

Daniel Park

dsp001@ucsd.edu | (562) 746-2910 | U.S. Citizen | [linkedin.com/in/danielparkaero](https://www.linkedin.com/in/danielparkaero) | **Portfolio:** danielparkaero.com

EDUCATION

University of California, San Diego

Bachelor of Science, Aerospace Engineering | GPA: 3.8

San Diego, CA

Expected Graduation: June 2027

SKILLS

Software: SolidWorks (FEA, CFD), AutoCAD, MATLAB, Python, C++, Arduino IDE, OpenRocket, VS Code, Microsoft Office

Hardware: 3D Printing, Drill Press, CNC, Electronics (Arduino), Fabrication, Soldering, Machining, Manufacturing

Soft Skills: Problem-Solving, Attention to Detail, Leadership, Communication, Collaboration, Korean (Bilingual)

TECHNICAL EXPERIENCE

UC San Diego Rocket Propulsion Laboratory | Lead Solid Motor Rocket Engineer

October 2024 – Present

- Led a team of 5 engineers to design, manufacture, and test a G-class solid rocket motor, achieving a target velocity of 633 ft/s and an apogee of 3,191 ft, ensuring structural integrity under dynamic loads.
- Applied SolidWorks Flow Simulation and OpenRocket to design and validate propulsion components, conducting FEA and CFD analyses to evaluate stress and strain distributions under combustion conditions.
- Designed, fabricated, and assembled the avionics bay using custom 3D-printed PLA enclosures; soldered and integrated Arduino-compatible sensors.
- Presented simulation data in Preliminary and Critical Design Reviews (PDR/CDR) to six senior engineers and a team of 40+ fellow engineers, incorporating feedback to improve structural integrity and aerodynamic efficiency.

UC San Diego Boechler Research Laboratory | Undergraduate Materials Testing Researcher

September 2024 – Present

- Designed and CNC-machined a custom polycarbonate impact enclosure using SolidWorks and T-slot aluminum framing to safely contain high-velocity material fragments during dynamic impact testing.
- Engineered oscilloscope-based data acquisition system using laser timing sensors; automated data analysis and reporting with MATLAB scripts.
- Programmed and integrated an electronic Arduino-controlled venting and pressure relief system with millisecond response times, enabling remote operation from over 20 feet away to ensure operator safety.
- Collaborated with Ph.D. researchers in weekly design and analysis sessions; authored Standard Operating Procedures and technical documentation for reproducible lab testing.

APstronautTutoring.com | Founder & Lead Tutor

November 2023 – Present

- Founded and operated an online tutoring business delivering high-level instruction in AP subjects to 100+ high school students, with a focus on preparing future engineers and scientists.
- Designed and taught engineering-first principles in AP Calculus AB/BC (limits, derivatives, integrals, differential equations), and AP Physics 1/C (free body diagrams, Newtonian mechanics, electricity & magnetism).

PROJECTS

High-Impact Containment Enclosure Box | Mechanical Design Lead

March 2025 - June 2025

- CAD-modeled and fabricated a custom impact enclosure using SolidWorks to safely contain high-velocity debris during high-strain-rate aerospace material tests.
- Fabricated the enclosure using drill presses and manual machining to create precise cutouts for test bars and cabling, ensuring compatibility with dynamic instrumentation setups.
- Documented the full design and build process, including mechanical drawings, assembly instructions, and operational procedures to support repeatability, safety compliance, and future system upgrades.

Arduino-Based Remote Controller & Live Sensor | Systems Integration Researcher

October 2024 – March 2025

- Developed a custom Arduino-based embedded control system to remotely operate high-pressure solenoid valves for projectile firing and venting during dynamic material impact tests.
- Programmed digital I/O control logic to process real-time sensor data from pressure transducers and voltage monitors; integrated LED indicators for live system status and threshold alerts.
- Engineered a failsafe hardware interface panel with labeled toggle switches (FILL, FIRE, VENT), debounce logic, and emergency override for safe testing.