Final Turbine Design Report

Collegiate Wind Competition 2022

Wildcat Wind Power at Kansas State University

Officers

Eric Christman – President, Student Lead
Hayden Dillavou - Vice President, Student Lead
Brianna Wagoner - Secretary
Caleb Stuber -Treasurer
Jakob Long – Project Development Lead
Lane Lundeen – Connection Creation Lead
Matthew Monsion – Electrical Design Lead
Michael Brosseit – Electrical Design Lead
Kavian Kalantari – Mechanical Design Lead
Josh Meurer – Mechanical Design Lead

Advisor

Dr. Hongyu Wu, Associate Professor, Electrical and Computer Engineering

Mechanical Design Team

Kavian Kalantari
Josh Meurer
Chase Eccles
Jacob Lowe
Rafael Lopez Barraza
Stephen Kielhofner
Wesley Smith

Electrical Design Team

Matthew Monsion
Michael Brosseit
Brianna Wagoner
Andrew DuLac
Andy Freshnock
David Pierson
Kent Detarding
Macie Sexten
Patrick Flett
Tyler Schooley

Contact

echris018@ksu.edu or hongyuwu@ksu.edu

Mike Wiegers Department of Electrical and Computer Engineering
3108 Engineering Hall
1701D Platt St.
Manhattan, KS 66506
785-532-560

Wildcat Wind Power dedicates our work this year to our longtime advisor, Dr. Warren White, who passed away in May of 2021. Dr. White cared deeply about our mission, and his presence is sorely missed.

Table of Contents

Executive Summary

This report details the design, construction, and testing of a small-scale wind turbine for the 2022 Collegiate Wind Competition, by the Wildcat Wind Power team at Kansas State University. Our primary design motivation was to design a wind turbine that could withstand the more rigorous testing conditions specified in the competition this year, and we developed a variety of control features to ensure that the turbine would meet the necessary operational requirements.

We used in-house analytical tools to design blades that would maximize the turbine's performance along its power curve, and continued to utilize 3D-printing for blade and nacelle construction. One of our most critical design efforts was that of an active pitching system, which used a pair of linear actuators in the turbine nacelle to pitch each of the three blades through a bearing which encircled the driveshaft. The active pitch system, which was also constructed out of largely 3D-printed components, provides more than 100° of rotation to each blade, and is used to regulate RPM and power production at high wind speeds. The turbine foundation, a new addition to the competition this year, was designed as an open-topped box constructed out of steel sheets and a steel plate, to be filled with sand during installation before connecting the competition stub piece.

The turbine electrical systems include sensors and controls on both the turbine and load sides of the PCC, with each unit containing its own microcontroller and mounted on a PCB. The turbine-side system contains a three-phase bridge rectifier, switching converters to power the various controls and a microcontroller, and the ability to receive power from the load during braking procedures. The load-side system contains a similar arrangement of sensors, converters, and microcontroller, as well as a variable resistance load system designed around MOSFETs. The two systems communicate through isolated UART communication lines, and share data regarding the position of the active system, electrical variables like power and voltage, and control variables like the load resistance and braking conditions.

Software for the system runs on Teensy microcontrollers for both the load and turbine systems, and tracks the turbine's operation to force it to operate through its simple control scheme. The load system was designed to communicate with our wind tunnel for testing purposes, which allowed for iterative, automated testing to collect data for power curve and regulation parameters.

Design Objective

The overarching objective for this design was to create a scaled wind turbine to operate in a wind tunnel that mimics the operational parameters of fixed-bottom offshore turbines. Key design parameters include a shallow-water foundation and turbine operation at any wind speed up to 22 m/s. These parameters create new requirements for turbine durability and reliability, introduce new manufacturing techniques for the foundation, and provide for increased control flexibility as turbine power and RPM regulation is not measured outside of the safety task. We see the increased 22 m/s wind speed as the primary motivator for design decisions. In building a regulation system, we prioritized the utilization of the system to optimize the turbine's power production throughout its power curve and minimize its cut-in wind speed, while ensuring it could limit turbine rotational speeds when needed. In building a foundation, we wanted to remain cautious of the destructive ability of the wind, and set out to build a foundation that could keep the turbine stable at 22 m/s, even if it weighed

more than a leaner design. Non-operational aspects of our design objective include designing a turbine and foundation that was easy to maintain and install, designing an electrical system that could operate full-system testing through a single control interface, and building a complete system that followed all design requirements set in the competition Rules and Regulations.

Mechanical Design

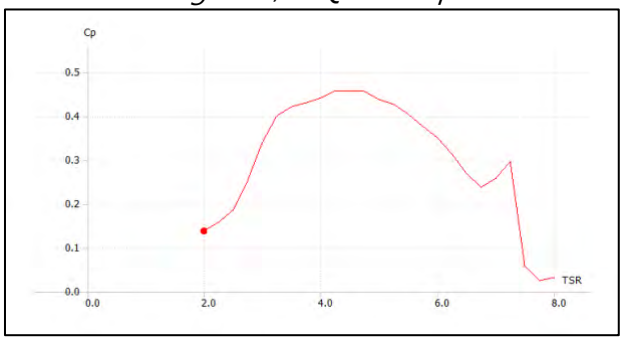
Blades

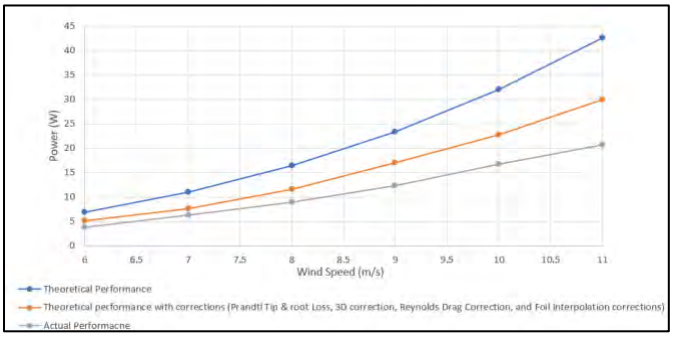
For this year's competition, we wanted to redesign our previous year's blade with more optimized airfoils and blade angles. Based on our research into blade element momentum (BEM) theory [1], we designed our new blade around a constant tip speed ratio (TSR), as the angle of airflow over a blade element at any wind speed is constant if the TSR is constant. To find the best airfoil at every blade element, we decided we would simulate every airfoil at every blade element position and write a program to select the best performing airfoil for each position.

Originally, we planned to use QBlade to run airfoil simulations, but a second piece of software was needed to calculate the Reynolds numbers and Mach airflow speeds for each simulation. MATLAB was chosen because of our familiarity with the software, and the simplicity of coding arrays in MATLAB compared to other coding languages. We designed our MATLAB program to calculate the Reynolds numbers and Mach speed for each blade element based on the number of blade elements, minimum and maximum blade radius, chord lengths, wind speed, and TSR. The program calculates data for QBlade extremely quickly, which along with QBlade's multi-threaded batch analysis, expedited the otherwise very lengthy process of running simulations for each airfoil. By exporting simulation data from QBlade as comma separated value (CSV) files, we could input the simulation data directly into MATLAB for analysis. We faced some problems with the simulation data importing correctly, but those problems were solved by writing a simple python script to edit the format of the CSV files into a format MATLAB could more easily read. Our MATLAB program compares each airfoil based on the maximum coefficient of force it produces in the direction of rotation, and outputs the best airfoil and angle for each blade element along the blade. Additionally, the program also keeps track of the optimum angle and calculates the ideal blade twist for each blade element along the blade.

Our analysis focused on NACA airfoils. These airfoils are generated from only 3 variables: the maximum camber, position of maximum chamber, and the maximum thickness ratio [2]. With only three variables, NACA airfoils allowed us to test a wide variety of different airfoils very quickly. The optimum blade according to our analysis is made of the airfoils: NACA 6809, 6806, 6206, and 6306, and has an average maximum coefficient of lift to drag ratio of 35.9. The program also calculated the optimum twist angle for each airfoil, considering the optimum angle of attack as well as relative wind angles. We chose to compare the performance of every airfoil at a wind speed of 8 m/s, as it was halfway between the maximum and minimum wind speeds used in the power curve test. Also shown is the actual power production, in comparison to the theoretical maximum production, and theoretical production when considering a number of different loss types. We recorded an increase in our power output from last year of 17.5 W to 20.5 W at a wind speed of 11m/s.

Figure 1, 2: QBlade Cp-λ Curve at 8 m/s, Theoretical vs Actual Performance





We chose to 3D-print the blades as it was the easiest and fastest way to produce the geometry accurately. Below is the 3D model in SolidWorks and the same blade 3D-printed.

Figure 3, 4: Blade SolidWorks Rendering, Photo

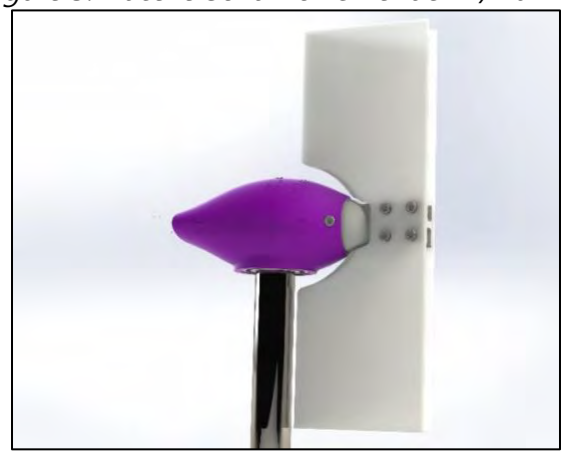
Calculation of the turbine annual power production was performed with data taken from the Galveston offshore area analyzed in the Project Development portion of the competition. The turbine power curve, from testing, was compared to a 5-minute time series of wind speeds in the location, and a final number of 91.74 kWh of yearly production was calculated. We did compare this metric to an onshore location in North Texas for the same year and hub height, for the sake of analysis, and found similar yearly production at 91.87 kWh.

Nacelle and Yaw Mechanism

The nacelle design was largely determined by the components that the nacelle must enclose or connect to, largely the active pitch hub mechanism, generator, and linear actuators. Our original nacelle design was cylindrical, with a rounded face facing the wind and a large acrylic tailfin on the opposite side. It had to be wider than we would have liked, due to the linear actuators for the active pitch system mounted on either side of the generator. The fin, coupled with a bearing between the nacelle and turbine tower, allow for simple passive yawing to ease in installation and testing. Comparison of the turbine testing results to simulated blade data showed that the turbine was not as efficient as it could be, and we believed this may have been due to large frontal area of nacelle. We attempted to reduce

the area by narrowing the top and bottom of that face, and smoothing the curve up to the top of the nacelle. This improved design, along with the acrylic tailfin and a connector piece, is shown below. The turbine tower was recycled from previous years, and is an aluminum tube welded to a flat aluminum base, with holes drilled for connection to the competition stub, and the yaw bearing set at the top.

Figure 5: Nacelle SolidWorks Render w/ Tail Fin

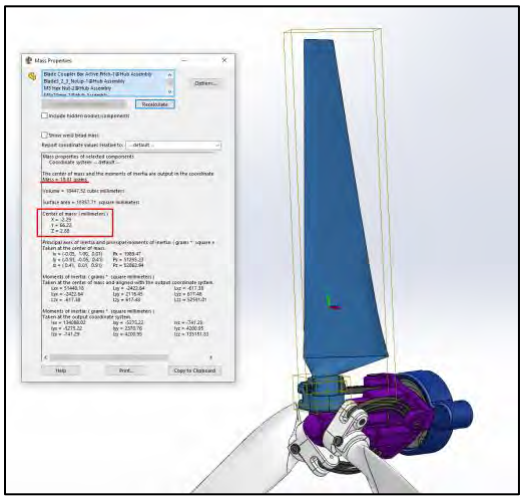


Mechanical Loading

Due to the nature of conducting stress analysis of 3D-printed parts, much of the review of our turbine relies on testing new designs in our wind tunnel, then making appropriate modifications and retesting. For example, one of the major issues with creating a finite element method (FEM) model of a 3D-printed part in SolidWorks is the way infill percentage is handled. Infill refers to the percentage of the inside of a part that has mass and is not empty space. To save time, material, and weight, parts are printed with a preset number of perimeter layers, and then the remaining cavity is filled in based on the given infill percentage (meaning that most of the interior is hollow save for some support structures). A known workaround to modeling the physical part involves creating a shell of the modeled part, making it completely hollow. This is usually the best approximation for parts created with low infill percentage, but it still doesn't consider the small, stiffening, webbed structures that make up the interior. Another issue with conducting an analysis of a 3D-printed part comes from how the layers are stacked. For traditional fused deposition modeling (FDM), the major weakness is along the layer lines. As a part is created, weaknesses are formed between layers due to the temperature differential in the cooled previous layer and the hot, viscous, current layer. This weakness makes print orientation a major factor for loading direction, as having these layer lines perpendicular to the major forces on the system make the part significantly weaker. Because of all these reasons, we mainly relied on physical testing data and component failure to be the driving factor for design considerations related to mechanical loading.

When testing different versions of our active pitch design, we noticed that one of the major points of failure involved the connection of the blades to the nosecone. With our mechanism, each blade is mounted to a bearing to allow for blade rotation, and then attached into a round peg on the nosecone. As RPM increases, the centrifugal force of the blades pulling on the nosecone was overcoming the glue we had used to keep it in place. This original design resulted in blades becoming detached from the turbine (catastrophic failure) at around 1500 RPM. Our target max RPM was 3500, so with some analysis in SolidWorks, using the center of mass of the blades relative to our axis of rotation, we translated this RPM into a max tension force that would be put onto the blades.

Figure 6: SolidWorks Mass Properties of Constructed Blade Assembly



$$T = mr_{COM}\omega^2 = \wedge.01881kg(.06626m)\left(\frac{3500\frac{rev}{min}}{60\frac{s}{min}} * 2\pi\frac{rad}{rev}\right)^2 = 170N$$

To remedy this issue and increase the maximum tension we could put on the assembly, we incorporated heat set metal inserts into the design. These inserts are heated (usually with a soldering iron) and then press fit into an undersized hole in a 3D-printed part, re-forming plastic around the grooves of the insert and solidifying it into the part.

Figure 7: Heat Set Insert Installed on Nosecone

We found through research that we could expect to resist a force of at least ~1300 N in tension on the insert [3]. The testing done doesn't incorporate the exact length, inner and outer diameter, or the surrounding geometry of our part, but we figured it to be a good enough estimate. This loading force was well over the 170 N that we expected to see at our target RPM from the simulation. This result was also backed up to an extent by the fact that we could max out our wind tunnel and run our turbine at over 4900 RPM, giving us a tension on the blade assembly of 330N.

Foundation System

The design of a fixed-bottom foundation presented a very new challenge to us this year, especially given our lack of experience in working with machined or welded components. We began the foundation design process by setting up a testing environment with sand in a plastic tub. After testing a variety of designs made from wood and other materials, we settled upon a design that resembled a box

with an open top, in which the tube that connected the foundation to the connection stub came up from the bottom of the box and out of the open top. The constructed design is shown in Figure 8. For installation, the wiring for the turbine will enter the top portion of the tube through the stub connector, and exit through a hole on the side of the steel tube 10cm from the top. This design weighs just under 8 lbs, which may be more than the weight of other team's foundations but should ensure it performs well at higher wind speeds.

Figure 8: Turbine Fixed-Bottom Foundation

The primary advantage of this design, which we first discovered by testing a flat base foundation without any sides, is that it uses the weight of the sand to anchor the turbine down, without weighing more than necessary as a standalone structure. The sides prevent the wet sand from acting like a fluid and shifting when the torque of the turbine is applied on the foundation due to the force of the wind. The sand cannot flow over the top of the anchor, and translation and rotation of the stub piece is prevented.

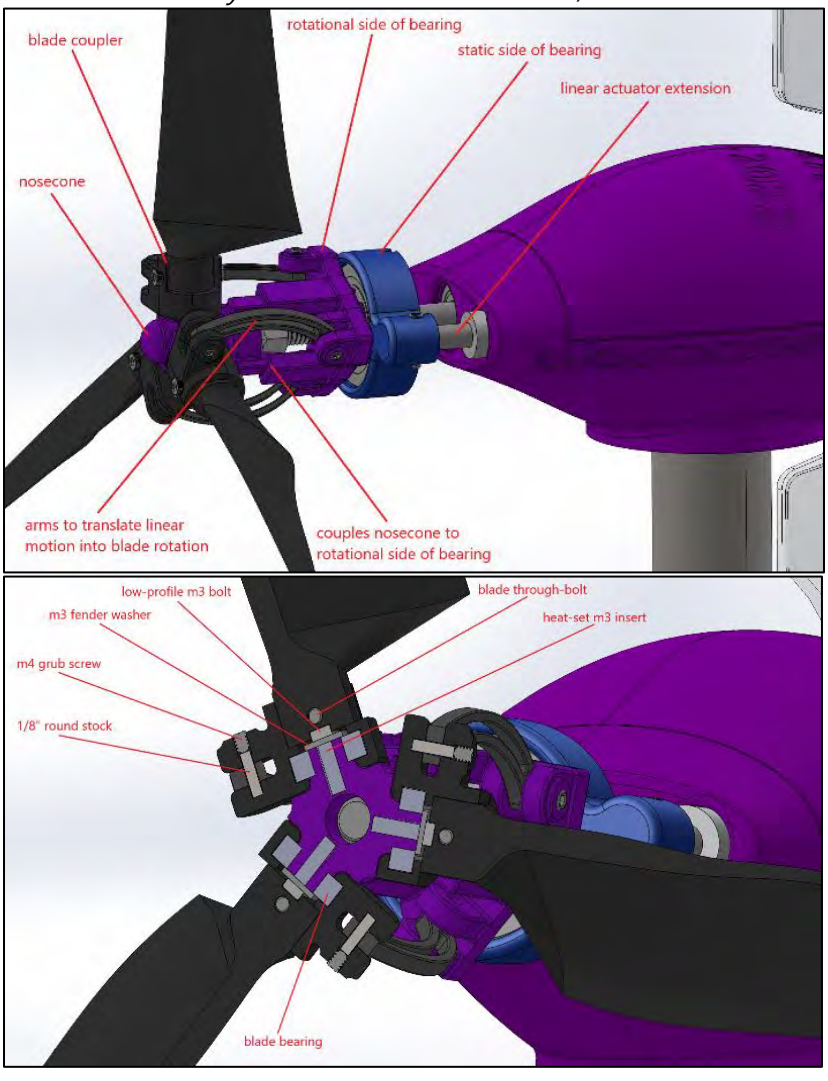
We considered many designs that included vertical plates parallel or perpendicular to the direction of wind flow and torque on the anchor, and tried to consider how the sand-water mixture would react to torque being applied on each foundation. We believe that it was imperative that the foundation contain some vertical structures perpendicular to the direction of wind flow, and the more cross-sectional area these structures have, the better resistance against torque they can provide. But, it can also be important for the foundation to have some resistance against any twisting motion, hence structures parallel to the wind flow need to be introduced as well. Even with all of these considerations, the best possible anchor would be an extremely heavy one that weighs the turbine down against any and all moment applied to the nacelle. We know, and our testing showed, that finding some way to use the weight of the sand to weight the turbine down would be critical. Our design was created to do this while minimizing the weight of the foundation itself.

Active Pitch System

The active pitch system is the turbine's primary mechanical actuator, and is the most technically significant subsystem design that we attempted this year. Originally, we wanted to implement an RC helicopter swashplate, but struggled to find one available for purchase that was rated for parameters

we expected the turbine to see, or had the same number of arms as the number of blades of our turbine. Through our research into swashplates and pitch systems, we also learned that our system did not need separate degrees of freedom for each blade, as a swashplate would, so an in-house system might require reduced complexity. We settled on a design in which two linear actuators, placed in the nacelle and aligned with the driveshaft, will push outwards on a plate that wraps around the rotating shaft. A bearing in the plate transfers this linear force to the rotating blades through curved arms connected to each blade. Each of the three blades are on their own bearing and free to rotate, and the outward or inward push of the actuators will pitch the blades over 100 degrees. Individual components of the assembly were 3D-printed, aside from the bearings and brass pin joints to attach moving components together. The linear actuators chosen were Progressive Automations PA-12 7.4V models, and while they proved difficult to program, they provide very strong precision in linear motion and perform well even under high windspeeds. A labelled SolidWorks screenshot of the full final design is shown below, along with a diagram of the blade coupling and connection.

Figure 9, 10: Active Pitch System Mounted on Nacelle, Blade Connection Diagram



Although the fundamental concepts of the design remained the same throughout the testing process, it went through many iterations before it performed as we had hoped. The first iterations of the design used print-in-place joints rather than discrete bearings and pin joints, and while they afforded a great deal of design flexibility, they produced too much friction during the pitching motion under load. Further iterations replaced the print-in-place joints with bearings and pins, which performed better under load, but oscillated during operation due to unequal loading between each blade and a high overall moment of inertia of the hub. The curved pitching arms were introduced to increase the total range of angles for the blades, and the size of the nosecone was reduced to limit oscillation. This system performed much better and operated much smoother under load. However, when wind speeds were increased, the blade bearings connections demonstrated a tendency to destructively slip out while the blades were pitching. To remedy this, additional fasteners were introduced to secure the blades to the nosecone, and the entire system held up much better at high wind speeds.

Electrical Design

Generator Selection

We selected a Nanotec DB42L01 Brushless DC Motor to act as the turbine generator, rated at 4000RPM and 75W. Brushless DC motors are widely available at the RPM and power ratings our turbine operates at, and tend to last longer than their brushed counterparts. The motor was selected after calculating the RPM and power values the turbine might see during operation at an efficient tip speed ratio and C_p. Although perfect RPM and power regulation would allow the motor to be sized to the values corresponding to 11 m/s operation, some leeway was given for imperfect regulation, and the motor was sized to values corresponding to 13.5 m/s operation. A smaller motor would be lighter and more efficient at lower RPM, which is why sizing was limited to this value. One benefit of this model is that it is equipped with a set of Hall-Effect sensors that output a signal alongside the main AC output. There are individual lines for each phase of the motor, and they produce a square wave in phase with the rotation of the motor. This signal is used for the measurement of RPM. The presence of the Hall-Effect sensors meant that we didn't have to design a standalone RPM sensor in the nacelle, as we have done in previous years.

Turbine Controls

The turbine-side control system includes a passive rectifier, power measurement sensors, communication equipment, and a microcontroller to run the controls and communicate with the turbine's other subsystems.

We originally planned to design and build a controllable (also known as PWM) rectifier for the competition turbine. We did succeed in building a working prototype that was able to efficiently rectify the three-phase AC signal from the generator into DC, and regulate its voltage by delaying the firing signals by adjusting phase. However, once we began to integrate it into the rest of the turbine-side systems, we discovered that it interacted very poorly with the control power circuitry designed to power the sensing and communications systems. The presence of a powered controllable rectifier on the system required a separate passive rectifier during start-up, and the rectifiers interfere with each other during normal operation, as well as interfering with the various buck-boost converters we used to power the rest of the controls. We were only able to make the decision to move away from the controllable rectifier after building and attempting to integrate it, hence its presence in previous design

reports. We chose to use a single monolithic diode rectifier instead, an IXYS chip mounted on the PCB that the active rectifier was designed on. The diodes conduct at 0.55V, a less-than ideal voltage but a necessary efficiency loss given that the passive bridge rectifier requires no external power source to operate. The signal produced sees minimal voltage ripple during normal operation, which ensures power ripple under 10% as required by competition specifications. We included two small LC filters, with total energy storage capacity under 2J, on the rectifier output to minimize the ripple as much as possible. A 5V buck/boost converter powers the turbine control systems from the output of the rectifier, and feeds directly in to the 7.4V boost converter used to power the linear actuators. The 5V rail has a 5V, 0.47F (5.875J) supercapacitor attached to it, which is charged during normal operation and used during the discontinuity safety task to briefly keep the controls powered while the pitch is pulled back.

The turbine-side system includes one major sensor, a power sensing INA260 IC. This IC measures current, voltage, and can calculate power. The INA260 reports this information to the turbine side system's microcontroller through the I2C protocol. Many breakout boards and libraries have been designed around the INA series ICs which made it very easy to implement a custom version right into our design. In addition to receiving signals from the hall effect sensor and I2C from the INA260, the turbine side microcontroller also has transmit and receive lines to the microcontroller on the load side system and to a TISN74 dual buffer/driver. The SN74 controls the two linear actuators which move our active pitch system. Using this IC allows us to have separate data transmit and receive lines going to the microcontroller but only one line to the linear actuators in the nacelle. On the other transmit line, information such as the RPM, voltage, current, power, and position of the linear actuators can be sent to the load side system via optically isolated UART. This information will help inform the load side system about what changes to make to either maximize power or control RPM. The load side system will then send instructions for the turbine side system on its receiving line.

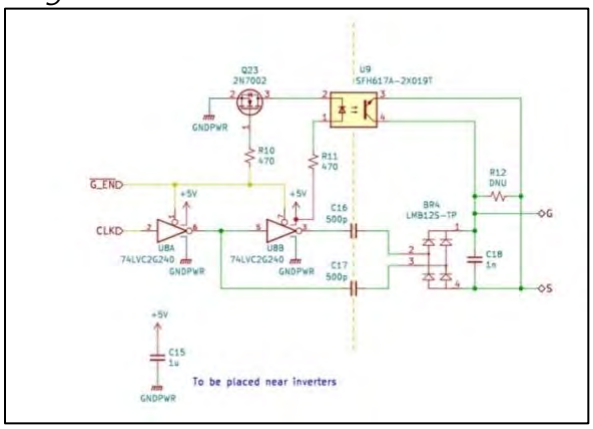
The main microcontroller chosen for the turbine was a Teensy 4.0. While the microcontroller is required to perform many tasks, the defining requirement was that it must send control signals to the linear actuators in the nacelle. The actuators were specified for their robustness and precision, not their communications compatibility, so they required a very large code library that took up a lot of space in the ATmega328P chips in the microcontrollers that we typically gravitate towards. The Teensy, with its larger memory and faster processor, can handle the actuator closed-loop controls much better, and thus was chosen for the turbine-side controls. The specific 4.0 model was chosen because of its smaller form factor and lower power consumption compared to the Teensy 4.1, which was used in the load side controls. Operating at 3.3V, both Teensys introduced a new level of complexity to the system, as we traditionally design sensors and controls at 5V, and the linear actuators require 7.4V.

Load Controls

We designed a variable resistance load for the 2020-2021 turbine, and consider it a genuine success. We wanted to build off that work, continue to innovate, and design a better variable resistance load for this turbine. It would provide the same benefits in optimization, regulation, and testing, although the overall control burden of this specific control system would be reduced with the introduction of the active pitch system alongside it. The load uses MOSFETs in combination with resistors increasing in a binary fashion, and creates different resistances by combining different combinations of resistors together. As the resistance of each set in the series doubles, a full linear range of resistances is achieved, ranging from 0.25 Ohms to 63.75 Ohms with a precision of 0.25 Ohms. Because the load must be able

to change resistance regardless of the power flowing through it, the FETs require a separate, isolated firing circuit (or gate driver circuit) to operate, which was the main challenge of implementing this system. The gate drivers accept a clock and enable signal as inputs. The clock line is common to all drivers and is switching at 200kHz by default. While the clock line could be used as a master disable to all the gate drivers, the enable pins (active low) are used to control the drivers. When the enable pins are set low, the back-to-back inverters begin switching at the clock frequency, which essentially pumps current through the 500pF coupling capacitors and bridge rectifier, which develops a DC voltage across the gate and source of the load FET. This voltage will allow the load FET to conduct through its channel, effectively shorting out the resistor associated with it. Conversely, when the enable pin is set high, the inverters cease to switch and the small-signal FET will allow current to pass through the diode of the optocoupler. The isolated NPN BJT of the optocoupler will then begin to conduct, which will tie the gate and source of the load FET together. This will cause the load FET to stop conducting through its channel and "unshort" the resistor associated with it. The gate driver was designed in this fashion (capacitively and optically isolated) so that the load and the driver circuit could use separate ground planes. Relays could have been used instead of FETs, but FETs are an overall more energy-efficient and modern component that reflects the sustainable nature of this competition. A schematic representation of the gate driver circuit is shown in Figure 11.

Figure 11: Gate Driver Circuit used in Load



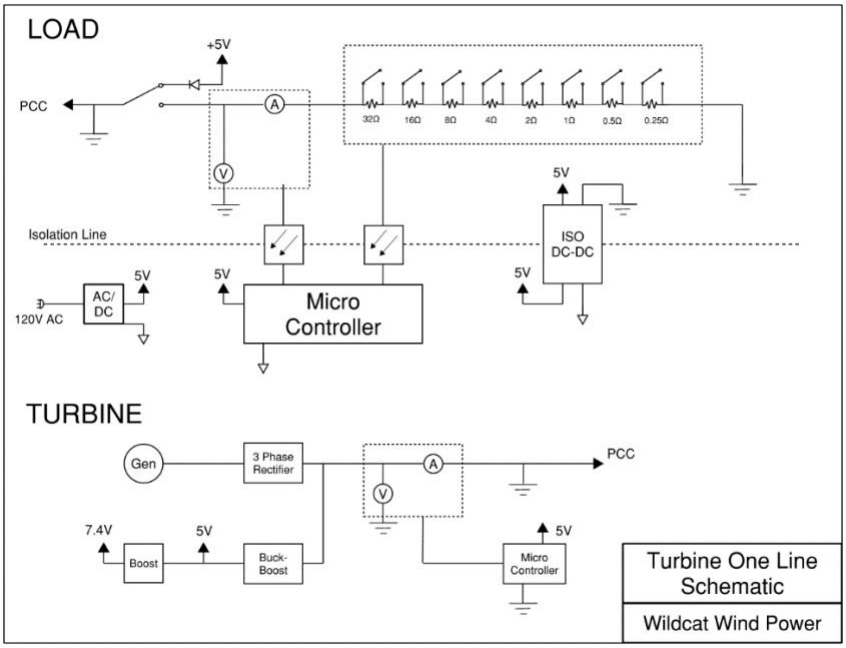
The load system includes its own set of communications circuitry and sensors. Communications between the load and turbine utilize the UART protocol. A dedicated isolating IC was used to only allow the incoming or outgoing data to pass without any power transfer. This IC is placed on the load-side system but could have been placed on the turbine-side system with the same outcome. Because the team did not opt to design or use differential line drivers, which is preferred for long board-to-board communications, the baud rate used in our system was set to a minimum to ensure no data corruption. The load system communicates with the INA260 power sensing IC via I2C protocol, rather than UART.

The load system will be controlled by a Teensy 4.1 microcontroller, which was chosen primarily to improve the team's data-logging and data-tracking capabilities during testing. The load has access to almost all of the turbine data through its communications systems, and the Teensy 4.1 is able to log testing data to a MicroSD card during testing for later analysis. The load also contains its own DC power supplies, such that it can and will be powered through a 120VAC wall outlet. The wall outlet will power the load controls and microcontroller, and supply power to the system for re-start after braking.

Power Systems

The generator, wiring, rectifier, filter, and load comprise the essential power system of the turbine during operation. The generator is rated at 24VDC for pulsed DC operation, and in use as a generator we have never seen it produce AC voltages beyond 18Vpp. At these lower voltages, the turbine should never produce above 48Vdc at the PCC. The filter is comprised of two LC (1uH, 3.3mF) filters in series on the DC side of the rectifier, to smooth the signal and limit high-frequency noise on the system. This filter combination provided the smoothest filtering and lowest losses during simulation of the rectifier system. Power wiring from the turbine nacelle to the electronics is a single 4-stranded 14 AWG cable, which should minimize wire losses even at the low voltages at which the generator operates. A simplified one-line diagram to illuminate the turbine power system is shown below.

Figure 12: Turbine One-Line Diagram



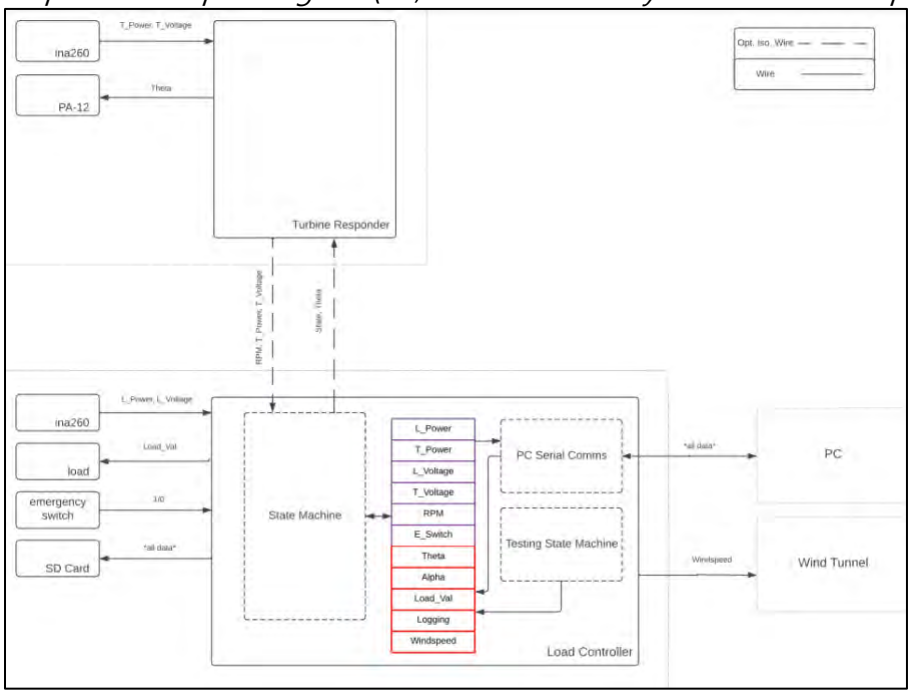
Software

The software development process is like previous years, but has been improved to simplify testing, data acquisition, and code collaboration. In the past, software would first be designed, written, and tested for individual electrical components to then be incorporated into the production code (for competition) to run on the microcontroller, and this year is similar. Turbine-load communication, variable load control, and peak tracking are all component level modules that have been adopted from last year's design. Off-the-shelf components such as the INA260 and PA12 came with libraries which made interfacing with these devices as easy as a method call. The Teensy 4.1 board has an on-board SD card interface and libraries are available for this as well. Once all component level modules and libraries were confirmed to work, a base level program was developed for the Teensy 4.1 on the load board to facilitate the development of an automated testing and data acquisition configuration, as well as the production code configuration. Alongside this, production code was developed for the Teensy 4.0 on the turbine board.

Turbine-load communication allowed a controller/responder setup where all inputs would be managed by the load controller and outputs forwarded to the turbine responder. As mentioned previously, the turbine responder's code could remain the same for both automated tests and production runs. However, on the load controller, there would be two configurations: one for automated testing and one for production code. For the sake of simplicity and clarity, these two configurations would be different codebases adapted from the same base level program.

The testing and data acquisition configuration for the load controller enables the system to interface with our wind tunnel and either automatically sweep through system parameters (pitch, resistance, wind speed) while logging data or take user input to specify system parameters. We designate theta as the pitch angle of the blades. The automatic tests are accomplished with a state machine which sweeps each parameter through a preset range. The step size, range to be swept through, and step time to account for transients can all be set prior to testing for each parameter. If the option is made to manually enter parameters, then the input is received via a serial connection to a PC. In this case, the user also has the option to log data or not. During data logging, all the system inputs and outputs are written to an SD card in .csv format for later export and analysis.

Figure 13: System Input and Output Diagram (PC, Wind Tunnel only used for Data Acquisition purposes)



When connection is made between the load controller and the turbine responder, the controller reads inputs [RPM, turbine voltage, turbine power, load voltage, load power, emergency switch] and produces outputs [state, theta, load resistance, PCC relay]. It will receive the first three inputs [RPM, turbine voltage, and turbine power] from the turbine responder via serial communication, read the remaining inputs via onboard sensors, determine the state of the system, generate outputs, then forward the first two outputs [state, theta] to the turbine responder. Similarly, the turbine responder will receive those two outputs, read the turbine voltage and power, update the pitch angle with theta, and then send its outputs back to the load.

All component level modules, libraries, configurations, and example code are stored in a GitHub repository. It is organized to have different examples for each component used, code from the previous year, all libraries used, and project folders for both the load controller and turbine responder. In each project folder, copies of the required libraries for that project are provided in the /src folder so it is easy for anyone to pull the code, open it in the Arduino IDE, and upload it without installing dependencies. Data archiving is performed by uploading test data to a shared drive for analysis and later use.

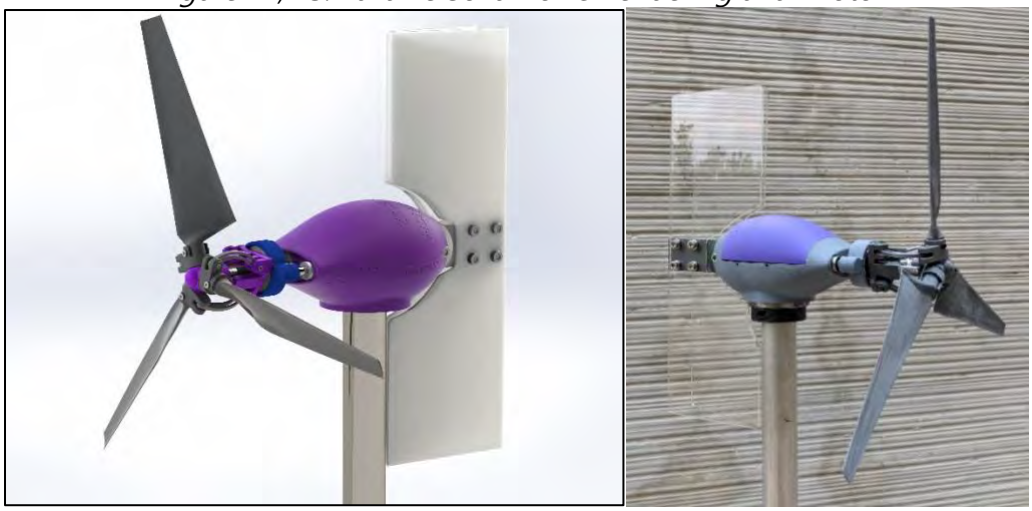
Assembly of Systems

Mechanical Assembly

The blades and nacelle were both 3D-printed, a manufacturing technique that we are comfortable using due to its ease of use in iterative design and manufacturing. Although we gained access to a Stereolithography (SLA) printer, we determined that both the blades and nacelle should be printed using our FDM printer, as it produces stronger, larger, and more reliable components. The main concern we faced with 3D-printing blades was the tendency of 3D-printed parts to fail along layer lines. To mitigate the risk of blade breakage, we printed the blades so that the layer lines were normal both to the shear force caused by the wind and the shear flow caused by the blade bending. We found it advantageous to conduct 3D design in SolidWorks through a shared file database on Microsoft Teams, so that different team members could work on separate turbine components at the same time. The nacelle yaw fin connector was 3D-printed, with the acrylic plating being laser cut. The active pitch system was constructed out of 3D-printed components and off-the-shelf bearings and fasteners.

To assemble the nacelle, hub, and blades, first each arm of the active pitch mechanism was attached to the rotating side of the bearing plate with pins and set screws, and then the other end of the arm was attached in a similar manner near the base of each blade. The blades were set in their respective rotating holders (allowed to rotate through a bearing), and the entire assembly was placed over the driveshaft and locked in rotation with it through a nut under the nosecone. The linear actuators were set on either side of the generator, and their arms attached to each arm on the nonrotating side of the main bearing plate. The cap of the nacelle, which contains printed brackets to hold each actuator in place, snugly fits on top of it after the actuators are mounted. The nacelle yaw fin is held in place behind the nacelle with a 3D-printed piece that is secured to the nacelle with fasteners. Care is taken during assembly to ensure that wiring for the actuators and generator will not be disturbed by the movement of the actuators or placement of the nacelle cap. The assembled turbine, along with its full SolidWorks model, is shown below.

Figure 14, 15: Turbine SolidWorks Rendering and Photo

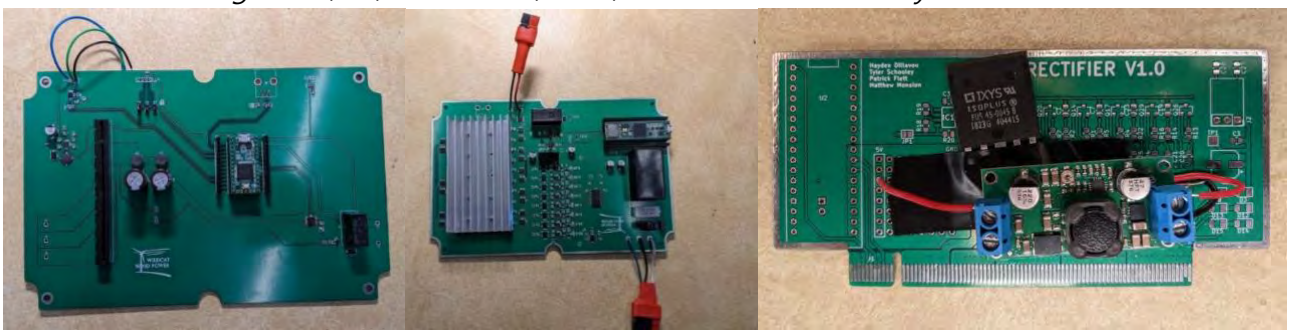


The assembly of the foundation began after testing of wooden mockups was complete. We chose to construct the design out of steel, due to it being the most easily available ferrous metal to us. For the thick metal base, we used 3/16" steel. To cut the steel, we used a plasma cutter to make a rough cut and then used an angle grinder to square out the sides. We then welded on a 1.5" outer diameter, 0.035" thick steel round tube to the center of the base. The sheet metal casing was made from 30-gauge steel. The steel that we bought was slightly too short to make it around the perimeter of the base, so we cut it into two pieces and aligned them opposite of each other around the base. This allowed for most of the perimeter to be twice as thick as the single sheet. We then drilled holes into the base and connected the casing to the base using steel rivets. We also used the same steel rivets to connect the two separate pieces of the casing on the top.

Electrical Assembly

All electrical circuits and systems outside of the turbine nacelle were designed on printed circuit boards (PCBs). We separated our systems into a load board, a turbine board, and a controllable rectifier board. All boards were ordered in early February along with their respective components and assembled mainly through a surface-mount assembly process in a PCB oven. As part of planning for the assembly process, a small test board was designed, ordered, and assembled in November using surface-mount components so that the team could learn how to assemble a surface-mount board. Surface-mount assembly presents the opportunity for the use of a wider variety of components for design, which is useful given the semiconductor shortages seen across the industry this year. The controllable rectifier system was designed to fit in a PCIExpress slot on the main turbine board, so that if the active system did not perform it could be replaced with a diode bridge rectifier. When it did not perform, it was replaced with a diode bridge rectifier soldered on to an unpopulated PCB. Images of the front sides of all three completed PCBs are shown below.

Figure 16, 17, 18: Turbine, Load, and Rectifier PCBs Fully Assembled



Wiring for the system was separated into two strands, power and communications. The turbine PCC was mounted in a NEMA-rated box contained with wire glands to connect the power strand and an inset RJ-45 connector for the communications wiring (Figure [D]). A 4-strand 14-gauge wire was used for the power transfer, and CAT5 cables were used for communications between the turbine and turbine electronics, as well as between the turbine and load. The load PCB was mounted in a similar container to the turbine-side board, and used similar connectors. The major non-SMT assembly of the board was the power resistors and necessary heat sink. The TO-247 package resistors were mounted with their thermal pads facing upwards, and an off-the-shelf aluminum heatsink was cut and then mounted over the resistors. Machine screws coming up through the bottom of the PCB secure the heatsink to the resistors through holes tapped at regular intervals in the heatsink.

Commissioning Checklist

The system was designed to be mobile with ease of commissioning in mind. While mobility is important, durability and reliability were not to be sacrificed. The integration of subsystems both electrically and mechanically was achieved with connections that are easy to install and uninstall. The commissioning checklist provides detailed instructions for the implementation of our system, and is shown below.

Step #	Process
1.	Run power and control wires through the hole in the foundation tubing, pull through to keep connectors away from water.
2.	Fill foundation with sand using shovels, move sand from center of container.
3.	Place foundation, wobble to move deeper down to competition depth, fill around foundation with sand, level the entire mechanism.
4.	Install stub and run wires upwards through it.
5.	Bolt the base plate of the tower to competition-provided stub, pulling wires up through tower.
6.	Attach three-phase power and control cables to nacelle base connectors.
7.	Attach nacelle to tower through compression fit on yaw bearing. Ensure compression fit of nacelle lid is secure. Check alignment of nacelle on tower, and use provided level to ensure tower is vertically aligned. Adjust foundation if necessary.
8.	Connect 3-phase power wires from the turbine to the APPs connected to the turbine control box labeled “3-Phase Power".
9.	Plug in the RJ-45 connector from the turbine to the female connector labeled “Turbine Controls”.

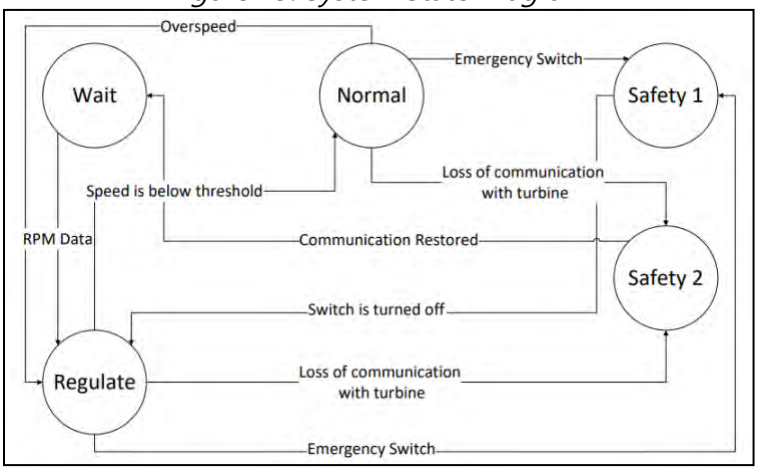
10.	Connect rectified power cord labeled "Turbine DC Power" from the turbine control box to the PCC via APPs. Connect the other side of the PCC to the load control box labeled "Turbine DC Power" using APPs as well.
11.	Using an ethernet cord, connect the load to turbine controls. The female RJ45 connectors are labeled "Load Controls" on both sides.
12.	Connect the emergency stop cable labeled "E-Stop" on the load control box to the competition provided stop button via a JST RCY connector.
13.	Plug the load control board into 120V wall power using the IEC connection cord.
14.	Ensure control box lids are secure and boards are powered.

Operation

Operational Modes and States

The turbine is set to operate between five different states, the relationship between the states is shown in the following figure. Because the competition does not require full power and RPM regulation at wind speeds above 11 m/s, the control system does not attempt to estimate the tunnel wind speed during operation, and the decision to regulate power and RPM is triggered by an RPM threshold. The speed threshold after which the turbine begins to regulate is 3500RPM, chosen because it reflects roughly a 12 m/s wind speed cutoff, ensuring regulation does not occur during the power curve task. The turbine has operated at over 4900RPM safely, so the 3500RPM cutoff is well within the margin of safe operation. Our experience working with 3D-printed blades tells us that they would fail due to internal stresses at roughly 5500RPM, so the safety margin should provide enough leeway.

Figure 19: System State Diagram



The operational functions of the turbine, and how each state is related to others, is explained in the following table. We define separate turbine states for each of the competition safety tests, as they are triggered by different system conditions. Regulation during the 'Regulate' state is achieved through a simple proportional controller that adjusts the blade pitch to keep the turbine between 2850 and 3450RPM. After the transition from the 'Wait' to 'Normal' state, the load resistance is adjusted to maintain the turbine-side voltage at 3.3V, high enough to continue to power the turbine controls and linear actuators, and this functionality is maintained until the turbine produces enough power to run the linear actuators regardless of load resistance. Only then does the load set a constant resistance at a

point to maximize power production. Discontinuity detection to enter the 'Safety 2' state is defined as the combination of nonzero RPM and a load voltage much greater than the turbine voltage.

State	Function	Enter Case	Exit Case	Operation
Wait	Wait until Cut-In	Default	Turbine-Side electronics on	Blades pitched to cut-in pitch, Load at max resistance
Normal	Power Curve	Turbine-Side electronics on OR Wind Speed < 12	Wind Speed > 12	Load Resistance Constant, Blades pitched to maximum power pitch
Safety 1	Emergency Stop	E-Stop Closed	E-Stop Open	Power turbine from load, feather blades, wait for E-Stop
Safety 2	Disconnect Brake	Discontinuity Detected	Continuity Detected	Power turbine from load, feather blades, wait for continuity
Regulate	Keep turbine at safe RPM	RPM > 3500	RPM < 2850	Regulate RPM through pitch control

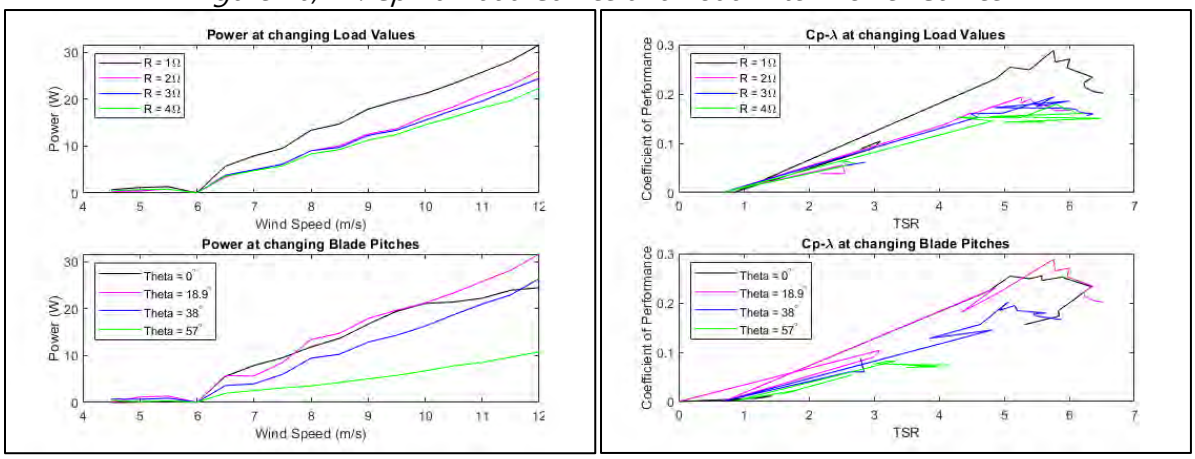
Testing of Systems

Subsystem testing was carried out throughout the design process, culminating in significant subsystem testing in February and March for the foundation system, blades, electrical systems, and active pitch mechanism. Once we began to connect subsystems and assemble the entire-turbine, we could begin full-system testing.

Preliminary full-system tests showed that the original active pitch mechanism could theoretically brake the turbine and regulate its power production and RPM, but higher-speed testing showed that the mechanism was unwieldy and oscillated heavily at higher wind speeds. The oscillations caused some joints to disconnect and the blades to shatter at one point, so we decided to go back and redesign the pitch mechanism to reduce the moment of inertia and dampen the oscillations. We also strengthened the connection between the blades and the hub, as mentioned previously. After reducing the diameter of the nosecone and the radius of the pitching arms, further testing showed reduced oscillations and operation without mechanical failure. After fixing these issues with the active pitch system, we turned our attention to our controllable rectifier design. Testing of the rectifier with the turbine and other electrical systems showed that, although it could operate cleanly if powered externally, its control power setup interfered heavily with its operation when it had to power itself. We eventually determined that fixing the control power issues would require new PCBs of the rectifier and turbine, and made to decision to switch back to a passive rectifier.

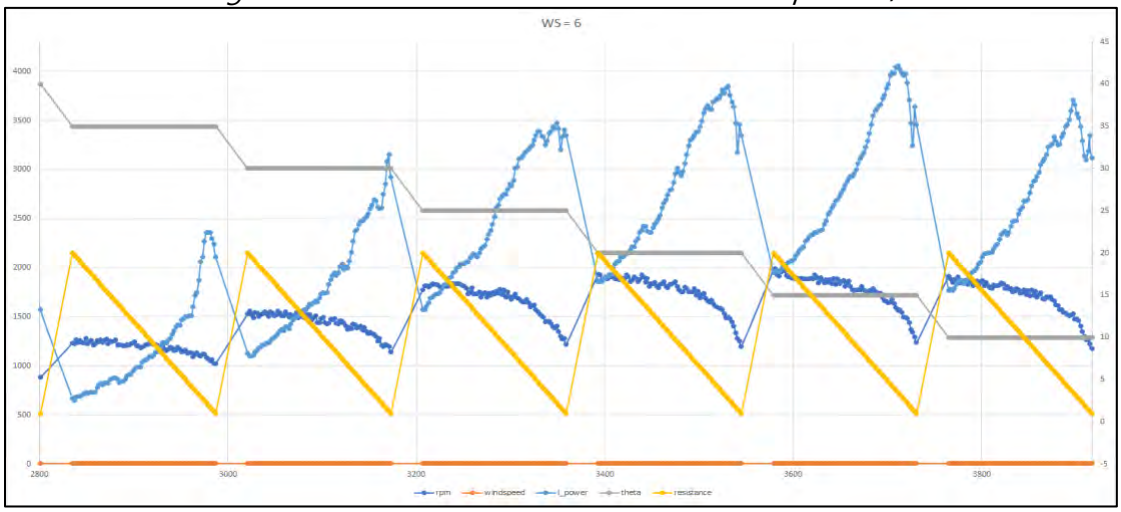
Once these decisions had been made, it was time to move into full-system testing to determine the operational parameters of the turbine control scheme. We developed a fully automated testing platform that integrated with our wind tunnel, such that we were able to automate a test that ran through multiple variables and track data, or manually change system parameters to see how the turbine reacted. We used the automated system to first determine a maximum operational RPM for the turbine, by operating it without RPM regulation to the highest wind speed our tunnel could produce. We determined it to be upwards of 4900RPM, well above our planned regulation cutoff of 3500RPM. Next, we set out to collect data on the pitch and load relationships, in order to determine the best pitch and load setpoints for the Power Curve task. The results from this round of data collection are shown as Cp-Lambda and Power Curve plots below, although they do not represent the final turbine data that we collected. This data served to validate our control scheme.

Figure 20, 21: Cp-Lambda Curves and Load-Pitch Power Curves



Next, we began to run through the competition safety tasks and develop RPM regulation for the higher wind speeds in the durability tasks. Tests of each saw strong performance in our regulation scheme, but we saw some minor issues during the safety tasks with our turbine-side microcontroller not always turning back on. We were able to fix the issue with a combination of software and hardware changes, and tested turbine braking up to 16 m/s in both the disconnect and E-stop conditions to ensure it would reliably brake. With the fundamental secure operations of the turbine (braking, regulating), we continued analyzing the relationship between load resistance, pitch, power, and RPM. Data collection showed that pitch angle and load resistance played a major role in power production, as seen in Figure 22 below, where a variety of load and pitch angles and their respective power outputs are shown at a single wind speed.

Figure 22: Load-Pitch-Power-RPM Relationship at 6m/s



Testing of the foundation system was conducted separately from turbine testing, due to the fixed floor of our wind tunnel, and focused on maximizing the torque that the anchor could withstand while set in wet sand. A model of the competition setup was created in a plastic tub filled with 200lbs of sand and filled with water, and a spring scale was attached to a wooden dowel to allow for measurement of torque. Designs, both preliminary wooden ones and the final steel mechanism, were installed in the

sand, and torque was applied to the base until it failed. Under this setup, we found our final box design could withstand 15-20 N*m before it began to shift, and more than 30 N*m before it failed. The testing setup in use is shown below. We also developed a platform in the wind tunnel to test the torque that the turbine will apply to the foundation. An arrangement of PVC pipes with a tiltable pivot point on the bottom was attached to the turbine base, and a pulley system along with spring scale was rigged in our wind tunnel to measure the tilt of the turbine against the scale under wind. The setup is also shown below, and showed loading near 1 N*m at 14 m/s, that decreased as the blades pitched out of the wind.

Figure 23, 24: Foundation System Testing

Review from 2021 Competition

Some design concepts, mainly the variable resistance load, were carried over from the turbine designed for the 2021 competition, but most of the work this year was spent on designing new turbine elements and controls. Even the variable resistance load itself was redesigned with power electronics to be more efficient and more accurate. The active pitch system was a completely new actuator that we had not formally attempted to design ever before, and the turbine blade design was fully revamped. A new style of generator was used, and obviously the foundation system involved many new lines of thinking. Although the system controls and operational states are similar to the system designed for the 2021 competition, the overall control requirements lessened, and degree of state estimation that was needed for the previous turbine was not implemented this year. One system, the turbine nacelle fin and yaw control system, was somewhat recycled from years past, but a new tailfin was still designed in the same style. The turbine tower was fully re-used from previous turbines.

References

[1] T. Burton, D. Sharpe, N. Jenkins, and E. Bossanyi, *Handbook of Wind Energy*. Chichester: John Wiley & Sons, Ltd, 2001.

[2] R. S. Shevell, *Fundamentals of Flight*. Englewood Cliffs, NJ: Prentice Hall, 1982.

[3] S. Hermann. Threaded Inserts for 3D Prints – Cheap vs Expensive. Dec. 7th, 2019. Accessed on: April 21st, 2022. [Online]. Available: https://www.cnckitchen.com/blog/threaded-inserts-for-3d-prints-cheap-vs-expensive