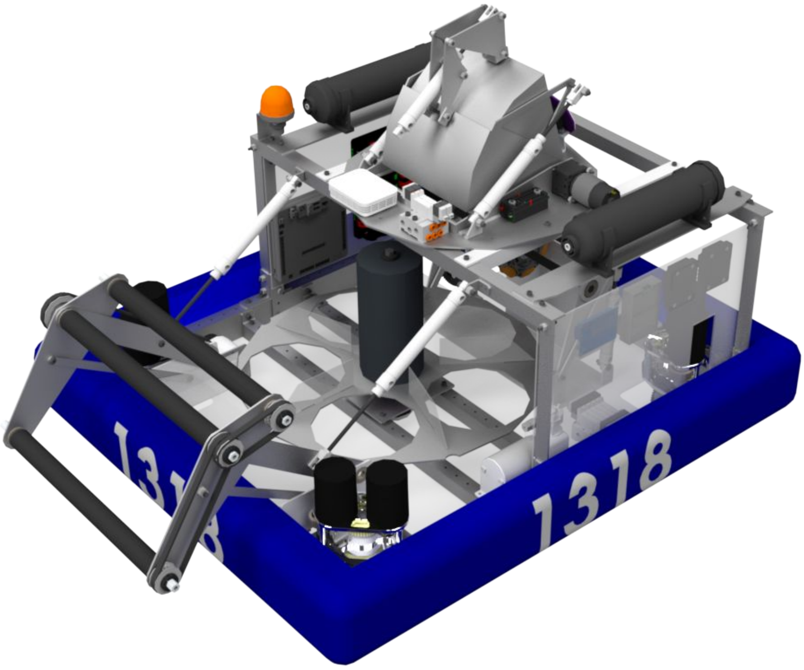
Issaquah Robotics Society

2021 Engineering Notebook



Our Robot

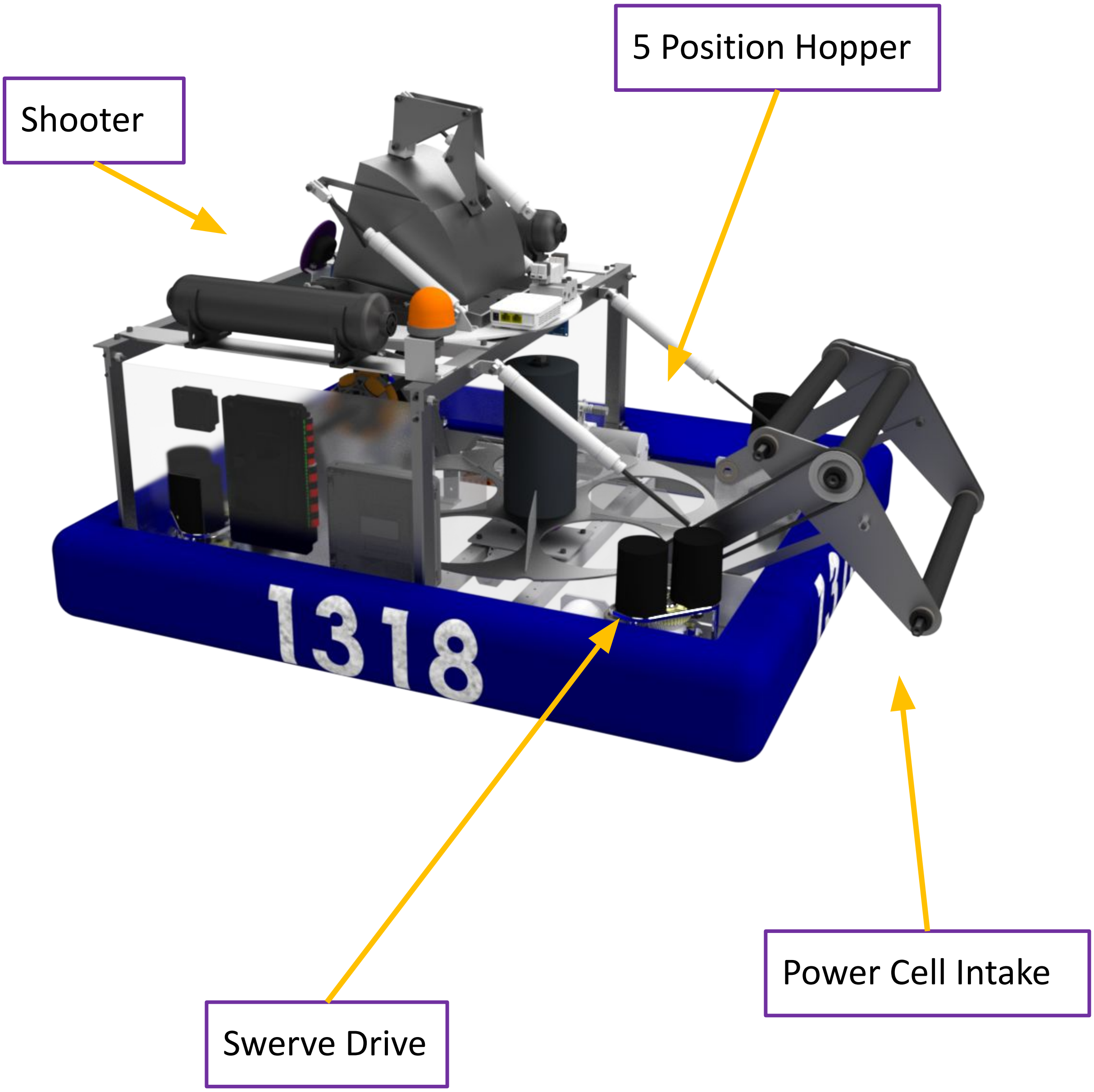

5 Position Hopper
Shooter
1318
Swerve Drive
Power Cell Intake

Table of Contents

Engineering Story

One of the goals of *FIRST* is to encourage students to apply the engineering process to various problems. IRS Team 1318 believes that engineering is about developing and implementing a design both effectively and efficiently. We incorporate the engineering process into the design process for the entire robot, integrating engineering into entire meetings instead of just using this process for specific tasks.

While this year was vastly different due to the COVID-19 pandemic, we applied the same adaptability and change needed in order to be a good engineer to organize and run our team and accommodate the new game. Despite the lack of in-person collaboration, our team was able to adapt and build an entirely new drive train, while working remotely. Then, using last season's robot as a starting point, our team improved and expanded upon the design in order to optimize it for this year's game.

Our Product Cycle

To ensure efficient and effective engineering, we follow a product cycle. This allows the IRS to continuously improve our robot while following a reliable process.

Brainstorming

1. Establish a strategy for the game.
2. Determine the most important tasks.
3. Conceive mechanisms that fulfill the tasks.

Design

1. Build CAD models.
2. Arrive at consensus on preferred designs.
3. Fine-tune and iterate mechanism designs.

Build

1. Assemble a robot based off of prototype designs.
2. Fabricate & assemble mechanisms.
3. Build competition robot.

Robot Evaluation

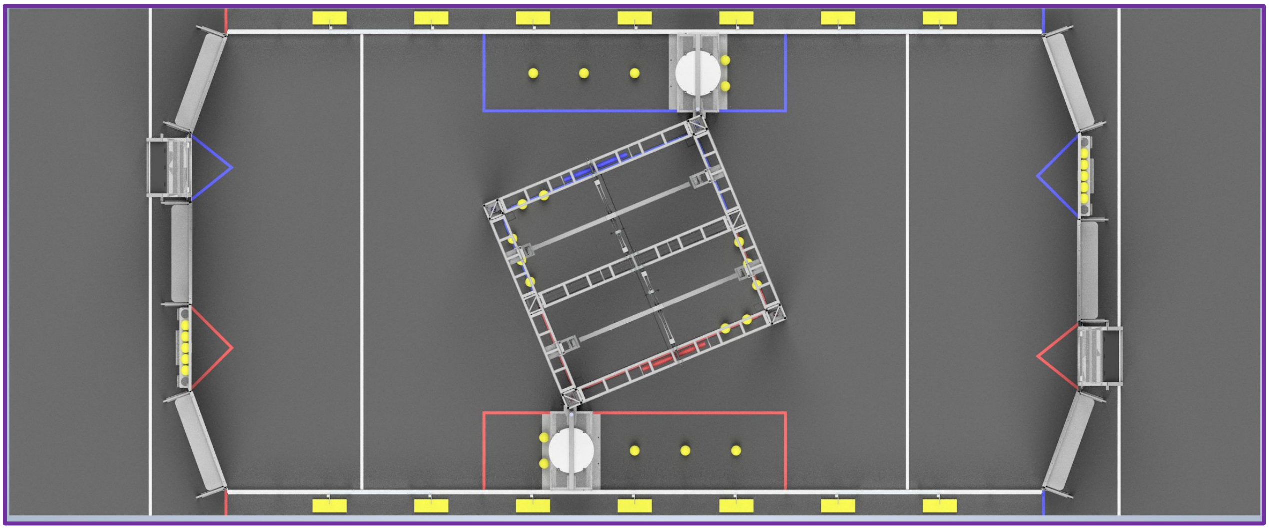
1. Test each mechanism separately.
2. Run robot through integrated tasks.
3. Practice with competition robot at field.

Post-Competition Analysis

1. Analyze recruitment, training, and build season.
2. Evaluate student leadership models.
3. Review our design and engineering processes.

The Game

Summary



In INFINITE RECHARGE AT HOME, teams demonstrate what their ROBOTS and drivers can do in a Skills Competition inspired by the INFINITE RECHARGE game. Teams virtually compete against each other and are scored based on the three challenges in which they perform best.

The Galactic Search challenge emulates the Autonomous Period of INFINITE RECHARGE. Teams locate and collect POWER CELLS as fast as possible on one of two pairs of paths.

In the AutoNav Challenge, teams program their ROBOTS to autonomously drive predetermined routes through three different paths as fast as possible.

In the Hyperdrive Challenge teams drive their ROBOTS remotely, without preprogrammed navigation, through four different paths as fast as possible.

The Interstellar Accuracy Challenge emulates the shooting challenges of INFINITE RECHARGE gameplay. Teams score POWER CELLS into a representation of the BOTTOM PORT, OUTER PORT, AND INNER PORT from four zones. Teams attempt to score as many points as possible in five minutes.

The POWER PORT Challenge emulates the teleoperated portion of INFINITE RECHARGE. Teams collect POWER CELLS and score them into a representation of the POWER PORT. Teams attempt to score as many points as possible in the POWER PORT in one minute.

The summary above was extracted from the 2021 FRC Game Manual

Strategy Development

Game Analysis - Tasks

By analyzing the required tasks for each challenge, we optimize our robot design and make a choice as to which three challenges we would be able to perform successfully.

Tasks Performed in Challenges

- Galactic Search Challenge
    - Intake Power Cells
    - Quickly drive a specific pre-programmed path
- AutoNav Challenge
    - Autonomously drive three predetermined paths
- Hyperdrive Challenge
    - Teleoperate robot through four potential paths
- Interstellar Accuracy Challenge
    - Shoot Power Cells into Power Ports
- Power Port Challenge
    - Intake Power Cells
    - Shoot into Power Ports

Strategy Development

Game Analysis - Design

Critical Robot Functions

- Autonomous
    - Detect Power Cells
    - Select appropriate path
- Intake Power Cells
    - Intake Power Cells from the ground while driving at appropriate speed
    - Intake Power Cells quickly and efficiently - “touch it, have it”
- Drivetrain
    - High level of maneuverability
    - Speed

Strategy Development

Approach To Design

Team Prototyping Decisions

Shooter CAD Model

Intake Prototype

Drivetrain Fabrication

Full Robot Design

We began by building our swerve drive train. We developed most of our robot virtually in collaborative CAD sessions. We then individually assembled each robot part in team members' garages.

Task Identification

CAD modelling

Individual Assembly

Parts integration

Finalized robot

A summary of our designing process

Hardware

Drive Train

Needs	Wants
Navigate different paths on field	Optimize maneuverability
	Minimize weight
	Balance maneuverability with stability

Possible Ideas

6 Wheel Kitbot	Swerve Drive
Fast	Extremely fast
Simple	Mechanically complex
Lower pushing force potential	High pushing force potential
Fewer motors required	Extra motors required
Less maneuverable	Extremely maneuverable

Notables

- The drive train is powered by eight Neo motors
- The drivetrain has PID control for position and velocity control, as well as a PID-based break mode

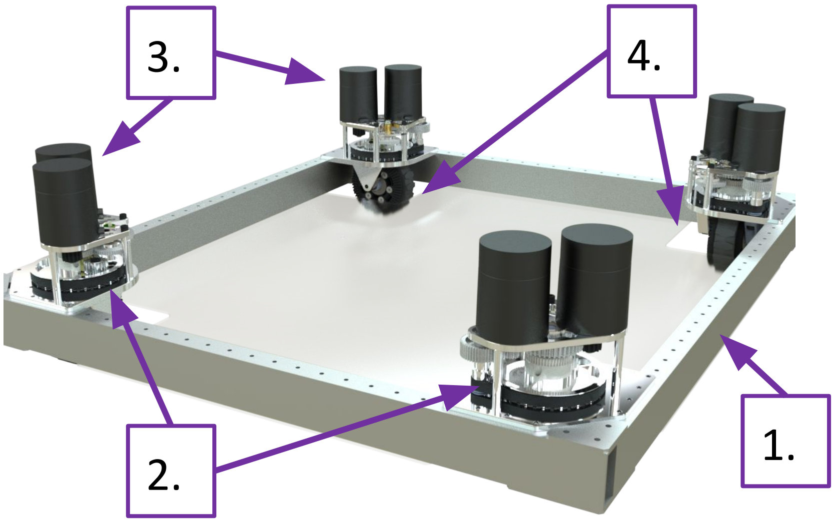
Hardware

Drive Train

Swerve Drive Train Design

The drive train this year is a 4-wheel swerve drive that utilizes four 4” Versa wheels. It also has 8 Falcon 500 motors coupled with a 8.95 : 1 gear ratio (about 11.1 feet/second max) within a SDS Swerve Drive Module.

The Final Product



Our modified VersaFrame Chassis

Parts:

1. VersaFrame 2x1.
2. 4 8.95 : 1 SDS Swerve Modules.
3. 8x Falcon 500 motors
4. 4x 4 inch Versa wheels

Power Cell System

Intake

Needs	Wants
Smoothly grab and transfer Power Cells to Hopper	Intaking with little driver accuracy
No possibility of jamming or easy way to remove jams	Intake easily while traveling at max speed

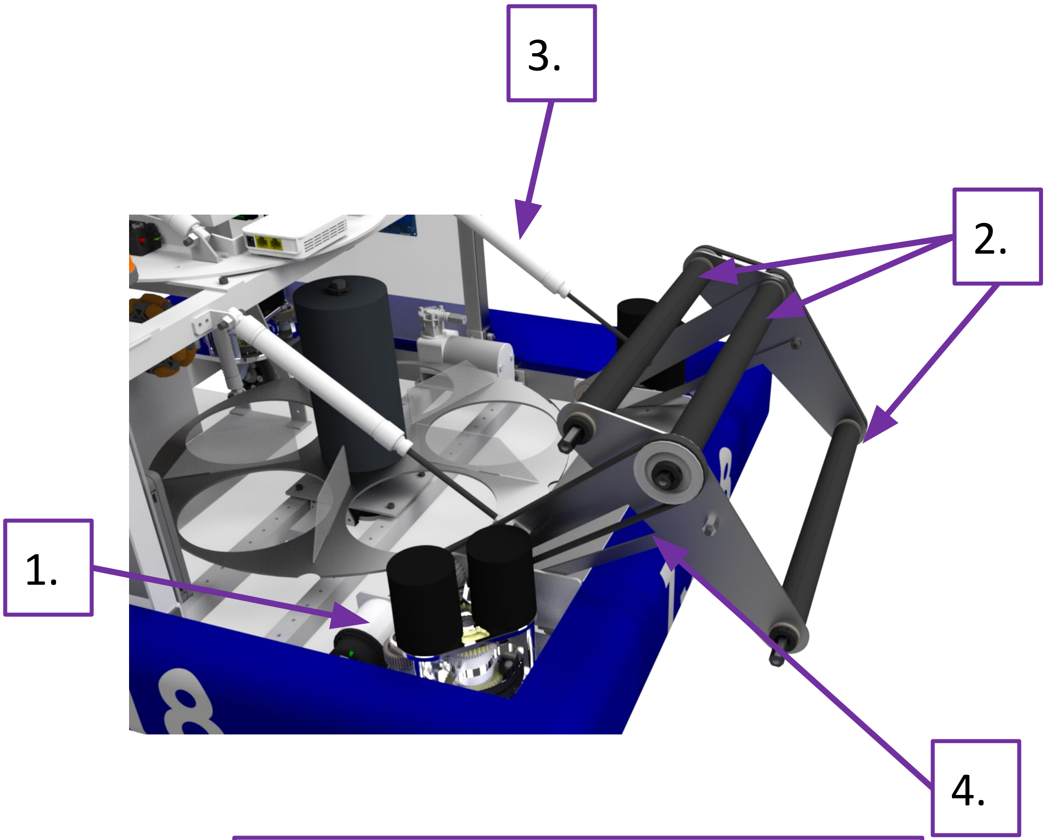
A CAD model of our intake linkage

Notables

- PVC rollers for greater surface area

Power Cell System

Intake



A side view of our intake assembly

Features:

- Over-the-bumper Power Cell intake
- Retractable
- 15-inch width to intake multiple Power Cells in one instance

Parts:

1. 1- Falcon 500 motor
2. 3- Versa Rollers
3. 2- 5” stroke pistons
4. 4 bar linkage

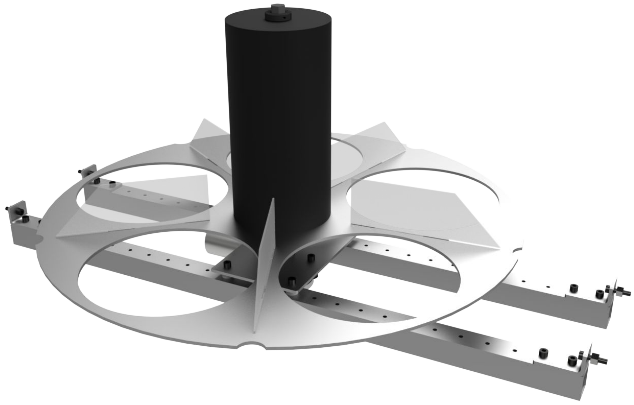
Power Cell System

Hopper

Needs	Wants
Index/serialize Power Cells	Have the ability to outtake 5 Power Cells in 1 second
Pass Power Cells from the intake to the shooter	Manufacture the hopper from aluminum

Possible Ideas

Serializer	5-slot Rotary Indexer
● Simple ● May exhibit a problem of jamming Power Cells	● Slightly more complex than a shooter ● Indexes Power Cells efficiently without the issue of jamming



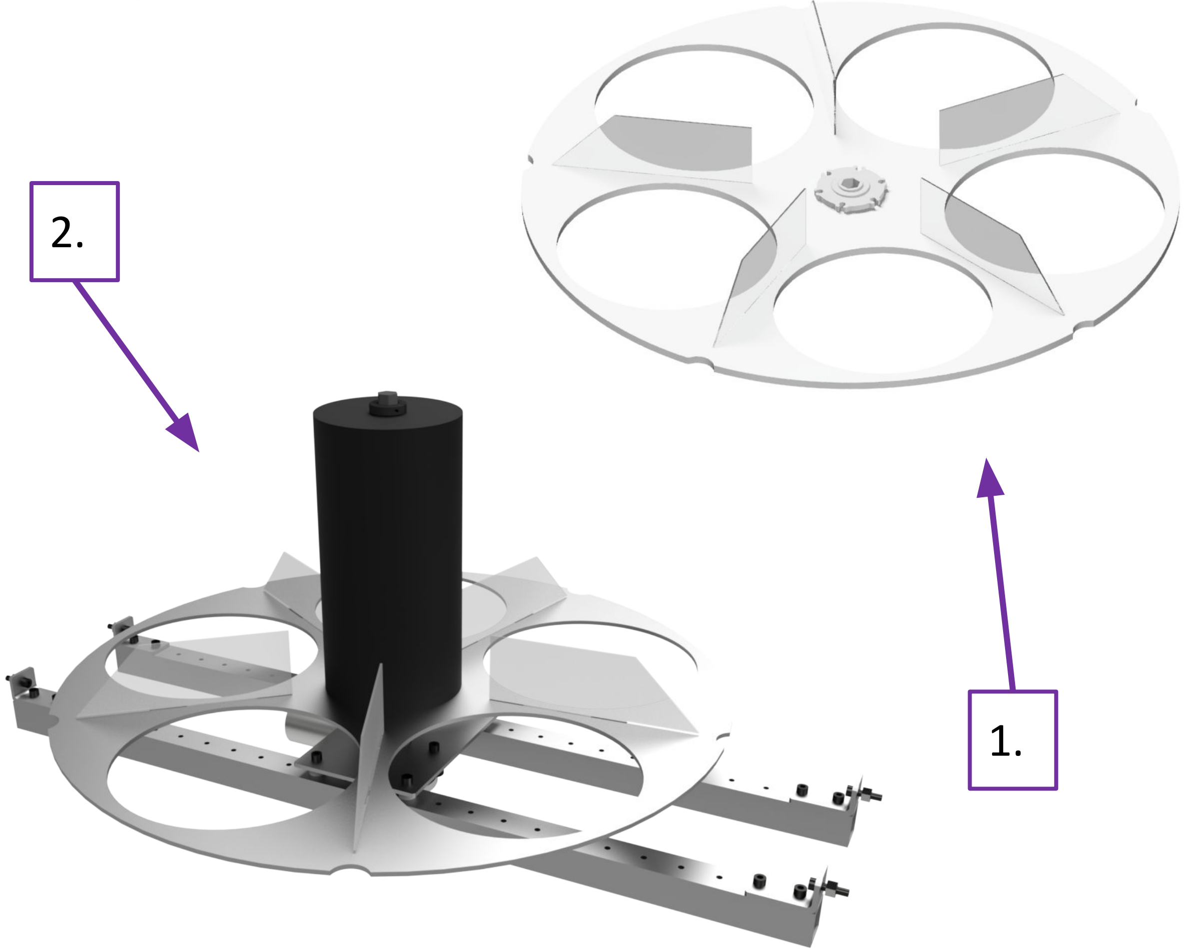
CAD model for our 5-slot indexer

Notables

- Snowblower Motor
- Use of 3D printed support

Power Cell System

Hopper



The hopper assembly

Features:

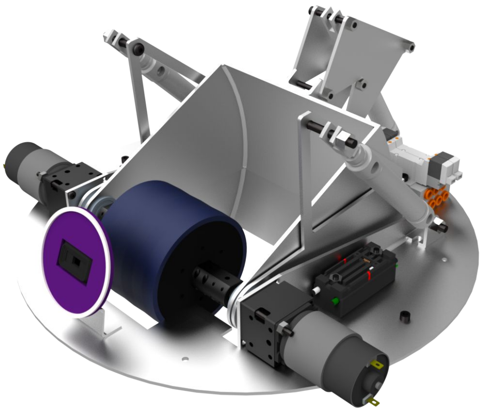
- A 5-slot Power Cell indexer
- An AM2235 motor with a VersaPlanetary 180° drive gearbox

Parts:

1. Carousel disk
2. Drive motor and gearbox

Power Cell System

Shooter



Needs	Wants
Ability to shoot the ball into the outer port	Ability to shoot the ball into the inner port
Single shooting position	Multiple Positions to shoot from

Possible Ideas

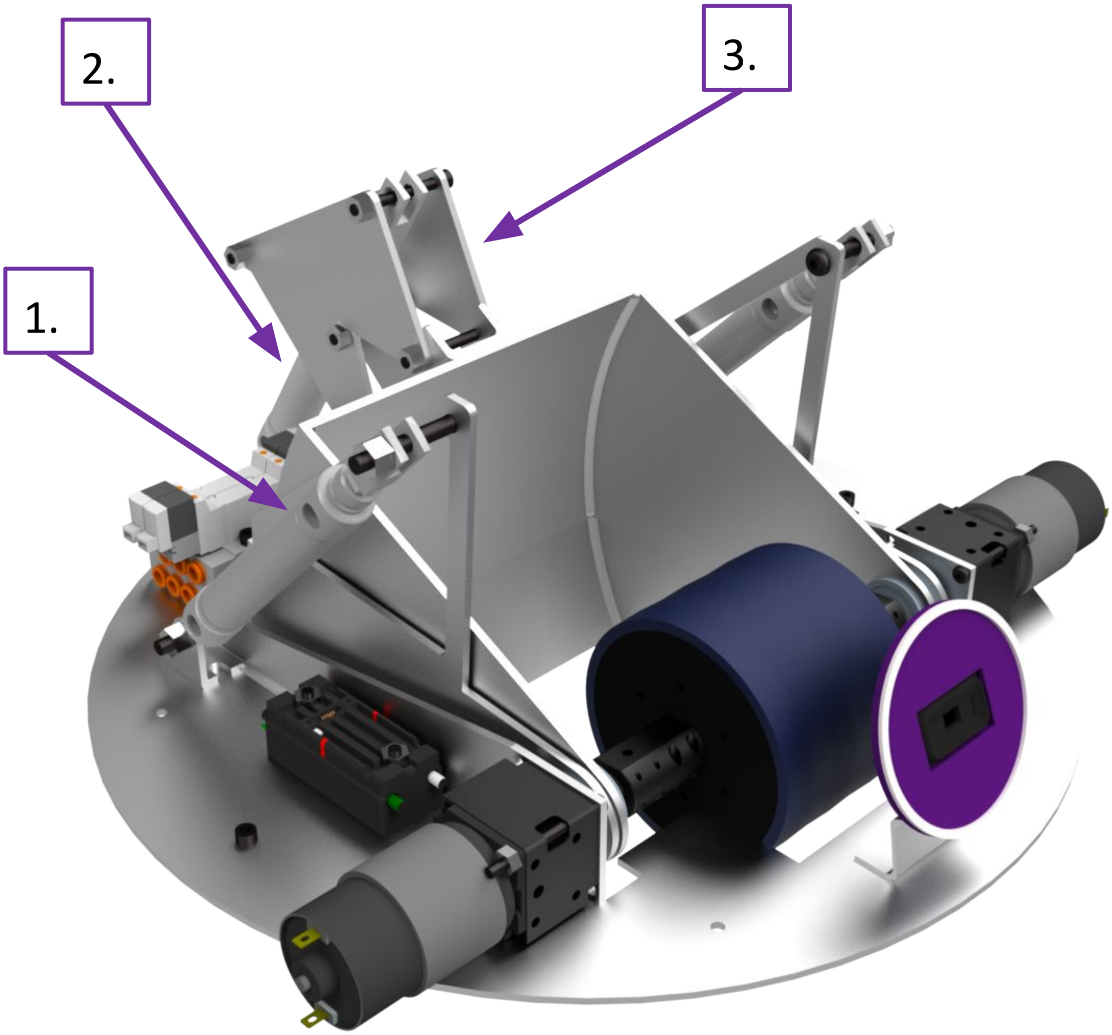
2 Piece Hood	3 Piece Hood
Pros	**Pros**
• Simple • Multiple positions to shoot from	• Can shoot from almost anywhere on the field • Smaller and collapsible

Notables

- Continuous, real-time tracking of retro-reflective targets
- Retractable hood for a low profile

Hardware

Shooter



Features:

- A variable hood that allows for multiple shooting angles.
- Two 775 pro motors for fast shooting.

Parts:

1. 2- 2.5” stroke pistons
2. 1- 2” stroke piston
3. 4-bar linkage

Electronics & Pneumatics

A section of the left-side electrical board.

Electronics Notables:

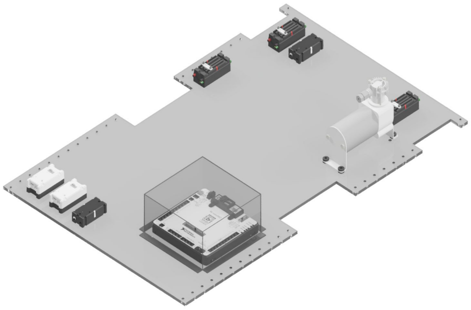
- 2 main electrical boards
    - Generally separated between pneumatics and main FRC control system
- Some electrical components mounted on relevant mechanisms to decrease complexity
- Large focus on pneumatic mechanisms

Electronics & Pneumatics

Left Board	**Right Board**
Our left board contains electronics such as 2 PCM's, and a Raspberry Pi and a relay module.	The right board contains the Power Distribution Panel, the RoboRIO, and the VRM.

Bellypan board

The Bellypan board is a polycarbonate board containing a motor controller, the motor for the Intake, and the circuit breaker.



Pneumatic Notables

- 2 PCM's
- 2 Manifolds
- 2 double solenoids mounted on the shooter
- Two air tanks
    - One on each side of shooter
- The ball manipulation system contains 6 pistons

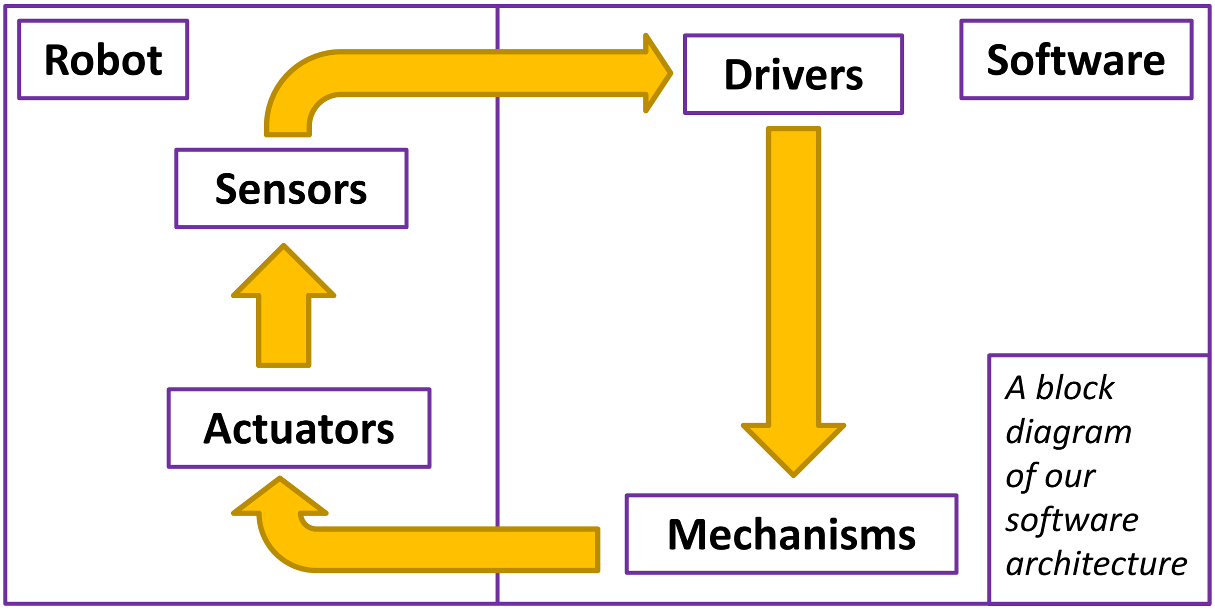
Software Architecture

Notables

- ✓ Macros
- ✓ PID feedback control
- ✓ Autonomous
- ✓ State machines
- ✓ Vision System

Dual input controllers with motion control

Our team uses Java for robot programming and utilizes Github as our source code repository.



A block diagram of our software architecture

Vision

Vision allows us to detect the retro reflective tape on the outer port of the scoring walls to align ourselves to be able to shoot Power Cells accurately. 2020 is the first year we have vision on a Raspberry Pi instead of on the RoboRIO, which will increase the framerate of the vision processed video.

Software Architecture

Vision (cont.)

We also used OpenCV on a Raspberry Pi to detect Power Cells and differentiate between the blue and red paths in the Galactic Search Challenge, allowing us to have a more competitive score.

Proportional Integral Derivative (PID)

- Velocity PID control allows us to regulate the velocity of the drivetrain to deal with inaccurate movement.
- Positional PID control gives us the ability to move our drivetrain to a specific position and maintain that position.

Smart Dash

Smart Dash receives information from the robot and displays selected data. This allows us to identify and/or solve problems more quickly.

Software Brakes

Virtual brakes use PID to keep the robot still.

Scouting Network

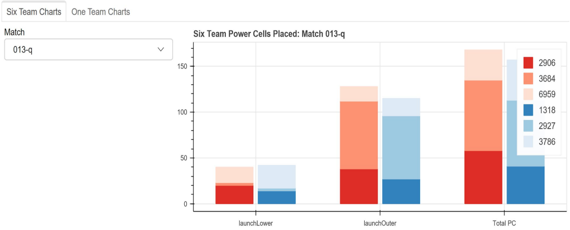
About

Our scouting network allows us to collect large amounts of data on the abilities of other robots. We then use this data in making better game strategies as well as for choosing teams during alliance selection.

Programming

The server app of our scouting program was coded in Java JavaScript, and Typescript. The data is stored in a database using Python in tandem with SQL. The graphs are then created using a Python graphing package called Bokeh and displayed on a dynamic server.

Data Analysis



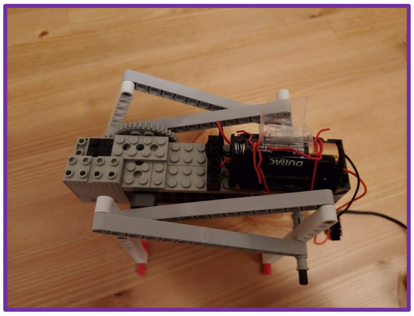
Match
013-q

The Bokeh server for generating dynamic graphs

Match: 004-q Team: 3586 Red 1: Teleop

2020 UI Design for the Teleop page of the Scouting System

Extras



A basic quadruped robot from a training kit distributed to new members

Fall Training

Before kickoff day students and mentors prepared for the virtual season ahead. They designed comprehensive training courses for new members, teaching CAD, as well as programming. They also developed build kits that taught fundamental mechanical skills.

CAD

We regularly use CAD to design parts and assemblies to make a completed robot model before its fabrication. Due to this year's unique circumstances, however, the majority of robot design was done in CAD.

Thank you to all of our sponsors!

BOEING

Microsoft

SUPERINTENDENT OF PUBLIC INSTRUCTION WASHINGTON

SPEEA IFPTE LOCAL 2001

ISSAQUAH SCHOOLS FOUNDATION

ISSAQUAH HIGH SCHOOL BOOSTER CLUB

FIRST WASHINGTON

RYVER

NEWPORT MANUFACTURING
CUSTOM LASER CUTTING
WELDING & FABRICATION

Audacitee
SCREEN PRINTING + EMBROIDERY

Whispering Pines
THE NORTHWEST'S RIVERSIDE
RV PARK in CLE ELUM, WA

ISSAQUAH HIGH SCHOOL

ISSAQUAH SCHOOL DISTRICT 411