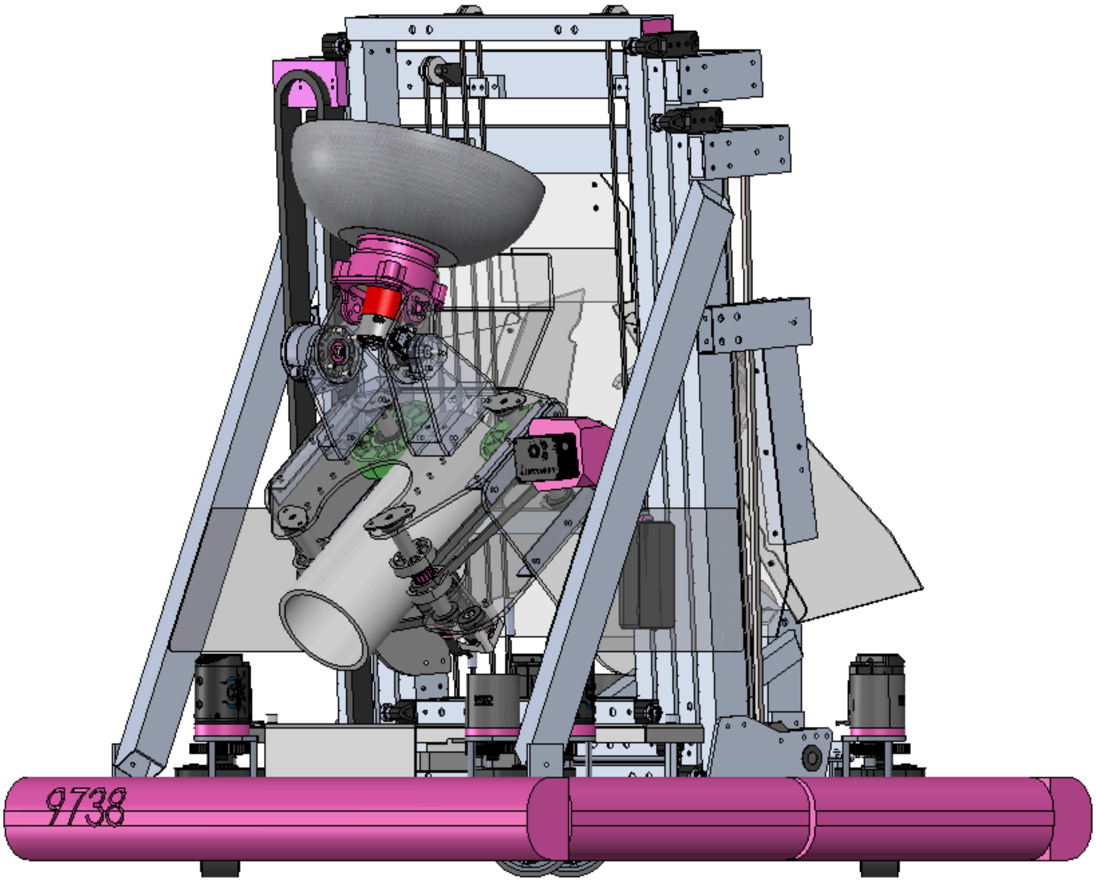
9738 Ionic Bond

FIRST ROBOTICS COMPETITION

Technical Binder

2025 REEFSCAPE - ZINC



ionic.bond_

frctelmond@gmail.com

Ionic Bond

Table of Contents

ZINC - Overview

Vacuum

Intakes Algae from the Reef and scores in the Processor and Net.

Lift

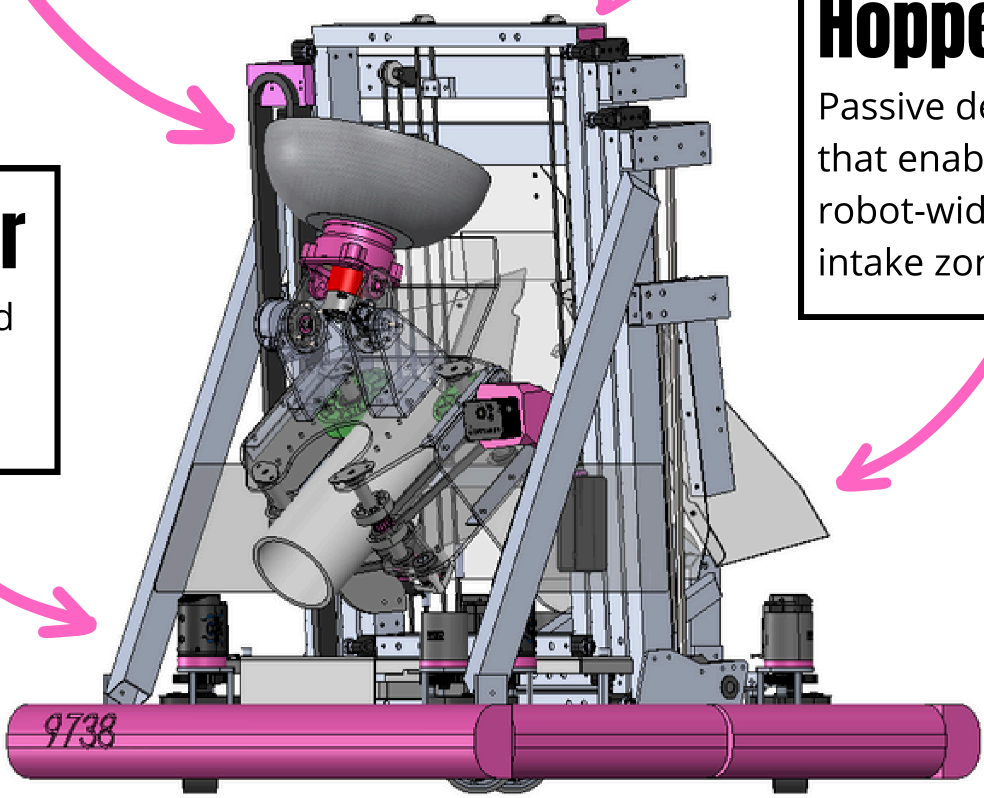
Self planned lift. Places game pieces at different heights from L1 to the Net.

End-Effector

Stores the coral and unloads onto the Reef.

Hopper

Passive device that enables robot-wide intake zone.



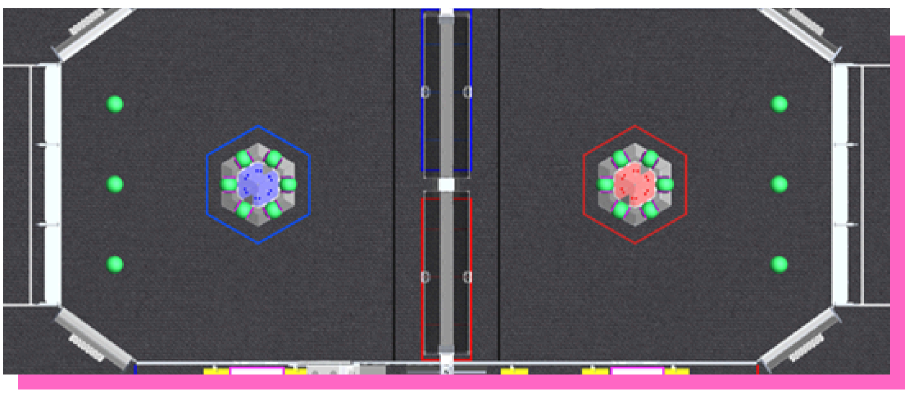
Game Analysis

Scoring Points

- **Fast and short cycle game** in which cycle times can be reduced a fast and maneuverable robot.
- **Intaking from one side and scoring on the other** is valuable especially in autonomous.
- **Algae is important in high-level gameplay** where the reef is going to be filled, and for Coopertition points.
- **Deep Climb is valuable scoring** and is worth not only for RP.

Obtaining Ranking Points

- **Three Ranking Points for a Win -** makes winning a match the most significant, and there should be our focus
- **Deep only, No Shallow -** shallow doesn't give too many points, and also isn't effective enough to acquire ranking points. If we climb it would be deep only.
- **Autonomous -** reliable autonomous is a must for acquiring the Auto RP.



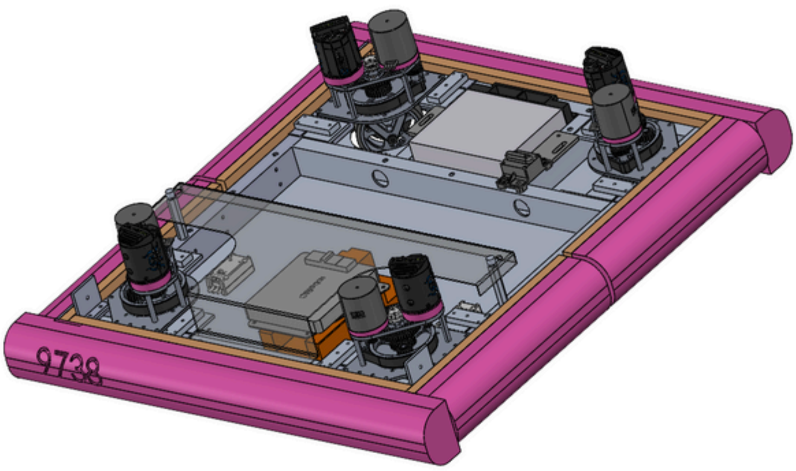
Drivetrain

Requirements

- High maneuverability and controllability
- Low center of mass (CoM)
- Robust and rigid

Details

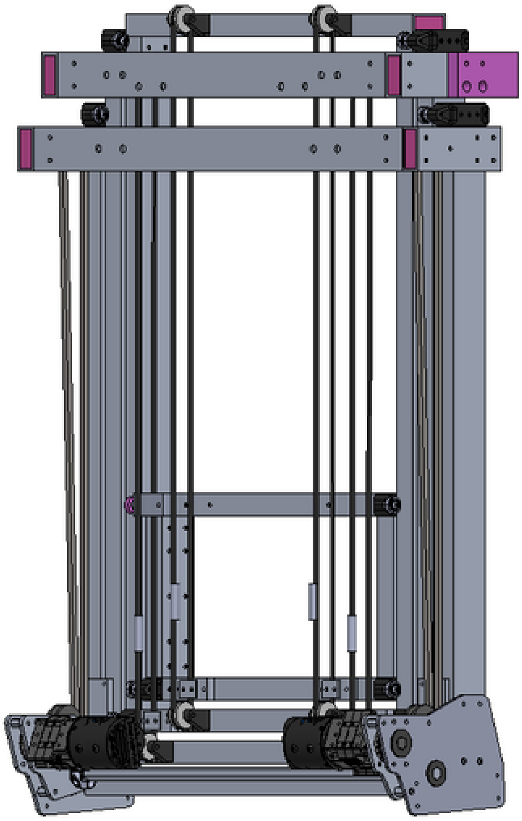
- Constructed with 50x25x3 mm aluminum profiles, connected by a 2mm aluminum belly pan.
- Under the belly pan is a 6 kg, 10 mm thick steel plate to keep the CoM low.
- A 2mm polycarbonate electronics cover is used to prevent damage and to avoid game pieces getting stuck.
- SDS COTS SDS MK4 Swerve Modules, powered by Kraken X60 motors and steered by NEO motors.
- Controlled by YAGSL (Yet Another Generic Swerve Library), a versatile and widely used swerve library with a generic implementation that supports most vendor hardware. YAGSL requires minimal setup and tuning, making it an ideal fit for us. **Additionally, we are proud to be the YAGSL Ambassador team in Israel.**



Lift

Requirements

- very rigid and robust to allow for consistent L1-L4 scoring, processor and net
- Low Center of Mass (CoM)
- Coral should pass through the elevator
- High controllability



Details

- A three-stage cascade elevator driven by two Kraken Motors with a 25:1 gear ratio, using a MAXPlanetary COTS gearbox.
- The elevator is tilted by 4 degrees to enable more robust Algae intake from the reef and L4 scoring.
- The first stage is driven by #25 chain, and the other two stages are powered by Kevlar rope. Driving from both sides increases rigidity.
- The structure is built from 25x50x1.5mm rectangular profile frames, with stress-bearing profiles having a 3mm thickness for added robustness.
- MAXLift COTS bearing blocks were used to reduce manufacturing complexity, modified by adding a 0.4mm steel sleeve to fit on the 50x25mm profiles.
- The drivetrain-width elevator is fixed directly to the drivetrain frame with 5mm aluminum plates for rigidity and to enable passthrough.

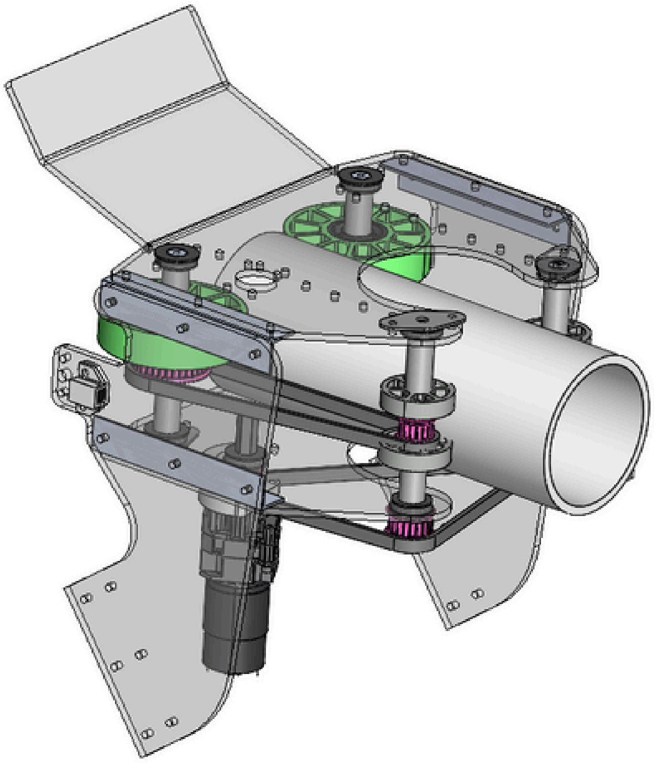
End-Effector

Requirements

- Lightweight in order to be center of mass (CoM) friendly
- reliable, simple and robust
- integrate with the hopper

Details

- Constructed from 5mm polycarbonate plates, one bent for coral centering
- 4in Compliant wheels on the back to allow centering of the coral into the subsystem. 2in Compliant wheels to allow for a better grip of the coral and accurate placement.
- Powered by a 775 Pro motor with a 4:1 gear ratio.
- Two Beam Break IR sensors for tracking the coral's position and returning feedback to the driver.



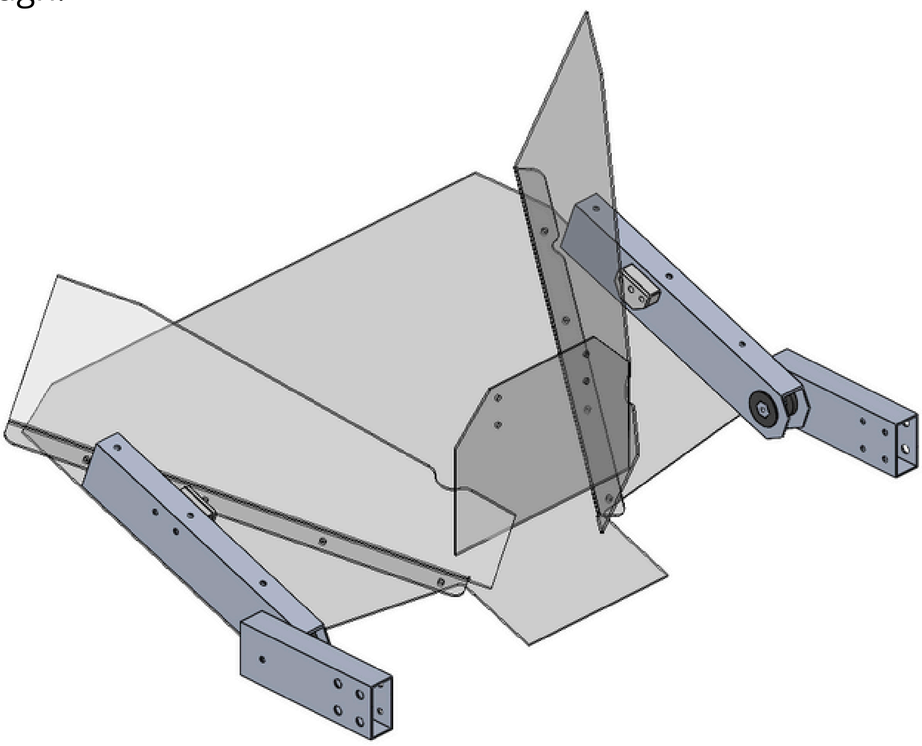
Hopper

Requirements

- Lightweight and Center of Mass (CoM) friendly
- Passive for reducing weight and mechanical complexity and increasing robustness
- Full-Width intake from the coral station
- Integrate with the End-Effector and the Elevator
- Don't interfere with the climbing

Details

- Constructed from 2mm polycarbonate sheets.
- Concaved to allow for better coral centering.
- Connected by a 1/2" hex shaft so it can fold and allow the chain to run through.



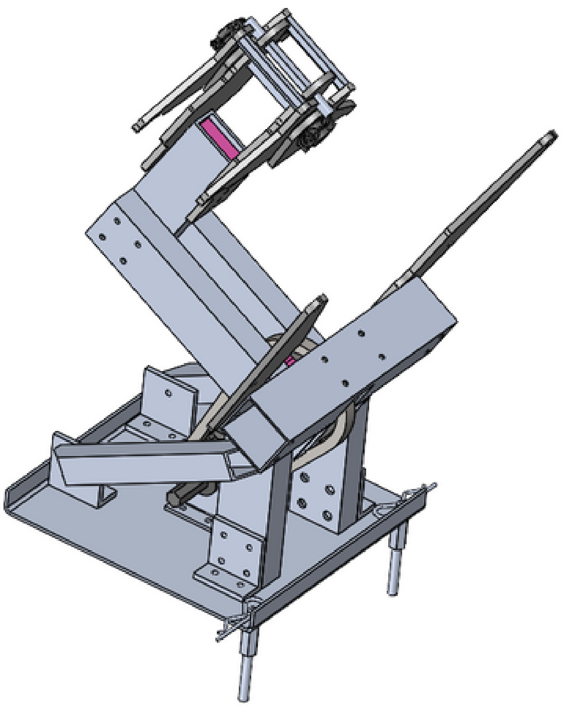
Climber

Requirements

- robust and reliable and quick Deep Cage climbing
- integrate with the Hopper
- modular connection to allow working on the mechanism without interfering with other subsystems and robot functions.

Details

- Secured to the robot by 2 Cotter Pins and a keyhole to allow fast connection and disconnection, and access to the robot battery.
- Barb based on the reliable118 Everybot design.
- Powered by a Kraken X60 motor with a 120:1 gear ratio and driven by a #35 chain.
- Features a passive centering device to ease the alignment to the cage.



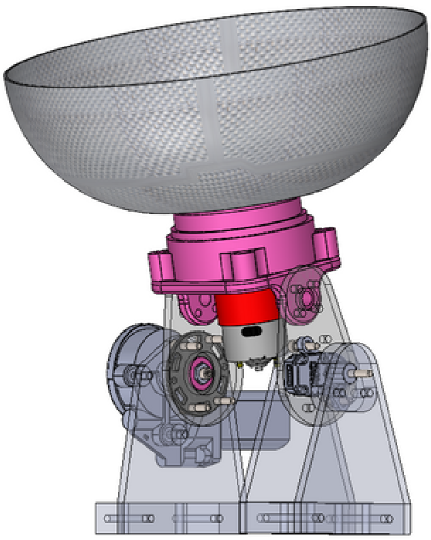
Vacuum

Requirements

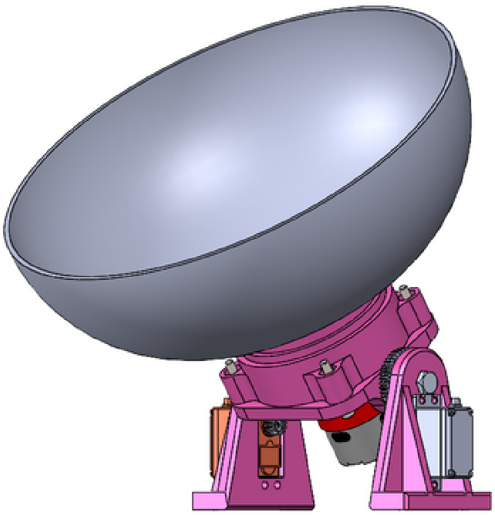
- Intake algae from the reef and score into the processor and net.
- Robust and reliable
- Lightweight and Center of Mass (CoM) friendly

Details

- Constructed of 5mm polycarbonate plates and PETG 3d printed parts to keep it lightweight.
- The suction cup is a custom fiberglass bowl made in cooperation with our sponsor, Nano Fiber.
- Vacuum is generated by a COTS generator that its motor was replaced by the legal AndyMark RedLine motor.
- Was powered by two 5-turn servos until the Pregional where the servos broke. The servos were replaced by a Snowblower motor



Post-Pregional



Pre-Pregional

Making the fiberglass bowl at our sponsor, Nano Fiber

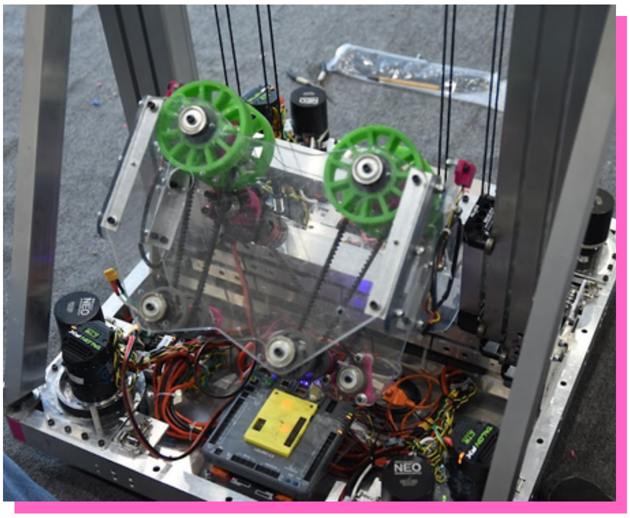
Electronics

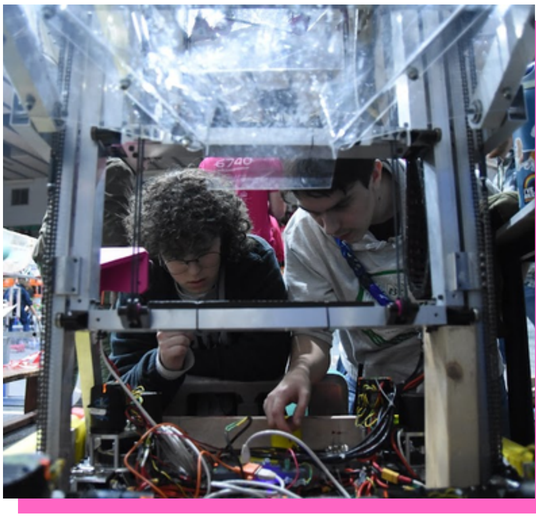
Requirements

- Robust and reliable

Details

- XT60 connectors are used for motor cables and are soldered in house.
- Dupont connectors are used for under 12 AWG wires and are crimped by a Molex crimp and secured by a COTS Dupont clip.
- roboRIO is mounted above the PDH to allow for easier access to the roboRIO ports and more space for the motor cables.
- roboRIO power supply connector was replaced by a more robust connection.
- All End-Effector and Vacuum motors are brushed to shorten CAN Bus length and increase robustness.
- Radio is mounted high to allow for better communication.
- The vast majority of the cables have a protective cover, cable protective chain, or run through aluminum profiles.
- Most of the electronic components are concentrated in one location to increase robustness.

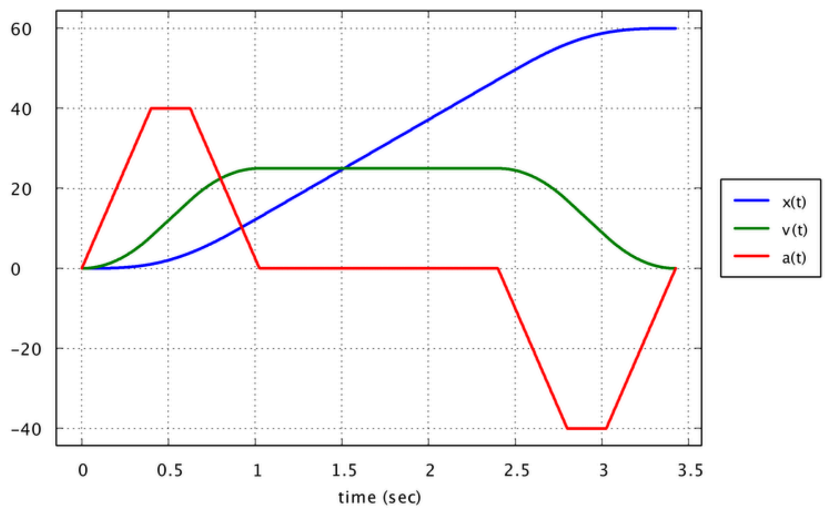




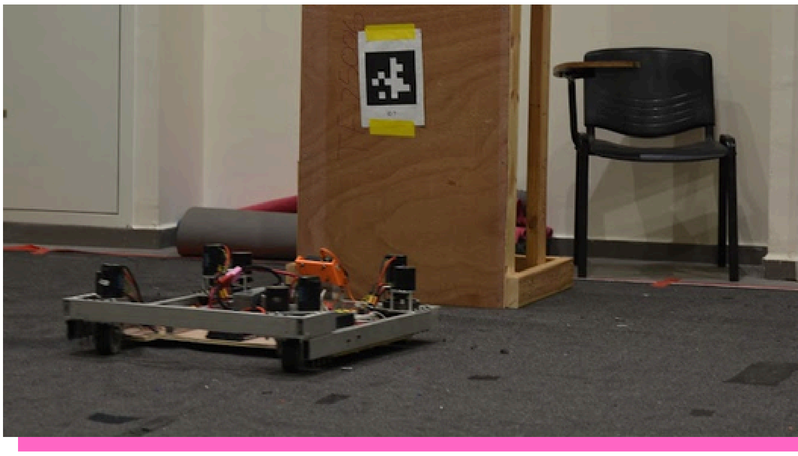
Programing

- **Generic IO Code-** To be able to handle physical or logic changes in the code easily, we create a base interface that each system's IO (inputs & outputs) implementations have to implement in order to act on a system. This makes major changes such as adding the servo pivot to our vacuum system much easier than without this kind of implementation for physical devices, instead of rewriting the entire system, we just had to create another implementation of the system's IO and use it instead of the old one, which also means we can switch back in no time in the future.

- **lift-** In order for our lift to accuratly reach the branches, we need to implement a feedback loop on the encoder's position. Due to the lift being powered by 2 Kraken x60 motors, we have access to using Motion Magic, which allows us to reach setpoints fast and accuratly with minimal tuning.



- **Auto Alignment to branches-** To automatically place corals on branches, we follow these steps:
  - **Locate the Closest Branch**: Identify the nearest branch to the robot on the field, factoring in the robot's angle relative to the branch. This is important because branches on the Reef's bridge between faces vary in angle, requiring different robot orientations.
  - **Adjust Robot Orientation**: Rotate the robot to align with the target branch, using a larger angle tolerance compared to the final movement. This prevents the robot from hitting the Reef while moving.
  - **Move to Coral Ejection Position**: Using a profiled PID controller, move the robot to the branch's coral ejection position, which is the safe spot from which the robot can place the coral.
  - **Lift Adjustment**: Once the robot is a certain distance from the target position, adjust the lift to match the branch's height.
  - **Score Coral**: After reaching the branch and positioning the lift at the correct height, the robot automatically scores the coral. This process ensures precise placement of corals while avoiding collisions with the Reef and maintaining safe positioning.



- **Computer Vision:** Physical odometry, which relies on wheel encoders and other sensors to track a robot's movement, can suffer from inaccuracies over time due to slippage, terrain irregularities, or sensor drift. As a result, robots may lose track of their exact position on the field, leading to errors in navigation and misaligned movements. To address this, vision-based systems like the ones we use are invaluable.

- **Limelight-** By using the limelight's high-precision camera and built-in algorithms, the Limelight provides real-time feedback on the robot, offering a reliable method for correcting odometry errors. The ease of integration and user-friendly interface make it an essential tool for teams looking to improve robot localization and ensure more accurate autonomous performance.

- **Synapse-** In addition to utilizing the Limelight 3G, we have developed our own vision framework, Synapse. Synapse is a highly efficient and precise vision system designed to enable rapid development of extensions and additional pipeline types for vision processing. With Synapse, we consistently achieve superior results compared to the Limelight, with significantly lower position noise over greater distances. While the Limelight 3G provided a maximum of around 20 FPS with ~11ms latency for 3D robot pose estimation from AprilTags, accompanied by noticeable pose estimation noise at larger distances, Synapse delivers a solid 100 FPS capture rate, with minimal latency of around ~2ms and noise. Additionally, it offers enhanced AprilTag detection at much greater distances, primarily due to the different lenses we use.

- **Standard Deviation -** When fusing together physical and vision odometry, it isnt smart to always trust vision 100%, such as times when the robot is rotating fast or the vision estimates it's in an unlikely position, for instance inside the reef or outside the field. To fix that, we use Standard Deviation - a weight is given to each of the axis determening how much we trust them, and according to that WPILIb's pose estimators can decide whether or not to replace our physical odometry with the vision estimates. This allows us to get high precision for aligning to the branches on the field.

$$\sigma = \sqrt[2]{\frac{\Sigma(x-\mu)^2}{N}}$$