



INTER IIT  
TECH MEET 14.0

LOW PREP



DRONA<sup>TM</sup>  
AVIATION

Control system for Drone Stabilization  
and Precision Motion



# Introduction

Indoor nano-drones operate with strict sensing and compute constraints. Without GPS or external tracking, stability depends entirely on how effectively the firmware interprets onboard sensors and closes the loop in real time. Even with reliable IMU data, a drone will drift or vary in altitude unless supported by a strong, fused control pipeline.

Modern systems achieve this by combining optical flow based motion, ToF altitude measurements, and IMU orientation into a consistent, low-latency estimate of the drone's state. This enables the aircraft to take off, lock its position, and maintain a steady hover autonomously.

In this challenge, the drone must hold its position and altitude by default, without any pilot input. Only when the user provides a command should it move, shifting precisely 10-20 cm in the indicated direction before stabilizing again at the new location. The focus is on delivering a robust, production-grade control system capable of clean sensor fusion, predictable behavior, and centimeter-level precision in real indoor conditions.

# About Us

Drona Aviation Pvt. Ltd., led by IIT Bombay alumnus Mr. Dinesh Sain, is a leading force in bringing drone technology into India's education landscape. We bring cutting-edge aerial robotics into classrooms through programmable drones, modular learning kits, and hands-on technical workshops designed for practical, experiential STEM learning. As India's leading programmable drone platform with 50,000+ drones deployed globally, our Made-in-India Pluto 1.2 and Pluto X have become trusted tools for learning drone flying, coding, tinkering, prototyping, and real-world innovation. These classroom-safe, curriculum-ready platforms are designed to build future-ready technical skills. Our nationwide impact includes the Atal Drone Module officially adopted by 8,000+ Atal Tinkering Labs under NITI Aayog, bridging academic learning with real drone applications and empowering the young generation of innovators across India.



# Problem Statement Description

## Your Mission

Build a complete, on-board stabilization and precision-motion system that can:

- Take off on user command (single button trigger)
- Hold position and altitude without any further user input
- Maintain a stable hover with minimal drift
- Execute user-triggered micro-movements (10-20 cm) based on pitch, roll, or throttle input
- Stabilize immediately after each commanded displacement
- Maintain altitude up to 2 meters with consistent behavior
- Resist and recover from external disturbances
- Deliver predictable, smooth control responses
- Filter noise and handle poor sensor conditions gracefully
- Demonstrate stable, repeatable, production-grade performance across multiple flights

Your implementation must run entirely within MagisV2 using only the provided APIs, sensors, and microcontroller compute.

## What You Need to Build

### 1. Real-Time Velocity Estimation:

- a. You must compute accurate horizontal velocities by:
  - i. Converting optical-flow pixel shifts into real-world motion using real-time altitude data
  - ii. Adjusting velocity estimates for yaw changes to maintain a stable world-frame reference
  - iii. Handling low-texture or low-light situations using fallback logic and confidence checks
  - iv. Maintaining high-frequency updates to support stable hover and precise micro-movements



- b. These velocity estimates directly drive:
  - i. Position locking during hover
  - ii. Controlled 10-20 cm user-triggered micro-displacements
  - iii. Smooth deceleration and settling at the new position
  - iv. Overall stability and drift minimization

## 2. Closed-Loop Control System

- a. The drone must be capable of:
  - i. Automatically stabilizing and entering hover mode once airborne
  - ii. Maintaining a fixed X-Y position and altitude without pilot stick input
  - iii. Executing precise 10-20 cm movements when the user applies pitch, roll, or throttle commands
  - iv. Stabilizing immediately at the new position after each micro-movement
  - v. Recovering cleanly from external disturbances (pushes or pulls)
  - vi. Minimizing drift during prolonged hover
  - vii. Responding to user inputs smoothly, without overshoot or oscillation
- b. A cascaded control architecture (Position  $\rightarrow$  Velocity  $\rightarrow$  Attitude) is expected, though teams may innovate beyond PID if properly justified.
- c. The controller must ensure:
  - i. Predictable transitions between hover and movement
  - ii. Stable behavior across different textures, lighting, and altitudes
  - iii. Clean re-entry into hover mode after each commanded displacement
  - iv. Deterministic, low-noise control actions suitable for production firmware

## 3. Sensor Fusion Under Real Conditions

- a. You must combine Optical Flow, ToF, IMU and Baro data using reliable and explainable logic such as:
  - i. Complementary filtering
  - ii. Kalman-style filters
  - iii. Adaptive gain fusion
  - iv. Any alternative approach that maintains deterministic behavior



- b. The fusion system must remain stable and deliver clean state estimates during:
  - i. Low-texture or uneven surfaces
  - ii. Rapid or uneven lighting changes
  - iii. Optical flow dropouts, spikes, or low-confidence readings
  - iv. ToF noise at higher altitudes
  - v. Slow yaw rotations during hover or micro-movements
  - vi. Micro-vibrations caused by motors or sudden directional shifts
- c. The fused state must support:
  - i. Accurate hover without user input
  - ii. Precise 10-20 cm user-triggered displacement commands
  - iii. Fast recovery back into stable hover after motion
  - iv. Smooth transitions even when sensor quality temporarily degrades
- d. Failure cases must degrade gracefully and should not cause divergence, oscillations, or unexpected drift.

#### **4. Altitude Stability up to 2 Meters**

- a. Your altitude controller must demonstrate:
  - i. Stable hover Minimum 0.5 Meters for minimum one minute
  - ii. Low oscillations
  - iii. Noise filtering
  - iv. Consistent behavior across heights
- b. A special evaluation includes a high-altitude consistency test up where stability, drift, and control authority are examined.
- c. No surface, lighting, or texture biases are allowed.

#### **5. Precision Motion (User-Triggered Micro-Movement Mode)**

##### **a. Autonomous Hover by Default**

- After takeoff, the drone must independently stabilize and hold its position and altitude without any user input.

##### **b. Hands-Free Position Lock**

- The system must maintain a drift-free hover until a user explicitly generates a control input.



**c. Input as a Displacement Command, Not Velocity**

- Any stick movement (pitch/roll/throttle) must translate into a discrete 10-20 cm movement, not continuous motion tied to stick holding.

**d. Directional Micro-Movement**

- Example behaviors:
  - Pitch forward → move forward 10-20 cm
  - Roll left → move left 10-20 cm
  - Throttle up → climb 10-20 cm

**e. Adaptive Mapping**

- Stick deflection may scale the displacement within the 10-20 cm range while retaining full control stability.

**f. Smooth Motion Profile**

- Each micro-movement must include controlled acceleration and deceleration with minimal jerk.

**g. No Overshoot**

- The drone must stop precisely at the commanded displacement with  $\pm 2$  cm tolerance.

**h. Instant Hover at New Position**

- After reaching the target point, the drone must automatically return to hover mode and lock the new X-Y-Z position.

**i. Robust Against Tiny Inputs**

- Small, accidental bumps or noise on the sticks must be rejected; only intentional inputs should trigger motion.

**6. Your design should reflect:**

- a. Deterministic and predictable behavior in both hover and micro-movement modes
- b. Clean transitions between user-triggered 10-20 cm movements and stable hover
- c. Low oscillations, minimal drift, and smooth control authority





# INTER IIT TECH MEET 14.0

- d. Modular, maintainable architecture suitable for real firmware integration
- e. Clearly tuned control and fusion parameters with documented reasoning
- f. Reliable performance across repeated flights, different indoor environments, and varied textures
- g. Stable behavior even under partial sensor degradation or brief dropouts
- h. Consistency across the entire altitude range up to 2 meters
- i. Robust fail-safe handling without unexpected drift or runaway motion
- j. Firmware quality that meets the standards required for production release within MagisV2

## Deliverables

### 1. Firmware Implementation

A complete, modular codebase running inside MagisV2 that includes:

- i. Sensor reading and filtering for Optical Flow, ToF, and IMU
- ii. Real-time velocity and position estimation for stable hover
- iii. Position-velocity-attitude control loops
- iv. Auto-hover after takeoff
- v. User-triggered micro-movement handling
- vi. Smooth motion execution and stable settling
- vii. Tunable control and fusion parameters
- viii. Robust fallback behavior under poor sensor conditions

### 2. Documentation

A clear technical document covering:

- i. System architecture overview
- ii. Control design and reasoning for hover and micro-movements
- iii. Sensor fusion strategy for OF, ToF, and IMU
- iv. PID tuning approach and key parameters
- v. Hover and micro-movement performance results
- vi. Limitations, failure cases, and lessons learned



### **3. Demonstration Video**

Showcase:

- i. Takeoff using the designated button
- ii. Stable hover with no user stick input
- iii. Horizontal drift within acceptable limits
- iv. Altitude stability Up-to 2 meter
- v. Recovery from small external disturbances
- vi. User-triggered 10-20 cm movements in all directions
- vii. Immediate stabilization at each new position and altitude

Dashboard overlays (velocity, ToF, OF) are encouraged.

## **Evaluation Criteria**

### **1. Stability & Precision (40%)**

- a. Hover stability without user input
- b. Minimal drift during extended hover
- c. Smooth execution of 10-20 cm user-triggered movements
- d. Accurate settling at the new position and altitude
- e. Low oscillations and predictable corrections

### **2. Control System Design (25%)**

- a. Correct implementation of a cascaded Position-Velocity-Attitude architecture
- b. Clean transition between hover mode and micro-movement execution
- c. Well-tuned control gains with stable, non-oscillatory response
- d. Predictable behavior during acceleration, deceleration, and settling
- e. Clear justification of control strategies and parameter choices

### **3. Sensor Fusion Quality (20%)**

- a. Clean and reliable fusion of Optical Flow, ToF, and IMU data
- b. Stable estimates during hover and micro-movement execution
- c. Robust handling of dropouts, spikes, and low-confidence readings
- d. Smooth behavior across varying textures, lighting, and altitudes
- e. Consistent state estimation supporting precise 10-20 cm movements





# INTER IIT TECH MEET 14.0

## **4. Engineering Innovation (10%)**

- a. Advanced filtering or sensor fusion approaches
- b. Efficient or optimized control strategies
- c. Useful visualization or debugging tools
- d. Adaptive or predictive features that enhance stability
- e. Creative but practical solutions that improve reliability

## **5. Documentation & Presentation (5%)**

- a. Clear and well-structured technical documentation
- b. Easy-to-understand explanation of design decisions
- c. Concise presentation of results and performance
- d. Effective use of visuals, graphs, or overlays
- e. Overall clarity and completeness of the submission