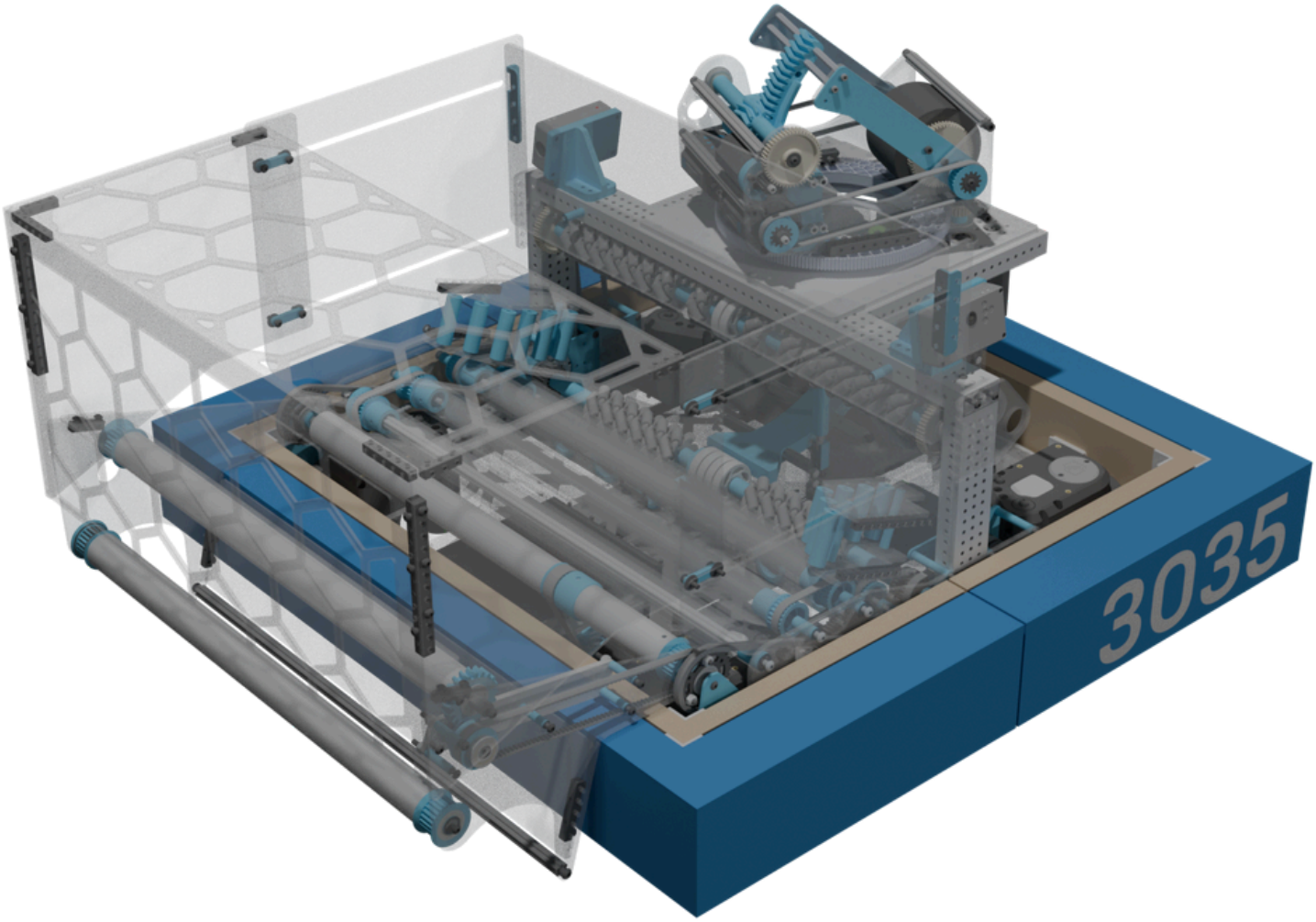


3035 ECHS ROBOTICS PROGRAM

DROID RAGE

2026

Technical Binder

DROID RAGE

Table of Contents

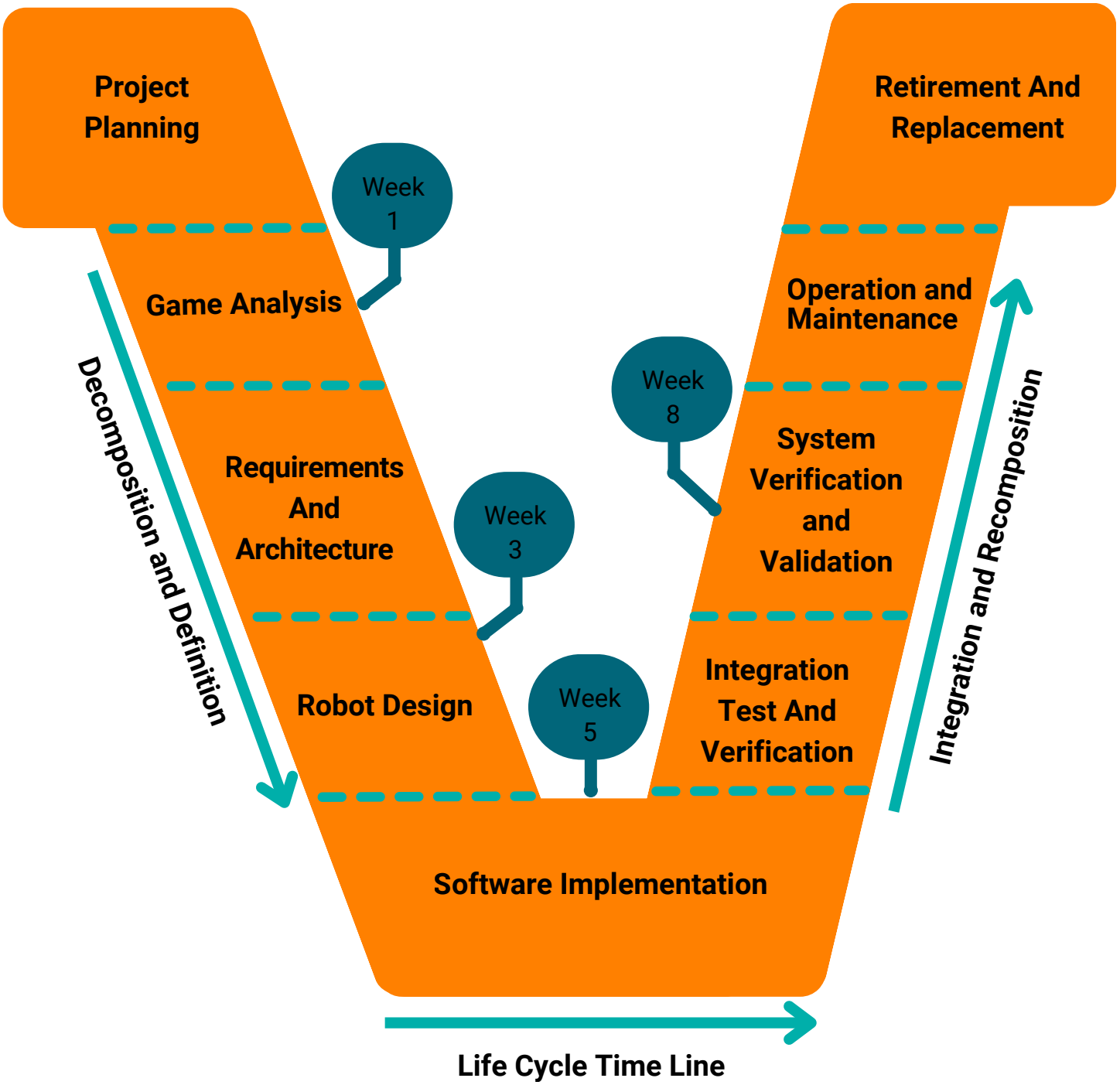
Systems of Engineering **1**

2.1 Systems of Engineering 3
2.2 Game Analysis 4
2.3 Requirements & Architecture 5
2.4 Scoring Mechanism 6
2.5 Drive Train 7
2.6 Indexer Evolution 8
2.7 Climb & Kicker 9
2.8 Intake Evolution 10
2.9 Software Saefty Systems 11
2.10 Software 12
2.11 Software Control & Localization 13
2.12 PID & Feedforward Control 14
2.13 Encoder & Software Integration 15

Future Plans **2**

3.1 Sustainability 16
3.2 Retirement 17

Systems of Engineering

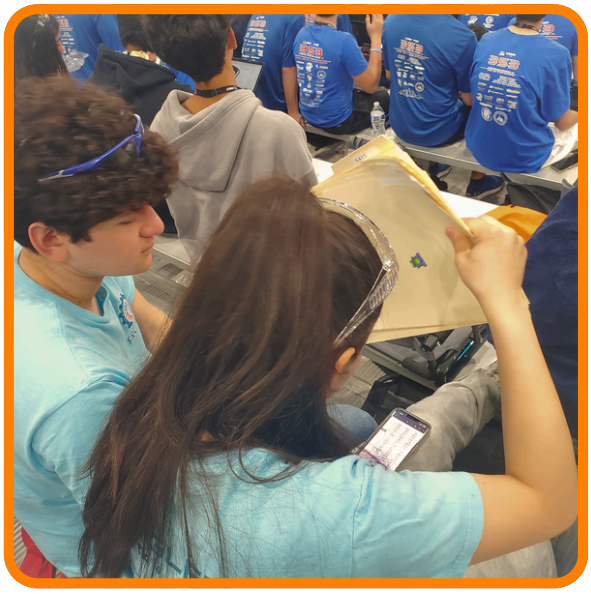


The **Systems Engineering Process**. The process is a structured and interdisciplinary approach focusing on following a series of steps emphasizing on designing, developing, and managing complex systems while considering the entire **life cycle** of **robot** from **concept to retirement.**

Game Analysis

Planning

- **Analysis**: Review REBUILT game reveal & manual to understand objectives, scoring, and regulations.
- **Iterative Design with CAD**: Develop & refine multiple iterations with CAD software, collaborating with mentors.
- **Education and Training**: Assigned tests, scouting assignments, & training sessions to ensure comprehension during competitions.



Central Goals

- Evaluate team **performance** and **compatibility** to select strong alliance partners during playoffs.
- Work with alliance partners and **collaborate** with other teams to **share** useful scouting information.
- Use collected data to **support** strategy decisions before and during matches.
- Gather and **analyze** match data with a dedicated scouting team.
- Use scouting insights to **improve** match strategy, alliance coordination, and overall performance.

Requirements & Architecture

Trends

- Drivetrain: Swerve
- Intake:
  - Slap down intake
  - Polycarbonate rollers
  - 4 bar intake
  - Spindexer
  - Sliding hopper
- Climb:

  Elevator latch

  Hook & winch

Trends in Practice

- Drivetrain: Swerve
- Intake:
  - Active polycarbonate rollers
  - Dropdown intake
  - Sliding out hooper
  - Cat tongue tape for grip
- Climb:
  - Andymark climber in a box stage 1

Needs

Autonomous

- Score 8 preloaded Fuel
- Intake and bring fuel from the center of the field
- Go underneath trench (22.5 in)
- Create reliable autonomous paths from both alliance sides
- Adjustable Shooter to score consistently from different distances
- Winning auto

Tele-Operated:

- Shooting Fuel not in Alliance Shift
- Shooting Fuel in Alliance Shift
- Go under trench
- Defend

Endgame:

- Level 1 climb

Wants

Autonomous:

- Drive over bump
- Level 1 climb
- Go over bump

Tele-Op:

- Dumping
- Shooting to Alliance Zone
- Go over bump
- Turret Auto-aim
- Launch Calculator

Endgame:

- Level 2 climb
- Level 3 climb

Scoring Mechanism

Adjustable Hood

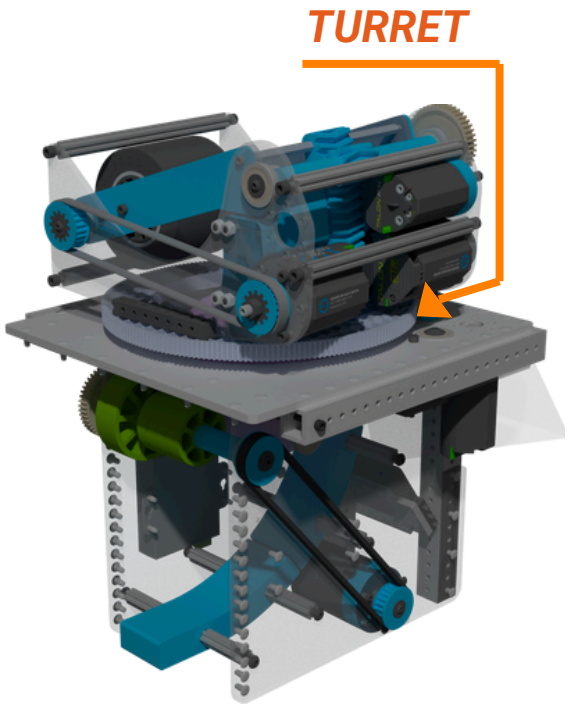
Goal:

Allow the robot to shoot **accurately** from **multiple** distances by adjusting the launch trajectory of the game piece

Features:

- 2 Kraken **X60's** directly powering Two 4" WCP **45A** Flexwheels at **80 ft/sec**
- Double Helical Rack & Pinion Hood Powered by 1 Kraken **X44**
- Gear Ratio **68.9 :1**
- ¼ in **Recycled** Polycarbonate Plates
- **1 Inch** of Compression Ran in unison with 3-D Printed Fuel Guide

Turret



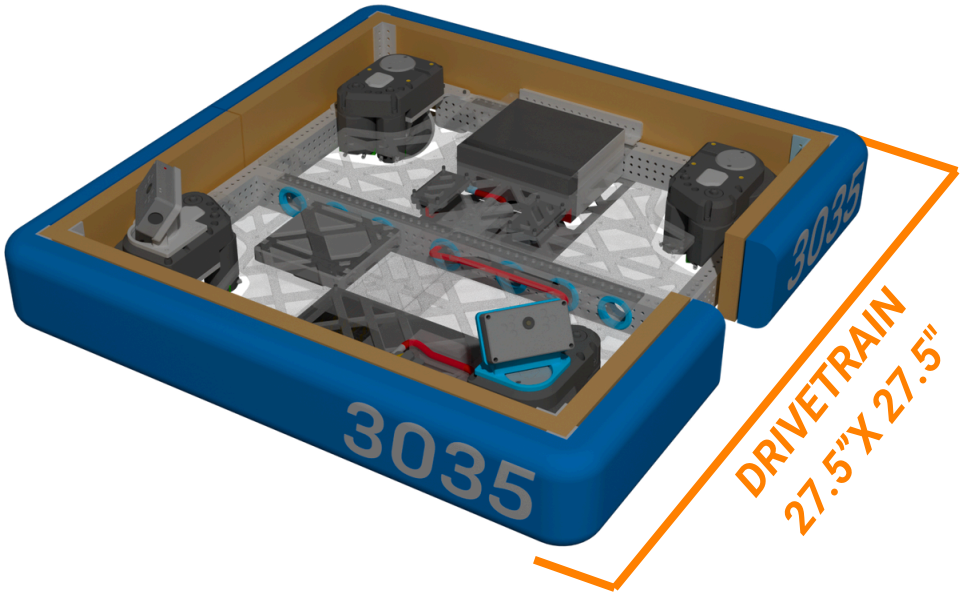
Goal:

Maintain **consistent** target alignment while the robot moves, improving accuracy and efficiency.

Features:

- **6.875 COTS** Andymark Turret Assembly
- Custom Turret Gearbox Powered by 1 Kraken **X44**
- WCP ThroughBore Encoder tracks turret position
- **40 :1** Gear Ratio
- ¼ Recylced Polycarbonate Plate & ⅛ Aluminum Turret Plate
- **270** Degrees Of Freedom

Drivetrain



MK5N Swerve

- Fully Enclosed gear driven system running a **4 wide x 2.25** wheel
- Drive motor powered by a **X 60**
- Steering motor powered by a Kraken **X44**
- **5.36:1, 15-18 ft/sec**, absolute encoder used for steering position feedback

Chassis

- Dimensions **27.5" x 27.5"**
- **1/4"** recycled polycarbonate split belly panel
- Electronics mounted underneath the belly panel
- Belly Panel **1 ½"** off the floor, raising the Center of Gravity (CG)

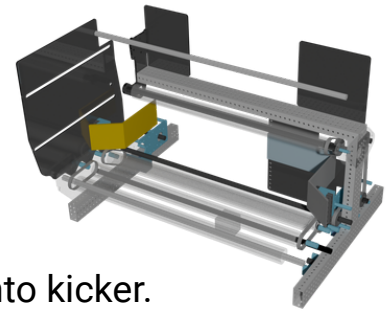
Indexer Evolution

Indexer V1

Pros:

Ideal **0.5"** compression

- Able to give the fuel strong momentum into kicker



Cons:

- Fuel jamming trying to go into kicker
- Balls favored equally on both sides
- Wheels, on mecacum shaft, intersecting with wheels intaking into kicker.
- Balls getting stuck between Polycarbonate rollers

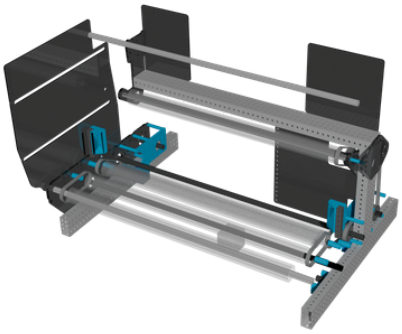
Indexer V2

Pros:

- More **effective** ball funneling than V1, increasing BPS and scoring capability.

Cons:

- Fuel **jamming** still occurs when feeding into the kicker due to loss of power
- Less fuel capacity



Carried over from V1 to V2 based on testing:

- 2" Vector Intake Wheel for intaking to kicker
- Climb placement
- 1 Kraken X60 on a 3:1 Gear Ratio providing ample torque & speed

Refinements made from V1 for V2:

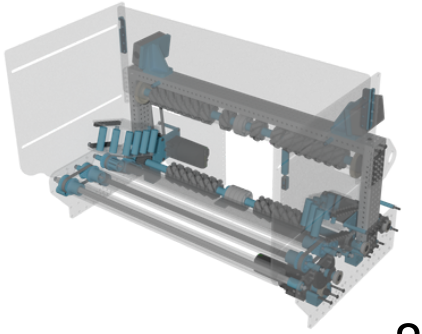
- 1" compliant wheels to 2"
- Added 4 2" Rev wheels on bottom shaft

Indexer V3

Pros:

Ideal compression

- L1 Climber removed to add a Kraken X60 on a 1:1 powering top roller leading to increased BPS (Balls per second)
- Bottom roller powered by 1 Kraken on a 3:1 preventing jamming
- Angled 3-D printed rollers for faster indexing



Cons:

- Lack of compression between kicker and indexer

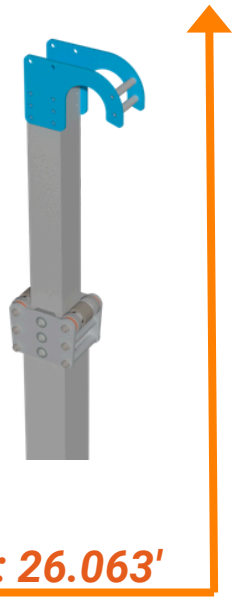
Climb & Kicker

Climb

Goal: Enable the robot to lift from the ground **quickly** and **reliably** during the endgame with little to no backdriving

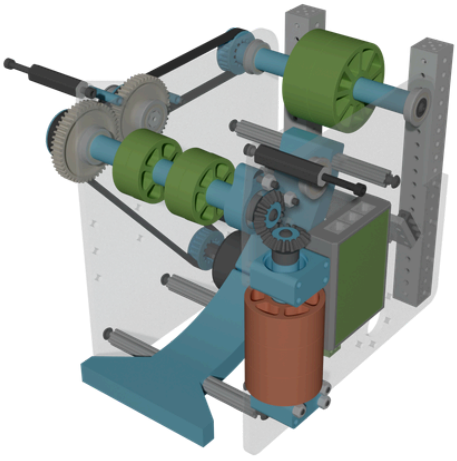
Features:

- Andymark Climber In a Box Stage 1
- ⅛ CNC'D Polycarbonate Hooks
- Nylon cable string
- Powered by Kraken X60 on a 48:1 ratio, with a 1:1 Rev Brake to prevent backdriving
- Resting height: 17.563, extension height: 26.063"



EXTENDS: 26.063'

Kicker



KICKER

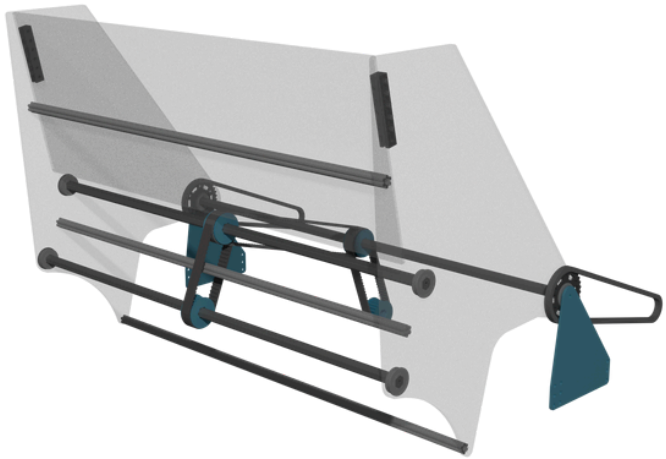
Goal: Provide guided acceleration to propel fuel upward into the shooter.

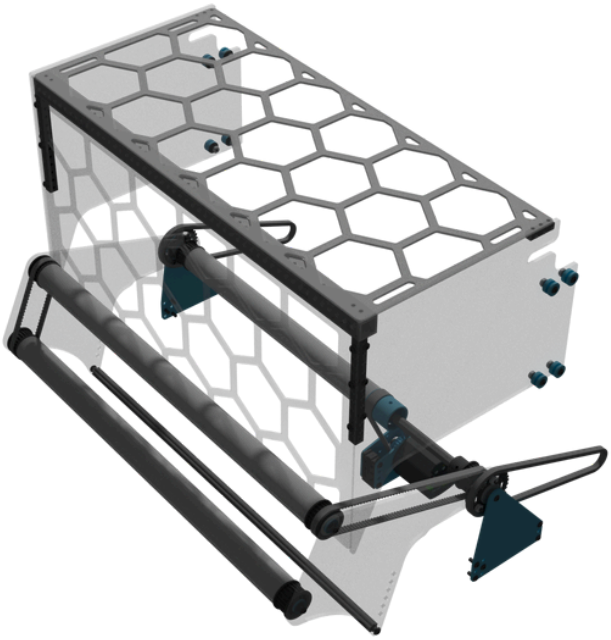
- Powered by 1 Kraken X60 Motor
- 2, 24 tooth pulleys powered on a 1:1 ratio.
- 4, 2" compliant wheels running 0.5 inches of compression
- Custom 3-D Printed Ramp used to guide fuel into the shooter

Intake Evolution

Intake V1

- 2 in, ½ in medium compliant wheel intake for grip
- Compression wasn't enough to intake fuel from the ground
- Minimized ability to score
- Wouldn't go low enough to pick up the fuel.





Intake V2

- Switched to a 1.75in polycarbonate roller for more consistency
  - Reversed the direction of first polycarbonate indexer roller to accommodate intake.
- Increased the rate at which we could intake fuel constantly
- Added a polycarbonate plate to increase the consistency of compression on the fuel when intaking

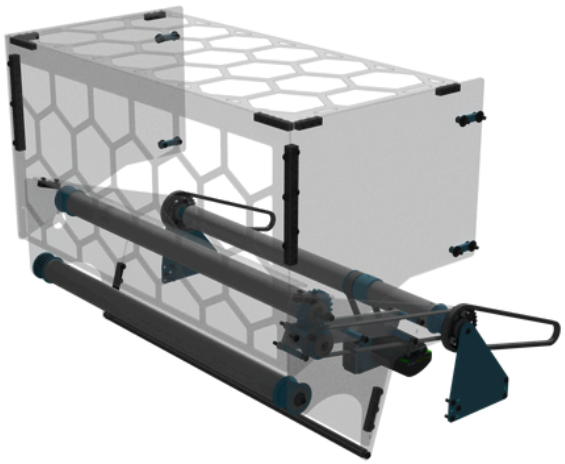
Intake V3

Pros:

- 3-D printed hopper guides for easier extension
- Touch it own it intake

Cons:

- ½ Shaft for polycarbonate back plate would bend due to the amount of fuel absorbed



Software Safety Systems

Real-Time System Alerts

- Implemented alerts to signal key warnings, such as low battery voltage, CAN errors, and hardware overheating
- Identifies issues early, preventing longer diagnosis time

System Safety Hardware

- Intuitive interface for controlling motors and subsystems
- Writers stay active even when motors are off, enabling telemetry checks to verify functionality without power
- Standardizes motor implementation using Java abstraction for consistent hardware management

```java
/**
 * Used to enable/disable the motor.
 * @param isEnabled
 * @return TalonEx (for call chaining)
 */
public TalonEx withIsEnabled(boolean isEnabled) {
    this.isEnabled=isEnabled;
    return this;
}
```

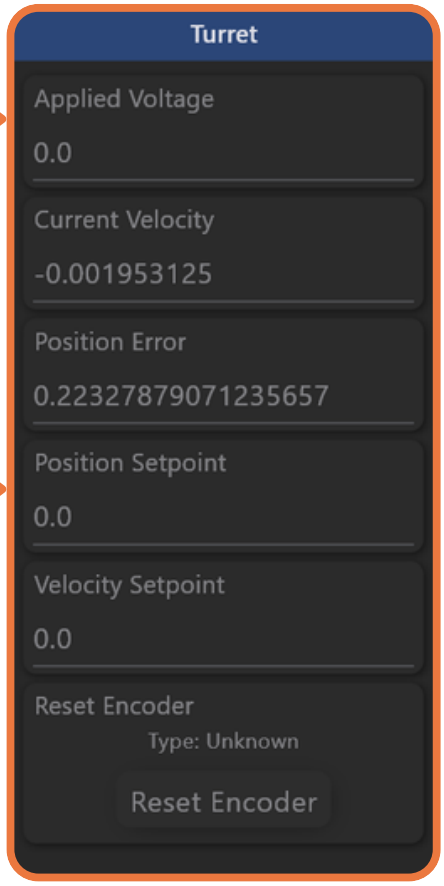
Software

Simple Motor, Encoder, and Subsystem Configuration

- Forces implementation using Method Chaining of various constants: offsets, idle mood, position conversion factor, velocity conversion factor, current limits, and more
- Creates a clear and easy base to work from
- Creates writers on Elastic with voltage given and encoder positions

The Amount of Voltage going to the mechanism **if enabled**

The Goal Position of the **Current Mechanism**



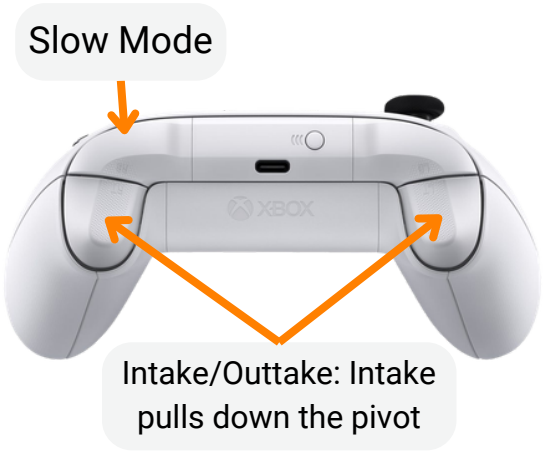
Automated System Identification

Our team uses a custom implementation of WPIlib's System Identification (SysID). We have implemented it in such a way where the four tests that SysID runs on our motors can be completely automated by stopping at the soft limits of our mechanisms.

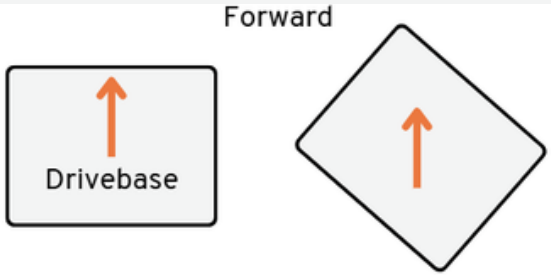
```
new SysIdRoutine.Mechanism(
    (voltage) -> {
        // Only apply voltage if within safe bounds
        // double currentAngle = getCurrentAngle().getRadians();
        // if (currentAngle >= minAngleRad && currentAngle <= maxAngleRad) {
            setVoltage(voltage);
        // } else {
        //     setVoltage(0); // Stop if at limits
        // }
    },
```

Software Control & Localization

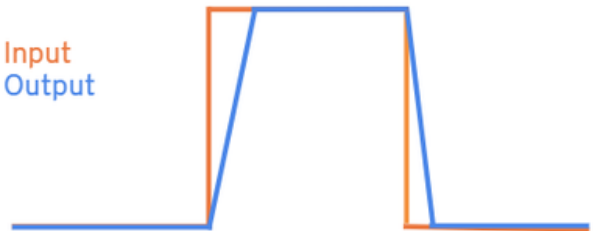
Driver Controls



Field-oriented Driving using Joysticks



Slew Rate Limiter: prevents quick rates of changes to avoid tipping



Vision-Assisted Odometry

- Seeds odometry with AprilTag vision
- Distance-weighted tag measurements
- Prevents large pose jumps
- 3–5 sec faster cycle times
- 10% scoring accuracy improvement

PID and Feedforward Control

Pure Feedforward

- Controls velocity and acceleration, but not position
- Works for small movements only
- System noise builds up errors over time

VS

Pure PID

- Less predictability as it reacts to errors
- Can have systems oscillate near setpoint
- Does not compensate for gravity on arms and elevators

Combination

- Faster Response
- Better stability - Reduces overshoots and oscillations
- Feedforward counteracts gravity and provides base control, allowing PID to focus on fine movements and precision using only kP

Trapezoid profiling

- Smoother motion
- Allows for limiting of velocity and acceleration during motion
- Gradually adjusts PID and Feedforward setpoints to reach a specified goal position

CTRE Motion Magic

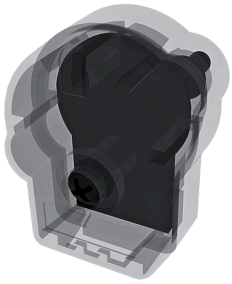
- Same advantages as Trapezoid Profiling
- Includes its own motion profiling
- Runs directly on Talon motor controllers
- Takes performance load off RoboRIO

Encoder & Software Integration

Encoders

1.) CANcoder

- Tracks the absolute rotation of our swerve wheels
- Custom implementation in code to simplify configuration



2.) Through Bore Encoder

- Tracks the position of our turret at a higher resolution than the motor's built in encoder
- Software converts rotations into radians and degrees

3.) Pigeon 2.0

- A gyro used to track Nomad's pitch, roll, and yaw
- Data used to implement anti-tip code and Field-centric driving

Organized Wiring

- Labeling all of our wires allows for us to be both safe and more efficient
- Wiring Diagram ensures every wire is accounted for
- Allows for quick identification of wires for quick troubleshooting

Sustainability

At the end of the season, the team inventories all components and determines whether they will be **kept, repurposed, or recycled**. The chart below shows this process.

We reach our sustainability standards by souricng our polycarbonate with a local company, **A&B Plastics**. By using the donated polycarbonate, the team is able to save it from going to the **landfill**, & use the **budget** for other expenses.

Retirement

Off-season

Once the season ends, we start our off-season outreach & recruitment. We will revaluate the successes and challenges of **NOMAD** during the **REBUILT** season, & use them as lessons to improve the future.

We will use **NOMAD** for:

- Buc Days FRC Robotics Rodeo- May 30-April 10
- Outreach Events (STEM Expos, showcases)
- Primary robot for driver's practice and software updates.

Recruitment

While awaiting the new season, **NOMAD** will be used for our recruitment events like our annual mixer. The robot will serve for demonstrations & workshops.