Max Freeman

max_freeman@berkeley.edu | (607) 327-7804 | linkedin.com/in/maxfreeman | Berkeley, CA

Education

University of California, Berkeley — **May 2025**
Master of Engineering in Mechanical Engineering, *Concentration in Controls & Robotics* — GPA: 3.98

Cornell University — **May 2024**
Bachelor of Science in Mechanical Engineering — GPA: 3.77 | *Magna Cum Laude*

Skills

Robotics & Controls: Feedback Control (PID, LQR, MPC) | State Estimation & Sensor Fusion | Dynamics & Kinematics
Programming & Software: Python | C++ | ROS2 | MATLAB | Simulink | Git | Linux | Optimization (CasADi, Pyomo)
Embedded Systems: Sensors & Actuators | Communication Protocols (CAN, I2C, SPI, UART) | Oscilloscope Debugging
Leadership: Project Management | Agile Development | Cross-Functional Collaboration | Stakeholder Communication

Highlighted Experience & Projects

Lead Robotics Engineer, *Multimodal Autonomous Robot Design (MEng Capstone)* — **September 2024 - May 2025**

- Led a team of 4 engineers in a full-stack robotics project to develop a multimodal robot capable of both driving and flying, owning system architecture, control system development, and hardware testing strategy.
- Developed and deployed a custom ROS2-based real-time flight controller in Python using a cascaded PID architecture, validating hover and attitude control through incremental tethered tests with built-in failsafes.
- Implemented and tuned real-time Model Predictive Control (MPC) algorithms for trajectory planning and obstacle avoidance, achieving <5 cm lateral error to final setpoints in hardware testing.
- Developed a lightweight Python simulator to validate controller behavior before hardware deployment.
- Conducted software-in-the-loop and hardware-in-the-loop testing to validate control strategies, accelerate iteration, and identify failure points early, mitigating downstream development risks.
- Contributed to a large-scale ROS2 codebase, integrating new control and simulation nodes into planning, perception, and actuation modules, and performing targeted refactors.

Control of Autonomous Flight Project, *University of California, Berkeley* — **September 2024 - December 2024**

- Developed a quadcopter flight controller in C++, achieving precise attitude, altitude, and position control.
- Implemented sensor fusion models in C++, utilizing data from optical flow, IMU, and Time-of-Flight sensors to improve sensor data accuracy and provide precise feedback for control.
- Diagnosed and resolved stability issues through targeted subsystem testing, sensor debugging, and PID controller tuning to validate fixes and improve system response.

Hardware Test & Integration Intern, *Lit Motors* — **June 2024 - July 2024**

- Collaborated with cross-functional teams at an EV startup to refine subsystem designs and develop testing infrastructure for a two-wheeled, self-balancing vehicle that uses Control Moment Gyroscopes (CMGs).
- Led development of a dual-plane dynamic balancing test rig for CMG evaluation, integrating structural hardware, sensors, and a belt-drive system to identify weight imbalances and reduce vibrations.

Fast Robots Project, *Cornell University* — **January 2024 - May 2024**

- Managed end-to-end development of an embedded autonomous RC robot, integrating sensors and motor drivers on an Arduino Nano and implementing real-time control logic in C++ and Python.
- Interfaced with Time-of-Flight and IMU sensors over I2C and SPI, applying sensor fusion algorithms and software-based filters in C++ to minimize sensor output noise by over 50%, enhancing sensor accuracy.
- Implemented and tuned distance-based PID controllers to improve tracking, reducing settling time by 40%.
- Diagnosed hardware and sensing issues using oscilloscope traces and telemetry data, resolving integration faults, and improving system responsiveness.

Lead Systems Engineer, *Magnus Ultralight Drone Research, Cornell University* — **August 2023 - May 2024**

- Integrated flight control hardware for an experimental eVTOL quadcopter that leveraged the Magnus effect to increase flight range, configuring the Pixhawk 6C for autonomous stabilization and sensor communication.
- Led post-flight diagnostics to investigate system failures, testing sensors, communication links, and mechanical components to identify root causes; implemented corrective actions, including ESC recalibration and mechanical stiffening to suppress a torsional mode, significantly improving flight stability.