FIRST®

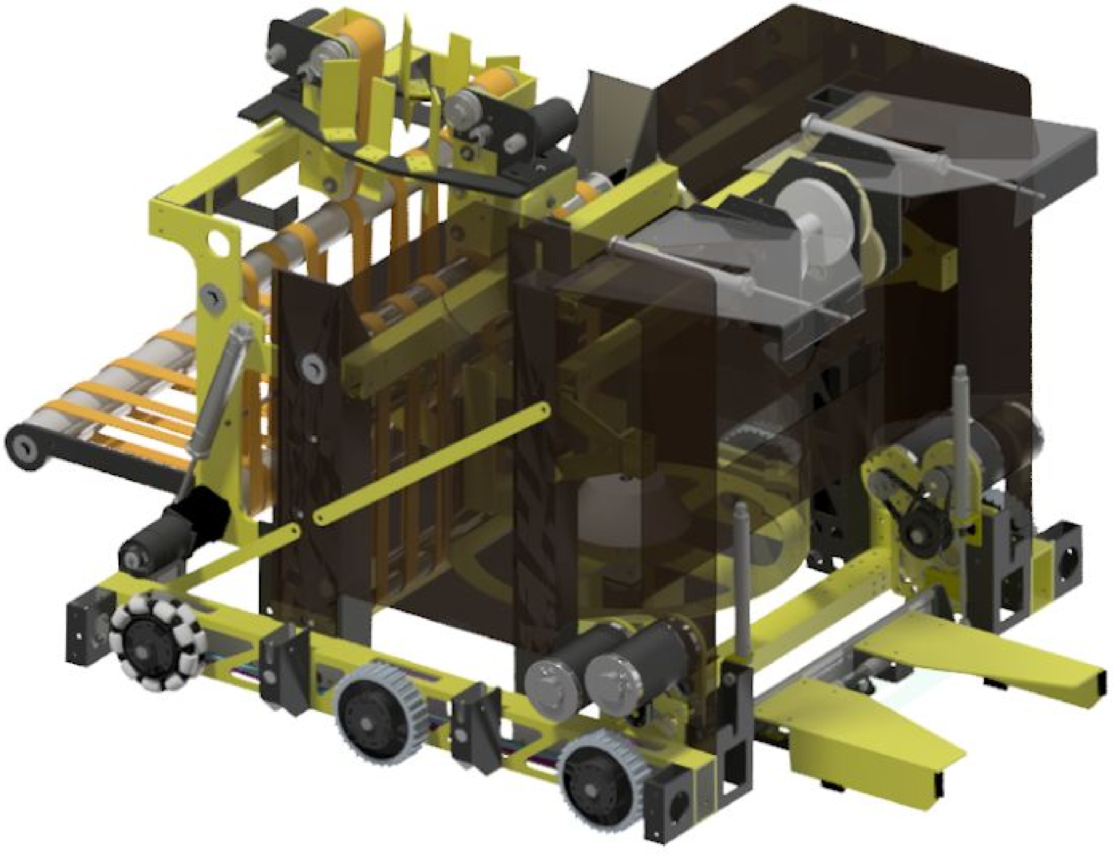
846

The Funky Monkeys

Lynbrook Robotics

Intentionally blank

The Funky Monkeys
Team 846



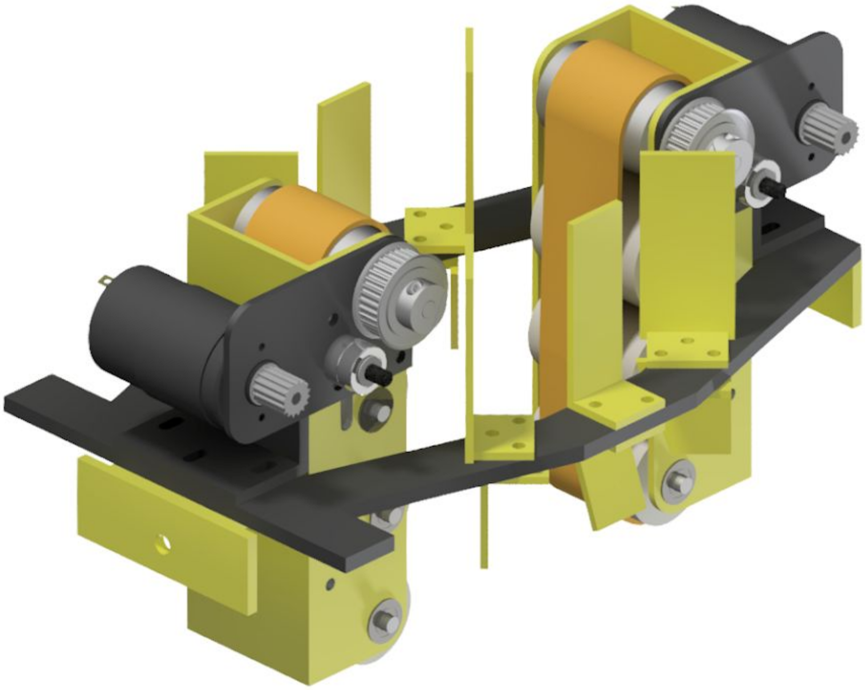
Introducing our 2017 Robot

Punk Monkey

Shooter

James Jiao (Sophomore), Gautam Rajesh (Junior)

The ball shooter shoots out fuel at a rate of around 4 fuel per second into the high goal. Its novel flipping design allows the robot to shoot from the walls cornering the boiler, allowing the shot to be taken from a safe position. We also implemented a double-sided belt design, allowing more energy to be transferred to the fuel in a smaller shooter profile.



Design Requirements:

1. Shoot the fuel into the high goal with 80% accuracy
2. Shoot fuel at a rate of at least 4 balls per second
3. Shoot from the two walls on either side of the boiler, so as to provide a safe location for continuous shooting that does not obstruct our allies.

To achieve these goals, we needed a shooter that was able to shoot from multiple positions and had the power needed to shoot a continuous stream of fuel.

Belt Shooter

This year we decided to step outside our comfort zone and design a never-seen-before belt shooter. The potentials of the belt shooter were immense, as shown through the energy and momentum calculations we did. There were three major upsides of having a belt shooter:

1. More distance that the rollers are in contact with the ball.

   As opposed to having a roller or two on each side, having a belt allows the fuel to be in contact with the rollers more. This year especially, the fuel balls compress in an unpredictable way, tending to bend in at a specific point. For the ball to be in contact with the rollers the same distance as the belt, the rollers must be much bigger. This means having a belt shooter leads to a smaller profile shooter, while still providing the necessary distance.

   So why does having more contact distance matter? Since E = F * d, the more energy the fuel is in contact with the rollers, the more energy is transferred. The forces that are imparted on the ball are limited due to the normal force, or the compression force. In other words, to increase the energy transferred, there needs to be a greater distance the ball is contacting the rollers.

2. No spin on the balls = less energy.

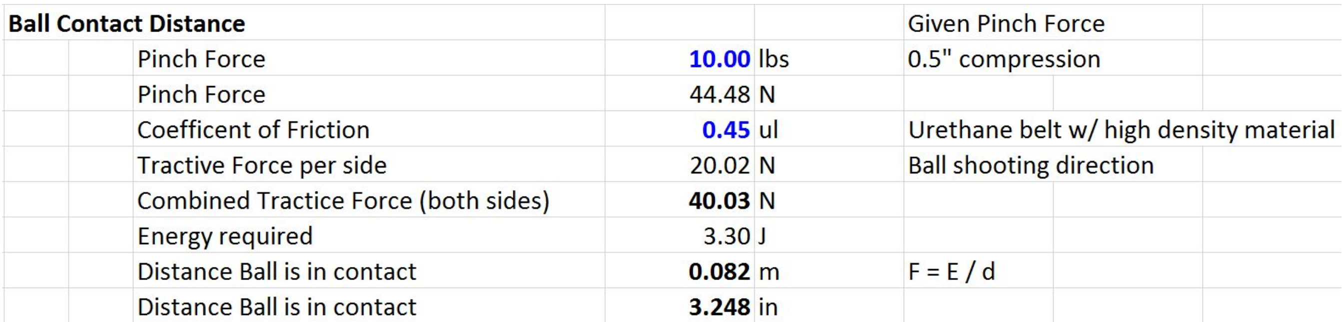
   Having rollers on each side means the ball will come out of the shooter with no spin. This is not the case for a single flywheel hooded design, which launches the fuel out with backspin. Backspin may stabilize the fuel as it flies, but we discovered that if we controllably decompress the ball as it exits and provide a secure guide, the fuel shot out can still be straight. To figure out how much momentum is required to put spin on the ball, we use the following equations:

$$\boxed{\begin{gathered} E = \frac{1}{2}m_{ball}v^2 + \frac{1}{2}I\omega^2 \\ I = \frac{2}{3}m_{ball}R^2 \\ E = \frac{1}{2}mv^2(1 + \frac{2}{3}) \end{gathered}}$$

   As one can see, the moment of inertia for a hollow sphere is $\frac{2}{3}mR^2$ This means that when you add spin in a single roller shooter, you must put two-thirds more energy into the ball.

3. Two times the tractive force to propel the ball.

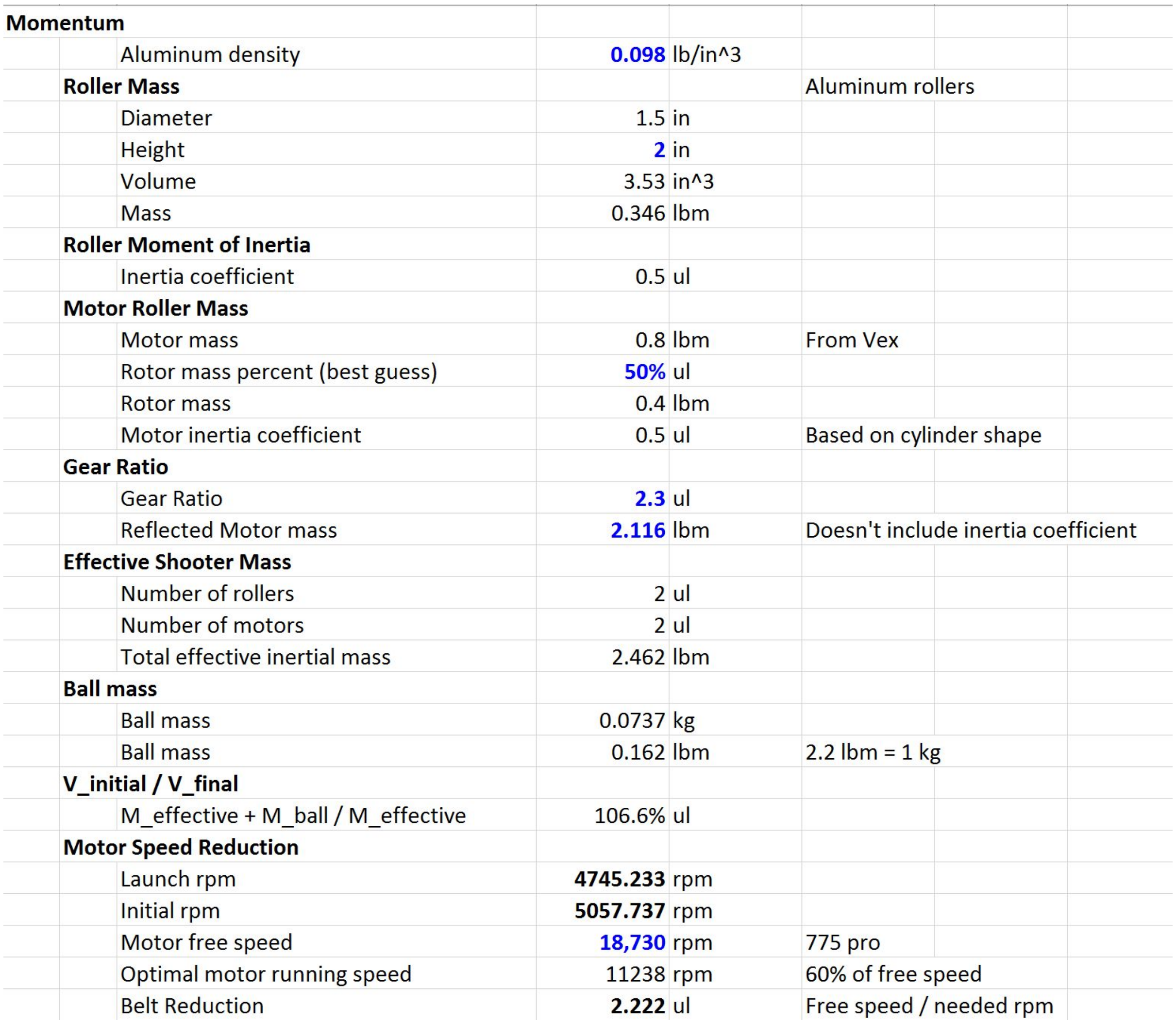
Having a double-sided shooter means you will be able to transfer force from two sides. Since the shooters are friction limited, this provides twice the tractive force when compared to a single flywheel shooter.



Ball Contact Distance			Given Pinch Force
Pinch Force	**10.00**	lbs	0.5" compression
Pinch Force	44.48	N	
Coefficent of Friction	**0.45**	ul	Urethane belt w/ high density material
Tractive Force per side	20.02	N	Ball shooting direction
Combined Tractice Force (both sides)	**40.03**	N	
Energy required	3.30	J	
Distance Ball is in contact	**0.082**	m	F = E / d
Distance Ball is in contact	**3.248**	in	

In a single-flywheel hooded design only half of the compression force can be used, as the other half is lost through pushing against the hood. In the double-sided shooter, each side can capitalize on the compression force, or pinch force, and double the combined tractive force.

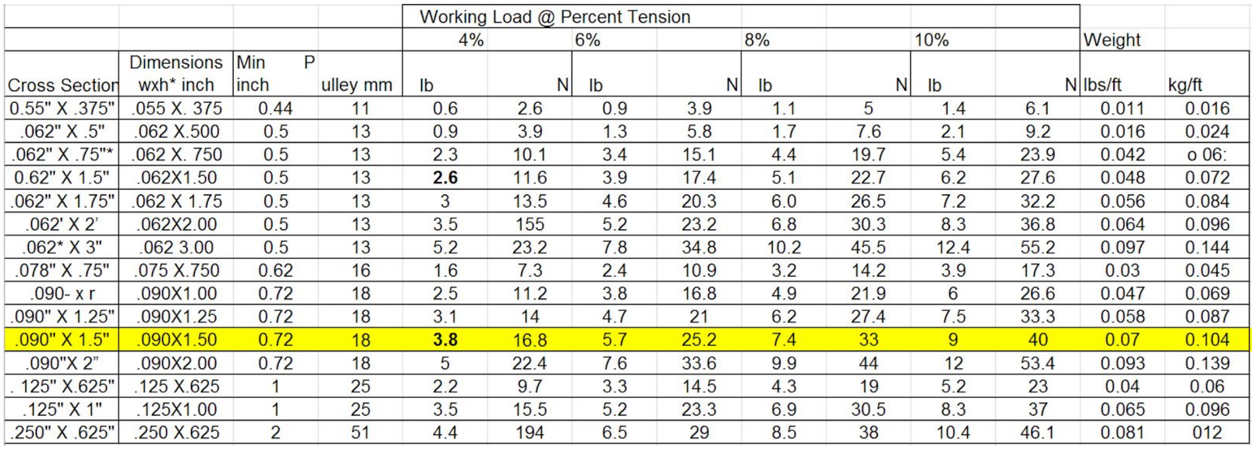
All of these factors led our team to develop the belt shooter, which is something we’ve never done or seen before.



Momentum				
	Aluminum density	**0.098**	lb/in^3	
Roller Mass				Aluminum rollers
	Diameter	1.5	in	
	Height	**2**	in	
	Volume	3.53	in^3	
	Mass	0.346	lbm	
Roller Moment of Inertia				
	Inertia coefficient	0.5	ul	
Motor Roller Mass				
	Motor mass	0.8	lbm	From Vex
	Rotor mass percent (best guess)	**50%**	ul	
	Rotor mass	0.4	lbm	
	Motor inertia coefficient	0.5	ul	Based on cylinder shape
Gear Ratio				
	Gear Ratio	**2.3**	ul	
	Reflected Motor mass	**2.116**	lbm	Doesn't include inertia coefficient
Effective Shooter Mass				
	Number of rollers	2	ul	
	Number of motors	2	ul	
	Total effective inertial mass	2.462	lbm	
Ball mass				
	Ball mass	0.0737	kg	
	Ball mass	0.162	lbm	2.2 lbm = 1 kg
V_initial / V_final				
	M_effective + M_ball / M_effective	106.6%	ul	
Motor Speed Reduction				
	Launch rpm	**4745.233**	rpm	
	Initial rpm	**5057.737**	rpm	
	Motor free speed	**18,730**	rpm	775 pro
	Optimal motor running speed	11238	rpm	60% of free speed
	Belt Reduction	**2.222**	ul	Free speed / needed rpm

After doing the momentum calculations to see how much momentum we would lose after shooting each ball, we saw that the motors only dropped 6.6% after each ball. The motor's will easily be able to spin back up to speed with such a little drop, allowing us to be able to shoot at a high rate.

After finalizing on a belt shooter, we had to choose the proper belts. We had compiled a list of flat belts and how much they stretched when experiencing force. From this we chose 1.5" wide 0.09" thick urethane belts for our belt shooter. With the forces the shooter is experiencing, we pre-tensioned the belts by 5% to compensate for the stretching.



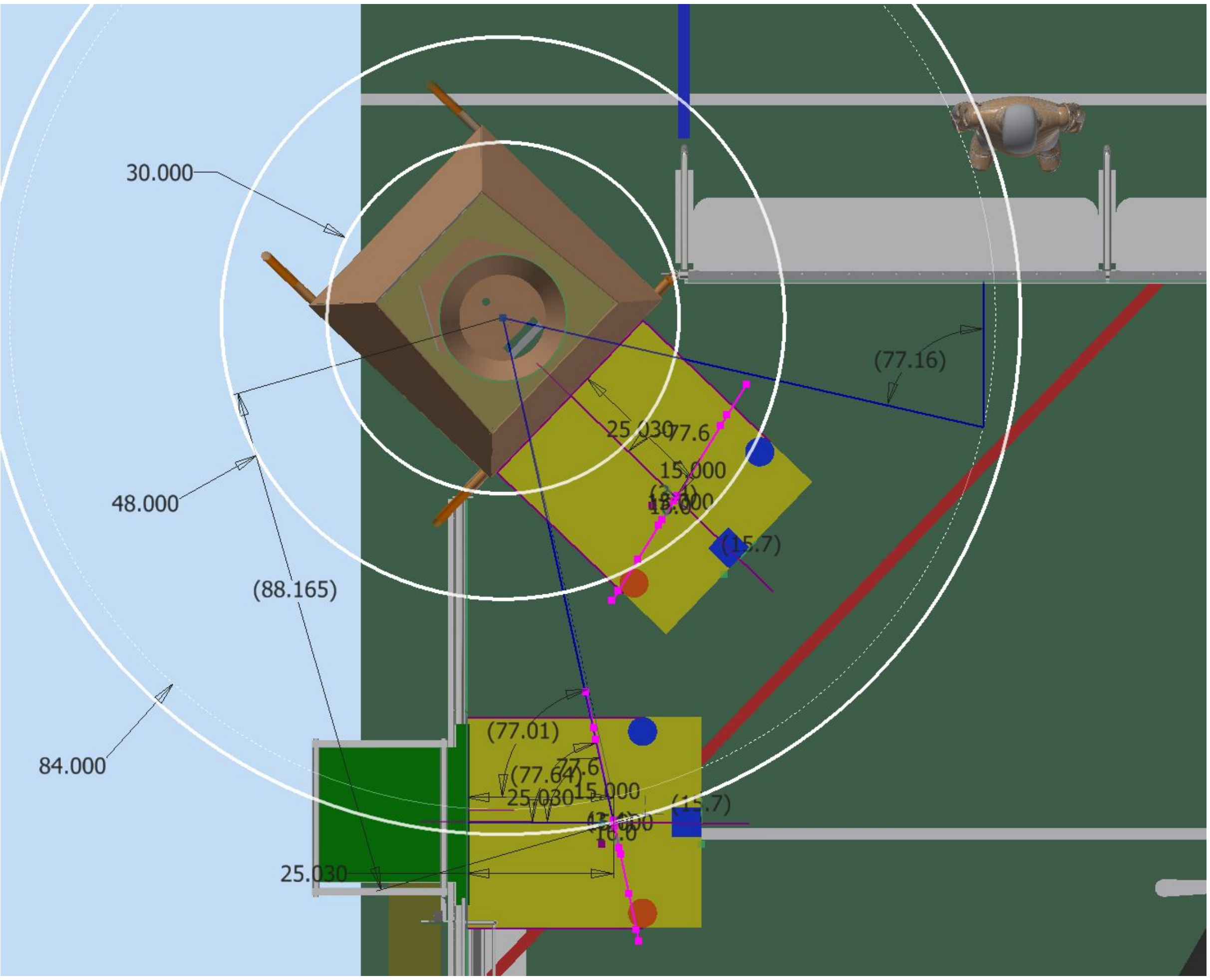
				Working Load @ Percent Tension									
				4%		6%		8%		10%		Weight	
Cross Section	Dimensions wxh* inch	Min P inch	ulley mm	lb	N	lb	N	lb	N	lb	N	lbs/ft	kg/ft
0.55" X .375"	.055 X. 375	0.44	11	0.6	2.6	0.9	3.9	1.1	5	1.4	6.1	0.011	0.016
.062" X .5"	.062 X.500	0.5	13	0.9	3.9	1.3	5.8	1.7	7.6	2.1	9.2	0.016	0.024
.062" X .75"*	.062 X. 750	0.5	13	2.3	10.1	3.4	15.1	4.4	19.7	5.4	23.9	0.042	o 06:
0.62" X 1.5"	.062X1.50	0.5	13	**2.6**	11.6	3.9	17.4	5.1	22.7	6.2	27.6	0.048	0.072
.062" X 1.75"	.062 X 1.75	0.5	13	3	13.5	4.6	20.3	6.0	26.5	7.2	32.2	0.056	0.084
.062' X 2'	.062X2.00	0.5	13	3.5	155	5.2	23.2	6.8	30.3	8.3	36.8	0.064	0.096
.062* X 3"	.062 3.00	0.5	13	5.2	23.2	7.8	34.8	10.2	45.5	12.4	55.2	0.097	0.144
.078" X .75"	.075 X.750	0.62	16	1.6	7.3	2.4	10.9	3.2	14.2	3.9	17.3	0.03	0.045
.090- x r	.090X1.00	0.72	18	2.5	11.2	3.8	16.8	4.9	21.9	6	26.6	0.047	0.069
.090" X 1.25"	.090X1.25	0.72	18	3.1	14	4.7	21	6.2	27.4	7.5	33.3	0.058	0.087
.090" X 1.5"	.090X1.50	0.72	18	**3.8**	16.8	5.7	25.2	7.4	33	9	40	0.07	0.104
.090"X 2"	.090X2.00	0.72	18	5	22.4	7.6	33.6	9.9	44	12	53.4	0.093	0.139
. 125" X.625"	.125 X.625	1	25	2.2	9.7	3.3	14.5	4.3	19	5.2	23	0.04	0.06
.125" X 1"	.125X1.00	1	25	3.5	15.5	5.2	23.3	6.9	30.5	8.3	37	0.065	0.096
.250" X .625"	.250 X.625	2	51	4.4	194	6.5	29	8.5	38	10.4	46.1	0.081	012

Needless to say, there were concerns about the belt shooter we had. One was the concern that the belts would de-center and potentially fall off the rollers. To fix this problem, we add aligning plates to ensure the belt wouldn't fall off and crowned the drive pulley. From our previous experience, by crowning the rollers, the belt would center itself.

Another concern was wear on the belts. As shown before, we chose to use polyurethane belts. After rigorously testing the shooter, we noticed that the ball was leaving residue as it passed through, possibly due to high forces and friction it experiences. However, we found absolutely no wear on the belts!

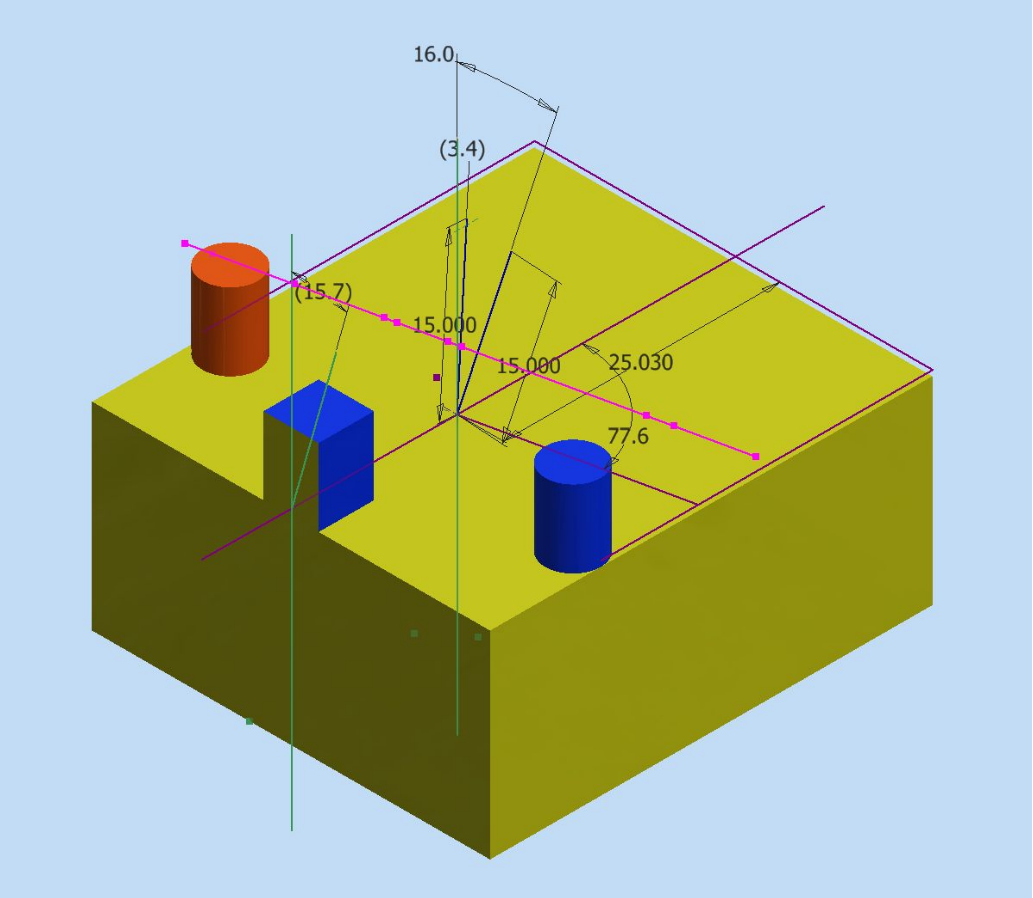
Flip Shooter

Another novel aspect our shooter incorporates is the pivoting mechanism that allows it to shoot from different angles without much movement. One of our design requirements was to be able to shoot from the walls bordering the boiler, as we anticipated heavy defense in this year's game.



However, we saw that a turret would need to rotate 150° to achieve the range we are requesting. That would take up a huge amount of space on our robot, not to mention all the wires and cables flinging around.

By drawing up a model of the field, we saw that if the exit vector pointed toward the goal, we could make the shot. Then we realized that a pivoting shooter could minimize the amount of rotation.

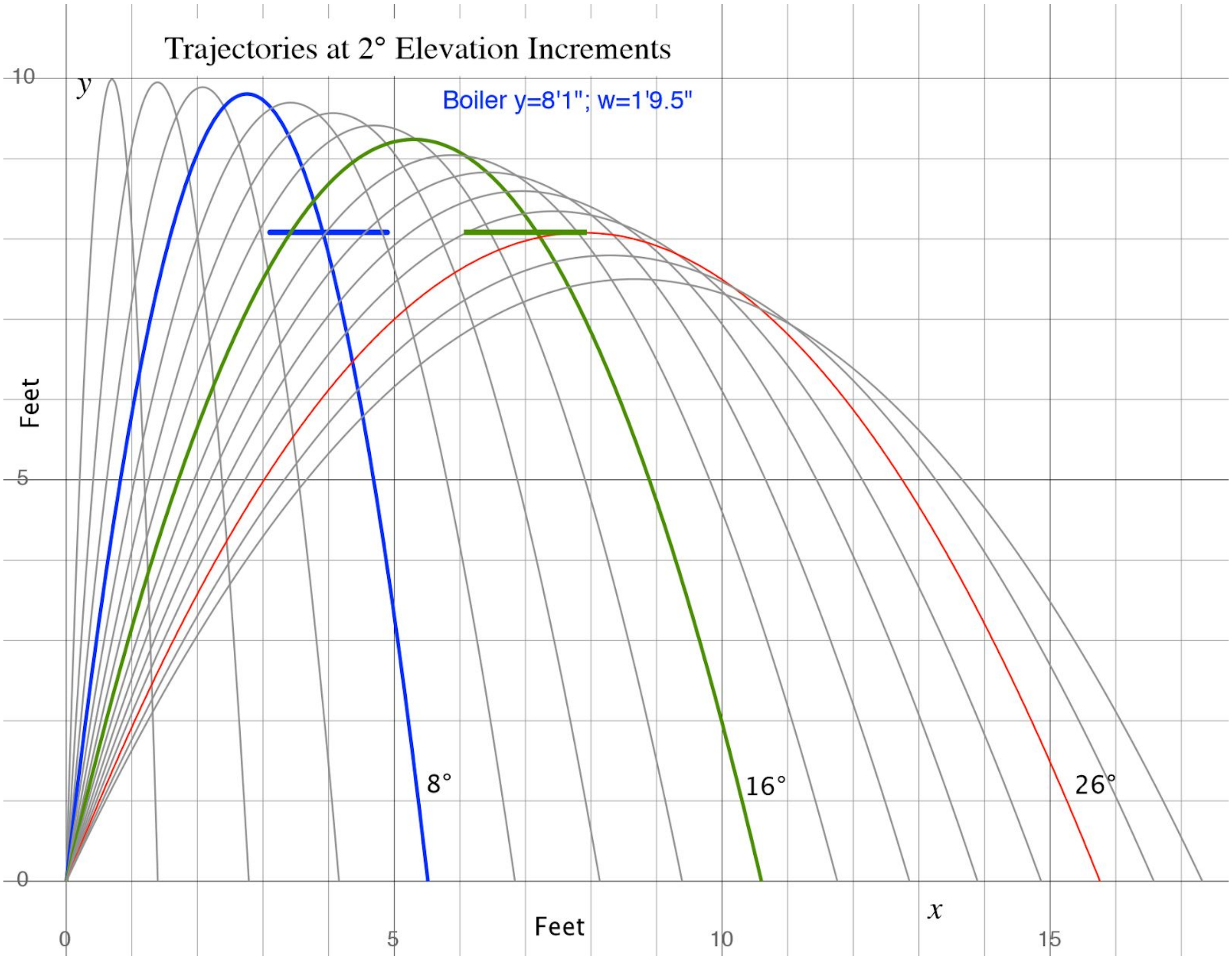


By using CAD to draw up a model, we saw that such a pivoting mechanism would only move ±15° each way, or 30° total. This is because each side will shoot the fuel from the two walls, so if our shooter was angled correctly, we could do so without much movement.

Analyzing the shot

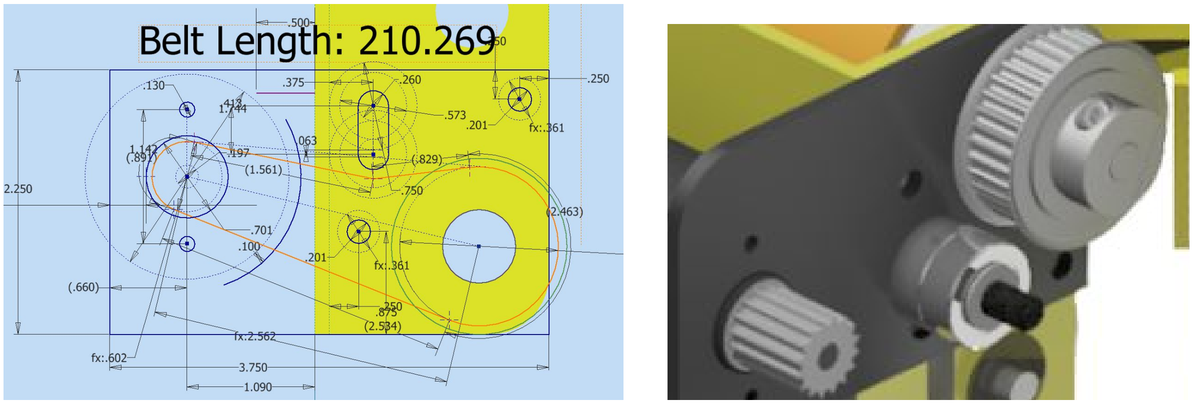
This year our shooter had to shoot lob-shots to get the balls into the opening of the high goal. By graphing different trajectories, we chose to shoot the balls at an angle of 16°.

The reason we chose 16° was because of the amount of error it provided. If the shot angle was too high, there would be little error allowed for the fuel to score (ex. At 8° there would only be ±2° of error allowed). If the shot angle was too low the shot wouldn't be able to make due to the angle of entry into the goal. By settling on 16°, we would have 5° of error in all.



Adjustability

We knew that implementing a pivot shooter required rigorous testing to make it accurate. That’s why we designed for adjustability for our belts, compression, and shooting angles.



Belt Tensioning geometry for our shooter.

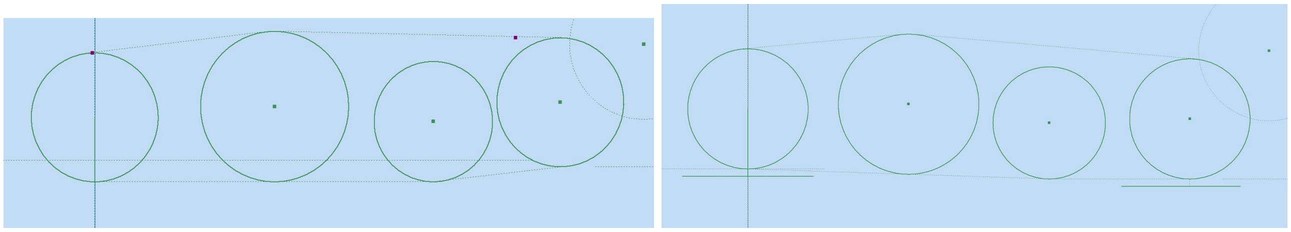
To power our rollers, we used a timing belt to transfer power from the motor to the driving pulley. Such a belt would need to be tensioned properly, so we added a sliding assembly with bearings to properly tension our timing belts.

To adjust for compression, we milled slots into the L-brackets that support the shooter. These slots allow the compression to be adjusted from no compression all the way up to 0.5” of compression. Because of the way the shooter supporting structure is designed, the shooter’s compression can be adjusted easily without disassembling any feature.

To adjust the angles on our pivoting structure, we used screws as stops for the pivoting pneumatic. To adjust the front/back angle, the shooter was designed to pivot on a pair of rod ends, which can also be easily adjusted.

Design Process

Of course, our shooter didn’t work perfectly straight out of assembly. When testing our first version, the right/left spread was quite big, much more than we had hoped. After doing more testing we narrowed the problem down to decompression. When the balls exited the shooter, they would “pop out” due to the decompression they were experiencing. Such an effect would send the ball far to the left or to the right, inconsistently.



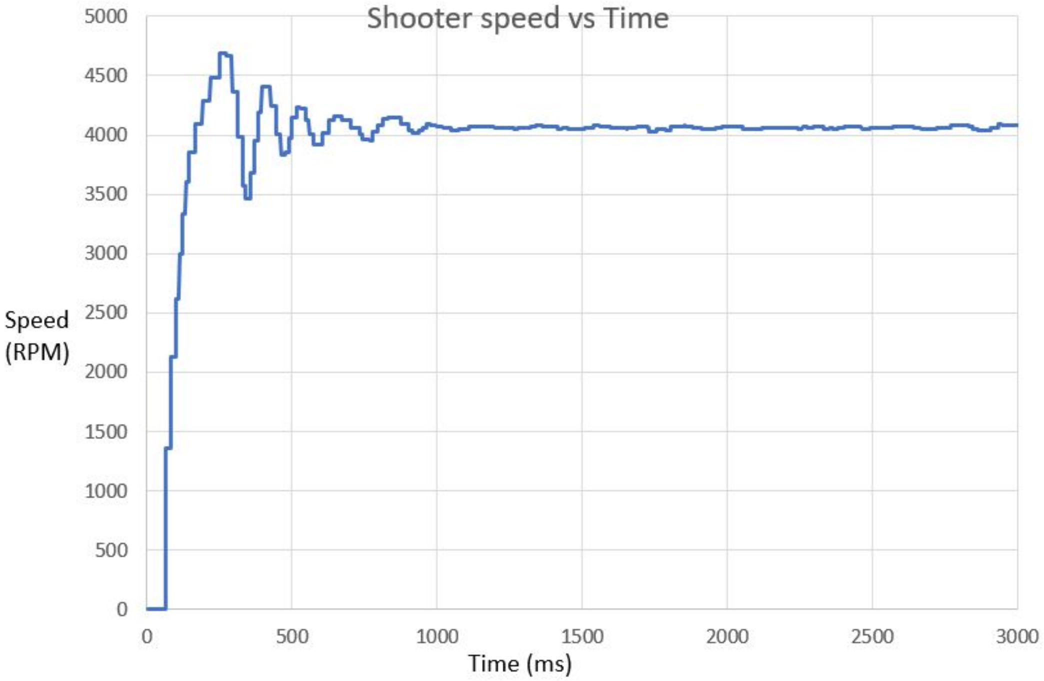
Original and Updated Roller profiles

Originally our shooter was designed to have a taper going in (left), to help facilitate the balls into entering the shooter. After realizing that decompression was a major issue, we redesigned the shooter to have a slight taper going out, as to help the ball slowly decompress over the course of the shooter. We also attached extra guiding fingers/rails that helped our shooter shoot straight.

Shooter Speed Control

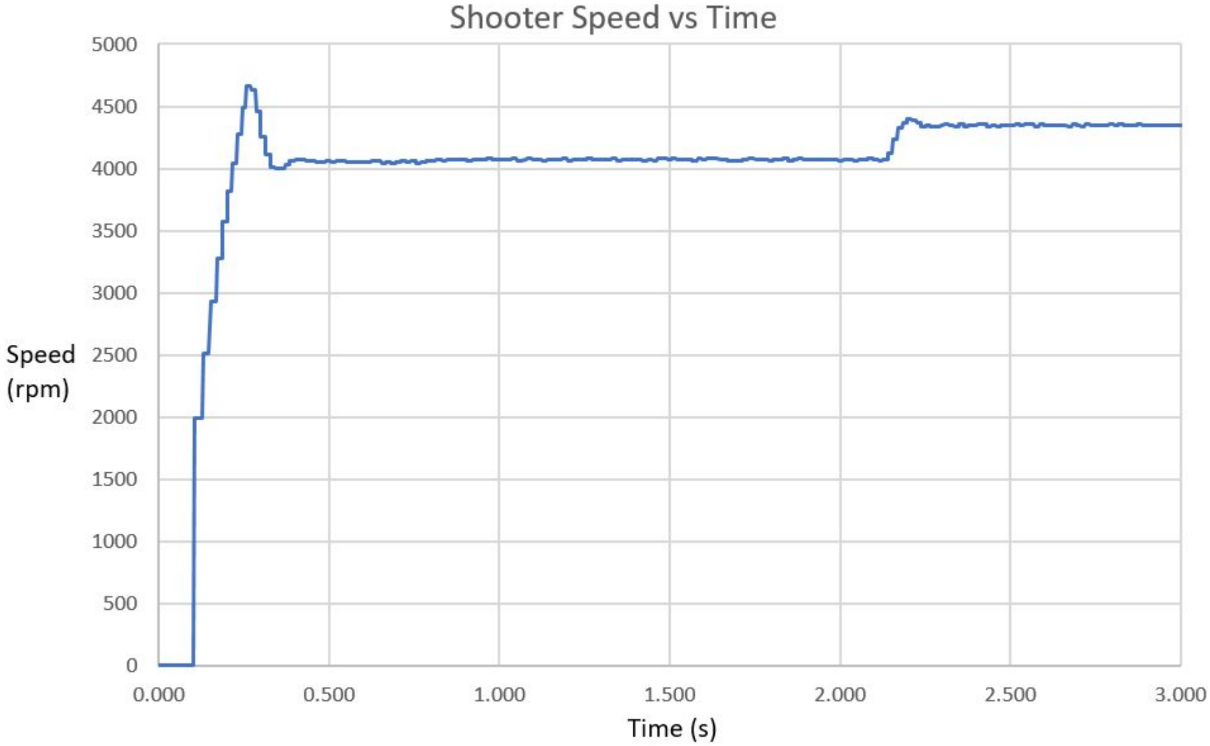
Philip Axelrod (Jr), Timothy Yang (Jr) and Shadaj Laddad (Jr)

Now that we had finished the hardware design, we moved onto designing the software to control the shooter. Using data from a hall effect sensor to detect a magnet attached the spinning rollers, we use a proportional feedback controller with a feedforward term to achieve our desired target velocity.



Ringing in the velocity of the shooter

Upon testing this controller, we noticed a severe issue. For the first second, the velocity oscillates about the target. This would affect the accuracy of our shot, as balls would leave the shooter at different speeds and thus travel different distances. To fix this issue, we reduced our proportional gain from applying 100% power per 1,500 rpm of error to a less aggressive 100% power per 3,000 rpm of error.

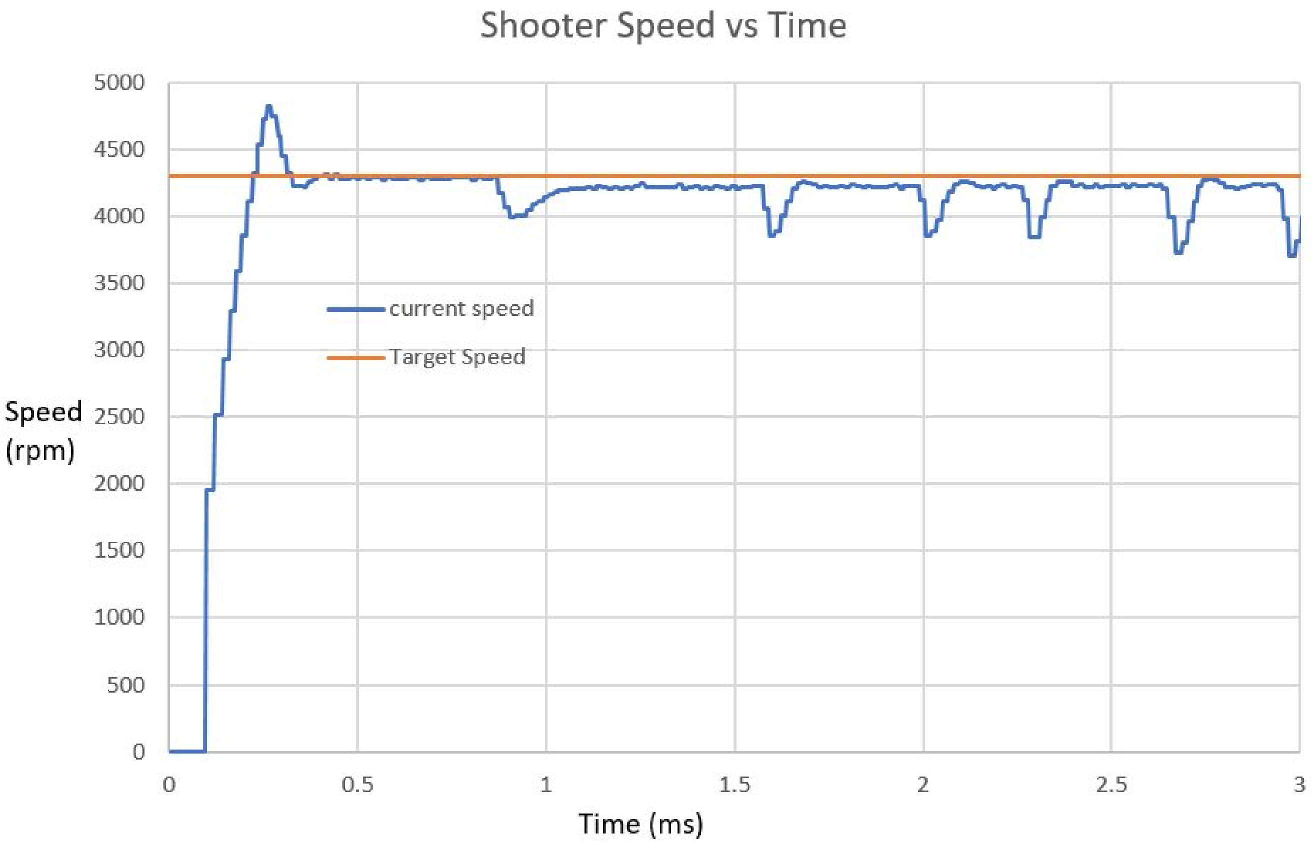


With a reduced gain, the settling time is improved to better than 200ms.

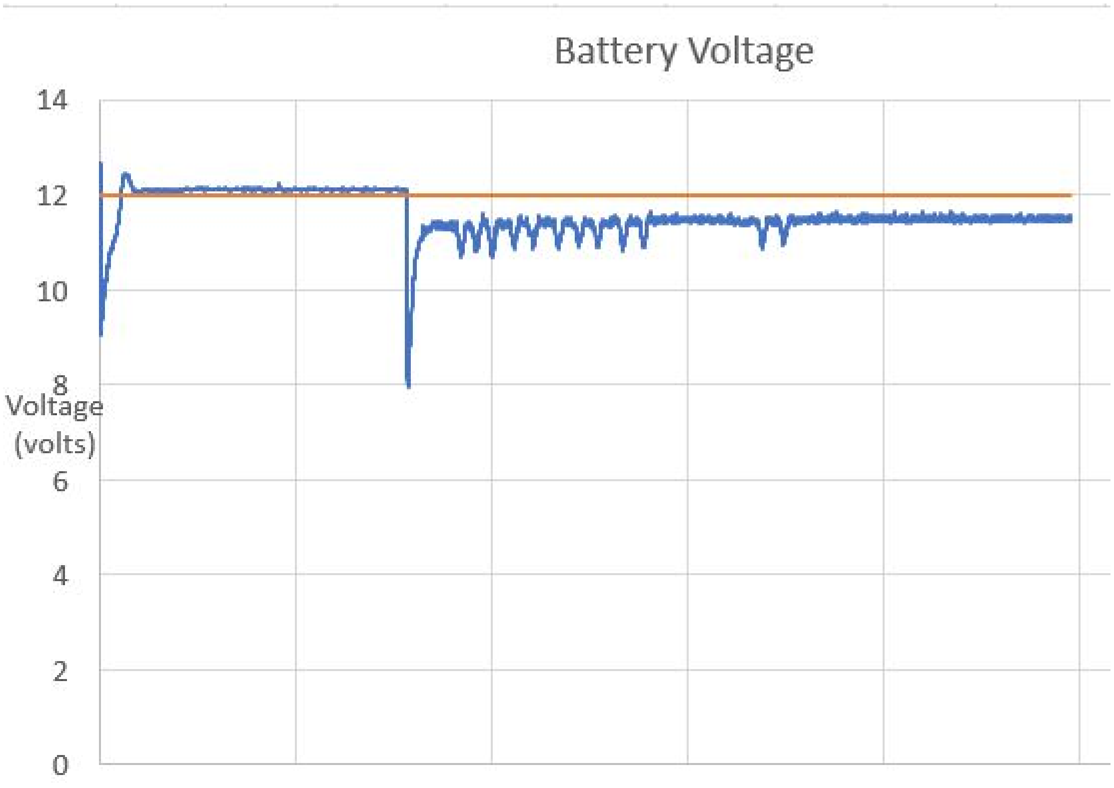
Reducing the gain produced dramatically better results, with oscillations lasting for less than 200ms. To simulate a ball traveling through the shooter, we increased the speed by the estimated velocity drop caused a ball being shot. This is reflected in the step in velocity at 2.1 seconds. This spikes lasts for less than ⅕ second, so that the shooter, at its maximum speed of balls at 4 times per second, would stabilize back to target speed in time for the next ball.

Compensating for Battery Voltage

Now that we had tested the shooter individually, we decided to run the shooter, agitator, and collector systems together to simulate the actual process of shooting. The drops in velocity at times 1.6, 2, and 2.3 and later correspond to the times that a ball pass through the shooter, and the speed settles before each shot. However, now we can see another problem.



At 0.8 seconds, the speed of the shooter drops below the target, and stabilizes about 100 rpm below the target. After this initial drop, the speed never reaches the target speed again. We hypothesized that the speed drop was being caused by a voltage drop when the robot's agitator and ball collector start running. Sure enough, we noticed a significant voltage drop around the time of the speed drop. After this initial drop, the battery voltage did not return to its previous value



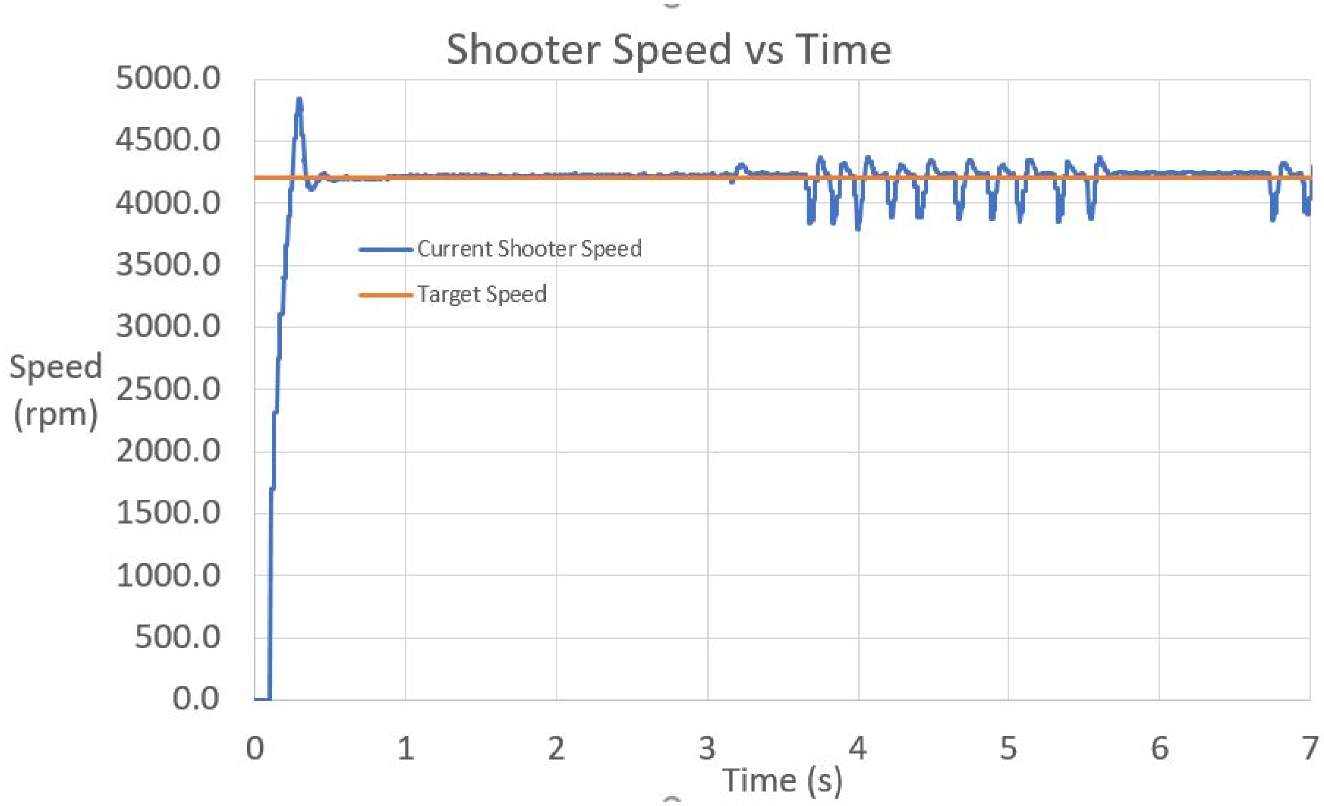
Untreated variations in the battery voltage affect the accuracy of the shooter, and compensation is required for a consistent shooting speed.

To compensate for this voltage drop, we simply adjust our motor controller pwm outputs with the following equation:

$$Out_{pwm} = Out \cdot \frac{V_{B-Nominal}}{V_{B-Measured}}$$

Thus if the battery voltage dropped by a factor of 80% from the nominal voltage, our pwm signal would get increased by a factor of 125%.

With the further addition of current limiting routines we were also eliminated the voltage drops as motor subsystems started. For systems with sensors, like the shooter, we use current limiting based on the current speed and calculated EMF of the motors. For sensorless systems like the ball collector, we use rate limiting on the pwm control signal to reduce the peak startup current.



After compensating for the battery voltage, and adding current limiting and gain adjustment, the speed is constant as other robot systems start and run, and the speed recovers quickly between shots.

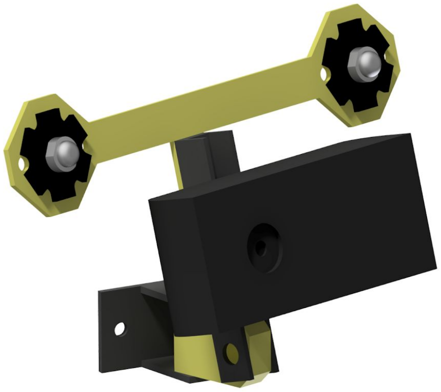
With both the battery compensation and the current limiting, we see now that the the speed remains at the target speed, even as the ball collector starts and runs, and the speed quickly recovers between shots.

This page intentionally blank.

Camera Mount

Gautam Rajesh (Junior)

To make the scoring of fuel faster, we decided to use a camera that points at the goal so that the robot can align itself to the boiler instead of manually aligning the robot and wasting precious time in the process.



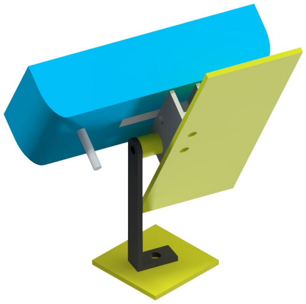
Design Requirements:

1. The Mount needs to be adjustable. Instead of guessing the angle of the camera during the design, the camera angle can be adjusted to maximize the accuracy of our shooter.
2. An overall rigid structure. Since the pivot points of the mount are only held with one screw, it is very possible that the angle of the mount could shift during the match, which would hinder the accuracy of the shooter.
3. The mount must be small. Since the mount is being placed at the edge of the robot. We needed to make sure that the mount didn't extend beyond the bumpers.

Design Process

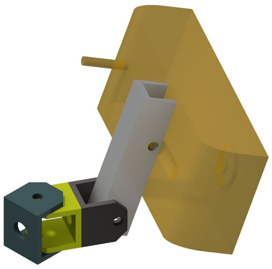
Rather than just buying a couple ball joints off the store, we decided to design our own adjusting mechanism. A big difference of our design compared with ball joints is that we have individual control of each degree of freedom. With ball joints, it is impossible to independently change the side-to-side and the up-to-down angles of the camera, making it much harder to find the optimal position for the camera. Having independent degrees of freedom gives us more control over the position of the camera, which in turn, increases the accuracy of our shooter.

Another problem we had to solve was the rigidity of the camera mount:



Top Left: In a previous design the pivot points were only connected by a small L-channel. This design proved to be too weak since each pivot point was only being supported on one side making it susceptible to losing its angle during a match.

Top Right: In our finalized design, we used C-channels for each degree of freedom. This gave the design more rigidity as each pivot was supported either side.



There are a couple things we did to save space. One was through the simplification of the pivoting structure. As shown above, the two degrees of freedom is achieved by 2 C-channels mounted back-to-back from each other. This proved to be too complicated and took up too much space. Instead of doing this, we combined both the axes onto one smaller solid block of aluminum, making the mount much smaller in size. Contrary to using LED rings, we decided to try and use 2 super-bright green LED’s. Unlike LED rings, which had to be placed around the camera lens, we could place these anywhere near the camera. This made the design of the mount much easier and much smaller, as we weren’t limited to one single location for the LED’s.

Collector

Jing-Chen Peng (Junior)

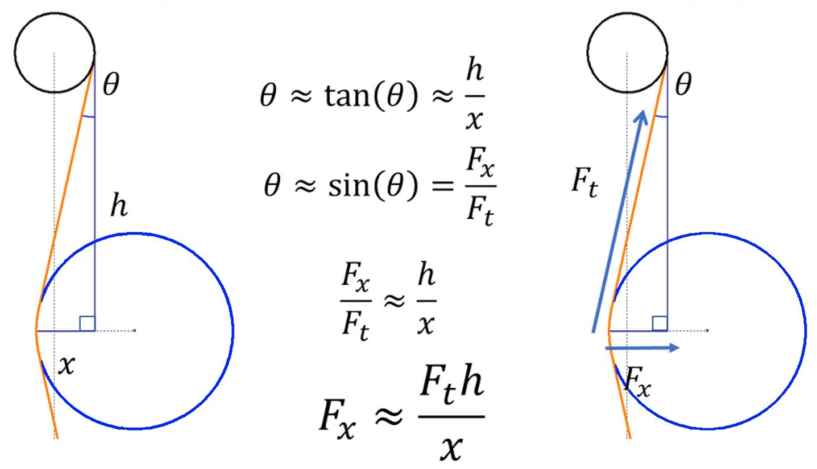
Our collector uses belts to move balls into the ball storage and feed balls from the ball storage into the shooter.

Design Requirements:

1. Collect multiple balls at once
2. Lift balls from ground into ball storage
3. Move balls from ball storage into shooter
4. Lift at the speed of four balls/second or greater

Lifting the Ball

Before we could start designing the collector in earnest, we had to ensure that the it was, indeed, possible to move balls upwards using belts. Using known quantities, we could calculate the lifting force on the ball and determine if our lift would work.



$$\theta \approx \tan(\theta) \approx \frac{h}{x}$$

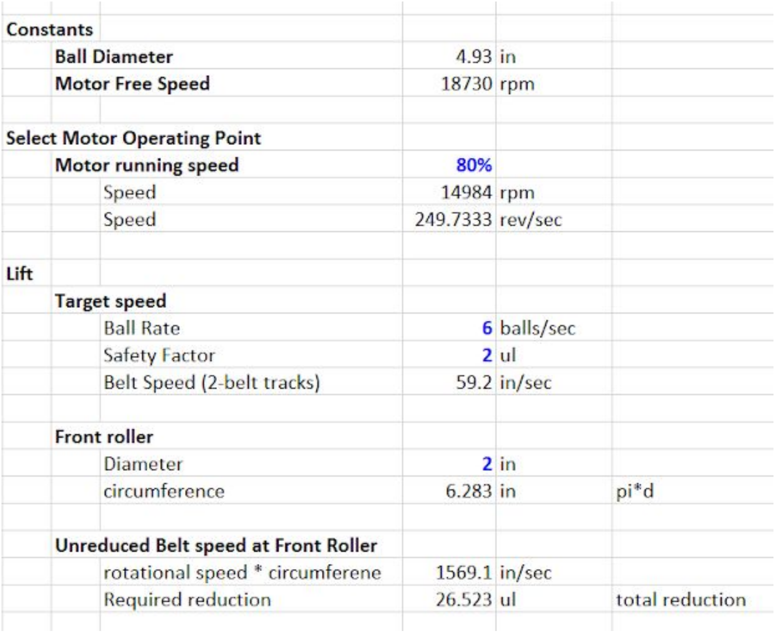
$$\theta \approx \sin(\theta) = \frac{F_x}{F_t}$$

$$\frac{F_x}{F_t} \approx \frac{h}{x}$$

$$F_x \approx \frac{F_t h}{x}$$

Using the small-angle approximation, we can derive a simple formula to calculate the pinch force on the ball from compression, belt tension, and belt length. Using this, we can then calculate the frictional force, or lifting force, between the belt and the ball.

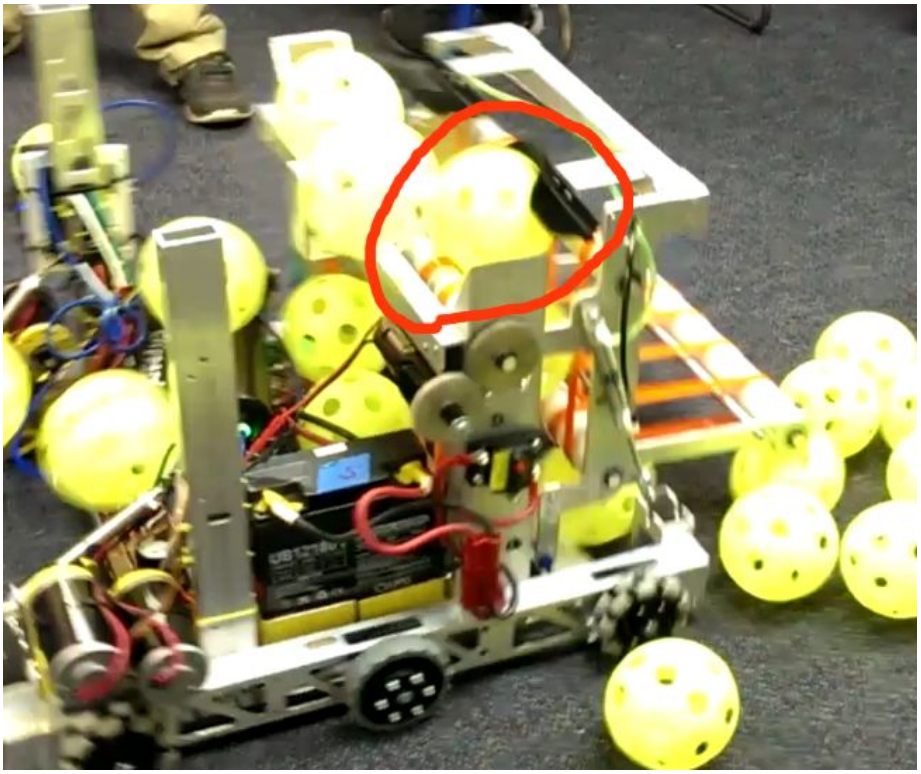
In addition to the force needed to lift the ball, we had to make sure that our collector can move balls fast enough to feed the shooter. Using our known target ball rate and motor operating speed, we could calculate the required reduction.



Constants					
	Ball Diameter		4.93	in	
	Motor Free Speed		18730	rpm	
Select Motor Operating Point					
	Motor running speed		**80%**		
		Speed	14984	rpm	
		Speed	249.7333	rev/sec	
Lift					
	Target speed				
		Ball Rate	**6**	balls/sec	
		Safety Factor	**2**	ul	
		Belt Speed (2-belt tracks)	59.2	in/sec	
	Front roller				
		Diameter	**2**	in	
		circumference	6.283	in	pi*d
	Unreduced Belt speed at Front Roller				
		rotational speed * circumferene	1569.1	in/sec	
		Required reduction	26.523	ul	total reduction

By relating the belts' desired surface speed with the motor's operating speed, we can calculate the required reduction.

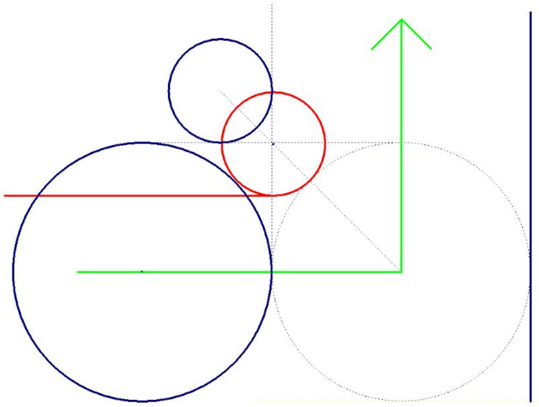
After balls are lifted off the ground, they must be deposited into the ball storage, located in the middle of our robot. To accomplish this, we originally planned to use a polycarbonate “hood” to redirect balls exiting the elevator towards the back of our robot. However, while prototyping, we discovered that placing flexible brushes at the exit of the elevator could accomplish the same thing. We chose to use brushes over a solid hood because they were simpler to design and machine, as well as being compact and easily replaceable.



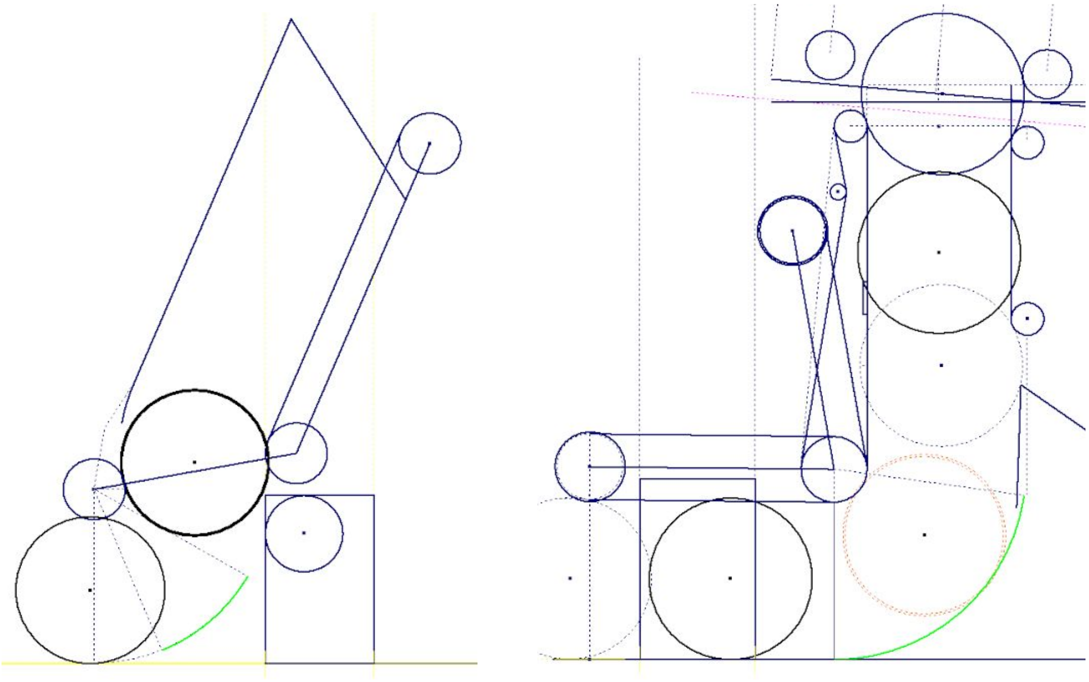
Our collector’s brushes in action. Balls being pushed up from the elevator are forced into the brushes and redirected towards the back of the robot, where the ball storage is located.

Design Process

During our design process, we had to decide whether to build a collector that rolled balls over our bumpers or one that moved balls through a gap in our bumpers. We came up with some basic sketches to show how each idea would work. Here, we realized that lifting the ball off the ground would require either high compression or a ramp.



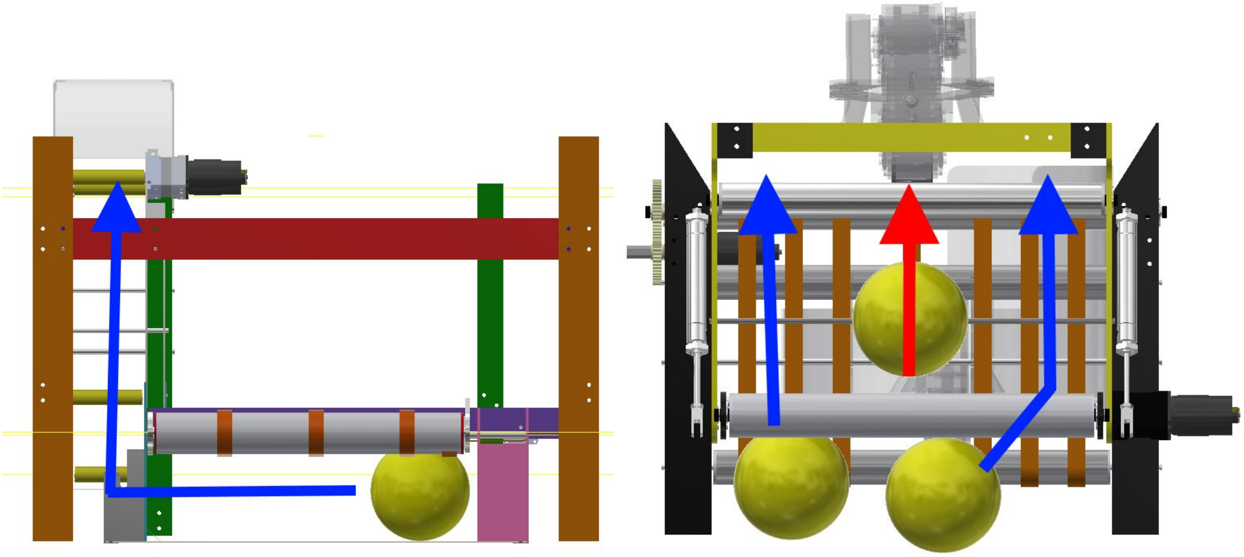
When collecting the ball, the roller must be high enough to let the ball pass under it, but low enough to pick the ball up. These are two conflicting constraints.



Left: One solution to this is to add a ramp. However, for over the bumper collection, this requires an additional mechanism to be deployed.
Right: Instead, we chose to collect balls through the bumper, allowing our collector to use a stationary ramp that does not need to be deployed.

Iteration

After deciding to collect balls through our bumpers, we had to decide how to move the ball from the ground to the top of our robot and into our ball storage. We considered two separate designs: One where balls would be swept to the side of the robot then picked up, and one where balls picked up would be carried upwards directly. We chose the latter design because it allowed the collector's belts to be used to collect balls and move balls from the ball storage to the shooter, saving space and design effort.



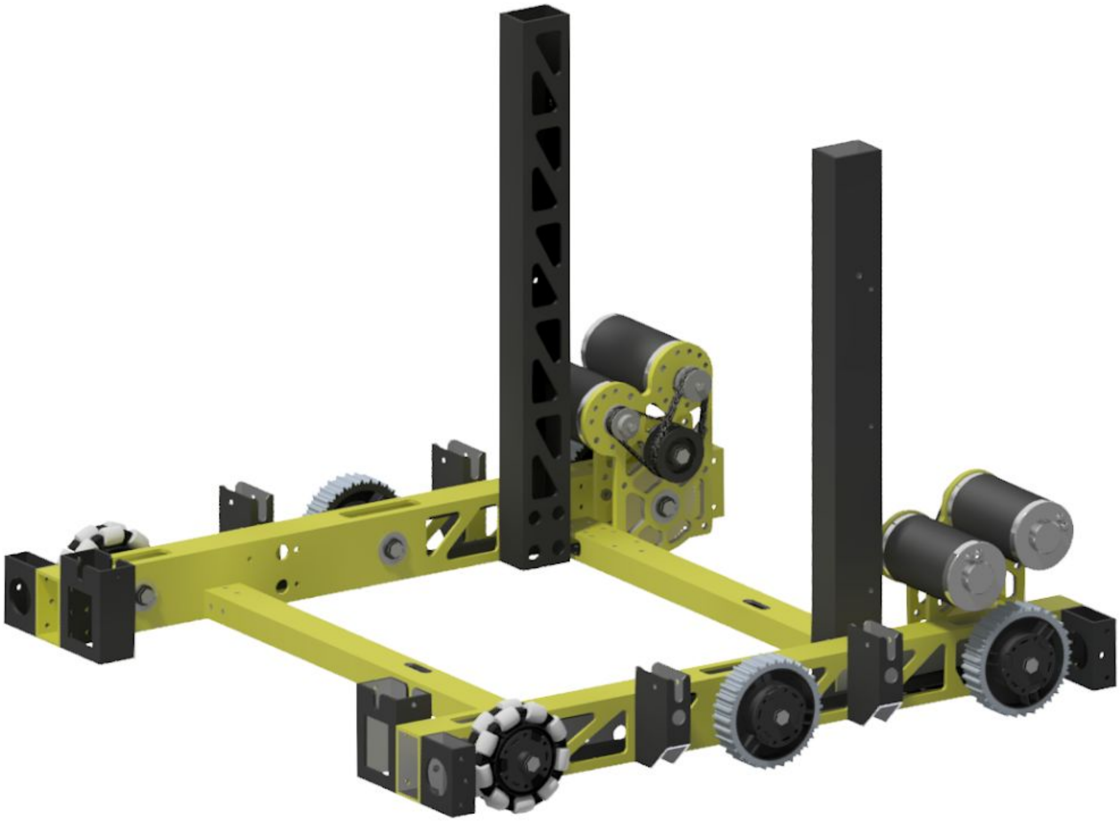
Left: Our initial collector design. It first moves balls into the robot, then pushes them sideways and finally up an elevator into the ball storage.

Right: In our final design, balls are collected upwards with belts into the ball storage (blue arrows), and balls coming out of the storage are simultaneously pushed up into the shooter (red arrow).

So far, the collector has shown that it can handle collecting from the ground and feeding the shooter in our tests. We're excited to see it in action during competition!

Drivetrain

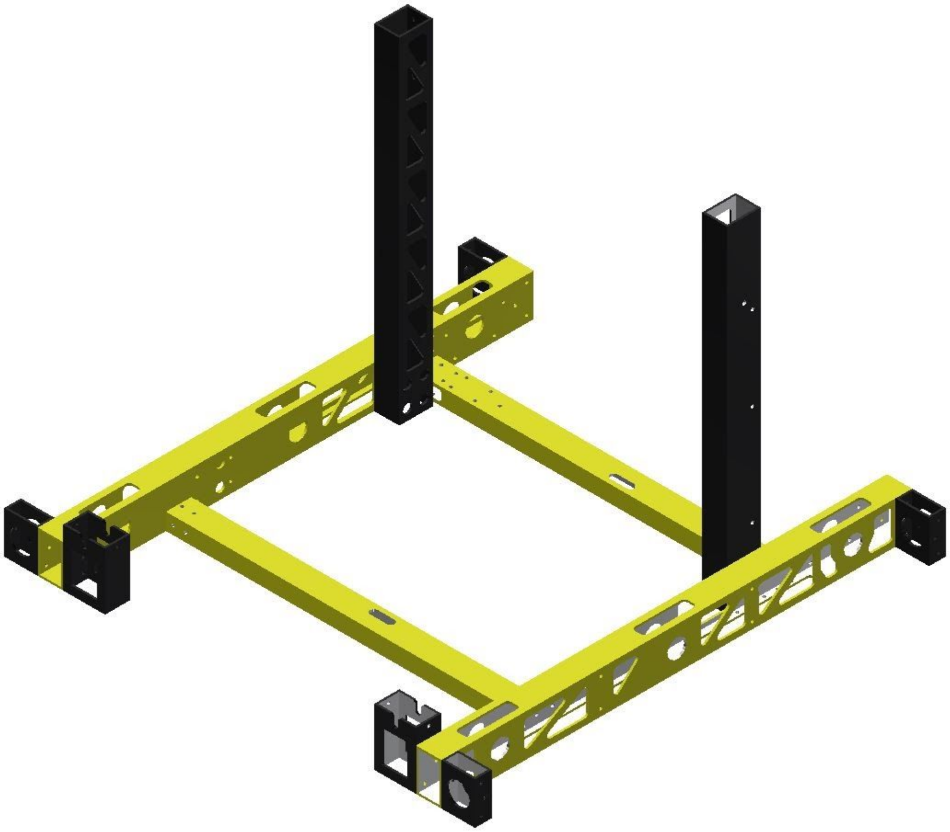
Ria Pradeep (Senior), Anika Singh (Junior), Eesha Deepak (Sophomore), Divya Pereira (Freshman), Priyanka Pereira (Freshman)



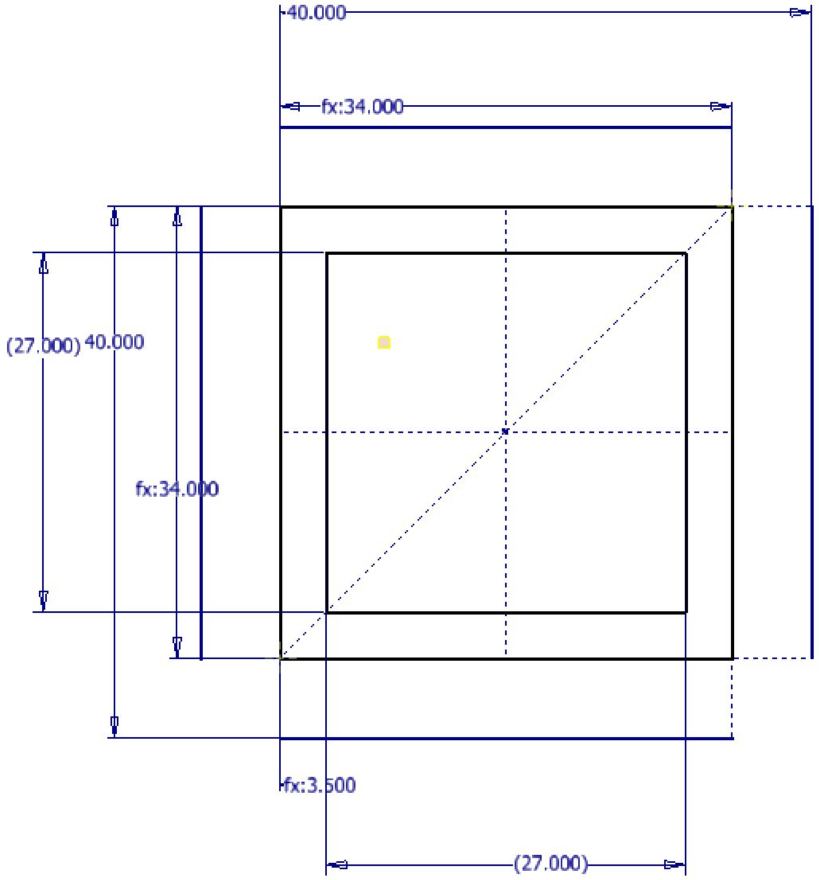
Design Requirements:

1. Drive quickly around the field (ideally at a speed of 15 feet per second).
2. Turn quickly.
3. Not drive over or get stuck on fuel or gears.

Drivetrain



While considering the robot dimensions with respect to the maximum robot volume, we realized that by keeping our frame square, we could extend systems on all four sides. This way, we can extend a collector forward, gear mechanism backwards, and anything else on the sides. We decided to go with the square base, as it gave flexibility to other systems.

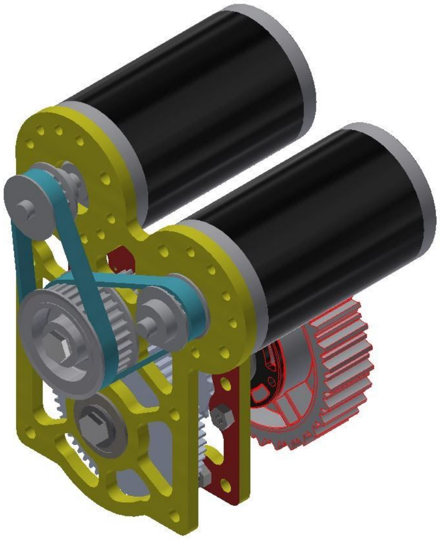


The black lines indicate the outline of the bumpers, and the blue lines indicate the extent to which other systems can extend. This particular sketch has a base of 34” x 34”, including bumpers.

Additionally, we kept our chassis low to the ground to avoid driving over fuel or gear. This way, we don’t get jammed over game pieces.

As we began the season, we wanted this drivetrain to be able to turn quickly. Therefore, we decided to go with a six-wheel layout, where the front or back wheels are omni-wheels while the middle is a traction wheel. We also added a slight center drop to assist with turning.

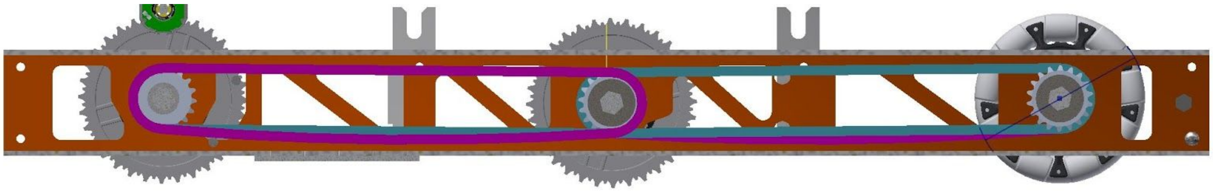
Drive Gearbox



For the past three years, we have designed drive gearboxes with motors that hang over the wheels to maximize space inside the robot for electronics. To continue with that feature this year, we had to lift the motors above the frame with a belt drive from the motors to the first stage reduction.

to have adjustability in tensioning, we created a pattern of mounting holes for the motors. Each set corresponds to a center to center distance incremented from the nominal distance. This pattern allows for adjustability in tensioning the belt.

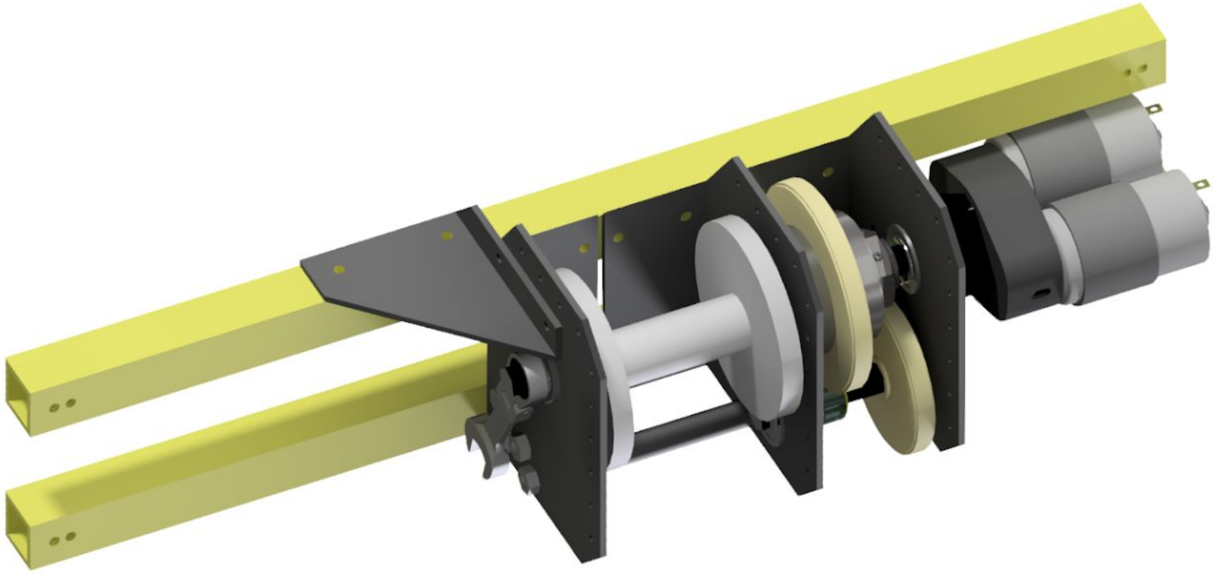
To save space inside the robot, our team has started putting our chain runs inside the drivetrain tubes. We’ve done this since last year, with great success. By doing so, other systems could mount directly to the inside of the side frame members.



Cross-section view of the inside of drivetrain tubes.

Climber

Andrew Ng (Sophomore), Atul Nair (Sophomore)

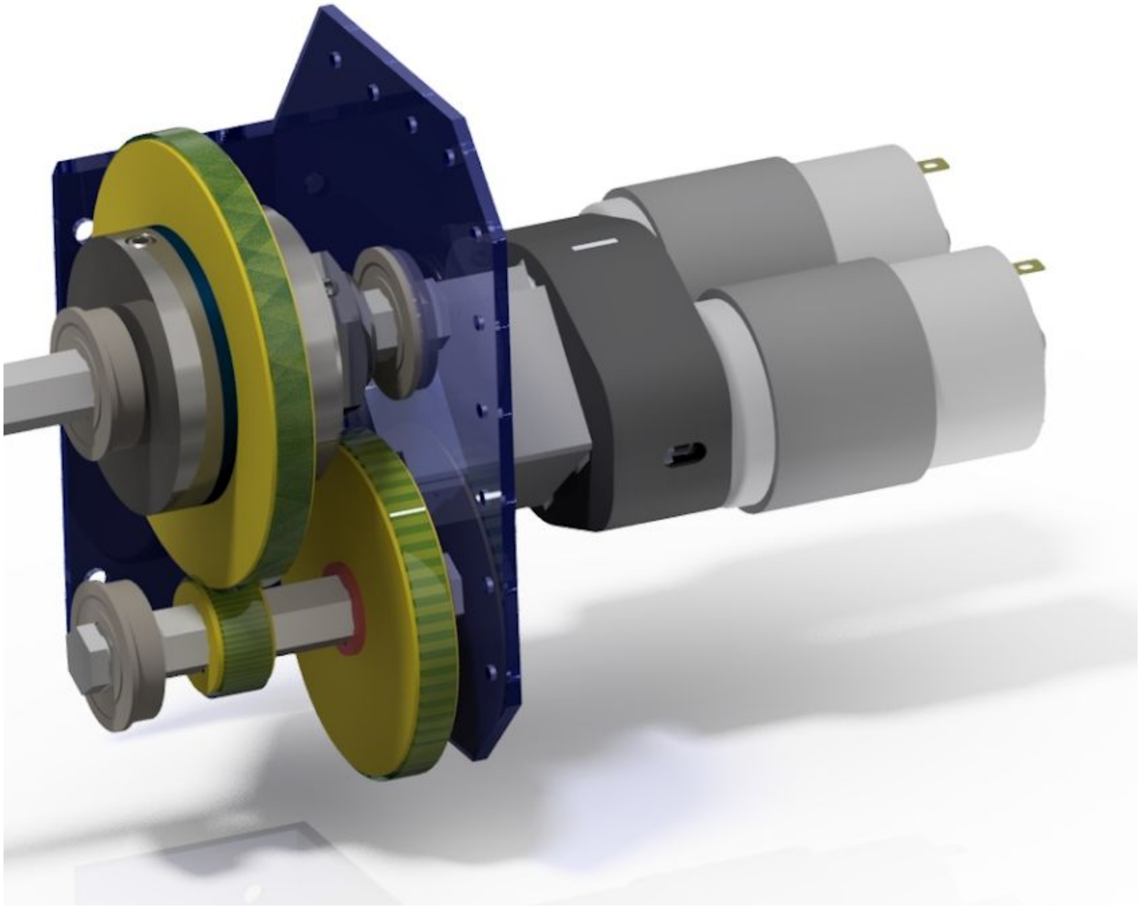


Our climber grabs the rope and pulls our robot up the climb in under five seconds. The climber uses velcro on the drum to stick to and control the rope.

Design Requirements:

1. Grab and control the rope in under 10 seconds
2. Climb the rope in under 5 seconds
3. Hold our position without power once we reach the top

Gearbox



When designing the climber, we realized that a lot of force would be exerted on the robot and the davit when we reached the top of our climb, since the robot would come to a sudden stop, and the motors would develop as much as 5 times more torque. This increase in torque could strip the planetary gearbox we were using or break the field.

We adjusted the torque limiter so that we could lift the robot with an extra 60 pounds of force, but loose enough to slip when the motors are still spinning at the end of our climb. To further protect our planetary gearbox, we added an intermediary stage to put the maximum torque on the planetary gearbox to be well within its load rating. This allowed us to safely use a single one planetary driven by two motors.

Prototype

to figure out how the capture the rope, we built a mockup of the robot out of wood, mounted a temporary climber, and used human weight to simulate the weight of the robot.

Sophomores Shreyas Mohidekar (Left), Andrew Ng (Center), and Atul Nair (Right) test the velcro winch on a mockup of the robot.

We began by looking at looking at a variety of designs to capture the rope, many of which were very complicated. Then, we found that velcro sticks to nylon rope very well, so we began experimenting with velcro.

The first issue we ran into was getting slack into the rope since the velcro cannot lift a load right after sticking to the velcro. However, we noticed by wrapping the rope a few times around the winch, a large force could be applied to keep the robot on the rope.

$$T_{load} = T_{hold} \times e^{\mu\theta}$$

$$T_{load} = T_{hold} \times e^{\left(\frac{\mu}{2\pi}\right)^{n}}$$

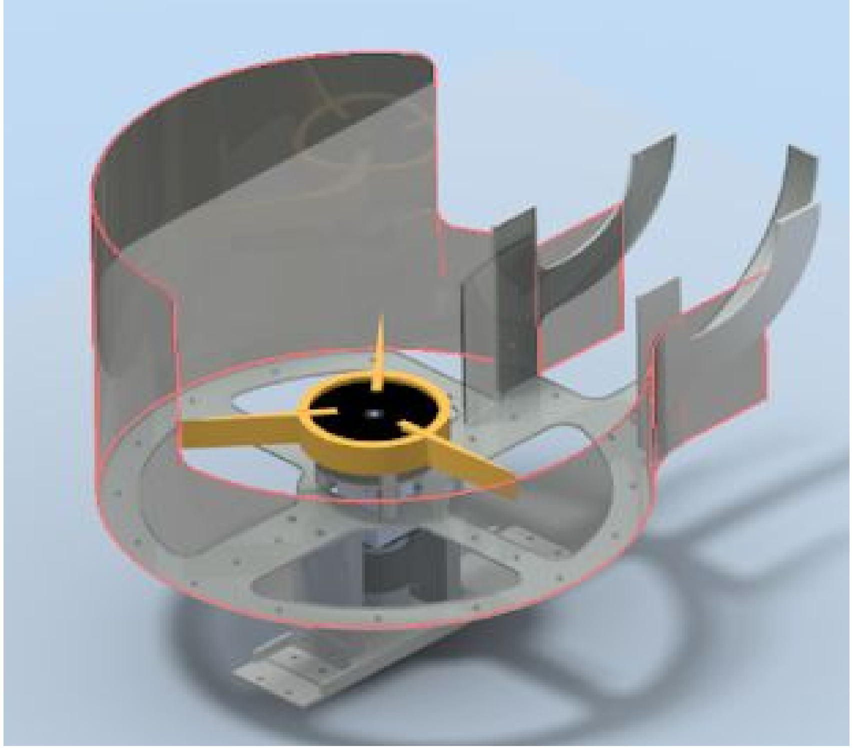
For every wrap around the drum, the maximum load the climber can support increases exponentially, so we estimated that two wraps should be plenty. By tying a slip knot in our rope, we allow our climber to wrap the rope around itself twice with no tension in the rope. This increase in wrap helps the climber hold on to the rope as the robot ascends.

Our climber has been consistent through our tests and practice runs, and we look forward to using it in competition.

Ball Storage

Zachary Wu (Freshman), Jonah Soong (Freshman), Shaunak Bhandarkar (Sophomore)

The Ball Storage is a cylindrical shaped container connected to the chassis. It can store up to 10 balls and ejects balls into the elevator connecting to the shooter.



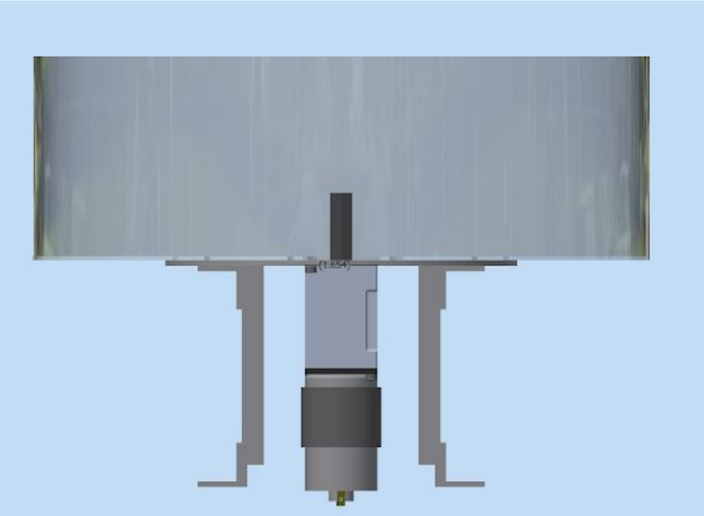
During our initial discussions, we decided that the best strategy was to shoot fuel into the high goal, opposed to dumping balls into the low goal. Our goal was to find the most efficient way to use the space we had to store balls, and be able to eject balls at the same speed as the shoot rate.

Design Requirements

1. Ability to store 10 balls
2. Able to feed balls into the shooter at the shooters max shooting rate
3. Ability to receive balls from the hopper

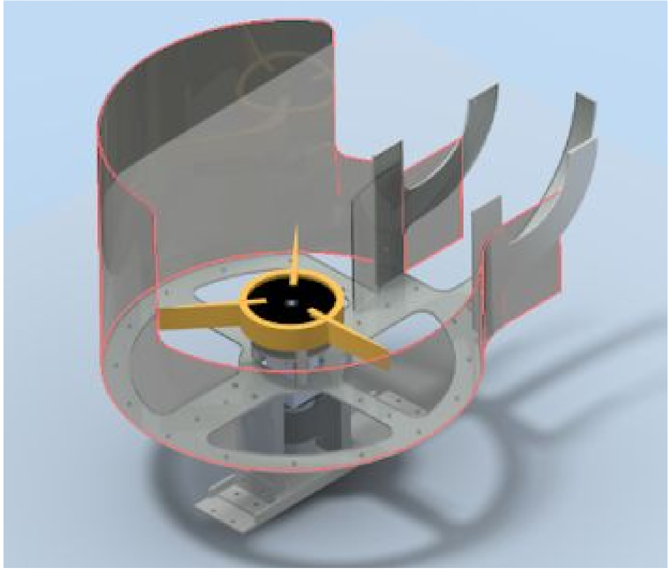
Changing The Storage

Our first design was a cylinder about 15 inches tall and 14.5 inches in outer diameter (OD). We predicted that we could hold 18 balls. However, we ended up shortening the height to only 5 inches tall to accommodate other subsystems. The motor and gearbox would be mounted right under the ball storage.



The 5 in version

After a few tests, we realized the most ideal way to hit the ball was in the middle to decrease the necessary torque. To make this change and to account for the length of the hex shaft in the motor gearbox, a mere 1.6 inches, we decided to move the gearbox partially above the bottom plate, as well as adding spacers and plates to raise the entire assembly.

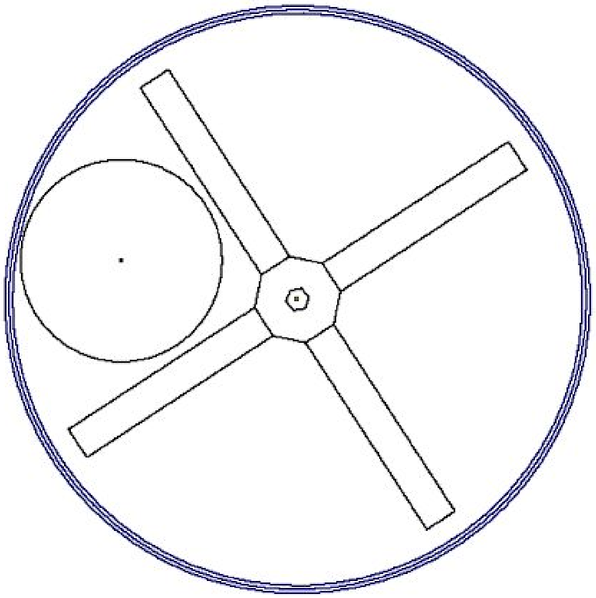


After gearbox was moved through bottom plate.

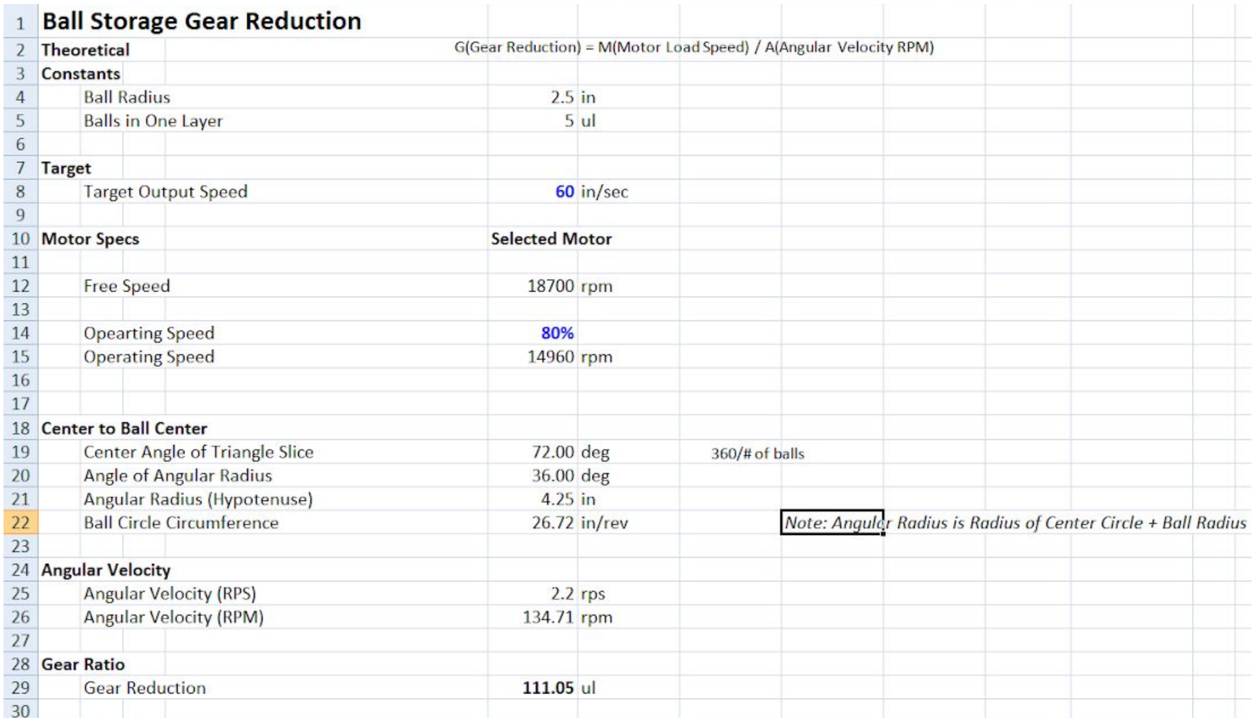
For the final version, we were able to move the shooter forward, allowing us to increase the size of our storage by at least one layer of balls. We also realized that we had no way of collecting balls from the hopper, so we cut out part of the outside polycarbonate wall to add a funnel. We also added an elevator-pipe to the shooter to accommodate for the changing design of the collector. The ball storage to shooter elevator was integrated into the collector to help save space.

Calculating the Gear Ratio

As this was the first year for us, we had to learn how to calculate the gear reduction ratio.



Our original idea of what our ball storage would be like(Top view)



1	**Ball Storage Gear Reduction**				
2	**Theoretical**		G(Gear Reduction) = M(Motor Load Speed) / A(Angular Velocity RPM)		
3	**Constants**				
4		Ball Radius	2.5	in	
5		Balls in One Layer	5	ul	
6					
7	**Target**				
8		Target Output Speed	**60**	in/sec	
9					
10	**Motor Specs**		**Selected Motor**		
11					
12		Free Speed	18700	rpm	
13					
14		Opearting Speed	**80%**		
15		Operating Speed	14960	rpm	
16					
17					
18	**Center to Ball Center**				
19		Center Angle of Triangle Slice	72.00	deg	360/# of balls
20		Angle of Angular Radius	36.00	deg	
21		Angular Radius (Hypotenuse)	4.25	in	
22		Ball Circle Circumference	26.72	in/rev	*Note: Angular Radius is Radius of Center Circle + Ball Radius*
23					
24	**Angular Velocity**				
25		Angular Velocity (RPS)	2.2	rps	
26		Angular Velocity (RPM)	134.71	rpm	
27					
28	**Gear Ratio**				
29		Gear Reduction	**111.05**	ul	
30					

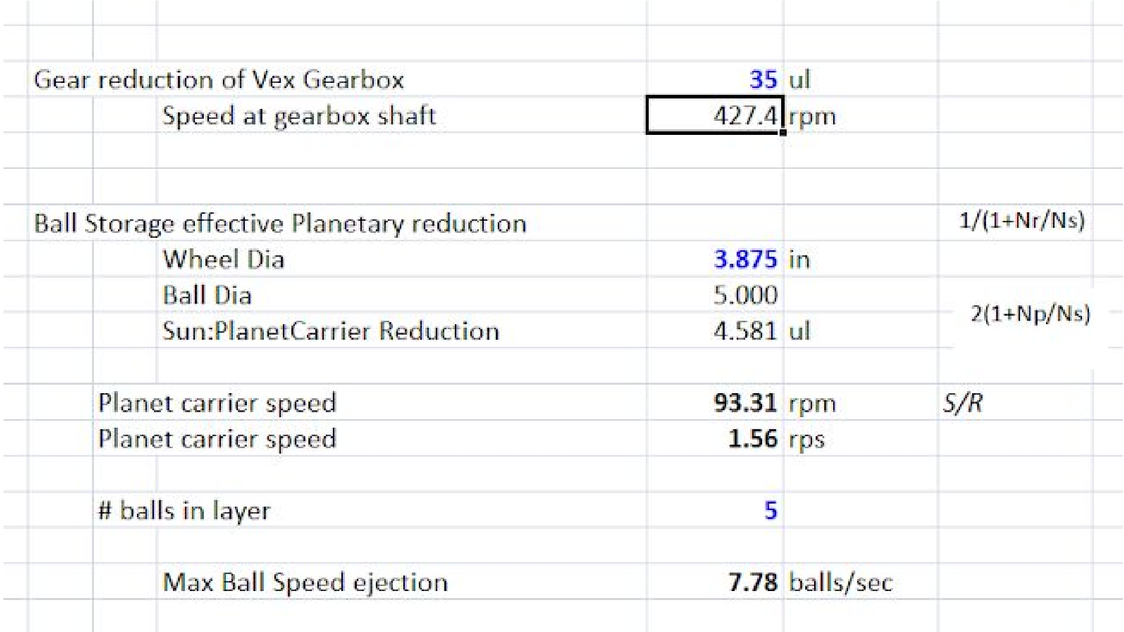
Our original calculations were based off the fact that we would be pushing the ball.

After prototyping with such a high gear reduction though, we realized we needed more speed to match the max shoot rate of the shooter. Our solution to this was to lower the gear reduction, so that the agitator would be faster. This not only allowed us to eject balls at the same speed as the shooter can shoot, but brought us out of the fragile gear reduction ratios.

Changing the Ratio

After running several tests, we realized our ratio was wrong because we assumed that the veins were pushing the balls, making it a 1:1 ratio for turning and moving. This however was not the case, through the testing we realized that a more accurate depiction was a planetary gearbox. That meant there would be a ratio between how much the agitator and balls moved. After prototyping with some different gear reduction ratios, we decided to use 35 as it could keep up with the shooter.

Checking



Gear reduction of Vex Gearbox	35	ul	
Speed at gearbox shaft	427.4	rpm	
Ball Storage effective Planetary reduction			1/(1+Nr/Ns)
Wheel Dia	3.875	in	
Ball Dia	5.000		2(1+Np/Ns)
Sun:PlanetCarrier Reduction	4.581	ul	
Planet carrier speed	**93.31**	rpm	*S/R*
Planet carrier speed	**1.56**	rps	
# balls in layer	5		
Max Ball Speed ejection	**7.78**	balls/sec	

Using the formula to find the sun:planet reduction, we found that it was 4.5:1. Using that reduction, we found that our maximum speed ejection was around 8 balls a second, when conditions were most ideal. This ejection rate gave us some slack just in case there was any jamming.

Agitator

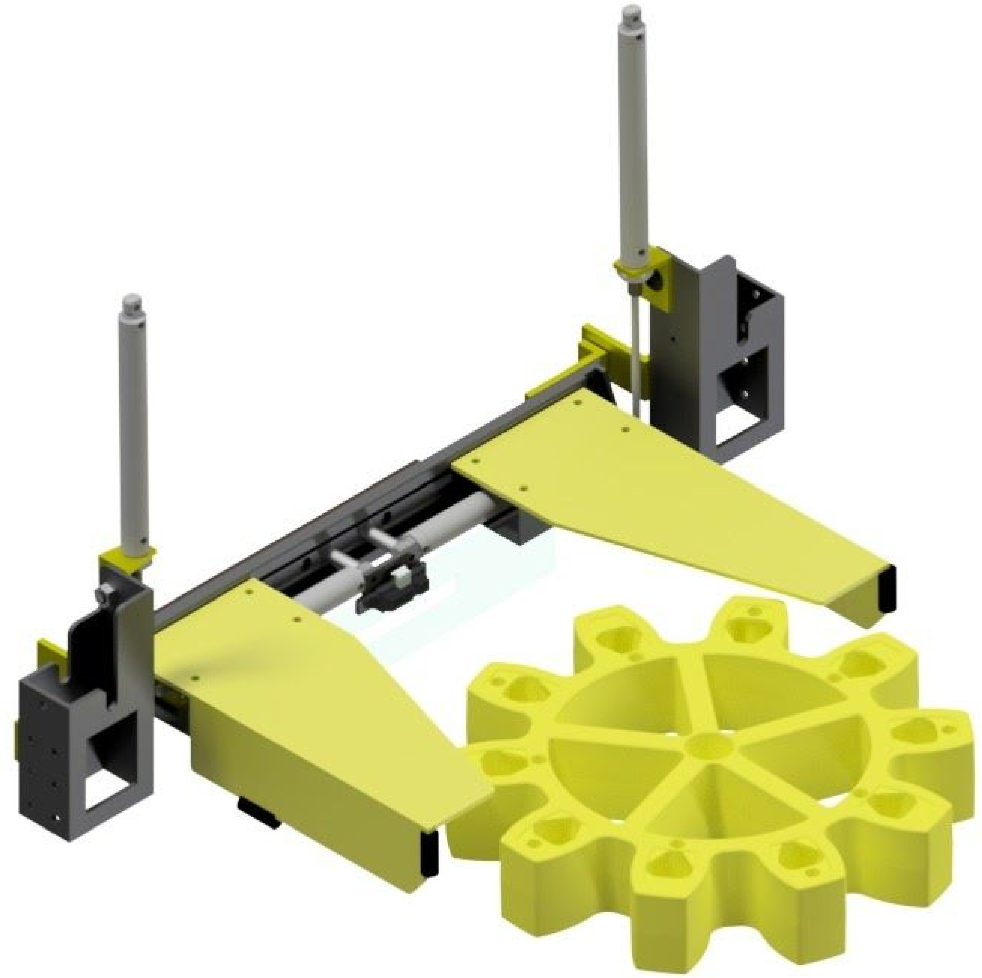
Our first agitator design was a wheel with flaps, but was not meeting our requirement. We tested a cone-shaped agitator for our second design. Our agitator was more than capable of ejecting four to five balls a second but it was not consistent. Our solution: a divider so the only path was out to the shooter. This agitator was perfect except the agitator was too high off the base. We decided to stick to an idea proposed from the very start, a wheel with flaps stuck to it to slap the balls. We increased the speed and stiffened and increased the number of flaps. The second design worked very well and was chosen for the purpose.

Final Design

Our final design is like our original plan: a cylinder that is about 5 inches tall and 14 inches in diameter. It has a hole 5 inches wide with a curved track which leads the balls out accurately. An agitator lies in the middle to push balls out. A divider lies in the middle to prevent balls from just going a circle. There is also a polycarbonate ramp to allow for collection of balls through the human player stations.

Gear Mechanism

Arthur Zhang (Junior), Edward Xie (Freshman)

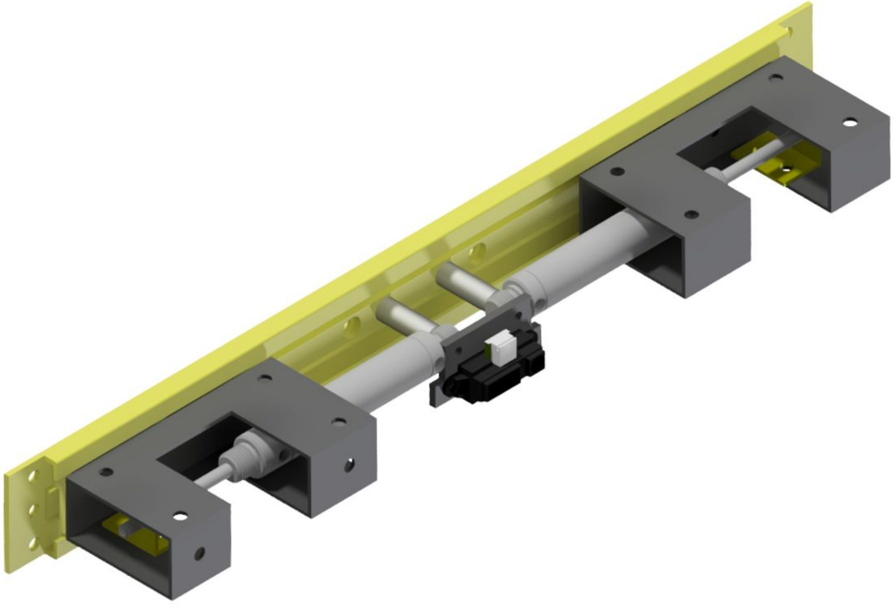


Design Requirements:

We set a few requirements for the Gear Mechanism:

1. Ground pickup
2. Wide range of gear collection
3. Securely holds gear to prevent being knocked out
4. Release gear quickly onto lift w/out pilot interaction
5. Use minimal amount volume in robot for the delivery

Linear Actuator and Clamps

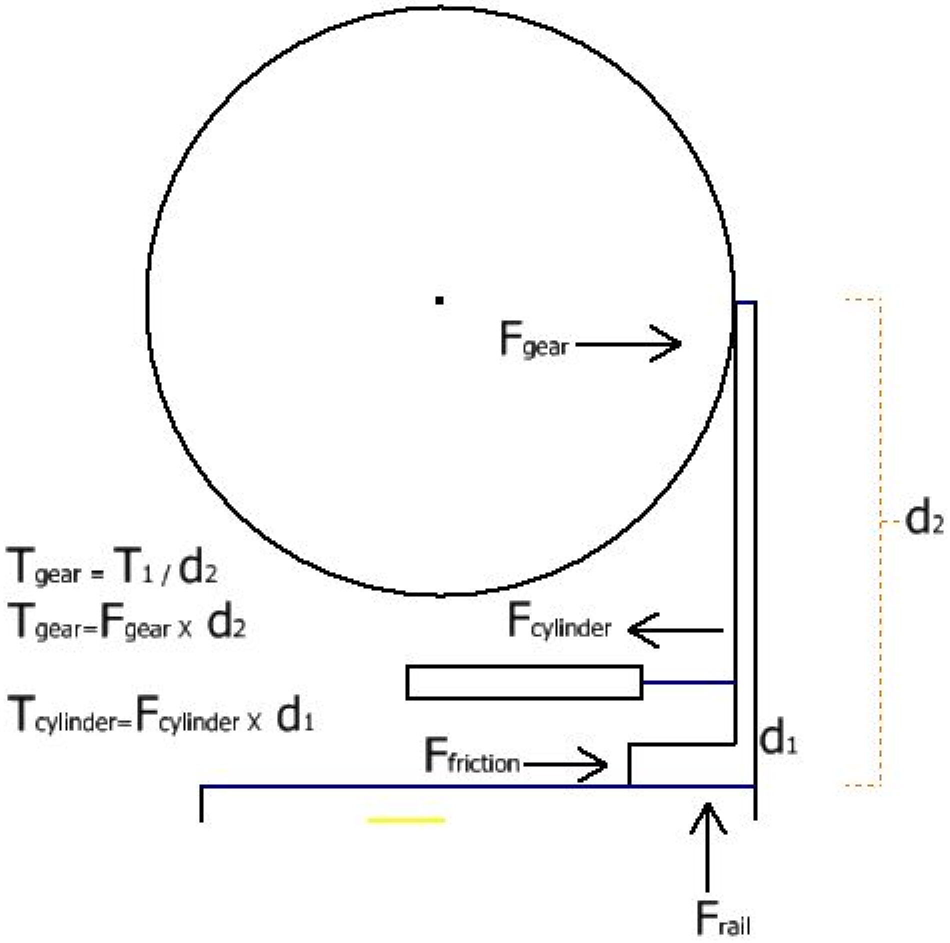


We chose to use two 9/16” bore pneumatic cylinders to actuate our system. The pneumatics each provide 13.4 lb of force, but we preferred a more secure grip on the gear.

To better grip the gear, we needed to find a material with a high coefficient of friction with the gear. To do so, we derived the formula below for calculating the coefficient of friction.

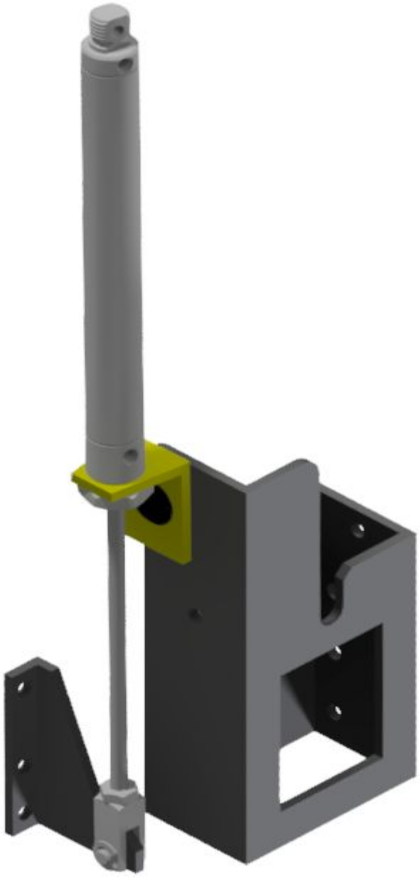
$$\mu = \tan \theta$$

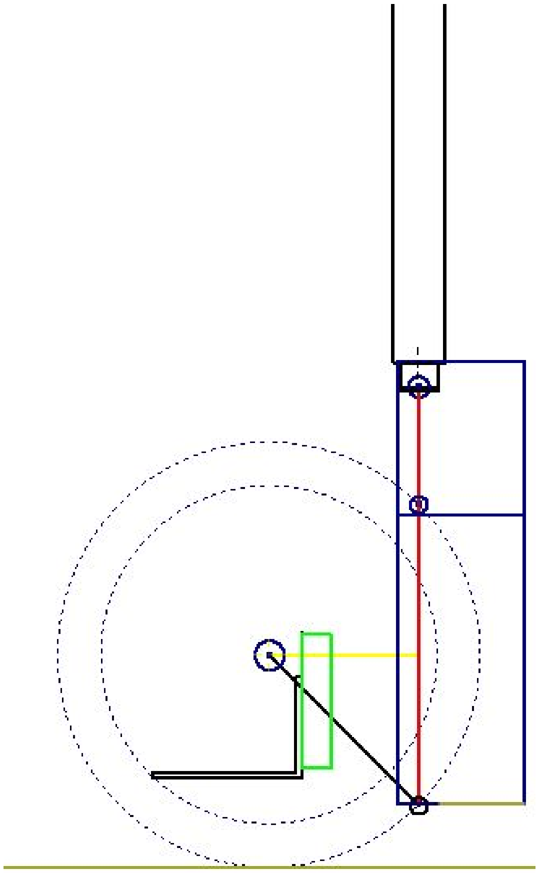
Then, we found the coefficient of friction by placing the gear on different material surfaces and gradually increasing the angle of the gear from the ground. We measured the angle at which the gear began to slide to calculate the coefficient of friction. After multiple rounds of testing, we tested that silicone matting was the best surface, providing a high coefficient of friction of 2.7 with the gear.



The gear is relatively far away from the cylinders, which lessens the force on the gear. So even with this extremely high coefficient of friction, we still needed to calculate the amount of force on the gear to ensure that it would be securely clamped. To do so, we analyzed the system of forces acting on the gear. By doing this, we could guarantee that our gear mechanism would securely grip the gear without the need for multiple iterations.

Lift Mechanism





$$\vec{T} = \vec{R} \times \vec{F}$$

$$T = F R \cos \theta$$

We used the equations above the calculate the amount of torque needed to lift up our gear mechanism. By calculating the force, torque, and energy needed in our pneumatic cylinders, we could create a working gear mechanism from solely our calculations rather than relying on iterating multiple times with multiple cylinders.

While testing the gear mechanism, we discovered that the gear would sometimes roll into our gear mechanism. To expand this range of gear collection, we added delrin rods to the front of the gear collector to prevent the gear from being caught in the gear mechanism.

All in all, our gear mechanism surpassed our expectations because it quickly and securely grips the gear over a wider area than expected, satisfying our design requirements and more.

Electrical

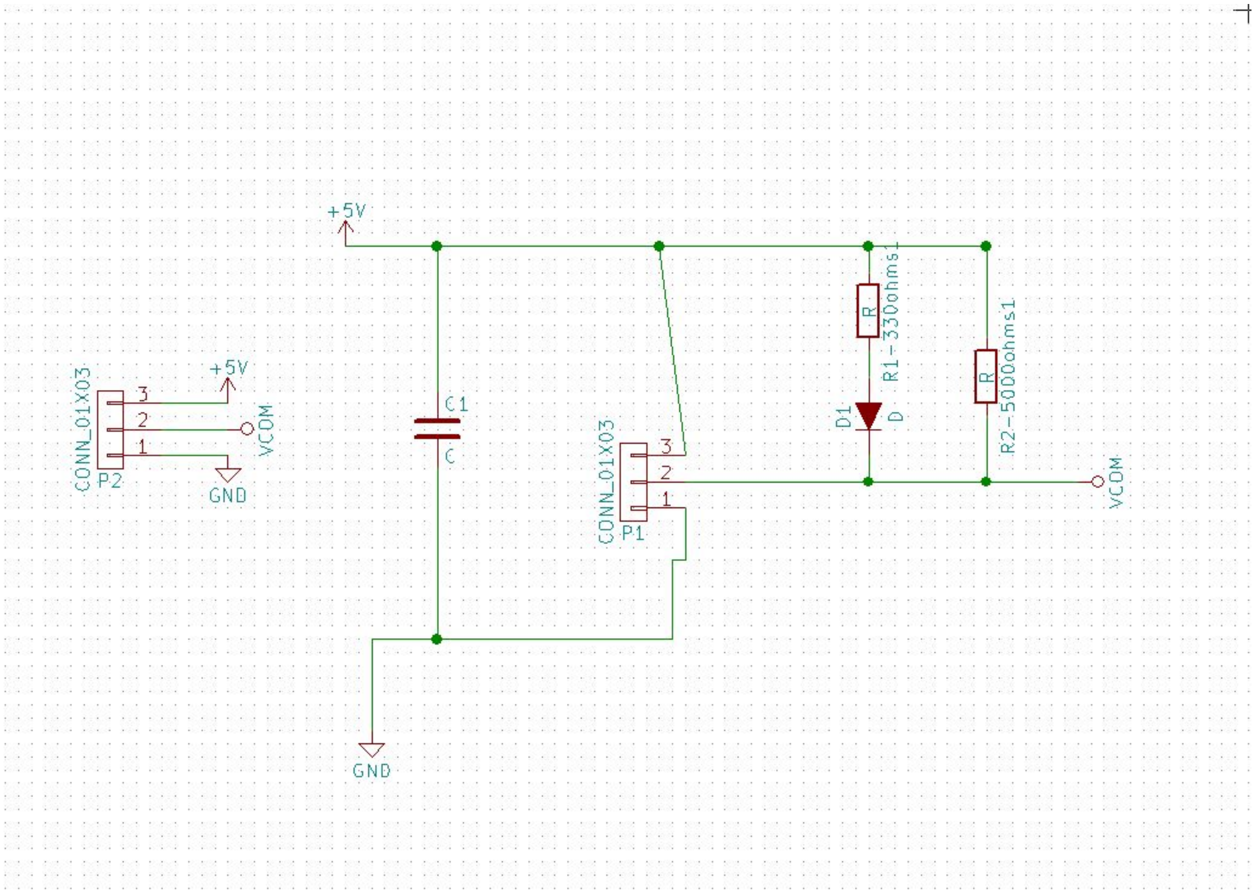
Shikhar Jagadeesh (Senior), Nathan Chen (Senior), Sean Chen (Junior), Ravi Varma (Junior)

Printed Circuit Boards

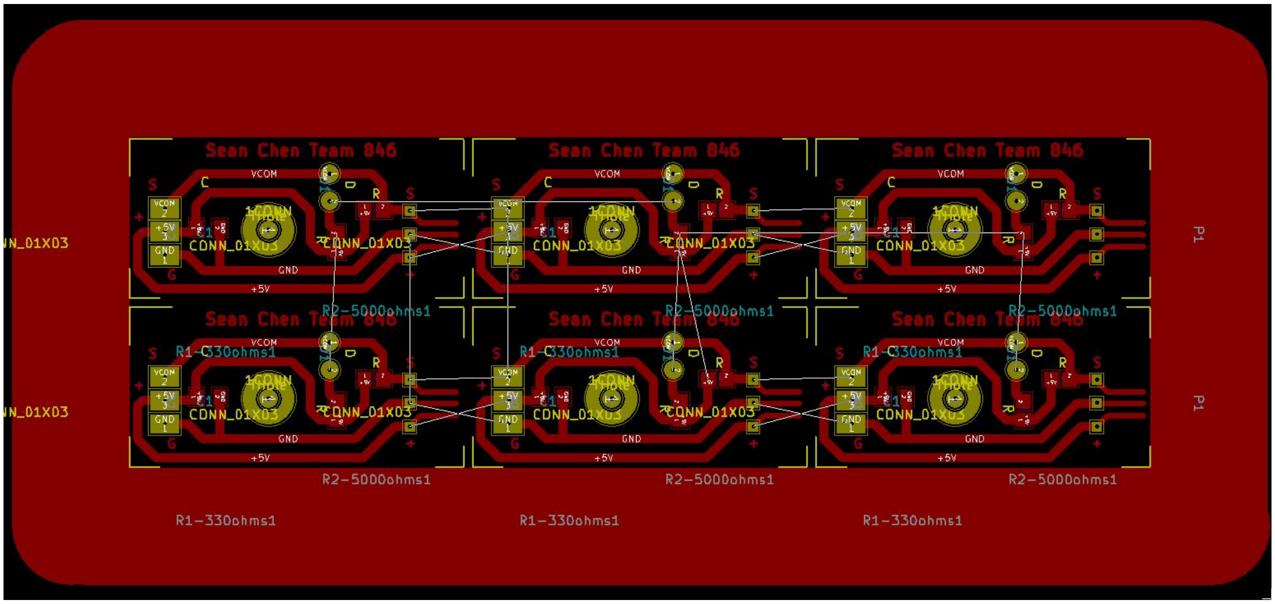
For the second year, the Funky Monkeys implemented printed circuit boards (PCBs) on our robot. However, instead of making our boards by sending our designs to a professional manufacturer, we made our PCBs in-house. The result was a fun, educational do-it-yourself experience.

Step 1: Design

Our PCBs were designed with KiCAD, a computer program that designs circuits and circuit board layouts. As shown below, we carefully label each component so when we are manufacturing, procuring the materials becomes much simpler.

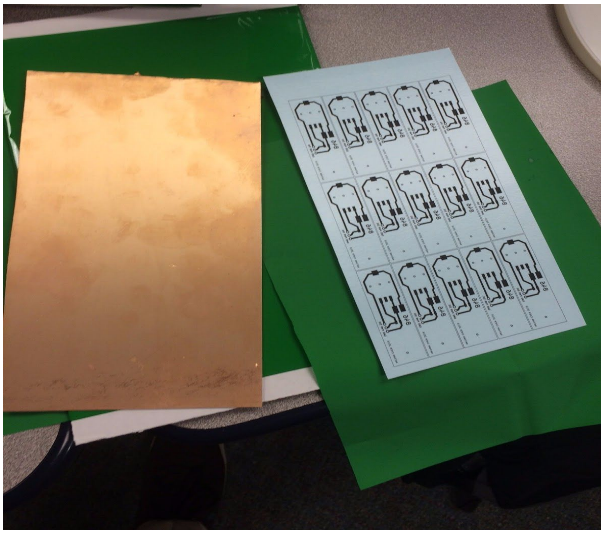


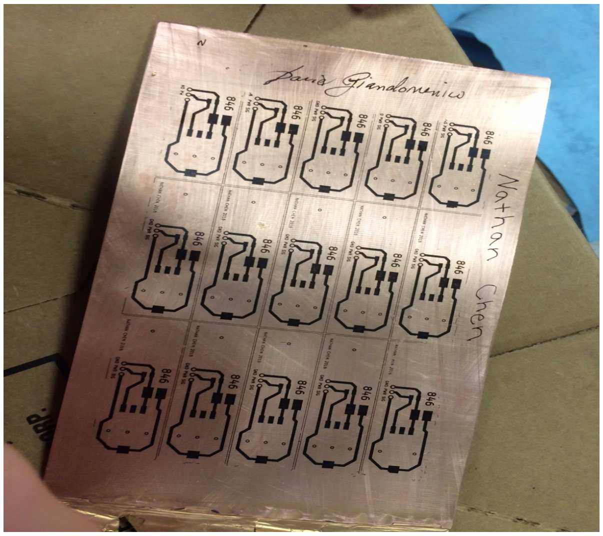
Shown below is a picture of a complete pcb printout. The red lines surrounding the pcb is meant to shorten the etching process, as if there is more copper to eat away, then the process takes longer.



Step 2: Printing

We then printed the board layout onto special transfer paper that could transfer ink onto copper.



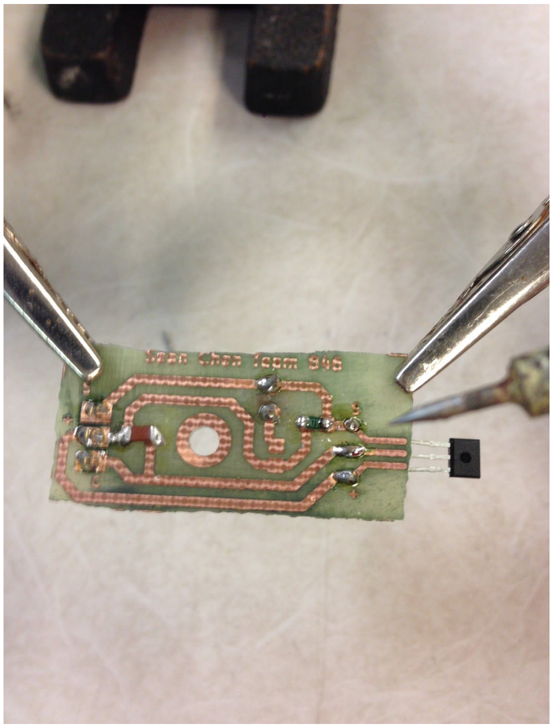


Step 3: Laminating and Etching

After the copper traces were marked by the transfer paper, we sealed them in with a green laminating foil. Then, we used ferric chloride to dissolve the rest of the copper, leaving only the copper paths we wanted.

Step 4: Drilling and Soldering

Finally, we drilled necessary holes and soldered on parts, completing the circuit board!



CAD in Robot Design

Ria Pradeep (Senior)

CAD Training for Members

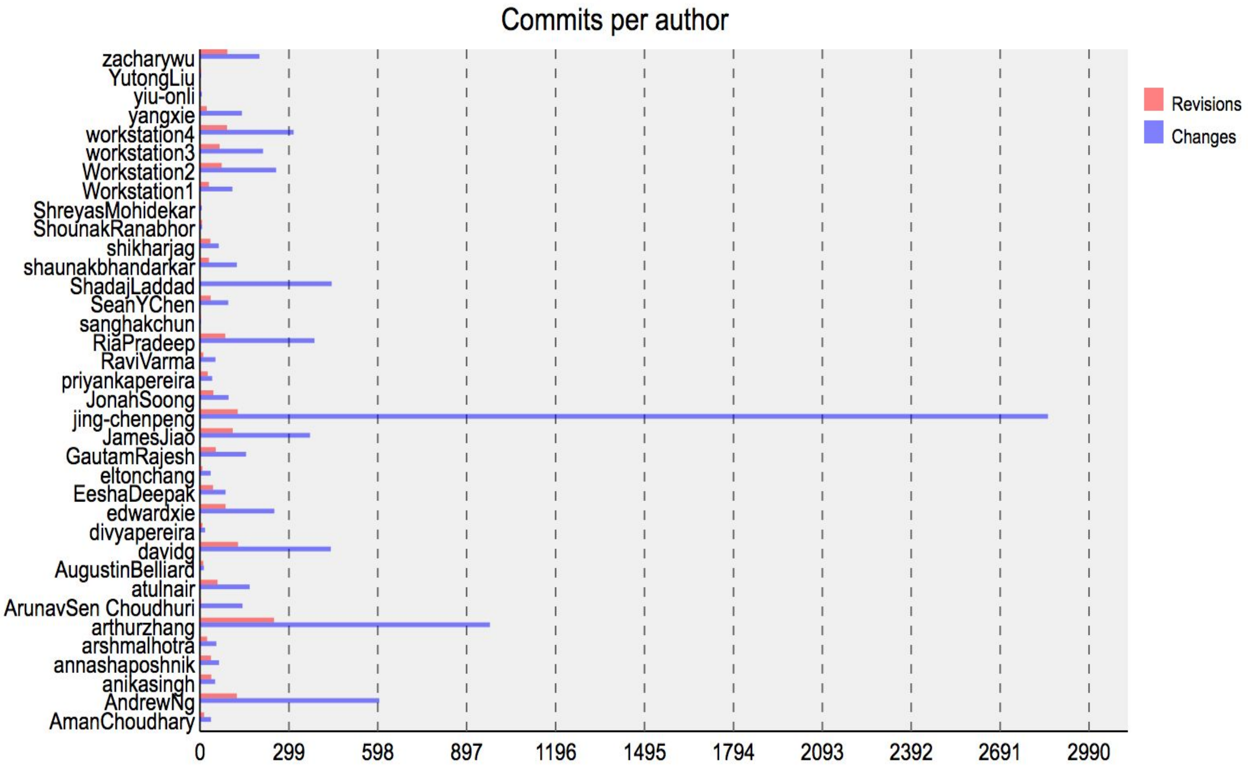
A primary tenet of Lynbrook Robotics is to teach CAD to new members to increase the number of people involved in the robot design process during build season. Over the past seven years, we have trained over 50 members each year in using the Autodesk Inventor CAD software. Our training includes both classroom-like instruction through a series of lectures and presentations as well as personalized teaching before and during build season. Students can gain comprehensive knowledge of Autodesk Inventor through individual designs that the students work on and smaller in-class design competitions. The students then use their new CAD skills to contribute during build season. In fact, this year we were even able to get four of our rookie members deeply involved in the complex design work.

Division of Responsibilities and Workload

Lynbrook Robotics uses a “top-down” approach to split up the design workload amongst multiple members. We analyze the necessary functionalities of our robot and then split up the design responsibilities amongst different subsystem leads, who collaborate to put together their CAD assemblies into one final robot model. Through a use of sub-assemblies and modular robot components, we can create a robot model that can be easily modified when necessary. Our robot’s three primary subsystems this year included our drivetrain, our shooter, and our collector. All three of these subsystems were student-led and student-designed.

File Management and Collaboration through Subversion

Since 2009, Lynbrook Robotics has used the Subversion version control software to manage and share our robot design files among the entire team. Using Subversion, a third of our membership is involved in the robot design, creating CAD models, making drawings for machining, maintaining bills of materials, and recording their design calculations on worksheets. We currently have 1790 revisions to our Subversion repository with contributions from 31 students.



Statistics for repository commits per author.

Conclusion

Through build season, our team has worked hard to produce a design that effectively plays the game, FIRST Steamworks. We build each subsystem around a list of functionality requirements, to design a competitive robot. Every aspect of the design is considered, from the overall structure to every last bolt.

To facilitate our design process, we break our robot into systems, which allows for numerous students to contribute to design. Our modular structure and subversion and git repositories also allow for simultaneous design work with easy assembly.

After much testing and long hours designing, we are excited to see our work come to life on the field. We are proud to present our 16th robot, *Punk Monkey*.