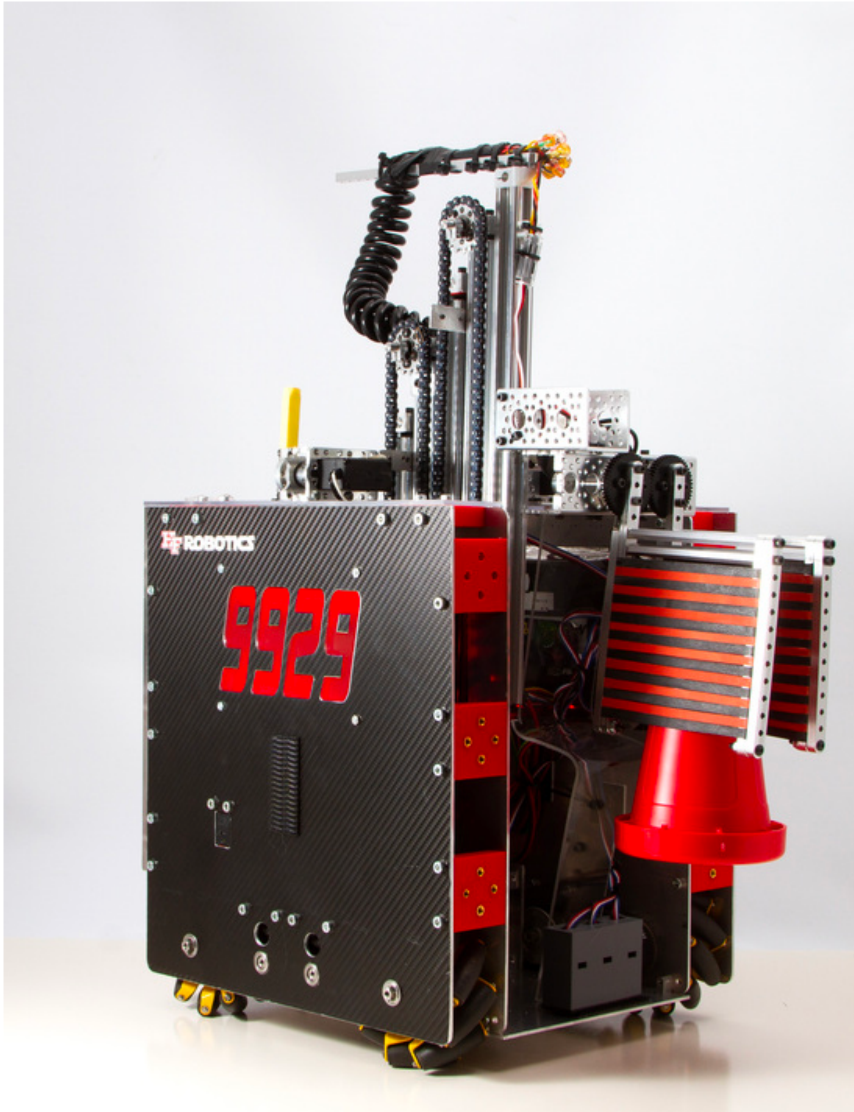


FTC#9929
Tech Ninja Team
Power Play
Engineering Portfolio
2022-2023

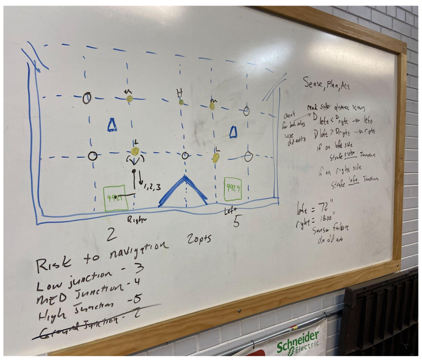
Team Overview

Hello! We are Tech Ninja Team (TNT) FTC#9929, a community team from Homewood-Flossmoor, Illinois.

We work out of an old funeral home which is now the Homewood Science Center. Our workshop is in the garage where they used to park the hearses!

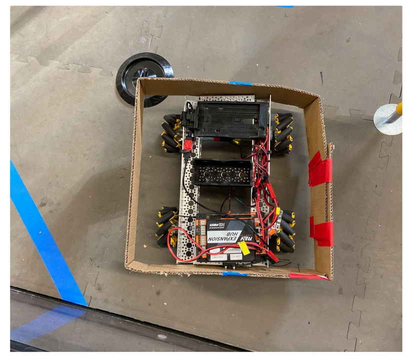
Four years ago, we purchased a CNC Router and started manufacturing our own parts out of it. We also use a 3D printer to print additional parts that are useful for our robot.

Our robot is a great representation of our team because it is quick, small, and compact, like a ninja.

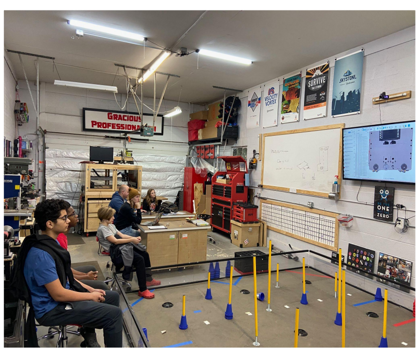


Autonomous risk to navigation and measurements

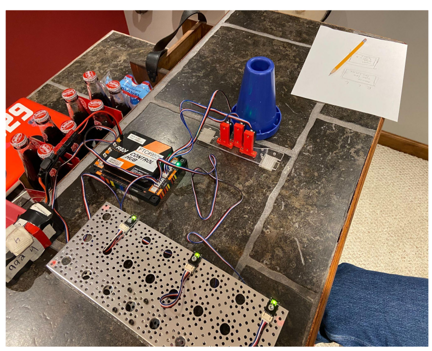
Preparations for match



First generation robot



Working on CAD



Prototype Cone Location Sensors

Team at a Competition

Strategy

During kick-off weekend we watched the reveal video and learned the rules of the game so that we could come up with a strategy for playing the game. We determined that our robot must:

- Be fast - the robot has to cover the entire field and score quickly, this requires a fast drive base as well as sub-second intake and release, and a 1-2 second lift.

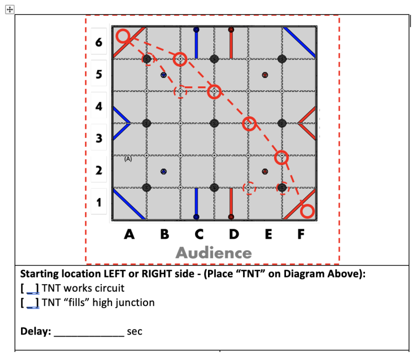
- Be maneuverable - the robot has to be small (12 inch square footprint) to navigate obstacles on the field.
- Lift over 33” - to reach high junction
- Be robust - we cannot score points with a broken robot
- Be easily serviceable - to be able to fix the robot quickly between matches

We also determined that game practice and strategy would play a significant role in our team’s success this year.

We developed a board game to better understand the game when our robot was not drivable.

We developed Match Planning Sheets to better communicate with our alliance partners.

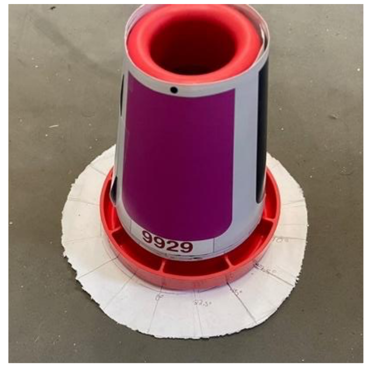
During our first and second league meets, our drive team refined their strategies on how to score the most points. We used this experience to develop a match planning sheet that would allow us to quickly plan a match with our alliance partners:



Our beacon fits on a cone, allowing us to score both a cone and the beacon at the same time. We have gone through a few beacon designs throughout the season:

For autonomous we experimented with AprilTags and OpenCV color detection. The AprilTags proved to be too sensitive to slight misalignments so we selected OpenCV color detection. We printed our own sleeves, and use colors that are “far” from colors on the field.

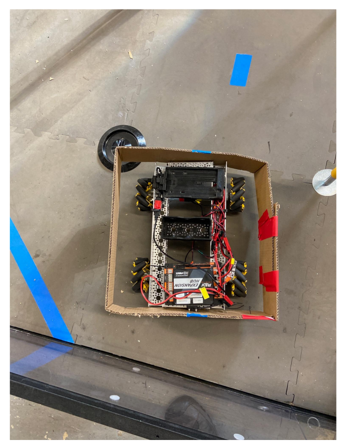
The AprilTags Sleeve proved too sensitive to slight misalignments.



OpenCV color detection works reliably, using colors that are mathematically “far” from those naturally occurring on the field.

With the signal sleeves above, we quickly programmed an autonomous that would park in the correct signal zone according to a custom sleeve we designed.

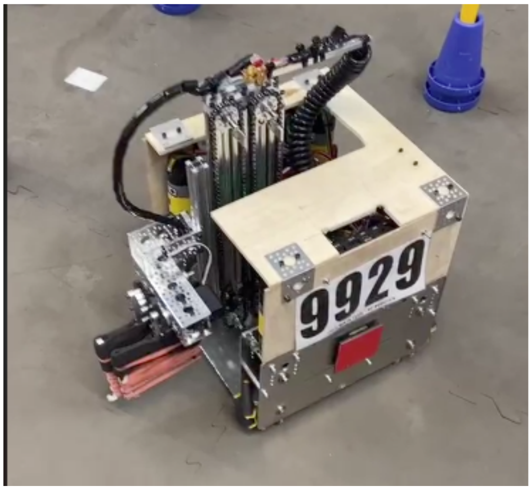
Later in the season, we programmed the robot to score on the low goal during autonomous. At this point in the season, our team reliably scores 20 points in autonomous, and 70 to 80 points in tele-op.



Robot Design: Drive Base

We immediately thought our robot needed to be smaller than normal this year to better navigate between junctions. But we still wanted our robot to be fast, so that it could complete circuits on a busy field. We tested our initial impressions by driving last year’s robot (a full 18” by 18” square) on the field, and then driving a smaller “push bot” we had on the field. Our small, very fast ‘push bot’ navigated between junctions better. We used cardboard to evaluate robot footprint sizes.

Based on this testing, we decided to build a robot with a footprint of 10 to 12 inches.

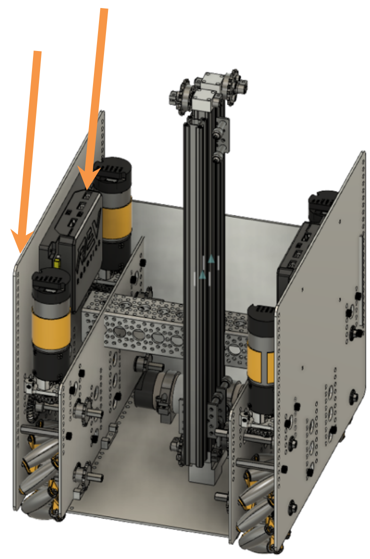


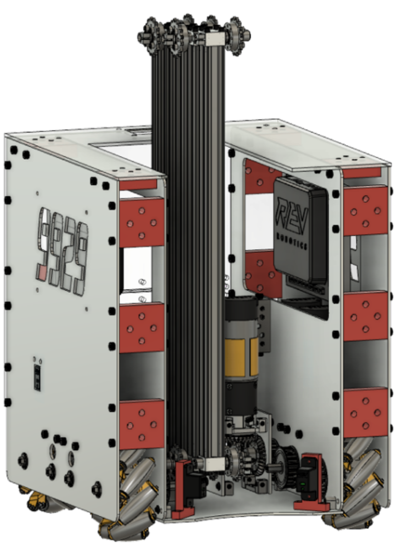
This is “version 1” of our robot as it appeared at our first league meet. It had a footprint of around 11” square.

As you can see, while we used CAD/CAM to cut some of the parts, we used plywood to prototype other parts as the design evolved.

This is “Version 1.5” of our robot as it existed in CAD.

Note the motors mounted vertically above the wheels. As we developed the robot, we learned that this resulted in too high a center of gravity – the robot had a tendency to want to tip over, especially under high acceleration or deceleration. As a result, we abandoned this design.

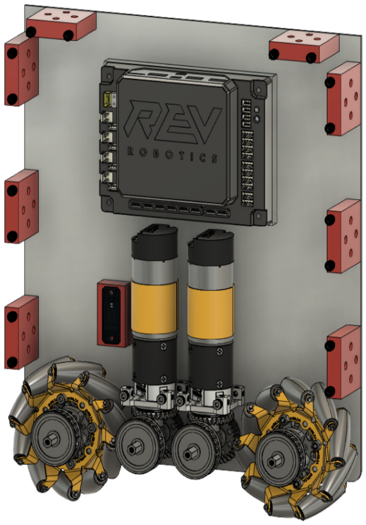




This is “Version 2” of our robot in CAD.

Note the motors are no longer above the wheels.

In order to lower the center of gravity, we squeezed the motors down in between the wheels. The image below illustrate the compact drive motor arrangement:



A final note about our drive base design: *speed*. By keeping our robot footprint under 12", and by keeping the center of gravity low, we were able to keep high speed drive motors (13:1 gear ratio) while still having a very drivable robot.

Robot Design: Gripper

An effective gripper is necessary to score more than a few points in this year's game. We determined that our gripper needed to be *fast* – that means more than just closing quickly. It means it needs to be *easy* for the drivers to align the robot with the cone. We also needed the gripper to be *light*, and to use a limited number of wires, because it would be at the top of the lift.

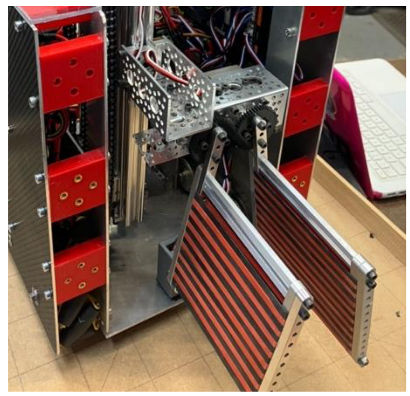
We initially explored ideas for a compliant-wheel intake that would spin to suck in cones and spin the opposite way to spit the cones out. As we began to encounter problems with that concept, we started to develop a basic "rubber band gripper" in parallel with the compliant-wheel intake. Ultimately we abandoned the compliant-wheel idea because it was simply too bulky and tuning it to be reliable seemed like it would take too much time.

We also thought of a rubber band gripper that would come down on the cones and pick them up from the sides.

Our early gripper design was very much a “prototype” which happened to work well.

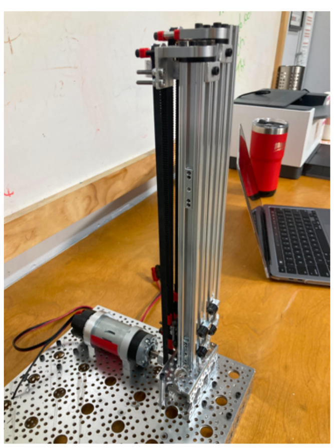
As the season progressed we experimented with different kinds of padding. We also experimented with methods for keeping the rubber bands in place. But after 2 league meets, our prototype was a bit of a mess.

Our next gripper design is more refined. We sandwiched the rubber bands between beams to keep them from moving.



Robot Design: Lift

We determined our lift needed to have over 33” of vertical travel. It also needed to be fast – our initial target was for it to fully extend in less than 2 seconds. Finally, it had to be robust and reliable, able to cycle hundreds of times without adjustment or repair.

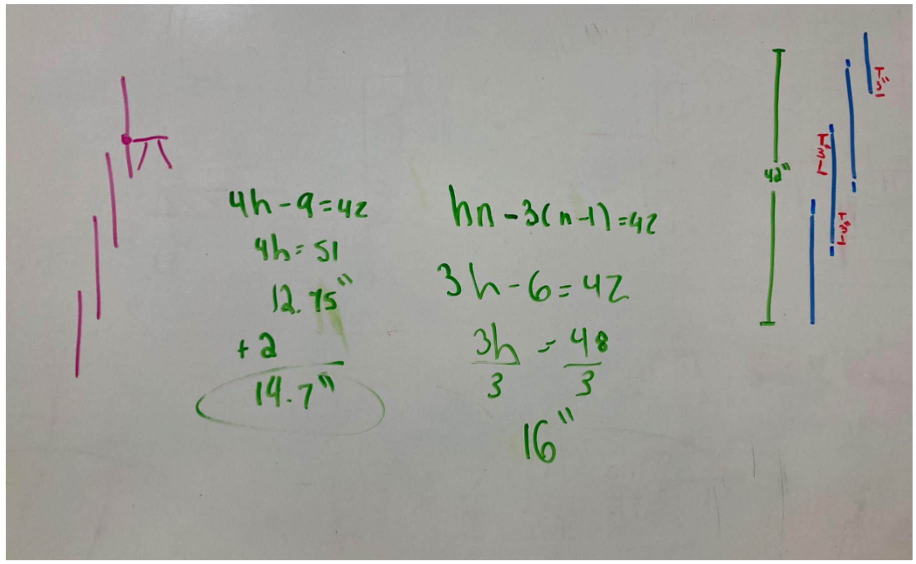


First, we mocked up a belt-driven linear slide using parts from previous seasons.

We used this to test which motor would give us the most speed, while still having enough power.

We determined that we wanted to use chain rather than the belt, as the chain is more robust than the belt. While our belt design is lightweight, the belt has a tendency to slip.

Having selected a chain design, we needed to calculate how many stages we needed to reach 33” of lift:



Using lifts from prior seasons, we derived an equation to determine the height of each stage, the number of stages, and the height of the retracted lift. We lose about 3” per stage, except for the last stage.

We needed 3 to 4 stages to reach a desired height of 42”, so we plugged 3 and 4 into the equation above for “n.” The height of each stage would be 16” for 3 stages and 14.7” for 4 stages. We decided to use 4 stages to allow room for the drive base and other components, and to lower the center of gravity of the robot. Our final lift stage is a shorter “carrier” stage that allows us to attach the gripper mechanism to the lift.

Fabricate

We first cut the channel and tapped the ends to accommodate the hardware necessary to hold the sprockets that ran the chain. We then assembled the sprockets, bearings, and cut chain to the appropriate length to limit the excess chain. We then added a second sprocket to the lowest stage and mounted a motor horizontally to the bottom of the robot in order to run the lift. We used a small length of chain to connect the motor and the lift. It was of utmost importance that throughout this w hole process, even the smallest details were accounted for because every component needed to be fabricated in a certain way or the lift would not be as efficient as it could be or not work at all.

Once we got everything working on the physical lift, we needed to address the constraint that we wanted the lift to be automated. In order to achieve this we added a magnetic limit switch to the side plate. This would allow the robot to know when the lift was at its lowest position, because the magnets attached to the lift would line up with the limit switch. We also used encoder counts to determine where the lift should stop at each junction. We tested and recorded the encoder counts when we manually drove the lift to the desired heights and then implemented those into the code and assigned various buttons on the operator’s controller to each height. The operator can still manually drive the lift which is immensely helpful to pick up cones in a stack, but the automated heights reduce cycle times and allow our robot to be more reliable. The gripper also automatically closes as the lift retracts to avoid hitting the side of our robot.

Implement

We bolted the motor and the lift to the bottom of the drive base. We also used the beam in the back of the robot to fasten the lift. This iteration worked fairly well, but had some major issues we needed to fix.

First and foremost, while the carrier stage allowed us to have a slightly shorter resting height, the weight of the gripper pulled it forward and put strain on the rollers as there was nothing holding the stage at the top. This made the stage unstable and a possible risk for failure. We decided to make the carrier stage the full height to combat this issue. While it made our resting height a little higher and added height to the extended lift that we did not necessarily need, it eliminated the issues of the short carrier stage.

In our second design of the drive base, there was no longer room to orient the motor horizontally so we needed to alter the lift to accommodate a vertical motor. We had to switch the sides that the chain ran on and create a bevel gear system to drive the lift, similar to how our drive base is run. Making this change also allowed us to correct the issue from the first iteration. The chain that connected the motor to the lift was too long and the slack caused many issues and hurt the efficiency of the mechanism. When we were forced to switch the motor orientation, we made sure that the motor ran the lift directly rather than through a chain.

Another small change we made was replacing the attachment links that fastened the chain to the channel. The ones we used previously were the wrong orientation and there were no threads for the screws to hang onto. We used links with a different orientation that allowed the screws to go in a different place and made the lift more secure.

One of the most recent changes we made was replacing a motor with a 19:2:1 gear ratio with a motor with a 26:1 gear ratio. While the 19:2:1 motor offered more speed, it did not have enough torque to consistently raise the lift to the maximum height required. This motor was also using an insane amount of voltage just to raise the lift. Once we replaced the motor we found that not much speed was lost and the lift was able to be raised with much more consistency and less struggle.

Robot Design: Other Features

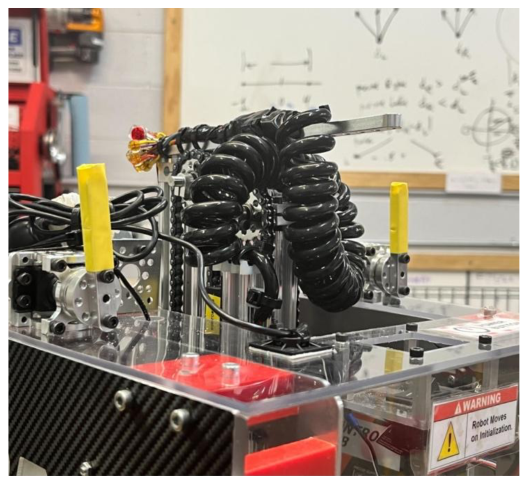
There are a couple of other features we are proud of – our edge-lit team numbers and our “pointers.”

The team numbers have a polycarbonate sheet behind them. That sheet is edge-lit by LED strips.

Because “red” is part of our team colors, we wanted a way to make it easy for referees and the audience to see what alliance our robot was on. The edge-lit LEDs allow us to indicate whether we are the red or blue alliance. They also signal the drivers that EndGame has started by turning Green.

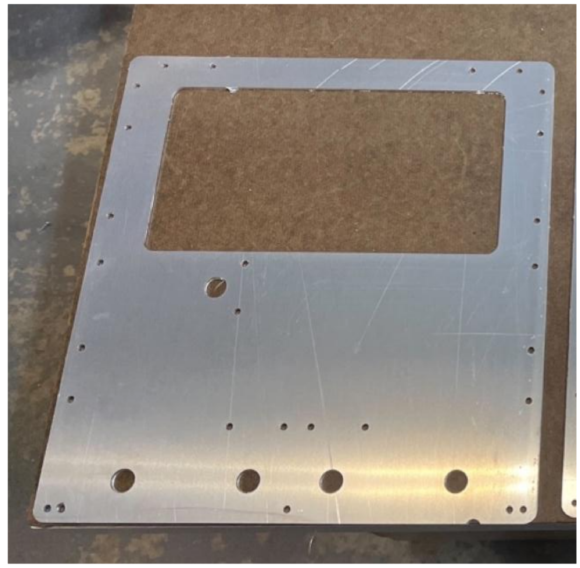
We also think the lights contrast well with the carbon-fiber wrap we used on the sides of the robot.

The “pointers” at the top of the robot serve an important function. They help the drivers “see” whether the robot is in position to drop a cone. These pointers use distance sensors mounted on the front of the robot to determine whether the robot is centered on a cone or junction.

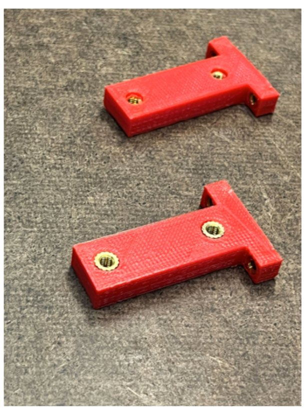


Fabrication

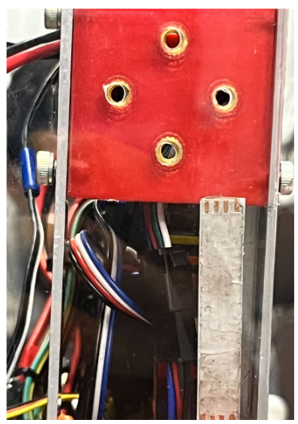
We fabricate many of our own parts. For example, we designed and cut the inner and outer “side plates” pictured below on a Shapeoko CNC router. We made all of the aluminum and polycarbonate plates on our robot ourselves.



While we have had a 3D printer for several years, this year we began exploring how to use heat-set inserts in our 3D-printed parts. These allow us to obtain strong threaded holes in 3D-printed parts. This has proven to be an especially effective solution for places we would have otherwise used standoffs. Our 3D-printed parts include the side-plate spacers, sensor mounts, and our battery holder.



3D-Printed sensor mounts allow us to quickly mount sensors, and the heat-set inserts help keep the mounts simple.



3D-Printed spacers with heat-set inserts allow us to create a strong structure while avoiding the need to buy standoffs or costly parts.

While 3D-Printed battery holders are common, the heat-set inserts allowed us to put this one on the back of our hinged "door."

Signal Sleeve Detection

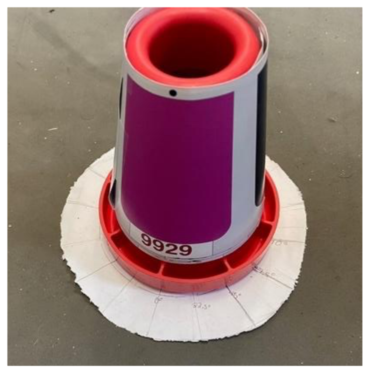
We use OpenCV to detect specific colors on our signal sleeve by eliminating all other colors in the scene. We taught it all three colors that corresponded to our signal sleeve and then determined which direction that the robot needed to move.

1. Robot filters out all colors other than the one it is interested in
2. Finds the color that corresponds with signal (box filtered to certain size, picks larger color area, rejects small amounts of other colors)
3. Parks in required zone

Colors are as far away as possible to colors present on the field and each other

For autonomous we experimented with AprilTags and OpenCV color detection. The AprilTags proved to be too sensitive to slight misalignments so we selected OpenCV color detection. We printed our own sleeves, which use colors that are "far" from colors on the field.

The AprilTags Sleeve proved too sensitive to slight misalignments.



OpenCV color detection works reliably, using colors that are mathematically "far" from those naturally occurring on the field.

With the signal sleeves above, we quickly programmed an autonomous that would park in the correct signal zone according to a custom sleeve we designed.

Later in the season, we programmed the robot to score on the low goal during autonomous. At this point in the season, our team reliably scores 20 points in autonomous, and 70 to 80 points in tele-op.

Automation and Driver Assistance using Sensors

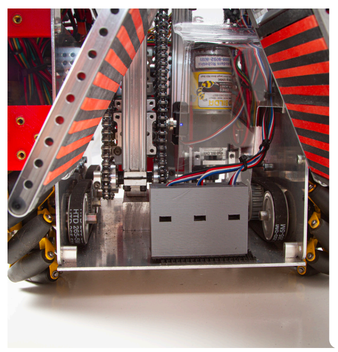
Lift / Scoring Mechanism

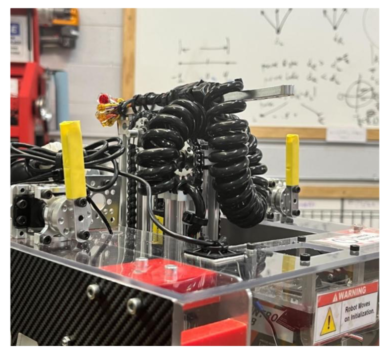
Once we got everything working on the physical lift, we needed to address the constraint that we wanted the lift to be automated. In order to achieve this we added a magnetic limit switch to the side plate. This would allow the robot to know when the lift was at its lowest position, because the magnets attached to the lift would line up with the limit switch. We also used encoder counts to determine where the lift should stop at each junction. We tested and recorded the encoder counts when we manually drove the lift to the desired heights and then implemented those into the code and assigned various buttons on the operators controller to each height. The operator can still manually drive the lift which is immensely helpful to pick up cones in a stack, but the automated heights reduce cycle times and allow our robot to be more reliable.

After our league qualifier, we added a few more adjustments to aid our drivers. We Now when a cone is gripped using the “grip” button the lift raises a few inches to allow the drive team to move away from the location quickly. We also reduced the speed of the robot when the lift is fully extended to reduce tipping.

“Pointers” (Cone and Junction Locators)

The “pointers” at the top of the robot help the drivers “see” whether the robot is in position to drop a cone. These pointers use distance sensors mounted on the front of the robot to determine whether the robot is centered on a cone or junction.





Other Features

We use various features to help our drive team during matches. Our controllers rumble to notify drivers that the endgame portion of the game has started. The LED lights on the side of the robot look cool, but also alert the drive team in a similar fashion to the controls.

Outreach

Connecting with our Community

We regularly work to promote robotics in our local community and to build relationships with local businesses and entrepreneurs.

We march in our community's Fourth of July Parade. This is a great way to reach a large number of community members and increase recognition of FIRST and our team. It was awesome to see people who remembered us from previous years as well as make connections with people who have never seen our team before. This year we also invited a few alums to march with us!

This year we made a connection with a local Flossmoor resident, Keith Janowski, who builds an animatronic “haunted house” every year for Halloween, which he calls the “Devil’s Playground.” Mr. Janowski showed us how he made each of his haunted-house creations: he used everything from old car parts (wiper motors) to garage door openers to pneumatic actuators. He gave us several great ideas and demonstrated using readily available parts to achieve great results!

We also work within our community with the Homewood Science Center, a local STEM center. The Homewood Science Center lets us use their garage as our workshop. In exchange, we attend and volunteer for HSC events.

Connecting with our League

As one of the older teams in our area we always look for ways to support the other teams in our league. We supported our league members by:

- Helping to diagnose robot problems at league meets.
- Making sure we have extra parts at every league event. This year we gave teams:
    - On-the-go cables. We always carry extras because we see teams having problems with worn-out cables every year
    - An expansion hub. With the supply-chain challenges this year, one team came to a league meet in dire need of an expansion hub. We loaned them one of ours.
    - Materials for beacon repair. A team broke their beacon at a league meet. We helped them repair it.
    - Power switch stickers. Again, we like to be prepared. When a team looked like it might fail inspection, we provided them with the stickers they needed.
- We hosted the 3rd league meet at our local Junior High School, Parker Junior High. This allowed us to *both* promote robotics in our community *and* support our league teams by providing a venue.

Sustainability

This season we were fortunate to once again be awarded a grant from Schneider Electric.

We have a lot of costs that Schneider helps cover:

- Power Tools
- Robot parts and spares
- Competition Registrations
- Sanitizing supplies!

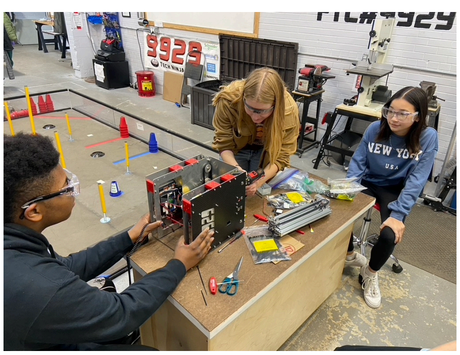
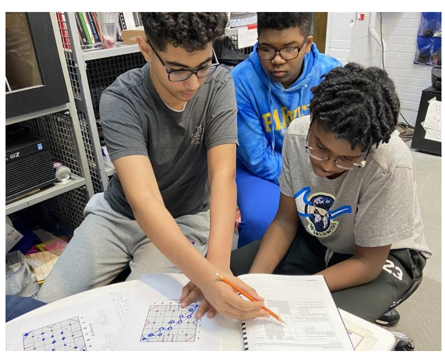
We look forward to our team's continued collaboration with Schneider Electric, and continuing our site visits and demonstrations at their offices and events where we get to talk to their engineers and technicians.

This season we have also been fortunate to receive an Amazon sponsorship. Amazon has donated many things that we are using this season. An example of this is the Dualshock controllers that we are using to control the robot this season. This had given us an advantage because before we weren't given controllers that rumble, but now they help us as the rumbles tell us when it is before the endgame and if we are close to an unseen junction and cones. Another occurrence of this are the tools we had gotten such as the step drill. We use the step drill for drilling holes on the plates on the robot. This Amazon sponsorship has helped us by giving us tools and items we have used throughout the season and will use for the seasons to come.

The Homewood Science Center has been invaluable to us for many seasons. They provide us with space to work and have introduced us to our sponsors. We give back by volunteering at their events, which in turn helps us with STEM outreach.

The Tech Ninja Team tends to recruit new members through our sister FLL teams. This year, the entire team are veterans of the FLL teams that meet next door to our workshop.

We all learned many new skills this season. As rookies we learned to set up and use our team's power and hand tools, programming the robot for auto and tele-op, developing strategies, and even writing engineering portfolio content. As experienced team members, one of us graduating this season to head off to college next fall – we learned to mentor, guide and pass on knowledge and traditions.

		
Using the die grinder	Calibrating the CNC router	Tapping threads
		
Operating the CNC router	Using OpenCV	Pit Management
		
Assembling the drive base	Developing strategies	Writing portfolio content