Experimental study of PID for attitude control of a quadcopter using an ESP32

Ahmed Khalil Moussaoui[1], Mohamed Habbab[1], Hazzab Abdeldjebar[1-3], Hireche Slimane[1], Ambrish Chandra[3], Hicham Gouabi[1-3], Alhachemi Moulay Abdellah[1]
[1]Laboratory of Command, Analysis and Optimization of Electro-Energetic Systems (CAOSEE), Tahri Mohamed University, Bechar, Algeria
[2]Energy Intelligence Research and Innovation Center (CR2ie) Cégep de Sept-Îles, Sept-Îles, Québec, Canada
[3]Department of Electrical Engineering, École de Technologie Supérieure, Montréal, Québec, Canada

Article Info

Article history:

Received Dec 24, 2023
Revised Apr 3, 2024
Accepted Apr 18, 2024

Keywords:

Acquisition
Communication MATLAB
ESP32
PID control
Quad-rotor

ABSTRACT

Aerial robotics encompasses intricate kinematics and dynamics that govern the flight of quad-rotor systems. Among the various methods employed for flight control using microcontrollers like the ESP32 developed by ESPRESSIF; the proportional integral derivative (PID) controller stands out as a widely adopted approach. The ESP32 microcontroller offers a superior interface, delivering enhanced performance and response time, particularly in dynamic environments. This article delves into the implementation and viability of the ESP32 platform for communication with MATLAB/Simulink, as well as real-time data acquisition to control the attitude of quadcopter withe chassis F450. The PID controller was designed to specifically work with the ESP32 platform and rigorously tested on an actual quadcopter during flight operations. lastly, a comprehensive analysis of the data gained and empirical results from the physical model demonstrates that the proposed framework is effective.

Corresponding Author:

Ahmed Khalil Moussaoui
Laboratory of Command, Analysis and Optimization of Electro-Energetic Systems (CAOSEE)
Tahri Mohamed University
B.P 417 Bechar 08000, Algeria
Email: khalilmoussaoui88@gmail.com

1. INTRODUCTION

The large-scale potential applications of drones (unmanned aerial vehicle) in the military and civilian sphere have generated increased interest in them. Amongst all the drones, quad-rotors have emerged as the most popular choice, owing to their rotor configuration that greatly simplifies their analysis and control. The obligation to employ these vehicles in potentially hazardous conditions while ensuring their reliability has revived interest in control technology.

However, quadcopters face the challenge of optimizing and fine-tuning the stability of attitude, altitude, and position through control commands or system design [1], [2]. For control, there are several methods and algorithms available for optimization, such as genetic algorithms, neural networks, control using fuzzy logic [3]-[8], and backstepping, sliding-mode control [9], [10]. Among the implemented controllers on microcontrollers like Arduino, STM32, or embedded microprocessor development boards such as Raspberry Pi [11], the proportional integral derivative (PID) controller [12]-[15] is the most commonly used. This is due to its practicality and ease of implementation based solely on the system tracking error [16], especially for quad-rotors given their configuration and control dynamics.

This article presents a demonstration of data acquisition and processing using the ESP32 microcontroller implemented on a quad-rotor drone. We also present the experimental outcomes of the PID regulator executed in MATLAB Simulink. The communication between MATLAB Simulink and the quadrotor's F450 platform is established through a UART port configured to interface with XBee (a series of communication modules). The quadrotor drone (6-DOF) incorporates MPU6050 sensors (accelerometer, gyroscope) to determine attitude stabilization during control [17]. The contribution of this project lies in establishing real-time communication between MATLAB Simulink and the drone's brain (ESP32 microcontroller) for the purposes of data acquisition, processing, and attitude control.

2. QUADROTOR

2.1. Quadcopter model

In this article, the configuration chosen for the quad rotor is the cross shape "X" Figure 1, which has four propellers driven by brushless motor. The rotation of propellers (1,3) is counterclockwise (CCW), whereas propellers (2,4) rotate clockwise (CW). The Pitch includes the movements that make the drone move forward and backward by swiveling on the y-axis. The Roll encompasses the lateral movement of the drone, which includes both left and right motion by rotation around the x-axis, while the Yaw allows a horizontal rotation on the spot around the Z axis. For the drone to move forward and backward (pitch movement) or move laterally (roll movement) it is necessary to create an inclination of the drone with respect to the horizontal. To achieve these angles, it is necessary to decrease the power of the motors in the intended direction of movement and increase the power of the two opposite motors in parallel to maintain the same altitude.

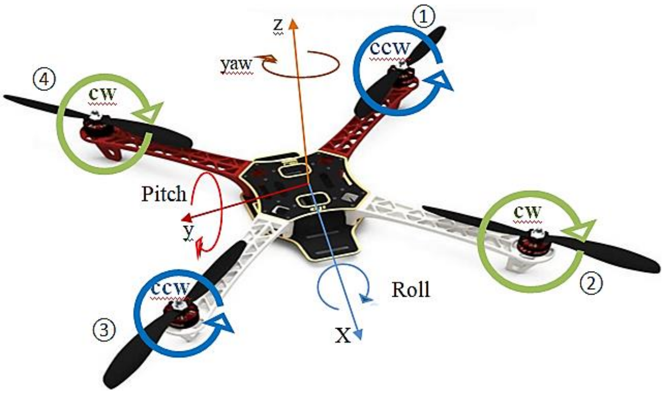

Figure 1. The quad-rotor model

2.2. The Quadrotor dynamics model

For the X model of the drone, the mathematical model can be divided into two parts, in (1) represents the translational dynamic model, while in (2) represents the rotational dynamic model [18]. In order to achieve control of the drone, the following equations were utilized these equations describe the motion of a rigid body and are derived from the Euler-Lagrange formalism [9], [18], [19]:

$$\ddot{x} = \frac{1}{m} U_1 \,(Sin\psi Sin\phi + Cos\psi Sin\theta Cos\phi)$$
$$\ddot{y} = \frac{1}{m} U_1 \,(Sin\psi Sin\theta Cos\phi - Cos\psi Sin\phi) \quad (1)$$

$$\ddot{z} = g + \frac{1}{m}(Cos\theta Cos\phi)U_1$$
$$\ddot{\phi} = \frac{(I_{yy} - I_{zz})}{I_{xx}}\dot{\psi}\dot{\theta} - \frac{J_r}{I_{xx}}(\Omega_d)\dot{\theta} + \frac{L}{I_{xx}}U_2$$
$$\ddot{\theta} = \frac{(I_{zz} - I_{xx})}{I_{yy}}\dot{\psi}\dot{\phi} - \frac{J_r}{I_{yy}}(\Omega_d)\dot{\phi} + \frac{L}{I_{yy}}U_3$$
$$\ddot{\psi} = \frac{(I_{xx} - I_{yy})}{I_{zz}}\dot{\phi}\dot{\theta} + \frac{1}{I_{zz}}U_4 \quad (2)$$

The (x, y, z) represent the center of mass's position in the inertial coordinate system[20]. Where, (ϕ, θ, ψ) correspond to the vehicle's attitude, commonly referred to as (roll, pitch, yaw), while U_1, U_2, U_3, and U_4

denote the torques responsible for controlling roll, pitch, and yaw, respectively. Additionally, m, I_{yy}, I_{xx}, and I_{zz} represent the mass and moments of inertia, J_r and Ω_d are the moments of inertia and angular velocity of the propeller blades [21], and g stands for the gravitational coefficient. Lastly, L signifies the arm length [9], [22].

3. PID CONTROL
3.1. The attitude controller
Using PID in (3) [23] control for each rotational angle is the standard quadcopter drone stabilization method [2], [5], [24]. The controllers regulate the motor speeds to correct the drone's orientation. This involves using measurements of the current angles and their corresponding rates of change, followed by a comparison with the desired target values.

$$U_i = k_P^i e_i + k_I^i \int e_i \, dt + k_D^i \dot{e}_i \text{ (i} = \phi, \theta, \psi) \quad (3)$$

Where $e_i = i_d - i$ and $\dot{e}_i = \dot{i}_d - \dot{i}$ are the error and derivative error between the desired signal and actual signal, and k_P^i , k_P^i , k_D^i are the PID gains parameter [22]. Due to mathematical simplification and stabilization, we used a PD controller instead of PID for yaw (yaw). In this project, there is no need for an altitude controller, so we utilized the throttle as a direct input. The contribution of each controller to the speed of each motor is provided in (4), where ω is a speed of the motor.

$$\begin{aligned} \omega_1 &= T + U_\phi - U_\theta - U_\psi \\ \omega_2 &= T + U_\phi + U_\theta + U_\psi \\ \omega_3 &= T - U_\phi + U_\theta - U_\psi \\ \omega_4 &= T - U_\phi - U_\theta + U_\psi \end{aligned} \quad (4)$$

4. IMPLEMENTATION
The controller-based flight control framework places high demands on computational resources, especially in the case of PID control, resulting in substantially increased hardware requirements. Therefore, the ESP32 microcontroller is responsible for communication and control developed by Espressif Systems. Small size, low weight, and controllable energy consumption make it a reasonable candidate for the proposed application.

4.1. ESP32_S2
ESP32_S2 is a series of system-on-chip (SoC) microcontroller systems, known as a celebrated platform for realizing robotics designs and projects Figure 2. The interface and characteristics of this microcontroller allow to connect with a wide variety of sensors and actuators. The ESP32-WROOM-32 Based Development board has two microprocessors Figure 3 (Tensilica Xtensa® Dual-Core 32-bit LX6, up to 600 DMIPS [25]) (32 bits each), namely core0 (“Protocol Core” or “PRO CPU") and core1 (“Application Core” or “APP CPU"), SMP (symmetric multiprocessing) which can be individually controlled in addition FreeRTOS firmware is already installed on the ESP32 board, which makes it a powerful microcontroller and therefore distinguishes itself from its predecessors. The integration of Wi-Fi + Bluetooth + BLE functionalities, targeting a wide range of applications (IoT) and embedded applications, which aims to acquire data and control various objects remotely, in particular, by adding intelligence to the system. Several development environments can be used to program the esp32 (Arduino IDE, Espressif IDF, Micropython).

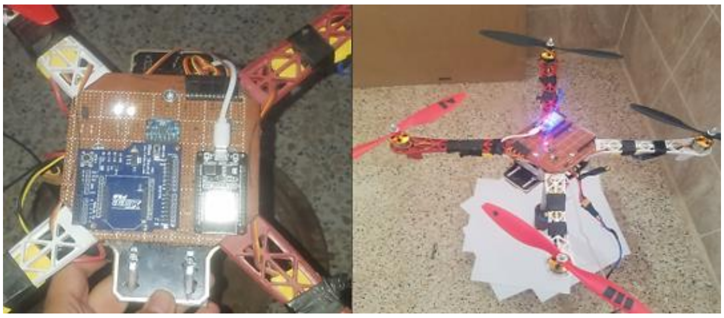

Figure 2. Experimental platform

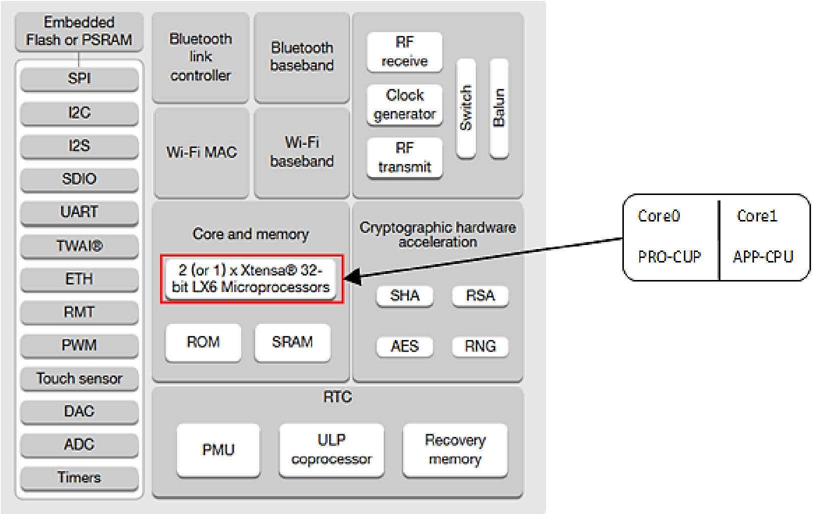

Figure 3. Function block diagram

4.2. Experimental platform

The drone developed for this project is constructed using an F450-type X quadcopter model Figure 1. It is equipped with four 1000 kW A2212 brushless motors, each featuring 1045 fixed propellers that are attached to the ends of the arms. The motors are connected to the controller (ESC) 30 A in such a way as to ensure drone movement. Each ESC has a controller (ATMEG8A) that controls the MOSFET-based rapid frequency.

It employs a switching system to manage the on/off ratio, and this, in turn, governs the power supply, thus influencing the motor speed. Electronic speed controllers (ESCs) are linked to the ESP32, to obtain input signals derived from the PC (Simulink) after processing. The whole system is powered by a battery (LiPo) that delivers 12.6 volts and has a capacity of 2200 (mAh), capable of discharging with a maximum capacity of 30C.

The block diagram of the quadcopter components, as illustrated in Figure 4, depicts the connections to the microcontroller and various instruments, as well as the signal processing workflow. The target values for (ϕ, θ, ψ) and thrust are established by the Simulink block (pc). Furthermore, the purpose of this signal is to compare it with the actual measured values obtained from the MPU6050 6DOF sensor. The MPU6050 sensor encompasses a 3-axis gyroscope and a 3-axis accelerometer, which are employed to determine the yaw, pitch, and roll attributes of the quadcopter. The ESP32 collects these measured and target signal values and transmits them to the PC for adjustments. The PC processes these signals and sends them back to the controller to handle the latest command PWM, which will subsequently be sent to the Electronic Speed Controllers (ESCs) of the drone.

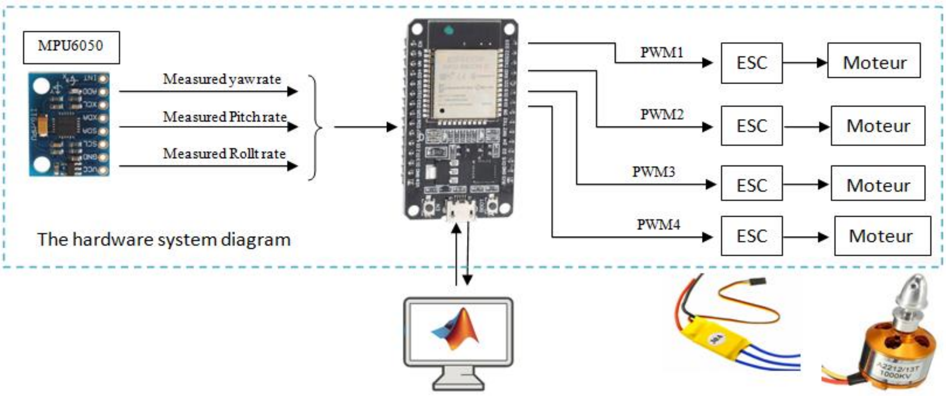

Figure 4. Block diagram of quadcopter components

4.3. Acquisition block

The acquisition diagram is implemented using the MATLAB-Simulink software. The control and processing structure is illustrated in Figure 5. The system we have created is divided into three sections:

i) Input blocks: They correspond to the system's setpoints (ch_1, ch_2, ch_3, ch_4) and play a similar role to that of an RC remote control. The reception block receives the angular velocity signals ($\dot{\phi}, \dot{\theta}, \dot{\psi}$) along with the roll and pitch angles; ii) Control block: This block is constructed based on (3) and (4), with a similar block for resetting the initial parameters during startup; iii) Output block: The PWM block converts the controller's output values into motor speed values (esc1, esc2, esc3, and esc4) to be compatible with the ESP32 microcontroller. These values are then sent via the USB port, which is configured to interface with XBee.

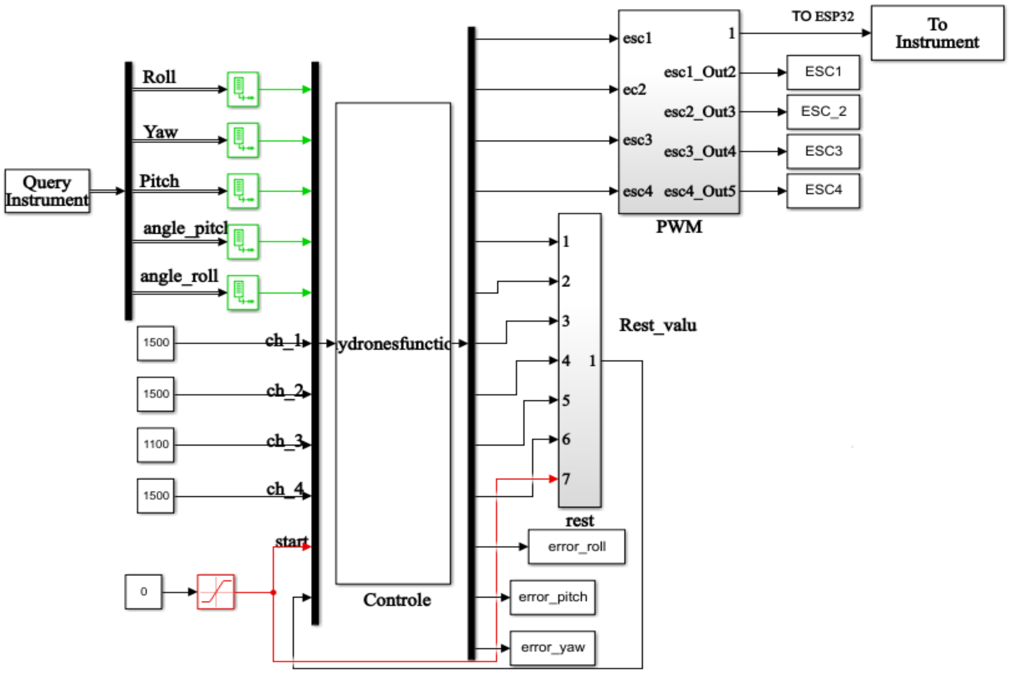

Figure 5. Block Simulink

5. RESULTS ANALYSIS

In this section, real-time acquisition and execution are performed to assess the performance of the drone using the ESP32 microcontroller and the PID controller defined in (3). Table 1 displays the PID parameters utilized in this experiment. The data was acquired using MATLAB/Simulink® R2017a with a sampling period of 5 ms. Subsequently, we evaluate the quadrotor's ability to hover with desired angles (ϕ, θ, ψ) = (0, 0, 0) degrees and control input [$ch\_{_1}$, $ch\_{_2}$, $ch\_{_3}$, $ch\_{_4}$] = [1.5, 1.5, 1.1,1.5] ms. The results are presented in Figures 6-9.

The results obtained indicate that the PID controller tends to be more reliable. The parameters of the controller (roll, pitch, yaw) are presented in Table 1. The quadcopter quickly reaches the equilibrium state, with all coordinates (roll angle, pitch angle) reaching their target values in approximately two seconds, as in Figure 6. However, there are some minor fluctuations due to propeller vibrations. Regarding motor speeds, we can observe the throttle response in Figure 8 (in Appendix). There are no dips below 1.1 ms because we implemented a saturation block to ensure continuous operation of the brushless motors and prevent them from stopping, as their operating range is between (1.2) ms. We can notice a slight increase in speed (esc1, esc2) compared to (esc3, esc4) in Figure 8 to correct the roll error in Figure 7.

The angular velocity error ($\dot{\varphi}, \dot{\theta}, \dot{\psi}$) in Figure 9 is almost zero because we are in the equilibrium position in Figure 10. This result is deemed acceptable and more reliable compared to the conclusions presented in the article [19]. There is a delay of 2 seconds before starting the flight to ensure that the initial conditions are correct at startup.

Table 1. PID parameters

	Kp	Ki	kd
Roll	1.32	0.05	18.0
Pitch	1.30	0.05	18.0
Yaw	4.11	0.03	0.0

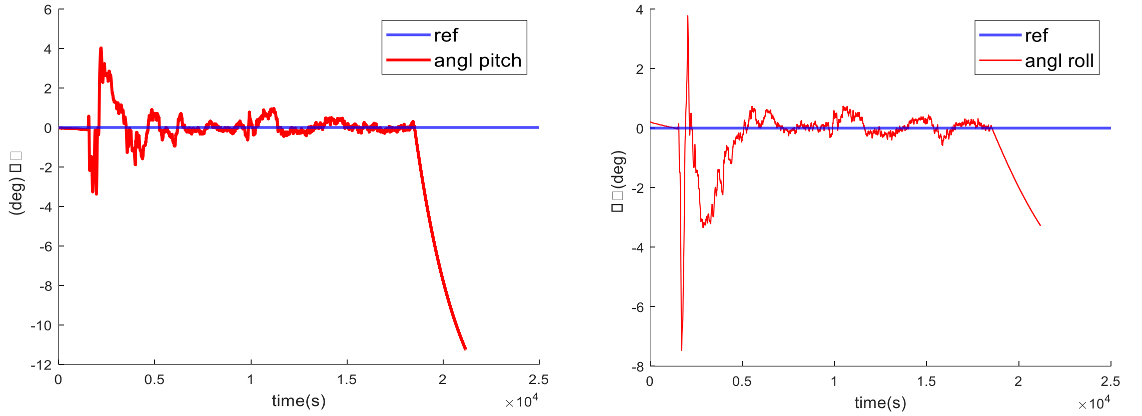

Figure 6. Angle roll, pitch

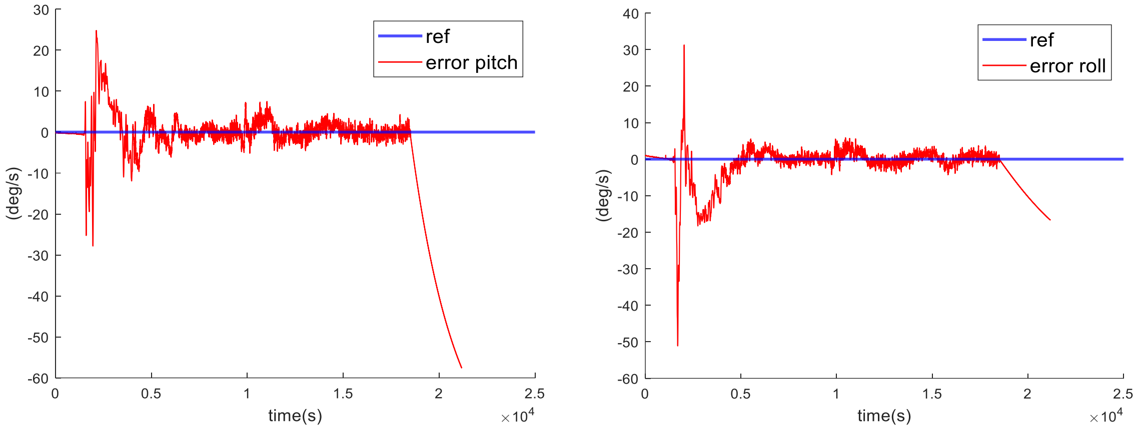

Figure 7. Error angle pitch roll

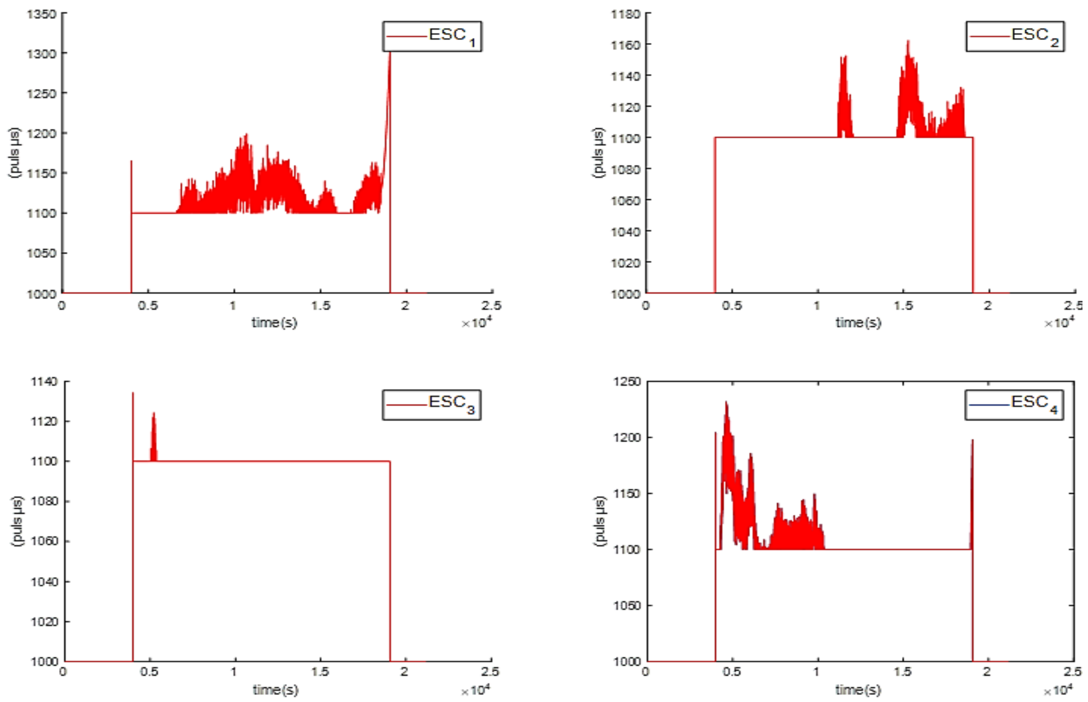

Figure 8. Speed motor

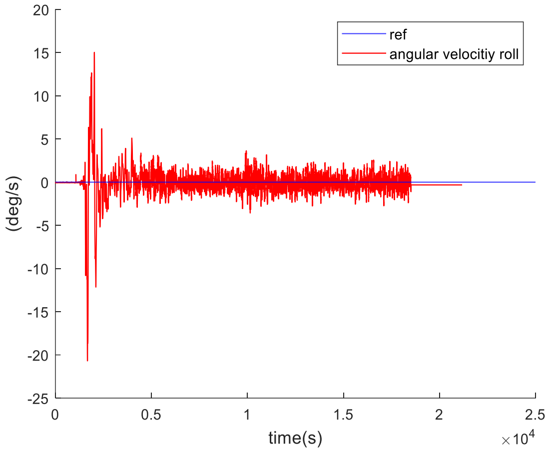

Figure 9. Angular velocity roll, pitch, yaw

Figure 10. Experiment of drone

6. CONCLUSION

The experiment's findings reveal that the responses of the attitude control coordinate Roll, Pitch, and Yaw (ϕ, θ, ψ) are satisfactory, with all coordinates reaching their desired values in approximately 2 seconds. However, angular velocities exhibit a slight fluctuation of about 2 degrees per second, which is minor and deemed acceptable. This confirms the effective and efficient performance of the system control settings and the microcontroller ESP32.

REFERENCES

[1] S. Seyedtabaii, "New flat phase margin fractional order PID design: Perturbed UAV roll control study," *Robotics and Autonomous Systems*, vol. 96, pp. 58–64, 2017, doi: 10.1016/j.robot.2017.07.003.

[2] F. Liu, H. Wang, Q. Shi, H. Wang, M. Zhang, and H. Zhao, "Comparison of an ANFIS and Fuzzy PID Control Model for Performance in a Two-Axis Inertial Stabilized Platform," *IEEE Access*, vol. 5, pp. 12951–12962, 2017, doi: 10.1109/ACCESS.2017.2723541.

[3] Y. Song, L. He, D. Zhang, J. Qian, and J. Fu, "Neuroadaptive Fault-Tolerant Control of Quadrotor UAVs: A More Affordable Solution," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 7, pp. 1975–1983, 2019, doi: 10.1109/TNNLS.2018.2876130.

[4] X. Guan, S. Lou, H. Li, and T. Tang, "Intelligent control of quad-rotor aircrafts with a STM32 microcontroller using deep neural networks," *Industrial Robot*, vol. 48, no. 5, pp. 700–709, 2021, doi: 10.1108/IR-10-2020-0239.

[5] A. Alkamachi and E. Erçelebi, "Modelling and Genetic Algorithm Based-PID Control of H-Shaped Racing Quadcopter," *Arabian Journal for Science and Engineering*, vol. 42, no. 7, pp. 2777–2786, 2017, doi: 10.1007/s13369-017-2433-2.

[6] X. Zhang *et al.*, "Compound Adaptive Fuzzy Quantized Control for Quadrotor and Its Experimental Verification," *IEEE Transactions on Cybernetics*, vol. 51, no. 3, pp. 1121–1133, 2021, doi: 10.1109/TCYB.2020.2987811.

[7] 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. 8, pp. 5044–5056, 2016, doi: 10.1109/TIE.2016.2552151.

[8] N. Maharani Raharja, E. Firmansyah, A. Imam Cahyadi, and I. Iswanto, "Hovering Control of Quadrotor Based on Fuzzy Logic," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 8, no. 1, p. 492, 2017, doi: 10.11591/ijpeds.v8.i1.pp492-504.

[9] O. Garcia, P. Ordaz, O. J. Santos-Sanchez, S. Salazar, and R. Lozano, "Backstepping and Robust Control for a Quadrotor in Outdoors Environments: An Experimental Approach," *IEEE Access*, vol. 7, pp. 40636–40648, 2019, doi: 10.1109/ACCESS.2019.2906861.

[10] H. Rios, R. Falcon, O. A. Gonzalez, and A. Dzul, "Continuous Sliding-Mode Control Strategies for Quadrotor Robust Tracking: Real-Time Application," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 2, pp. 1264–1272, 2019, doi: 10.1109/TIE.2018.2831191.

[11] A. Dorzhigulov, B. Bissengaliuly, B. F. Spencer, J. Kim, and A. P. James, "ANFIS based quadrotor drone altitude control implementation on Raspberry Pi platform," *Analog Integrated Circuits and Signal Processing*, vol. 95, no. 3, pp. 435–445, 2018, doi: 10.1007/s10470-018-1159-8.

[12] N. M. Salma and K. Osman, "Modelling and PID control system integration for quadcopter DJIF450 attitude stabilization," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1235–1244, 2020, doi: 10.11591/ijeecs.v19.i3.pp1235-1244.

[13] M. Alabsi and T. Fields, "Flight controller learning based on real-time model estimation of a quadrotor aircraft," in *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 2019, vol. 233, no. 9, pp. 3298–3312, doi: 10.1177/0954410018795524.

[14] K. P. Valavanis, "2017 International Conference on Unmanned Aircraft Systems (ICUAS'17) [Conference Reports]," 2017. doi: 10.1109/MCS.2017.2743558.

[15] S. Khatoon, I. Nasiruddin, and M. Shahid, "Design and Simulation of a Hybrid PD-ANFIS Controller for Attitude Tracking Control of a Quadrotor UAV," *Arabian Journal for Science and Engineering*, vol. 42, no. 12, pp. 5211–5229, 2017, doi: 10.1007/s13369-017-2586-z.
[16] B. R. Yenugula and M. Zia-ur-Rahman, "Stability control structure of hovercraft prototype utilising PID controller," *Bulletin of Electrical Engineering and Informatics*, vol. 6, no. 4, pp. 348–350, 2017, doi: 10.11591/eei.v6i4.864.
[17] A. Tagay, A. Omar, and M. H. Ali, "Development of control algorithm for a quadcopter," *Procedia Computer Science*, vol. 179, pp. 242–251, 2021, doi: 10.1016/j.procs.2021.01.003.
[18] A. Noordin, M. A. M. Basri, Z. Mohamed, and A. F. Z. Abidin, "Modelling and PSO fine-tuned PID control of quadrotor UAV," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 7, no. 4, pp. 1367–1373, 2017, doi: 10.18517/ijaseit.7.4.3141.
[19] M. S. M. Ashraf, M. S. Z. Abidin, M. S. A. Mahmud, M. K. I. Abd Rahman, and Z. M. N. S. Motea, "Modelling and parameters identification of a quadrotor using a custom test rig," *International Journal of Power Electronics and Drive Systems*, vol. 9, no. 2, pp. 865–872, 2018, doi: 10.11591/ijpeds.v9.i2.pp865-872.
[20] H. H. Kang, D. K. Lee, and C. K. Ahn, "Finite-Memory-Structured Online Training Algorithm for System Identification of Unmanned Aerial Vehicles With Neural Networks," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 5846–5856, 2022, doi: 10.1109/TMECH.2022.3190053.
[21] B. Xu, "Composite Learning Finite-Time Control with Application to Quadrotors," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 10, pp. 1806–1815, 2018, doi: 10.1109/TSMC.2017.2698473.
[22] A. Noordin, M. A. M. Basri, and Z. Mohamed, "Simulation and experimental study on pid control of a quadrotor MAV with perturbation," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 5, pp. 1811–1818, 2020, doi: 10.11591/eei.v9i5.2158.
[23] A. A. Najm and I. K. Ibraheem, "Nonlinear PID controller design for a 6-DOF UAV quadrotor system," *Engineering Science and Technology, an International Journal*, vol. 22, no. 4, pp. 1087–1097, Aug. 2019, doi: 10.1016/j.jestch.2019.02.005.
[24] V. D. Bui, T. T. Nguyen, N. H. Than, and D. M. Nguyen, "A suitable structure to control the system of quad-rotor miniature aerial vehicles," *International Journal of Power Electronics and Drive Systems*, vol. 9, no. 4, pp. 1634–1643, 2018, doi: 10.11591/ijpeds.v9.i4.pp1634-1643.
[25] Espressif Systems, "ESP32-S2 Series Datasheet SoC with Xtensa ® Single-Core 32-bit LX7 Microprocessor Supporting IEEE 802.11b/g/n (2.4 GHz Wi-Fi)," 2023, [Online]. Available: https://www.espressif.com/en/support/download/documents.

BIOGRAPHIES OF AUTHORS

Ahmed Khalil Moussaoui was born in Bachar, Algeria, in 1988. He got L.Sc. and M.Sc. degrees in Electrical and Electronic Engineering from the University of Bashar, Algeria, in 2010 and 2012, respectively. He has been working since September 2014 for a Ph.D. in the Department of Electrical and Electronic Engineering. He is currently a research fellow in Laboratory Control and Analysis and Electrical Power Improvement Systems, Department of Electrical and Electronic Engineering, Bechar University. Its main research activity focuses on drones, programming, and process control. He can be contacted at email: khalilmoussaoui88@gmail.com.

Mohamed Habbab was born in 1969 at Relizane-Algeria. He received the state engineer degree in Electronic Engineering in 1994 from the University of Sciences and Technology of Oran (USTO), Algeria the M.Sc. degree and the Ph.D. degree in the Electrical Engineering from University Tahri Mohamed, Bechar, Algeria in 2015 and 2018 respectively. He is currently professor of electrical engineering at University Tahri Mohamed Bechar, Algeria. His research interests include power electronics, and artificial intelligence and their applications. He can be contacted at email: habbab.mohamed@univ-bechar.dz.

Hazzab Abdeldjebar received the state engineer degree in electrical engineering in 1995 from the University of Sciences and Technology of Oran (USTO), Algeria the M.Sc. degree from the Electrical Engineering Institute of the USTO in 1999, and the Ph.D. degree from the Electrical Engineering Institute of the USTO in 2006. He is currently professor of electrical engineering at University of Bechar, Bechar, Algeria, where he is director of the Research Laboratory of Control Analysis and Optimization of the Electro-Energetic Systems, actually he is invited Professor in from École de Technologie Supérieure (ETS), Université, Montréal, Canada. His research interests include power electronics, electric drives control, and artificial intelligence and their applications. He can be contacted at email: hazzab.abdeldjebar@etsmtl.ca or abdeldjebar.hazzab@gmail.com.

Hireche Slimane was born in 1985 at Oran-Algeria. He received the electronical engineering diploma option: Control from the university of Bechar, Algeria in 2008, and the magister degree option: Automatic from the University of Bechar, Algeria in 2013. He is a Ph.D. student in the Electrical Engineering at the University of Bechar since 2014. His research interests include diagnostic automatic system, electronic, command and automatic. He can be contacted at email: etn.sliman@gmail.com.

Ambrish Chandra received B.E. degree from the University of Roorkee, India, in 1977, the M.Tech. degree from IIT Delhi, in 1980, and the Ph.D. degree from the University of Calgary, in 1987. Before joining as an associate professor at the ÉTS, Montreal, in 1994, he worked as an associate professor at IITR. Since 1999, he has been a full professor of electrical engineering with ÉTS. From 2012 to 2015, he was the director of the Multidisciplinary Graduate Program on Renewable Energy and Energy Efficiency with ÉTS. He is currently the director of the master program in electrical engineering. The primary focus of his work is related to the advancement of new theory and control algorithms for power electronic converters for power quality improvement in distribution systems and integration of renewable energy sources. He can be contacted at email: ambrish.chandra@etsmtl.ca.

Hicham Gouabi was born on the 20th November 1992 in Bechar, Algeria. He received the master degree in Automation of Industrial Processes: automatic control from the M'Hamed Bougara University, Boumerdes, Algeria in 2016. He is a member of the Control Analysis and Optimization of the Electro-Energetic Systems (CAOSEE) Laboratory since 2016. His research interests cover, power electronics, control of multilevel converters, artificial intelligence, and their applications. He can be contacted at email: gouabi.hicham@univ-bechar.dz.

Alhachemi Moulay Abdellah was born on the 2nd of January 1999 in Adrar, Algeria. He received an M.S. degree in Automation and Industrial Computing from Tahri Mohammed University in 2022. He has been a member of the Control Analysis and Optimization of the Electro-Energetic Systems (CAOSEE) Laboratory since 2023. His research interests cover, Power electronics, control of multilevel converters, artificial intelligence, and renewable energy. He can be contacted at email: alhachemi.abdellah@univ-bechar.dz.