Engineering Portfolio

Silver Eagles #21428

2023 - 2024

Meet Our Team

We are the **Silver Eagles** from Salesian High School in New Rochelle, NY. We first formed in 2021, starting with 4 members all new to the world of robotics. We've since grown to 13 members and have setup an infrastructure to be able to teach new members the skills they need to contribute in many ways. We're comprised of students, ranging from 9th to 12th grades, all from the NYC and Westchester areas.

David
Coach

Charlie
Captain

Michael
Captain

Jorge
Lead Builder

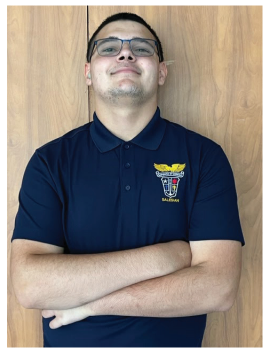
Andres
Youth Mentor

Brandon
Mentor

Curtis
Programmer

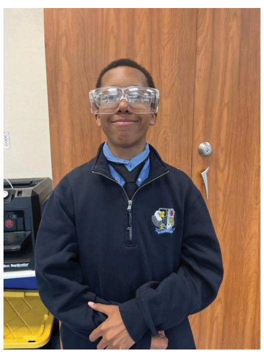
Matthew
Lead Designer

Quincy
Operations



Aiden
Builder

Ms. Gomez-Botero
Mentor



Wilkin
Builder

Walter
Youth Mentor

Kai
Builder

Alex
Builder

Goals

Outreach

- Inform current students not yet involved in engineering about First and the various robotics and engineering opportunities in the school
- Teach students from other schools in the distract about engineering in our Saturday Academy program.

Team

- Have new and existing members learn new skills
- Improve communication between sub-teams
- Make sure everyone has fun and feels included

Robot

- Learn how to implement a pathing algorithm to autonomously control our robot
- Be able to consistently launch a drone and score points

Post-Competition Reflections

	What Went Well	Improvement Areas	Action Items
First Competition	• Drive train is intuitive and reliable • Push capability on robot came in handy to help score points	• No autonomous • No drone launcher • Can't pick up more than 1 pixel • Can't pick up pixels reliably	• Add autonomous functionality • Switch from passive intake to active intake • Add drone launcher
Second Competition	• Autonomous parking Active intake can pick up pixels • Able to reliably place pixels on board one at a time	• Autonomous program can't place pixel on board • Drone launcher can't clear the field reliably • Active intake sometimes takes a few seconds to bring pixel onto plate	• Add ability to pick up and place two pixels at once • Add more detailed autonomous capability Improve drone launching capability • Add hanging capability

Community Outreach

Our commitment to community outreach is at the core of our First Tech Challenge journey. Through programs like the Saturday Academy, collaborations with Edgemont Robotics, and engagement with Don Bosco Prep, we strive to inspire, educate, and empower the next generation of engineers. These initiatives showcase our technical prowess and exemplify our dedication to building a solid, inclusive robotics community.

Saturday Academy

One of the ways we reach out to the community is by holding a Saturday Academy, **a free enrichment program for local students in 6th-8th grade to allow them to study topics that aren't taught in most schools in our area**. They can learn fundamentals of robotics, coding, and problem-solving every Saturday. We aim to inspire the next generation of engineers and innovators through engaging hands-on activities and interactive sessions. They work directly in our lab, and most of the materials they use are 3D printed by our lead 3D designer, **Matthew Carrasco,** during our meetings.

School-Wide

Alternatively, we source our outreach through community held **club fairs, in-school advertisements, as well as announcements** relating to the team's success and status within competitions. This has attracted several new members, many of whom take leadership roles quickly. Such outreach has also provided an interest and cooperation from those outside of the program and realm of robotics, not only assisting in the development of already existing projects and ideas, but allowing new ones to be created through such collaborations. **We also inform current students, that aren't yet involved in engineering, about *First* and various robotics + engineering opportunities in the school, through our school's morning broadcast**, AM Salesian.

Collaborations

Our involvement with **Edgemont High School's robotics team** has led to notable achievements, including successful competition performances and increased student participation. Our teams come together to practice and share ideas throughout the season. The camaraderie built through this collaboration has created a robust support network within the broader robotics community, fostering an environment where knowledge and expertise are freely exchanged.

Our commitment to positively impacting students' educational experiences is at the heart of our community outreach. Our team actively engages with our brother school, **Don Bosco Prepatory High School** in Ramsey, New Jersey, who helped us with our process of **building up our team in our opening stages two years ago**. This initiative introduced students to the world of robotics and aligns with our mission to promote STEM education and foster a passion for technology.

Through sustained efforts at Don Bosco Prep & their robotics team, we have **witnessed a significant growth in interest and participation in STEM-related activities**. By incorporating robotics into our school's educational landscape, we contribute to the development our students, preparing them for the challenges and opportunities in the rapidly evolving technological landscape.

Sustainability

Funding

Silver Eagles would not be where it is now without the generous donors who sponsor our robotics teams. These sponsors are our school, **Salesian High School**, and its network of alumni donors: the **Boeing Company**, the **DEKA Foundation**, **Salesian Missions**, and **New Rochelle School District** (Title IV program). We also appreciate our brother school, Don Bosco Prep, in Ramsey, NJ, for helping us get started two years ago. Thanks to their tremendous support in training and providing us with much-needed supplies, we were able to compete in our very first competitions as our own team. These sponsors provided us with many expensive tools and equipment, including the GoBilda and Rev robotics starter kits and a new practice field every year specifically for the new game mode.

Mentorship

Mentorship is a very important factor in the teaching process. Each season, we have freshmen or new members pair them up with existing members who already have experience in robotics. What we also do is show them the introductory video and explain to them anything that they have questions about. Alongside, we teach new members skills that they need to work on the robot, depending on their interests. We like to be as welcoming as possible to new members at all times; this is an essential aspect of our team.

Recruitment

As a eam it is important to attract and keep in members of Robotics at our school. We take every opportunity possible to show ourselves off and have new people consider joining. We start as soon as possible at the Salesian Open House; here, the students take a tour of the building and get to see what our school and community are like and incoming students get to see the robotics room while we explain about the team. We also give people the opportunity to ask questions and even drive the robot. This not only makes kids want to join robotics, but also Salesian as a whole.

Our school also features an activity fair around October in which all students go to the gym and each club has their own dedicated sections. The robotics team has a section in which we give a brief explanation of robotics and demo our robot and let students control it. We also make sure to give new members a warm welcome to the team and make sure to help them if they're confused and make the best impression of ourselves as possible so that people can stay and enjoy their time

Mentors

We are honored to have two volunteer alumni and our own Computer Science teacher as mentors to our team this year. Our mentors have been incredibly instrumental to our own development. Mr. Lanza has a background in mechanical engineering, Mr. Walsh in design and programming, and Ms. Gomez-Boterro has a background in Computer Science. They **oversee our daily meetings** and **hold post-school lessons** sometimes where they teach us about various topics, such as programming (introductory and robotics-based), graphic design, and robot design.

We also have senior-year members that can volunteer to become youth mentors. Their primary roles are to share the robotics and engineering knowledge they've gained over the last few years. Our youth mentors, **Andres** and **Walter**, are planning on returning as mentors after they graduate.

Finanaces

Assets

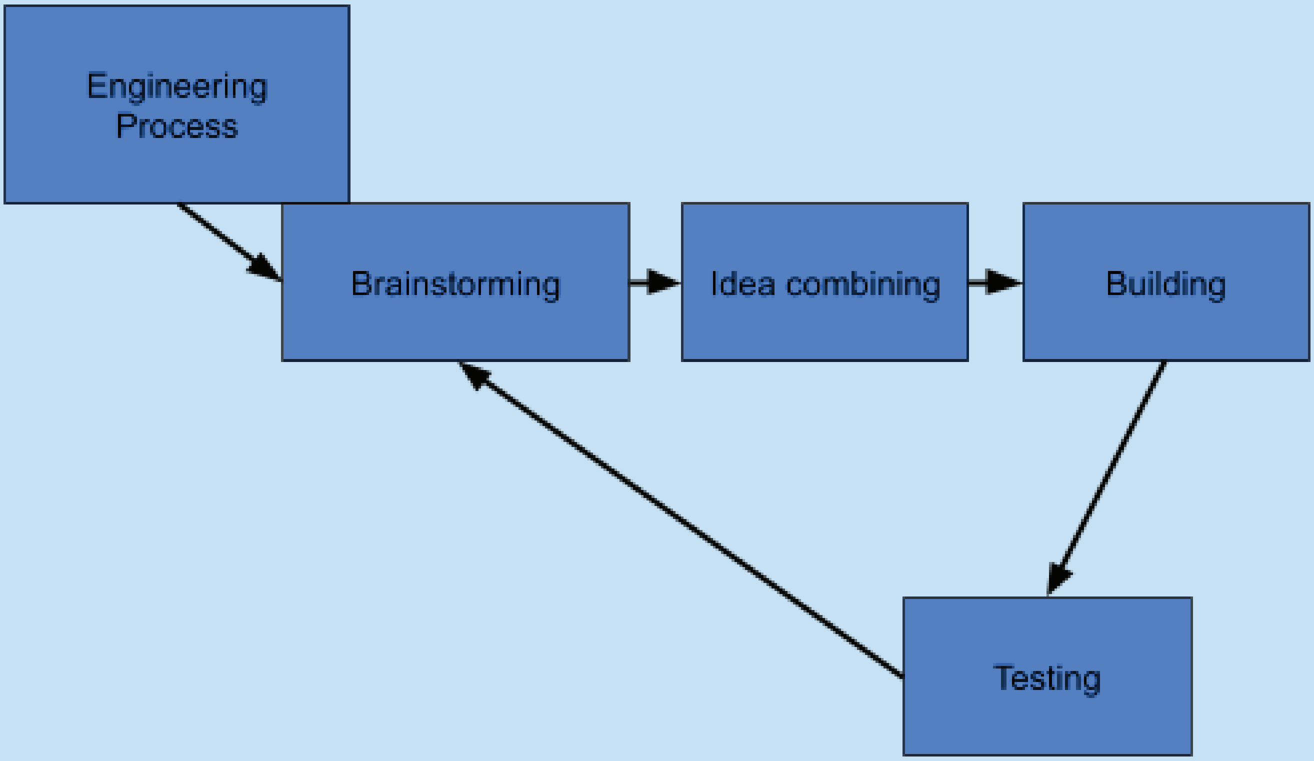
Source	Amount
Boeing	+ $1, 700
Salesian Missions	+ ~$1,500
New Rochelle School District	+ $4,377
School Activity Fee	+ $2,400
DEKA Foundation	+ ~$2,000
Total	**~$11,977**

Liabilities

Source	Amount
Participation	- ~$975
Travel	- ~$300
Robot Parts	- $2525.17
Total	

Remaining Balance: ~$8,176.83

Our Process



Our engineering process consists of much brainstorming, building, testing, and iterating. Every team member has the chance to pitch their ideas, and through collaborative discussion we combine aspects of everyone's contributions into our comprehensive game plan.

We keep speed and efficiency in mind at every step of the build process, but our primary focus is **communication amongst team members**. Since our team is able to have a dedicated space in our school – our robotics workshop, we primarily have all the subteams (design, building, and programming) all in the same space at once. This makes it easy to collaborate all together. When we're not meeting in person or brainstorming from home, we utilize Google Meet for calls, and Google Docs and Classroom to collaborate.

Our build team has the most members out of our subteams. This enables us to plan multiple robot features at once. We split up our build team into groups, each focusing on an aspect of the robot. To ensure our work can come together nicely, our building and desing leaders, **Jorge** and **Matthew**, oversee each of the groups and enable cross-group communication.

Our Strategy

Autonomous Phase

Our initial way of scoring points is to have our robot park during the autonomous stage. We created 4 programs, one for every starting position on the field, that when active, move our robot next to the appropriate scoring board to park.

Tele-Op Phase

Our primary way we plan on scoring points is to place pixels on the board. We have built our robot to be able to actively intake pixels. As we drive, we can drive over pixels to intake them and when we get to the board, our claw can pick up the two pixels at once and place them.

We designed our robot to have good synergies with other teams' strategies:

- ◇ If our ally's robot is one that primarily pushes pixels next to the scoring board, we can have our robot stay around the board and score pixels that our ally brings to us.
- ◇ Our robot can also work with other robots who's strategy is primarily score. We can score alongside them, and when they are at the scoring board.

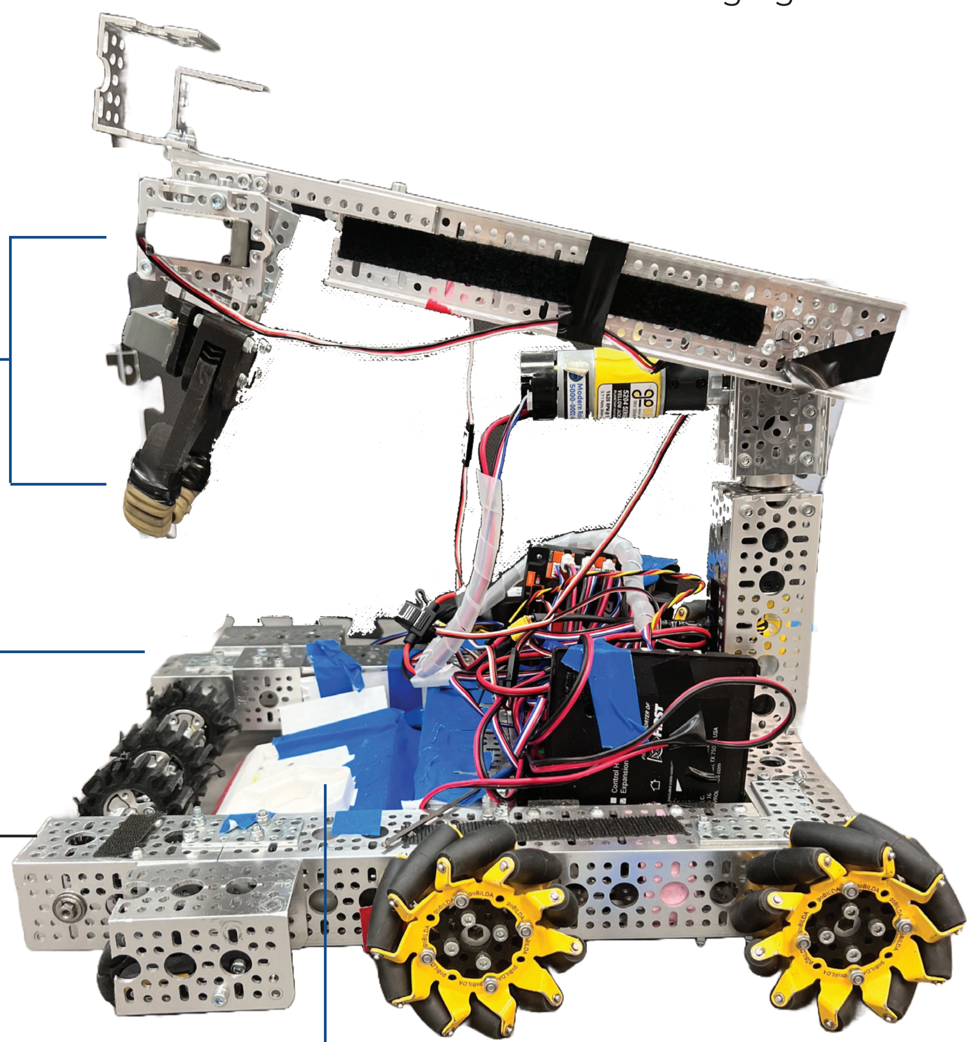
End Game

Our final way of scoring points is with our paper airplane launcher. We have a 3D-printed airplane launcher attached to the base of our drive train, next to the base of the turret motor. Using a goBILDA structure piece, we propped the airplane launcher up at an appropriate angle. Our team came up with a repeatable paper airplane design, so we can have more consistency. Towards the end of the driving phase, we will launch the paper airplane from the back half of the field.

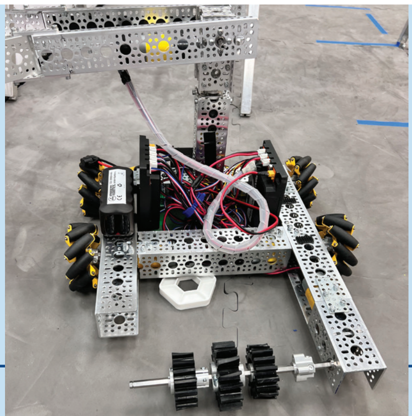
The Future

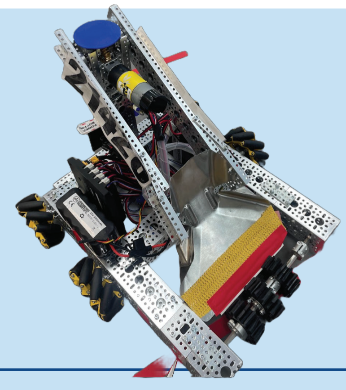
We also plan to improve our autonomous functionality so we can place pixels as well as park. We also are planning to make our out-take system to be more fine-tuned so it can pick the up the pixels on the intake plate faster

Robot Overview



Previous Iterations:



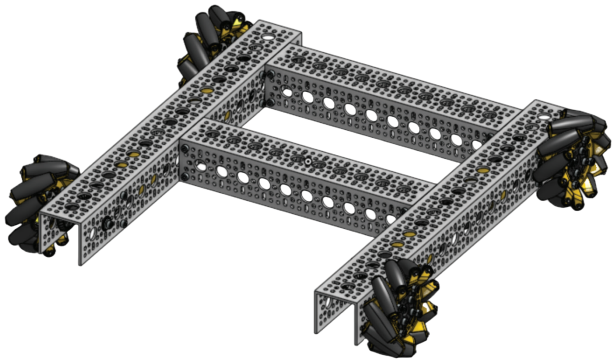


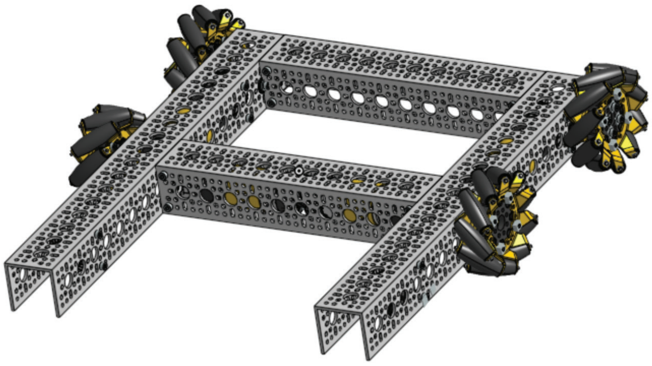
Build Process: Drive Train

Goal: Make a drive train that's light, and able to move and maneuver quickly

Version 1

The first version of the drivetrain was designed and assembled quickly in order to have a full robot to drive, test, and iterate on. It has a U-channel in the middle, where we attached the turret-claw assembly.



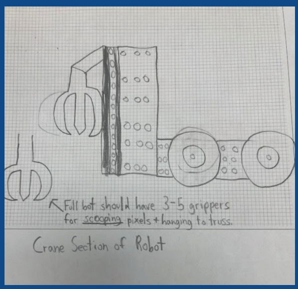


Version 2

After assembling and testing the robot on the home field and at the John Jay qualifier, we redesigned almost all of the robot. On the drivetrain in particular, we moved the front wheels to the middle, making room for another motor for an intake spinner. We also moved the turret-claw assembly to the back of the robot, allowing more room for our intake's redesign.

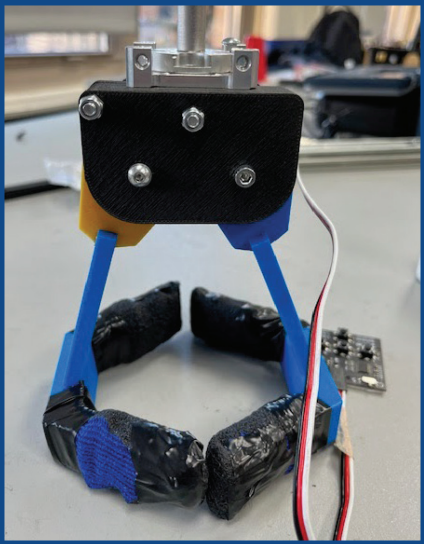
Previous Ideas

Build Process: Pixel Intake + Out Take



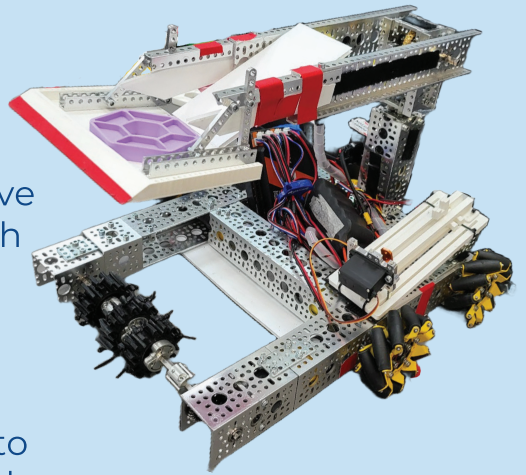
Version 1

Our first design for intake (and outake) design was intaking passively. We had a 3D-printed claw, with two extensions with grips on them. The claw was designed to open, via a servo, and wrap around a pixel. Then can be rotated backwards, with a second servo and the claw can open again for pixel depositing.



We chose not to go with this robot design for a few reasons. The primary reason was that we were only capable of picking up on pixel at a time, and it required a lot of precision. The next major reason was because the robot becomes inefficient when placing pixels on the board.

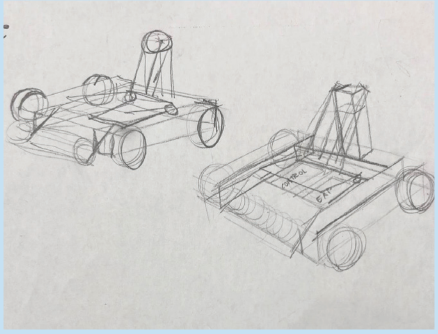
This design averaged ~2 pixels placed per match.



Version 2

For our second iteration, we had decided to go with an active intake as opposed to a passive one. We had not worked with active intakes before so it seemed like a challenge, but we pushed on. We started by drawing up some ideas, got to prototyping, and eventually made a proof of concept.

We designed a rod with gekko wheels, powered by motor, to intake pixels onto a 3D-printed plate. The plate is connected to a turret motor and a worm-gear motor, creating an “arm” that is capable of vertical and horizontal motion. When the pixel is on the plate, our drivers will drive to our team’s scoring board, rotate the robot, and have our robot’s arm rotate completely backwards. This lifting of the arm would then cause our pixel to drop backwards. To accurately deposit the pixel onto the board, we added a ramp, acting as a guiding slide, to minimize unwanted falling pixel movement.



This design averaged ~5 pixels placed per match.

Current Design

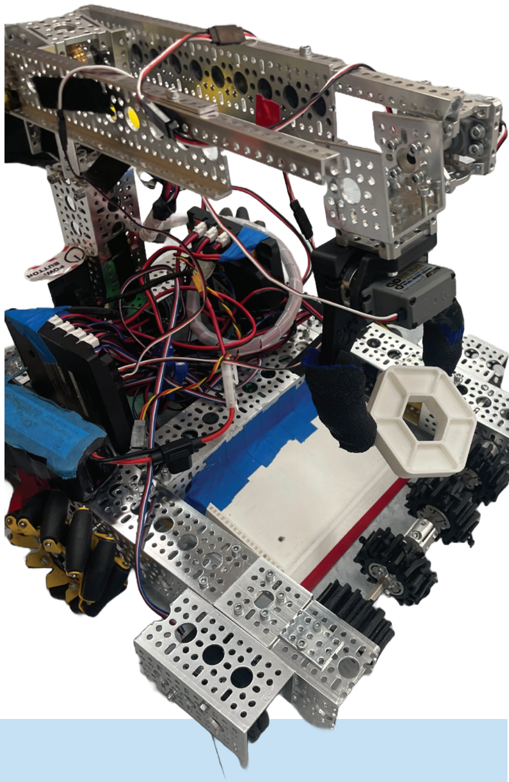
Build Process: Pixel Intake + Out Take

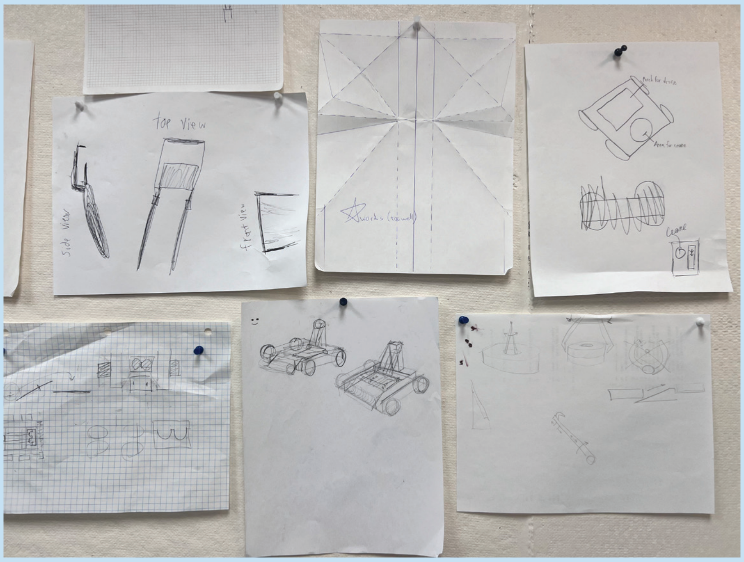
Goal: Develop a system capable of picking up and placing two pixels

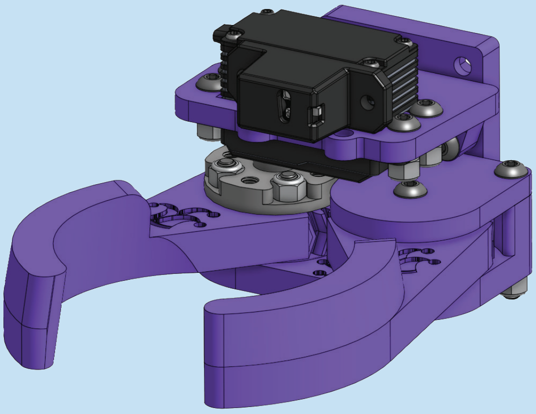
Our latest intake design is the amalgamation of our previous two iterations. We combined the idea of a passive-intake claw, alongside an active-intake, to gain the benefits of both.

Our latest design persists an intake rod, with gekko wheels that spin inward and receive pixels onto a plate. Instead of this plate because directly attached to the arm of our robot, our arm now features a claw attached to a wrist.

Our claw has enough grip to pick up two pixels at once by their center holes. When we pick up pixels, we can lift our arm up, rotate our wrist back, and open the claw to deposit pixels on our alliance board.







Build Process: Drone Launcher Overview

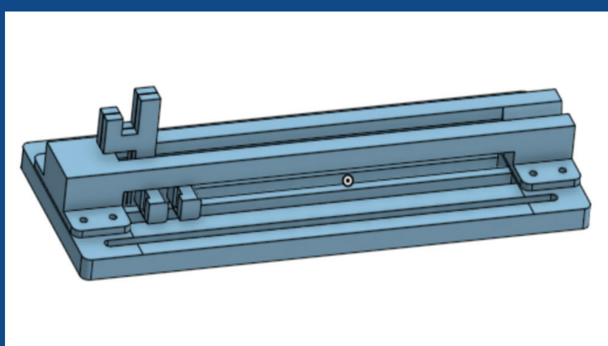
The paper airplane launcher proved to be a problem at first for our team, but eventually we landed on a design that worked for our robot.

Version 1

Our first design was composed of two Gobilda Rhino wheels, each attached to a motor. The components were then attached to Gobilda U-Channels: one for structure that will be attach to the robot and another as an "airstrip."

We found this design was **too bulky and would have inconstitent results**. Sometimes the plane would go far, other times it would get bent by the wheels when they spun.

Version 2



Our second iteration was **much more effective**. We found a design that **Team 503 Frog Force** kindly made public for other teams to use.

We 3D printed the design, and assembled accordingly. We mounted the plane onto the "arm" of our robot. Resistance is built with rubberbands pulling back a tab, held by a servo that can release the tab and fly the plane.

Build Process: Drone Launcher

After finding a suitable drone launcher design, our next challenge was to find a good mounting point on our robot. We experimented with putting it on the arm of our robot, but that seemed to not be stable because the movement of the arm would unload the plane easily.

We found a good place to mount it on our drive train, next to our robot's turret. This proved to be mostly reliable. We ran into two issues:

1. When launching the drone, it would not always fly over the truss and wouldn't always clear the field. This was because had not mounted our launcher at at angle.

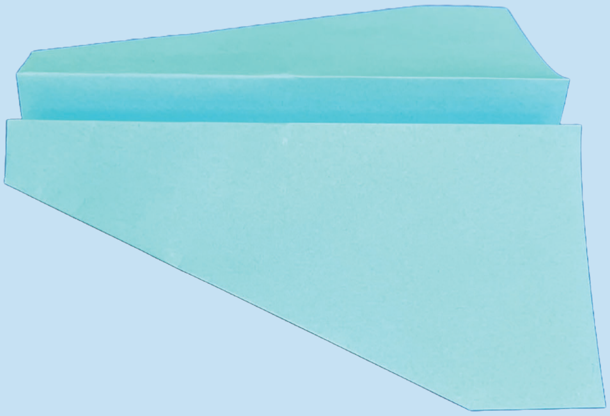
   We solved this by mounting the font of the launcher on a small U-Channel bracket. We found a good angle and was able to solve this issue quickly.

2. Our second issue was that we'd still experience the plane falling out of the launcher from time-to-time while making fast movements when driving.

   After investigation, we were able to discern that there wasn't enough grip in the launcher to hold the drone in place. We fixed this by adding masking tape along the inside of the drone.

Drone Shape

Another aspect of our drone launcher that we spent a lot of time on, was the actual drone itself. We workshopped a lot of different drone designs, keeping track of how size and folding patterns affected flight.



Autonomous Driving

First Iteration

For our initial implementation, the programmers wanted to try using a machine learning model to recognize pixels and our team prop. Using the FTC Machine Learning (ML) platform, we trained a TensorFlow model to recognize pixels. This was our team's first time using ML, so results varied. Our model would only sometimes recognize a pixel. We pressed forward on training the model further until it was accurate to a point we were pleased with. We also trained the same model to recognize our team prop. Despite our progress in learning more about ML, we ran into a pitfall where we weren't quite sure on how to get our robot to drive the places where we needed it to in order to place a pixel and park during the autonomous stage.

After our first competition this season, we actually learned how to solve this issue from another team, The SharkBots. They were very kind and explained to us some of their autonomous practices and implementation. From them, we learned that using odometry wheels would allow us to program our robot to effectively determine its location on the field and keep track of where it drives. This, in turn, allows our robot to move more accurately and confidently during the autonomous phase.

Second Iteration

For our second iteration, we were hoping to combine our knowledge gained from our first attempt with our ML training and the new information we learned from SharkBots. However, we decided that in order to give our team more time to fully flesh out other features on the robot, the simplest approach for autonomous implementation would be best for now. This approach would be to effectively provide hard instructions for the robot to follow, with us programming it to follow a series of steps, moving a specified number of tiles at a time.

We were able to calculate the math to have our robot drive one field tile at a time by displaying telemetry data to our Driver Hub, using the encoder capability in our wheels' motors, and with trial and error. This was calculated by comparing how far our drive train went (when set a certain speed) after a certain number of seconds. This then allowed us to code the capability to drive forward by a certain number of inches.

We have 4 autonomous programs, 1 for each starting phase on the competition field. Each program tells our program to attempt to push a pixel on one of our tape markers, then drive towards the board to park.

Our next iteration of our autonomous phase, we are hoping to implement odometry to more accurately control our robot during. We also wish to reimplement our camera, so we can appropriately determine which tape marker to place the pixel on; it would be ideal, too, if we can implement the ML model we trained from our first attempt. We're looking forward to our next iterations.