Chapter Outline

Components and Architecture: Tasks, Components, and Operation

Chapter 1 Introduction

1.1 Concepts and Applications

1.2 History and Trends

1.3 Components and Architecture

1.4 Coordinate Systems and Control Principles

1.5 Assembly and Debugging

1.6 RflySim Toolchain

1.7 Learning Path and Experiments

1.8 Exercises and Discussion

3 Units in This Section

1.3.1 Overall Framework

Textbook: Sec. 1.2.1

1.3.2 Composition and Design Principles

Textbook: Sec. 1.2.4

1.3.3 Intelligent Control Architecture

Textbook: Sec. 1.2.5

Mission Scenario

Three System Views of an Inspection Mission

- **Mission:** Approach a designated inspection point, capture clear images, and adjust the route for new obstacles.
- **Components:** A camera captures images; the autopilot and onboard computer process data; propulsion components produce motion.
- **Architecture and feedback:** Trace information and energy; observe the resulting motion and correct the next action.

Mission Functions	Physical Components	Program Operation
Observe, decide, execute	Sense, compute, drive	Continuous targets and feedback
See Unit 1.3.1	See Unit 1.3.2	See Unit 1.3.3

Components identify the parts; architecture explains cooperation; feedback corrects actions.

1.3 System Components and Architecture

1.3.1 Overall Framework

Unified System Framework

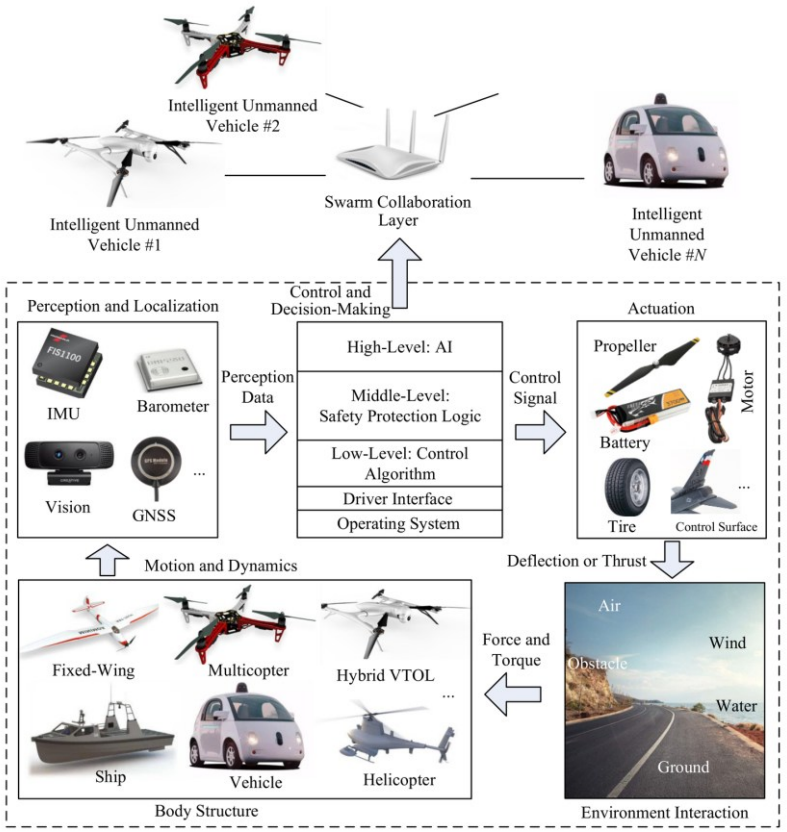


Fig. 1.6 Unified Framework

Systems classified by medium (underwater, surface, ground) or platform (multicopters, robot dogs, other robots) share six internal layers: perception, control, actuation, environment, structure, and swarm.

- **Observe and localize:** The camera observes inspection points and obstacles; localization estimates pose. Together they define mission state.
- **Set targets and act:** Planning selects an approach route; control tracks motion targets; actuators move the vehicle.
- **Correct and replan:** Position error changes control output. Replan if a new obstacle invalidates the route.
- **Coordinate vehicles:** Share potential-defect locations and progress. Allocation changes targets; each vehicle retains its own control loop.

A unified framework supports sharing data, algorithms, and components for greater efficiency

1.3.1 Overall Framework

Six Layers and Their Responsibilities

- **Perception and localization:** Measure the vehicle and its surroundings; pass pose, obstacles, and other perception data to decision-making.
- **Control and decision-making:** Plan targets, handle constraints and faults, and send control signals to actuation.
- **Actuation and execution:** Motors and servos follow commands, acting on the environment through rotation, deflection, or jets.
- **Environment interaction:** Air, water, and ground produce forces and torques; wind, electromagnetic effects, and obstacles create disturbances.
- **Body structure:** Mount components and transmit loads; perception and localization measure the resulting motion and dynamics.
- **Swarm collaboration:** Exchange vehicle states, allocate tasks, and coordinate paths and behavior.

The six layers describe the vehicle–environment feedback loop and extend the analysis to multiple vehicles.

1.3.1 Overall Framework: Perception and Localization

Turning Measurements into Usable State Information

- **Measurement and estimation:** Images, point clouds, and inertial data are inputs; localization estimates position, velocity, and attitude.
- **Inspection information:** Identify inspection points, obstacles, relative positions, and traversable space for route planning.
- **Data meaning:** Positions need a frame; observations need timestamps. Compare data in the same time and frame.
- **Quality:** Occlusion or degraded localization makes estimates unreliable; limit speed or change viewpoint according to estimation quality.



Diagram: Measurements to State and Map

Perception turns raw measurements into state and environment information; coordinates, time, and quality determine usability.

1.3.1 Overall Framework: Control and Decision-Making

Mission Planning and Actuator Commands

- **Mission planning:** Choose an approach route within obstacle and endurance limits; set position or velocity targets.
- **Feedback control:** Compare targets and actual state; compute corrective forces and torques.
- **Control allocation:** Map total demands to motor or servo commands using vehicle configuration and actuator limits.
- **Inspection loop:** Correct small errors; generate new high-level targets if an obstacle invalidates the route.
- **Safety protection:** Monitor battery, localization, and communication; on faults, hover, return, or land.



Planning sets motion targets; control corrects state errors; allocation converts control demands into actuator commands.

1.3.1 Overall Framework: Actuation and Execution

From Commands to Vehicle Motion

- **Information:** Commands specify thrust, speed, or angle; they do not supply the energy that drives motors.
- **Energy:** The battery supplies power, electronic speed controllers (ESCs) regulate current, and motors act on the environment through propellers or transmissions.
- **Motion:** Forces at different locations jointly change translation and rotation; sensors observe the result again.
- **Output limits:** Beyond current, speed, or travel limits, increasing a command produces no additional control effort.



Commands regulate energy conversion; actuators produce forces and torques within current and load limits.

1.3.1 Overall Framework: Body Structure

Component Mounting, Load Transfer, and Measurement Quality

- **Structure:** The body, arms, and landing gear provide mounting locations and transmit loads and driving forces.
- **Measurements:** The inertial measurement unit (IMU) measures acceleration and angular rate; frame vibration adds to its measurements.
- **Aerodynamics:** Body shape and surface affect aerodynamic forces; weight reduction and aerodynamic efficiency both affect motion.
- **Contact:** Tires, ground, and normal load determine grip, limiting acceleration, turning, and braking.

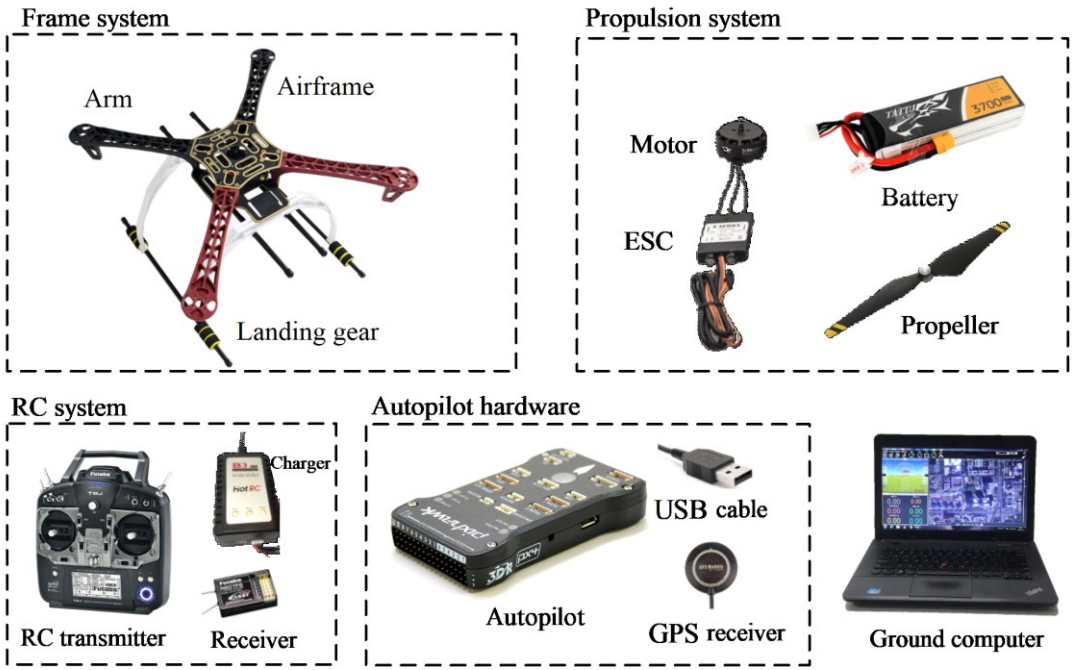


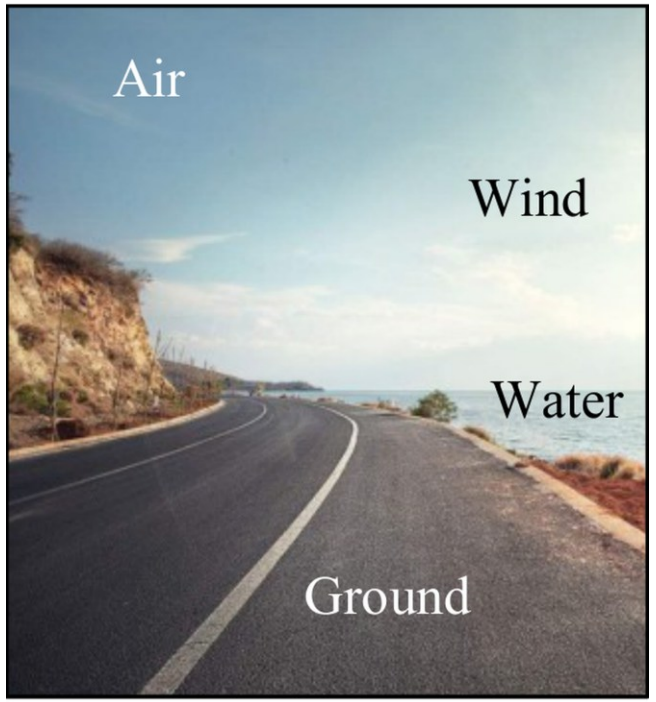
Photo: Frame and associated components

Structural design affects load support, measurement quality, and motion performance.

1.3.1 Overall Framework: Environment Interaction

Environmental Effects on Forces and Observations

- **Air and weather:** Wind and air density affect aerodynamic forces; rain, fog, and temperature affect equipment and sensing.
- **Ground contact:** Friction enables driving and braking; slips and collisions change motion.
- **Electromagnetic conditions:** Geomagnetic anomalies, motors, and power wiring affect measurements; radio interference affects links.
- **Obstacles and lighting:** Sudden obstacles require replanning; occlusion and lighting changes may degrade visual perception.



The environment changes both actual motion and what sensors observe.

1.3.1 Overall Framework: Swarm Collaboration

Task Handoffs and State Sharing Across Vehicles

- **Task handoff:** A survey vehicle sends a potential defect's location, type, and time. A follow-up vehicle accepts the inspection task if capable.
- **State sharing:** Vehicles report position, battery, and progress; a shared list distinguishes completed and pending items.
- **Coordination:** Task allocation sets individual targets; path coordination reduces conflicts. Each vehicle still runs its own fast feedback control.
- **Fault response:** Reassign tasks based on remaining payloads and energy after a node leaves. Expired messages cannot represent current state.

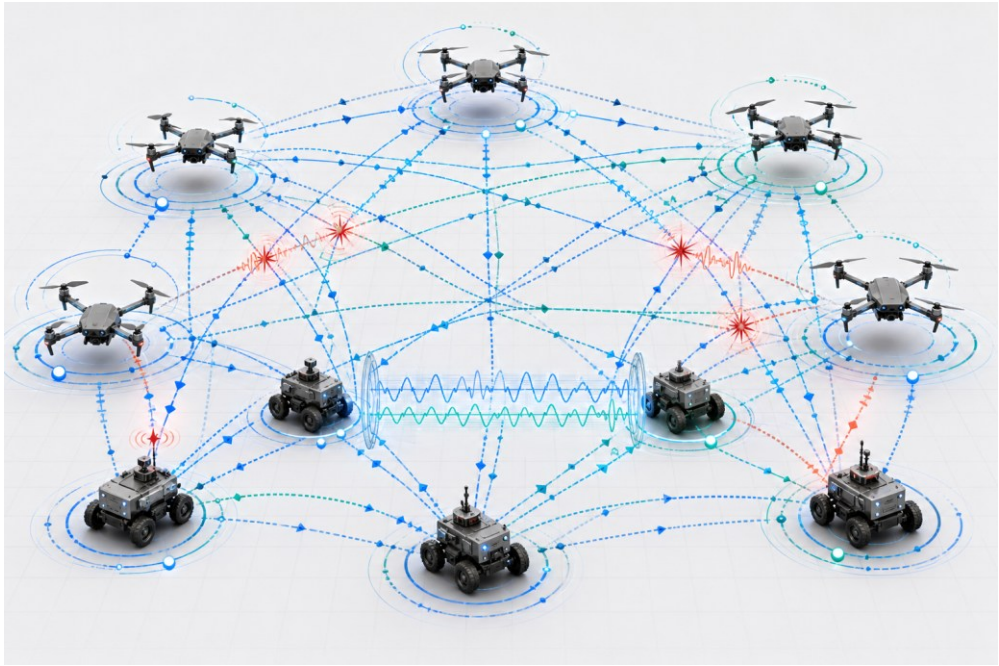


Diagram: Node Links and Coordination

Shared state enables task allocation; after a node leaves, reorganize the mission around the remaining capabilities.

Chapter Outline

Components and Architecture: Tasks, Components, and Operation

Chapter 1 Introduction

1.1 Concepts and Applications

1.2 History and Trends

1.3 Components and Architecture

1.4 Coordinate Systems and Control Principles

1.5 Assembly and Debugging

1.6 RflySim Toolchain

1.7 Learning Path and Experiments

1.8 Exercises and Discussion

3 Units in This Section

1.3.1 Overall Framework

Textbook: Sec. 1.2.1

1.3.2 Composition and Design Principles

Textbook: Sec. 1.2.4

1.3.3 Intelligent Control Architecture

Textbook: Sec. 1.2.5

1.3.2 Composition and Design Principles

Multicopter Exploded View: Components and Connections

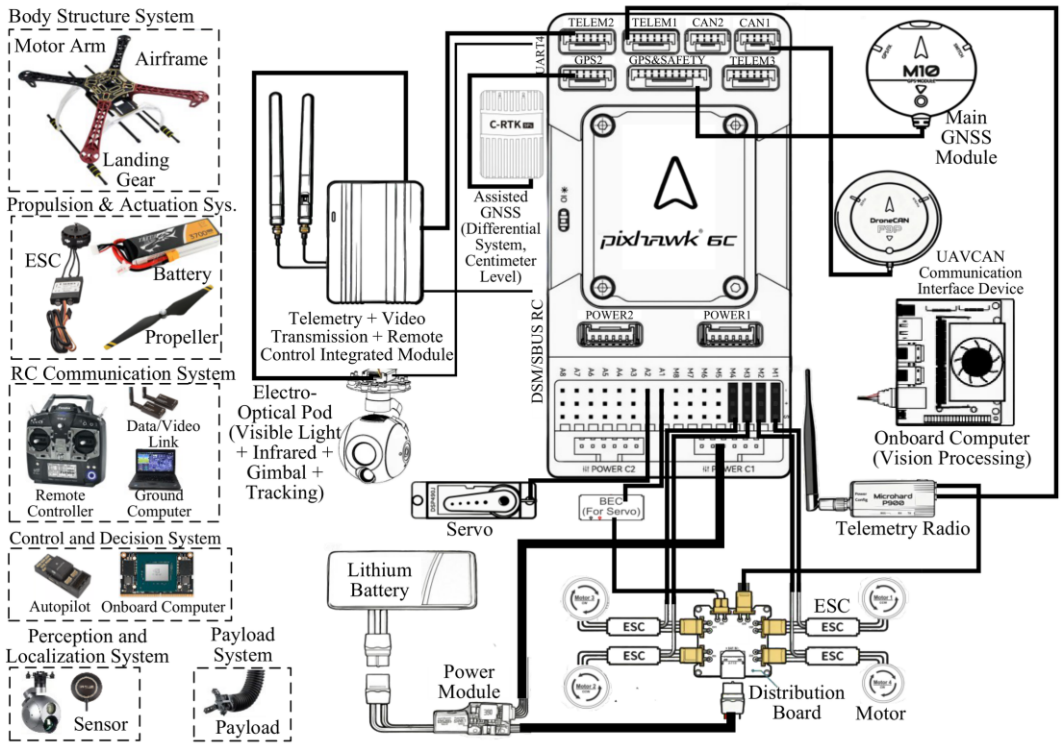


Fig. 1.13 Multicopter Assembly

- **Component groups:** Six subsystems cover structure, perception, control, actuation, communication, and payload.
- **Propulsion power:** Battery power passes through distribution wiring and ESCs to motor windings.
- **Control power:** A power module or similar device supplies the autopilot's required voltage separately from motor power.
- **Control signals:** The autopilot sends commands through signal interfaces; these carry no motor power current.
- **Connections:** Supply voltage and polarity determine correct power-up; interface pinouts and protocols determine correct signal reception.

1.3.2 Composition and Design Principles

Fig. 1.13 Detail: Motor Power, Regulated Supply, and Signals

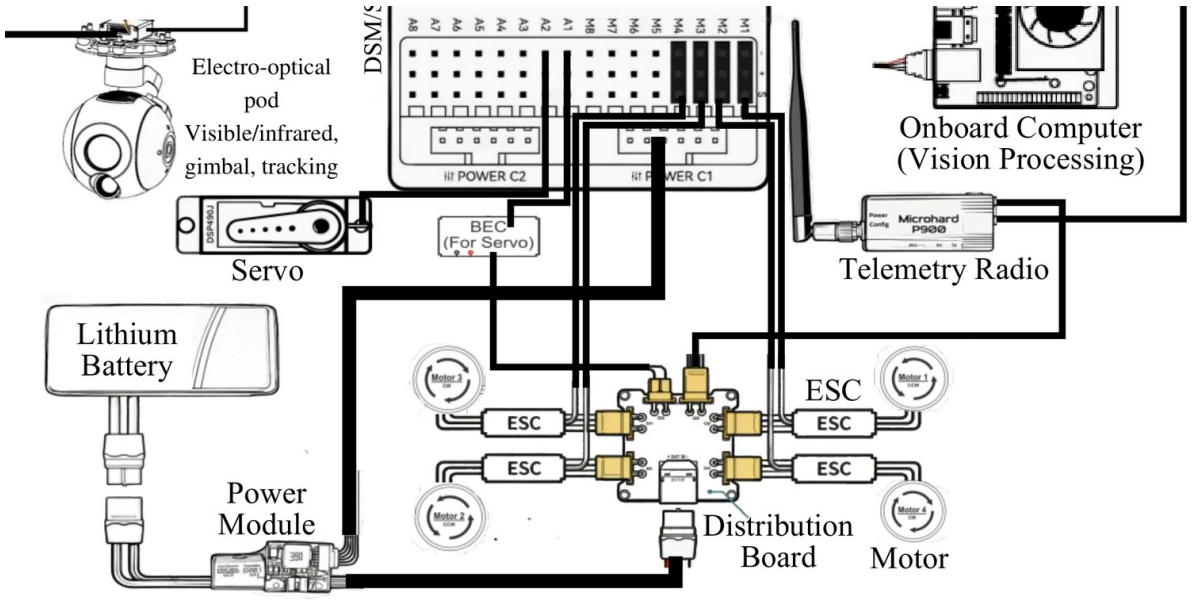


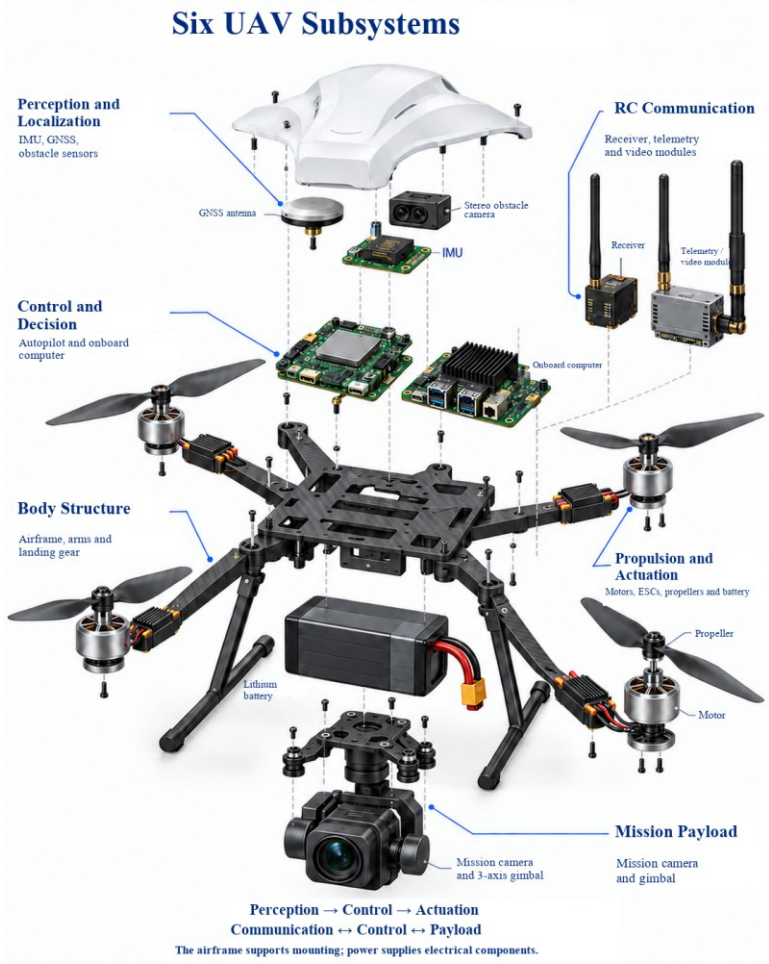
Fig. 1.13 Detail: Multicopter Power and Signal Wiring

- **Motor power:** Battery→power module→ distribution board→four ESCs→four motors; this path carries high current.
- **Autopilot supply:** The power module sends regulated power to POWER and measures battery voltage and current.
- **Servo supply:** A battery eliminator circuit (BEC) steps down distribution-board power and feeds the servo output header.
- **Control signals:** M1–M4 signal wires connect to the ESCs; servo signals use group A outputs.

1.3.2 Composition and Design Principles

Functions of Six Subsystems

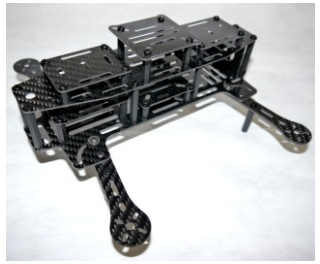
- **Four shared areas:** Structure, perception/localization, control/decision-making, and actuation appear as both layers and component systems.
- **Additional systems:** RC communication links the ground and other vehicles; mission payloads include cameras and pods as working equipment.
- **Layers only:** Environment interaction covers external conditions; swarm collaboration combines communication hardware and collaborative algorithms.
- **Inspection example:** The camera captures images; the onboard computer assesses targets; the autopilot and propulsion hold the viewing position.



Functional layers and component systems do not map one to one; one component can serve several functions.

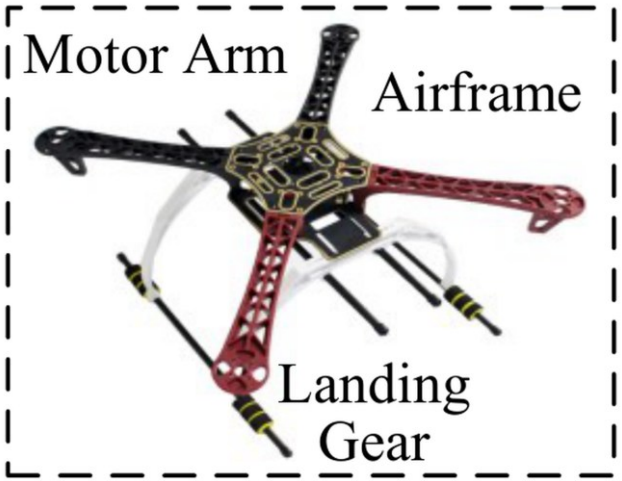
1.3.2 Composition and Design Principles: Body Structure System

Tradeoffs Among Mass, Strength, and Drag





- **Weight reduction:** A lighter frame can reduce hover thrust, but excessive thinning increases deformation and vibration.
- **Strength and stiffness:** Strength concerns failure; stiffness concerns deformation. Both affect mounting and load transfer.
- **Materials and joints:** Carbon fiber, aluminum alloys, and engineering plastics differ; joints and manufacturing affect overall performance.
- **Shape and layout:** Fixed-wing designs balance lift and drag; multicopters need clearance for propeller disks, payloads, and maintenance.







Weight reduction can benefit the vehicle only if load-bearing and mounting requirements are also met.

1.3.2 Composition and Design Principles: Perception and Localization System

Integrated and External Sensors: Mounting and Connections

- **Integrated sensors:** Built directly into the autopilot board; common examples are inertial sensors, magnetometers, and barometers.
- **Board interfaces:** Measurements commonly use Serial Peripheral Interface (SPI) or Inter-Integrated Circuit (I^2C).
- **External sensors:** Global Navigation Satellite System (GNSS) modules and external magnetometers connect separately, allowing relocation or mission-specific replacement.
- **Expansion interfaces:** Universal Asynchronous Receiver/Transmitter (UART), Controller Area Network (CAN), and Universal Serial Bus (USB) connect peripherals.



Photos: Different sensor forms

Integrated means built into the board; external means separately connected. Sensing role and mounting are different dimensions.

1.3.2 Composition and Design Principles: Perception and Localization System

IMUs, Magnetometers, and Barometers: What They Measure

- **Inertial sensing:** Accelerometers measure specific force (net non-gravitational force per unit mass); gyros measure angular rate.
- **Axis counts:** Three accelerometer and three gyro axes give 6 axes; a three-axis magnetometer gives 9; a barometer gives 10.
- **Aiding references:** Geomagnetic data aid heading; pressure changes aid altitude. Both depend on the environment.
- **Device forms:** Civil systems mostly use micro-electro-mechanical systems (MEMS) as chips or packaged modules. Noise and drift require fusion.



Integrated inertial chip



Xsens packaged module



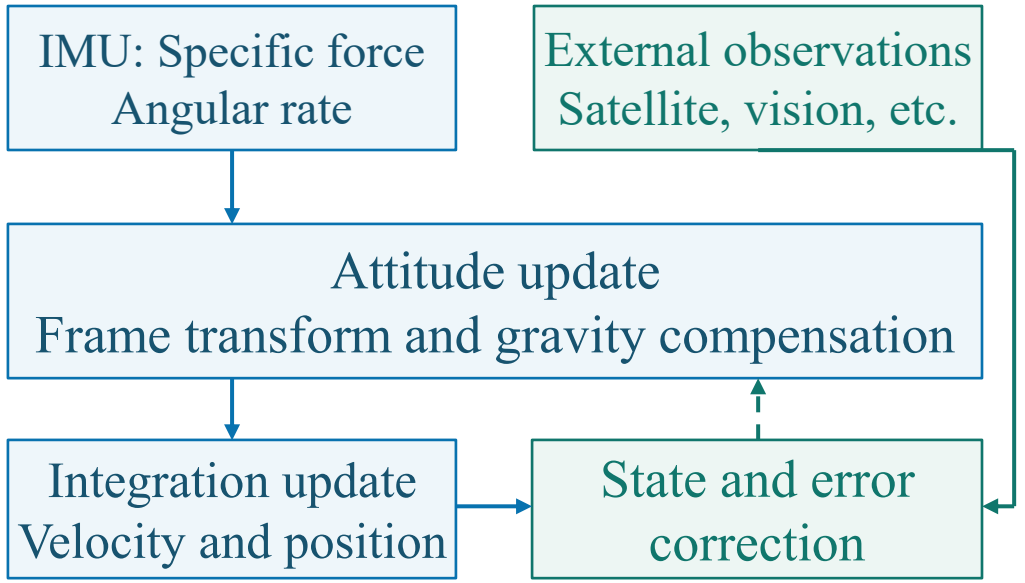
Barometer
Aids altitude estimation

Inertial measurements provide continuous motion information; magnetic field and pressure add heading and altitude references.

1.3.2 Composition and Design Principles: Perception and Localization System

Inertial Navigation: Motion Integration and Error Correction

- **Attitude and coordinates:** Correct inertial sensor biases, update attitude, transform specific force into the navigation frame, and account for gravity.
- **Motion integration:** Starting from the initial state, integrate acceleration to obtain velocity, then integrate velocity to obtain position.
- **Error propagation:** Bias, noise, and attitude errors enter the integrals, causing velocity and position drift.
- **Observation correction:** Satellite or vision observations constrain motion; fusion corrects state and any sensor biases that can be estimated.

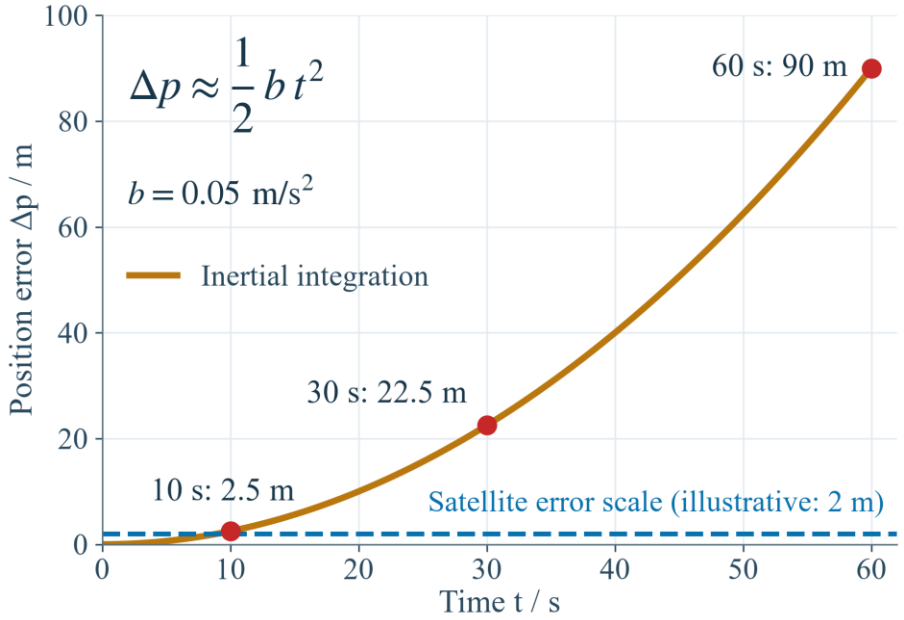


Inertial measurements continuously estimate motion; suitable observations must constrain long-term errors.

1.3.2 Composition and Design Principles: Perception and Localization System

Inertial Drift: A Numerical Example

- Error source: Accelerometer bias or attitude error leaves spurious acceleration b; velocity error is about bt and position error about $bt^2/2$.
- **Example:** $b = 0.05\ \mathrm{m/s^2}$ (about 0.3° tilt) gives about 2.5 m at 10 s, 22.5 m at 30 s, and 90 m at 60 s.
- **Device differences:** MEMS devices drift more; fiber-optic or laser gyros retain attitude longer but are larger and more expensive.
- Observation correction: Satellite or vision updates provide position; fusion corrects position and velocity and estimates bias to keep errors bounded.

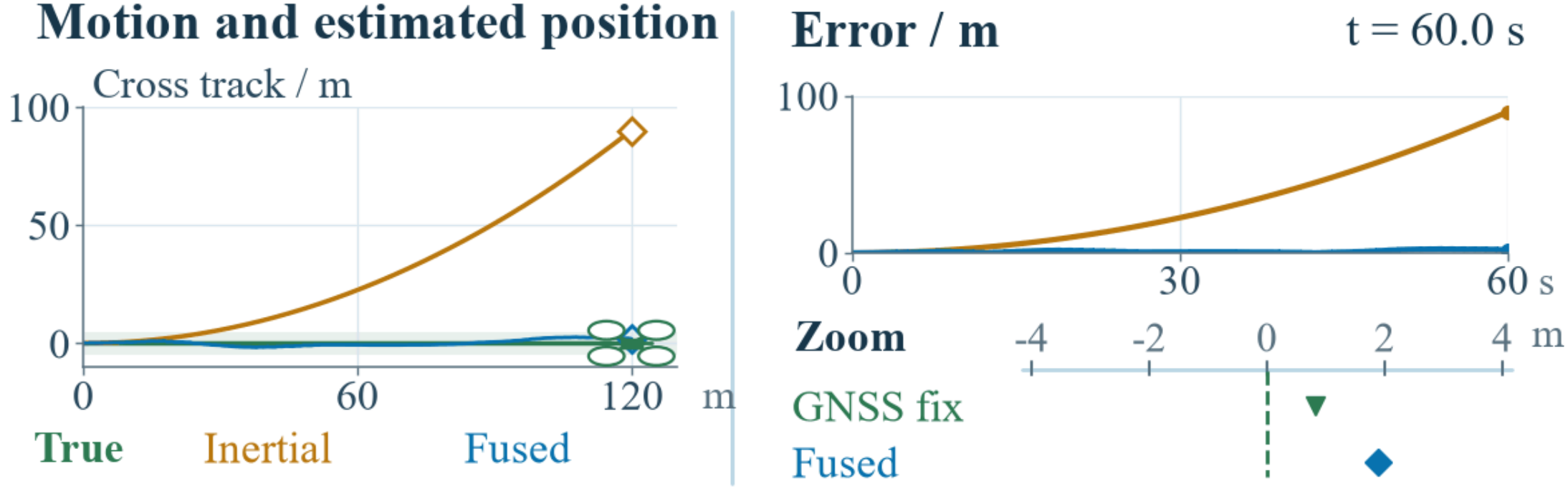


Plot: Position error from inertial integration

Constant acceleration bias makes position error grow with time squared; external observations can limit drift.

1.3.2 Composition and Design Principles: Perception and Localization System

Inertial Drift and External Observation Correction



Animation: Position error from constant acceleration bias and external correction (teaching simulation)

Inertial integration alone: Acceleration bias accumulates into position error.

Satellite correction: Periodic fixes reduce drift; update interval and noise affect residual error.

1.3.2 Composition and Design Principles: Perception and Localization System

GNSS: The General Term and Individual Systems

- **General term:** GNSS supplies positioning, velocity, and timing signals to receivers.
- **Individual systems:** The U.S. Global Positioning System (GPS), China's BeiDou, Europe's Galileo, and Russia's GLONASS are peer systems.
- **Multiple constellations:** Receivers can combine signals from several systems to increase available observations and improve positioning geometry.
- **Limits:** Buildings and signal reflections still cause errors; complex environments often require fusion with other sensors.



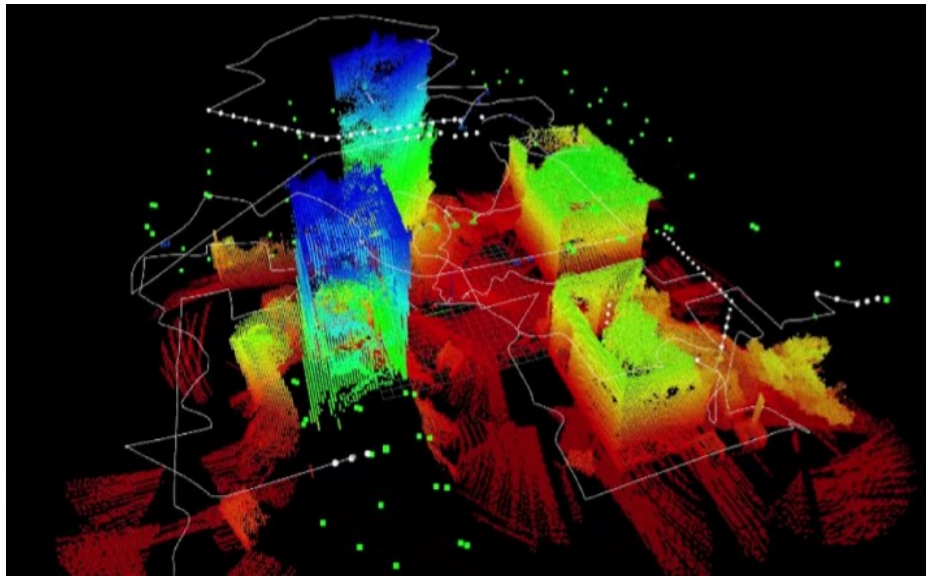
GNSS positioning: Travel time × light speed gives pseudorange with receiver clock bias; four or more satellites determine 3D position and that bias.

GNSS provides a global position reference; complex environments often require fusion with inertial and other observations.

1.3.2 Composition and Design Principles: Perception and Localization System

Vision and LiDAR Localization Without GNSS

- **Denied environments:** Indoors, tunnels, or strong interference may block or degrade satellite signals.
- **Environmental reference:** Cameras and light detection and ranging (LiDAR) estimate relative motion from features or local maps; vehicles need a common frame.
- **Inertial aid:** Inertial data provide continuous short-term motion; external observations constrain accumulated error.
- **Degradation:** Vision depends on light and texture; repetitive or uniform geometry may weaken LiDAR localization.



SLAM (simultaneous localization and mapping): Estimate pose and build a map together using cameras or LiDAR.

Vision or LiDAR localization often uses a local map; multiple vehicles need a common coordinate frame.

1.3.2 Composition and Design Principles: Perception and Localization System

External Sensors: Complementary Observations and Fusion

- **Satellite positioning:** Provides position, velocity, and time references; accuracy depends on the receiver, blockage, and reflections.
- **Cameras and LiDAR:** Image features and point-cloud geometry describe the environment for localization, mapping, and obstacle recognition.
- **Aiding observations:** Downward ultrasound or laser ranging gives relative altitude; optical flow with range and attitude can estimate horizontal velocity.
- **Fusion requirements:** Align time and coordinates to compare observations; quality checks detect anomalies and degradation.

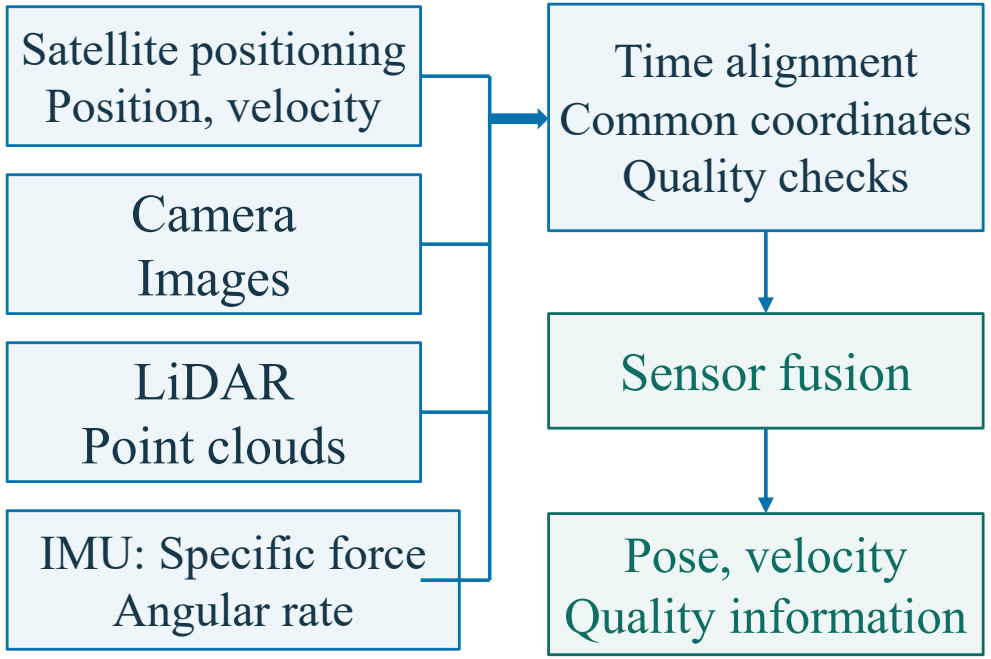


Diagram: Multi-sensor fusion

Complementary references support reliable fusion when time, coordinates, and measurement quality are checked.

1.3.2 Composition and Design Principles: Perception and Localization System

External Magnetometers: Why Placement Matters

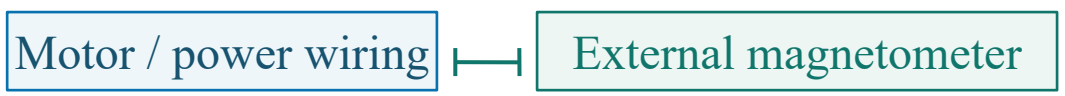
- **Geomagnetic reference:** A magnetometer measures the local magnetic field and combines it with attitude to aid heading estimation.
- **Interference:** Motors, ESCs, power wiring, and magnetic materials alter the local field and affect heading observations.
- **Distance:** A single wire carrying 20 A produces about 80 μT at 5 cm, exceeding the Earth's field; field strength falls with distance.
- **Calibration:** Orientation, fixed magnetic bias, and changing load affect measurements; calibration cannot replace good placement.







GNSS: satellite signals.
Magnetometer: magnetic field.
Some modules combine both.

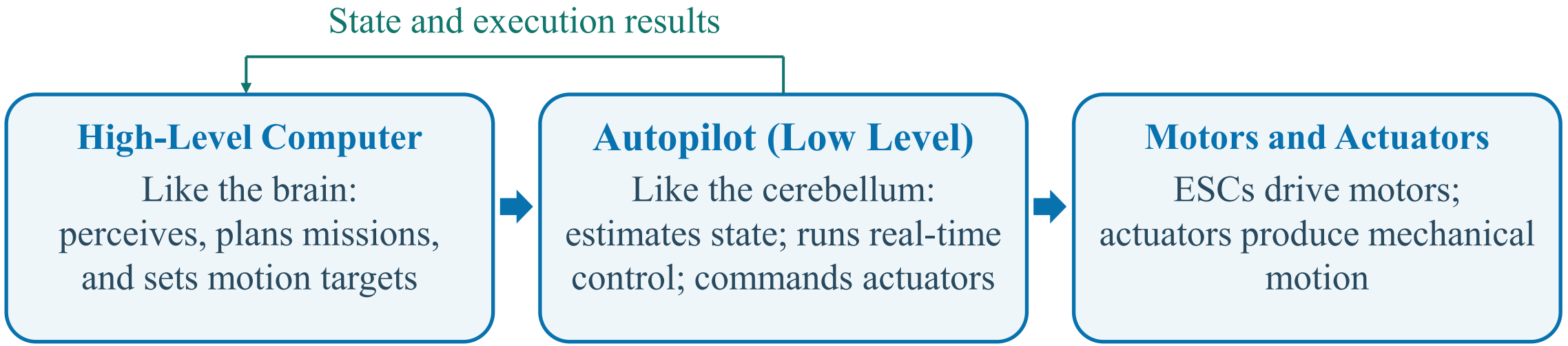


Motors and power wiring change the local magnetic field; separation helps stabilize heading observations.

1.3.2 Composition and Design Principles: Control and Decision System

Autopilot and Onboard Computer: Connections and Roles

> This book uses autopilot as the general term; on UAVs, it is commonly called a flight controller.



The high level sends mission targets; the autopilot runs real-time control and returns state.

1.3.2 Composition and Design Principles: Control and Decision System

Autopilot and Onboard Computer: Hardware and Tasks

> Vision needs image throughput and compute; motion control needs timely sampling and output.

Component	Hardware	Tasks
Autopilot	Microcontroller unit (MCU): a chip integrating processing, memory, and input/output; integrated sensors and interfaces	Real-time scheduling, state estimation, motion control, and protection
Onboard computer	Orin NX, RK3588, Raspberry Pi, and similar platforms	Operating system, perception, planning; cooperates with the autopilot through interfaces

Hardware must meet computing demands, control deadlines, and interface requirements.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

How Energy Produces Vehicle Motion

- **Energy sources:** Electric systems use batteries or other electrical supplies; fuel-powered systems use fuel and engines for mechanical work.
- **Actuators:** Motors, servos, or engines convert input into adjustable rotation, angle, or thrust.
- **Force generation:** Wheels use ground friction, propellers accelerate air, and jets expel working fluid.
- **Multicopter combination:** The battery powers ESCs, ESCs drive motors, and motors turn propellers; combined propeller thrust controls motion.

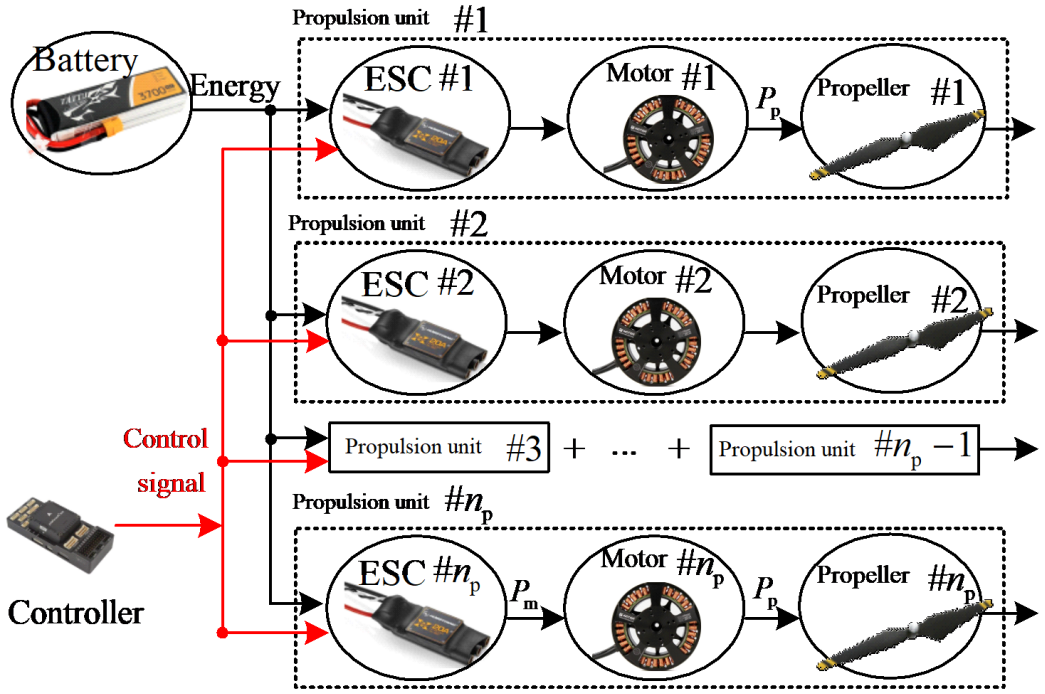


Diagram: Multicopter propulsion units

Actuators use energy to produce forces and torques, changing motion through interaction with the environment.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

Thrust and Reaction Torque

- **Thrust:** Rotating blades accelerate air; the reaction produces thrust along the propeller axis.
- **Reaction torque:** The motor drives the propeller and applies opposite torque to the frame; counter-rotating propellers can balance yaw torque.
- **Size and matching:** Larger diameter can increase thrust, but also power demand and rotational inertia; motor and frame spacing must match.
- **Pitch:** The theoretical advance per revolution; a 1045 propeller has a 10-inch diameter and 4.5-inch pitch.



(a) Two-blade propeller (b) Three-blade propeller (c) Four-blade propeller

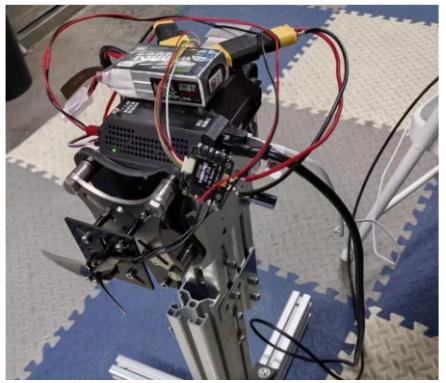


Photo: Motor and propeller

Motor applies driving torque to propeller

Propeller applies opposite torque to motor and frame

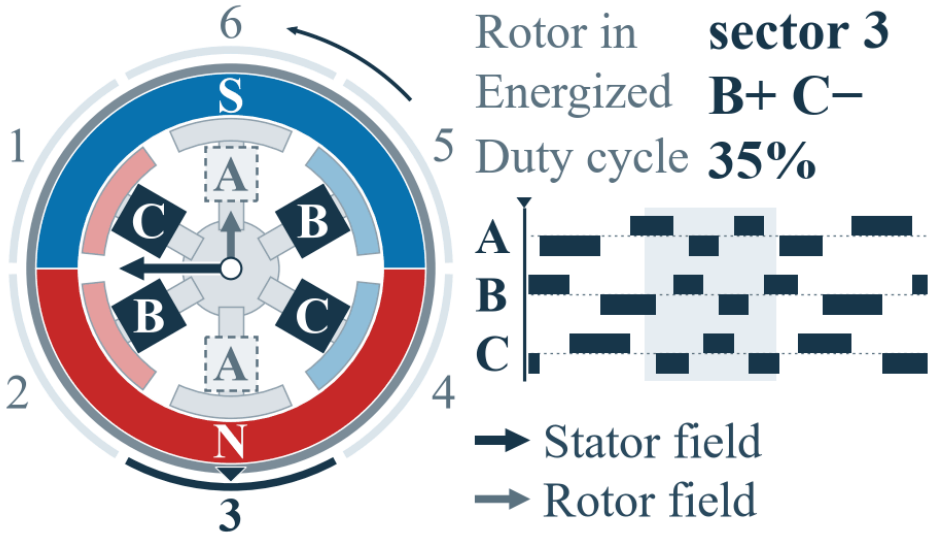
Same axis, opposite directions

Propellers produce thrust and reaction torque; rotation direction and propulsion matching both affect control.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

BLDC Commutation and Matching

- **Function:** A brushless direct current (BLDC) motor converts electrical energy into torque to turn a propeller.
- **Commutation:** Electronic circuits switch phase excitation according to rotor position, replacing mechanical brushes.
- **Features:** No brush friction, long life, and easy speed control; efficiency and heating depend on design and operating point.
- **Matching:** Select speed, torque, and power for payload and maneuvering needs; also check current, cooling, and propeller load.



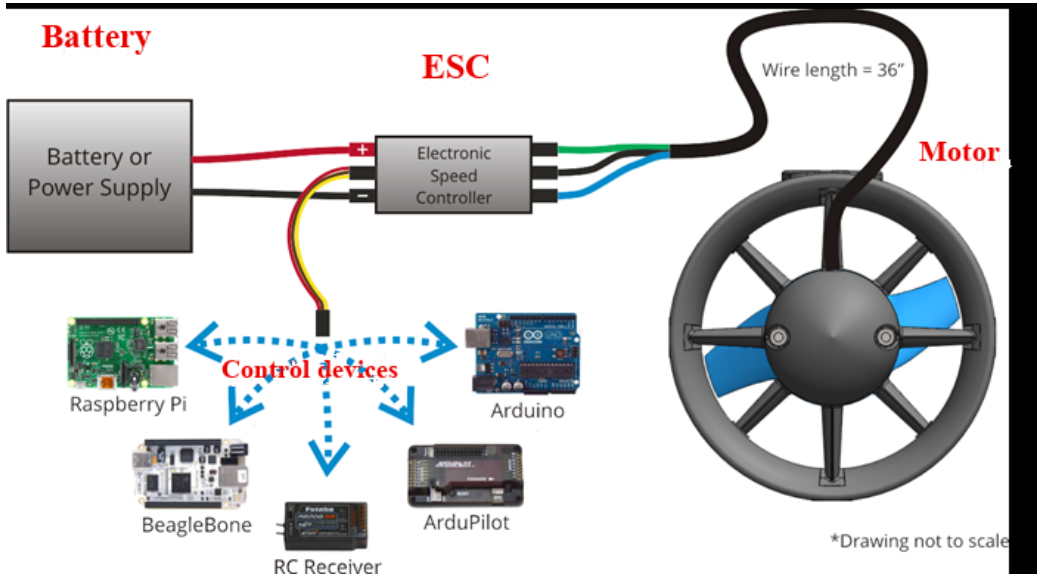
Principle: Sequential stator excitation, timed to rotor position, creates a rotating magnetic field that keeps the permanent-magnet rotor turning.

Brushless motors use electronic commutation; efficiency and heating depend on the propeller, supply, and operating point.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

Electronic Speed Controllers: Commutation and Power

- **Role:** An electronic speed controller (ESC) receives control commands and commutates a brushless motor.
- **Input command:** Traditional pulse width modulation (PWM) encodes throttle with about 1–2 ms pulses; digital protocols such as DShot are alternatives.
- **Internal drive:** The ESC regulates phase voltages and currents; internal power modulation differs from input throttle encoding.
- **Limits:** Supply, current, and cooling limit continuous output; temperature/speed feedback and protection vary by model.

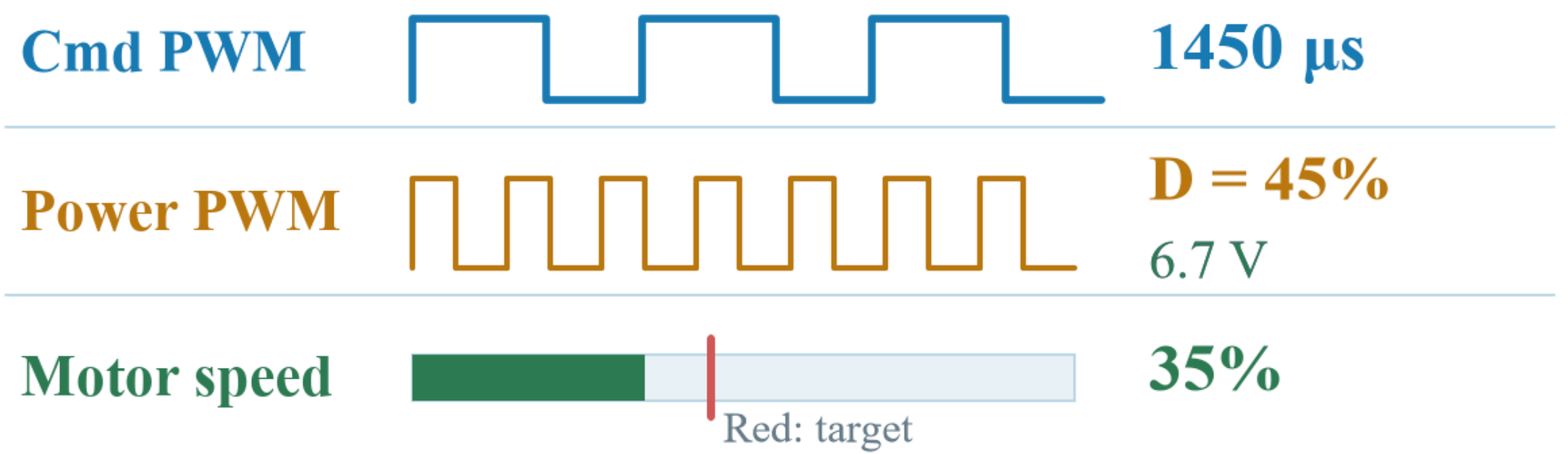


Principle: The ESC uses rotor position for three-phase commutation and PWM for output-voltage and power control.

The flight controller encodes the target; the ESC performs commutation and power regulation.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

Two PWM Roles: Command Pulse Width and Power Modulation



Animation: Input throttle pulse width and internal ESC power modulation serve different roles (conceptual)

Pulse-width command: About 1–2 ms pulses encode the throttle target.

Power switching: The ESC regulates internal duty cycle and sequences three-phase commutation.

1.3.2 Composition and Design Principles: Propulsion and Actuation System

Batteries: Lithium Types, Endurance, and Fuel Cells

- **Lithium polymer (Li-Po):** High energy density and high discharge current; most common in small multicopters. Polymer refers to the electrolyte; most use pouch packaging.
- **Cathode types:** Lithium iron phosphate (LFP) offers thermal stability and long cycle life; ternary lithium has higher specific energy and less mass for equal storage.
- **Endurance and power:** Energy ≈ voltage × capacity. A 14.8 V, 5 Ah battery stores about 74 Wh; at 300 W average draw, theoretical flight time is about 15 min. Discharge rating limits instantaneous current.
- **Hydrogen fuel cells:** The stack converts hydrogen–oxygen chemical energy to electricity. A tank carries hydrogen; fuel delivery and cooling add system mass.



Photo: Battery pack


Photo: Cylindrical cells

Stored energy affects endurance; peak power limits short-term loads; supporting equipment adds mass.

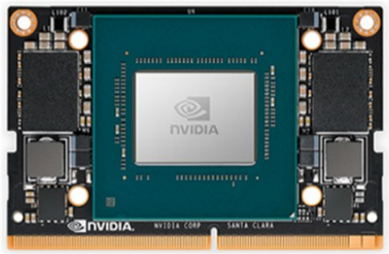
1.3.2 Composition and Design Principles: Propulsion and Actuation System

Ground-Vehicle Motor Control: Motor Controllers and ECUs

- **Definition:** Electronic control unit (ECU) is a general term for an automotive control module, with different possible functions.
- **Motor control:** A motor controller regulates current, torque, or speed from commands and monitors temperature and operating state.
- **Hierarchy:** A higher-level ECU requests torque or speed; the motor controller regulates the motor and returns state.
- **Platform differences:** Ground vehicles must coordinate braking, steering, and grip; aircraft failure responses cannot simply be copied.



Autopilot



Onboard computer



ECU

Ground vehicles also close command–feedback loops and coordinate propulsion, steering, and braking.

1.3.2 Composition and Design Principles: RC Communication System

RC, Telemetry, and Video: Connecting Ground and Vehicle

- **Remote control (RC):** The ground sends control inputs and mode selections; an onboard receiver passes signals to the autopilot.
- **Telemetry/data:** The vehicle returns position, battery, and mission state; the ground station sends waypoints, parameters, or missions.
- **Video:** The vehicle returns images or video for inspection and first-person view (FPV) control.
- **Mission interaction:** The onboard system returns annotated images and locations; after ground confirmation, it receives a task and reports progress.



The capabilities affected by a communication loss depend on which service is lost.

1.3.2 Composition and Design Principles: RC Communication System

Communication Links: Band Examples and Tradeoffs

> These bands are textbook examples; availability depends on radio design and regional operating rules.

Link	Typical Band Examples	Tradeoffs
Remote control	2.4 GHz, etc.	Continuous control-input updates Latency and jitter affect response
Telemetry/data	433 MHz or 915 MHz	State updates as needed Task delivery needs confirmation
Video	5.8 GHz, etc.	Better quality needs more bandwidth Compression and buffering add latency

Control needs timely delivery; video uses bandwidth. Capacity allocation affects shared-link response.

1.3.2 Composition and Design Principles: RC Communication System

Networking: Point-to-Point, Multipoint, Ad Hoc, and Cellular

- **Point-to-point:** One ground station connects to one vehicle; simple and common for single-vehicle missions.
- **Point-to-multipoint:** One ground station connects to several vehicles; vehicle-to-vehicle information passes through the station.
- **Ad hoc:** The station and vehicles interconnect; messages can take multiple hops through other nodes, supporting swarms.
- **Public 4G/5G:** Mobile base stations and public-network servers provide connectivity; coverage depends on base-station signals.

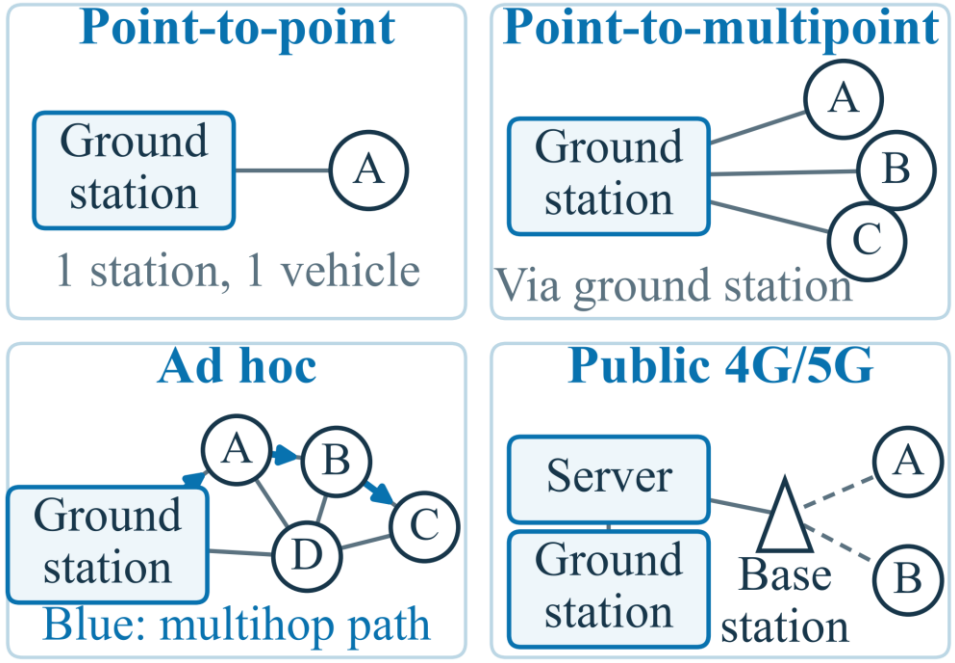


Diagram: Four network topologies

Network topology determines message forwarding and how widely a lost node affects the system.

1.3.2 Composition and Design Principles: Mission Payload System

Mission Payloads and Platform Requirements

- **Mission role:** A payload directly performs observation, measurement, communication, or physical work.
- **Common forms:** Cameras, electro-optical pods, LiDAR, specialized sensors, communication relays, and deployment devices.
- **System requirements:** Check mass, center of gravity, mounting space, power, and data bandwidth against mission needs.
- **Functional role:** One camera may support navigation or mission imaging; classify it by actual use and define shared interfaces.



Pod



Gripper



Weapon

The mission payload defines the work and sets requirements for the platform, power, and communication.

Chapter Outline

Components and Architecture: Tasks, Components, and Operation

Chapter 1 Introduction

1.1 Concepts and Applications

1.2 History and Trends

1.3 Components and Architecture

1.4 Coordinate Systems and Control Principles

1.5 Assembly and Debugging

1.6 RflySim Toolchain

1.7 Learning Path and Experiments

1.8 Exercises and Discussion

3 Units in This Section

1.3.1 Overall Framework

Textbook: Sec. 1.2.1

1.3.2 Composition and Design Principles

Textbook: Sec. 1.2.4

1.3.3 Intelligent Control Architecture

Textbook: Sec. 1.2.5

1.3.3 Intelligent Control Architecture

Low- and High-Level Systems: Components and Roles

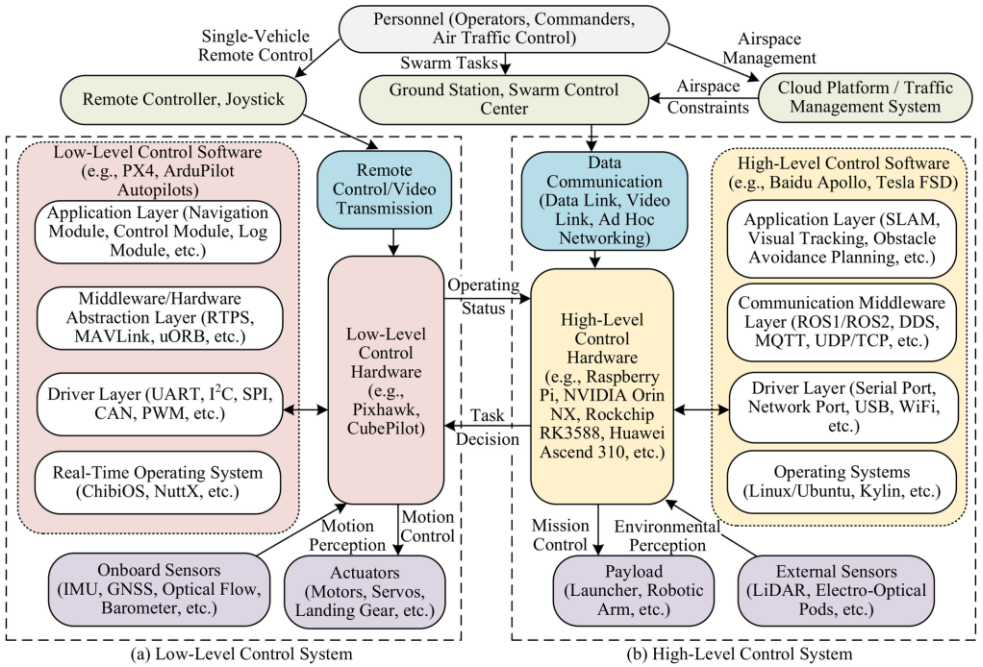
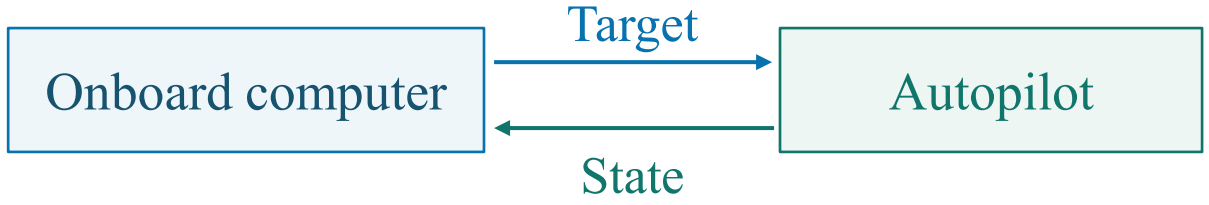


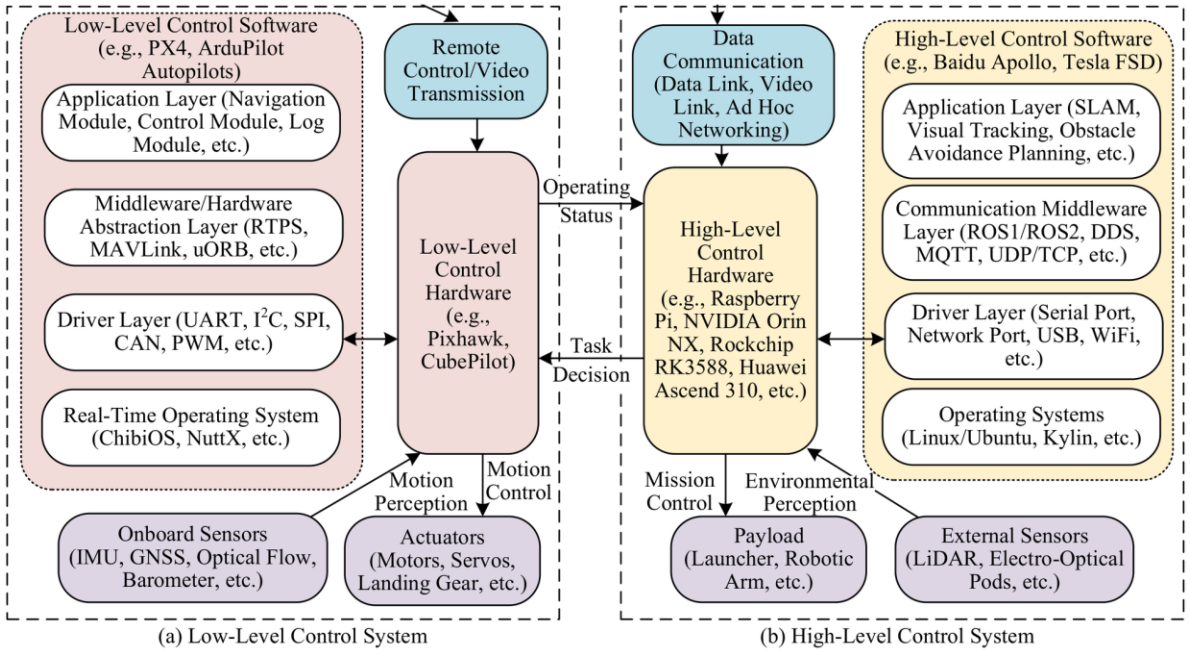
Fig. 1.14 Control Architecture

- **Low-level loop:** The autopilot reads sensors, controls actuators, and handles basic motion and protection.
- **High-level missions:** The onboard computer perceives and plans; a person or ground system supplies missions.
- **Collaboration:** Targets go down; state and results return. Exceptions follow predefined modes.
- **Rationale:** Embedded processors and real-time systems support low-level reliability and timing; high-performance boards process high-level images.



1.3.3 Intelligent Control Architecture: Hierarchical Collaboration

Connections Among Observation, Computation, and Execution



Low-level branch (left)
Sensors→hardware→actuators
Motion sensing and control
High-level branch (right)
External sensors→hardware→payload
Environment perception; mission control
Bidirectional interface (center)
Mission decisions down; operating state up
Onboard in this figure
Broadly, sensors connected to the autopilot; GNSS and optical flow are usually external modules.

Fig. 1.14 Detail: Low- and High-Level Operating Branches

1.3.3 Intelligent Control Architecture: Hierarchical Collaboration

Low- and High-Level Control: Tasks and Timing

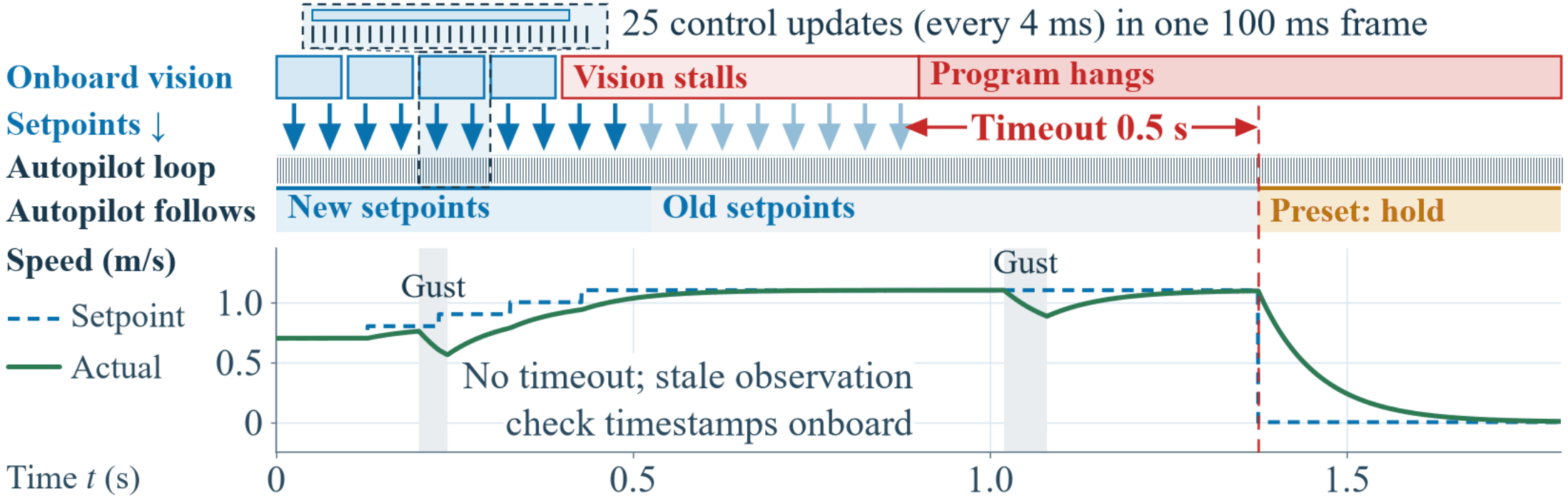
> Attitude control must continue during image processing; the high level supplies trackable targets.

Aspect	Low: More Stable Tracking	High: Smarter and More Collaborative
Goal	Stable, timely motion-target tracking	Perception and autonomous missions
Typical tasks	State estimation; attitude and position control	Recognition, mapping, planning, collaboration
Hardware	Embedded processors and sensor interfaces	High-performance boards or remote computing
Focus	Control deadlines, fault detection, protection	Workload, data freshness, degraded operation

The low level emphasizes control deadlines; the high level emphasizes mission computation. Both handle faults.

1.3.3 Intelligent Control Architecture: Hierarchical Collaboration

High-Level Timeout: What Low-Level Control Can Retain



Animation: Update timing, communication delay, and timeout fallback in hierarchical control (conceptual)

Despite variable high-level computation time, low-level control keeps its fixed cycle and handles stale targets.

1.3.3 Intelligent Control Architecture: Low-Level Control System

Autopilot Hardware and Real-Time Control

- **Platform:** A processor runs estimation and control software; sensors provide measurements; interfaces connect peripherals and actuators.
- **State update:** Sampling and fusion estimate attitude, velocity, and position. Later observations leave the controller with older state.
- **Control output:** Targets and estimated state determine actuator commands; mode management selects target sources and protection handles faults.
- **Real-time scheduling:** A real-time operating system (RTOS) schedules by priority; interrupts or shared-resource contention can delay control output.

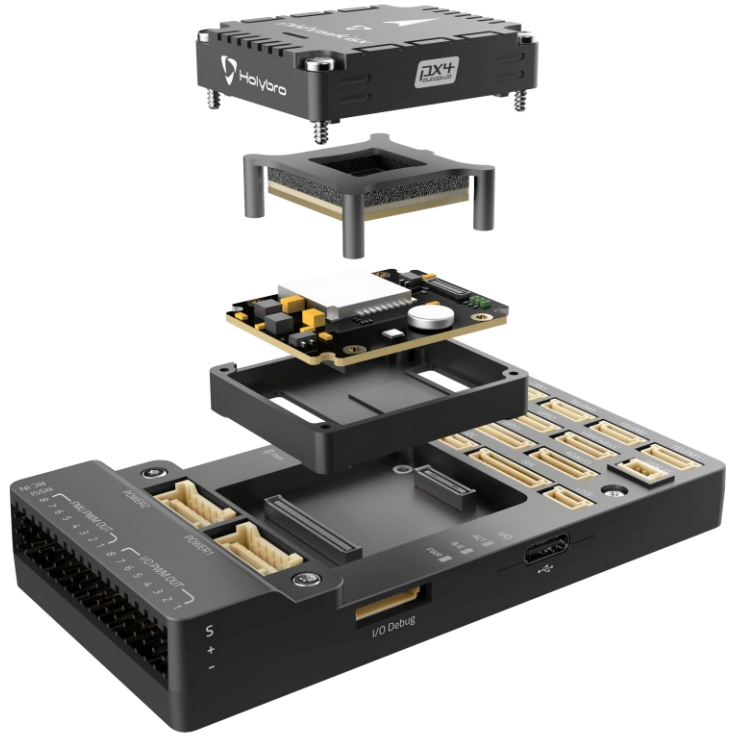


Photo: Pixhawk 6X board example

Sample → Estimate → Control → Output

The autopilot links sampling, estimation, and control; real-time scheduling keeps the loop updating on time.

1.3.3 Intelligent Control Architecture: Low-Level Control System

Autopilot Software: Sensors to Outputs

- **Drivers:** Encapsulate device access and interrupts; serial ports and buses connect sensors and actuators and provide output interfaces.
- **Scheduling:** The RTOS manages timing, priorities, and synchronization to meet the agreed execution schedule.
- **Algorithms:** State estimation supports attitude, velocity, and position control; allocation converts demands into actuator commands.
- **Communication:** Micro Air Vehicle Link (MAVLink) and similar lightweight protocols exchange state and commands with ground stations and onboard computers.

NuttX RTOS

RTOS

PX4 autopilot

Autopilot Software

THE AUTOWARE FOUNDATION

Autonomous-Driving Stack Example

ARDUPILOT

Autopilot Software

Drivers acquire measurements; estimation produces state; control and allocation generate actuator commands.

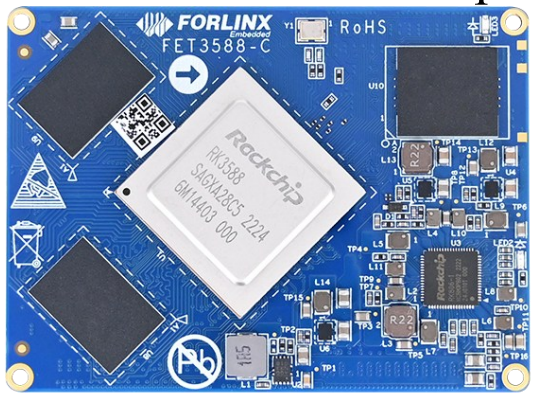
1.3.3 Intelligent Control Architecture: High-Level Control System

Onboard Computers: Workload and Hardware Integration

- **Workload:** Recognition processes images; mapping and localization maintain pose and maps. Data volume and update rate jointly determine load.
- **Hardware:** Jetson developer kits provide peripheral connections; an RK3588 compute module needs a carrier board for vehicle integration.
- **Software support:** Acceleration depends on software; sensor drivers and interfaces determine whether images and point clouds reach algorithms continuously.
- **Onboard cost:** Compute and carrier boards add mass and power draw; cooling uses space and payload capacity, and thermal conditions limit sustained processing.



Jetson developer kit



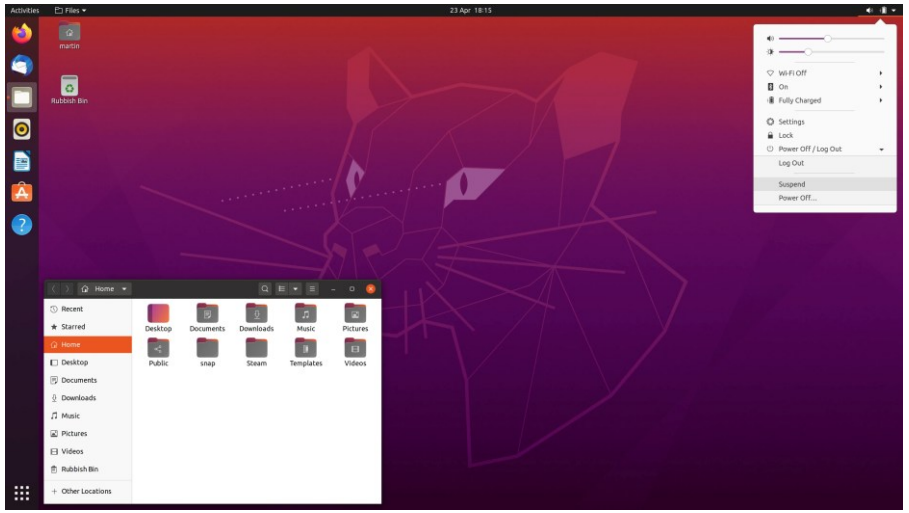
RK3588

Onboard computation supports perception and planning; software, interfaces, power, and cooling limit sustained performance.

1.3.3 Intelligent Control Architecture: High-Level Control System

Onboard Operating Systems and Middleware

- **Operating system:** Ubuntu and other Linux systems manage compute, memory, storage, and networks; drivers provide camera/radar data.
- **ROS 2:** Robot Operating System 2 is a robotics framework; its ROS middleware (RMW) interface can use DDS or other implementations. Linux is one supported environment.
- **Communication:** Data Distribution Service (DDS) moves node data; Message Queuing Telemetry Transport (MQTT) supports publish/subscribe.
- **Topics and services:** Subscribers receive image/pose streams as needed. Services use one request and one reply for queries/settings.



The operating system manages hardware; middleware moves data between modules. Together they support application algorithms.

1.3.3 Intelligent Control Architecture: High-Level Control System

Onboard Software: Algorithms, Data Alignment, and Networking

- **Applications:** SLAM jointly estimates pose and map; recognition extracts environmental targets; planning generates feasible paths.
- **Data coordination:** Timestamps and frame transforms align observations;
  queueing and processing add end-to-end delay.
- **High-level communication:** Ad hoc data links, 4G/5G, or
  Wi-Fi connect other vehicles, ground systems, and the cloud
  to exchange missions and perception results.

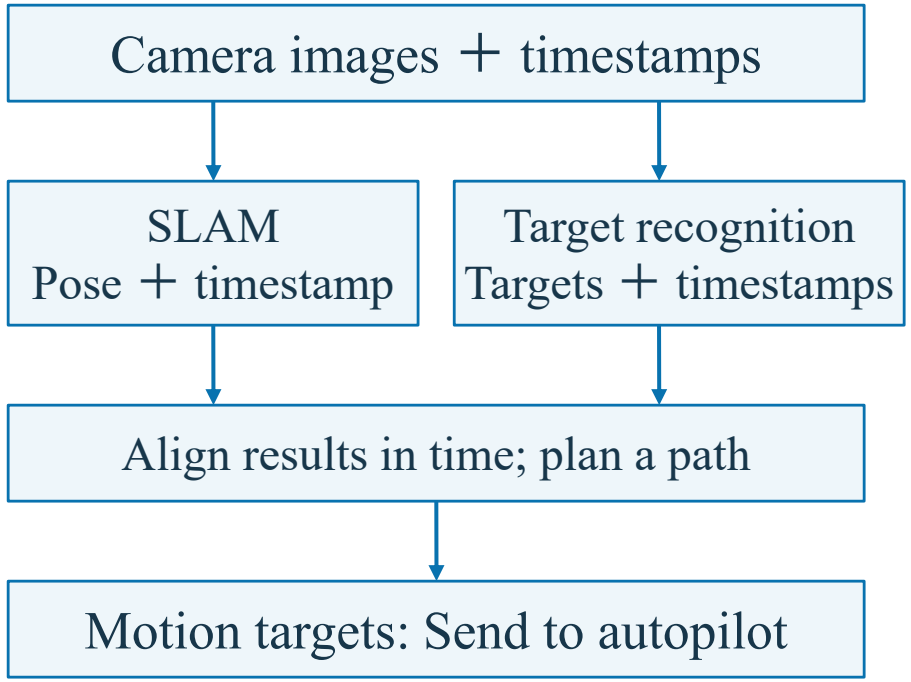


Diagram: Parallel Vision Processing

System software provides resources and communication; autonomy still requires localization, recognition, and planning algorithms.

1.3.3 Intelligent Control Architecture: Hierarchical Collaboration

Bidirectional Communication: Onboard Computer and Autopilot

- **Target delivery:** The high level sends position or velocity targets;
  the autopilot tracks them in the specified frame.
- **State feedback:** Pose, mode, and health return upward; error determines whether to continue approaching or replan.
- **Interface contract:** Frames and units define target meaning;
  timestamps, update periods, and timeout rules define freshness.
- **Completion:** Receiving a message confirms delivery; position error, speed, and dwell time determine arrival.

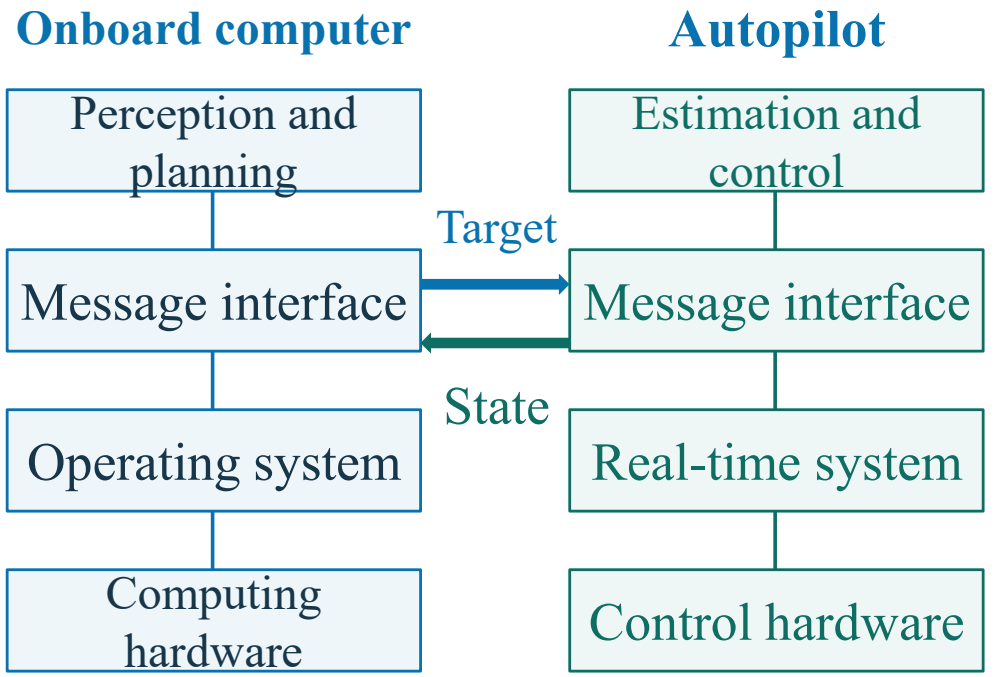


Diagram: Software Layers and Interfaces

Message delivery does not mean arrival at the target; subsequent actions require state feedback.

1.3.3 Intelligent Control Architecture: Hierarchical Collaboration

Redundancy and Fallback: Retaining Control After a Fault

- **Redundancy:** Back up critical sensing, computation, or power to reduce single-point failure effects.
- **Fault response:** Detect the anomaly and isolate its source, then switch to backup or reallocate available control capability.
- **High-level failure:** Continuing the mission, manual takeover, or return depends on localization, mode, communication, and predefined strategies.
- **Common-cause failure:** Shared power or the same interference
  can disable primary and backup together; use independent power and separate wiring.

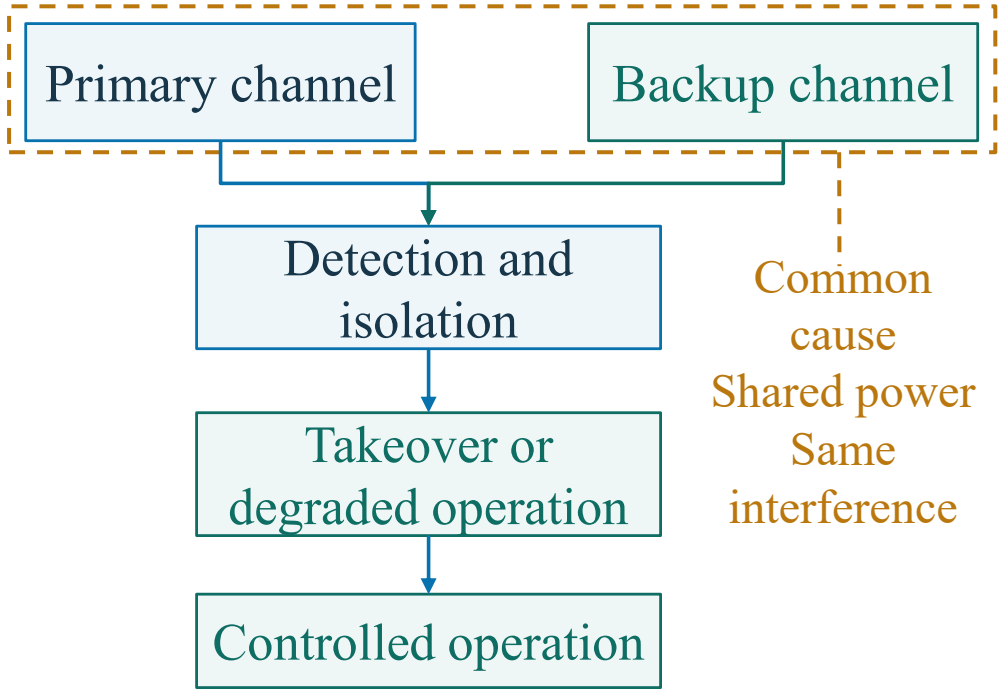


Diagram: Fault Detection and Backup Switching

Redundancy must preserve essential functions after faults; detection and reliable takeover are critical.

Class Discussion

Inspection Architecture Exercise: Trace a Mission

Discussion Questions

- **Trace the mission:** From approach and image capture, identify observations, computation, targets, and actuator outputs.
- **Identify interfaces:** What passes among camera, onboard computer, and autopilot? How do power and data wiring differ?
- **Analyze faults:** If images stop updating but localization remains valid, or magnetic interference occurs, which judgment is affected first? What remains possible?
- **Transfer:** On a ground vehicle, what processing stays? Which actuators and constraints change?

Answer Guide: Three Main Threads

- **Six layers:** Trace perception, decision-making, actuation, and environmental effects in one vehicle's loop, then swarm collaboration.
- **Six subsystems:** Map functions to parts; check mounting, power, communication, and mission capability.
- **Two levels:** Split work by compute load and control timing; connect targets, state, and execution results.

Use specific components, interfaces, and failure scenarios to explain whether the system decomposition is appropriate.

Class Discussion

Inspection Architecture Exercise: Reference Answers

- **Mission:** Camera and localization provide observations; the onboard computer sets approach targets; the autopilot tracks them; ESCs and motors produce motion.
- **Interfaces:** Camera→onboard computer carries images; onboard computer↔autopilot carries targets and state. Keep power and signal wiring separate.
- **Faults:** Stale inspection images first affect recognition and target updates; the high level checks freshness. An interrupted target or keepalive stream triggers its failsafe on timeout. Magnetic interference first affects heading estimation.
- **Transfer:** Perception and planning can remain; replace propeller propulsion with wheel drive, steering, and braking, considering grip and protective stopping.

Understanding architecture means tracing components, data, and fault effects through a real task.