WindPack

North Carolina State University
Design Final Report
April 23rd, 2026

Faculty Advisor: Darius Carter
Team Lead: Luke Caffarey

Mechanical Team:
Lead: Gabriel Cruz
Taylor Downes
Braddy Briggs
Russell Lapham
Edwardo Velarde
Noah Burch

Electrical Team:
Lead: Chaudry Khan
Pio Pezzi
Dylan Thunell
Shadi Hajiabbaszadeh
Anirudh Arunprasad

Team Contact Information
Windpack-org@ncsu.edu

Special Thanks:
Joseph Deneke
Jonathan Rabe

Table of Contents

1. Executive Summary

The Windpack team at North Carolina State University is proud to present its inaugural turbine design for the 2026 Collegiate Wind Competition. Developed within a single-semester timeline, our design philosophy is centered on Modularity, Simplicity, and Reusability. We utilized a "Clean Sheet" approach to engineer a 45 cm HAWT featuring an A18 airfoil and a D6374 150kv generator. While a late-stage mechanical alignment issue during initial testing prevented the completion of our full power-mapping matrix, the modular design allowed us to identify the root cause, *rotor eccentricity*, and establish a clear remediation path for competition.

2. Design Overview

2.1 Design Objective

To build a reliable turbine capable of generating power between 5–11 m/s while surviving a 13 m/s survival case. Our one-semester goal was to prioritize a simple architecture that supports the learning curve for new members while maintaining competitive performance.

2.2 Previous Year Designs

North Carolina State University did not previously compete in the CWC and spring 2026 is actually our first semester as a club. As such, our design was mainly based off of the designs of the 2025 CWC winners with our main inspirations being Cal Maritime, Cal Poly, and CU Boulder. We tried our best to learn what worked as well as what didn't from each team's turbine design report and mixed different elements from each school, such as pitch system, blade design, and braking system in order to create our turbine for this year.

3. Technical Design

3.1 Mechanical Systems and Load Analysis

3.1.1 Blade Analysis

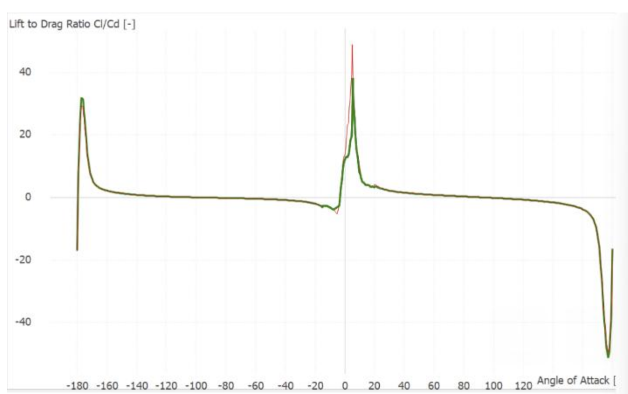


Figure 1: Polar analysis of the A18 airfoil showing lift-to-drag ratio.

Our aerodynamic design process began in QBlade CE v2.0.9.4, where we analyzed multiple profiles before selecting the A18 airfoil, inspired by its highly effective lift-to-drag ratio documented in the 2025 competition. Once the aerodynamic geometry was finalized, the optimized blade coordinates were exported into SolidWorks. This allowed us to independently design a custom root attachment tailored specifically to our active pitch system.

During our initial manufacturing phase, we prototyped the blades using standard PLA. However, after comparative physical testing, we determined that standard PLA was too brittle and lacked the necessary durability for high-wind testing. We subsequently transitioned to ABS plastic, which provided superior impact resistance and the flexural strength required to survive the 13 m/s durability test. To interface with the pitching mechanism, these ABS blades were engineered to securely slide over a 6mm rotational shaft and are mechanically locked in place using an M3 set screw.

Figure 2: Evolution of blade prototypes from PLA (bottom) to final ABS (top).

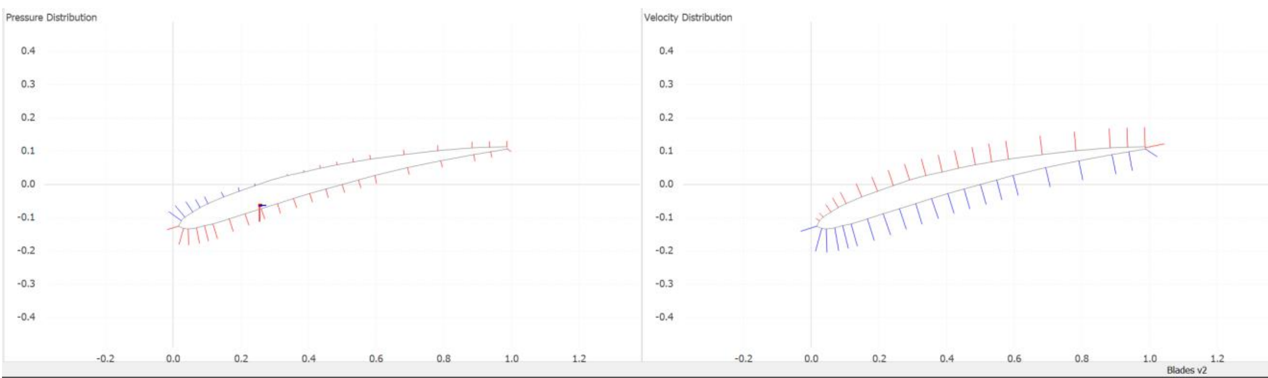


Figure 3: Simulated pressure and velocity distributions across the A18 airfoil at design speed.

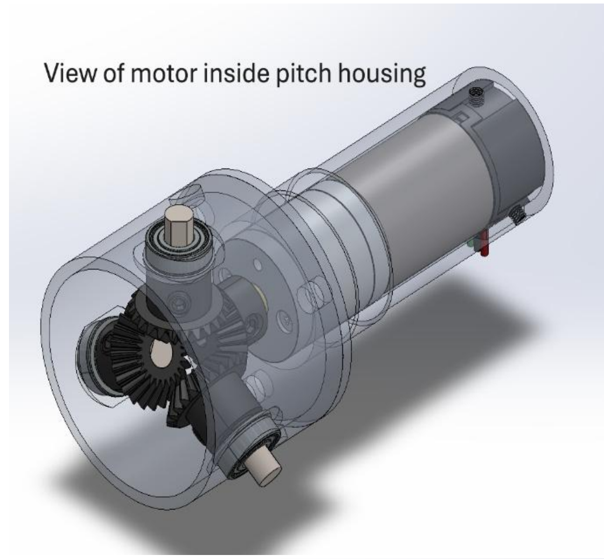
3.1.2 Pitch System

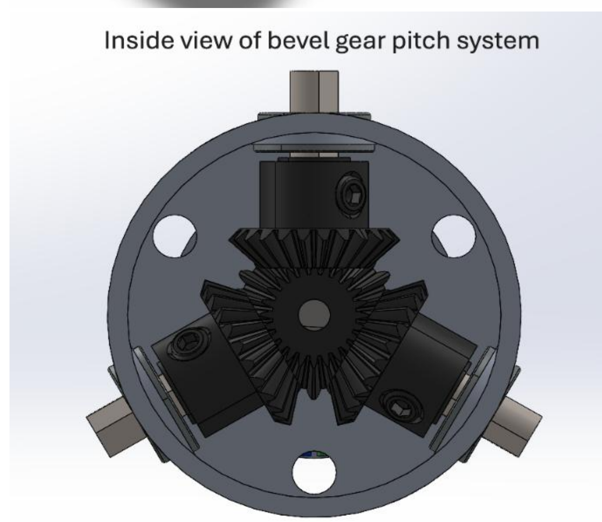
Our design evolution was driven by the need for active control. While our first iteration utilized a helicopter-style swash plate, it lacked the necessary sturdiness and was difficult to adapt for a three-blade configuration. Our second iteration utilizes a bevel gear system, which fits within our 0.04-meter radius hub.

To ensure the pitch motor could overcome the aerodynamic forces of the wind, we used the moment coefficient found in our QBlade simulation (recorded as negative 0.106) to calculate the resisting torque.

Using an air density of 1.189 kilograms per cubic meter and additionally a maximum wind speed of 14 meters per second, we determined that the total torque on the drive gear is 0.0524 Newton-meters. We selected a Pololu 499:1 Metal Gearmotor because its stall torque of 3.04 Newton-meters provides a factor of safety of approximately 58.

To validate how the air actually behaves across the blade at this load, we analyzed the pressure and velocity distributions (see Figure 3). These distributions confirm that our airfoil maintains a strong pressure differential without significant stall, ensuring reliable operation up to our estimated speed of 3500 RPM.


View of motor inside pitch housing


Inside view of bevel gear pitch system

$$T_{blade} = \frac{C_m \rho}{2} \int_0^R (U_\infty^2 + (r\omega)^2) * L_c^2 dr$$

This equation (taken from CU Boulder) determines the maximum torque that the wind blade will produce that resists our need to pitch. We need to know this in order to decide how strong of a motor is needed as well as gear ratio. If we go for a weaker motor then we need a higher gear ratio. This complicates things and since we have a small volume to fit the gears in we want to keep a gear ratio of 1:1

For the equation above we used the following values:
Cm is moment coefficient of the blade = .106
Rho is density of the air, competition is at Penn State at 1154 ft above sea level and air density = 1.189 kg/m^3
R is length of blade = .18 m
U is wind speed, top speed faced at survival phase of competition is 13 m/s so to make sure we can handle that we will assume windspeed is 14 m/s
Omega is the angular velocity which is 3500 rpm = 366.52 rad/s
L is chord length. Chord length is not a function but split into 100 data entries in Qblade so we used Riemann sum to approximate the length.

That is the torque for an individual blade so afterwards this must be multiplied by 3 to see how much torque will be exerted on the drive gear connected to the motor.

Total torque on drive gear is .0524 Nm so needed a motor that could handle that load. We settled on the Ponolulu 499:1 Metal Gearmotor 25Dx73L mm LP 6V because it had the torque we needed and came in a small form factor that fit within the .04 m radius hub. This motor had a stall torque of about 3.04 N-m which met our requirements and gave us a factor of safety of approximately 58. Once the motor was decided a housing was modeled in SolidWorks around it making room for the gears as well. The motor was put in the housing through the back for easy assembly and disassembly. To keep the motor in place, we added 3 set screws around the housing to ensure the motor turns with the rest of the housing.

3.1.3 Tower & Foundation

To accommodate our single-semester timeline, we prioritized simplicity, modularity, and reusability. The tower consists of a carbon-steel pipe TIG-welded to a machined steel base disc. We implemented an extruded aluminum rail for the nacelle, allowing the generator and pitch housing to slide independently for precise center-of-gravity tuning. We then 3d printed a clamp that holds the aluminum rail and pipe together.

Figure 4: Picture of Stanchion holding Aluminum Rail and Tower Together

To prevent dynamic resonance during testing, we engineered the tower, so its fundamental natural frequency (f_n) remains safely above our maximum operating speed of 3500 RPM (58.3 Hz). To accurately calculate this, we utilized the kinetic energy equivalence method to convert the continuous distributed mass of the tower into a discrete, single-degree-of-freedom system.

By combining the transverse stiffness of the carbon-steel pipe with the equivalent tip mass, which adds exactly 33/140 of the tower's distributed mass (m_b) to the concentrated mass of the nacelle assembly (m_t), we arrived at our final design formula:

$$f_n = \frac{1}{2\pi}\sqrt{\frac{3EI}{L^3\left(\frac{33}{140}m_b + m_t\right)}}$$

For a value of a natural frequency of 61.2 Hz, or 3,672 rpm. Well beyond what we realistically expect.

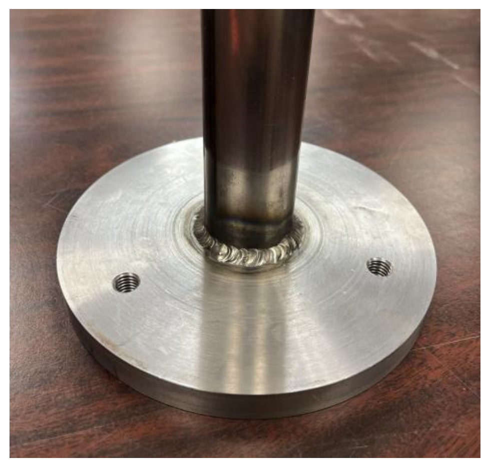


Figure 5: Picture of TIG-weld between mounting plate and tower.

3.2 Electrical Systems

3.2.1 System Architecture

Our electrical architecture is divided into three distinct functional blocks: power generation, load optimization, and control logic. The system begins with a mechanical-to-electrical conversion via an AC alternator. This 3-phase AC power is routed through parallel electronic braking circuits before being converted to a direct current (DC) Point of Common Coupling (PCC) by a 3-phase rectifier. Following rectification, the load optimization stage utilizes a buck-boost converter and a fixed high-wattage resistor, managed dynamically by a Maximum Power Point Tracking (MPPT) algorithm.

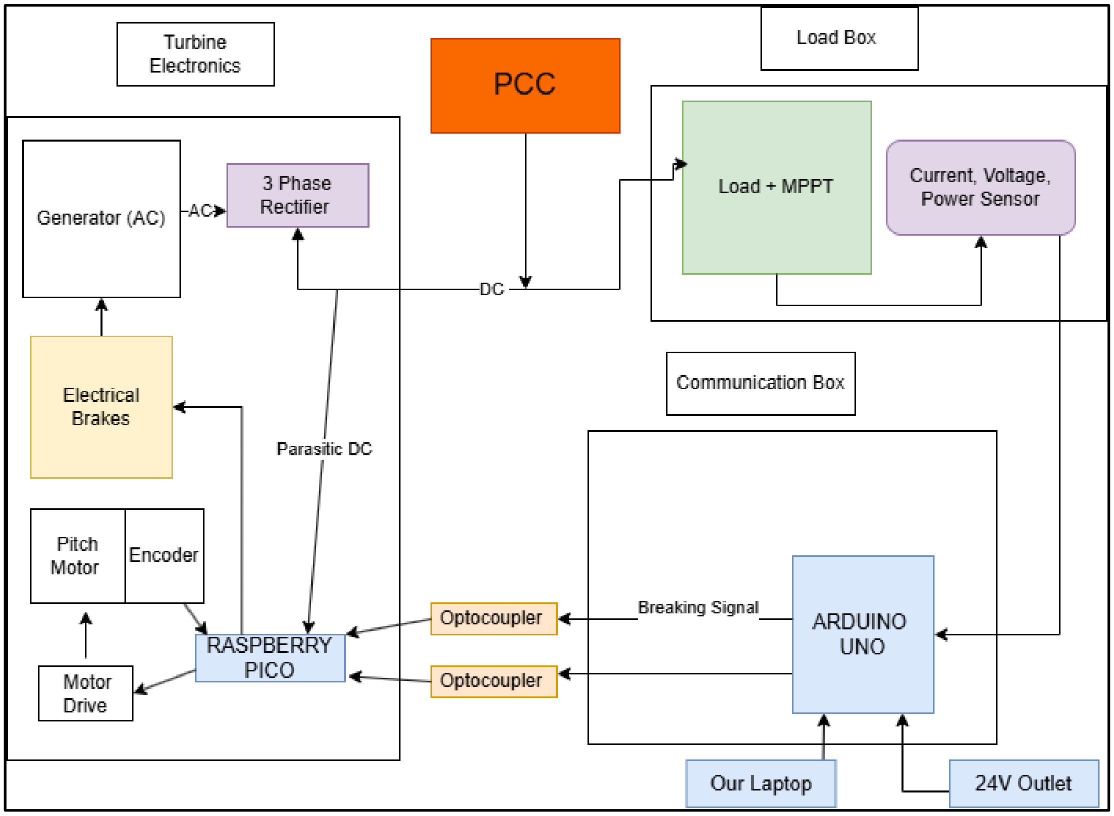


Figure 6: Diagram of Electrical Layout

3.2.2 Generator Specifics and Three-Phase Braking

The D6374 motor features 7 pole pairs and a functional voltage constant (K_V) that yields approximately 1V per 150 RPM, perfectly aligning with our 5 m/s to 11 m/s operational wind range.

During early prototyping, a single-phase braking strategy was attempted. However, this introduced severe asymmetrical magnetic loading, resulting in violent torque ripple and mechanical vibration that threatened the drivetrain's structural integrity. To resolve this, we integrated a Three-Phase Short-Circuit Brake into our custom PCB. To keep fabrication simple and modular, this new board architecture simply consolidates three of our earlier single-phase braking circuits onto a single footprint, alongside a dedicated mounting spot for the 3-phase rectifier.

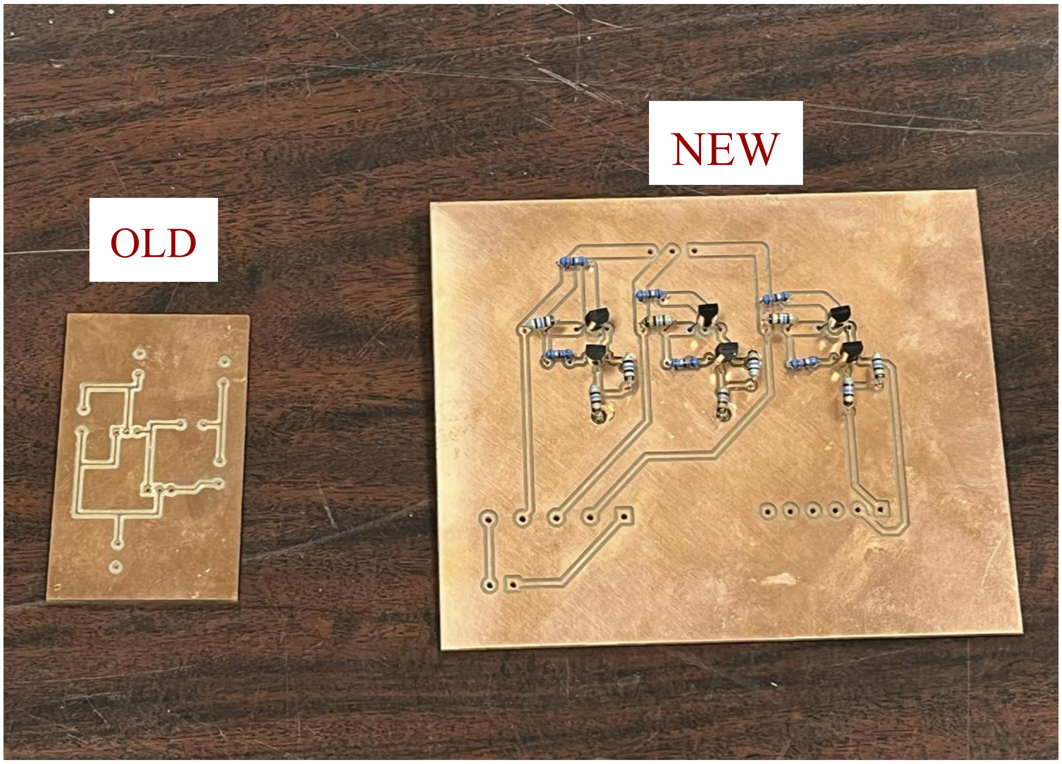


Figure 7: Picture Comparing New and Old Milled PCBS.

Activated by a pre-biased NPN BJT driving high-power N-channel MOSFETs, this system simultaneously shorts all three generator phases to ground. The resulting balanced braking torque (T_b) effectively stalls the rotor:

$$T_b \approx \frac{V_{gen}^2}{R_{short} \cdot \omega}$$

Where V_{gen} is the generated voltage, R_{short} is the negligible resistance of the MOSFET short, and ω is the angular velocity. This transition to a three-phase short eliminated the mechanical shock and drastically increased the lifespan of the turbine.

3.3 Load Management and Power Sink

In alignment with our commitment to reusability, the Windpack team opted to repurpose a legacy solar Maximum Power Point Tracking (MPPT) controller, generously donated by EasySun, to serve as our turbine's primary load management system.

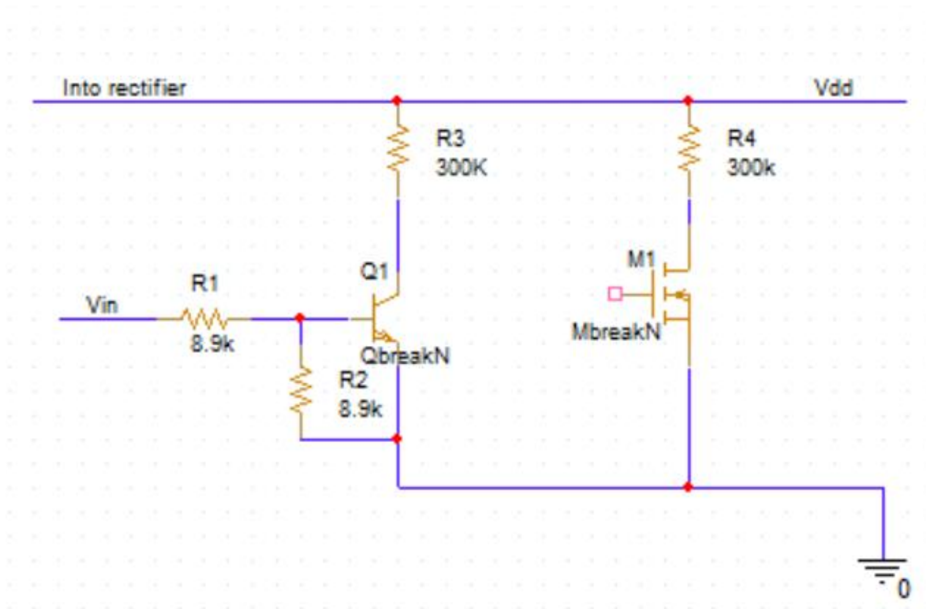


Figure 8: Diagram of braking circuit.

While solar and wind power profiles differ, the fundamental goal remains the same: optimizing the *V-I* curve to extract maximum power from the source. By utilizing this donated MPPT, we were able to implement a sophisticated power-tracking algorithm without the overhead of designing a custom DC-DC converter from scratch. The MPPT is interfaced with a high capacity 5Ω, 250W power resistor, which serves as our dedicated load sink.
This specific resistance value was selected to ensure the turbine can safely dissipate the full power generated at 11 m/s wind speeds while remaining well within the thermal limits of the resistor. Based on our power calculations:

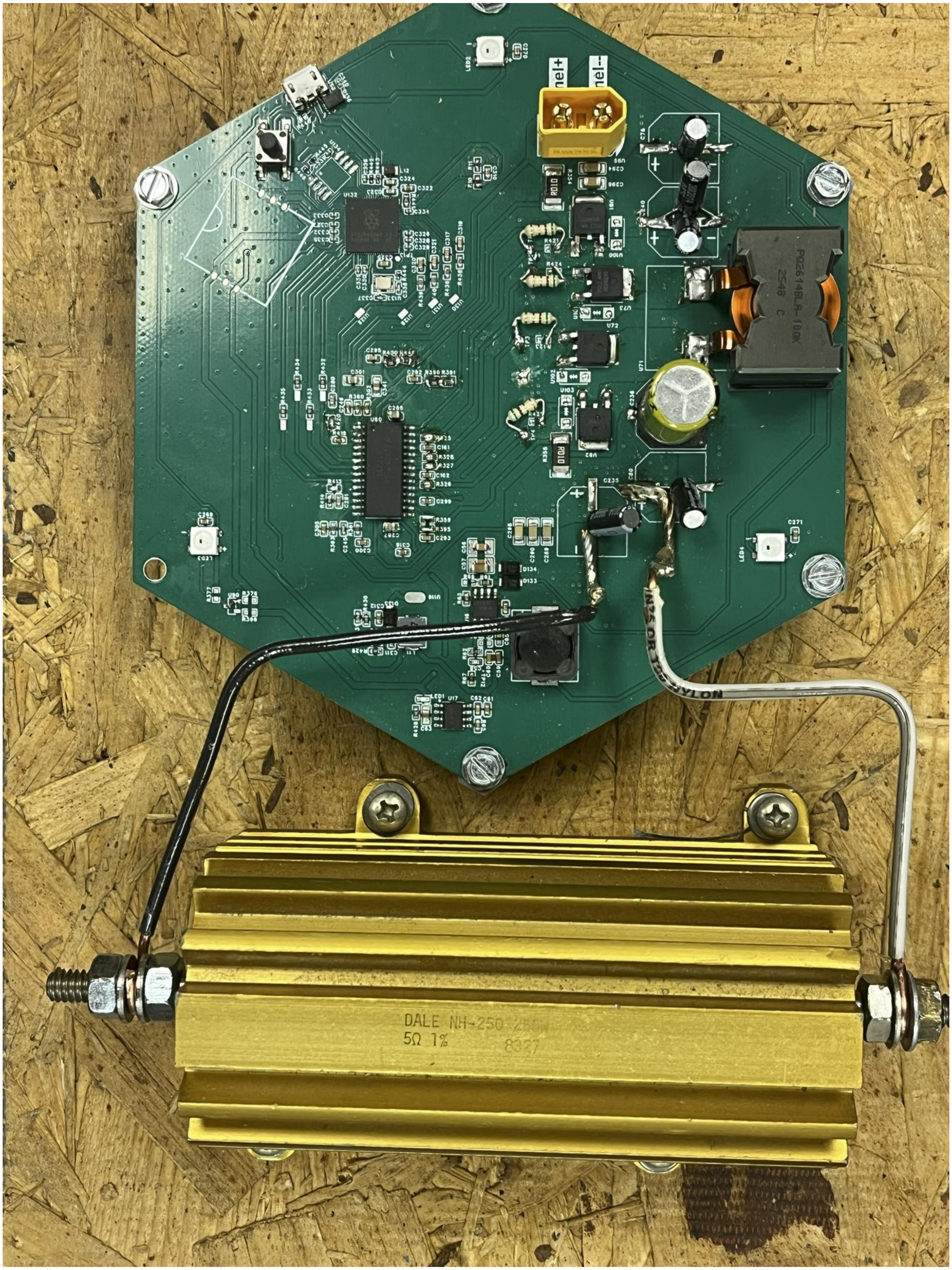


Figure 9: Picture of load box PCB and resistors.

3.4 Control Systems and Instrumentation

3.4.1 Primary Controller and Pitch Actuation

We chose an Arduino Uno R3 as the brains of our turbine, keeping it safely located on the load side of the PCC. It continuously reads our voltage and current numbers and uses that data to decide when to hit the brakes or adjust the blade pitch.

To handle the aerodynamics, we use a single pitch motor inside the rotor hub that rotates all three blades at the same time. The Arduino talks to a Texas Instruments L293DNE motor driver, which bumps up the low-voltage logic signals into the higher currents we actually need to physically turn the blades against the wind.

3.4.2 Sensor Telemetry and Fail-Safe Isolation

To monitor system health, we integrated an **INA226 sensor** that communicates with the Arduino via I²C. If the sensor detects a voltage spike beyond our safety threshold, common in gusty 13 m/s conditions, the Arduino triggers the braking sequence.

Safety was a huge priority for us, so we used **HP 6N139/138 optocouplers** to isolate the Arduino's 5V logic from the higher-voltage generator side. We designed this as a **hardware fail-safe**: the braking circuit requires a constant HIGH signal to stay disengaged. If the Arduino loses power or a wire gets disconnected, the signal naturally drops LOW. This instantly engages the MOSFETs to short the generator phases, bringing the turbine to a safe stop without needing any active software intervention.

3.5 Simulation and Modeling

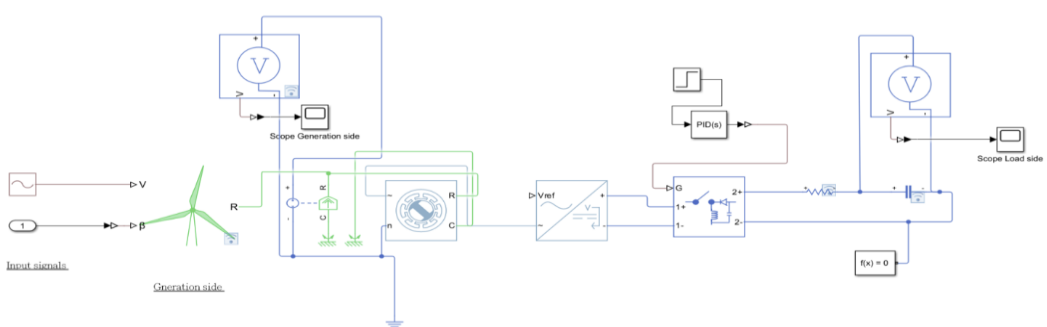


Figure 10: Diagram of Simulink Simulation

The beauty of the simulation was that it let us iterate rapidly; we tested various motor constants, different blade dimensions, and several control schemes to figure out what was actually buildable within our single-semester timeline. By virtually "swapping" components in the model, we could determine the most optimal setup for power generation without the cost and time of physical prototyping. Figure X shows one example of a proposed design we modeled, tracking the power path from the initial wind input and blade pitch through to the final rectified DC load.

4. Final Assembly and Team Organization

Our final turbine came together with three main systems: aerodynamics (blades and hub), mechanical (pitch, nacelle, and tower), and electrical (generator, braking, and controls). We mounted the blades, pitch assembly, and generator to an extruded aluminum rail using 3D-printed brackets and T-Slotted Fittings. This rail connects to the welded steel tower via a 3D-printed central stanchion to form our modular nacelle. The tower itself is welded to a heavy steel base disc, which has machined holes so we can securely bolt it to the wind tunnel floor.

To keep the workload manageable, we split the group into an Electrical Team and a Mechanical Team, each with its own lead. The mechanical side broke down into even smaller subteams to spread the fabrication tasks out fairly. Since this was our first year competing, making sure everyone was on the same page was a huge priority. We held weekly all-hands meetings to make big design decisions and help everyone see how their specific parts fit into the final turbine. We also had weekly subteam meetings to get hands-on and actually build the hardware. Day-to-day, we used slack and text.

5. Assembly and Commissioning Checklist

1	Mount the Tower	Bolt the welded base disc to the wind tunnel plate and check that everything is rigid and plumb.
2	Install the Nacelle	Place rail atop tube, then secure in place with stanchion. Ensure that it is facing directly into the wind. (Components should already be installed on railing)
3	Check the Rotor and Pitch	Attach the 3D-printed hub to the motor shaft. Move the pitch linkage by hand to make sure the blades sweep smoothly from 0° to 90° without catching.
4	WIRE IT UP!	Connect the generator's 3-phase lines to our custom braking PCB, and run the DC output over to the load box.
5	Boot Up	Power on the Arduino and make sure it's successfully reading the voltage sensor.
6	Test the Fail-Safe	While the turbine is sitting still, unplug the Arduino's power. Verify that the optocouplers drop and the short-circuit brakes automatically engage.
7	Final Clearance	Do a quick visual check to make sure all wires are tucked away and nothing is blocking the rotor's sweeping path.

6. Testing

We originally mapped out a comprehensive three-phase testing protocol to validate our Simulink models: checking our cut-in speed at 5 m/s, mapping the MPPT power curve up to 11 m/s, and finally proving our fail-safe brakes at the 13 m/s competition maximum.

But as any engineer knows, simulations only go so far. During our single wind tunnel time slot, as soon as the rotor started spinning up, the turbine started vibrating way more than we expected. Instead of pushing through and risking our 3D-printed hub shattering or snapping the tower mounts, we made the call to hit the kill switch and abort the run.
Because we had to cut the test short, we weren't able to fully map our MPPT curve or physically test the high-speed survival braking under heavy aerodynamic loads. While it was a tough break, that brief tunnel attempt gave us incredibly valuable real-world data.

We sourced the issue to our 3D-printed parts. They just weren’t aligning perfectly due to some minor oversights in our CAD models.

Figure 11: Gabriel Cruz Setting up Turbine for our Failed Test

References

Amrita Vishwa Vidyapeetham. (n.d.). *Wind turbine characteristics*. VLab. https://vlab.amrita.edu/?sub=3&brch=175&sim=1078&cnt=1

Department of Energy. (n.d.). *How a wind turbine works - text version*. Office of Energy Efficiency & Renewable Energy. https://www.energy.gov/cmei/systems/how-wind-turbine-works-text-version

Dursun, E. (2016). Determination of wind turbine power coefficient using a simple method. *International Journal of Intelligent Systems and Applications in Engineering*, *4*(Special Issue-1), 163–166. https://dergipark.org.tr/tr/download/article-file/303217

Instructables. (n.d.). *L293D motor driver*. https://www.instructables.com/L293D-Motor-Driver/