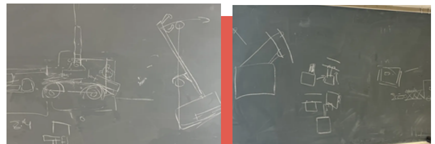
DESIGN PROCESS

BRAINSTORMING

We make **consistent** efforts to maintain a group-based effort in developing our robot's design. This is done through a mix of online calls, messages, and classroom team meetings.



Day one brainstorming robot ideas

THE IMPORTANCE OF CAD

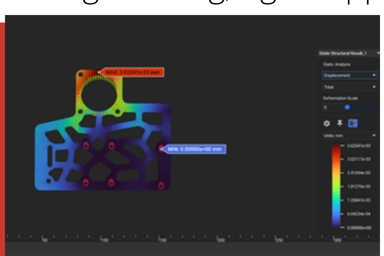
CAD is an important tool for planning and documenting our **engineering design process**. We use **Onshape** as our collaborative CAD platform, and through it, we can design and modify **prototypes** for our robot's drivetrain and mechanisms. After prototyping, we finalize our designs before beginning the manufacturing process.

MACHINING

3D printing allows us to make **complex and versatile** parts while being able to **rapidly iterate** on our designs and produce parts in a matter of hours. We use **Prusa MK4 printers and Bambu X1C printers** to make our CAD parts come to life.

Laser Cutting is essential for prototyping custom flat sections of the robot and allow for **rapid testing and design**.

The CNC Mill allows us to machine custom parts and plates. This is crucial as it grants us greater **flexibility** in our designs while providing a strong, rigid support structure for our robot.



Machining: In the pivot plates we used FEA programs to analyze and optimize the structural strength of the parts.

Our school robotics lab

STRATEGY

Observing Restraints

- Each cycle is very short meaning each **minor inefficiencies build up** quickly
- We needed the sample to enter and leave the robot quickly, as turning will complicate driving and slow us down, meaning it should be able to intake from both sides and score through both sides
- Minimize bot size to make the climb challenge as manageable as possible

Simplicity

We follow **KISS (Keep It Straightforward and Simple)**

- We aimed to keep mechanisms simple so we wouldn't run out of time for software (a problem we've struggled with in the past.)
- We combined our scoring and intaking mechanism, eliminating possible **points of failure**.
- We aimed to primarily score through **baskets**, as it eliminates the possibility of human player error, and is very straightforward

Improvements

However, our robot isn't perfect..

- The driver has to be very skilled to be able to intake from the submersible quickly and consistently. We plan to combat this by putting a camera on the intake, which would allow the claw will be able to automatically align and adjust to the sample.

ROBOT OVERVIEW

TELESCOPE: EXTENDS AND TRANPORTS INTAKE

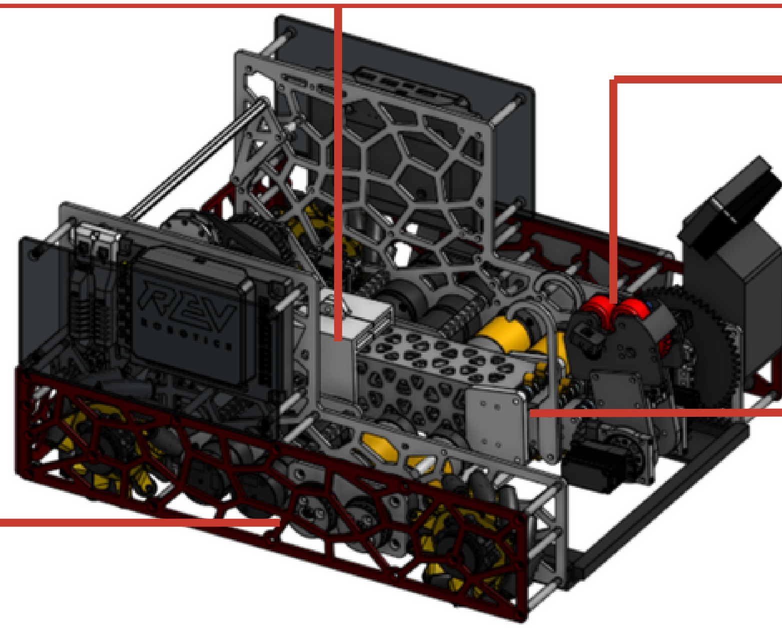
- We used box tubing over traditional linear slides due to its size, weight, and increased support. Compared to the 12 linear slides needed to reach the high basket, the box tubing is only a **fraction of the weight** and size, and functions at a comparable speed.
- Our coaxial telescope is made of 4 stages of pocketed box tubes, commonly found in FRC, which **pivots and extends** from the same shaft. It is cascade strung for the extension and continuously strung for the retraction.

INTAKE: COLLECTS AND SCORES GAME ELEMENTS

Our wheel intake is extended into the submersible and angled down to grab samples. Once it contacts a sample, it immediately sucks the sample in and the telescope is retracted. This works in conjunction with the sweeper to make sure we can **intake from nearly everywhere** in the submersible with **speed and precision**. The driver can easily reorient the samples by pushing them slightly, so we can intake the sample no matter the orientation.

DRIVETRAIN: SUPPORTS AND MOVES THE ROBOT

We designed our drive train around a fully custom, lightweight, pocketed aluminum chassis. The low weight and slim profile allow us to achieve our design objectives of high **speed** and **controllability**, while leaving space to **accommodate** our other mechanisms.



CLIMB: HANGS ON BAR

Aluminum hook attachments allow for a level 2 climb using the telescope and no other actuators in just 4 seconds. The hook latches on to the first bar and pivots, allowing the robot to go over the barrier.

ROBOT TIMELINE

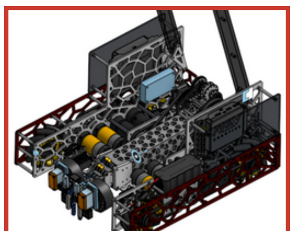
QUAL 1 (12/7):

High- 164 pts

OPR - 56 pts

Robot:

- No Autonomous
- Claw Intake + Differential Wrist
- No Climb



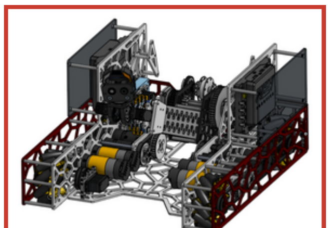
QUAL 2 (1/18):

High- 218 pts

OPR - 156 pts

Robot:

- 3-4 Sample Autonomous
- Wheel intake
- Sweeper mechanism



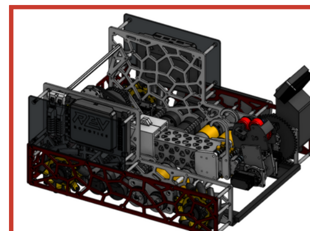
SUPER QUAL (2/8):

High- 288 pts

OPR - 176 pts

Robot:

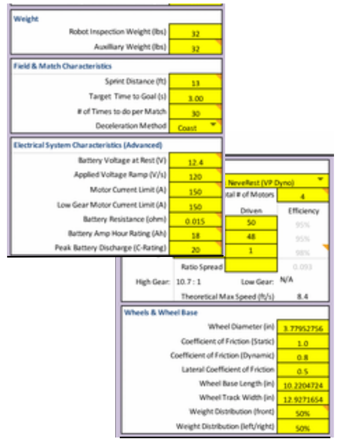
- 3-5 Sample Autonomous
- Limelight
- Level 2 Ascent



DRIVETRAIN

DESIGN REQUIREMENTS

- Keep a low center of mass for good balance and to be both **quick** and **controllable**.
- **Accommodate** our other mechanisms, most notably a **telescope** mechanism that will run through the center of the bot.



Weight	
Robot Inspection Weight (lbs)	32
Auxiliary Weight (lbs)	32

Field & Match Characteristics	
Sprint Distance (ft)	13
Target Time to Goal (s)	3.00
# of Times to do per Match	30
Deceleration Method	Coast

Electrical System Characteristics (Advanced)	
Battery Voltage at Rest (V)	12.4
Applied Voltage Ramp (V/s)	120
Motor Current Limit (A)	150
Low Gear Motor Current Limit (A)	150
Battery Resistance (ohm)	0.015
Battery Amp Hour Rating (Ah)	18
Peak Battery Discharge (C-Rating)	20

NeveRest (VP Dyno)

tal # of Motors: 4

Driven	Efficiency
50	95%
48	95%
1	98%

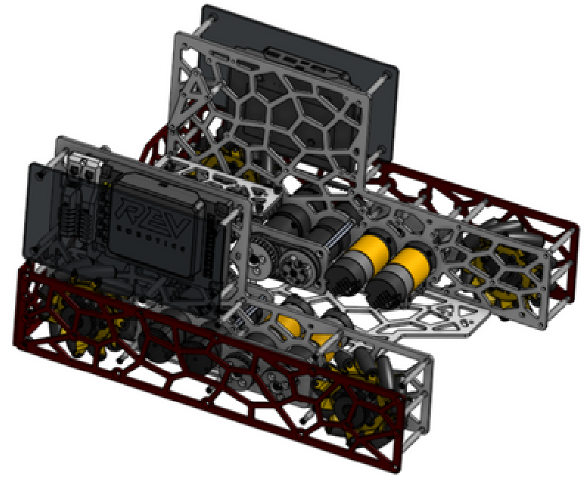
Ratio Spread: 0.093

High Gear: 10.7 : 1 Low Gear: N/A

Theoretical Max Speed (ft/s): 8.4

Wheels & Wheel Base	
Wheel Diameter (in)	3.77952756
Coefficient of Friction (Static)	1.0
Coefficient of Friction (Dynamic)	0.8
Lateral Coefficient of Friction	0.5
Wheel Base Length (in)	10.2204724
Wheel Track Width (in)	12.9271654
Weight Distribution (front)	50%
Weight Distribution (left/right)	50%

we used a spreadsheet to help calculate the optimal motor ratio to make our robot fast



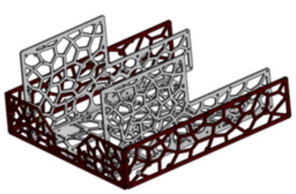
We decided on a 1:1.1 gear ratio with a 435 rpm motor for our drivetrain.

DESIGN ITERATIONS

Issues:	Solution:
The pivoting telescope will generate large moments of **inertia**, making our robot **unstable** and prone to possible tipping.	The 8 motors we use on our robot makes up a **significant** amount of the weight (7.67 lbs). Placing them at the center and bottom of our chassis ensures a low center of gravity, allowing for greater stability.
Issues: If there is no clear area to deploy the intake it may get stuck on top of samples which slows down our cycle time.	**Solution:** Designed the **sweeper** to take advantage of the space beneath the barrier and push samples out of the way.

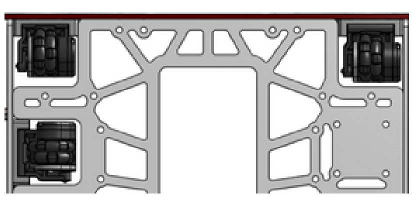
DESIGN FEATURES:

Pocketing



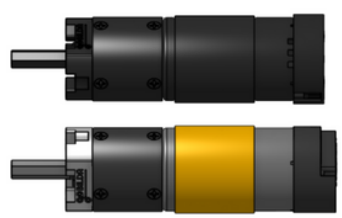
Pocketing allows the aluminum plates to be over 50% lighter, increasing speed.

Odometry



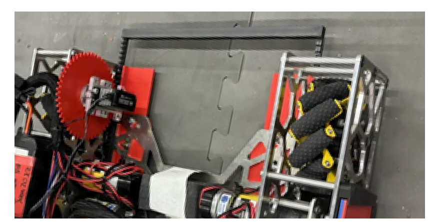
We decided to use a 2-wheel odometry with GoBilda's pinpoint odometry for localization. A 3rd wheel is there as a backup.

Low Profile Motor Caps



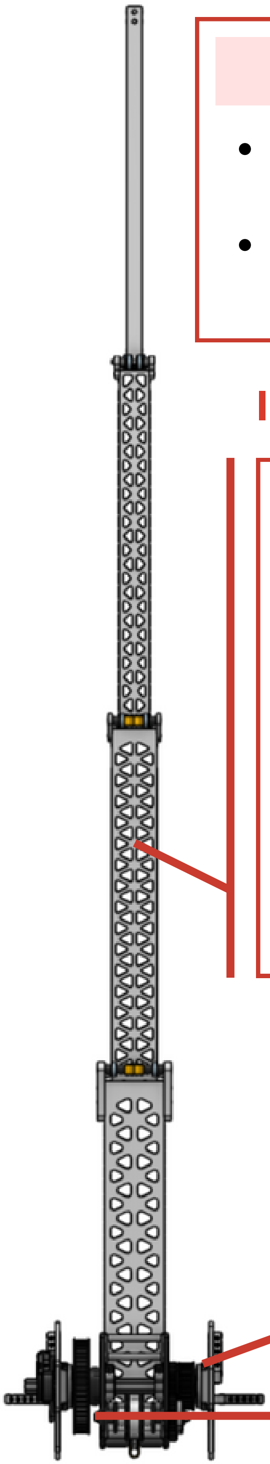
To **optimize** and maximize compactness, we used custom low profile motor caps, sparing us 5.1mm: just enough clearance to make a difference by not hitting our belts.

Sweeper



Thin plastic piece that pushes samples out of the way. We used a rack and pinion design to position the samples conveniently for the roller intake.

TELESCOPE



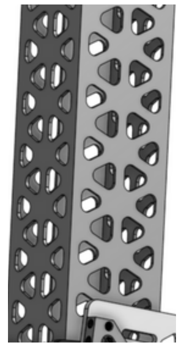
DESIGN REQUIREMENTS

- Needed a mechanism that could reach **Necessary scoring areas quickly.**
- Can extend **both vertically and horizontally**, using all we can of the 42-inch expansion limit.

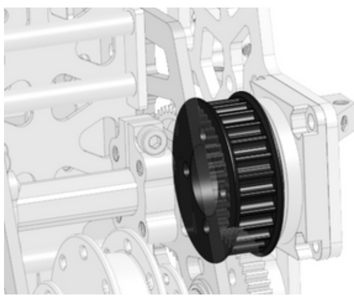
INNOVATIVE FEATURES:

BOX TUBE

The box tube cannot be machined traditionally. Because it is 3D, we needed to use a box tube rig and learn a new machining method in order to cut and pocket them as we did. This both reduces weight and makes stringing much easier.



Using a gear ratio of 1:1.2 on an 1150 rpm motor, the telescope can be fully extended 36 inches in under 400 milliseconds at a speed of about **2.6 m/s**. This allows us to easily score in the high basket for quick and efficient cycles.



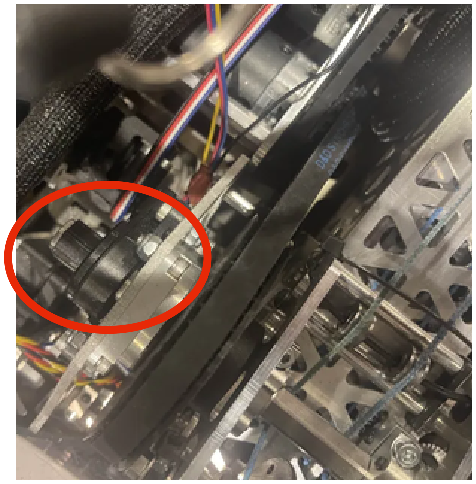
ITERATION 1:

Issues:	**Solution:**
When assembled, the box tube was very loose, with the very last stage drooping 1-2 inches from its intended spot when fully extended.	The problem lied in inaccurate 3D prints. Through reprinting and trying multiple tolerances, we found a good balance between **accuracy and friction**. The final stage now droops less than a fifth of an inch from its target!

ITERATION 2:

Issue:	**Solution**:
Due to the size, the backlash from the pulleys and motors becomes a big problem, and makes it difficult to stay within our expansion limit.	We decided to incorporate a hard stop, physically preventing the telescope from going past where we want it.

- Absolute encoders give accurate position data, **mitigating the effects of the backlash** through code caused by the pulleys and motors.
- Low-density pocketing reduces weight and makes stringing the telescope much easier.



INTAKE

DESIGN REQUIREMENTS

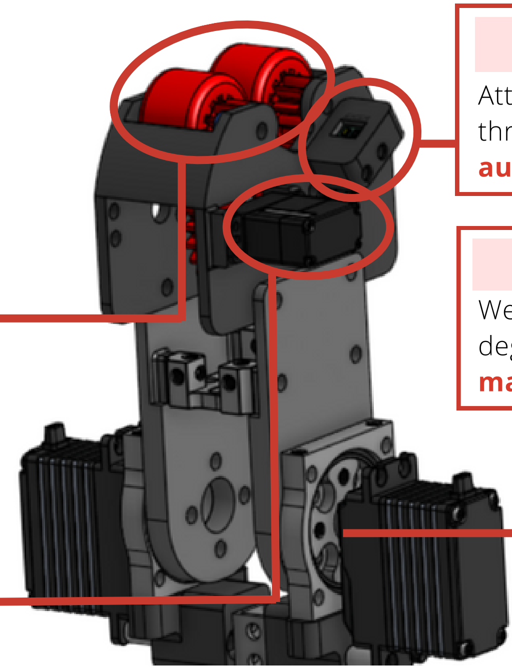
- **Quickly** and **efficiently** intake samples
- Work in **unity with our telescoping arm**
- Avoids pushing samples away from the robot

TPU WHEELS

We 3D printed the wheels out of **compliant** TPU, allowing for **easy collection and storage** of the sample without it being expelled easily

POWERING

Due to the small scale of the intake, we can power this using a geared down **AXON MICRO**, allowing for near **instant intaking** once in position.



COLOR SENSORS

Attaching color sensors allows us to speed up our cycle, either through **automatic expulsion** of an incorrect sample or **automatic retraction** of a correct one.

INNOVATIVE WRIST

We previously used a differential wrist to give our claw an extra degree of freedom. However, we found that it was **hard to maintain** and not necessary.

Our current wrist is powered by AXON MAXes and supported by surrounding pillow blocks. We found that compared to our differential wrist, the servos were much more durable and far better at **resisting accidental impacts** without burning out and breaking.

ITTERATIVE PROCESS

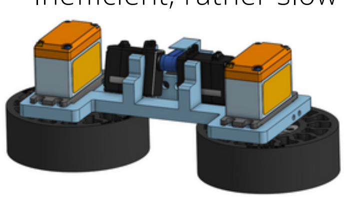
SIDE WHEEL

Pros:

- Large margin of error when intaking
- Can intake in any orientation

Cons:

- Too bulky
- Wheels would get stuck on samples
- Pivot servos prone to breaking
- Inefficient, rather slow



CLAW

Pros:

- Much faster
- High adaptability
- Larger, stronger servos
- Allows driver to adjust angle based on sample orientation

Cons:

- Wrist prone to breaking
- Limited speed
- Too precise, hard for driver

WHEEL (CURRENT)

Pros:

- Only requires contact on **one corner** of a sample to intake
- Very short distance allows for near instant intaking (cycle time ~4-6 seconds!)
- Lightweight, compact, and fast, allowing for precision
- Works well with the robot's main scoring method

Issues: Cannot score specimen	**Solution:** Instead of going to low basket and scoring less, we can drop off samples after the bucket is filled to the human player station where our teammate will score them
Issues: Gets stuck on samples if it is deployed on one	**Solution:** We created a sweeper in order to orient the samples and push them away so that the intake can easily grab them and clear the way for any future cycles

Although this intake cannot score specimen, we believe this is the better choice because this drastically **improves our cycles times**, allowing us to be one of the fastest sample robots in the world. At high levels, we can still support our teammates.

CLIMB

DESIGN REQUIREMENTS:

- Needed a **very fast, yet very consistent** ascent that doesn't intrude much on our tele-op scoring.
- Uses as little additional actuators as possible

THOUGHT PROCESS:

The most glaring thing about the climb is year is the **difficulty** of the level 3 ascent. We knew it would take a large amount of time to design and iterate on this climb. We also noticed how little points this was worth. A level 2 ascent with 2 addition sample cycles would be worth even more, and would take only about 15 seconds. **The opportunity cost for a level 3 ascent was too high.** We decided to focus our energy on a level 2 ascent instead, with plans of a level 3.

Power Calculations

We knew that as long as the telescope can stall and keep the robot up, we can use our pivot motors to move the robot above the barrier and ascend.

2 Motors * 7.9kg stall * (20T/22T) = 14.4 kg

Since our robot is only about 11kg, we are able to reach and stall on the first bar.

2 Motors * 68.4kg stall * (50T/26T) GR = 263 kg

Because motors have the highest efficiency and act ideally at 1/2 the stall torque, we divide this by 2 Ending up about **131.5kg*cm torque** needed.

Torque = radius * force * sin θ

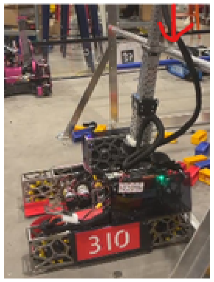
Torque = 13cm * 10 kg * 1

Torque = 130 kg*cm

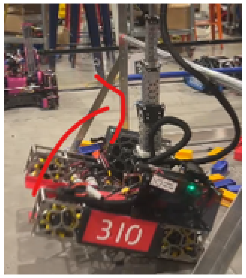
We have enough torque to climb due to the value of θ decreasing as the robot pivots.
The final time it takes us to climb is about 2 seconds!

HOW DOES IT WORK?

We incorporated the climb within the telescope. The 2 retraction motors stall on the first bar, and the pitching motors rotate the chassis, putting us above the barrier.



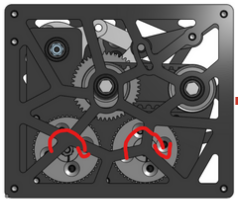
Motors stall



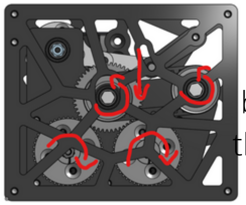
Pivot motors activate

Final climb position

Research &Development:



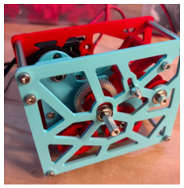
Disengaged, motors drive wheels as normal



Engaged, gear is pushed down and meshes with ones driven by motors and now has the power of both.

Power Take Off (PTO)

Due to a **lack of high power actuators**, we cannot dedicate any more independent motors a climb. We implemented a PTO between our drivetrain plates that transmit the drivetrain's power to hooks which will be actuated by the power of 4 drivetrain motors.



Working gear shifter PTO prototype, a different version is on our robot.

Challenges:

- Time Constraints
- Storing the hooks without interference
- Unpredictable robot path moving upwards
- Achieving a final, stable robot position with actuators off

Polycarbonate Strip Hooks

Inspired by FTC 14343 Escape Velocity, we plan to climb using strips made of 0.1 inch polycarbonate. Due to its flexible nature, we are able to store and release it in order to reach the second bar quickly. Using with string, we are able to pull the robot to the second rung.

SOFTWARE DEVELOPMENT

Robot Mechanisms

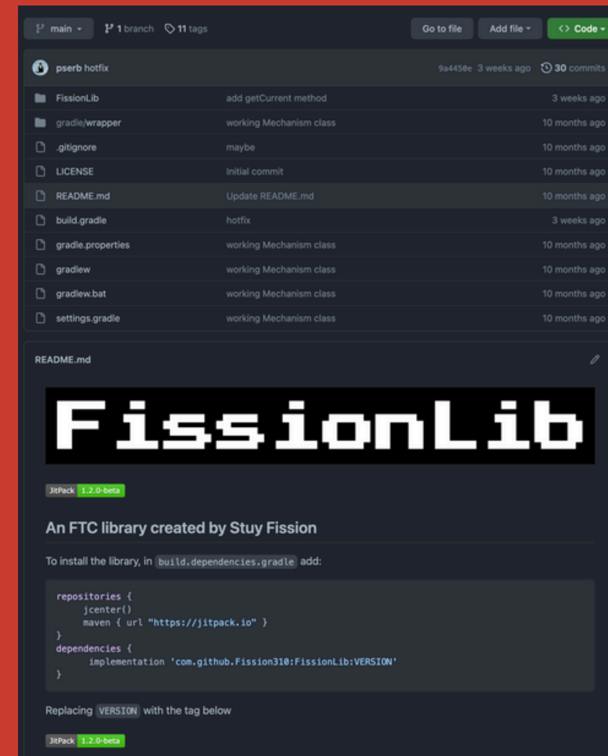
- Decided to use Java via Android Studio to program the bot for versatility.
- Chose GitHub for remote distribution and collaboration among team members.
- First step in coding the robot mechanisms (pivot, scoring arm, claw) was to set limits for the servos.
- Used an Axon Servo Programmer to set limits for each servo and recorded numerical right and left bounds.
- Utilized FTC Dashboard to adjust and tune constants in real time while the bot was running.
- Created Mechanism classes from FissionLib to map player-held controllers and their buttons to servo and motor functions.

Drivetrain and RoadRunner

- Used Roadrunner Library for field coordinate-based robot control with headings.
- Enabled complex autonomous movements using trajectories and splines.
- Tuning was more complex than PID but offered better accuracy and flexibility.
- FTC Dashboard's live error graphs simplified the tuning process.
- Added a two-wheel odometry system to track robot position.
- Odometry enables autonomous localization and improved accuracy over motor encoders.

Limelight

- Used Limelight during autonomous to detect and analyze samples, gathering target angles.
- Started by detecting correct colors for samples and creating silhouettes to remove shadows.
- Tuned constants helped distinguish samples by leaving edges and removing shadows.
- Found center points from images, providing 3 angles for calculation.
- Used these angles to determine strafing, sweeping, and extending the telescope for sample collection.



FissionLib

Our very own custom open-source FTC library is hosted on GitHub and made available to other teams as a package through JitPack. It features a class to easily implement motion profiling for a DC motor. To make robot functions simpler to implement, we implemented our own CommandSequence classes to allow teams to incorporate separate threads and run sequences of functions concurrently without disruption.