16633 Bots in Black Engineering Portfolio

Accolades:

- 2019-2020 Winning Alliance
- 2019-2020 3rd Place Motivate Award
- 2019-2020 State Championship Winning Alliance
- 2020-2021 Top Ranked Team
- 2020-2021 1st Place Connect Award

16633

Engineering Section

Engineering Principles and Values

As a team, we most benefit from aiming to build **complex** yet **robust** robots. By focusing on reliability and simplicity, we are able to perform well at FTC competitions while exploring challenging yet rewarding engineering problems. By keeping our design simple yet functional, we are able to positively impact other people, teams, and organizations from exposing the **practicality** and **science** behind our engineering process, which allows ideas to inspire team members, younger students, and others in our community to advance their current engineering skill level, serving the *STEM Pipeline (reference Team Plan, Sustainability, STEM Pipeline section)*

Engineering Design Process

Due to our experience with designing and building in FTC, we have come to utilize the design process called the “**Waterfall Approach**.” In previous years, we used “**Evolutionary Approach**,” where we could think of ideas and quickly build them. If it didn’t meet our needs, we would rebuild it with an improvement. *However*, we still mix in the Evolutionary Approach when needed. *More information on our design evolution is in our Engineering Notebook. Just Ask!*

Our process involves analysing the game at the beginning of the season, then brainstorming potential strategy options, and creating expectations playing the game given our current skill and experience level. After generating ideas of what is key for our robot design, members:

- prototype designs through a variety of means (CAD, drawings, physical experimentation)
- implement our designs, looking for major fallbacks in our design

if verification proves successful, we maintain our design

else, the cycle starts over *(the evolutionary process component)*

Here is a diagram of how the Waterfall Approach works.

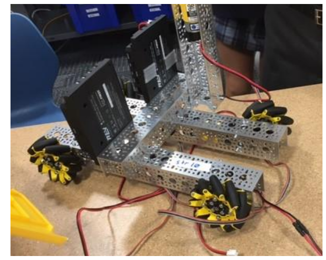
Requirements → Design → Implementation → Verification → Maintenance

Our Robot

Robot Chassis	**Process used: Waterfall**	**Iterations: 2**

V1 Thinned Mecanum Drivetrain

Our first robot drivetrain was a thinned mecanum drivetrain. In the past, we have typically used a drivetrain where there are 4 mecanum wheels, and the frame is sized to fit with small margins into the 18”x18” sizing constraint. However, this year we observed that the major way to score in this game is through freight, which sits in the warehouse. However, to get into the warehouse without going over the barriers, you can go through the 13” gap. Thus, we designed our mecanum drivetrain with a 13” width limit.



In fact, this design inspired most of the highschool Marist teams' design of their chassis, specifically teams *15156, 14101, 14100, 16632,* and *14468*.

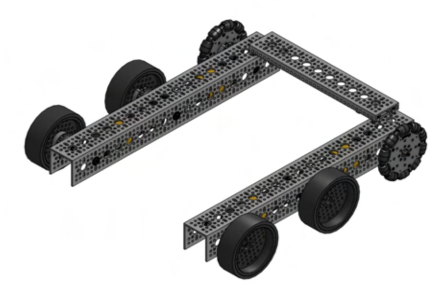
V2 6WD Tank Drive

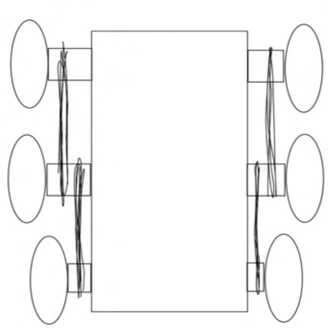
When we saw that this year's game had a barrier, we immediately knew we wanted to be able to travel over it for <u>two reasons</u>.

- We did not want to be <u>limited in how big we could build the robot</u> because of the barrier.
- If a robot were to become disabled in the gap, we could <u>still complete normal game operations</u> by traveling over the barrier.

With all of these requirements in mind, we knew that a six wheeled robot with elevated back wheels would be key, and would increase competitiveness, especially at the league tournament.

Our drivetrain contains four traction wheels, two in the front and two in the middle, and two omni wheels in the back. The omni wheels in the back are raised up by 2mm using the goBilda Drop-center Bearing Plate. In order to have more accurate turning, the front two wheels had to be off of the ground so they would not drag across the playing field as we turn. The front wheels and back wheels are chained driven. Powering all six wheels provides us with a lot of torque making it easy to go over the barrier.





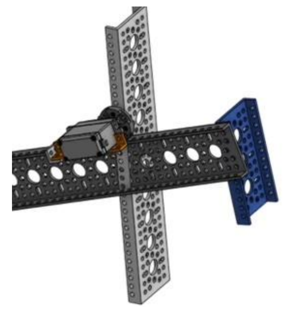
Intake	**Process used: Mix**	**Iterations: 4**

V1 Grasping Claw on Arm (Rotating)

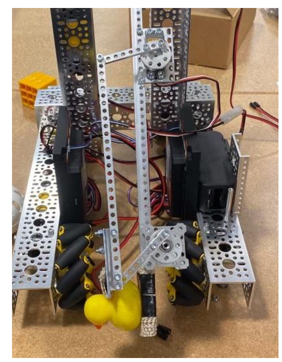
When our team began to look at the game, our first order of business would be a way to pick up freight and put it on either one of the hubs (a strategy decision made separately). A common approach to this would be a claw that can grasp the freight, which are all similar to some extent in size and material (except the duck, which would still be graspable by the claw due to the range of motion).

Our first design decision was how we were going to make the claw. We could either:

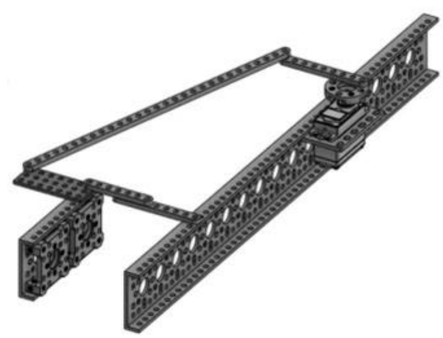
a. Use two rotating pieces powered by two servos that synchronously grasp the freight from the sides. This allows for slightly more holding power but adds points of mechanical failure (two servos)
b. Use one fixed piece and one rotating piece. The rotating piece pushes the freight into the fixed piece, adding pressure to the freight so that it does not fall out. This decreases how much pressure we can apply to the freight but increases the durability of the design



V2 Grasping Claw on Arm (Linkage)

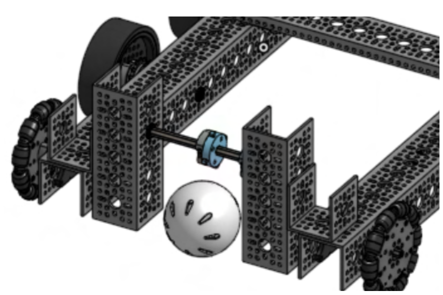


With the rotating design of the V1 claw, we found that the rotating flat piece would contact the freight at different places (due to different sizes and shapes), and only a part of the rotating piece comes into contact with the freight. We realized that we need the moving piece to rather not rotate, but move horizontally. There are a couple ways to do this that we thought of, mainly either using: <u>a rack and pinion gear system</u>, where a gear (powered by an actuator) moves along a flat gear, instead of another circular gear, causing it to move sideways, or a <u>linkage</u> where a lever (in our case a GoBilda beam) is powered by a rotating servo to cause another lever to move sideways, thus a sideways motion. *We chose the linkage* because the circular gear on the rack and pinion tends to come off the track of the flat gear, breaking the whole system.



When building the linkage, we learned a valuable lesson to always understand the science behind what you are building before constructing. When we built the linkage, we did not know which lever should have a rotating point and which one should be fixed. We should have learned how they worked beforehand in order to save a lot of time.

V3 Roller Design



In order to pick up freight, we strategized ways for the robot to pick up freight. From experience in previous years and our previous intake version, an active intake would be the fastest and easiest to control. We figured that, even with improvement, a claw would take time to position accurately to pick up freight.

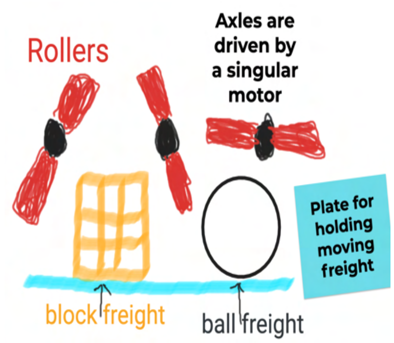
We first had a singular axle mounted at the front of the robot, which would feed into an outtake bucket *(not pictured in CAD to the right)*.
For the rollers,

- We first tried polyurethane tubing, which was a little too stiff and small. The freight would move around the rollers because of the stiffness.
- We then tried rubber hosing, which gave us the perfect balance of strength and flexibility, while also being cheap.

However, this design would not be long enough to pass through to our outtake.

V4 Widened Roller & Plate Combo
Our next iteration necessitated a longer passthrough to our outtake bucket, which would be at the opposite end of the robot. In order to accomplish this however, we would need a ramp on the bottom that would allow for the freight to travel on instead of the floor into the outtake bucket. This would also require more rollers to move the freight.

- We first created a *CAD model* of an intake, where there would be a plate on the bottom and 2 or 3 axles above the plate.
- Throughout many reiterations, we found a spacing between the rollers attached to our axles and the plate, about the size of a ball freight.
- We tested cardboard, a 3d printed plate, and finally chose **corrugated plastic**, which is <u>flexible</u> but <u>robust</u>

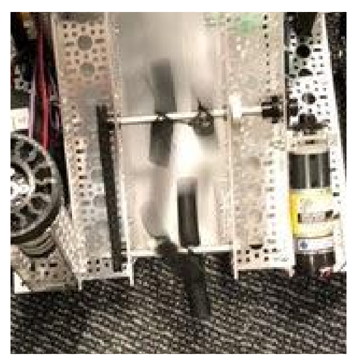


Advantages

- Super fast and no driver finesse required to pick up blocks.
- Picks up all shapes of blocks.
- Can carry blocks over the barrier

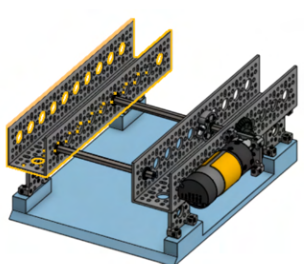
Disadvantages

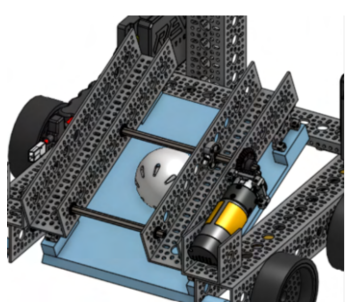
- Can not pick up team scoring element
- Could possibly pickup more than one element at a time
- Requires a bigger chassis.



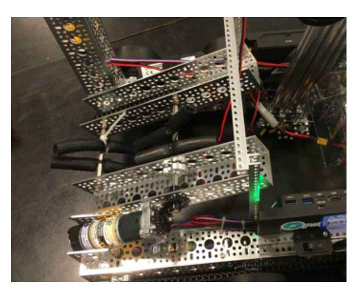
Intake Fully Implemented

In order to drive the intake, we use a 312 RPM GoBilda motor, geared slightly down in order to give us additional speed. In addition, we geared the motor to the shafts using bevel gears, allowing the motor to be perpendicular to the axle shafts instead of inline.





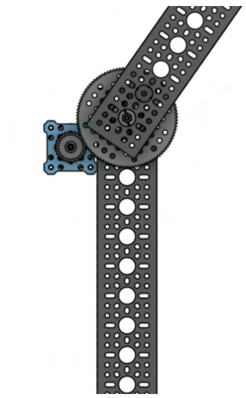
Intake Without Rollers



Outtake	**Process used: Evolutionary**	**Iterations: 3**

V1 Arm & Claw *(see Intake V1 & V2)* **Combo**

For our outtake we wanted something that could easily deposit cubes, spheres, and ducks while being simple. We prototyped a simple arm in the Robot in 3 days challenge. This used a long u-channel piece that rotated around another u-channel fixed upward to the robot. To power this, we used a motor that was geared down to pivot the arm. We realized an arm for depositing freight was hard to position properly where the claw could perfectly intake freight.



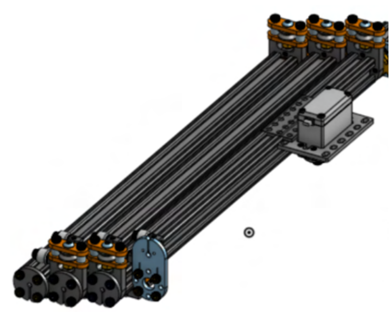
<u>While our arm design stayed the same (until V2), the claw went through 2 iterations, *which are described in depth in Intake V1 and V2*.</u>

CAD model of arm

V2 Lift & Bucket Combo

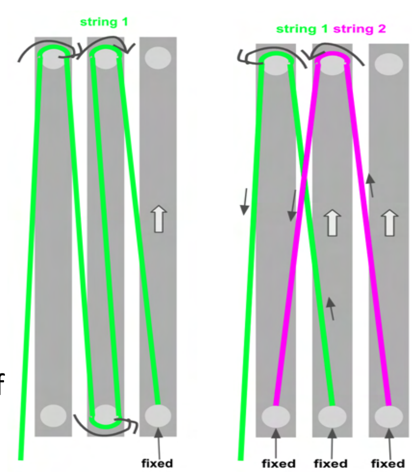
Linear Slider v1

Using what we learned from the Robot in 3 Days challenge, we decided to use a pulley and string based linear slide. Our linear slide and bucket systems consisted of 3 metal extrusions with pulleys attached to each end. We chose this because it was easy to move our outtake to all 3 levels of the team shipping hub, which is crucial for autonomous. A linear slide was faster than an arm, especially in the configuration we had it in.



In a traditional FTC linear slide mechanism, there is 1 continuous string that powers all the pulleys and moves the extrusions up one by one. However, to fully make use of the speed of the linear slides, we researched a way of stringing the slides in a way that moves all the extrusions at the same time thus making them faster. This is called cascade rigging and the way it works is that:

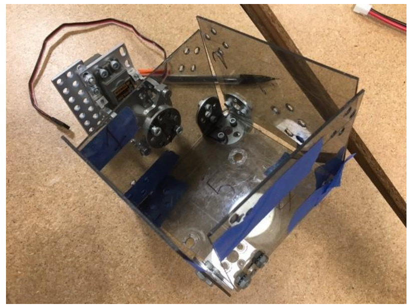
- The first stage has a spool powered by a motor on one end and the other end of the string goes over the top pulley of the first stage and is tied to the bottom pulley of the second stage.
- The 3rd stage moves up when the 2nd stage moves up because one end of another string is tied to the bottom of the 2nd stage while the other end is tied to the bottom of the 3 stage.
- We power the spool with a motor that has high torque because the motor is lifting 2 extrusions at the same time.



Continuous vs. Cascading

Bucket v1

To place blocks onto each shipping level and the shared shipping hub, we made a box that could hold cubes, balls, and ducks. Our first design used poly-carb panels that had inaccurate dimensions, with metal L-brackets that required us to drill holes.



Bucket v2

The first box was not very sturdy, so we decided to create a new one with increased rigidity. Equipped with the knowledge we learned from constructing the 1st box, we made the 2nd box. This box was carefully cut so perfect squares with the correct dimensions would be made. The end result was a 5 sided box that had clean edges and accurate dimensions. The walls are held together by electrical tape, allowing flexibility.

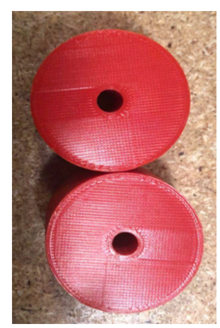
Lip design helps freight enter the robot

In addition, we cut one side a little longer, so that there would be a lip between the intake and the bucket to prevent freight falling out.

Bumpers	**Process used: Evolutionary**	**Iterations: 2**

V1 Rollers

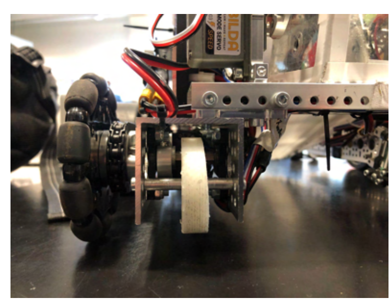
We decided to add rollers to the front and back of the chassis to get over the barrier better. In CAD, we added standoffs to the chassis to get an idea of where the rollers would go. Then, we sketched cylinders in CAD and used the chassis as a reference to how big they needed to be.



V2 Elevated Rollers

We noticed that since the rollers were mounted lower than the barriers, the robot would dig itself under the barrier. In addition, when we went to go size the robot we saw that it did not fit. We iterated another prototype where:

- We mounted the standoffs further inside the chassis
- We increased the diameter further so that the rollers would still stick out and work. After testing them they worked great. The robot got over the barrier effectively and was still size legal.



Team Shipping Element (Capstone)	Process used: Waterfall	Iterations: 2

V1 Magnetic Capstone

We wanted something that we could easily pickup laying on its side or standing straight up. The first design of our team shipping element was a cone made from a plastic folder (cut using a 2D cone template) and magnets, then later 3D printer the cone (for magnetic strength reasons). We taped neodymium magnets inside the capstone and attached ceramic magnets to an arm on a servo on our lift. We tested different magnets to achieve a good balance of strength. The capstone attaches and comes off easily (due to the tested balance of magnetic strength), we would just have to hover it over the Alliance Shipping Hub and push the capstone off.

V2 Hook and Loop

After testing this design we noticed two problems that required us to create a new version:

- The magnetic force required to pick up the capstone when not upright is more
- We couldn’t get a good strength balance for delivery, even after testing different magnet types (neodymium vs. ceramic) and sizes

Learning our lesson from V1 (not to use magnets), we brainstormed ways to pick it up, and we thought of a simple hook and loop. We thought this would be faster and more reliable than magnets. We started by building an L-shaped piece attached to a servo on the linear slide. The capstone has a rope loop for the hook to go into. During our *verification*, we saw this worked except for when it was on its side because the rope would be parallel to the ground. To fix this, we added weight to the bottom so it would roll into an orientation where the robot can grab it.

Carousel Spinner	Process used: Waterfall	Iterations: 1

V1 Motorized Compliant Wheel

Since the moment we first saw the game video, we knew that we should keep our carousel spinner design simple. This version was used on our first Robot in 3 Days robot and our 6-wheeled robot. Our initial brainstorming for the carousel was that:

- A motor would need to be able to spin the wheel fast for deliveries
- The actuator would need to have the capability to track it’s speed for adding complex software (the motor would need an encoder)
- The wheel would need compliance with the carousel and have grip to turn the carousel

We then used a grippy, squishy, 72mm diameter compliant wheel that was actuated by a 312 RPM (high speed) motor (1:1 gear ratio configuration) facing upwards on the robot. The tricky part was getting the measurement right so that the wheel would reach the carousel vertically.

Software Engineering Section

Software Principles and Values

One of our team goals is to not only build complex, intelligent robots, but to also cultivate **field-ready STEM skills** that can be used in the professional workspace. A major component in the array of skills we deem necessary to learn is programming. Our programming goals this year had three main goals: to promote programming knowledge in our **team**, **school**, and **community**. *You can find more information on our STEM pipeline and our development of STEM skills in our team and beyond in the "Outreach" section and "Sustainability" in the "Team Plan" Section. You can also find all information in the "Our Software" section of this portfolio mentioned in detail in our Control Award and Engineering Notebook, which you can ask for!*

EFFORTS IN TEAM:

1. Increased java knowledge in our team through CodeHs lessons
    - A. Beginner lessons focused on **variables**, **loops**, **conditionals**, and **methods**.
    - B. More intermediate lessons focused on **arrays** and more difficult **loop**, **variable** and **conditional** problems.
    - C. More advanced lessons focused on using the **distance algorithm** in a KNN Nearest Neighbors Problem, exploring **searching algorithms** in a maze problem, and the use of **trigonometry and calculus** in robotics through some engineering problems
2. Organized code into an online GitHub repository to increase collaboration in the code. This allowed everyone to remotely access and edit the code from anywhere.
3. **Provided Published Resources**, such as presentations, describing the ins and outs of our code for easy comprehension *(ask us about them if you want to see the presentations)*.

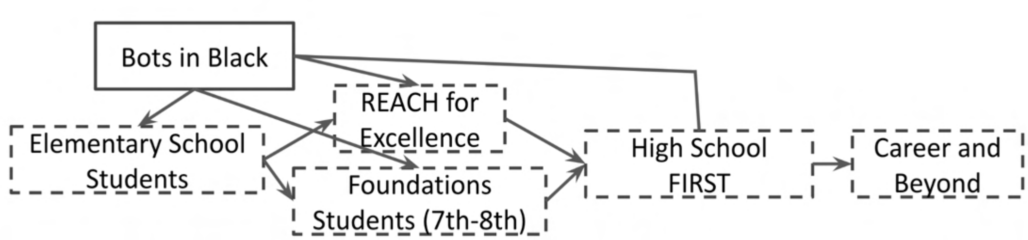
EFFORTS IN SCHOOL:

1. We were the first team to have a working camera and we **shared our knowledge** with other teams
2. We made various presentations on our challenges and discoveries with computer vision.
3. We worked directly with several Marist Teams in setting up their computer vision and just with simple programming issues.

EFFORTS IN COMMUNITY:

1. Teaching SketchUp, Java Programming, Robotic Design, Scratch, and EarSketch at REACH *More information on REACH is under "Outreach" in the "Team Section"*

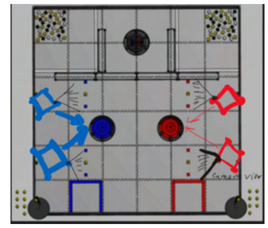
Below is a diagram of how our role, inspiring STEM in our team, school and community works:



Our Software

Autonomous	**Current Version: Dual Control Auto**	**Iterations: 3**

<u>What</u>- We drive diagonally at the Shipping Hub using a different camera on either side
<u>Why-</u> We use a tank drive, so an obstacle this year has been getting a reliable autonomous (tank cannot strafe like mecanum). With two cameras, we can go diagonally on the field to the shipping hub
<u>How -</u> First the robot drives straight up to the shipping hub, places a block, returns to the wall, then goes to park



Driver Control	**Final Version:Dual Control**	**Iterations: 2**

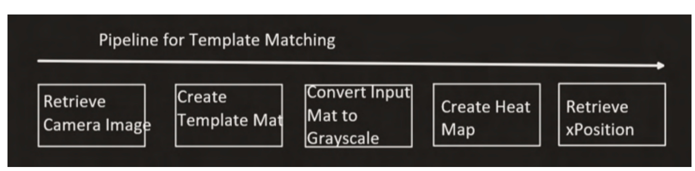
<u>Iteration 1</u> **TeleopTank** Designed for only one controller, it's easy to use but lacks many features.
<u>Iteration 2</u> **DualControlTeleop:** After Meet 1, we wondered if our Alliance Shipping Hub delivery time could be faster with two controls, by having one person control the outtake and the other the intake and driving. We could also have more buttons to use (one controller isn't enough).

There are basically 4 types of controlled robot functions in the code
1.) functions player 1 always controls
2.) functions player 2 always controls
3.) functions necessary for robot functionality that player 2 controls
4.) functions unnecessary for robot's functionality that player 1 controls

A special button also reverts all controls to player 1 as a fallback for bugs and issues.

1	2	3	3
Drive(left and right joystick)	Linear slider automatic positions(dpad up and down)	Manual linear slider control(right and left triggers)	Capstone manual control(right and left triggers)
Forward and reverse intake(dpad right and left)	Carousel speed modifiers(right and left bumpers)	Bucket cycle(a button)	
Capstone arm automatic positions(right bumper)	Automatic Carousel Control, the timed spinners(x and b)		
drivespeed(speed cap is back button and speed decreaser is left bumper)			
Normal carousel spinners(x and b)			

Computer Vision	**Used in : Autonomous**	**Iterations: 3**



Step1 Retrieve Camera Image: Get camera input through processFrame method
Step 2 Create Template Mat: Create a Mat containing the image we want to locate, a barcode in our case because its colors of black and white are easier for the camera to locate, then matching the mat with 255 (black) and 0 (white)
Step 3 Convert Input Mat to Grayscale: Next, we use ImgProc to make the mat grayscale.
Step 4 Create Heat Map: Take two Mats and the type of mat we want (heat mat). Using Normalized Correlation (SSD Sum of Squared Differences seen on the right) the method returns us a heat mat where the brightest point (greatest value) is the area of best fit and the darkest is

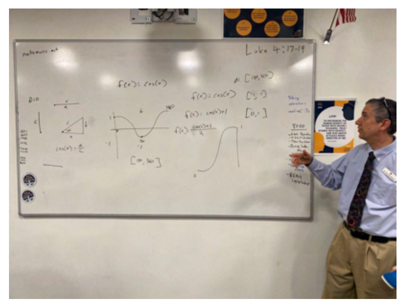
the least. For more detail into SSD, ask about our presentation!

Step 5 Retreive Xpos: Using the MinMax class we find the greatest (brightest) point in the Mat and locate its xPosition. We then call a getter method of the Pipeline from the autonomous to retrieve the x value and use it to determine the zone(if xPos < 100, zone = 1 ect..).

Carousel Code	**Used in : Driver Control**	**Iterations: 3**

Spinning ducks off the carousel in autonomous and endgame is a **high scoring point opportunity** that our team wanted to take maximum advantage of at the beginning of the season. The motor can only spin so fast until the momentous resistance of the duck being spun overpowers the disk coefficient of friction, causing the duck to spin off. This puzzle generated some interesting solutions and we are really proud of the evolution of our code.

Iteration 0 - **Manual Power** The carousel would spin when a button was held down.
Iteration 1 - **On/Off Toggle** Turns the carousel on and off with a single button.
Iteration 2 - **Blue and Red Carousel** Spin clockwise and counterclockwise for Red & Blue sides.
Iteration 3 - **Timed Carousel Turning** Used Elapsed Time to control the spinning of the carousels
Iteration 3.5 - **Carousel Speed Tuning** - Functionality to increase and decrease carousel speed
Iteration 4.0- **Cosine Acceleration-** Accelerating up to our max speed. With the help of our robotics teacher, we learned about the different types of functions with curves, and we learned that a cosine function works best and by modeling its upward curve we could replicate a slow acceleration at the start and gets faster as time increases.



$$power(\vartheta) = (cos(\vartheta - \pi) + 1) \times 1/2$$

In this equation above, we first

- Getting a 0-1 value from inputting 180-360 in the cos()
- We specify an integer counter variable (about 1 - 3). By multiplying 180 (the amount of the possible x values for the isolated function) by the counter, we can get a derivative of how fast we want to accelerate.
- Then, we convert our counter (theta) to radians, which is used by Math.pi() in order to determine our increasing motor power!

PID Controller	**Used in: Autonomous**	**Iterations: 1**

A PID (proportional: the size of the error, integral: past error, derivative: rate of change) controller is an algorithm that is used to achieve a desired output based on given input. This year, our issue was that our robot turned blindly autonomous (without any sensor input), so we turned unreliable most of the time, messing up our autonomous and causing us to lose potential points. We use the input of an IMU, which measures the robot's heading, to turn accurately. We create our error, the difference between our target and where we are, and run it through the equation that has coefficients we manually change, and for every loop cycle, the algorithm gives us an output of motor power for our drivetrain.

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_p de/dt$$

Team Section

Team Summary

We are the Bots in Black, team 16633 from Marist School in Atlanta, Georgia. As a third year FTC team, our mission is to:

***B**reak barriers, **I**nnovate, and **B**uild community*

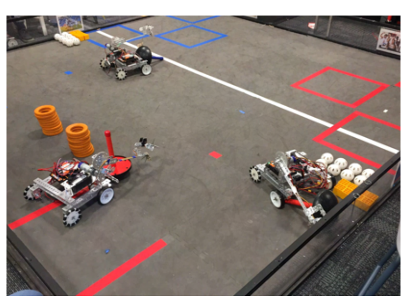
We focus on **promoting STEM to the community** through various forms of outreach, assisted by building reliable and efficient robots, with a concentration on the control systems involved. We achieve by helping others succeed in order to promote gracious professionalism. We strive to **build our engineering skills**. We work to solve problems, constantly undergoing trial and error! We excel on spreading STEM values throughout our community, and including all members on our team!

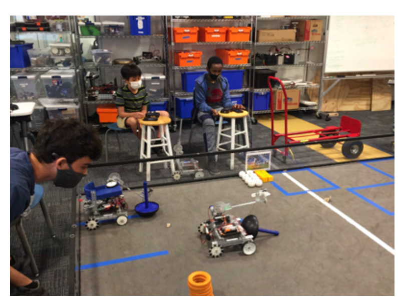
Outreach

Reach for Excellence	Impact: 82 people	Category: Motivate

Reach for Excellence is a tuition-free academic and leadership enrichment program that offers students of limited income a mix of academic, cultural, and community-based experiences that prepare them for the challenges of college-preparatory programs and high schools.

Our team this summer volunteered and assisted with Computer Science and Engineering instruction at REACH summer sessions. They helped empower 6th through 8th Grade students in the Atlanta area with skills in programming and engineering. In addition, we remixed an FTC game where students would build robots using FTC build systems and compete in a customized game in an FTC style, giving them a taste of FIRST and how it works. Overall, we volunteered at REACH for **145 hours**.





In addition, REACH allows us to maintain our *STEM Pipeline* by teaching the REACH students, who feed a significant number of incoming students at Marist School and in our robotics teams.

ItGresa Robotics	**Impact: Team Members**	**Category: Connect**

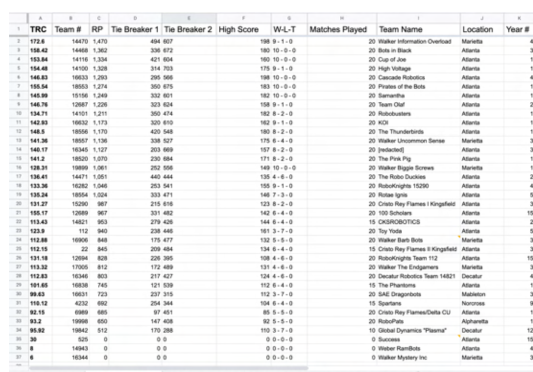
One of our close friends from team 14470s parents run a company called *ItGresa Robotics*, a combat robotics vendor. We reached out about the **Engineering Design Process** through Zoom!

Table of Outcomes and Lessons:

- They told us of their design process, which is very similar to ours, of
  - Ask: what is the problem
  - Imagine: think of many creative possible solutions to the problem
  - Plan: lay out steps of our the solution is going to be created
  - Create: follow the plan
  - Improve: after testing, make the solution better
  - Communicate: talk to others about feedback for the solution, possibly making a new iteration!
- A common error that engineers make is upsetting the balance between perfectionism and rashness. Meet the goal in a strategic, efficient way by sticking to your plan.
- The core of the engineering design process is a problem-solution model. First find a problem, then create a solution. If the problem is not completely solved or met, then you improve that solution or make a new one!

Scouting Spreadsheet	**Impact: 36 FTC Teams (~252 people)**	**Category: Motivate**

Our team created a Google Sheets spreadsheet that is a source of information for <u>teams in our league</u> scouting other teams, whether that may be for alliance selection or researching the opposing alliance. For example, the spreadsheet holds multiple sheets: a comprehensive list of all the teams, individual and combined meet results, team information and history, and a homemade point-based team rating system!



TRC	Team #	RP	Tie Breaker 1	Tie Breaker 2	High Score	W-L-T	Matches Played	Team Name	Location	Year #
172.6	14470	1,470	494	607	198	9 - 1 - 0	20	Walker Information Overload	Marietta	4
158.42	14468	1,362	336	672	180	10 - 0 - 0	20	Bots in Black	Atlanta	3
153.84	14116	1,334	421	604	160	10 - 0 - 0	20	Cup of Joe	Atlanta	1
154.48	14100	1,328	314	703	175	9 - 1 - 0	20	High Voltage	Atlanta	1
146.83	16633	1,293	295	566	198	10 - 0 - 0	20	Cascade Robotics	Atlanta	4
155.54	18553	1,274	350	675	183	10 - 0 - 0	20	Pirates of the Bots	Atlanta	1
145.99	15156	1,249	332	601	182	10 - 0 - 0	20	Samantha	Atlanta	1
146.76	12687	1,226	323	624	158	9 - 1 - 0	20	Team Olaf	Atlanta	2
134.71	14101	1,211	350	474	182	8 - 2 - 0	20	Robobusters	Atlanta	1
142.93	16632	1,173	320	610	162	9 - 1 - 0	20	KOI	Atlanta	1
148.5	18556	1,170	420	548	180	8 - 2 - 0	20	The Thunderbirds	Atlanta	1
141.36	18557	1,136	338	527	175	6 - 4 - 0	20	Walker Uncommon Sense	Marietta	3
140.17	16345	1,127	203	669	157	8 - 2 - 0	20	[redacted]	Atlanta	3
141.2	18520	1,070	230	684	171	8 - 2 - 0	20	The Pink Pig	Atlanta	1
128.31	19899	1,061	252	556	149	10 - 0 - 0	20	Walker Biggie Screws	Marietta	1
136.41	14471	1,051	440	444	135	4 - 6 - 0	20	The Robo Duckies	Atlanta	2
133.36	16282	1,046	253	541	155	9 - 1 - 0	20	RoboKnights 15290	Atlanta	4
135.24	18554	1,024	333	471	146	7 - 3 - 0	20	Rotae Ignis	Atlanta	5
131.27	15290	987	215	616	123	8 - 2 - 0	20	Cristo Rey Flames I Kingsfield	Atlanta	3
155.17	12689	967	331	482	142	6 - 4 - 0	20	100 Scholars	Atlanta	15
113.43	14821	953	279	426	144	6 - 4 - 0	15	CKSROBOTICS	Atlanta	2
123.9	112	940	238	446	161	3 - 7 - 0	20	Toy Yoda	Atlanta	5
112.88	16906	848	175	477	132	5 - 5 - 0	20	Walker Barb Bots	Marietta	3
112.15	22	845	209	484	134	6 - 4 - 0	15	Cristo Rey Flames II Kingsfield	Atlanta	3
131.18	12694	828	226	395	108	4 - 6 - 0	20	RoboKnights Team 112	Atlanta	15
113.32	17005	812	172	489	131	4 - 6 - 0	20	Walker The Endgamers	Marietta	3
112.83	16346	803	217	427	124	4 - 6 - 0	20	Decatur Robotics Team 14821	Decatur	4
101.65	16838	745	121	539	112	6 - 4 - 0	15	The Phantoms	Atlanta	1
99.63	16631	723	237	315	112	3 - 7 - 0	20	SAE Dragonbots	Mableton	3
110.12	4232	692	254	344	104	6 - 4 - 0	15	Spartans	Norcross	9
92.15	6989	685	97	451	85	5 - 5 - 0	20	Cristo Rey Flames/Delta CU	Atlanta	1
93.2	19998	650	147	408	92	5 - 5 - 0	20	RoboPats	Alpharetta	1
95.92	19842	512	170	288	110	3 - 7 - 0	10	Global Dynamics "Plasma"	Decatur	12
30	525	0	0	0	0	0 - 0 - 0	0	Success	Atlanta	15
8	14943	0	0	0	0	0 - 0 - 0	0	Weber RamBots	Atlanta	4
6	16344	0	0	0	0	0 - 0 - 0	0	Walker Mystery Inc	Marietta	3

We made flyers that show a link to the spreadsheet and QR code, and we passed these out to all competing teams. It really helped teams at the league tournament!

Assisting Rookie Team 12687	**Impact: 10 people (4 present)**	**Category: Motivate**

At Marist Robotics, our role as a veteran team is to share our First Tech Challenge expertise with younger teams, so that the STEM pipeline in our program is continuous, making sure that teams young and old can learn, compete, build, and code to their fullest potential. Specifically, this season we **assisted** Team 12687, a first-year, 7th grade team. Not too much unlike us 3 years ago, they have a goal to advance to the state championship. Our team discussed with them their FTC interests.. Then, we told them about how advancement works. They were really excited about the Connect Award, so we discussed ways they could reach out to STEM professionals. We told them about our experience winning the Connect Award in the 20-21 season, and the ways we connected with professionals. Before long, they had events lined up with their team. In addition, our team captain told them about how they can best manage and coordinate with their team in order to increase involvement and improve their team dynamic, a crucial part of winning FTC awards.

Cristo Rey Lunch	**Impact: 3 people**	**Category: Connect**

During the school day in robotics, Marist endorses the Cristo Rey Jesuit Highschool work program by employing student interns. Before the school year, our team, the Cristo Rey Computer Science & Engineering interns, the STEM teacher from Cristo Rey, and our robotics teacher had lunch to talk about the school year, robotics in FTC and in the classroom, and an introduction to STEM at Cristo Rey by Ms. Cooke, the Cristo Rey STEM teacher. We were able to gain and share valuable insight into the engineering field.

Reveal Day and Build Days with Cristo Rey	**Impact: 24? people**	**Category: Motivate**

On the reveal day of this year's FTC game, we **hosted** our friends at Cristo Rey Jesuit Highschool, teams 16282, 16906, and 19998 to watch the game reveal. We then **assisted** in brainstorming strategy, prototyping, and building! In addition, we helped these teams throughout the season by **hosting** multiple weekend build days throughout the season for the three teams to be able to use the tools and materials in our lab, since our lab has more manufacturing supplies since we provide for 16

teams. We also lent three REV control hubs to the teams and lent multiple Tetrix build system kits in order to give them the ability to build a competitive robot.

Note - hosted is used in this context to a unofficial FTC event, rather than official events as described in Appendix P of the Judges' Manual

Atlanta League Hosting	**Impact: 36 FTC Teams (~252 people)**	**Category: Motivate**

Our team hosts the GA Atlanta League, the one that we currently participate in through the Marist Robotics program. We **support** the league set up and clean up the meets to provide a competition for many teams in the Atlanta area, serving as an ambassador for FIRST by hosting 36 FTC teams, roughly around 252 people total! In addition, we provide building tools for teams to use during the meets, and we are always willing to lend a hand to teams!

Open House	**Impact: ~100**	**Category: Motivate**

One of our members volunteered for our school's Open House, where they were a representative for the Marist Robotics Program. "I had the opportunity to talk to students interested in Marist and robotics, and I showed them our robot, described our team, and talked about FIRST and FTC. For example, one 8th grader looking to go to Marist said that he did a lot of computer programming (which is one of my passions inside and outside of robotics), and I told him how I was able to apply programming concepts and skills into our FTC robot, and the fun that I had doing that. An adult that I talked to worked at Amazon Web Services, and programmed their database service. He said that he uses Java and Android Studio, two technologies we use to code our robot, in order to build database services used by major companies! Not only was I able to promote FIRST and FTC, but I also learned of opportunities that FIRST opens me up to!"

Team Plan

Outreach Aim

On the Bots in Black, we acknowledge that FTC is not only just robots. A key component to the First Tech Challenge is interacting with your community, promoting STEM and gaining real-world advice from technological professionals. Through all of this, we commit to the values of our team and school organization.

Sustainability

Budgeting

At Marist Robotics, our parent organization, robotics teams are formed through the elective class at school. Thus, our FTC team is funded by Marist School. Our lab, materials, and team costs are also subsidized by registration fees from team members.

Our team fundraises in order to buy parts that may not be available in our lab. Specifically, we acquired multiple sponsorships that allow us to advance our equipment. For example, we received a sponsorship from the robotics parts company GoBilda that allowed us to buy drivetrain parts that we could not previously buy. In addition, we acquired 3 new sponsors this year: REV robotics, SendCutSend, and ItGresa Robotics.

STEM Pipeline and Team Skill Improvement

The Bots in Black represent our larger organization, Marist Robotics, maintaining our school's STEM Pipeline (the flow and improvement of STEM knowledge in our community). Info on community, specifically pipeline efforts, located in the Software *Principles and Values Section*.

Our plan is to accomplish this through reaching younger and new STEM students, specifically the Foundations (7th and 8th Graders at Marist) students and REACH for Excellence students, an organization that brings a significant number of students to Marist School.

- We mentor our Foundations students by showing and teaching them **engineering, building, designing, and coding technologies, skills, and ideas** of older teams.
- We volunteer at REACH sessions to teach students about STEM concepts (ones above).
- We held new member sessions to **teach our team** about REV and GoBilda FTC systems.

Recruitment

Our member recruitment comes from **promoting FTC to fellow students** at our school. We recruited 3 new members this year from that alone. Additionally, along with other Marist robotics teams, we participated in a scrimmage FTC match during a school assembly, to show our classmates what FIRST is like and to encourage those thinking about joining. We also **gained new knowledge from our mentor**. We talked with our Computer Science Teacher largely about Advanced Math. He taught us how to find a *derivative* & an *integral*, *vector* equations, *trigonometry*, and robot arm *kinematics*. These topics helped us design our complex software.

Performance

In addition to focusing on our community and our STEM skills as a team, we wanted to focus on strategy for the design of our robot and pre-match strategy. We started with a Robot in 3 Days build, where we built a robot that could play the game while keeping the design simple and efficient. We did this so that we could a better idea of strategy and the type of robot we wanted to build. The next way we accomplished this goal was by having our pit team maintain a practice schedule of a buildup of time spent doing driver practice the weeks before meets. Finally, we created strategy sheets with us and our partner's strategy. This also helped us spread **Gracious Professionalism** because we were able to have meaningful conversations and interactions with other teams, whether we won or lost, and we were able to answer questions about our robot and inspire others about STEM through our experiences! All of this has allowed us to maintain a **Top 5 Position** and set the **Georgia highscore** with 14470 through communication and strategy!