UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name	:	MUHAMAD HILMI BIN MAT ASRI
Date of Birth	:	
Title	:	DESIGN AND DEVELOPMENT OF WIRELESS COMMUNICATION AND REAL-TIME WAYPOINT MONITORING SYSTEM FOR UNMANNED SURFACE VEHICLE
Academic Session	:	SEMESTER II 2021/2022

I declare that this thesis is classified as:

☐	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
☐	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☐	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

(Student's Signature)

New IC/Passport Number
Date:

(Supervisor's Signature)

PROF. MADYA IR. DR. ADDIE IRAWAN BIN HASHIM
Name of Supervisor
Date: 18/2/2022

NOTE: * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

SUPERVISOR'S DECLARATION

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 Electronics Engineering Technology (Computer System) with Honours.

____

(Supervisor's Signature)

Full Name : ASSOC. PROF. IR. DR. ADDIE IRAWAN BIN HASHIM

Position : ASSOCIATE LECTURER

Date : 18/2/2022

Universiti Malaysia PAHANG
Engineering • Technology • Creativity

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student's Signature)

Full Name : MUHAMAD HILMI BIN MAT ASRI

ID Number : TG18008

Date : 18/02/2022

DESIGN AND DEVELOPMENT OF WIRELESS COMMUNICATION AND REAL-TIME WAYPOINT MONITORING SYSTEM FOR UNMANNED SURFACE VEHICLE

MUHAMAD HILMI BIN MAT ASRI

Thesis submitted in fulfillment of the requirements
for the award of the Bachelor of
Electronics Engineering Technology (Computer System) with Honours

Faculty of Electrical & Electronics Engineering Technology
UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2022

ACKNOWLEDGEMENTS

First of all, I would like to express deepest gratitude to my supervisor, Assoc. Prof. Ir. Dr. Addie Irawan Bin Hashim, whom are very experienced for giving me the opportunity to express my engineering skill and guiding me in an interesting final year project. His supervision has aided to channel and apply my knowledge and idea and at the same time provided freedom and support to explore new concepts.

My appreciation also goes to the one who always see the potential in me and support my interest, my family. Their affection and kindness have bought motivation to always achieve the best that I can because there is always someone who encourage me to keep going.

Last but not least, special dedication to Faculty of Electrical & Electronics Engineering Technology (FTKEE) Universiti Malaysia Pahang (UMP) for giving me the opportunity to involve in a project that help me see better in the engineering world. Finally, I hope all the experience and knowledge gained through this project will gives me the utmost benefit to my future career. Thank you.

ABSTRAK

Kenderaan tanpa pemandu di permukaan air (SUV) ialah kenderaan yang dikendalikan atau dikawal tanpa ada sebarang pemandu pada kenderaan tersebut. SUV dapat beroperasi dengan pelbagai cara, antaranya adalah menggunakan alat kawalan jauh atau secara automasi dengan menggunakan pandu arah automatik. Kenderaan yang memiliki pemandu akan melibatkan kos yang tinggi terutamanya keperluan pemandu itu sendiri dan memiliki keupayaan yang terhad dimana ketika melakukan pekerjaan dikawasan yang merbahaya yang dapat mendatangkan risiko kepada pemandu tersebut. Dengan menggunakan alat kawalan jauh untuk mengawal kenderaan tersebut, ia dapat melakukan pelbagai kerja-kerja dengan baik dan lebih berpatutan pada kos. SUV adalah ciri pilihan lain untuk kenderaan di permukaan seperti bot dan kapal dimana beroperasi secara pengawalan jarak jauh atau bersama pandu arah automatik. Walaubagaimanapun, kenderaan tanpa pemanda di pasaran pada masa ini kebanyakannya berharga mahal atas sebab jenama atau belum lagi meluas digunakan sehingga mencapai tahap penggunaan keperluan. Oleh sebab itu, projek ini mengambil langkah untuk mencipta sebuah SUV pada harga yang berpatutan yang memiliki reka bentuk yang mudah menggunakan konsep katamaran. Rangka kapal dicipta daripada plastik Polivinil Kloridadan digunakan bersama polistirena. Selain itu, projek ini direka bentuk dengan alat kawalan jauh tanpa wayar dengan transmisi frekuensi radio, memiliki paparan masa sekarang pada kenderaan yang menunjukkan permukaan air, dan unit Sistem Kedudukan Global (GPS) untuk mengenalpasti lokasi koordinat kenderaan bersama Sistem Global untuk Komunikasi Mudah Alih (GSM) untuk memuat naik maklumat data dalam perisian antara muka grafik (GUI). Sistem kawalan kenderaan untuk menggerakkan SUV dibangunkan dengan sistem alat kawalan jauh untuk memandu mengemudi kenderaan. Hasil jangkaan projek itu adalah untuk membangunkan prototaip mini SUV dengan kawalan jarak jauh tanpa wayar dan memiliki system pemantauan yang digunakan di kawasan air yang tenang.

ABSTRACT

Surface unmanned vehicle (SUV) is a vehicle that is operated without any crew onboard. Operate in various ways such as remotely by using remote control or autonomously by using autonomous navigation. Surveying using manned vehicle may cost a lot especially in terms of requirement of passenger or crew and limited for risky task that may have potential to jeopardize the passenger or crew such hazardous or confined area. A remote operated unmanned vehicle is able to carry out the task more economic and efficient with unmanned features. The SUV is an alternative feature for surface vehicle such as boat and ship in which remotely operated and some with autonomous configuration. However, the current existed commercial SUV majority with the high prices either because of the brand or the size of market that still not reach the essential product level. Therefore, this project has taken initiative to develop an optimal cost of SUV boat with simple design using catamaran concept. Two hulls are created from Polyvinyl Chloride (PVC) Plastic part and the body made by polystyrene. Also, this project is designed with a wireless remote control with radio frequency (RF) transmission, has real-time viewing on surface water vehicle and Global Positioning System (GPS) unit to configure the coordinate location of the SUV with Global System for Mobile Communication/General Packet Radio Service (GSM/GPRS) technology to upload the data information in customized graphical user interface (GUI) software. The controller unit for the propulsion of the SUV is developed with remotely operated and this system is also covered on driving the rudder of the vehicle for pitching its motion. The expected outcome of the project is to develop a mini-SUV prototype with long haul wireless control and monitoring for calm and low tidal water area.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

d	Distance travelled
t	Time travelled
S	Speed

LIST OF ABBREVIATIONS

PVC	Polyvinyl Chloride
RF	Radio Frequency
GPS	Global Positioning System
GSM/GPRS	Global System for Mobile Communication/General Packet Radio Service
GUI	Graphical User Interface
ASV	Autonomous Surface Vehicles
ROV	Remotely Operated Vehicles
ITS	Intelligent Transportation Systems
API	Application Programming Interface
TX	Transmit
RX	Receive
IDE	Integrated Development Environment
USB	Universal Serial Bus

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

The SUV is commonly defined as water transport vessels that has the ability of operating on the surface of water without onboard human operator with either automated navigation or remotely controlled, such as Autonomous Surface Vehicles (ASV) or Remotely Operated Vehicles (ROV) [1]. ROV is using RF communication technology to having wireless communication that connects the controller with the SUV, the communication are build this way to interchange data or receiving control commands [1].

The application of autonomous SUV has been used in various application such as in offshore applications for Oil and Gas Industry. While, ROV has been used in other applications, estuarine and fluvial environment, military development, and detecting contaminants [1]. The ROV commonly used during the collection of real-time data which using ROV contribute to greater data density on both a temporal and spatial scale [2]. The collection of a vast amount of data is vital to predict the future trends of environment indicators that giving the idea for preservation in a specific area. The SUVs are also perform in dangerous task, such as in military and civil application [1].

The advantages of the SUV over manned vehicles are SUV have the potential to reduce risk for manned forces that could expose to hazardous environment, such as turbulence sea, heavy sea conditions, or nuclear or waste contaminated environment. The SUV also not confined due to limitation imposed by human crew, such as temperature, space, or movements [1].

Nevertheless, the aim of this paper is to propose a mini surface water vessel using catamaran approach with wireless communication for long haul remotely control and prepared with real time virtual monitoring panel for SUV waypoint coordination.

1.2 Problem Statement

The SUV is becoming more widely used due to the ability throughout the outcome to complete a task. Equipment cost and frequent prevention maintenances are considered as the main factors in SUV platforms in terms of demand. This study presents a design of mini surface water vessel using catamaran approach which is low cost design because of its simplicity but has the robust design in terms of stability.

Besides that, the process of collecting real-time data through remotely operated could achieve more both on spatial and temporal scale compared to autonomous operated. Thus, establish a wireless communication for long haul remotely control system in SUV would benefit in term of diversity working area and application.

Moreover, movement waypoint coordination is helping on observing the vehicle spot location especially during surveyor. Having access of data movement waypoint would organize every movement of the SUV in terms of unwanted repetition of surveying. With using popular technology of navigation system combines with microcontroller, inexpensive real-time monitoring panel for SUV waypoint coordination could be developed.

1.3 Objectives of Project

The aim of this project is to develop and design Unmanned Surface Vehicle boat that able to monitor surrounding and surveyor. The main issue of project is to focus on several solution as follow:

i. To design and establishing wireless communication for long haul monitoring for the SUV unit.

ii. To design a real time virtual monitoring panel for SUV waypoint coordination.

1.4 Research Scope

The scope of this project is focusing on developing and establishing a wireless communication of real-time viewing monitoring system for SUV unit. The real-time monitoring system is designed as GUI that shows coordinate waypoint of the SUV and having real-time view of surface water.

1.5 Thesis Roadmap

The proposal highlights the discussion and progress of development real-time viewing monitoring system for SUV unit.

Chapter 2: This chapter will be discussing on literature review. The literature review will be emphasized on the project previous research presented by other people in same area with relevant issue to SUV boat and monitoring system. There are about overview of the SUV and monitoring design and technology in SUV. Lastly, there is a summary of the literature review regarding to the project.

Chapter 3: This chapter will be highlighted on monitoring system structure. The monitoring system of SUV work and the detail about the component used to design the monitoring system for the SUV.

Chapter 4: This chapter will be focusing on the result for the development of Surface Unmanned Vehicle. The discussion on the testing hardware of the surface unmanned vehicle will also be explained in this chapter.

Chapter 5: This chapter will conclude the overall project. The future recommendation will be discussed in this chapter.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Literature review is a research method of study scholarly sources to provides knowledge, identify relevant theories, and information of existing research. In this literature review will be emphasizes in Development of the SUV. There are about the design and technology in SUV. Lastly, there is a summary of the literature review regarding to this project.

2.2 Overview of SUV Design and Control Technology

The development of the SUV that able to control a boat without onboard human operator, instead of using remote control operated with real-time data collection capability and have environmental monitoring. This project is proposed by Department of Marine Technology, UTM Johor Bharu [3]. The SUV is capable to perform tasks in variation of environments without jeopardizing human life and operated in more economic and efficient than manned vehicle [3].

The USV boat in this project is designed based on modified jet-ski hull structure as in Figure 2.1 and equipped with other instruments for such as Global Positioning System, Inertia Movement Unit and etc. [3]. The design can improve roll stability of the boat, have greater payload, and redundancy in hull rotation by applying catamaran design on jet-ski hull structure [3]. The catamaran design usually using two hulls but by applying the design to jet-ski is able to make only a hull on catamaran design and have the abilities of the design [3].

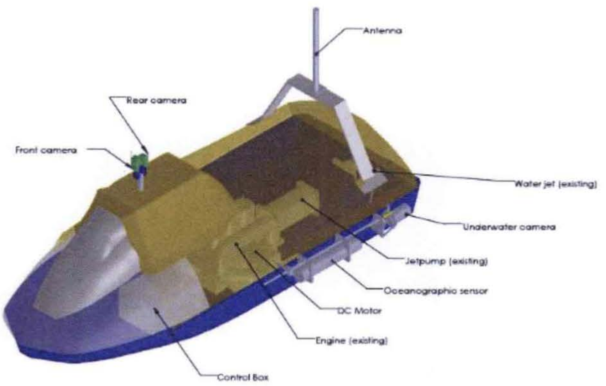


Figure 2.1: The jet-ski hull structure SUV [3].

The project is proposed for designing and control an Automated Surface Vehicle (ASV) which is used for able to do any missions that are not constrained due to restriction imposed by human or jeopardize to human life [4]. The objective of the project is to produce a low-cost and easily transportable ASV that can greatly reduce the involved with continuous supervision [4]. This project also made a custom-made winch as in Figure 2.3 is to lower down the payload into water [4]. This project is proposed by School of Engineering, Deakin University.

Some key features were highlighted to make inexpensive and robust design which is reduce manufacturing cost and use off the shelf equipment which is extruded aluminium for easy and rapid assembly, design a custom made winch which is capable of carrying a payload of 20 kilograms, include the navigating system on ASV for navigating feature, and the system can be operated remotely and autonomously [4].

The ASV design is used catamaran configuration created with PVC pipes is shown at Figure 2.2 [4]. Custom design of winch as in Figure 2.3 at the centre of the two hulls which is used to lower down the payload into the water up to 30 meters and also convey appreciable amount of payload [4]. The two hulls need to consider because will impact the weight of the aluminium frame, winch, payload and electronics part of the system can be used. The buoyancy calculation and the weight distribution are used to get the parameter and payload of the boat [4]. The frame of the ASV is made of extruded aluminium rods to make the design easily transportable which can rapidly assembly and reduced cost in fabrication process [4].

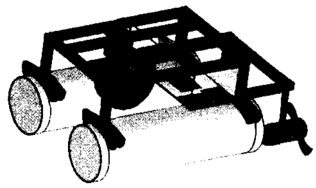


Figure 2.2: Catamaran Design [4].

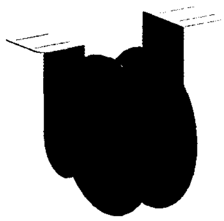


Figure 2.3: The design of custom winch [4].

The remote control is for controlling the propulsion of the SUV. Remote control is most usual ways to control any robot, because the remote control has the effective range to controlling and not strictly limited to line of sight operation [5]. By implement microcontroller to Radio Control, Radio Control can have better interface to diversified the function of the robot [5]. A radio control usually consists of a transmitter that is used to capture inputs and send the input to the receiver through signal wave. The receiver will capture the signal wave and converted the signal to electrical signal. The Figure 2.4 shown the example of remote control taken from the book written by John David Warren, there is a topic in the book that is discussed about how to control robot using remote control.

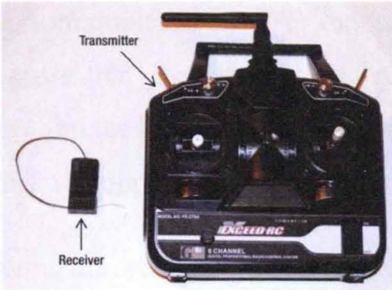


Figure 2.4: 2.4 GHz transmitter and receiver [5].

The Figure 2.5 shows the remotely control robot proposed by J-D Warren, Josh Adams, and Harald Molle, is a remote-controlled robot which design to capture video while explorer. The propulsion of the robot is control by remote control system to make the robot move to the location of the operator instructed. The first-person view camera is for controlling the robot without the operator on the robot location.

Figure 2.5: The Explorer Boat [5].

The autopilot system is for controlling the system by giving the location required and then, the boat will operate automatically. The autopilot based on Arduino needs Ardupilot PCB and a GPS module for the controller. The Ardupilot PCB is the heart of the system to make the system autopilot [5]. The Ardupilot PCB has all the elements on board to control a vehicle by GPS. The same microprocessor is used in Arduino and the Ardupilot PCB which is Atmega328 [5].

The ArduPilot board communicate to GPS module are using UART (serial) connection [5]. A GPS is a sensor that gives locations to Ardupilot. The GPS measures the actual position, the velocity and the heading of the boat. Input from the GPS will send the data to the Ardupilot to make Ardupilot moves the location given.

The Figure 2.6 is showing the architecture of Ardupilot RoboBoat. The RoboBoat is an Autopilot boat proposed by J-D Warren, Josh Adams, and Harald Molle is an automatic controlled boat. The movement control is using Mission Planner software system to have coordinate from GPS where the boat to heading continuously without supervision [5].

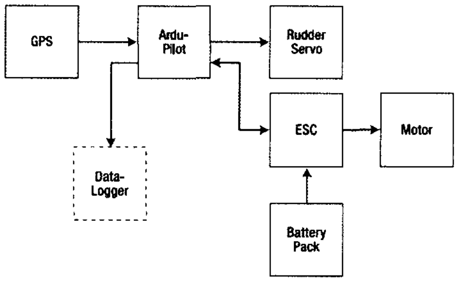


Figure 2.6: Architecture of Ardupilot RoboBoat [5].

The Figure 2.7 shows that the SUV prototype by Satetha Siyang and Teerakiat Kerdcharoen, Bangkok in 2016 which it is controlled by Ardupilot APM 2.6 board from 3D robotics company which was designed to covers all sufficient sensors to build an autonomous mobile robot. According to the prototype, an external ground positioning system, GPS was used for the navigation. The ground control station consists of laptop computer, a mission planner program and three communication modules linking with the USV which the first is telemetry module to connect with Ardupilot to obtain sensor data related to mobility of the USV, the second is XBee module to collect sensor data related to water properties and the third is radio transmitter module for manual operation to control the USV [6].

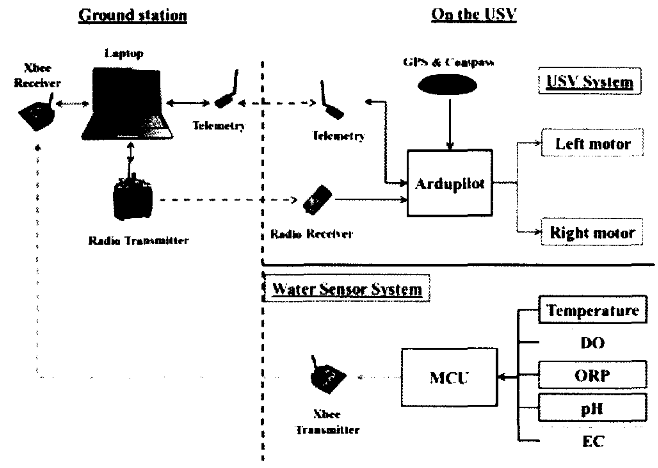


Figure 2.7: Example of control design of Surface Unmanned Vehicle [6].

The mission planner program is a free open source program developed by Michael Oborne and was used to control the USV in the research. Telemetry module was used to connect a mission planner program with APM 2.6 Ardupilot controller board. It is a small, light and inexpensive open source radio platform typically allowing ranges much longer than 300 m. Thus, the ranges can often be extended to several kilometres with the use of appropriate patch antenna. The telemetry is a 915 MHz version and use USB Serial Port to connect with a computer [6].

2.3 Monitoring Design and Technology

Vehicle tracking system is widely used especially in vehicle position tracking system, vehicle anti-theft tracking system, fleet management system, and Intelligent Transportation Systems (ITS) to track and display vehicle location in real time [7]. This project is proposed by Department of Electrical and Computer Engineering, Kettering University, USA. The vehicle tracking system capable to display the vehicle location continuously on the map using Google API through Smartphone Application [7].

The tracking system in this project is designed using microcontroller, GPS module and GSM/GPRS module for the hardware. C programming language is used as a software program to control or as instruction that embedded in the microcontroller. The GPS is used to provide information of location coordinates, speed, and time. The

GSM/GPRS module is used to established connections between in-vehicle device and a remote server for transmitting the vehicle's location information using TCP/IP connection the GSM/GPRS network [7].

The Figure 2.8 shows the tracking system layout [7]. The GPS module is received the data location from the satellites and is read by microcontroller. Then, the data location information will send to the web server through GSM/GPRS module. The web server is connected to database and Google Maps API. The data location will be stored in the database and could be monitored using Smartphone Application.

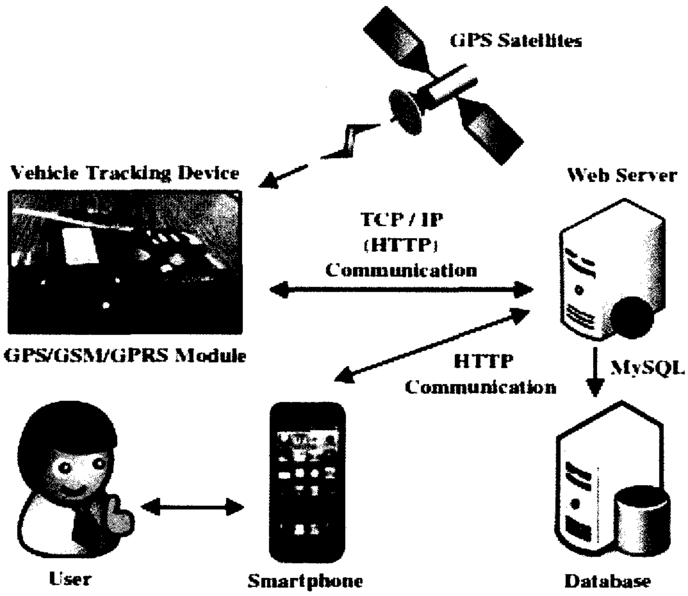


Figure 2.8: Vehicle System layout [7].

The project is proposed to developed real-time vehicle tracking system using Web-based technologies [8]. The objective of the project is to developed a continuous monitor a moving vehicle tracking system. This project is proposed by Department of Computer Engineering and Information Technology, Yangon Technological University Myanmar.

The tracking system is designed to tracking and positioning data location of a vehicle by using GPS, GSM/GPRS and database technologies. The GPS is provided the data location of the vehicle and GSM is provided the HTTP connection to server [8].

With using these two modules the data location information of vehicle is available to the system over the internet on the map. The Figure 2.9 is shown the block diagram of Real Time vehicle tracking System using Web-Based Technologies.

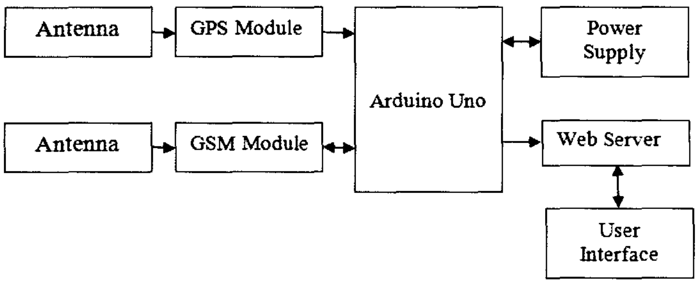


Figure 2.9: Block Diagram of Real Time vehicle tracking System using Web-Based Technologies [8].

The Global Positioning Satellite (GPS) system is for coordinate the position of the sensor by acquire latitude and longitude coordinate on a map [5]. GPS systems are use the signals from several tracking satellites in space to calculate the position of the GPS, and then create outputs of specific set of latitude and longitude coordinates. Other than position coordination, the GPS also provides time and speed detection. The output from GPS is a National Marine Electronics Association (NMEA) standard string or ASCII sentences that needs to be decoded to obtain valuable information such as latitude, longitude, speed and direction [5]. The Figure 2.10 shown the example of GPS sensor taken from the book written by John David Warren, there is a topic in the book that is discussed about what GPS system do and provide.

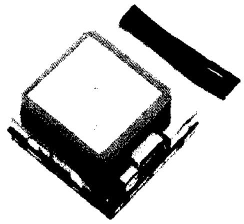


Figure 2.10: The EM406 GPS sensor [5].

The Global System for Mobile Communications (GSM) is a technology to transmit and update the information. By implement GSM to microcontroller, the device able to give the information in real time. A GSM mobile will require a Subscriber Identity Module (SIM) or known as SIM card to be operated and operates over a network range subscribed by a network operator. The GSM mobile will be receiving digit command by SMS from another cell phones and send the requested data to the cell phones through serial communication. GSM also works through GSM network, which using AT command to create communication between connected server with GM module [7]. The Figure 2.11 shown the GSM module.

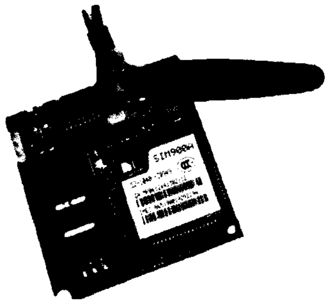


Figure 2.11: GSM module.

2.4 Summary

From the research and study, the development of the SUV is considered to build using catamaran approach with remotely control.

For monitoring system and technology in this project is choosing to implement tracking system technology on the SUV which many tracking systems uses in literature studies above. The tracking system has brought a lot of benefits in industry such as management system, and intelligent transportation systems. The monitoring system for waypoint coordination is used GPS technology and GSM/GPRS technology, which data information that obtained from the GPS will sent to the web server through GSM. The GUI that is used in this project will do improvement on the adding some information on the GUI such as speed and able to see time at the coordinate location.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this project the SUV boat is designed based on catamaran approach. The SUV boat is developed with remotely operated controlled mode. The SUV is prepared with First Person View (FPV) camera for viewing on the surface of water and GUI that able to trace the waypoint and real-time coordination of the unit. Generally, the whole project system consists of three scope of development as shown in Figure 3.1.

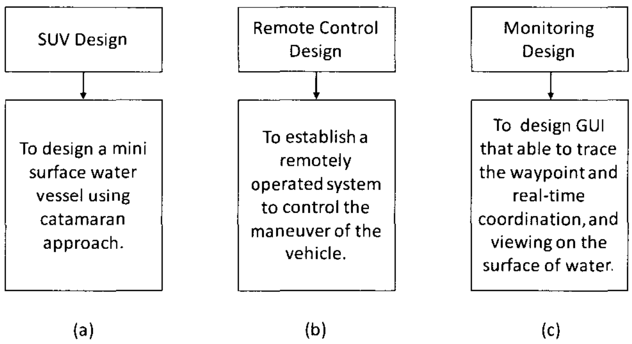


Figure 3.1: The scope project of the development of SUV for water monitoring and surveyor. (a) SUV design scope. (b) Remote Control Design Scope. (c) Monitoring System Design scope.

The scope of this paper is focusing on monitoring system design scope. In this chapter will be explaining on the monitoring system design and fabrication. This chapter also will explain how the monitoring system of SUV work and the detail about the component used to design the monitoring system for the SUV.

As shown in Figure 3.2, the developed monitoring system consists of microcontroller, GPS, GSM, and web server. The data input from the GPS will sends to the microcontroller.

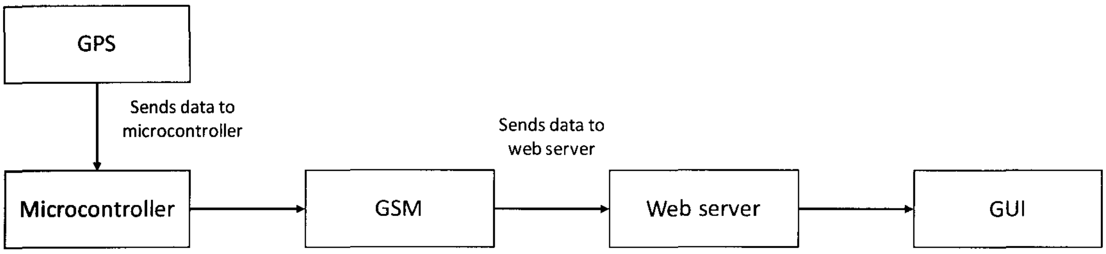


Figure 3.2: The block diagram of monitoring system.

The monitoring waypoint system is using microcontroller to control the input and output instruction of GPS and GSM modules. The GPS is used to obtain the data coordinate at periodic time intervals. The GSM is used to transmit the data to the web server. Thingsboard.io is used as GUI to display the monitoring panel such as map and waypoint of the SUV movement.

For the FPV camera of the SUV, the camera is using RF transmission which the camera as transmitter and using the receiver that is connected to the smartphone as On-Screen Display. Figure 3.3 shows the block diagram of surface water viewing of monitoring system.

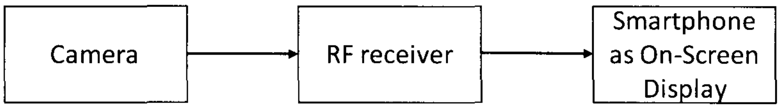


Figure 3.3: The block diagram of surface water viewing monitoring system.

3.2 SUV for monitoring and surveyor System Design

3.2.1 Flowchart

The monitoring system flowchart is shown in Figure 3.4. As in Figure 3.4, monitoring system works when the system is On. As soon as, the system On, the FPV camera will display the front view the SUV and the GPS will starts looking the data location of the SUV. After the data has been captured, the data is sent to the web server through GSM module. The data location will be displayed on the GUI.

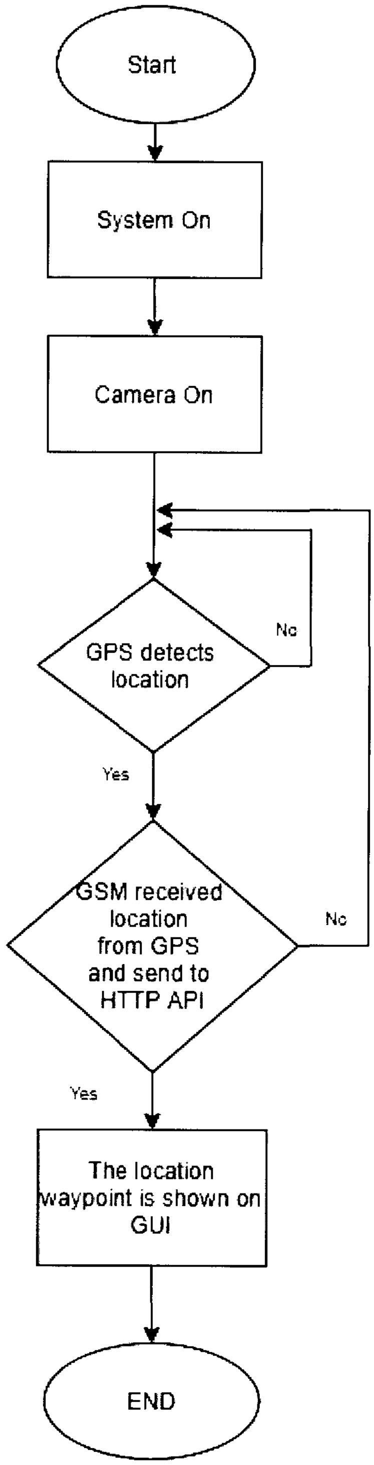


Figure 3.4: Monitoring process flowchart.

3.3 GUI of Monitoring System

The information feedback of the system is using Internet of Things (IoT) platforms for uploading data from sensor to the GUI. The GUI can be accessed by both Personal Computer (PC) or smartphones by using Thingboard.io webpage. Thingsboard is an open source server-side platform that allows to monitor and control IoT devices. Figure 3.5 shows the GUI design of monitoring system.

In the GUI, the information of data location waypoint, current location of the SUV, the speed of the SUV, and coordinate of latitude and longitude with time is shown.

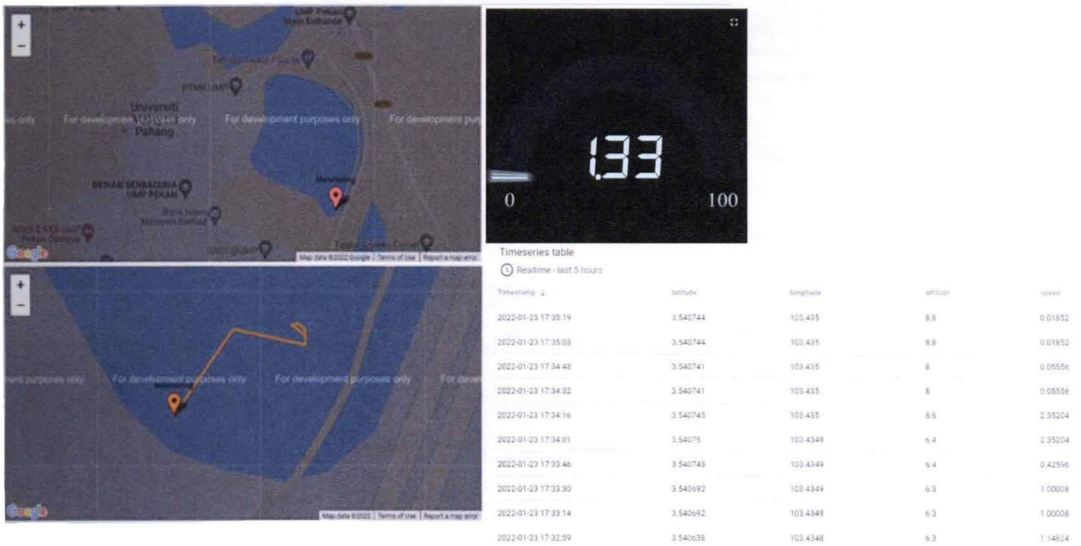


Figure 3.5: Monitoring System GUI designed for SUV monitoring.

3.4 Monitoring System Unit

The monitoring system unit used in the SUV is made up with microcontroller, SIM900A module and NEO M8N GPS module. The connection of the module with the microcontroller is shown in Figure 3.6. The microcontroller is used to control the function of a device by interpreting data that been receives from central processor. The coordinate location of the SUV is captured through GPS module and the data is transmitted to web server by using GSM/GPRS technology.

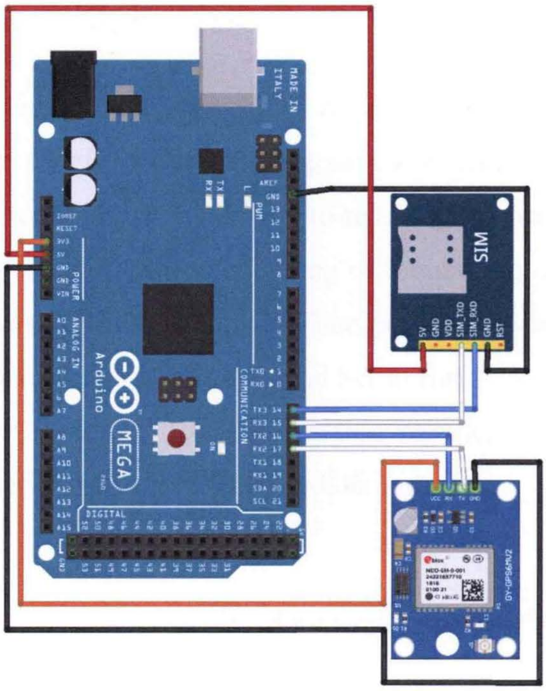


Figure 3.6: Monitoring System Unit Circuit.

GPS module and GSM module are connected to the microcontroller. The GPS module is using Transmit (TX) and Receive (RX) pin to transfer and receive the data location and that pin is connected to the TX and RX pin of the microcontroller. The TX pin of the GPS module is connected to RX pin of microcontroller and the RX pin of the GPS module is connected to the TX of microcontroller. The main reason for this connection is because one device is transmitted the data out and another device is received the data information. Therefore, the TX pin of GPS is sent data to the RX microcontroller and the RX pin of GPS receives data from the microcontroller TX pin. The GSM module connection is connected with the same way as GPS module with different port. Thus, the TX pin of GSM/GPRS is sent data to the RX microcontroller and the RX pin of GPS receives data from the microcontroller TX pin.

The monitoring system unit is collected the data location information through GPS, and formats the data information into a system-specific packet format which is JSON format and sends the data to the server through GPRS. When the GPRS connection was established, the connection between service provider server is created using HTTP protocol and the data information at microcontroller is able to sends to the server.

3.5 Embedded Monitoring Program System

The software used in this project is Arduino IDE (Integrated Development Environment) software which function to program the microcontroller and monitor the hardware. This computer software is allowed to receive data from microcontroller and display the data on the screen during debugging or monitoring, and able to sends data command from computer to microcontroller. The process of data exchanged between computer and microcontroller is using Universal Serial Bus (USB) cable. This computer software is perfect for real-time data transmission monitoring during programming. Figure 3.7 shows the serial monitor in Arduino IDE software.

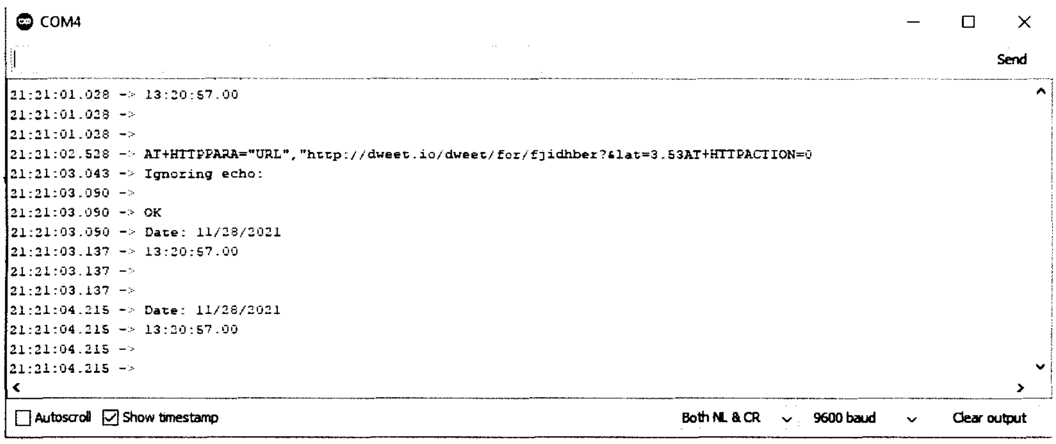


Figure 3.7: Sample of serial monitor prompt in Arduino IDE software.

This software is using C programming language. The program for monitoring system consists of set up from GPS to obtained data, and set up the GSM/GPRS module to upload the data information to the Thingsboard server. The Figure 3.8 shown the working process of the programming code for monitoring system.

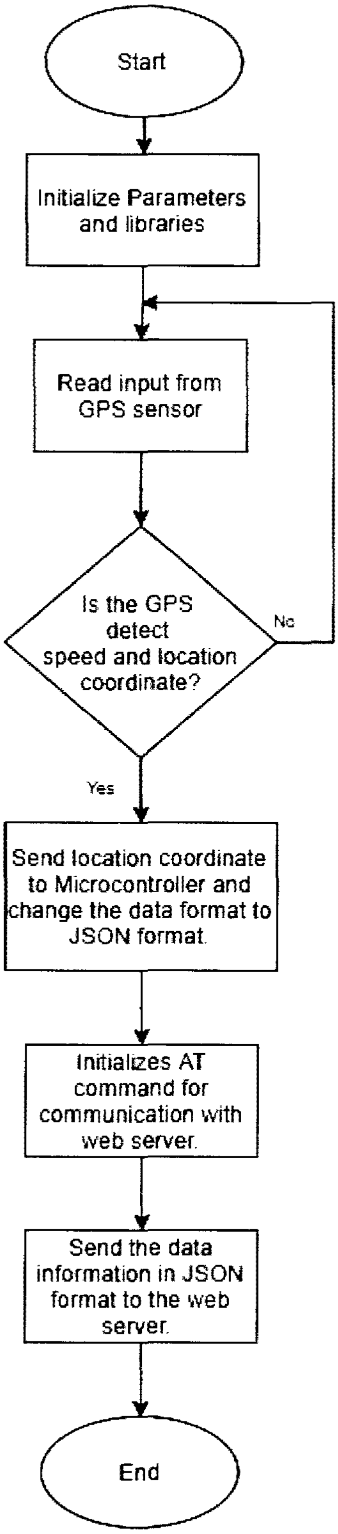


Figure 3.8: Embedded monitoring system flowchart.

3.5.1 Embedded Program For GPS

The GPS system is responsible to determine the data location of the SUV. The microcontroller is used as a medium to make the GPS system operate the way it was instructed by programming. TinyGPS++ library is used to encode National Marine Electronics Association (NMEA) sentences from GPS. The Figure 3.9 shows the embedded program code flowchart process of GPS module for this project.

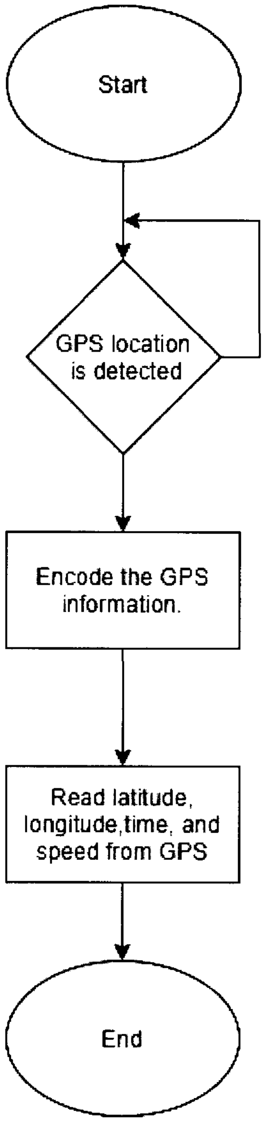


Figure 3.9: GPS embedded program flowchart.

After that, the data information from the GPS module is used in the web server. The web server requires system-specific format to read data information content. Thingsboard.io is used JSON format for the key-value content. To make that possible the data information needs to assign into a document. The DynamicJsonDocument is used to allocates memory of the data information document, and SerializeJson and serializeJsonPretty is used for minified the document for sending to web server.

3.5.2 Embedded program for GSM

In this system, the information received from GPS module is sent to the server through GSM network. Attention (AT) command is used in programming for communication between GSM module with web server. AT commands are instruction to control a modem.

The AT commands that are used in this program code is AT+SAPBR, AT+HTTPPARA, AT+HTTPINIT, AT+HTTPDATA, and AT+HTTPACTION. AT+SAPBR is used for setting configuration of GPRS. First, the SAPBR is used to set the Access Point Name (APN) of the SIM card provider. AT+SAPBR = 1, is to open bearer, which the data service is allows the transmission between information signal and network interfaces. AT+HTTPINIT is used to initializes the HTTP to be used. To set the HTTP parameter is used AT+HTTPPARA command. The parameters that was set is URL. For AT+HTTPDATA is used to input data to the web server. AT+HTTPACTION is used to set the method action that use in this project. AT+HTTPACTION = 1, which means POST request method is used. The embedded program for this project is attached on the Appendix.

The Figure 3.10 shows the embedded program code flowchart process of GSM/GPRS module for this project

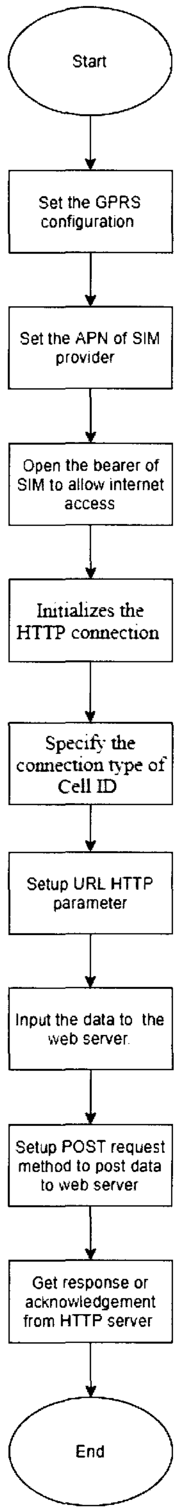


Figure 3.10: GSM embedded program flowchart.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter will be focusing the real-time result obtained during the project period. The project of implementing real time virtual panel for SUV waypoint coordination is completed. This project is to verify and analyse the effectiveness of implementing SUV monitoring system at UMP lake, the waypoint coordination is recorded using GPS and is uploaded to the server using GSM module. The result will be highlighted in this chapter.

4.2 Waypoint Monitoring

The SUV coordinate location is being collected by GPS and sent to web server through GSM and being plotted in the GUI. The latitude and longitude point coordination are sent to the web server each 15 seconds and the waypoint is generated from the uploaded data. The location for testing the movement of SUV is at UMP lake B as shown in Figure 4.1.

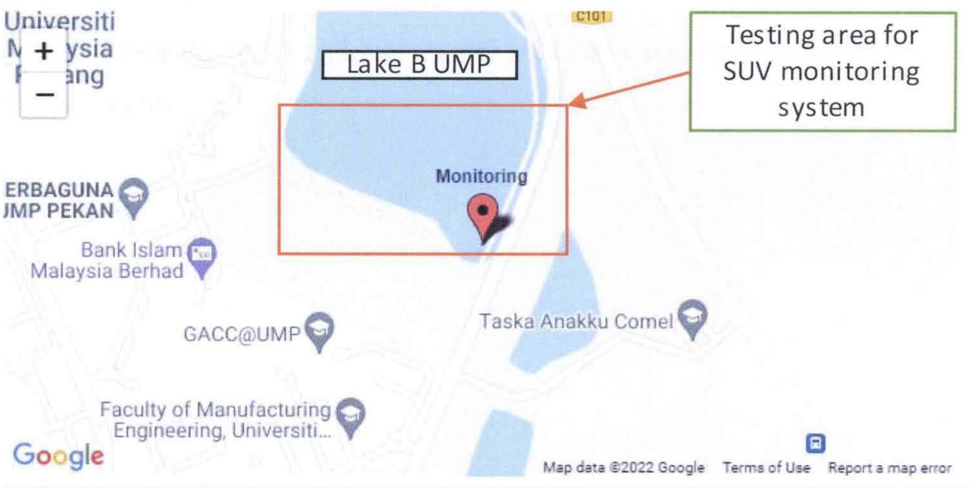


Figure 4.1: The block diagram of monitoring system.

4.3 Preliminary Test

The preliminary test has been taken on land and on the surface of water. The test on land is taken due to check either the GPS unit is able to operated when there is magnet at the surrounding that possibly block the GPS signal. Figure 4.2 shows the testing area on land which the testing area is at the car parking space.

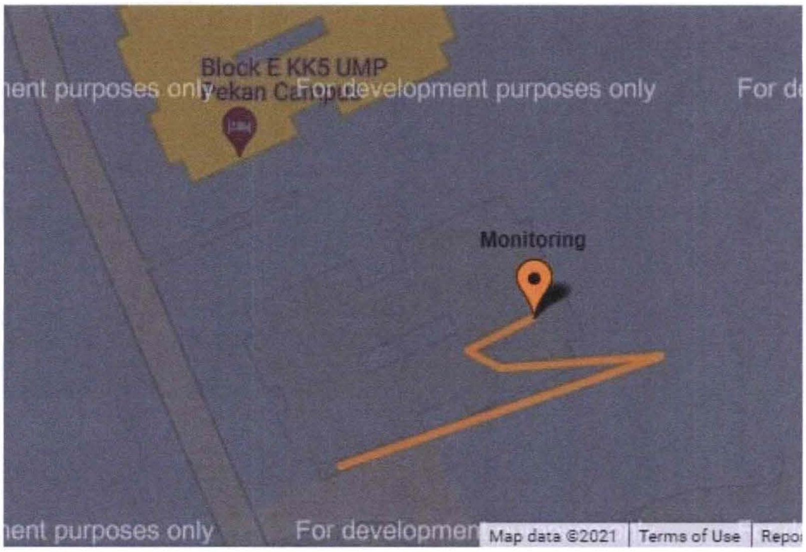


Figure 4.2: The preliminary test of monitoring system at car parking area.

During on the surface of water, the output problem that has been determined is the GSM have not enough power to operated. The battery is able to provide enough power to the servo, microcontroller, GPS and GSM during no load operated but not when considered with strong wind, and wave water area that require high current consumption for the servo motor.

4.4 Field Test and Results

Figure 4.3 below shows the real-time result of virtual panel GUI of waypoint coordination during project period. The movement of path drawing are recorded using Thingsboard.io GUI.

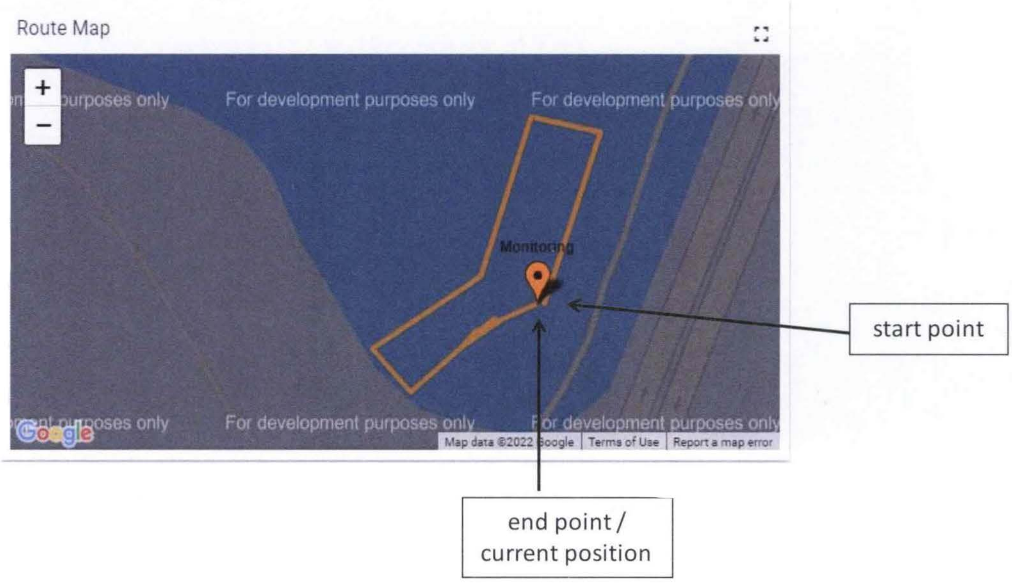


Figure 4.3: The waypoint coordination of monitoring system.

Figure 4.4 shows the series movement of the SUV through virtual panel. The first picture is where the SUV begin to deploy and start moving along the waypoint that was drawn in Figure 4.4. The last picture of Figure 4.4 is shown the current position or latest position of the SUV that was tracked. At the current position indicator as in Figure 4.5, the latitude coordinate, longitude coordinate and time is shown when click at the indicator.

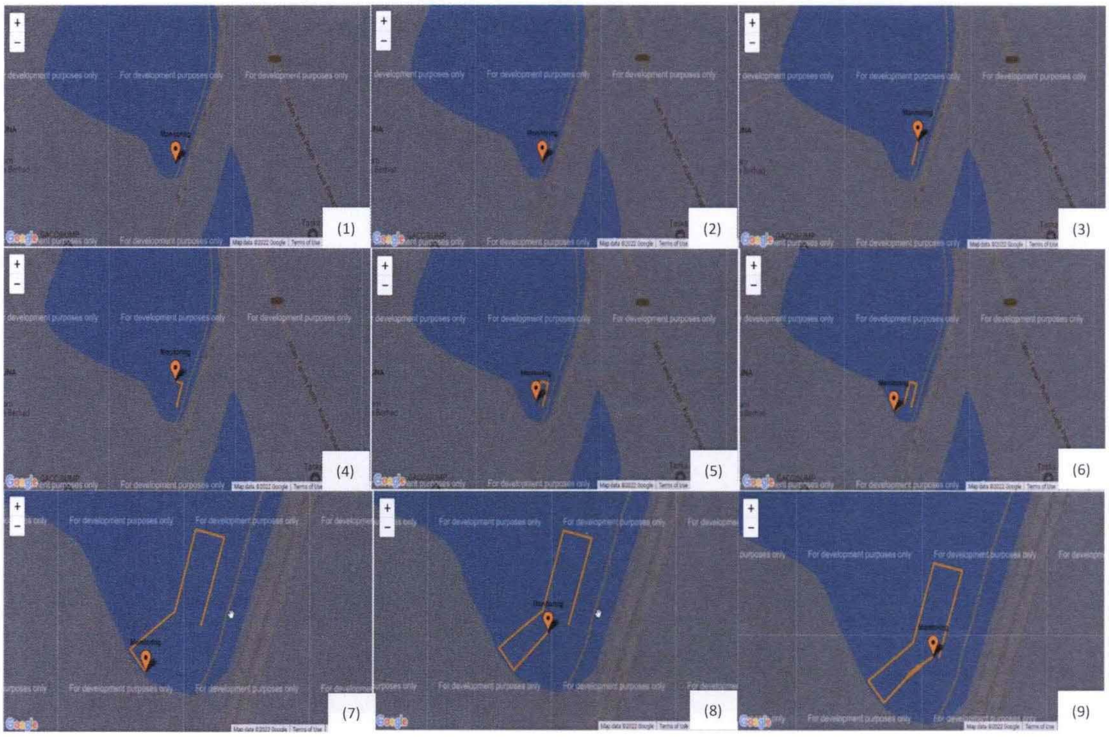


Figure 4.4: The series movement of waypoint coordination.

Figure 4.5: The status location coordinate of monitoring system in map.

Figure 4.6 is shown the speed of the SUV in the GUI. The speed of the SUV is measured by using GPS. The difference between two location coordinates will determine the speed of the SUV. Speed (S) is calculated with distance travelled (d) divide by time travelled (t), as express in Equation 4.1 is as follows;

$$S = \frac{d}{t} \tag{4.1}$$

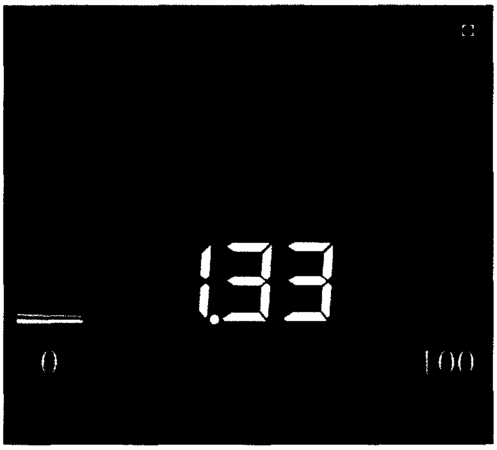


Figure 4.6: Speed indicator of the SUV.

Figure 4.7 shows the data information that is used in the GUI. The data is uploaded in every 15 seconds. The data consists of the latitude, longitude, altitude, and speed of the SUV.

Timeseries table

Realtime - last 5 hours

Timestamp ↓	latitude	longitude	altitude	speed
2022-01-23 17:35:19	3.540744	103.435	8.8	0.01852
2022-01-23 17:35:03	3.540744	103.435	8.8	0.01852
2022-01-23 17:34:48	3.540741	103.435	8	0.05556
2022-01-23 17:34:32	3.540741	103.435	8	0.05556
2022-01-23 17:34:16	3.540745	103.435	8.6	2.35204
2022-01-23 17:34:01	3.54075	103.4349	6.4	2.35204
2022-01-23 17:33:46	3.540743	103.4349	6.4	0.42596
2022-01-23 17:33:30	3.540692	103.4349	6.3	1.00008
2022-01-23 17:33:14	3.540692	103.4349	6.3	1.00008
2022-01-23 17:32:59	3.540638	103.4348	6.3	1.14824

Figure 4.7: The data information of monitoring system.

Figure 4.8 shows the camera view of the SUV. The output video image from camera is displayed using mobile phone screen. The RF receiver is connected to phone that allows mobile phone to receive the video image from camera.

Figure 4.8: The camera view of the SUV.

The project successfully obtained the required data information but there are some technical problems has found out. From the generated waypoint on the map there has slightly uneven with the amount of data that was sent to the web server. All the coordinates that was obtained is shown in the data information monitoring system in GUI but only some of it has plotted for generate waypoint movement of the SUV.

CHAPTER 5

CONCLUSION

5.1 Conclusion

In conclusion, this project is developed a SUV for water monitoring and surveyor and successful to operated. Catamaran approach design that has been developed for the SUV platform design is stable enough and have remarkable buoyancy to through tidal water area. Moreover, the design is simple and low cost. The SUV is also implemented with system that able to trace waypoint path of real-time coordination on the GUI panel and real-time viewing on the water surface.

The increased acceptance of unmanned vehicle system has opened the field for SUVs to be employed as a tool in many operations such as monitoring and surveyor. In addition, the SUVs have potential to reduce risk for manned forces and perform tasks that cannot be done by manned vehicles. The developed SUV in this project have the basic vehicle utility with tracking system technology for having organize track record data collection. The SUV has completed a series of test at UMP lake B.

5.2 Recommendation

The next improvement for this project is regarding the operation system which adding the autonomous operation ability. With having autonomous operation SUV, the repeatable working area would not require human intervention for operation. In aspect of hardware, the microcontroller could be changed to microcontroller that have multi core processor for multithread operation. In aspect of implementation, some sensors also can be added to create specific task operation such as sensor for detecting water quality of the area. In addition, adding safety system where the SUV will return back to station when the battery is low and giving precaution indication when remote controller distance area is almost out of signal.

REFERENCES

[1] J. Villa, J. Paez, C. Quintero, E. Yime, and J. Cabrera, "Design and control of an unmanned surface vehicle for environmental monitoring applications," in 2016 IEEE Colombian Conference on Robotics and Automation (CCRA), 2016: IEEE, pp. 1-5.

[2] E. T. Steimle and M. L. Hall, "Unmanned surface vehicles as environmental monitoring and assessment tools," in OCEANS 2006, 2006: IEEE, pp. 1-5.

[3] O. Yaakob et al., "Development of unmanned surface vehicle (USV) for sea patrol and environmental monitoring," in Proceedings of the International Conference on Marine Technology, Kuala Terengganu, Malaysia, 2012, pp. 20-22.

[4] G. Thirunavukkarasu, L. Patrick, B. Champion, L. Chua, and M. Joordens, "Design and development of a low-cost autonomous surface vehicle," in 2017 12th System of Systems Engineering Conference (SoSE), 2017: IEEE, pp. 1-6.

[5] J.-D. Warren, J. Adams, and H. Molle, "Arduino for robotics," in Arduino robotics: Springer, 2011, pp. 51-82.

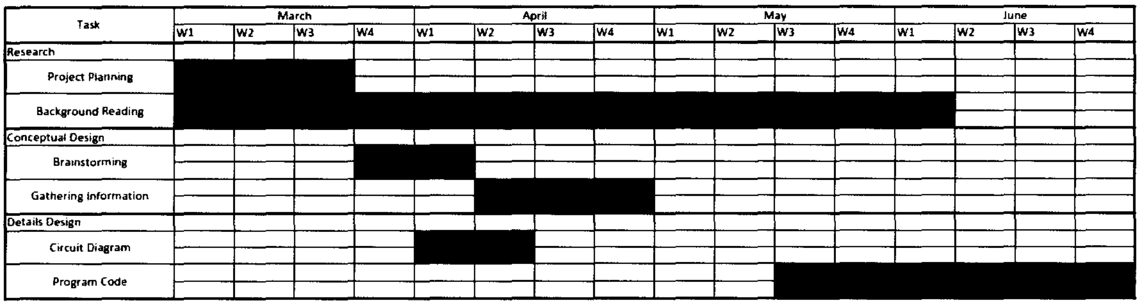
[6] S. Lee, G. Tewolde, and J. Kwon, "Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application," in 2014 IEEE world forum on internet of things (WF-IoT), 2014: IEEE, pp. 353-358.

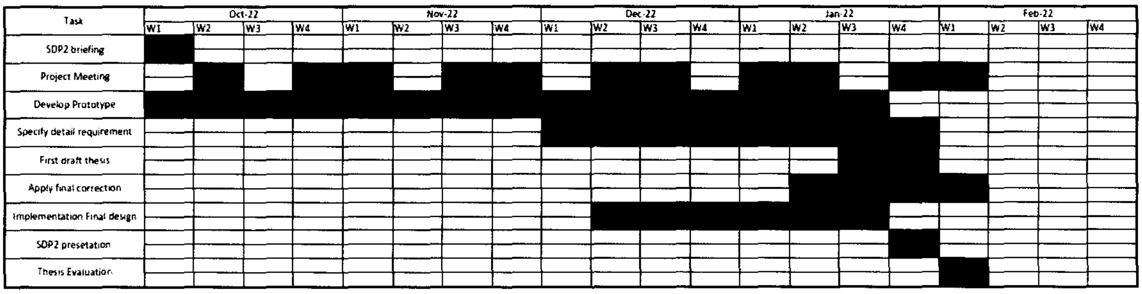
[7] J. M. M. Khin and N. N. Oo, "Real-time vehicle tracking system using Arduino, GPS, GSM and web-based technologies," International Journal of Science and Engineering Applications, vol. 7, no. 11,433-436, 2018.

[8] S. Siyang and T. Kerdcharoen, "Development of unmanned surface vehicle for smart water quality inspector," in *2016 13th International conference on electrical engineering/electronics, computer, telecommunications and information technology (ECTI-CON)*, 2016: IEEE, pp. 1-5.

APPENDICES

Appendix A: Gantt Chart for the Development of the SUV for monitoring and Surveyor





Appendix B: List Component Price for SUV development.

Table 5.1: Costing for development of the SUV

No	Description	Quantity	Price	Total
1	Flysky FS-i4 AFHDS 2A 2.4GHz 4CH Radio Sysrem Transmitter for RC Helicopter Glid	1	RM 100.85	RM 100.85
2	Eachine ROTG01 Pro UVC OTG 5.8G 15+0ch Channel FPV Receiver Audio Smartphone	1	RM 62.07	RM 62.07
3	Eachine TX06 700TVL FOV Smart Audio Mini FPV Camera AIO Transmitter for RC Drone	1	RM 43.40	RM 43.30
4	UNO Atmel DIP ATMEGA328P UNO R3 / Arduino Uno R3	1	RM 64.50	RM 64.50
5	SIM900A GSM GPRS Wireless Extension Module Board Free Antenna for Arduino/ Handphone SIM CARD Send SMS Call Ringcall	1	RM 34.40	RM 34.40
6	U-blox NEO-M8N GPS Module Tracking with Antenna GY-GPSV3-NEO Ublox	1	RM 64.50	RM 64.50
7	12V Underwater Thruster Brushless Motor 80mm propeller 3-5kg 3s	1	RM 109.80	RM 109.80
8	40A brushless ESC speed control	1	RM 23.17	RM 23.17
9	TIGER 2S-4S RC Lipo Battery Batteries 4500 5400 2s 7.4v 3s 11.1v	1	RM 119.66	RM 119.66
10	Syma X5C X5C-1.37V 25C 300mah 500mah 800mah 1200mah 1S LiPo Li-Po Battery – RC Car Drone Quadcopter Helicopter MP3	1	RM 17.90	RM 17.90

11	Tiger RC Lipo battery 2s 4500maH 7.4v	1	RM 69.30	RM 69.30
12	SG90 180/360-degree 9g Micro Servo Motor (blue) with Accessories	2	RM 7.83	RM 15.66
13	Male to Female | Jumper Wire Cable 40-Wires M-F- 10cm / 20 cm / 30cm for Arduino Raspherry Breadboard Pin-Headers Sensor	1	RM 3.69	RM 3.69
14	Male to Male | Jumper Wire Cable 40- Wires M-M- 10cm / 20cm / 30cm for Arduino Raspberry Breadboard Sensor	1	RM 3.69	RM 3.69
15	Mini USB Cable (Data & Power) @ USB Type-B (Male) to USB Type-A (Male), for Arduino NANO	1	RM 3.59	RM 3.59
16	FTDI – USB to TTL Serial, (TX/RX) UART Converter Module FT232RL (Programmer Arduino Pro, Pro Mini, LilyPad, STM)	1	RM 9.59	RM 9.59
17	Strip (Vero) Board, Pitch 2.54mm Veroboard Stripboard PCB Protoboard	1	RM 6.89	RM 6.89
18	Clear Acrylic Display Case 25x25x10cm	1	RM 90.00	RM 90.00
19	[H&D] Feilun FT012-4 Steering Rudder Tail Vane Spare Parts Kits for Feilun FT012 RC Boat	1	RM 13.18	RM 13.18
20	3/4" (20mm) PVC PIPE 6.0000 FT	6	RM 1.00	RM 6.00
21	3/4" (20mm) PVC 90' 4.0000 NOS	4	RM 1.50	RM 6.00
22	3/4"(20mm) PVC TEE'D' 4.0000 NOS	4	RM 2.00	RM 8.00

23	PVC END CAP 20MM (3/4") 4.0000 NOS	4	RM 1.00	RM 4.00
24	PVC PIPE CUTTER EA23-1/2/4 – 12/72 9071728	1	RM 12.50	RM 12.50
25	JEBSEN PVC PIPE CEMENT 100g WB104-6AB – 80 9555395800489	1	RM 3.31	RM 3.31
26	Glue stick 8pcs B20cm x 1 set	1	RM 3.85	RM 3.85
27	Hinge GQ-288 1inc x 1pcs	1	RM 1.70	RM 1.70
28	Silicon sealant 1018GP VS X 1pcs	1	RM 7.90	RM 7.90
29	Paper clip combination VS x1set	1	RM 1.25	RM 1.25
TOTAL				RM 876.75

Appendix C: Embedded Programming Code

The program code consists of remote-control process program code, and monitoring process program code.

Monitoring Process Program Code:

```
#include <TinyGPS++.h>
#include <ArduinoJson.h>

#define GSM_PORT Serial3
#define GPS_PORT Serial2
#define DEBUG_SERIAL

TinyGPSPlus gps;

double hour, minute, second, month, day, year;
double Latitudee,Longitudee,Altitudee,Speede;
String Coursee;
int Houre, Minutee, Seconde;

enum _parseState {
PS_DETECT_MSG_TYPE,

PS_IGNORING_COMMAND_ECHO,

PS_HTTPACTION_TYPE,
PS_HTTPACTION_RESULT,
PS_HTTPACTION_LENGTH,

PS_HTTPREAD_LENGTH,
PS_HTTPREAD_CONTENT
    };

enum _actionState {
AS_IDLE,
AS_WAITING_FOR_RESPONSE
};

byte actionState = AS_IDLE;
unsigned long lastActionTime = 0;

byte parseState = PS_DETECT_MSG_TYPE;
char buffer[80];
byte pos = 0;
```

```
int httpResult = 0;
int contentLength = 0;

void resetBuffer() {
memset(buffer, 0, sizeof(buffer));
pos = 0;
}

void sendGSM(const char* msg, int waitMs = 500) {
GSM_PORT.println(msg);
while(GSM_PORT.available()) {
parseATText(GSM_PORT.read());
}
delay(waitMs);
}

void dropGSM(unsigned long now) {
while(GSM_PORT.available()) {
lastActionTime = now;
parseATText(GSM_PORT.read());
}
}

void parseATText(byte b) {

#ifdef DEBUG_SERIAL
Serial.write(b);
#endif

buffer[pos++] = b;

if ( pos >= sizeof(buffer) )
resetBuffer(); // just to be safe

#ifdef DEBUG_SERIAL
#endif

switch (parseState) {
case PS_DETECT_MSG_TYPE:
{
if ( b == '\n' )
resetBuffer();
else {
if ( pos == 3 && strcmp(buffer, "AT+") == 0 ) {
parseState = PS_IGNORING_COMMAND_ECHO;
```

```
}

else if ( b == ':' ) {
#ifdef DEBUG_SERIAL
Serial.print("Checking message type: ");
Serial.println(buffer);
#endif

if ( strcmp(buffer, "+HTTPACTION:") == 0 ) {
#ifdef DEBUG_SERIAL
Serial.println("Received HTTPACTION");
#endif
parseState = PS_HTTPACTION_TYPE;
}
else if ( strcmp(buffer, "+HTTPREAD:") == 0 ) {
#ifdef DEBUG_SERIAL
Serial.println("Received HTTPREAD");
#endif
parseState = PS_HTTPREAD_LENGTH;
}
resetBuffer();
}
}
}
break;

case PS_IGNORING_COMMAND_ECHO:
{
if ( b == '\n' ) {
#ifdef DEBUG_SERIAL
Serial.print("Ignoring echo: ");
Serial.println(buffer);
#endif
parseState = PS_DETECT_MSG_TYPE;
resetBuffer();
}
}
break;

case PS_HTTPACTION_TYPE:
{
if ( b == ',' ) {
#ifdef DEBUG_SERIAL
Serial.print("HTTPACTION type is ");
Serial.println(buffer);
#endif
parseState = PS_HTTPACTION_RESULT;
```

```
resetBuffer();
}
}
break;

case PS_HTTPACTION_RESULT:
{
if ( b == ',' ) {
#ifdef DEBUG_SERIAL
Serial.print("HTTPACTION result is ");
Serial.println(buffer);
#endif
httpResult = atoi(buffer);
parseState = PS_HTTPACTION_LENGTH;
resetBuffer();
}
}
break;

case PS_HTTPACTION_LENGTH:
{
if ( b == '\n' ) {
#ifdef DEBUG_SERIAL
Serial.print("HTTPACTION length is ");
Serial.println(buffer);
#endif

contentLength = atoi(buffer);
// now request content
if ( contentLength > 0 ) {
GSM_PORT.print("AT+HTTPREAD=0,");
GSM_PORT.println(buffer);
}
else
actionState = AS_IDLE;
parseState = PS_DETECT_MSG_TYPE;
resetBuffer();
}
}
break;

case PS_HTTPREAD_LENGTH:
{
if ( b == '\n' ) {
contentLength = atoi(buffer);
#ifdef DEBUG_SERIAL
Serial.print("HTTPREAD length is ");
Serial.println(contentLength);
```

```
Serial.print("HTTPREAD content: ");
#endif

parseState = PS_HTTPREAD_CONTENT;
resetBuffer();
}
}
break;

case PS_HTTPREAD_CONTENT:
{

#ifdef DEBUG_SERIAL
Serial.write(b);
#endif

contentLength--;

if ( contentLength <= 0 ) {
// all content bytes have now been read

parseState = PS_DETECT_MSG_TYPE;
resetBuffer();

#ifdef DEBUG_SERIAL
Serial.print("\n\n\n\n");
#endif

actionState = AS_IDLE;
}
}
break;
}
}

void setup()
{

GSM_PORT.begin(9600);
GPS_PORT.begin(9600);
#ifdef DEBUG_SERIAL
Serial.begin(9600);
#endif

sendGSM("AT+SAPBR=3,1,\"APN\",\"digi\"");
sendGSM("AT+SAPBR=1,1",3000);
sendGSM("AT+HTTPINIT");
```

```
sendGSM("AT+HTTPPARA=\"CID\",1");
}

void loop()
{
unsigned long now = millis();

while (GPS_PORT.available() > 0)
if (gps.encode(GPS_PORT.read()))

GPSInfo(Latitudee,Longitudee,Altitudee, Speede, Coursee, Houre, Minutee, Seconde);

if (now > 5000 && gps.charsProcessed() < 10)
{
#ifdef DEBUG_SERIAL
Serial.println("No GPS detected");
#endif
while(true);
}



if ( actionState == AS_IDLE ) {
if ( now > lastActionTime + 30000 ) {

DynamicJsonDocument object(1024);

object["latitude"] = Latitudee;
object["longitude"] = Longitudee;
object["altitude"] = Altitudee;
object["speed"] = Speede;
object["course"] = Coursee;
object["hour"] = Houre;
object["minute"] = Minutee;
object["second"] = Seconde;

serializeJson(object, Serial);
String sendtoserver;
serializeJsonPretty(object, sendtoserver);

delay(1000);

if(Latitudee = 0 && Longitudee = 0)
{
}
else
{
```

```
GSM_PORT.println("AT+HTTPPARA=\"URL\",\"http://demo.thingsboard.io/api/v1/P
nnyJoUROfl0O0WVFdjd/telemetry\"");

delay(1000);
dropGSM(now);

GSM_PORT.println("AT+HTTPPARA=\"CONTENT\",\"application/json\"");
delay(1000);
dropGSM(now);

GSM_PORT.println("AT+HTTPDATA=" + String(sendtoserver.length()) +",100000");
#ifdef DEBUG_SERIAL
Serial.println(sendtoserver);
#endif

delay(1000);
dropGSM(now);

GSM_PORT.println(sendtoserver);
delay(1000);
dropGSM(now);

GSM_PORT.println("AT+HTTPACTION=1");
delay(1000);
dropGSM(now);

GSM_PORT.println("AT+HTTPREAD");
delay(1000);
dropGSM(now);
}

delay(1500);
lastActionTime = now;
httpResult = 0;
actionState = AS_WAITING_FOR_RESPONSE;

}
}
else {

if ( now > lastActionTime + 15000 )
{
actionState = AS_IDLE;
parseState = PS_DETECT_MSG_TYPE;
resetBuffer();
}
}
```

```
dropGSM(now);
}

void GPSInfo(double &Latitude, double &Longitude, double &Altitude, double
&Speed, String &Course, int &Hour, int &Minute, int &Second)
{

if (gps.location.isValid())
{
Latitude = gps.location.lat(), 4 ;
Longitude = gps.location.lng(), 4 ;
}
if (gps.time.isValid())
{
Hour = gps.time.hour()+8;
if(Hour>23)
{
Hour-=24;
}
Minute = gps.time.minute();
Second = gps.time.second();
}
if (gps.altitude.isValid())
{
Altitude = gps.altitude.meters();
}
if (gps.speed.isValid())
{
Speed = gps.speed.kmph();
}
delay(1000);
}
```

Remote-control Process Program Code:

```
#include<Servo.h>

int  motor_pwm, servo_pwm ;
#define motor 5
#define servo_motor 6
#define servo_left_motor 3

long rc_pulse_servo,rc_pulse_motor, dutycycle_servo, dutycycle_motor;

Servo servo_right;
Servo servo_left;
Servo motorset;
```

```
void setup()
{
 Serial.begin(9600);

 pinMode(10,INPUT);
 pinMode(11, INPUT);
 servo_right.attach(servo_motor);
 servo_left.attach(servo_left_motor);
 motorset.attach(motor);
}

void loop()
{
 rc_pulse_servo = pulseIn(10,HIGH);
 rc_pulse_motor = pulseIn(11, HIGH);

 Serial.println(rc_pulse_motor);
 if(rc_pulse_motor > 992 && rc_pulse_motor < 2100)
 {
  ControlMotor(rc_pulse_motor);
 }

 if(rc_pulse_servo > 992 && rc_pulse_servo < 2000)
 {
  ControlServo(rc_pulse_servo);
 }
}

void ControlServo(long dutycycle_servo)
 {
 int servo_pwm = map(dutycycle_servo,992,2000,0,180);
 servo_right.write(servo_pwm);
 servo_left.write(servo_pwm);
 }

 void ControlMotor(long dutycycle_motor)
 {
 int motor_pwm = map(dutycycle_motor,992,2100,0,180);
 motorset.write(motor_pwm);
 }
```