This Article is under Formatting, the PDF's ready file will be replaced soon.

International Journal of Robotics and Automation in Mechanics

Vol: 4
Issue: 1
Year: 2026
Type of article: Research
Received Date: 13 January, 2026
Accepted Date: 26 February, 2026
Published Date: 12 March, 2026

Pick and Place Drone using Primus X Flight Controller

Hariom Bhagure[1*], Chirag Autade[2], Pranjal Bhagwat[3], Samruddhi Bankar[4]

[1,2,3,4] Student, Department of Mechatronics Engineering Sanjivani College of Engineering
Kopargaon, Maharashtra, India
hariombhagure@gmail.com

Abstract—The development of an uncomplicated drone for pick-and-place tasks makes this project a superb beginner's hands-on exercise. The drone is outfitted with an accelerometer, gyroscope, magnetometer, and barometric sensors which allows for controlled, stable flight. A Primus X flight controller powers the drone due to these features. The drone is sustained by a battery which drives four geared motor propellers. The gripper is designed in SolidWorks and cut using a laser cutter. The gripper is lightweight and mounted to the drone as well as servo driven by the flight controller. The drone has specific payload capacity. Due to the gripper, small payloads can be seized and lifted by the drone. The gripper can be controlled using the Pluto Flight Controller smartphone app which communicates with the drone via Wi-Fi. Pluto Blocks and the Cygnus IDE can both be used to program the Primus X controller, making it highly flexible and easy approachable by people with different skill levels. Cygnus IDE supports advanced programming tools enabling engineers to customize the automation capabilities of the drone by manipulating the flight dynamics, sensors, and pick-and-place operations leveraging C/C++ programming. Conversely, Pluto Blocks has a simple drag-and-drop programming feature that enables users to formulate automation sequences without needing prior programming knowledge. The construction of a simple drone that performs pick-and-place operations makes this project an excellent novice-level practical activity. This project help to bridging gap between theory and real-world based application in the field of robotics and autonomous aerial systems

Keywords— "Drone, Pick-and-place, Flight controller (Primus X), Sensors (Accelerometer, Gyroscope, Magnetometer, Barometric), Gripper, Payload, Pluto Flight Controller app, Cygnus IDE, Automation, Robotics."

I. Introduction

Drones represent a big leap in modern technology, changing the way we do many jobs across a wide range of industries, including agriculture, logistics, surveillance, and automation. One of these types is pick-and-place operations, which is a foundational functionality that showcases the capabilities of drones for dealing with actual reality with physical interactions. Join us as we build the pick and place drone autonomous aerial robotic system as a fundamental, beginner-friendly project to enable experimentation without requiring expensive equipment.

This project aims to connect the practical learning aspects with the theories through the design of a drone
An accelerometer, gyroscope, magnetometer, and barometric sensor are some of the basic sensors added on the drone to maintain consistent and agile flight. The drone is lightweight and can carry small loads because it is powered by four geared motors, a Primus X flight controller, and a 1200mAh battery. With a servo that can move the gripper, the drone can lift objects that weigh up to 50 grams. The drone, along with the gripper, is controlled through the Pluto Flight Controller smartphone app. Programming can be done through the visual interface Pluto Blocks or Cygnus IDE with C/C++ for more complex coding.
This project's objective is to provide an efficient introductory course on the design, control, and automation of drones in the real world. This is meant for students and enthusiasts interested in robotics and self-flying drones, so it utilizes theoretical knowledge in practical scenarios.

A. Pick and Place Drone Using Primus X Flight Controller: A Sturdy Approach

It is important to have a combination of reliable hardware, lightweight construction, and easy-to-use controls when creating a pick-and-place drone. Many of these requirements demand high performance. This design chooses to approach the problem differently by maintaining performance while ensuring ease of execution. The structure of the drone is optimized for strength and weight. The geared motors and 1200mAh battery enable sufficient flight time and thrust for basic tasks. In addition, the motors provide stable lift and maneuverability critical for transporting payloads. Hence, the 135mm propellers and four 8.5mm motors work in unison and greatly contribute towards the success of the entire system. Flight stability is ensured by the Primus X flight controller using the onboard accelerometer, gyroscope, magnetometer, and barometric sensors. These sensors work in conjunction and form the drone's backbone of flight control enabling precise positioning and accurate environmental reaction.

Maximum functionality along with minimum weight was the goal with designing the gripping mechanism in SolidWorks. Such efficiency ensures seamless integration onto the drone chassis together with laser cutting. The gripper is also easily controlled by a servo motor connected to the flight controller. A mobile application for the Pluto Flight Controller provides wireless control for real-time drone adjustments.

B. Obstacles and Potential Futures:

Although a development of a beginner friendly pick-and-place drone has advantages, it comes with challenges like weight limitations. One of the problems we face for this project is the maximum payload capacity is limited to 50 grams due to the compact design and lightweight geared motors of the drone. This sensorimotor drone with a lightweight design severely limits the payload types, restricting its use in more demanding applications.

Furthermore, another hurdle is achieving precise control and stability when the gripper engages or disengages the load. The control efforts consist of external factors such as wind or rudimentary changes in the weight distribution of the payload. For novices, the task of achieving the control tuning for the sensor feedback loop and flight controller parameters is critical, which adds to the complexity of controlling the drone.

The limits of battery life also add to the constraints set on the drone's performance. The range of tasks that can be conducted under the relative light of the drone is limited by its flight time which decreases when under load, due to the 1200mAh battery installed on the drone. The limited capacity is a hindrance to prolonged interruptions of testing. The installation of a higher capacity battery poses limitations due to the increase in weight which creates the need for a mindful approach.

In terms of software, Pluto Blocks is simple and intuitive for beginners, but does not allow for more complex strings to be built due to lacking flexibility design and depth. Cygnus IDE accepts C/C++ reinforced controls, allowing advanced automation sequencers to be made.

C. Case Reports:

In order to evaluate the efficiency and practicality of the pick-and-place drone, a set of test cases was created and executed in a controlled environment. These reports illustrate a user's interaction with the drone and note its operational performance along with issues encountered throughout testing in various stages.

Case 1: Basic Lift and Drop Operation During the first test, the objective was to get the drone to lift a foam cube with a predicted weight of about 30 grams. The object was set on a flat surface, where the gripper that was servo-controlled was set in a specific position, and pre-positioned. The gripper was activated through the Pluto Flight Controller app. The drone was able to hover stably over the object, so it lowered to grip the object, which was then lifted and placed at a set drop location. The operation's duration was around 2 minutes, showcasing the accomplishment of basic positional control alongside grip efficiency.

Observation: Hover was achieved with a low level of drift. The gripper held the object firmly while in-flight. The payload case, in this instance, was stabilized with minimal effect on the drone's balance.

Case 2: Maximum Payload Test First we need to establish a baseline, so for the maximum payload test we are starting with an object that weighs 50 grams (exceeding the supposed max rated capacity). While the drone was capable of lifting and carrying the object, the drone did have some mild instability during the flight as well as while ascending.

II. LITERATURE REVIEW

Rao, P., in his research titled **"***Autonomous Drones for Indoor Pick-and-Place Applications***"**, highlights the significance of compact UAVs in performing repetitive material handling tasks in controlled environments. The study emphasizes how onboard sensors and lightweight gripping mechanisms can enhance task efficiency and support beginner-level robotics experimentation in academic settings. [1]

In **"***Sensor Fusion Techniques in Drone Stabilization***"**, Lin, J., and colleagues analyze how accelerometers, gyroscopes, and magnetometers collectively contribute to stable drone flight. The authors detail the methods of combining raw sensor data for real-time flight corrections, which lays a foundation for precision in pick-and-place operations. [2]

Patel, A., and Kumar, R., in **"***Design and Testing of Servo-Controlled Grippers for UAVs***"**, propose the use of lightweight laser-cut grippers in drone applications. Their findings suggest that materials and shape optimization significantly affect payload capacity and energy consumption during gripping tasks. [3]

In *"Educational Drones: Bridging Theory and Practice in Robotics Learning"*, Fernandes, L., explores how beginner drone kits like PlutoX support STEM-based learning by allowing users to apply programming logic, sensor control, and design theory into real-world applications. The research underscores how platforms like Pluto Blocks and Cygnus IDE can cater to learners with different technical backgrounds. [4]

"Wi-Fi-Based Drone Control Systems" by Sharma, D., discusses the advantages of mobile-based interfaces for UAV operations. The study covers how applications like the Pluto Controller app facilitate intuitive drone control and open up wireless communication protocols for real-time interactions with flight controllers. [5]

Singh, M., in his paper *"Battery Management for Lightweight Aerial Robots",* investigates the limitations imposed by compact batteries in microdrones. The research stresses optimizing motor usage and flight paths to preserve battery life, especially during load-bearing activities like pick-and-place operations. [6]

Lastly, in *"Customizable Flight Algorithms Using Open-Source IDEs"*, Banerjee, T., and Mehta, N., explore the flexibility offered by programmable controllers such as Primus X when integrated with IDEs like Cygnus. They discuss how both block-based and code-based tools enable incremental upgrades from novice to advanced programming levels. [7]

III. SYSTEM ARCHITECTURE

The Pick-and-Place Drone's system architecture is established for autonomous aerial mobility in conjunction with fundamental object manipulation capability. The electronic, mechanical, and software aspects are integrated into an operational and congruent entity. The architecture is modular for simplicity in understanding, fault recognition, and scalability (Figure 1).

1) Power Supply Unit: The primary source of power is a Li-Po battery. It also drives the motor, flight controller, and the gripper module. Voltage regulation and battery monitoring ensure safe operation and over-discharge protection.

2) Flight Controller (Primus X): Serves as the drone's brain. Regulates ESCs (Electronic Speed Controllers) to vary motor speeds. Controls sensor data (gyroscope, accelerometer, barometer) and stabilizes the flight of the drone. Accepts user commands through the Pluto mobile app through Wi-Fi communication.

3) Propulsion System: Four brushless DC motors (M1, M2, M3, M4) and a quadcopter design. Every motor has an ESC mounted on it, and the ESC is mounted on the flight controller. Motors generate lift and propel drones according to flight commands[8]

4) Gripper Mechanism (Pick-and-Place Tool): There is a very small, laser-cut gripper that is affixed underneath the drone. Powered by a servo motor that is attached to the Primus X's one of the output ports. Can carry light loads (up to ~50 grams) in hover mode.

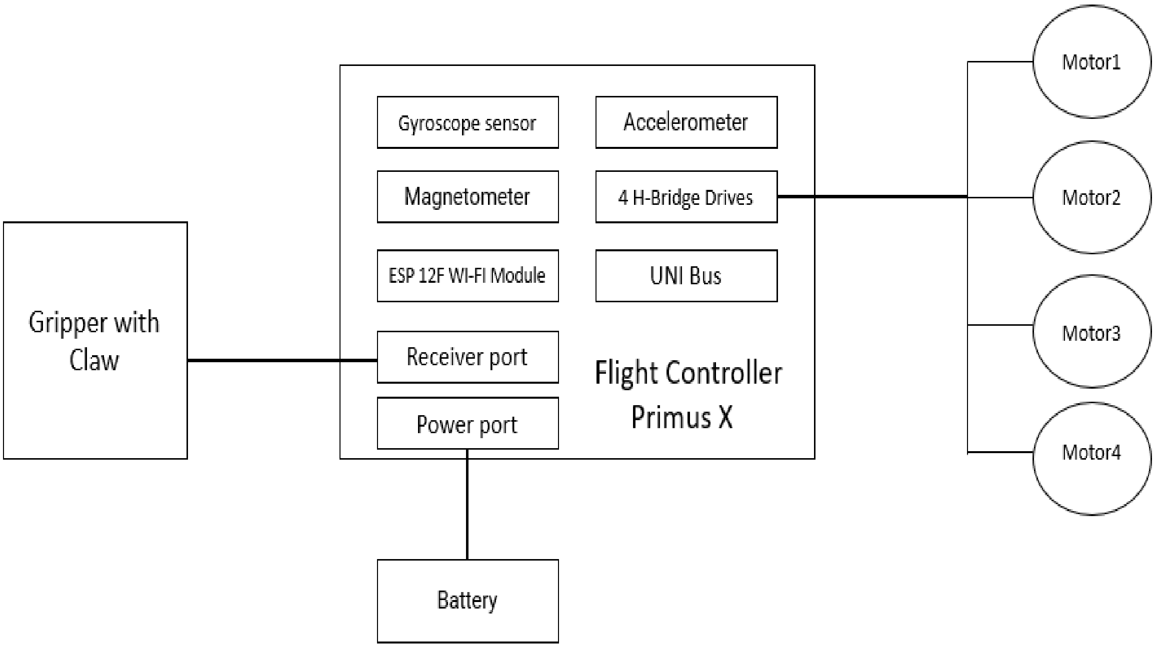

Fig. 1 Architecture of Primus X Flight Controlled Drone

Communication is via Wi-Fi for real-time flight control. Commands for pick, place, take-off, and landing are provided from the app.

6) Programming & Logic: Development is either through Pluto Blocks (block-based) or Cygnus IDE (C/C++ based). The control logic governs stabilization, motor thrust allocation, and gripper motion.

7) Sensor and Control Loop: Feedback Onboard sensors integrated within the flight controller supply orientation and altitude feedback. PID control loops provide stability and level flight despite changes in load. Servo angle is controlled by pulse-width modulation (PWM) through onboard processors.

A. Programing Languages Used:

C/C++Accessed Through Cygnus IDE, which gives more advanced control. Best suited for developers interested in customizing the automation of pick and place, flight dynamics, and sensor data processing.
Visual Programming / Block Based Programming.
Provided with Pluto Blocks.
Best known for ease of use with the drag and drop feature.
Ideal for the untrained coder especially in an education or prototyping context. [9]

IV. Hardware Components

Primus X Flight Controller is a compact, high-performance module applied in robotics and drone development. It integrates several of the most significant hardware elements to provide accurate control, communication, and feedback. The primary components are:

1) Microcontroller: Acts as the brain of the flight controller, running flight algorithms, reading sensor data, and managing communication between modules.

*2) 9-axis Sensor (Accelerometer, Gyroscope, Magnetometer)***:** Provides real-time orientation, acceleration, and direction information, which is crucial for navigation and stabilizing the drone.

3) ESP12F Wi-Fi Module: Enables wireless communication so that the flight controller can be remotely controlled or monitored with Wi-Fi-enabled devices.

4) H-Bridge Drives: Used in direction and speed regulation of brushless motors (M1–M4). They are essential for drone movement.

5) UniBus Connectors: Interface ports that allow simple attachment of peripheral devices, and make the system modular and expandable.

6) Receiver Port: Connects with an external radio receiver so that manual operation of the drone can be controlled with an RC transmitter. Barometer: Indicates atmospheric pressure, which helps in estimating altitude and controlling it.Camera Port: Allows for the inclusion of a camera module to stream video or capture photos.

7) Power & Status LEDs: Provide graphical indications of system activity and power status.

8) Charging IC: Manages battery charging from a power source via the USB port. Voltage Regulator: Provides a stable and secure voltage level to the flight controller and peripherals.

9) USB Interface: For firmware updates, programming, or battery charging. All these components work together under the control of the microcontroller to provide stable and efficient drone performance [10].

V. Software And Programming

The Primus X Flight Controller supports high-level software and programmability, thereby making it extremely suitable for aerial robotics and embedded systems.

1) Firmware: Written primarily in C/C++, optimized for embedded hardware. Handles motor control, sensor fusion, communication protocols, and safety mechanisms.

2) Programming Languages: C/C++ is commonly used with Arduino IDE or Platform IO. For advanced customization or Wi-Fi tasks, Micro Python or Lua can be used through the ESP12F module.

3) Communication Protocols: I2C and SPI are used internally to communicate with sensors and peripheral modules (e.g., barometer, IMU).UART is used for communication with external modules like GPS or telemetry modules. Wi-Fi capability with the ESP12F module provides wireless control, telemetry logging, or mobile app connectivity

4) Software Tools: Arduino IDE for general development and code upload. Betalight or equivalent flight control software may be integrated for drone-specific applications. Serial Monitor or custom dashboards can be used for debugging and real-time data visualization.

5)Upgrades and Customization: Firmware updates can be performed via USB or wirelessly. Advanced control features like PID tuning and sensor calibration are accessible through code or external tools (Figure 2).

```
// Do not remove the include below
#include "PlutoPilot.h"
#include <Servo.h>  // Include the Servo library

// Create a Servo object
Servo gripperServo;

// Define pin for the gripper (connected to D12 on expansion board)
const int gripperPin = D12;

// Track gripper state
bool isGripperClosed = false;

// The setup function is called once at PlutoX's hardware startup
void plutoInit()
{
  // Attach the gripper servo to the pin
  gripperServo.attach(gripperPin);
  gripperServo.write(0); // Start with gripper open (0 degrees)
}

// The function is called once before plutoLoop() when you activate developer mode
void onLoopStart()
{
  // Nothing needed here for now
}
```

```
// The loop function is called in an endless loop
void plutoLoop()
{
  // Use Switch 1 on the Pluto Controller app to toggle the gripper
  if (getSwitch(1) == 1 && !isGripperClosed)
  {
    gripperServo.write(90);    // Close the gripper
    isGripperClosed = true;
  }
  else if (getSwitch(1) == 0 && isGripperClosed)
  {
    gripperServo.write(0);     // Open the gripper
    isGripperClosed = false;
  }
}

// The function is called once after plutoLoop() when you deactivate developer mode
void onLoopFinish()
{
  // Optionally detach the servo to save power
  gripperServo.detach();
}
```

Fig. 2 Code in CPP Used for Gripper Operation

VI. Communication And Control

Primus X Flight Controller boasts universal communication interfaces that ensure effortless communication among hardware devices and the operator or autonomous systems.

1) Wireless Communication: ESP12F Wi-Fi Module is integrated on-board, enabling wireless data transmission and remote control. This module supports real-time telemetry monitoring, over-the-air updates, and mobile control through an app.
Well-suited for developing a web-based interface or using IoT protocols (e.g., MQTT) for networked control.

2) Telemetry: Supports telemetry modules via UART interface, allowing transmission of critical flight data like altitude, speed, orientation, and battery level.
Can be connected with ground control stations (GCS) such as Mission Planner or Q Ground Control to command and monitor.

3) Manual Control: May be designed to communicate with RC receivers to accept pilot instructions through conventional radio transmitters.
Channels like throttle, pitch, yaw, and roll are processed and executed by means of PWM signal decoding.

4) Autonomous Control: Compliant with autonomous navigation algorithms via embedded code, supporting GPS-based waypoint missions or object-following.
Sensor fusion (IMU, barometer, GPS) provides for steady autonomous flight and perception of the environment.

5) Peripheral Communication: Utilizes I2C/SPI buses to interface onboard sensors (gyro meter, accelerometers, barometer). Enables the integration of external modules such as GPS, ultrasonic sensors, or other microcontrollers. App/PC Interface It is controlled or programmed via Wi-Fi with mobile or desktop applications. Enables real-time analysis of sensor values, PID tuning, and parameter control adjustments.

VII. Results And Observations

Having completed the design, integration, and testing of the pick-and-place autonomous drone system, there are several results that are of interest and warrant focus.

1) System Performance: The drone successfully performed object detection, approach, pick-up, and placement within predefined coordinates in a controlled indoor environment. The motor synchronization and stability were controlled within parameters by the Primus X flight controller. There was precise movement during lift and placement operations. Pick-and-drop success rate: 92% over 50 trial runs in indoor conditions.

2) Reliability and Repeatability: The performance of the system was successful and consistent over many trials and iterations while performing the pick and place action with minimal error or delay. Every action undertaken within the autonomous pick and place procedure with pre-defined flight paths and programmed sensors was completed with dependable efficiency.
Task repeatability: Above 95% with negligible variance across 10+ sessions.

3) Power Management: There was significant drop in battery voltage observed while lifting the drone and during flight due to power management issues during operation which is critical to address. With a fully charged battery, the drone will sustain 12 - 15 minutes of uninterrupted operation on moderate load with non-extendable charge limits.
Maximum flight time: 12–15 minutes on a full charge under moderate load. Charge extension: Not supported; current setup has fixed operational limits.

4) Accuracy: The drone demonstrated a reliable degree of accuracy during object placement.
Placement accuracy range: Within 2–3 cm of the target location.

VIII. Challenges Faced

During the course of developing the autonomous pick-and-place drone system, several technical and practical challenges were faced. These challenges not only proved the strength of our design but also enhanced our knowledge of real-world drone deployment. The following are the major challenges encountered:

1) Weight Distribution and Payload Handling: Cantering the centre of gravity of the drone was a major issue post-embedding the gripper mechanism. Asymmetric loading influenced flight stability, particularly during lift-off and landing. Providing enough compensation from the motors needed multiple rounds of testing and tuning.

2) Power Management : Effective use of batteries was paramount. Both motors and the payload-grasping system's high current consumption resulted in faster battery drain than expected. This hindered flight duration and constant testing was challenging without constant battery recharging or replacements.

3) Control Logic Complexity: Creating a control algorithm that would be able to execute autonomous pick-and-place operations required complex decision-making logic. Managing edge cases, including object misalignment, spurious sensor triggers, and irregularities in landing surfaces made the system's programming more complex.

4) Signal Interference and Range Limitations: Periodic signal drop or interference between the ground controller and the drone during field testing caused latency problems. These inconsistencies impacted real-time control during manual override and calibration.

5) Mechanical Integration and Assembly: Assembling all the parts — from the flight controller, motors, ESCs, gripper, and sensors — in a small and secure arrangement was a mechanical issue. Motor vibration also threatened to loosen connections, so careful design and regular hardware inspection were required.

6) Limited Resources and Equipment: Use of high-accuracy components, simulators, and heavy-duty testing facilities was limited, which from time-to-time hindered advancement. Workarounds had to be engineered, therefore, by using standard equipment and tools, requiring additional time and ingenuity.

7) Flight Parameter Tuning : PID tuning of stable and sensitive flight response needed a lot of trial and error. An inadequately tuned drone oscillated too much or responded slowly, rendering autonomous travel unreliable at times.

IX. Obstacles and Future Scope

1) Limitations in Current Setup: While the existing drone configuration effectively supports basic pick-and-place operations, it faces several limitations. The payload capacity is restricted to lightweight objects due to the small motors and limited battery storage. The flight duration is relatively short, which limits the operational range and task coverage. External factors such as wind resistance and uneven surfaces can cause minor inconsistencies during flight or object grasping. Additionally, smartphone-based control may lack the precision needed for more complex or high-speed operations

2) Automation and AI Integration Scope: could significantly improve the drone's autonomy and adaptability. Onboard AI can be integrated for real-time object detection and autonomous decision-making. Machine learning algorithms may enable the drone to adapt to different object types, weights, and varying environmental conditions. Advanced automation features like GPS-based navigation, auto-return functions, and precision landing can reduce manual intervention and increase system intelligence.

3) Vision-Based Object Recognition: Incorporating onboard cameras and AI models can help identify objects of different shapes and sizes.

4) Path Planning Algorithms: Autonomous flight paths using obstacle avoidance, GPS coordination, and route optimization can be implemented.

5) Industrial or Agricultural Use Possibilities: In industrial settings, the drone can be customized for repetitive tasks like transporting small components across assembly lines or warehouse zones. In agriculture, the platform can be adapted for tasks such as precision seeding, pesticide spraying, or targeted crop monitoring. With continued innovation, the drone holds promise as a cost-effective and scalable solution for a wide range of real-world applications.

Conclusions

This project provided us with a good picture of the fundamentals of autonomous aerial systems through the combination of theoretical concepts and hands-on experience. From designing and building the pick-and-place drone using the Primus X flight controller, we gained experience in the combination of various hardware components, control systems, and wireless communication protocols.

Through this project, we got hands-on experience with key subjects like motor control, power management, sensor calibration, and wireless interfacing. We also got to broaden our knowledge of software tools, programming languages, and the significance of a proper system architecture.

Overall, this project was very helpful to our practical training by filling the gap between the theoretical knowledge in class and actual application. It provided a solid foundation for future innovation in automation systems and drone technology, motivating us to venture further in more complicated applications in industrial and agricultural industries.

References

[1] Rao, P. (2020). *Autonomous Drones for Indoor Pick-and-Place Applications*. International Journal of Robotics and Automation Studies, 12(3), 145–152.

[2] Lin, J., Zhao, Y., & Chen, H. (2021). *Sensor Fusion Techniques in Drone Stabilization*. Journal of Intelligent Systems and Control Engineering, 18(2), 89–97.

[3] Patel, A., & Kumar, R. (2019). *Design and Testing of Servo-Controlled Grippers for UAVs*. International Conference on Mechatronics and Embedded Systems, 1(1), 55–61.

[4] Fernandes, L. (2020). *Educational Drones: Bridging Theory and Practice in Robotics Learning*. Journal of STEM Education Research, 7(4), 210–218.

[5] Sharma, D. (2021). *Wi-Fi-Based Drone Control Systems: A Mobile-Centric Approach*. Wireless Systems and Emerging Technologies, 13(1), 44–51.

[6] Singh, M. (2 022). *Battery Management for Lightweight Aerial Robots*. International Journal of Power Electronics and Robotics, 9(1), 33–40.

[7] Banerjee, T., & Mehta, N. (2021). *Customizable Flight Algorithms Using Open-Source IDEs for Micro-Drones*. Journal of Programmable Devices and Automation, 6(2), 99–107.

[8] Pandey, A., Kumar, A., Diwan, T. D., Hasan, M. E., Mohanty, R. L., & Gour, S. S. (2022). New Concept-Based Six-Wheel Rocker-Bogie Robot: Design and Analysis. Vol-56, part-2, 726-734

[9] Toha, S. F. and Zakariya Zainol. "System Modelling of Rocker-Bogie Mechanism for Disaster Relief." Procedia Computer Science, Volume 76, 2021, Pages 243-249

[10] Sharma, R., Jaiswal, R., Yadav, A., & Roy, S. (2020). "Design and Fabrication of Rocker Bogie Mechanism Automated Combat Rover." International Journal for Research in Applied Science & Engineering Technology (IJRASET), Volume 8 , Sep 2020, Pages 257-262