

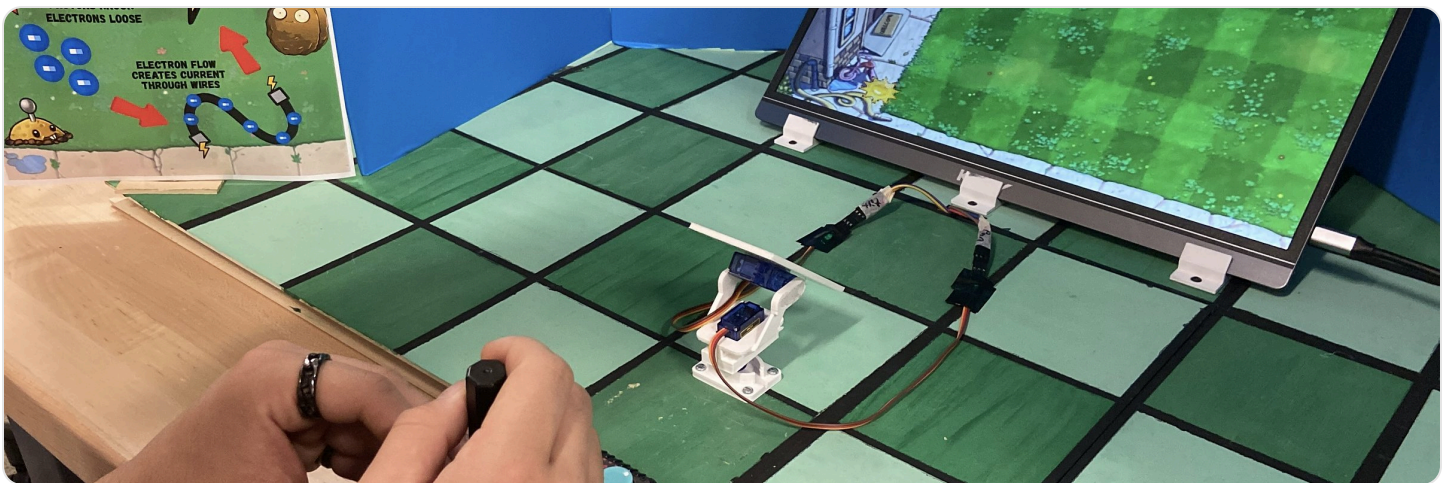
Evan Mah

Mechanical Engineering Portfolio · Northeastern University

mah.e@northeastern.edu
linkedin.com/in/evan-mah-855b57269
evanmah.com

Mechanical engineering student at Northeastern, class of 2029. I design, build and debug hardware. For the firmware and apps around it, I write the specs and test on the device, and AI coding agents write the code.

Looking for a Summer 2027 internship or a Fall 2027 co-op (July–December) in hardware, motorsports or product engineering.



Projects

1. Solar Tracker Arcade, Feb – Apr 2026
2. Phone Box, Aug – Oct 2026
3. Pasta Truss Bridge, Jan – Feb 2026
4. Formula SAE: carbon-fiber mold tooling, Spring 2026 – present

Skills

Design	SolidWorks, AutoCAD, Fusion 360, Design for manufacturing
Build	CNC routing, FDM printing, Laser cutting, Composite molds, Soldering
Embedded	Raspberry Pi Pico, ESP32 wiring and bench debugging, I2C, PWM, serial
Code	Python, MATLAB, MicroPython
AI-directed	CircuitPython firmware, React Native iOS app, Electron desktop app

Education

Northeastern University, B.S. Mechanical Engineering, 2025 – 2029

Cornerstone of Engineering (CAD, programming), Statics, Mechanics of Materials, Thermodynamics

Experience

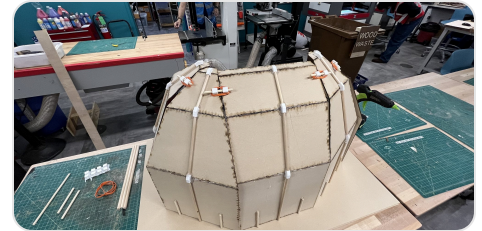
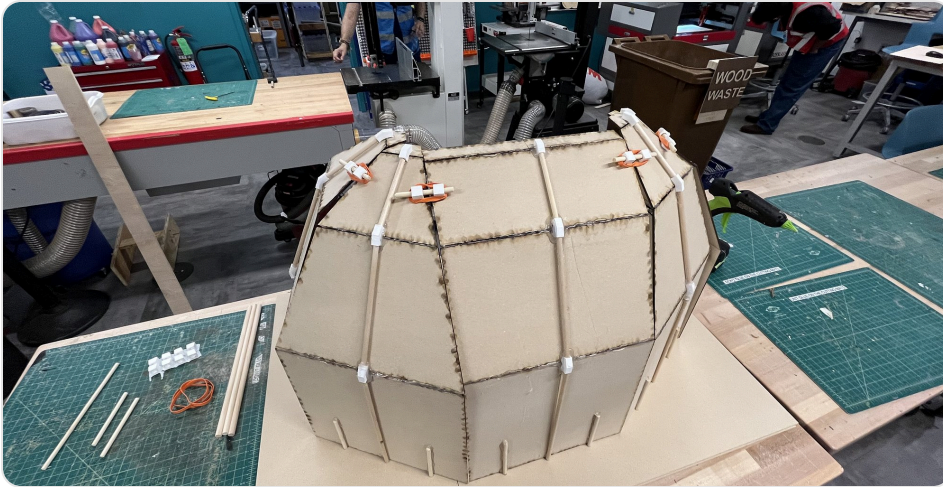
Northeastern Electric Racing, Aerodynamics subteam, Spring 2026 – Present

Buddhist Tzu Chi Foundation, Volunteer, Jun 2024 – Jun 2025

1. Solar Tracker Arcade

Feb – Apr 2026 · 5 students, Cornerstone of Engineering (Project 2)

A kid-sized Plants vs. Zombies arcade exhibit that teaches why solar panels tilt. A joystick aims a servo-mounted sunflower at a lit LED sun on a 91 cm MDF dome.



91 cm

Dome span

440

Addressable LEDs

24

Printed brackets

88%

Mid-game quiz correct (53 / 60)

What I did

- Wrote the LED sequence code with Owen (blue target, green marker for where the panel points), then debugged the LED and servo code with Owen and Antonio until the lag and the panel-to-light sync were fixed
- Built most of the game's screens in MATLAB, including the start screen, difficulty select, quiz and score screens
- Built the proof of concept: a joystick read in MATLAB and sent over USB serial to a Raspberry Pi Pico driving the pan and tilt servos, and filmed the demo video
- Wrote the pre-game and mid-game quiz with Jackson and planned the evaluation, mapping each question to a learning goal
- As Milestone 4 project manager, broke the work into the Gantt chart, assigned owners, ran the mentor meeting, and wrote the memo and presentation

My role

Game software and electronics.

Teammates Jack and Owen designed and built the dome and brackets; Antonio drew the wiring diagrams. I was also project manager for one of the course's eight milestones.

Skills

Raspberry Pi Pico, MicroPython, MATLAB, Serial communication, Debugging, Servo control, Project management

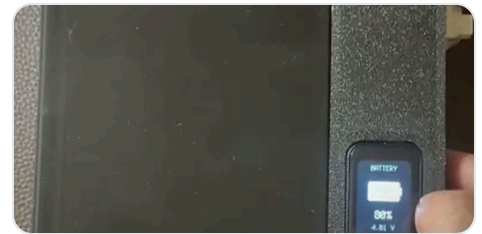
What broke: The first version lagged: the servos trailed the joystick and the LEDs fell behind the panel. The final code sends short serial lines to each Pico, the LED Pico polls for input instead of blocking, and its link runs at 921,600 baud instead of 115,200.

Full write-up: evanmah.com/projects/solar-tracker-arcade

2. Phone Box

Aug – Oct 2026 · Solo personal project

A 3D-printed phone lockbox with a servo latch, a touchscreen countdown and a tunable emergency override, paired over Bluetooth to an iOS app. Working prototype.



\$44

Prototype parts cost

2

Enclosure revisions

5

Printed fit-test coupons

~1 A

Servo stall current, moved to the battery rail

What I did

- Designed the two-part enclosure in SolidWorks around the ESP32-S3 touchscreen board, servo latch, LiPo and switches, through two full revisions
- Printed fit-test coupons for the hinge, charging port, screen bezel, power switch and locking tab, and checked each against the real parts before printing the full box
- Found an intermittent servo failure on the bench, traced it to two causes (PWM timer drift and an undersized 3.3 V rail), and moved the servo to the battery rail with bulk capacitance
- Hand-modified a MAX17043 fuel-gauge board (cut traces, soldered and crimped it onto the existing I2C bus) for a battery percentage from a fuel-gauge chip instead of a raw voltage
- Wrote the behavior specs for the lock, override and call alert, and tested every firmware and app build on the device

My role

Mechanical design, printing, hardware and bench debugging by hand. Firmware, app and backend written by AI coding agents to my specs, reviewed and tested by me on the device.

Skills

SolidWorks, Design for printing, Fit-test prototyping, Power debugging, Soldering, I2C, BLE

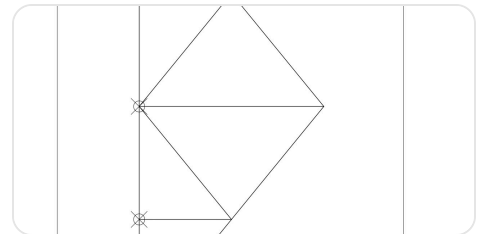
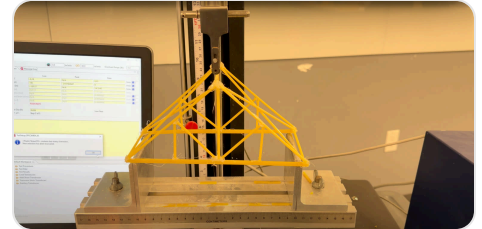
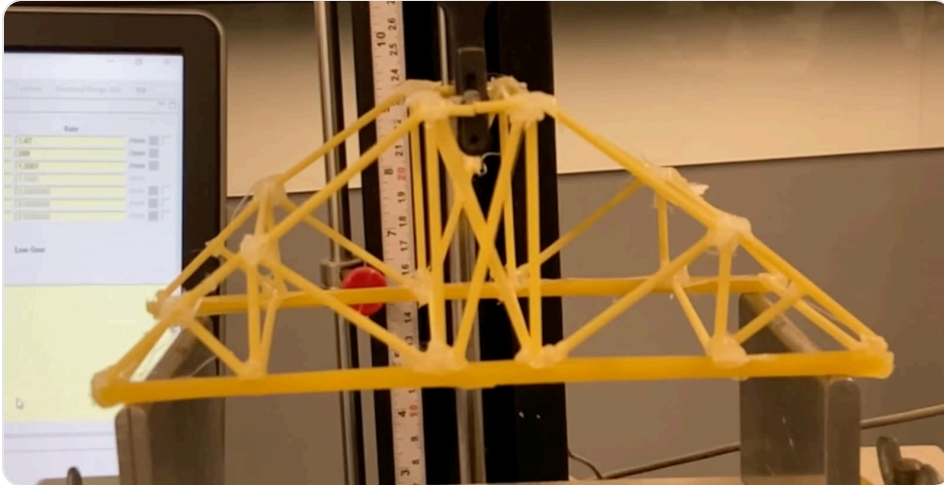
What broke: The servo failed only sometimes. Bench-cycling the lock showed two faults: the CPU clock scaled down under the servo's PWM timer, and the servo's ~1 A stall current sat on a 3.3 V rail that couldn't supply it. I re-assert the 50 Hz PWM before every move, moved the servo to the battery rail with bulk capacitance, and added a brownout counter that recovers the box on its own.

Full write-up: evanmah.com/projects/phone-box

3. Pasta Truss Bridge

Jan – Feb 2026 · 5 students, Cornerstone of Engineering (Project 1)

Five pasta truss designs for a LEGO astronaut's Mars crevasse crossing, tested to failure. Our merged design scored about 288 load-to-weight, second in the class, but was disqualified for being too low for the astronaut.



~288

Team design, load / weight

+52%

Over our best first bridge

66.9

My first bridge, load / weight

DQ

Too low for the astronaut

What I did

- Designed, drew in AutoCAD, built and load-tested a Waddell A truss (27 g, 22 cm span, 9 cm tall)
- Built the team Gantt chart and assigned owners and intermediate deadlines

My role

Schedule and task owner, plus my own Waddell A truss design, AutoCAD sketch, build and test.

Skills

AutoCAD, Truss design, Load testing, Failure analysis, Project scheduling

What broke: Our merged bridge scored about 288 load-to-weight, second in the class, then failed the minimum-height rule for the LEGO astronaut. We had chased the ratio and stopped checking a rule we wrote down at our first mentor meeting. My schedule had design, build and test milestones but no requirements check; every timeline I've made since has one.

4. Formula SAE: carbon-fiber mold tooling

Northeastern Electric Racing · Spring 2026 – present

I make the carbon-fiber molds for the car's body panels and wings: from the CAD model, through CAM and the CNC router, to a mold surface ready for infusion.

- Model aerodynamic molds for body panels and wings in SolidWorks
- Generate CAM toolpaths in Fusion 360 and check and fix the G-code before each CNC router cut
- Machine and assemble multi-piece molds, and prep mold surfaces for carbon-fiber vacuum infusion
- Routed 5+ carbon-fiber molds for the FSAE car, from digital model to finished mold

Skills: SolidWorks, Fusion 360 CAM, CNC routing, Composite tooling, Design for manufacturing

Software side project: Lumen

An open-source desktop browser with an AI agent in the sidebar that can operate the browser, with an approval step before anything risky.

My role: Product owner, specs, test design, review and releases. AI coding agents wrote most of the code.

- Specified each feature and the rules for when the AI must ask before acting
- Wrote performance budgets (startup, bundle size, Settings open time) as tests that fail the build
- Ran parallel AI coding agents, one per feature branch, with CI on every change before it landed
- Shipped Windows and signed macOS builds with a self-updater

Full write-up: evanmah.com/projects/lumen