), where the X value falls below 3500, marks the end of the sequence. This can point to the need for a possible fix or modification that would bring the sensor back to a more neutral setting. In summary, the dataset reveals a period of over-tilting upward, as evidenced by X values continuously surpassing 3500. Although there appears to be some variation in the orientation, the overall trend is consistent with an upward tilt above a given threshold, as indicated by the varying Y values. The last data point, when X is less than 3500, suggests that the sensor orientation has to be corrected.

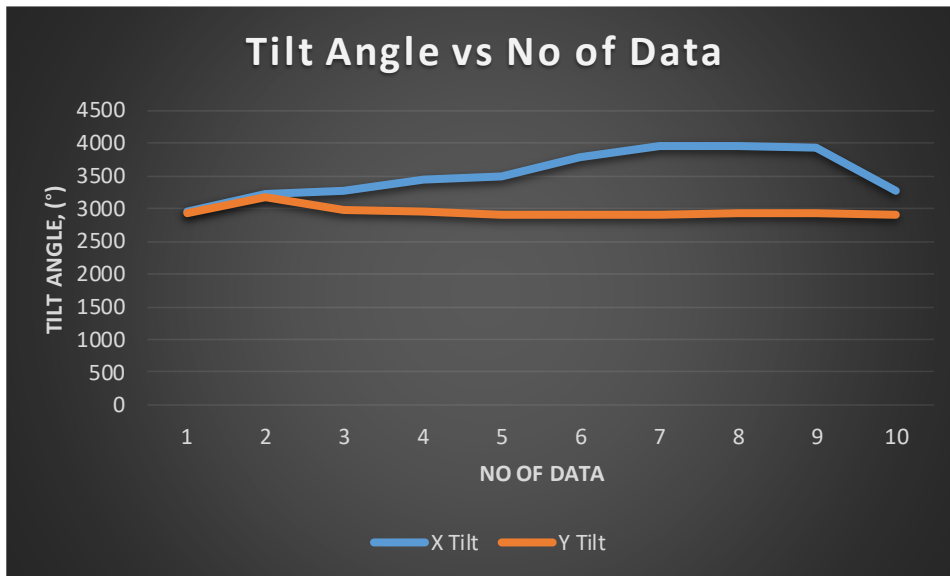


Figure 4.18 Test 2 Tilt Angle at moving condition for Direct Connection

The data point (2992, 2919) at the beginning of the dataset based on Figure 4.19 indicates a starting position. The X values progressively get smaller as we move through the next data points. The X value drops below the designated 2400 threshold when we get to (2266, 2920), suggesting an excessive downward tilt of the sensor. This may indicate that the sensor is beginning to tilt too much downward. As one goes through the dataset, the sensor seems to maintain a downward tilt with X values like 2155, 2124, and 2104. Variations in the Y values also point to a possible change in the orientation of the sensor. The data point (2951, 2867), where the X value increases over 2400, marks the end of the sequence. This could point to the need for a possible fix or modification that could put the sensor back in a more neutral position. In conclusion, the dataset shows a phase of excessive downward tilt, with X values continuously dropping below 2400. The fluctuating Y values imply that the orientation of the sensor has changed correspondingly. When X increases beyond 2400 in the last data point, it might indicate that the sensor's position has been adjusted or corrected, possibly bringing it back to a more neutral tilt.

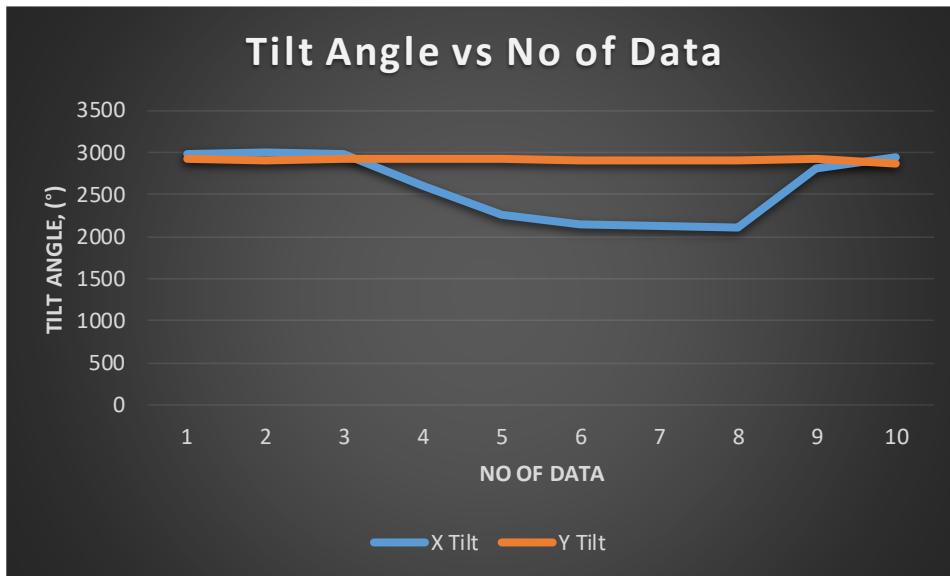


Figure 4.19 Test 3 Tilt Angle at moving condition for Direct Connection

4.3.2 Blynk IoT

By establishing a wireless link between the custom sensor system and different devices using Blynk IoT. By doing away with the requirement for physical connections and enabling seamless communication, this novel approach dramatically improves data collection efficiency. Blynk IoT enables the sensor system to wirelessly transfer data to linked devices, providing a more adaptable and practical means of data collection and monitoring. This makes the setup process easier and improves data accessibility overall, enabling real-time monitoring and analysis. By integrating Blynk IoT into the system, sensors can be accessed and controlled remotely, offering a flexible and effective way to manage data collecting. In addition to simplifying setup, wireless networking presents opportunities for scalability and flexibility across a range of applications. The configuration of the digital pin and dashboard to receive the transmitted data is shown in Figure 4.20 and Figure 4.21.

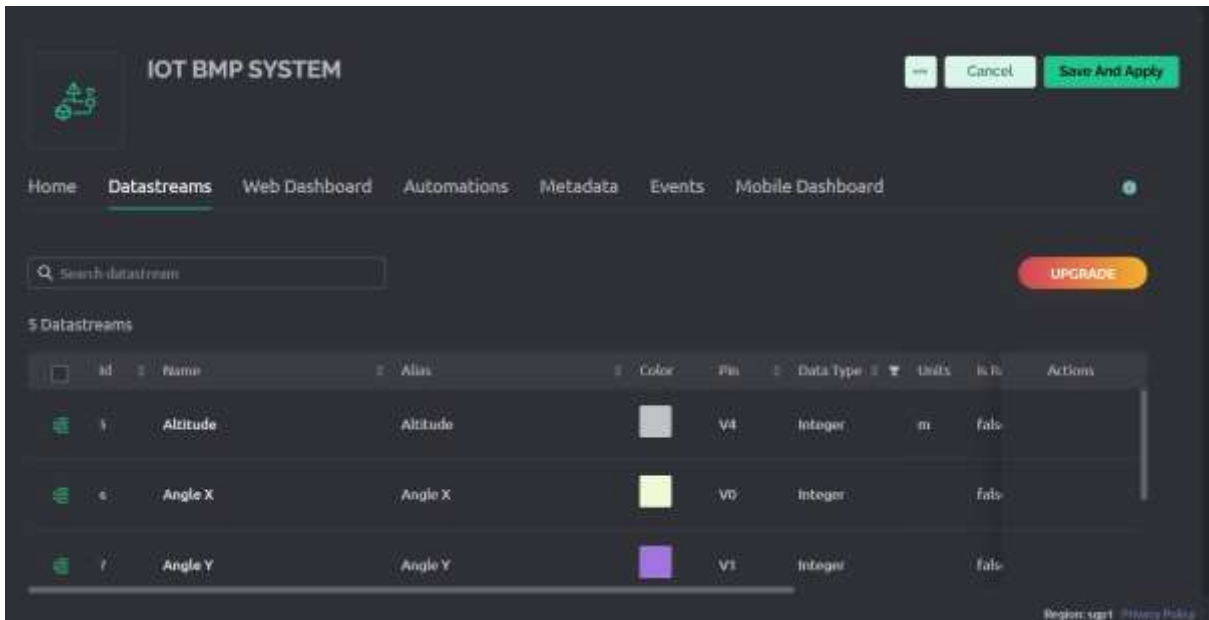


Figure 4.20 Blynk Datasreams Setup



Figure 4.21 Blynk Dashboard Setup

The extra function that was added to the original function to allow connectivity between the sensor system and the Blynk IoT system is shown in Figure 4.22.

```
sketch_arduino.cpp
1 #define BLYNK_TEMPLATE_ID "TSP2021004"
2 #define BLYNK_TEMPLATE_NAME "IOT BMP SYSTEM"
3 #define BLYNK_AUTH_TOKEN "XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
4 #define BLYNK_PRINT Serial
5
6 //
7 // LIBRARY STUFF
8 //
9
10 // ----- Blynk WiFi connectivity (ESP8266) ----- //
11 #include <WiFi.h>
12 #include <WiFiClient.h>
13 #include <ESP8266WiFi.h>
14
15 // ----- Blynk Library ----- //
16 #include <BlynkArduino.h>
17 #include <math.h>
18 #include <BlynkEsp8266.h>
19
20 // ----- Blynk BLYNK_API ----- //
21 #include <BlynkAPI.h>
22
23 // ----- User API ----- //
24 char ssid[] = "bilal_2_dunghani1"; // ... WPA or WPA2 NETWORK (without IP, better name) ... //
25 char pass[] = "00lay9222"; // ... PASSWORD OF YOUR NETWORK (without IP, better name) ... //
26
```

Figure 4.22 Blynk Additional Function

Figure 4.23 illustrates how a notification is sent straight to a smartphone when a specific threshold value is reached, allowing the pilot to be more aware of their surroundings.



Figure 4.23 Blynk Wireless Notifications

4.3.3 Data collection using IoT

4.3.3.1 Test at rest condition

Table 4.5 shows altitude values in the first test range from 92.33 to 92.83, illustrating the variation in this dataset. Accordingly, during the course of the ten data points, the recorded Temperature values stay consistent at 26.2°C. This test highlights the stability of the sensor by showing consistent Temperature readings even with slight variations in Altitude.

The second test keeps temperature readings steady at 27.3°C. On the other hand, the Altitude values show a greater range, ranging from 99.04 to 99.88. This variation in altitude aligns with the wider range of altitudes noted in the second test reference. This test clearly shows that the sensor can provide consistent temperature measurements at different altitudes.

A fresh set of circumstances, including constant temperature measurements of 26.2°C, are the subject of the third test. The GY-68 sensor's recorded altitude data show a range of 113.15 to 114.24. This range is wider than in the earlier tests, suggesting a distinct set of environmental circumstances. The consistent temperature readings, in spite of the fluctuation in altitude, support the sensor's capacity to deliver accurate data in a variety of situations. For Tests 1, 2, and 3, the average altitude values are 92.564, 99.462, and 113.745, in that order. In a similar vein, the three tests' average temperature readings are 26.2°C, 26.73°C, and 26.4°C. These averages provide a comprehensive view of the sensor's performance in various test situations.

The temperature readings are consistently stable, indicating that the GY-68 barometer sensor is reliable and suitable for real-time monitoring applications. It also

suggests that the sensor maintains accuracy under a variety of situations. Tests 2 and 3's larger altitude ranges demonstrate the sensor's adaptability to a variety of environmental circumstances.

Based on Table 4.5 The average altitude based on sea level taken on test no 1 is 92.564 m, by subtracting it the actual data with the data taken, the initial altitude value is 21.564 m.

Table 4.5 Dataset for Altitude and Temperature with Blynk IoT at rest condition

No of Test No of Data	1		2		3	
	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)
1	92.33	26.20	99.46	26.80	113.74	26.40
2	92.83	26.20	99.46	26.70	113.74	26.40
3	92.33	26.20	99.38	26.80	114.07	26.40
4	92.50	26.20	99.88	26.80	114.24	26.40
5	92.58	26.20	99.21	26.70	114.07	26.40
6	93.42	26.20	99.55	26.70	113.40	26.40
7	91.91	26.20	99.80	26.70	113.15	26.40
8	93.00	26.20	99.46	26.70	113.06	26.40
9	91.91	26.20	99.04	26.70	114.24	26.40
10	92.83	26.20	99.38	26.70	113.74	26.40
Average	92.564	26.20	99.462	26.73	113.745	26.40

In the Figure 4.24, the numbers range from 91.91 to 93.42, indicating a relatively limited range. This implies that the GY-68 barometric sensor reliably captures Altitude

readings within this particular range when it is stationary. The narrow range of Altitude values indicates how consistently and accurately the sensor records variations in air pressure.

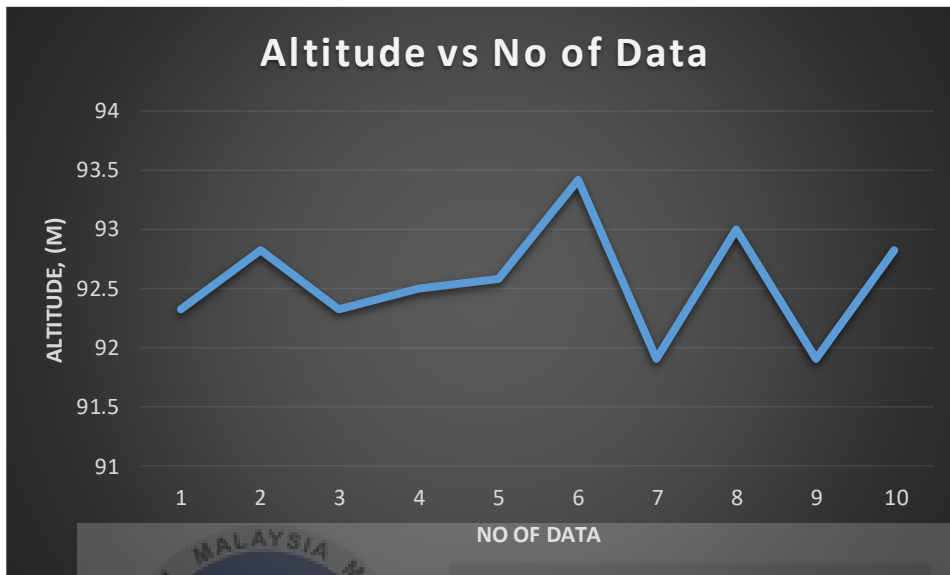


Figure 4.24 Test 1 Blynk IoT Altitude at rest condition

The Figure 4.25 shows that, for all ten data points, the average temperature is 26.2°C. The consistency of the temperature readings suggests that the GY-68 sensor keeps a steady temperature reading when it is at rest. The sensor's capacity to provide accurate and consistent temperature measurements is demonstrated by the recurrent recurrence of the same temperature value. This capability is crucial for applications where upholding a steady temperature baseline is imperative.

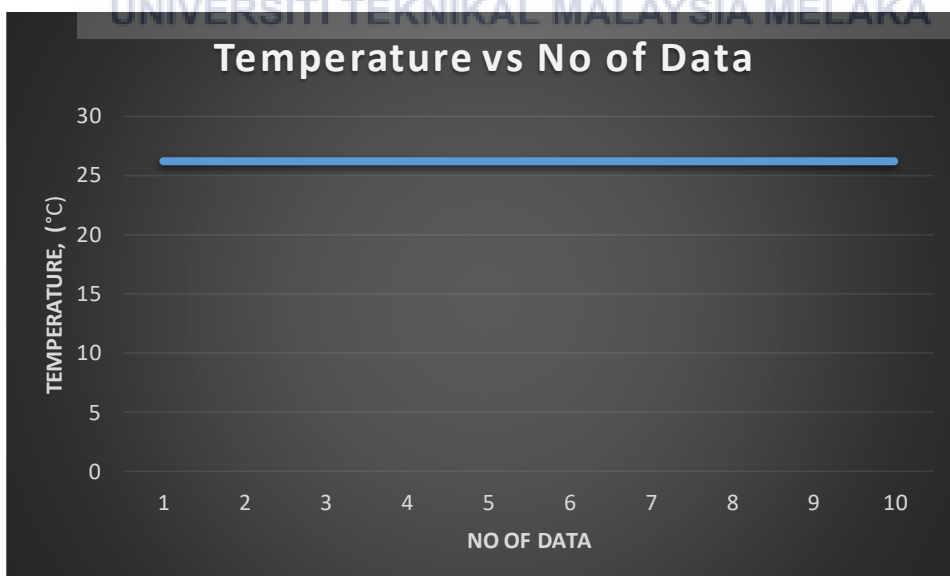


Figure 4.25 Test 1 Blynk IoT Temperature at rest condition

When looking at the Figure 4.26, one can see that the values are constantly in the restricted range of 99.04 to 99.88. This suggests that the GY-68 barometric sensor offers a consistent set of altitude readings when it is at rest. The sensor's accuracy in measuring changes in atmospheric pressure is demonstrated by the small fluctuations in Altitude measurements, which also imply that it will remain reliable in providing an accurate Altitude reading even when there is no activity.



Figure 4.26 Test 2 Blynk IoT Altitude at rest condition

In Figure 4.27, the recorded values demonstrate a minor variance from 26.8°C to 26.7°C over the ten data points. This shows that the GY-68 sensor retains a relatively steady temperature reading even while it is at rest. Though small environmental variations could be the cause of the temperature values' slight variations, the sensor's general consistency suggests that it can provide accurate and constant temperature readings even when there is no activity.

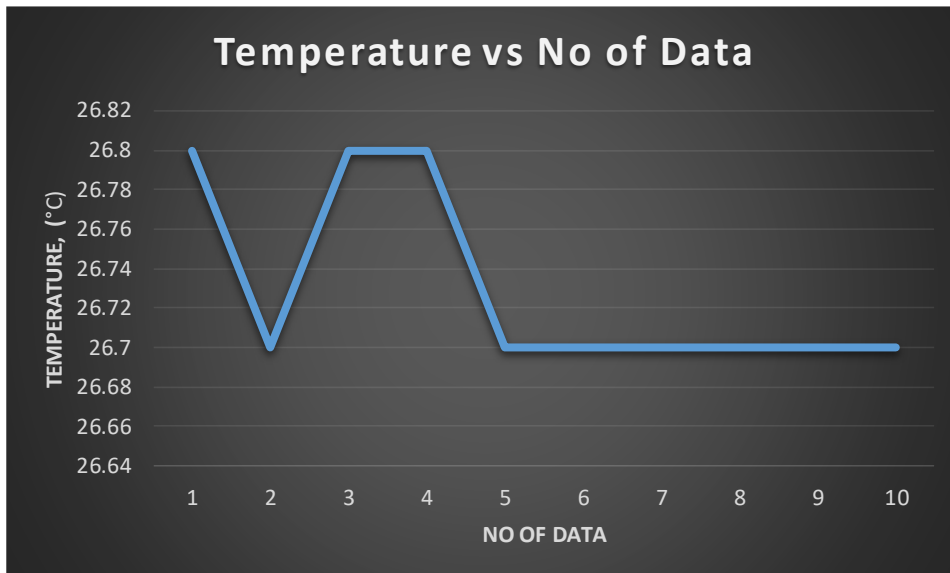


Figure 4.27 Test 2 Blynk IoT Temperature at rest condition

The GY-68 barometric sensor recorded a stable range of readings, ranging from 113.06 to 114.24, on Figure 4.28 while it was at rest. The sensor's accuracy in recording variations in atmospheric pressure, even when there is a lapse in activity, is demonstrated by the stability of its altitude readings.

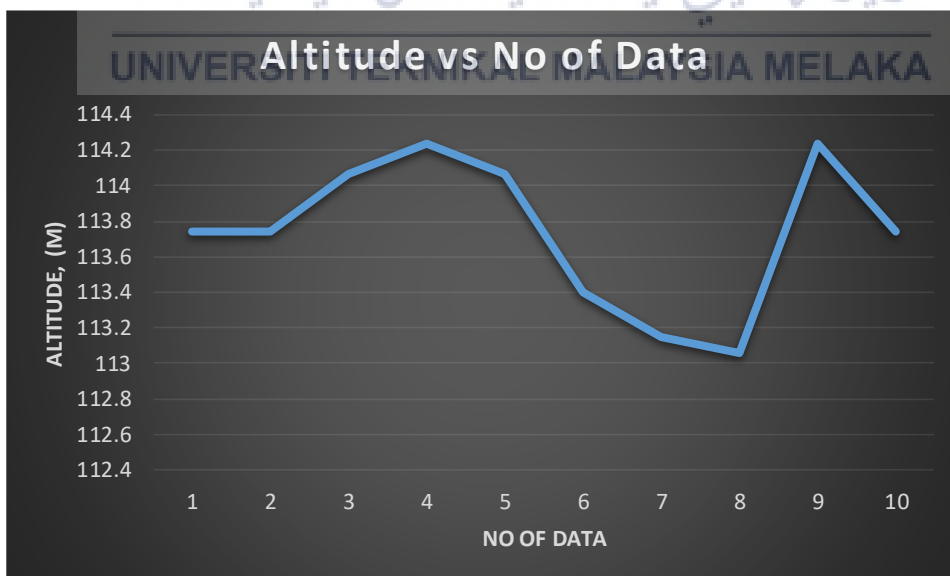


Figure 4.28 Test 3 Blynk IoT Altitude at rest condition

The recorded values in Figure 4.29 stay at 26.4°C for each of the ten data points. This consistent temperature reading highlights the GY-68 sensor's capacity to deliver a steady and accurate temperature reading even at rest.

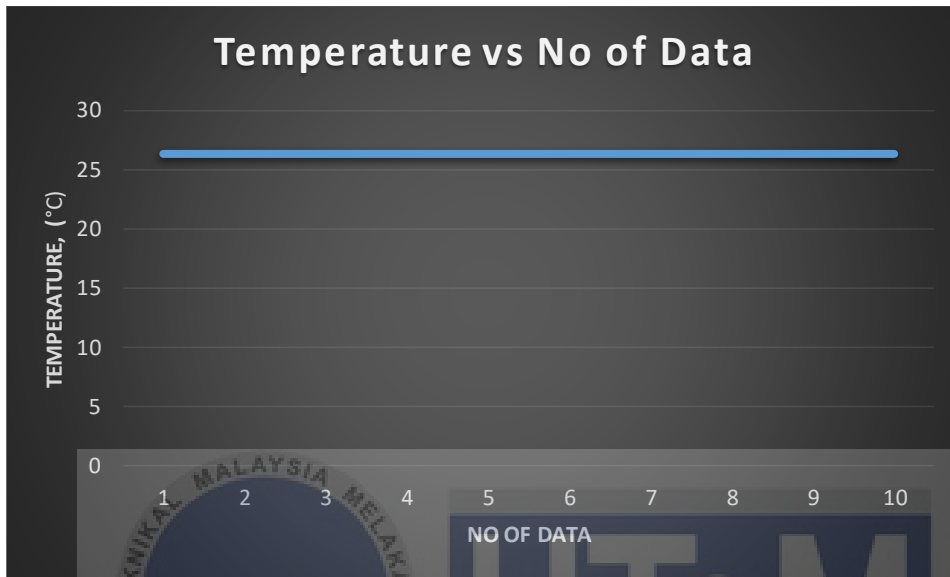


Figure 4.29 Test 3 Blynk IoT Temperature at rest condition

Based on Table 4.6, Test 1, the Y coordinates alternate between 2921 and 2974, and the X coordinates span from 2910 to 3011. The data on the tilt axis appears to exhibit some degree of fluctuation, as evidenced by the constant but modest variance observed across measurements. Test 1's X and Y values are close together, indicating that there may be a correlation between the two axes.

In a similar vein, Test 2 has Y values that oscillate between 2882 and 2974 and X values that range from 2960 to 3008. This variation suggests that the tilt axis mechanism is sensitive or responsive. Across ten measurements, the system remains stable, with tiny shifts ascribed to either intrinsic system nuances or small-scale external variables. These shifts are consistent with the features listed in the reference text.

Test 3 records coordinates in the ranges 2905 to 3024 for X and 2912 to 2972 for Y. The two axes exhibit minor oscillations, which are consistent and suggest that the tilt axis system's response is under control. The findings in the reference text are supported by the correlation between X and Y values in Test 3, which points to synchronised movement or interdependence between the two axes.

To sum up, the information shown in the table illustrates a tilt axis system with potential correlation between X and Y coordinates, stability, and responsiveness, all of which are outlined in the reference text.

Table 4.6 Dataset for Tilt Angle with Blynk IoT at rest condition

No of Test Tilt Axis Data No	1		2		3	
	X (°)	Y (°)	X (°)	Y (°)	X (°)	Y (°)
1	2910	2921	2998	2940	3011	2969
2	2913	2922	3001	2956	3011	2967
3	2914	2923	3005	2951	3011	2974
4	2912	2927	2930	2882	3017	2972
5	2907	2925	3005	2945	3009	2970
6	2916	2921	3003	2944	3014	2966
7	2918	2924	3005	2943	3024	2972
8	2883	2890	3008	2958	3015	2945
9	2912	2919	3003	2954	2999	2948
10	2913	2921	2960	2912	2997	2997

Based on Figure 4.30, the X and Y coordinates captured during a state of rest by a sensor—specifically, the SN-ADXL335-CY—are shown on the graph. The sensor readings should be fairly constant during rest, resulting in a graph with a horizontal line representing the X values and a vertical line representing the Y values. Nonetheless, throughout the course of the 10 observations, the observed graph displays modest variations in both X and Y coordinates. This means that the sensor is not totally steady and shows small changes even

when it is thought to be at rest. A number of variables, including intrinsic sensor noise, measurement accuracy, and external conditions impacting the SN-ADXL335-CY, could be responsible for these variations.

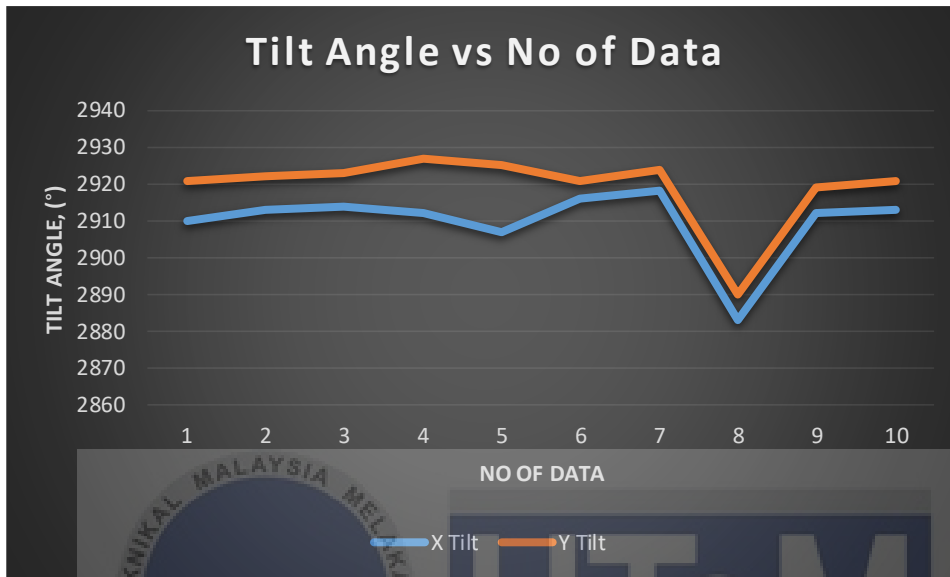


Figure 4.30 Test 1 Tilt Angle at rest condition for Blynk IoT

Figure 4.31 shows the X and Y coordinates that the SN-ADXL335-CY sensor recorded while at rest are shown in the graph that is being shown. The sensor readings should be quite stable in a resting state, with the X and Y values forming a horizontal and vertical line, respectively. Though not purposefully moving during the rest state, the observed graph shows discernible variations in both X and Y coordinates across the 10 observations that demonstrate the sensor's sensitivity.

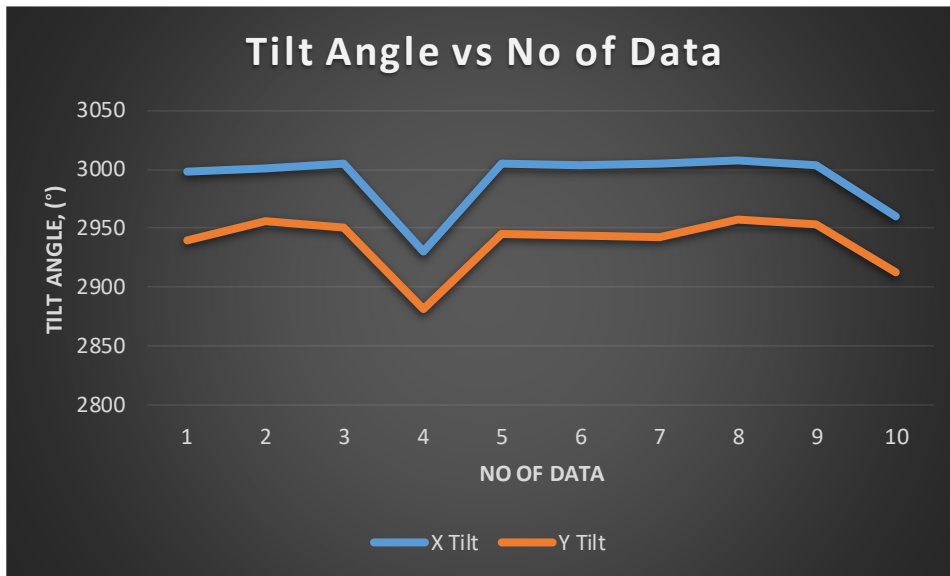


Figure 4.31 Test 2 Tilt Angle at rest condition for Blynk IoT

Based on Figure 4.32, the horizontal and vertical lines on the graph would ideally remain constant in the ideal rest state for the X and Y variables. However, a systematic offset or calibration difference in the sensor may be indicated by the observed variations in both axes across the measurements. In conclusion, investigating the possibility of systemic biases or calibration problems within the sensor might help to provide a more nuanced interpretation of the observed fluctuations in the graph during the assumed rest state, in addition to taking noise and sensitivity into account.

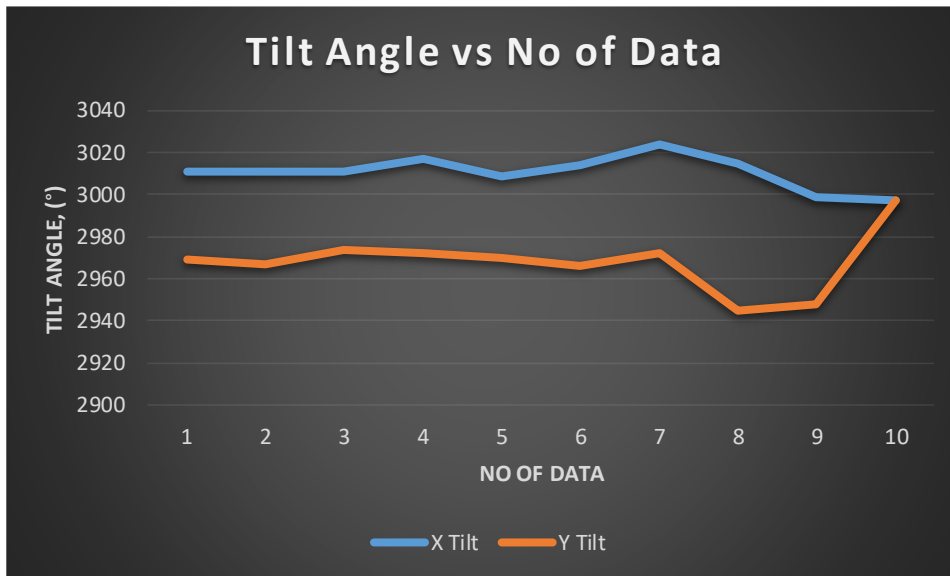


Figure 4.32 Test 3 Tilt Angle at rest condition for Blynk IoT

4.3.3.2 Test at moving condition for Blynk IoT

Ten data points with recorded temperature and altitude readings from three different experiments utilising a GY-68 barometer sensor are shown in Table 4.7. In the initial test, the temperature stays constant at 26.2°C but the altitude varies between 92.33 and 92.83. This highlights the stability of the sensor and demonstrates its capacity to deliver accurate data even in the presence of minute variations in altitude.

In the second test, the temperature remains at 27.3°C, but the range of altitude values is wider, ranging from 99.04 to 99.88. This deviation is consistent with the larger range of altitudes observed in the second test, indicating that the sensor can provide reliable temperature readings at a variety of elevations.

In the third test, the temperature remains constant at 26.2°C, whereas the altitude varies between 113.15 and 114.24, signifying different environmental conditions. The sensor's accuracy in a variety of scenarios is highlighted by its ability to produce constant

Temperature readings even in the face of altitude variations. The sensor's performance under varied situations is further supported by the average temperature and altitude results for each test.

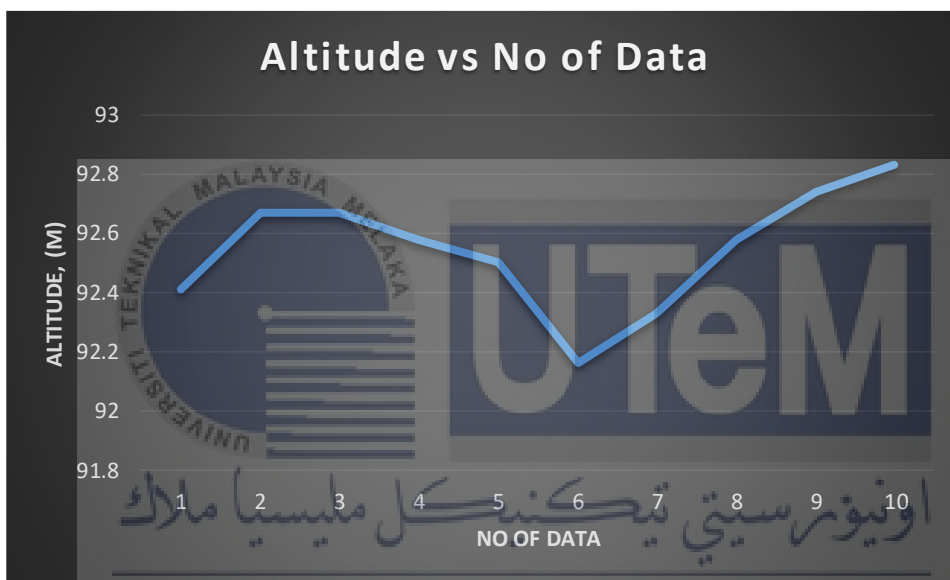
In summary, the greater altitude ranges seen in Tests 2 and 3 demonstrate the GY-68 barometer sensor's dependability in giving steady temperature readings and adjusting well to various environmental conditions.

Based on Table 4.7 The average altitude based on sea level taken on test no 1 is 92.547 m, by subtracting it the actual data with the data taken, the initial altitude value is 21.547 m.

Table 4.7 Dataset for Altitude and Temperature with Blynk IoT at moving condition

No of Test No of Data	1		2		3	
	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)	Altitude, Sea Level (m)	Temperature (°C)
1	92.41	26.30	100.05	26.40	113.32	26.30
2	92.67	26.30	99.55	26.40	113.32	26.30
3	92.67	26.30	99.54	26.40	113.48	26.30
4	92.58	26.30	99.80	26.40	112.98	26.30
5	92.50	26.30	99.38	26.40	114.07	26.30
6	92.16	26.40	100.38	26.40	113.82	26.30
7	92.33	26.40	99.29	26.40	113.90	26.30
8	92.58	26.40	99.63	26.40	113.74	26.30
9	92.74	26.40	99.21	26.40	114.66	26.30
10	92.83	26.30	99.21	26.50	114.66	26.30
Average	92.547	26.34	99.604	26.41	113.795	26.30

Figure 4.33 indicates that the Altitude values span a rather small range within this dataset, ranging from 92.41 to 92.83. With the exception of the sixth data point, where the temperature is recorded at 26.4°C, the majority of the data points show consistent temperature readings of 26.3°C, indicating a steady testing environment. This divergence can be a sign of a small change that occurred at that particular time. Even if there aren't many variations in this set, the altitude variations can nevertheless indicate the sensor's ability to detect minute variations in height.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA
 Figure 4.33 Test 1 Blynk IoT Altitude at moving condition

Figure 4.34 indicates that for the most of the observations, the readings show a remarkable stability at 26.3°C, with the exception of the sixth and tenth data points, when the temperature is recorded at 26.4°C and 26.3°C, respectively. The majority of the dataset's temperature measurements are consistent, which highlights the sensor's dependability in keeping the temperature steady despite the circumstances. The few differences that were seen could be explained by natural oscillations or external influences on the sensors. In

conclusion, the simultaneous study of temperature and altitude demonstrates how well the sensor records minute variations in altitude while retaining steady temperature values.

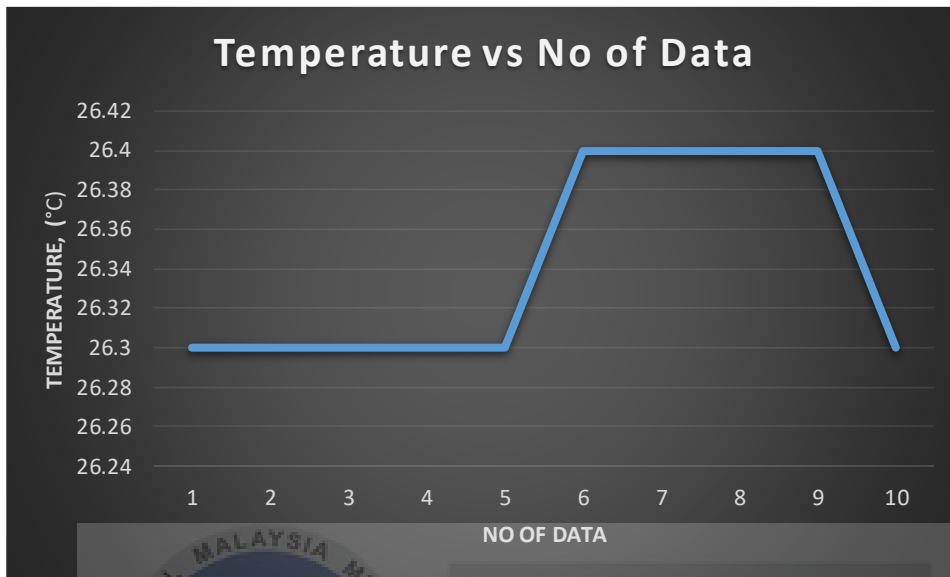


Figure 4.34 Test 1 Blynk IoT Temperature at moving condition

Figure 4.35 Altitude values show that they vary from 99.21 to 100.38, suggesting a wider range in this dataset than in the prior one. This shows that the range of elevations experienced in the matching test was more important. The Temperature readings, however, remain continuously at 26.4°C for the first nine data points, indicating the sensor's capacity to maintain a steady temperature even when faced with fluctuating altitudes. The last data point shows that the temperature increased slightly to 26.5°C while the height remained at 99.21. This divergence demonstrates the sensor's sensitivity to minute changes and may point to a particular condition or circumstance during that measurement.

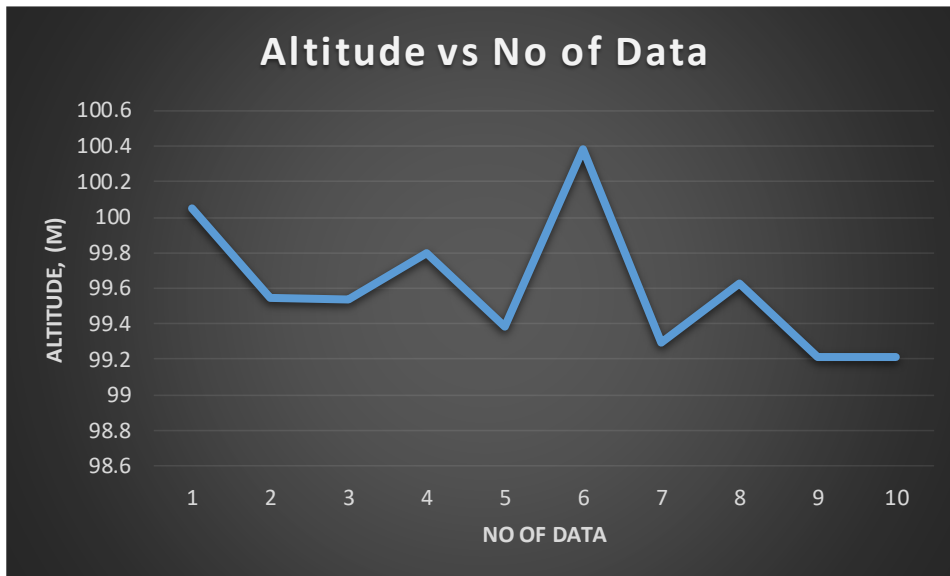


Figure 4.35 Test 2 Blynk IoT Altitude at moving condition

The temperature readings in Figure 4.36, 26.4°C are consistently maintained for the first nine data points, indicating that the sensor can maintain a consistent temperature even at varying elevations. The tenth data point shows a tiny increase in temperature to 26.5°C, which is equivalent to an altitude of 99.21 and introduces a slight divergence in temperature. This discrepancy may be the result of experimental circumstances particular to that particular measurement or other influences. The examination of this information, in summary, highlights the sensor's versatility to a greater range of altitudes while preserving a generally constant temperature; particular deviations imply its reactivity to subtle environmental changes.

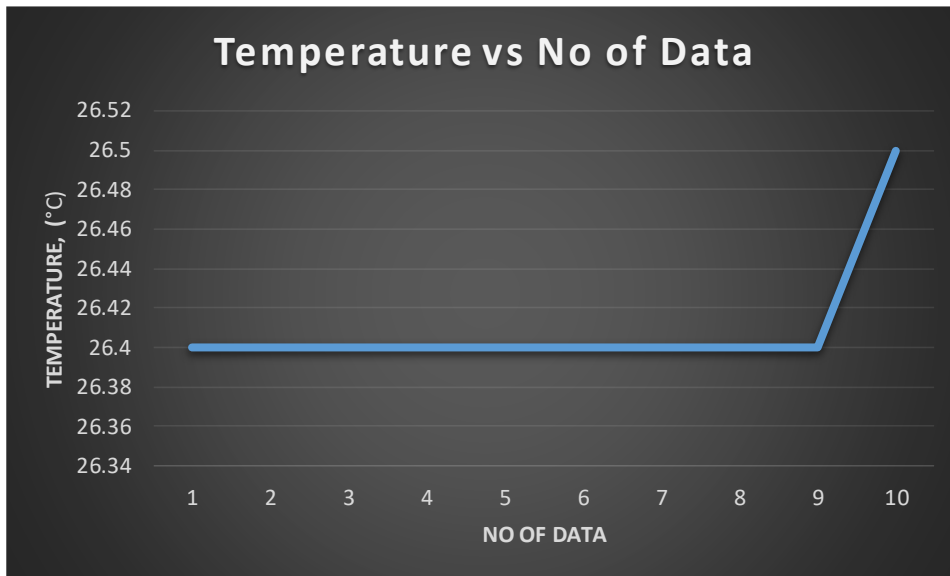


Figure 4.36 Test 2 Blynk IoT Temperature at moving condition

According to Figure 4.37, When the altitude values in the dataset are examined, they fall between 112.98 and 114.66, indicating a reasonably large range of elevations covered during the related test or observation. In contrast to the earlier datasets, this diversity in Altitude values suggests a wider range of climatic variables. On the other hand, the temperature values stay constant at 26.3°C for all ten data points. This temperature homogeneity, even at different elevations, highlights how dependable the sensor is in keeping a steady temperature in a variety of environmental conditions. The dataset highlights the sensor's adaptability in a variety of environmental circumstances by demonstrating its capacity to record and report variations in altitude while maintaining temperature stability.

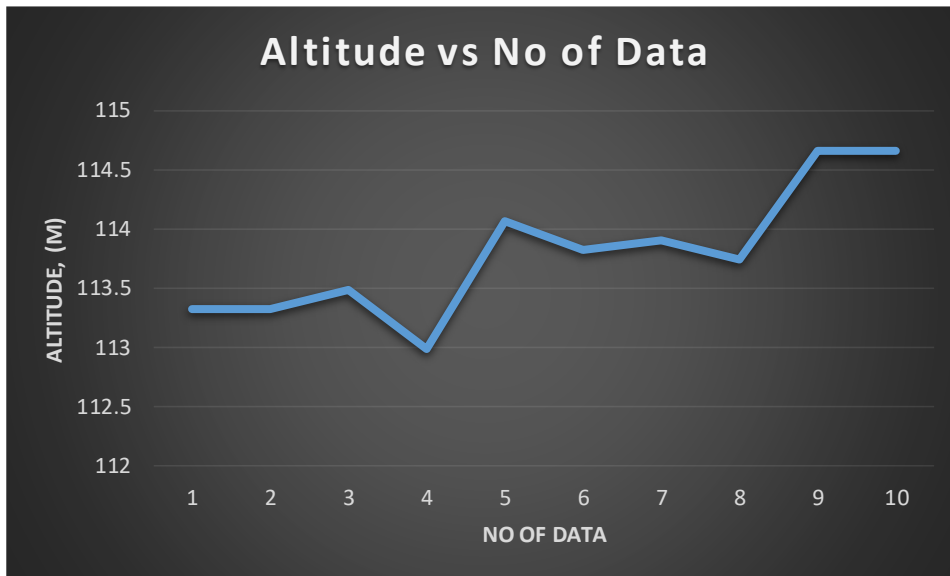


Figure 4.37 Test 3 Blynk IoT Altitude at moving condition

According to Figure 4.38, temperature readings, we find that they register at 26.3°C on average across the dataset. This temperature stability shows that, even in the face of altitude variations, the sensor is able to deliver accurate and constant temperature readings. This dataset's widely distributed elevations show how well the sensor captures variations in altitude while maintaining temperature stability. The temperature consistency in the dataset strengthens the trustworthiness of the sensor and confirms its applicability for applications that need accurate and consistent temperature monitoring over a wide range of altitudes. To sum up, the simultaneous examination of temperature and altitude in this dataset demonstrates how flexible the sensor is over a broad altitude range while retaining consistent temperature accuracy.

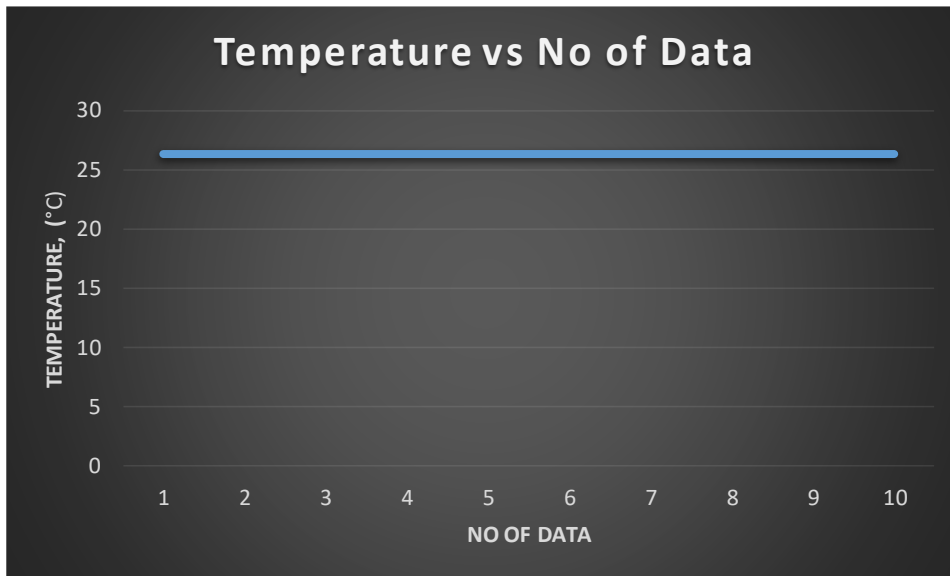


Figure 4.38 Test 3 Blynk IoT Temperature at moving condition

Based on Table 4.8, Test 1's readings, which vary from 2903 to 3703, show that the 10 data points' measurements varied. This range represents the tilt that was seen along the X-axis during the test. Similarly, the X-axis values fluctuate in Tests 2 and 3, indicating that the tilt angles will alter as the test goes on. There are two possible explanations for these fluctuations in the X-axis measurements: changes made purposefully throughout the experiment or ambient influences. Now let's look at the Y-axis data. The values in Test 1 indicate differences in the tilt along the Y-axis for the 10 data points, and they range from 2866 to 3013. Test 2 and Test 3 that follow likewise exhibit variations in the Y-axis values, which correspond to variations in the tilt angles unique to each test. When combined with the X-axis data, the Y-axis data offers a thorough comprehension of the tilt behaviour that was seen under experimental conditions.

To sum up, the data set includes tilt measurements for three separate experiments along the X and Y axes. The differences in the values that were collected for each data point indicate that there were dynamic shifts in the tilt angles throughout each test.

Table 4.8 Dataset for Tilt Angle with Blynk IoT at moving condition

No of Test	1		2		3	
Tilt Axis	X	Y	X	Y	X	Y
Data No	(°)	(°)	(°)	(°)	(°)	(°)
1	2903	2866	3004	2947	2960	2890
2	3134	2922	2998	3131	2957	2895
3	3408	2969	2992	3479	2960	2891
4	3438	2890	3001	3626	2960	2894
5	3703	2921	3015	3790	2971	2896
6	3712	2848	3047	3809	2944	2314
7	3827	2929	2995	3824	2955	2233
8	3885	2936	2997	3459	2950	2315
9	3979	2926	2992	2989	2934	2341
10	2913	3013	2935	2886	2934	2341

Based on Figure 4.39, The starting position is (2903, 2866), the first data point. Subsequent data points, such as (3134, 2922), (3408, 2969), and (3438, 2890), demonstrate a steady increase in the Y values. Data points such as (3703, 2921) and (3712, 2848) show that the Y values are rising and are clearly getting close to or beyond 3500. This pattern points to a steady tilt of the sensor to the right. Keeping with the dataset, we can see that there is a persistent rightward tilt at sites like (3827, 2929) and (3885, 2936) when the Y values exceed 3500. With the next data points (3979, 2926) and (2913, 3013), the pattern continues, supporting the interpretation of a right tilt. The starting position is represented by the first data point, (2903, 2866). Subsequent data points, such as (3134, 2922), (3408, 2969), and (3438, 2890), demonstrate a constant increase in the Y values. Data points such as (3703, 2921) and (3712, 2848) show how the Y values are clearly rising and getting closer to or beyond 3500. The sensor is consistently inclined to the right, as indicated by this pattern.

According to the dataset, we can see that there is a constant rightward tilt at points like (3827, 2929) and (3885, 2936) as the Y values get closer to 3500. The remaining data points (3979, 2926) and (2913, 3013) show that the trend continues, supporting the conclusion that there is a right tilt.

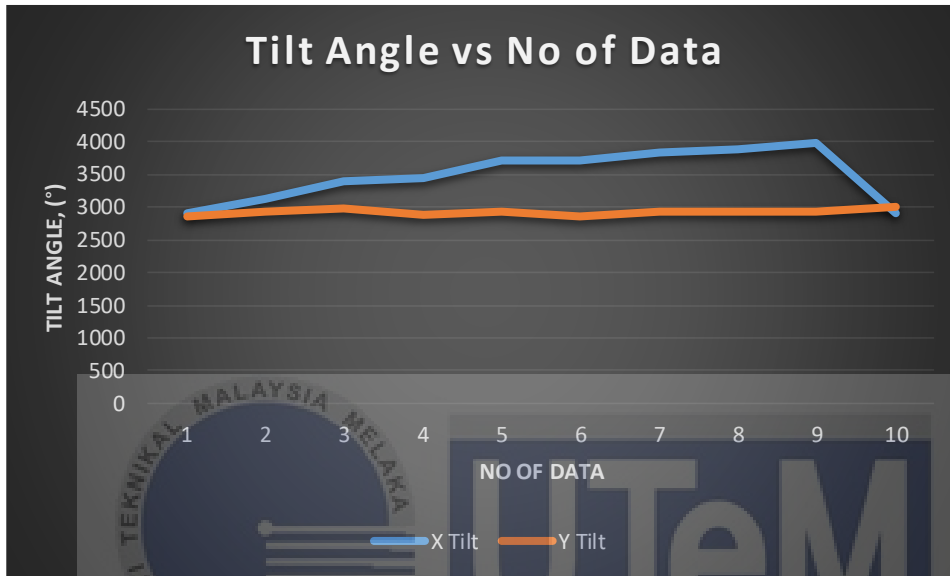


Figure 4.39 Test 1 Tilt Angle at driving condition for Blynk IoT

According to Figure 4.40, the initial position of the dataset is represented by the data point (3004, 2947). Following data points show a progressive increase in X values. The X value above the designated 3500 threshold when we get to (2992, 3479), which denotes an upward tilt in the sensor. This may indicate that the sensor is beginning to tilt too much. As one goes through the dataset, the sensor seems to maintain an upward tilt with X values like 3001, 3015, and 3047. Additionally, the Y values rise, indicating that the orientation of the sensor is inclined. The data point (2935, 2886), where the X value is less than 3500, marks the end of the sequence. This could point to the need for a possible fix or modification that could put the sensor back in a more neutral position. In summary, the dataset demonstrates a period of over-tilting upward, with X values continuously surpassing 3500. The growing

Y values point to a matching upward tilt in the orientation of the sensor. The last data point, where X is less than 3500, can indicate that the sensor has to be positioned differently or that a correction is needed.

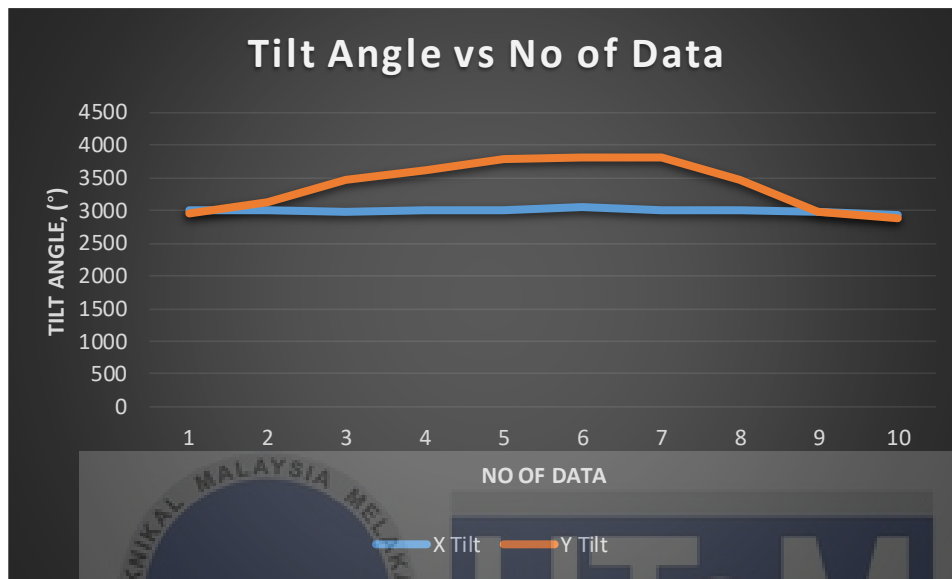


Figure 4.40 Test 2 Tilt Angle at driving condition for Blynk IoT

Based on Figure 4.41, the data point (2960, 2890), which represents an initial position, is where the dataset starts. The next set of data points shows a progressive decline in Y values. The Y value drops below the designated threshold of 2300 when we get to (2944, 2314), suggesting an excessive leftward tilt of the sensor. This shows that the sensor is starting to tilt significantly in the leftward direction. As one looks over the dataset, the sensor seems to maintain a leftward tilt with Y values like 2233, 2315, and 2341. Variations in the X readings also point to matching shifts in the orientation of the sensor. The data point (2934, 2341), where the Y value stays below 2300, marks the end of the sequence. This might point to a persistent leftward shift. In conclusion, the dataset shows a phase of excessive leftward tilt, with Y values always falling below 2300. The fluctuating X readings

imply matching shifts in the orientation of the sensor. The last data point validates a persistent leftward tilt, since the Y value stays below 2300.

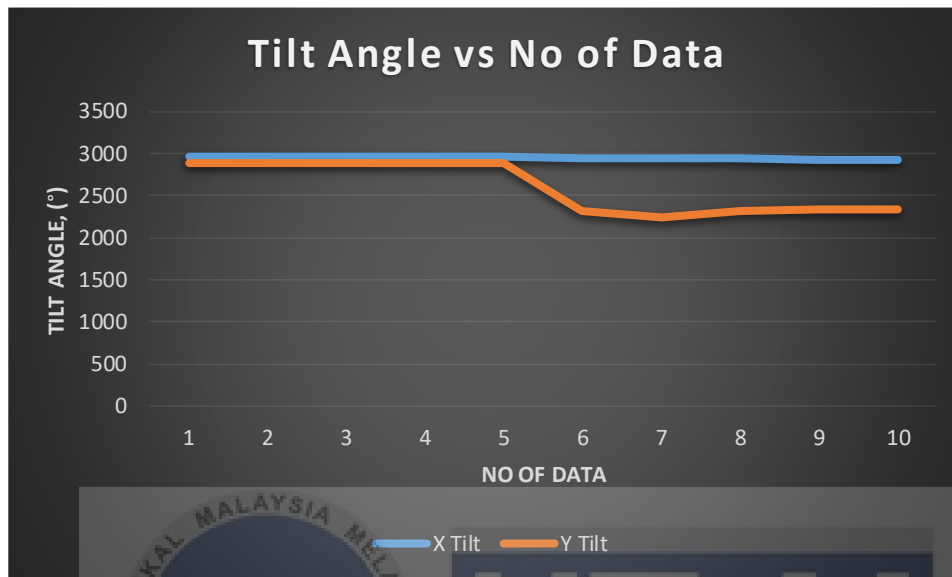


Figure 4.41 Test 3 Tilt Angle at driving condition for Blynk IoT

4.4 Summary

Based on the actual data, the initial reading in the tests is approximately 20–21 metres. Similar to the accelerometer's initial data, which read the realm of thousands, the sensors should undergo calibration also before taking measurements to make sure the initial reading starts at 0.

The GY-68 barometer sensor showed stability in stationary situations, keeping consistent temperature readings regardless of altitude changes. Temperature and average altitude readings fluctuated slightly but remained within reasonable bounds between testing. The graphs produced by the sensor showed reliable patterns of temperature and altitude during stationary states, demonstrating the accuracy with which it recorded changes in the atmosphere.

The GY-68 barometric sensor demonstrated accuracy throughout periods of inactivity by maintaining consistent altitude measurements with little changes when it was at rest. The temperature readings were comparatively constant, with only minor changes. These findings support the sensor's adaptability and dependability in a variety of settings, both stationary and at rest.

For the SN-ADXL335-CY accelerometer sensor, stationary data revealed a probable link between X and Y coordinates, with small fluctuations. Movement tests demonstrated stability, responsiveness, and correlation between the two axes, all of which were consistent with the anticipated traits listed in the reference material. These results highlight the accuracy and versatility of the sensor in detecting tilt motions.

Both sensors proved to be dependable and accurate in both stationary and moving situations, which qualifies them for applications needing reliable measurements under a variety of circumstances.

Most importantly, the paper contains a thorough examination of tilt data for both Blynk IoT and direct USB connections. The records clearly reveal patterns of tilting movements and convincingly illustrate the sensor's responsiveness to different tilt orientations.

Crucially, the investigation highlights an important discovery: the outcomes via Blynk IoT are equally dependable as those from a straight USB connection. This discovery confirms the efficacy of the wireless Blynk IoT strategy and highlights its equal dependability in providing precise and timely tilt data.

In summary, the project successfully addresses the shortcomings of RC airships by integrating cutting-edge sensor technologies with Blynk IoT. The thorough report demonstrates the sensor system's dependability and highlights the similarity in outcomes between Blynk IoT and a direct USB connection, establishing the integrated solution as a reliable and flexible option for real-time monitoring applications in a variety of contexts, including environmental studies, research, and surveillance.



CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, this project's successful completion has resolved a significant issue pertaining to remote control airships, namely the constraints impeding their ability to gather data and maintain situational awareness. The main goal of improving pilot awareness has been significantly met by the painstaking construction and integration of a custom sensor board and telemetry module for the RC airship's control system. The system's strong performance is demonstrated by the comprehensive testing and assessment carried out in a variety of settings, which indicates a noteworthy advance in the operational capabilities of RC airships in research roles. This creative approach shows off technological prowess and creates opportunities for further developments in the field of unmanned aerial vehicles. A paradigm shift in the operation of remote control airships is made possible by the smooth integration of sophisticated sensors and telemetry systems, which enable more accurate and efficient data collection in a variety of environmental circumstances. The project's successful results highlight the significance of utilising cutting-edge technologies to get over current restrictions in aerial research platforms.

5.2 Recommendations

For future improvements, it is recommended that opportunities for further system refinement be investigated for future initiatives. This include calibration of the barometric and accelerometer sensor for better reading values especially on the initial reading, optimisation of the integrated components, with a focus on potential advancements that

could elevate the system's performance, notably in the field of auto-stability. Furthermore, continuing research and development initiatives ought to be a top priority given how quickly the unmanned aerial system ecosystem is changing. Technology can be made more flexible and user-friendly by investigating automation features and intuitive interfaces, particularly those connected to the accelerometer for auto-stability. This project establishes a strong foundation for the incorporation of cutting-edge technologies into remote control airships, including the creative use of accelerometers for auto-stability systems. It also paves the way for future developments and breakthroughs in the fascinating field of aeronautics and cutting-edge sensor systems.



REFERENCES

- Álvarez, J.L., Mozo, J.D. and Durán, E. (2021) 'Analysis of single board architectures integrating sensors technologies†', *Sensors*, 21(18), pp. 1–28. Available at: <https://doi.org/10.3390/s21186303>.
- Barón, J.C.S. and Barón, M.J.S. (2014) 'Application of SHT71 sensor to measure humidity and temperature with a WSN', *2014 IEEE 9th IberoAmerican Congress on Sensors, IBERSENSOR 2014 - Conference Proceedings*, pp. 2–8. Available at: <https://doi.org/10.1109/IBERSENSOR.2014.6995554>.
- Biju, M. and Pant, R.S. (2017) 'Design and development of an indoor autonomous airship', *23rd AIAA Lighter-Than-Air Systems Technology Conference, 2017*, (June), pp. 1–14. Available at: <https://doi.org/10.2514/6.2017-3996>.
- Devine, J. *et al.* (2018) 'MakeCode and CODAL: Intuitive and efficient embedded systems programming for education', *ACM SIGPLAN Notices*, 53(6), pp. 19–30. Available at: <https://doi.org/10.1145/3211332.3211335>.
- Dhillon, S.S. and Chakrabarty, K. (2003) 'Sensor placement for effective coverage and surveillance in distributed sensor networks', *IEEE Wireless Communications and Networking Conference, WCNC*, 3(C), pp. 1609–1614. Available at: <https://doi.org/10.1109/WCNC.2003.1200627>.
- Dudak, J. *et al.* (2019) 'Serial Communication Protocol with Enhanced Properties- Securing Communication Layer for Smart Sensors Applications', *IEEE Sensors Journal*, 19(1), pp. 378–390. Available at: <https://doi.org/10.1109/JSEN.2018.2874898>.
- Elfes, A. *et al.* (1998) 'A semi-autonomous robotic airship for environmental monitoring missions', *Proceedings - IEEE International Conference on Robotics and Automation*, 4(May), pp. 3449–3455. Available at: <https://doi.org/10.1109/ROBOT.1998.680971>.
- Fitch, D., Southwick, J. and Morganti, J. (2002) 'Multi-sensor data processing for enhanced air and surface surveillance', *AIAA/IEEE Digital Avionics Systems Conference - Proceedings*, 1, pp. 1–7. Available at: <https://doi.org/10.1109/dasc.2002.1067952>.
- Gawale, A.C. *et al.* (2009) 'Design, fabrication and operation of remotely controlled airships in India', *18th AIAA Lighter-than-Air Systems Technology Conference*, (May), pp. 1–14. Available at: <https://doi.org/10.2514/6.2009-2855>.
- Grunde, U. (2013) 'Embedded configurable sensor interface devices for seamless data acquisition', *2013 21st Telecommunications Forum Telfor, TELFOR 2013 - Proceedings of*

Papers, pp. 526–528. Available at: <https://doi.org/10.1109/TELFOR.2013.6716282>.

Hua, H., Zhang, Z. and Feng, H. (2021) ‘Research Status and Method of Aviation Sensor Performance Evaluation’, *Journal of Physics: Conference Series*, 1769(1). Available at: <https://doi.org/10.1088/1742-6596/1769/1/012050>.

Kang, J. *et al.* (2012) ‘Modular approach in sensor board design’, *SenSys 2012 - Proceedings of the 10th ACM Conference on Embedded Networked Sensor Systems*, pp. 365–366. Available at: <https://doi.org/10.1145/2426656.2426711>.

Motiwala, N.I. *et al.* (2013) ‘Conceptual approach for design, fabrication and testing of indoor remotely controlled airship’, *Advanced Materials Research*, 690 693, pp. 3390–3395. Available at: <https://doi.org/10.4028/www.scientific.net/AMR.690-693.3390>.

Olejnik, A. (2020) ‘Integration and Investigation of Selected On-Board’.

Vilkov, J. V. *et al.* (2018) ‘The automated system of telemetry data multiline reception, processing and analyzing’, *IOP Conference Series: Materials Science and Engineering*, 450(2). Available at: <https://doi.org/10.1088/1757-899X/450/2/022032>.

Yu, Z., Hu, Y. and Huang, J. (2018) ‘GPS/INS/Odometer/DR Integrated Navigation System Aided with Vehicular Dynamic Characteristics for Autonomous Vehicle Application’, *IFAC-PapersOnLine*, 51(31), pp. 936–942. Available at: <https://doi.org/10.1016/j.ifacol.2018.10.060>.

APPENDICES

APPENDIX A List of distribution network parameters.

NO	Project Activities	Plan vs Actual Plan	October		November				December				January				February		
		Week	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
1	PSM BRIEFING	Plan								MID-TERM BREAK									
		Actual																	
2	Chapter 1: Introduction	Plan																	
		Actual																	
3	Chapter 2: Literature Review	Plan																	
		Actual																	
4	Chapter 3: Methodology	Plan																	
		Actual																	
5	Chapter 4: Result & Discussion	Plan																	
		Actual																	
6	Chapter 5: Conclusion	Plan																	
		Actual																	
7	Formatting and Grammar Improvement	Plan																	
		Actual																	
8	Slide Presentation	Plan																	
		Actual																	
9	Final Improvement	Plan																	
		Actual																	
10	Final Presentation	Plan																	
		Actual																	
11	Report Submission	Plan																	
		Actual																	

FAKULTI TEKNOLOGI DAN KEJURUTERAAN MEKANIKAL

Tel : +606 270 1184 | Faks : +606 270 1064

Rujukan Kami (Our Ref):
Rujukan Tuan (Your Ref):
Tarikh (Date): 31 Januari 2021

Ketua Perpustakaan
Perpustakaan Laman Hikmah
Universiti Teknikal Malaysia Melaka

Melalui

Dekan
Fakulti Teknologi dan Kejuruteraan Mekanikal
Universiti Teknikal Malaysia Melaka

Tuan

PENKELASAN TESIS SEBAGAI TERHAD BAGI TESIS PROJEK SARJANA MUDA

Dengan segala hormatnya merujuk kepada perkara di atas.

2. Dengan ini, dimaklumkan permohonan pengkelasan tesis yang dilampirkan sebagai TERHAD untuk tempoh **LIMA** tahun dari tarikh surat ini. Butiran lanjut laporan PSM tersebut adalah seperti berikut:

Nama pelajar: SHARIZAN BIN YUNUS (B092010134)


Tajuk Tesis: INTEGRATION OF SENSORS AND TELEMETRY SYSTEM FOR AN RC AIRSHIP USING CAD AND EMBEDDED SYSTEM

3. Hal ini adalah kerana IANYA MERUPAKAN PROJEK YANG DITAJA OLEH SYARIKAT LUAR DAN HASIL KAJIANNYA ADALAH SULIT.

Sekian, terima kasih.

“BERKHIDMAT UNTUK NEGARA”
“KOMPETENSI TERAS KEGEMILANGAN”

Saya yang menjalankan amanah,



TS. MUHAMMED NOOR BIN HASHIM
Penyelia Utama/ Pensyarah Kanan
Fakulti Teknologi dan Kejuruteraan Mekanikal
Universiti Teknikal Malaysia Melaka