STM JOURNALS

International Journal of
Robotics and Automation
in Mechanics

Volume 4, Issue 1, 2026
January–June
DOI (Journal): 10.37591/IJRAM

https://journals.stmjournals.com/ijram

Review IJRAM

Pick-and-Place Drone Using Primus X Flight Controller

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

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 allow for controlled, stable flight. A Primus X Flight Controller powers the drone due to these features. The drone is sustained by a battery that 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 a 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 easily 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 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 helps to bridge the gap between theory and real-world 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

INTRODUCTION

Drones represent a significant leap in modern technology, changing the way we perform 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 to the drone to maintain a 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 1200 mAh battery. With a servo that can

***Author for Correspondence**
Hariom Bhagure
E-mail: hariombhagure@gmail.com

[1]Student, Department of Mechatronics Engineering Sanjivani College of Engineering Kopargaon, Maharashtra, India

Received Date: January 13, 2026
Accepted Date: February 26, 2026
Published Date: March 12, 2026

Citation: Hariom Bhagure, Chirag Autade, Pranjal Bhagwat, Samruddhi Bankar. Pick-and-Place Drone Using Primus X Flight Controller. International Journal of Robotics and Automation in Mechanics. 2026; 4(1): 27–35p.

move the gripper, the drone can lift objects weighing up to 50 g. The drone, along with the gripper, was controlled using the Pluto Flight Controller smartphone application. Programming can be performed using the visual interface Pluto Blocks or Cygnus IDE with C/C++ for more complex coding.

The objective of this project 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; therefore, it utilizes theoretical knowledge in practical scenarios.

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 1200 mAh 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 135 mm propellers and four 8.5 mm motors work in unison and greatly contribute to 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 of designing the gripping mechanism in SolidWorks. This efficiency ensures seamless integration onto the drone chassis with laser cutting. The gripper is easily controlled by a servomotor connected to the flight controller. A mobile application for the flight controller provides wireless control for real-time drone adjustments.

Obstacles and Potential Futures

Although the development of a beginner-friendly pick-and-place drone has advantages, it comes with challenges, such as weight limitations. One of the problems we faced with this project was that the maximum payload capacity was limited to 50 g owing to the compact design and lightweight geared motors of the drone. This sensorimotor drone with a lightweight design severely limits the payload types, thereby restricting its use in more demanding applications.

Furthermore, another challenge is achieving precise control and stability when the gripper engages or disengages the load. 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 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 imposed 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 owing to the 1200 mAh battery installed on the drone. The limited capacity hinders prolonged interruptions in testing. The installation of a higher-capacity battery poses limitations owing 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 because of the lack of flexibility in design and depth. Cygnus IDE accepts C/C++ reinforced controls, allowing advanced automation sequencers to be developed.

Case Reports

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 and issues encountered throughout testing at 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 approximately 30 g. The object was set on a flat surface, where the servo-controlled gripper was set in a specific position and pre-positioned. The gripper was activated using the Pluto Flight Controller app. The drone was able to hover stably over the object; therefore, it was lowered to grip the object, which was then lifted and placed at a set drop location. The operation duration was approximately 2 min, showcasing the accomplishment of basic positional control alongside grip efficiency.
- *Observation:* Hovering was achieved with a low drift level. The gripper held the object firmly during flight. In this instance, the payload case was stabilized with minimal effect on the drone balance.
- *Case 2: Maximum payload test.* First, we need to establish a baseline; therefore, for the maximum payload test, we start with an object that weighs 50 g (exceeding the supposed maximum rated capacity). While the drone could lift and carry the object, it exhibited mild instability during the flight and while ascending.

LITERATURE REVIEW

The study highlighted the significance of compact unmanned aerial vehicles (UAVs) in performing repetitive material-handling tasks within controlled environments. It emphasized that integrating onboard sensors and lightweight gripping mechanisms can improve task efficiency while also supporting beginner-level robotics experimentation in academic settings [1].

The study examined sensor fusion techniques for achieving stable drone flight by integrating data from accelerometers, gyroscopes, and magnetometers. It highlighted the importance of combining raw sensor data for real-time flight corrections, providing a foundation for improved stability and precision in pick-and-place operations [2].

The study investigated the use of lightweight laser-cut grippers for drone applications. The findings suggested that optimizing the material and geometry of the gripper can significantly influence payload capacity and energy consumption during gripping operations [3].

The study explored how beginner-friendly drone kits, such as PlutoX, can support STEM-based learning by enabling users to apply programming logic, sensor control, and design principles to real-world applications. It also highlighted how platforms such as Pluto Blocks and Cygnus IDE can accommodate learners with varying levels of technical knowledge and experience [4].

The study examined the advantages of mobile-based interfaces for unmanned aerial vehicle (UAV) operations. It highlighted how applications such as the Pluto Controller app can facilitate intuitive drone control while enabling wireless communication protocols for real-time interaction with flight controllers[5].

The study investigated the limitations imposed by compact batteries in microdrones and emphasized the importance of efficient energy management. It highlighted the need to optimize motor usage and flight paths to conserve battery power, particularly during load-bearing activities such as pick-and-place operations [6].

The study explored the flexibility offered by programmable controllers such as Primus X when integrated with development environments such as Cygnus. It highlighted how both block-based and code-based programming tools can support a gradual transition from beginner to advanced programming, enabling incremental development and customization of flight-control algorithms [7].

SYSTEM ARCHITECTURE

The system architecture of the Pick-and-Place Drone was established for autonomous aerial mobility in conjunction with fundamental object manipulation capability. The electronic, mechanical, and software aspects were 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 power source is a Li-Po battery. It also drives the motor, flight controller, and gripper module. Voltage regulation and battery monitoring ensure safe operation and overdischarge protection.
2. *Flight controller (Primus X):* Serves as the brain of the drone. Regulates electronic speed controllers (ESCs) to vary motor speeds. It controls the sensor data (gyroscope, accelerometer, and barometer) and stabilizes the flight of the drone. It accepts user commands through the Pluto mobile app via Wi-Fi communication.
3. *Propulsion system:* Four brushless DC motors (M1, M2, M3, and M4) and a quadcopter design. Each 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):* A very small laser-cut gripper is affixed underneath the drone. The system is powered by a servo motor attached to one of the output ports of the Primus X. You can carry light loads (up to ~50 g) in hover mode.
5. *Communication interface:* The drone was wirelessly controlled using the Pluto Controller App on a mobile device. Communication is performed via Wi-Fi for real-time flight control. Commands for picking, placing, take-off, and landing were provided by 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 the load. The servo angle is controlled by pulse-width modulation (PWM) using onboard processors.

Programming Languages Used

C/C++Accessed through Cygnus IDE, which provides more advanced control. It is 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. It is best known for its ease of use with the drag-and-drop feature. This is ideal for untrained coders, especially in an education or prototyping context [9].

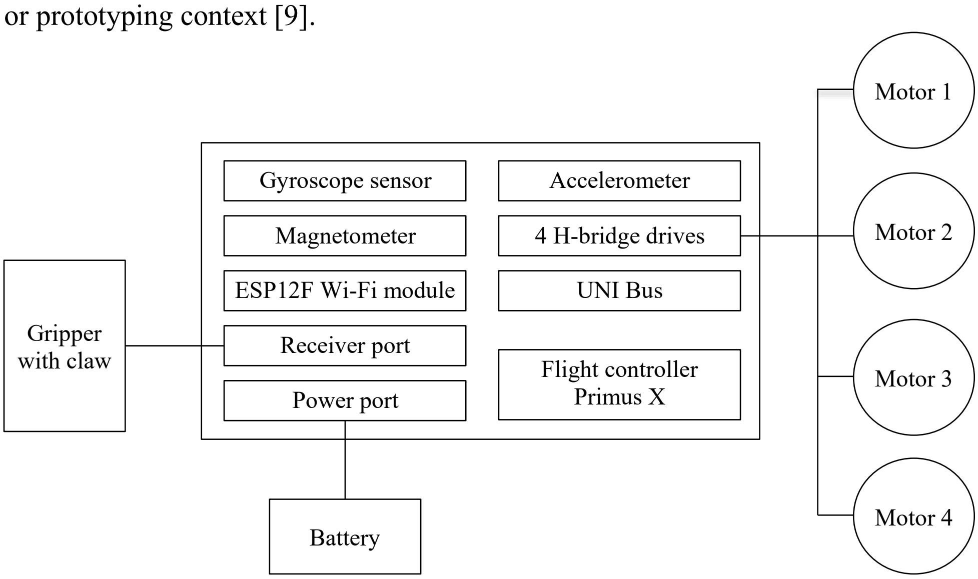

Figure 1. Architecture of Primus X flight-controlled drone.

HARDWARE COMPONENTS

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

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 using Wi-Fi-enabled devices.
4. *H-Bridge drives:* Used in the direction and speed regulation of brushless motors (M1–M4). They are essential for drone movements.
5. *UniBus connectors:* Interface ports that allow the simple attachment of peripheral devices and make the system modular and expandable.
6. *Receiver port:* Connects to an external radio receiver so that the manual operation of the drone can be controlled using an RC transmitter. *Barometer:* Indicates atmospheric pressure, which helps estimate and control the altitude. *Camera port:* Allows for the inclusion of a camera module to stream video or capture photos.
7. *Power and status LEDs:* Provide graphical indications of system activity and power status.
8. *Charging IC:* Manages battery charging from a power source through 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 a microcontroller to provide stable and efficient drone performance [10].

SOFTWARE AND PROGRAMMING

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

1. *Firmware:* Written primarily in C/C++ and optimized for embedded hardware. It handles motor control, sensor fusion, communication protocols, and safety mechanisms.
2. *Programming languages:* C/C++ is commonly used with the Arduino IDE or Platform IO. For advanced customization or Wi-Fi tasks, MicroPython 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 and IMU). UART is used for communication with external modules such as GPS or telemetry modules. Wi-Fi with the ESP12F module provides wireless control, telemetry logging, and 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. A serial Monitor or custom dashboard 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, such as PID tuning and sensor calibration, are accessible through code or external tools (Figure 2).

COMMUNICATION AND CONTROL

The Primus X Flight Controller boasts universal communication interfaces that ensure effortless communication between hardware devices and operators or autonomous systems.

Wireless Communication

The ESP12F Wi-Fi Module was integrated on board, enabling wireless data transmission and remote control. This module supports real-time telemetry monitoring, over-the-air updates, and mobile control using an application.


```
// 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();
}
```

Figure 2. Code in C++ programming language used for gripper operation.

It is well-suited for developing a web-based interface or using IoT protocols (e.g., MQTT) for networked control.

Telemetry

Supports telemetry modules via the UART interface, allowing the transmission of critical flight data, such as altitude, speed, orientation, and battery level.

It can be connected to ground control stations (GCS), such as Mission Planner or Q Ground Control, to command and monitor.

Manual Control

They may be designed to communicate with RC receivers to accept pilot instructions through conventional radio transmitters.

Channels such as throttle, pitch, yaw, and roll are processed and executed using PWM signal decoding.

Autonomous Control

Compliant with autonomous navigation algorithms via embedded code, supporting GPS-based waypoint missions or object following.

Sensor fusion (IMU, barometer, GPS) provides steady autonomous flight and perception of the environment.

Peripheral Communication

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

RESULTS AND OBSERVATIONS

After completing the design, integration, and testing of the pick-and-place autonomous drone system, several results were of interest and warranted focus.

System Performance

The drone successfully performed object detection, approach, pick-up, and placement within the pre-defined coordinates in a controlled indoor environment. Motor synchronization and stability were controlled within the parameters by the Primus X Flight Controller. The movements were precise during the lift and placement operations. *Pick-and-drop success rate:* 92% of over 50 trial runs under indoor conditions.

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.

Power Management

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

Accuracy

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

CHALLENGES FACED

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

- *Weight distribution and payload handling:* Centering the center of gravity of the drone was a major issue after embedding the gripper mechanism. Asymmetric loading influences flight stability, particularly during lift-off and landing. Providing sufficient compensation from the motors required multiple rounds of testing and tuning.
- *Power management:* Effective use of batteries is paramount. The high current consumption of both motors and the payload-grasping system resulted in a faster battery drain than expected. This hindered the flight duration, and constant testing was challenging without constant battery recharging or replacement.
- *Control logic complexity:* Creating a control algorithm that can execute autonomous pick-and-place operations requires complex decision-making logic. The management of edge cases, including object misalignment, spurious sensor triggers, and irregularities in landing surfaces, made the system's programming more complex.
- *Signal interference and range limitations:* Periodic signal drop or interference between the ground controller and drone during field testing caused latency problems. These inconsistencies affect real-time control during manual override and calibration.
- *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 threatens to loosen connections; therefore, careful design and regular hardware inspection are required.
- *Limited resources and equipment:* Use of high-accuracy components, simulators, and heavy-duty testing facilities was limited, which occasionally hindered advancement. Therefore, workarounds had to be engineered using standard equipment and tools, requiring additional time and ingenuity.

- *Flight parameter tuning:* PID tuning of stable and sensitive flight responses requires a lot of trial and error. An inadequately tuned drone oscillates excessively or responds slowly, rendering autonomous travel unreliable at times.

OBSTACLES AND FUTURE SCOPE

- *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 because of the small motors and limited battery storage. The flight duration is relatively short, which limits the operational range and task coverage of the aircraft. 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 required for more complex or high-speed operations.
- *Automation and AI integration scope:* can 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 drones to adapt to different object types, weights, and varying environmental conditions. Advanced automation features, such as GPS-based navigation, auto-return functions, and precision landing, can reduce manual intervention and increase system intelligence.
- *Vision-based object recognition:* Incorporating onboard cameras and AI models can help identify objects of different shapes and sizes.
- *Path planning algorithms:* Autonomous flight paths using obstacle avoidance, GPS coordination, and route optimization can be implemented.
- *Industrial or agricultural use possibilities:* In industrial settings, drones can be customized for repetitive tasks, such as transporting small components across assembly lines or warehouse zones. In agriculture, the platform can be adapted for tasks such as precision seeding, pesticide spraying, and targeted crop monitoring. With continued innovation, drones hold promise as a cost-effective and scalable solution for a wide range of real-world applications.

CONCLUSIONS

This project provided us with a good understanding of the fundamentals of autonomous aerial systems through a combination of theoretical concepts and hands-on experience. By 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 gained hands-on experience with key subjects such as motor control, power management, sensor calibration, and wireless interfacing. We also broadened our knowledge of software tools, programming languages, and the significance of proper system architecture.

Overall, this project was very helpful for our practical training by bridging the gap between theoretical knowledge in class and actual application. This provides a solid foundation for future innovations in automation systems and drone technology, motivating us to venture further into more complicated applications in the industrial and agricultural sectors.

REFERENCES

1. Proia S, Cavone G, Camposeo A, Ceglie F, Carli R, Dotoli M. Safe and ergonomic human-drone interaction in warehouses. 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan. 2022. p. 6681-6686. doi:10.1109/IROS47612.2022.9981469.
2. Guo H, Chen X, Yu M, Uradziński M, Cheng L. The usefulness of sensor fusion for unmanned aerial vehicle indoor positioning. Int J Intell Unmanned Syst. 2024;12(1):1-18. doi:10.1108/IJIUS-01-2023-0006.
3. Abdulganiyu A, Babawuya A, Aliyu Y, Oguntowo BI. Mountable drone mechanism: A sustainable approach. J Mater Environ Sci. 2025;16(5):894-908.

4. Ren P, Terpenny J, Hong D, Goff R. Bridging theory and practice in a senior level robotics course for mechanical and electrical engineers. In: Proceedings of the 2009 ASEE Annual Conference & Exposition; 2009 Jun; Austin, TX, USA. Washington (DC): ASEE Conferences; 2009. p. 14.291.1-14.291.15. doi:10.18260/1-2--4915.
5. Chi G, Yang Z, Xu J, Wu C, Zhang J, Liang J, et al. Wi-Drone: Wi-Fi-based 6-DoF tracking for indoor drone flight control. In: Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '22); 2022 Jun; Portland, OR, USA. New York (NY): Association for Computing Machinery; 2022. p. 56-68. doi:10.1145/3498361.3538936.
6. Wang H, Kong L, Fang Z, Zou R, Zhang Z, Zhao X, et al. Symbiotic energy paradigm for self-sustaining aerial robots. Nat Rev Electr Eng. 2025;2(5):302-319. doi:10.1038/s44287-025-00168-4.
7. Shekh M, Rani S, Datta R. Review on design, development, and implementation of an unmanned aerial vehicle for various applications. Int J Intell Robot Appl. 2025;9(1):299-318. doi:10.1007/s41315-024-00359-6.
8. Pandey A, Kumar A, Dhar Diwan T, Ehtesham Hasan M, Lochan Mohanty R, Singh Gour S. New concept-based six-wheels rocker-bogie robot: Design and analysis. Mater Today Proc. 2022;56:726–734. doi:10.1016/j.matpr.2022.02.243.
9. Toha SF, Zainol Z. System modelling of rocker-bogie mechanism for disaster relief. Procedia Comput Sci. 2015;76:243–249. doi:10.1016/j.procs.2015.12.349.
10. Sharma R. Design and fabrication of rocker bogie mechanism automated combat rover. Int J Res Appl Sci Eng Technol. 2020;8:257–262. doi:10.22214/ijraset.2020.31390.

