

## Development of control system for quadrotor unmanned aerial vehicle using LoRa wireless and GPS tracking

Teddy Surya Gunawan<sup>1</sup>, Wan Atheerah Yahya<sup>2</sup>, Erwin Sulaeman<sup>3</sup>, Mira Kartiwi<sup>4</sup>, Zuriati Janin<sup>5</sup>

<sup>1,2</sup>Department of Electrical and Computer Engineering, International Islamic University Malaysia, Malaysia

<sup>1</sup>Fakultas Teknik dan Ilmu Komputer, Universitas Potensi Utama, Indonesia

<sup>3</sup>Department of Mechanical Engineering, International Islamic University Malaysia, Malaysia

<sup>4</sup>Departement of Information Systems, International Islamic University Malaysia, Malaysia

<sup>5</sup>Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia

---

### Article Info

#### Article history:

Received Jan 17, 2020

Revised Apr 29, 2020

Accepted May 11, 2020

---

#### Keywords:

GPS

LoRa

PID controller

Quadrotor UAV

Telemetry data

---

### ABSTRACT

In the past decades, there has been a growing interest in unmanned aerial vehicles (UAVs) for educational, research, business, and military purposes. The most critical data for a flight system is the telemetry data from the GPS and wireless transmitter and also from the gyroscope and accelerometer. The objective of this paper is to develop a control system for UAV using long-range wireless communication and GPS. First, Matlab simulation was conducted to obtain an optimum PID gains controller. Then LoRa wireless was evaluated during clear and rainy days. Static and dynamic points measurement was conducted to validate and optimize GPS accuracy. GeoMapping in Matlab and Google GPS GeoPlanner were then used to analyze the traveled UAV flight path.

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.*



---

### Corresponding Author:

Teddy Surya Gunawan,  
Department of Electrical and Computer Engineering,  
International Islamic University Malaysia, Malaysia.  
Email: [tsgunawan@iium.edu.my](mailto:tsgunawan@iium.edu.my)

---

## 1. INTRODUCTION

In the past few years, the quadrotor unmanned aerial vehicle (UAV) has gained interest due to its compact size, agility, and autonomous flight [1]. UAV is an aircraft without a pilot on board and can be flown by a remote controller from a ground control station. UAVs can fly autonomously and also manually by remote controller. There are many types of UAV across the world, and they have specific task and mission that depends on the person who built the UAV. A quadrotor system can be used for surveillance [2, 3], rescue [4, 5], aerial photography [6, 7], precision agriculture [8, 9], and other works that are dangerous or space limited for human beings [10].

At first, UAV was not autonomously controlled. It was controlled by remote control and by manual radio, just like a typical aircraft. The control systems of quadrotor UAV has been conducted by many researchers. In [11, 12], the rigid body dynamics of a quadrotor UAV and several effects of aerodynamics were investigated. Several controller can be implemented, including the PID controller [12], backstepping controller [13, 14], sliding-mode controller [15, 16], adaptive control [17], fuzzy logic [18, 19], quantitative feedback [20], reinforcement learning [21], etc. A review of various control systems' performance can be found in [22].

Despite a sophisticated digital avionic system installed on the present aircraft, which defines aircraft altitude, speed, and continuous position, we still need the UAV in various emergency situations. In this paper, we introduce the use of long-range (LoRa) wireless module and also the GPS tracking to enable the Quadrotor UAV to obtain telemetry data, for example, a continuous GPS position. The dynamics and control system of the quadrotor UAV was modeled and simulated in Matlab. In order to get precise and continuous data telemetry from the UAV to the ground control station (GCS), we use the long-range (LoRa) wireless communication system [23].

## 2. QUADROTOR UNMANNED AERIAL VEHICLE (UAV) MODELLING

The majority of commercial quadrotors are marketed with the aim of doing aerial photography, e.g., Aeryon scout, Microdrones md4-1000, Quanser Qball, Parrot AR. Drone 2.0, DJI Phantom 3, AscTec, Mikrokopter, etc. In addition, there are many communities and university-based open-source projects for quadrotors development, e.g., STARMAC X4, ANU X4-Flyer, Arduino ATmega based, Mikrokopter based, STM32 based, etc [24]. The structure of the quadrotor UAVs investigated in this paper is as shown in Figure 1. There are two key states of the quadrotor UAV, i.e., position  $(X, Y, Z)$  and attitude (roll, pitch, yaw). The position describes the translation between the world frame and body frame. The attitude describes rotation between the world and body frames. Roll describes how the right and left sides are elevated with respect to each other (if the left side pushed down, and the right side is pushed up, the UAV will move to the left). Pitch describes how the front and back are elevated with respect to each other (if the front is moved down, and the back is up, it will move forward). And yaw describes the rotation of the drone (if the front, back, and both sides are leveled, it can turn around the yaw axis without shifting position).

The dynamic of quadrotor UAVs can be formulated as described in [12, 17]. The position  $p_x, p_y, p_z$  and attitude (roll, pitch, yaw)  $\varphi, \theta, \psi$ , can be formulated as:

$$\begin{aligned} \ddot{p}_x &= (\cos \varphi \sin \theta \cos \psi + \sin \varphi \sin \psi) \frac{\tau_4}{m} + d_1 \\ \ddot{p}_y &= (\cos \varphi \sin \theta \sin \psi - \sin \varphi \cos \psi) \frac{\tau_4}{m} + d_2 \\ \ddot{p}_z &= \cos \varphi \cos \theta \frac{\tau_4}{m} - g + d_3 \\ \ddot{\varphi} &= \dot{\varphi} \dot{\psi} \left( \frac{J_{zz} - J_{xx}}{J_{yy}} \right) + \frac{J_R}{J_{yy}} \dot{\varphi} \Omega_R + \frac{L}{J_{yy}} \tau_1 + d_4 \\ \ddot{\theta} &= \dot{\theta} \dot{\psi} \left( \frac{J_{yy} - J_{zz}}{J_{xx}} \right) - \frac{J_R}{J_{xx}} \dot{\theta} \Omega_R + \frac{L}{J_{xx}} \tau_2 + d_5 \\ \ddot{\psi} &= \dot{\varphi} \dot{\theta} \left( \frac{J_{xx} - J_{yy}}{J_{zz}} \right) + \frac{1}{J_{zz}} \tau_3 + d_6 \end{aligned} \quad (1)$$

where  $m$  is the mass of the quadrotor and its payload,  $J_{xx}, J_{yy}, J_{zz}$  are the moments of inertia of the quadrotor,  $L$  is the length from the barycentre to the center of each rotor,  $J_R$  and  $\Omega_R$  provide the moments of inertia and angular velocity of the propeller blades,  $d_1, \dots, d_6$  are bounded disturbances.  $\tau_1, \tau_2, \tau_3$  are thrusts in the direction of roll, pitch, yaw and  $\tau_4$  is the collective thrust. The relationships between the thrust  $[\tau_1, \tau_2, \tau_3, \tau_4]^T$  and control signals  $[u_1, u_2, u_3, u_4]^T$  for motors are shown as:

$$\begin{aligned} \tau_1 &= c_T(u_2 - u_4) \\ \tau_2 &= c_T(u_1 - u_3) \\ \tau_3 &= c_M(u_1 - u_2 + u_3 - u_4) \\ \tau_4 &= c_T(u_1 + u_2 + u_3 + u_4) \end{aligned} \quad (2)$$

where  $[u_1, u_2, u_3, u_4]^T$  are control signals that range from 0 to 1,  $C_T$  and  $C_M$  are the parameters which translate control signals to thrusts and torque. The proportional integral derivative (PID) controller can be used due to its relatively simple structure, which can be easily understood and implemented in practice.

$$u(t) = K_P e(t) + K_I \int_0^t e(\lambda) d\lambda + K_D \frac{de(t)}{dt} \quad (3)$$

where  $u(t)$  is the input signal,  $e(t) = u(t) - \tau(t)$  is the error signal,  $K_P, K_I, K_D$  are the proportional, integral, and derivative controller gains, respectively.

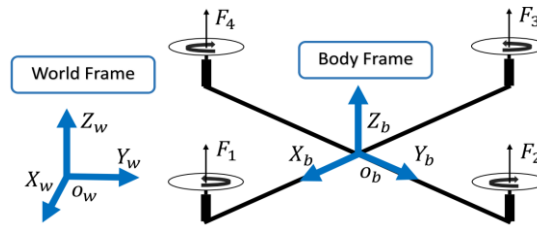


Figure 1. The structure of quadrotor UAVs adopted from [21]

### 3. DESIGN OF CONTROL SYSTEM FOR QUADROTOR UAV

For the UAV design, the original UAV weight is estimated to be 550 grams with the battery. The weight of the additional battery, LoRa wireless, GPS modules, and Arduino Mega microcontroller are counted as the payload, which is estimated to be around 250 grams. Total weight is estimated to be around 800 grams. For the design of a wireless module, there are several characteristics required, such as low data rate, longer range, and very low power consumption. The high data rate is not required due to the nature of the commands being sent or the data being received. In our research, we just transmit and receive GPS locations from the UAV to the ground station. If image or video was required to be transmitted, then a higher data rate will be necessary. For long-range control and data transmission, wireless communication should have long-range with relatively high fidelity. Moreover, power consumption needs to be minimal to conserve limited battery power at UAV.

In this research, the long-range wireless module is used to obtain a continuous reading from the GPS sensor, and the communication range should be between 5 to 10 km. The wireless module should be capable of self-standing in terms of communication infrastructure and power supply since standard communication systems could have collapsed during unpredictable events and emergencies. For this purpose, the wireless module used in this research is Semtech Lora 868 MHz 915 MHz SX1272 Libelium radio frequency module [25], as shown in Figure 2 (a). It has a dual-frequency band, i.e., 863-870 MHz (Europe) and 902-928 MHz (US). The transmission power is 14 dBm. The sensitivity is -134 dBm. Moreover, the communication distance can cover more than 22 km.

The SX1272 module uses the SPI protocol (Serial peripheral Interface) pins for communication. The SPI port allows faster communication and frees up the Waspote's UART (Universal Asynchronous Receiver/Transmitter) for other purposes. The SX1272 module does not implement any security method. Encryption is provided through the Waspote Encryption library. Specifically, through the AES algorithm with the symmetric key, with a length of 128, 192, or 256 bits. Programming the microcontroller is a simple three steps process: write the code, compile the code, and upload the code into the microcontroller.

The GPS module selected in this research is Ublox Neo-6M, as shown in Figure 2 (b). It produces outputs of the current position and timing data in the format of NMEA 0183 and has a positioning accuracy of 10 meters without and 5 meters with WAAS (Wide Area Augmentation System). It provides augmentation information to GPS receivers to enhance the accuracy and reliability of position estimates and also expand the GPS, with the goal of improving its accuracy, integrity, and availability. GPS U-Blox Neo 6M modules give different approaches for geo-tagging and stitching images based on GNSS (global navigation systems).

The power supply required for this GPS module is 3.6 V suitable for Arduino. The DC current through any digital I/O pin is around ten mA. The antenna gain is 50 dB, and the UART timing has a maximum bandwidth of 100 kbit/s. Many researchers have used this GPS U-Blox because of its usability and data accuracy. Currently, there are 32 GPS satellites orbiting the Earth, in which ten satellites was enough to provide high accuracy data, while four satellite will be the minimum. GPS satellite send course acquisition (C/A) code to GPS receiver on earth station and give the accurate positioning to the user on Earth. The GPS user on Earth can even get the distance from one point to another point according to GPS tracking.

For the payload at the UAV, Arduino Mega will be used and connected to GPS and LoRa wireless modules [26, 27]. Multiprotocol radio shield, as shown in Figure 2 (c), is used to connect Arduino Mega and the Wireless Libelium LoRa module. For the ground station, Arduino Uno and LoRa receiver are used and connected to the computer to obtain the transmitted data from UAV. The specification of the UAV is a quadcopter with six axes of movement, as shown in Figure 3. The controller is a 2.4 GHz transmitter. The length and width of the quadcopter are 500 mm with a height of 190 mm. LiPo battery has a capacity of 7.4 V/2200 mA and power of 16.3 W. The payload for the UAV is including the GPS receiver, LoRa wireless module, and an HD camera for First person view (FPV). However, to minimize battery usage for a longer flight, the HD camera captured the images and saved it in the SD card, rather than real-time transmitting it to the ground station.

This UAV is controlled by a microcontroller such as Arduino and other sensors. Both of them have great importance for increasing the efficiency of the UAV and can increase the performance of the aircraft system. The payload prototype is made from plastic material for the cover of the body. A hierarchical control system is

designed where a high-level controller is responsible for GPS location and positioning and also controlling long-range radio frequency modules.

The platform of the controller has a total of 4 actuators that are responsible for orienting and translating the craft according to the outputs of the controller. The actuator lines are made up of the electronic speed controller (ESC), the motors, and the propellers. The ESC obtains power from the 7.4 V main battery and an input signal in the form of a pulse-width-modulation (PWM) command from the Arduino mega. The ESC then translates these PWM commands into the appropriate rotating magnetic field used to control the speed of the motors.

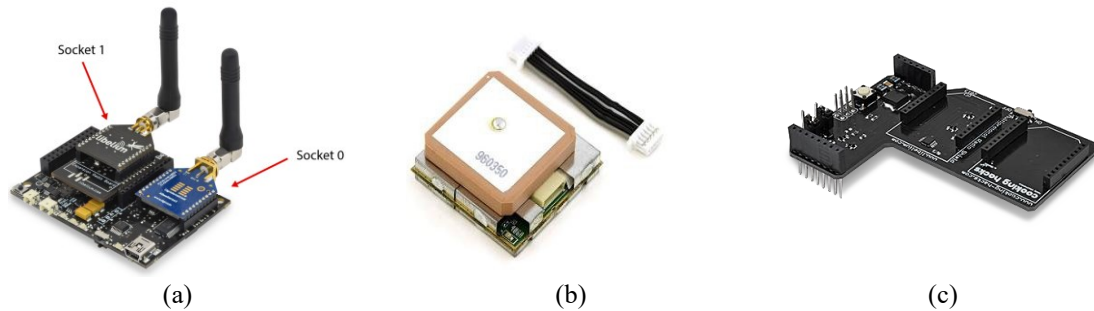


Figure 2. Hardware design for control system of quadrotor UAV;  
(a) LoRa SX1272, (b) GPS Ublox Neo-6m, (c) arduino multiprotocol shield

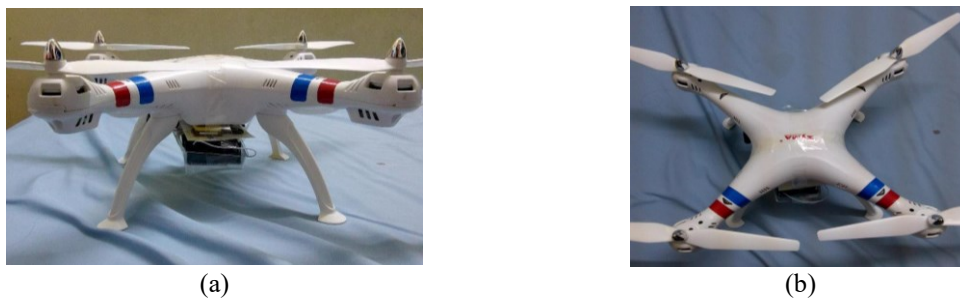


Figure 3. Front and top view of the quadcopter; (a) front view and (b) top view

#### 4. EXPERIMENTAL RESULTS AND DISCUSSION

This section describes the quadrotor Matlab simulation, experimental setup, and experiments on the LoRa wireless communication and GPS accuracy. Matlab simulation confirmed that the PID controller provided a good control system performance. Moreover, the next experiments evaluated the GPS accuracy of the pre-determined UAV flight path using GPS GeoPlanner.

##### 4.1. Matlab simulation

Matlab simulation of the aerodynamic design was used for prototyping and flight control system design of a quadrotor UAV model. Additionally, the GeoMapping Matlab toolbox is utilized for a comprehensive GPS calculation and validation. With the time calculation utilities, we can compute Julian dates, decimal year, and leap year. Aerospace toolbox provides three options for visualizing flight data. First, the interface to flight gear flight simulator lets us visualize vehicle dynamics in a sophisticated 3-D simulation framework. The flight simulator can playback flight-test data through flight gear and effectively reconstruct behavioral anomalies in the flight-test results. Matlab Aerospace Toolbox includes functions for controlling the position and altitude of a vehicle in the flight-gear simulator by using double-precision values of longitude, latitude, altitude, roll, pitch, and yaw. Second, the interface to simulink 3D animation lets us use the flight data to control vehicle position and attitude in a virtual-reality scene. The dynamics of the quadrotor are taken from [28] with some slight modification, e.g., the weight of UAV is set to 0.8kg, the length of the quadrotor is set to 0.27 m, etc. Figure 4 shows the Simulink model, while Figure 5 shows the simulation results, in which the designed PID controller has achieved a good performance. It simulates UAV to move up from the ground until a pre-defined height (modify  $z$ ). After that, the desired  $x$  and  $y$  are set to become increasing sinusoidal from 0 to 90 seconds and kept constant afterward. The simulation results show the desired position in 3D versus the actual position, forming an up and spiral movement of the UAV.

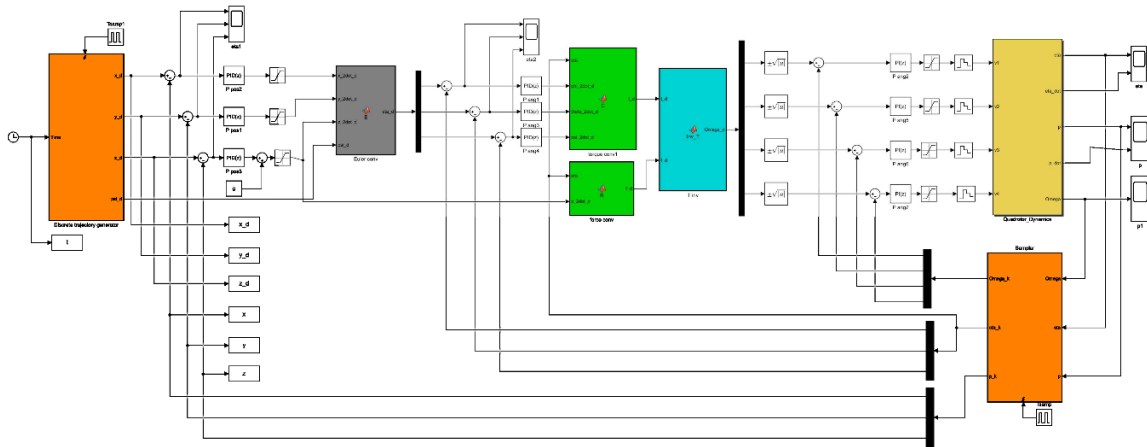


Figure 4. Simulink model of the quadrotor UAV

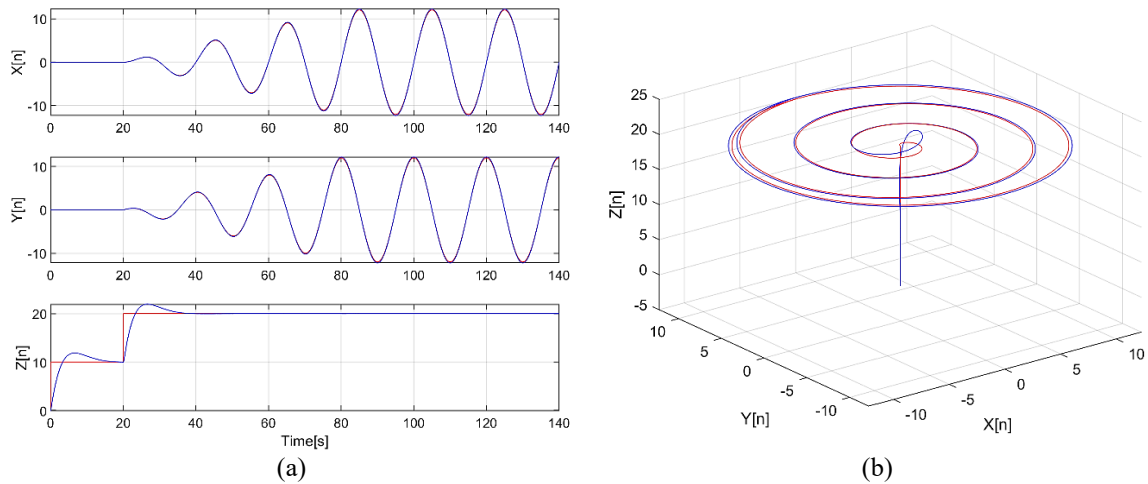


Figure 5. Quadrotor UAV simulation results; (a) desired and actual position X, Y, Z and (b) desired and actual position in 3D

**4.2. Experimental setup**

The experimental setup is shown in Figure 6. We have divided the experiments into static and dynamic points measurements. For static points measurement, the ground control station is placed at one static location, and the GPS coordinates of that location are measured using the GPS Ublox NEO, in which we obtained Point A. We then moved the GPS receiver to another location, and the GPS coordinates at that location were transmitted using LoRa to the ground station and recorded to obtain Point B. The distance between point A and point B then can be calculated.

For dynamic points measurement, the ground station GPS location has been recorded earlier (from static measurement). The LoRa transmitter is attached to the quadrotor UAV. The UAV is then flown to any locations, and its positions were transmitted to the ground station and recorded. For analysis purposes, the recorded GPS locations were then plotted using the GeoMapping tool in Matlab, in which we can further analyze the dynamic flight path.

**4.3. Experiment on LoRa**

Figure 7 shows an example of LoRa wireless receiver output at the Ground Station during a clear day and rainy day. The SNR value and packet length during the rainy day were found to be (-31, 129), (-31, 129), (-31, 127), and (3, 255), in four consecutive readings. It showed that during rain, the wireless signal is attenuated, and the SNR fluctuates to -31 dB as it experiences disturbances and noises. From this experiment, the LoRa module was found to be working as expected.

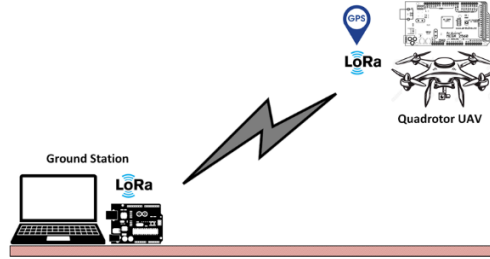


Figure 6. Experimental setup between ground station and quadrotor UAV

```

COM4 (Arduino/Genuino Uno)
Getting Channel --> ERROR
Getting Payload Length --> OK. Payload Length is: 9
Getting Header --> OK. Header (HeaderON = 0; HeaderOFF = 1): 1
Getting Power --> OK. Power is ('M'=0x0F; 'L'=0x00; 'H'=0x07): 0xSetting Power: state 1
Getting CRC --> OK. CRC (CRC_ON=1; CRC_OFF=0): 0
Getting Node Address --> OK. Node Address is: 9
Getting Mode --> ERROR

the Temperature of Lora sx1272:          9
SNR value: 0
-----
Reading configuration in module:
-----
Getting Channel --> ERROR
Getting Payload Length --> OK. Payload Length is: 9
Getting Header --> OK. Header (HeaderON = 0; HeaderOFF = 1): 1
Getting Power --> OK. Power is ('M'=0x0F; 'L'=0x00; 'H'=0x07): 0xSetting Power: state 1
Getting CRC --> OK. CRC (CRC_ON=1; CRC_OFF=0): 0
Getting Node Address --> OK. Node Address is: 9
Getting Mode --> ERROR

the Temperature of Lora sx1272:          9
SNR value: 0
-----
Reading configuration in module:
-----
Getting Channel --> ERROR
Getting Payload Length --> OK. Payload Length is: 9
Getting Header --> OK. Header (HeaderON = 0; HeaderOFF = 1): 1
Getting Power --> OK. Power is ('M'=0x0F; 'L'=0x00; 'H'=0x07): 0xSetting Power: state 1
Getting CRC --> OK. CRC (CRC_ON=1; CRC_OFF=0): 0
Getting Node Address --> OK. Node Address is: 9
Getting Mode --> ERROR

the Temperature of Lora sx1272:          9
SNR value: 0

```

(a)

```

COM29 (Arduino/Genuino Uno)
SNR value: -31
-----
Reading configuration in module:
-----
Getting Channel --> ERROR
Getting Payload Length --> OK. Payload Length is: 129
Getting Header --> OK. Header (HeaderON = 0; HeaderOFF = 1): 1
Getting Power --> ERROR
Getting CRC --> OK. CRC (CRC_ON=1; CRC_OFF=0): 0
Getting Node Address --> OK. Node Address is: 129
Getting Mode --> ERROR

the Temperature of Lora sx1272:          127
SNR value: -31
-----
Reading configuration in module:
-----
Getting Channel --> ERROR
Getting Payload Length --> OK. Payload Length is: 129
Getting Header --> OK. Header (HeaderON = 0; HeaderOFF = 1): 1
Getting Power --> ERROR
Getting CRC --> OK. CRC (CRC_ON=1; CRC_OFF=0): 0

```

(b)

Figure 7. LoRa receiver output during a clear and rainy day; (a) clear day and (b) rainy day

#### 4.4. Experiment on GPS accuracy and GeoMapping using Matlab and Geoplanner

As described in the previous section, the dynamic measurement was used to evaluate GPS accuracy. Point A is located at the Petanque area of Female Sport Complex, IIUM, while Point B is located at Female Sport Complex Gym, IIUM. This experiment was repeated eight times, and the result is shown in Table 1. In conclusion, the GPS receiver module is working as intended. Figures 8 and 9 showed the GeoMapping result using Matlab and GPS Planner, respectively. The distance traveled during the dynamic GPS measurement is around 10.23 km. In our experiment, the quadrotor UAV was controlled using manual remote control. The GPS coordinates are then recorded and transmitted to the ground station using LoRa. For the autonomous flight, we could create a flight plan based on locations in which the UAV could fly on its own following



the flight path. This can be used later on, along with the image or video recording, for various purposes, such as geospatial mapping, precision agriculture, disaster area mapping, etc.

Table 1. Latitude, longitude, and distance between two points

Delta (latitude)	longitude (B-A)	Mean latitude	Point A	Point B	Lat2	Long2	Distance (m)	Altitude Difference
-0.0007	-0.0008	3.25365	63.3	148.2	4.9E-07	0.999993225	117.6	84.9
-0.0007	-0.00048	3.25365	63.3	165.7	4.9E-07	0.999994813	117.61	102.4
-0.0007	101.6716	3.25365	0	165.7	4.9E-07	0.999994813	117.61	165.7
-0.00062	101.6715	3.25369	0	171.1	3.84E-07	0.999995931	117.61	171.1
-0.0015	-0.00078	3.25413	72.1	171.1	2.25E-06	0.999976174	187	99
-0.00152	-0.00078	3.25414	72.2	171.1	2.31E-06	0.999975534	188	98.9
-0.00176	-0.00077	3.25434	99.2	204	3.1E-06	0.999967195	212	104.8
-0.00186	-0.00083	3.25429	99.2	172.1	3.46E-06	0.999963349	134	72.9
Average:							148.93	112.46



Figure 8. Sample of GPS route that was traveled in real flight experiment using GeoMapping

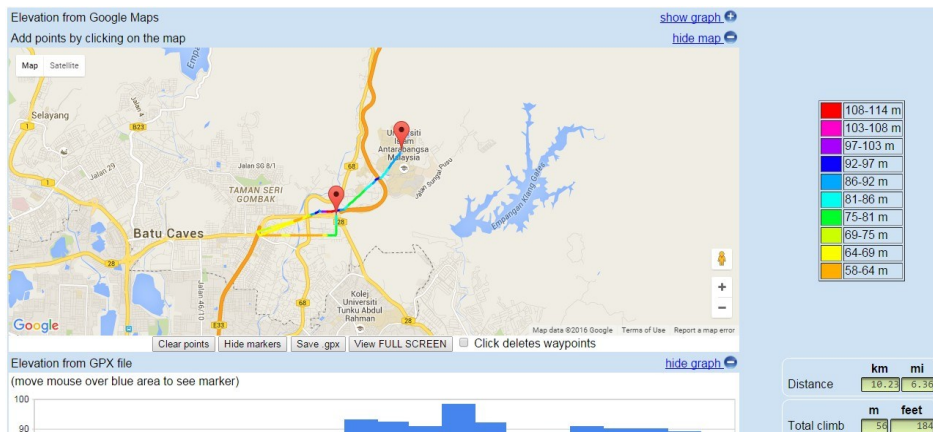


Figure 9. Sample of GPS route that was traveled in real flight experiment using GPS GeoPlanner

5. CONCLUSIONS

This paper has presented the design of the control system of quadrotor UAV using LoRa wireless communication and GPS modules. Matlab simulation was conducted to obtain an optimum PID gains controller. Next, the suitability of the LoRa wireless module to be used for telemetry data was evaluated during a clear and rainy day. The GPS telemetry data was validated and optimized using static and dynamic points measurements. Lastly, GeoMapping in Matlab and Google GPS GeoPlanner were used to analyze the real flight path. Future works include improved simulation of the quadrotor UAV using the available open-source drone, the use of various sensors and images for telemetry data, and autonomous flight using a pre-determined flight plan.

## ACKNOWLEDGMENTS

The author would like to express their gratitude to the Malaysian Ministry of Education (MOE), which has provided research funding through the Fundamental Research Grant, FRGS19-076-0684. The authors would also like to thank International Islamic University Malaysia (IIUM), Universitas Potensi Utama, and Universiti Teknologi MARA (UiTM) Shah Alam for providing facilities to support the research work.

## REFERENCES

- [1] S. Norouzi Ghazbi, Y. Aghli, M. Alimohammadi, and A. Akbari, "Quadrotors Unmanned Aerial Vehicle: A Review," *International Journal on Smart Sensing & Intelligent Systems*, vol. 9, no. 1, pp. 309-333, 2016.
- [2] R. L. Finn, D. J. C. L. Wright, and S. Review, "Unmanned aircraft systems: Surveillance, ethics and privacy in civil applications," *Computer Law & Security Review*, vol. 28, no. 2, pp. 184-194, 2012.
- [3] H. Kim and J. Ben-Othman, "A Collision-Free Surveillance System Using Smart UAVs in Multi Domain IoT," *IEEE Communications Letters*, vol. 22, no. 12, pp. 2587-2590, 2018.
- [4] M. Silvagni, A. Tonoli, E. Zenerino, and M. Chiaberge, "Multipurpose UAV for search and rescue operations in mountain avalanche events," *Geomatics, Natural Hazards and Risk*, vol. 8, pp. 18-33, 2017.
- [5] J. Qi, D. Song, H. Shang, N. Wang, C. Hua, C. Wu, X. Qi, and J. Han, "Search and rescue rotary-wing uav and its application to the lushan ms 7.0 earthquake," *Journal of Field Robotics*, vol. 33, no. 3, pp. 290-321, 2016.
- [6] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 899-922, 2016.
- [7] C. Kanellakis and G. Nikolakopoulos, "Survey on computer vision for UAVs: Current developments and trends," *Journal of Intelligent & Robotic Systems*, vol. 87, pp. 141-168, 2017.
- [8] V. Puri, A. Nayyar, and L. Raja, "Agriculture drones: A modern breakthrough in precision agriculture," *Journal of Statistics and Management Systems*, vol. 20, no. 4, pp. 507-518, 2017.
- [9] U. R. Mogili and B. Deepak, "Review on application of drone systems in precision agriculture," *Procedia computer science*, vol. 133, pp. 502-509, 2018.
- [10] P. Liu, A. Y. Chen, Y. N. Huang, J. Y. Han, J. S. Lai, S. C. Kang, T. H. Wu, M. C. Wen, and M. H. Tsai, "A review of rotorcraft unmanned aerial vehicle (UAV) developments and applications in civil engineering," *Smart Structures and Systems*, vol. 13, no. 6, pp. 1065-1094, 2014.
- [11] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 153-158, 2007.
- [12] A. L. Salih, M. Moghavvemi, et al., "Modelling and PID controller design for a quadrotor unmanned air vehicle," *2010 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*, pp. 1-5, 2010.
- [13] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, "Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor UAV," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 6, pp. 5044-5056, 2016.
- [14] F. Chen, W. Lei, K. Zhang, G. Tao, and B. Jiang, "A novel nonlinear resilient control for a quadrotor UAV via backstepping control and nonlinear disturbance observer," *Nonlinear Dynamics*, vol. 85, no. 2, pp. 1281-1295, 2016.
- [15] J.-J. Xiong and G.-B. Zhang, "Global fast dynamic terminal sliding mode control for a quadrotor UAV," *ISA transactions*, vol. 66, pp. 233-240, 2017.
- [16] O. Mofid and S. Mobayen, "Adaptive sliding mode control for finite-time stability of quadrotor UAVs with parametric uncertainties," *ISA transactions*, vol. 72, pp. 1-14, 2018.
- [17] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor UAVs: A design trade study with flight evaluations," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1400-1406, 2012.
- [18] M. Santos, V. Lopez, and F. Morata, "Intelligent fuzzy controller of a quadrotor," *2010 IEEE international conference on intelligent systems and knowledge engineering*, pp. 141-146, 2010.
- [19] B. Erginer and E. Altuğ, "Design and implementation of a hybrid fuzzy logic controller for a quadrotor VTOL vehicle," *International Journal of Control, Automation and Systems*, vol. 10, no. 1, pp. 61-70, 2012.
- [20] A. M. Hairon, H. Mansor, T. Gunawan, and S. Khan, "Travel angle control of quanser bench-top helicopter based on Quantitative Feedback Theory technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 1, no. 2, pp. 310-318, 2016.
- [21] X. Lin, Y. Yu, and C. Sun, "Supplementary Reinforcement Learning Controller Designed for Quadrotor UAVs," *IEEE Access*, vol. 7, pp. 26422-26431, 2019.
- [22] A. Zulu and S. John, "A review of control algorithms for autonomous quadrotors," *Open Journal of Applied Science*, vol. 4, pp. 547-556, 2014.
- [23] C. Trasviña-Moreno, R. Blasco, Á. Marco, R. Casas, and A. Trasviña-Castro, "Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring," *Sensors*, vol. 04, no. 17, pp. 460, 2017.
- [24] M. Bangura, "Aerodynamics and Control of Quadrotors," PhD Thesis, Australian National University, 2017.
- [25] Libelium, "Waspote LoRa 868 MHz 915 MHz SX1272 Networking Guide," 2015.
- [26] T. S. Gunawan, I. Rahmithul, H. Yaldi, M. Kartiwi, and N. Ismail, "Prototype design of smart home system using internet of things," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 7, no.1, pp. 107-115, 2017.
- [27] T. S. Gunawan, I. R. H. Yaldi, M. Kartiwi, and H. Mansor, "Performance Evaluation of Smart Home System using Internet of Things," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 1, pp. 400, 2018.
- [28] T. Bresciani, "Modelling, Identification and Control of a Quadrotor Helicopter," MSc Thesis, Lund University, 2008.