Engineering Notebook

Blueberry
Wolf

#11029 E

college composition book

Table of Contents

Table of Contents Pt. 2

First Meeting

- Held online at 3:45 PM (Est. Time)
- We established roles and the specific team members that we would have (see next event)
- Adrian explained some ideas and designs he was thinking of
- We had all already learned about what the challenge would be this year and some strategies for completing it.
- Adrian built a prototype - (see page)
- We quickly realized that due to 2020's special circumstances we must work extra hard to succeed.
- However, since Adrian has many parts and even a field at home, the main challenge would be communication and working together

Team Members:

Roles:

The Team

Team Members: Adrian Palamorev, George Zhao, Timothy Jin (me)

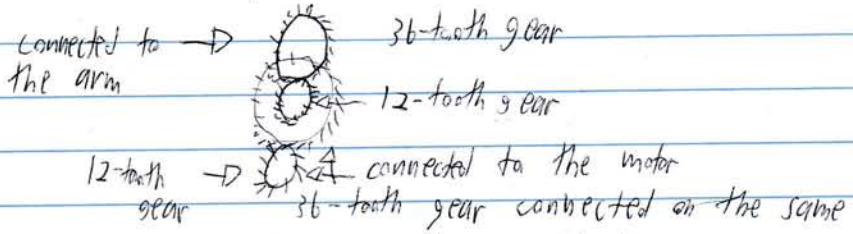
Roles: Adrian - Driver, builder, and programmer

George - Helps out with everything

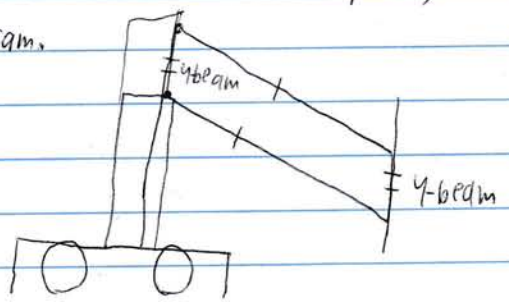
Timothy - In charge of writing the notebook

First Major Robot

- For this game, a robot must be very fast.
- There are a lot of risers, and moving around the field is needed.
- Because of this, the gear ratio for the drive train should be 2.5 (a 60-tooth gear to a 24-tooth gear for speed)
- I have found out that having only one of the driving wheels connected to the motor that powers a side of the drive base faster. Unfortunately, this leads to the challenge of the robot tipping, and then not really being able to move. The robot tips forwards, and the driving wheels [redacted] (ones connected to the motors) are on the back. To fix this issue, the wheels directly connected to the motors for driving should be on the front.
- Since the risers are not light, the robot's arm should have a lot of torque. Otherwise, two motors are needed for the arm, which will probably be the case in future robots.
- The arm ratio is 1:9 for torque, two 1:3 ratios linked together.



- To keep the front of the arm completely straight, the arm should be a parallelogram.



First Major Robot, Continued

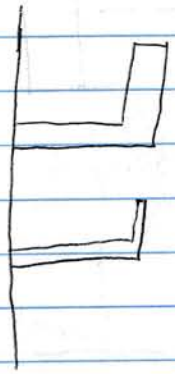
- To help the mobility of the robot, there is an H-drive with a ratio 2:1 for speed (a 48-tooth gear to a 24-tooth gear)

← 48-tooth gear

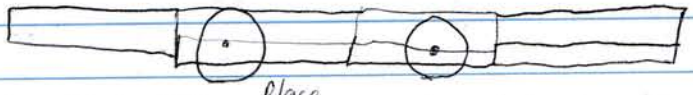
← axle for wheel attached on the 24-tooth gear

- To have an H-drive, all of the wheels need to be of the sliding type.

- The arm for picking up the risers is a basic-two hook arm

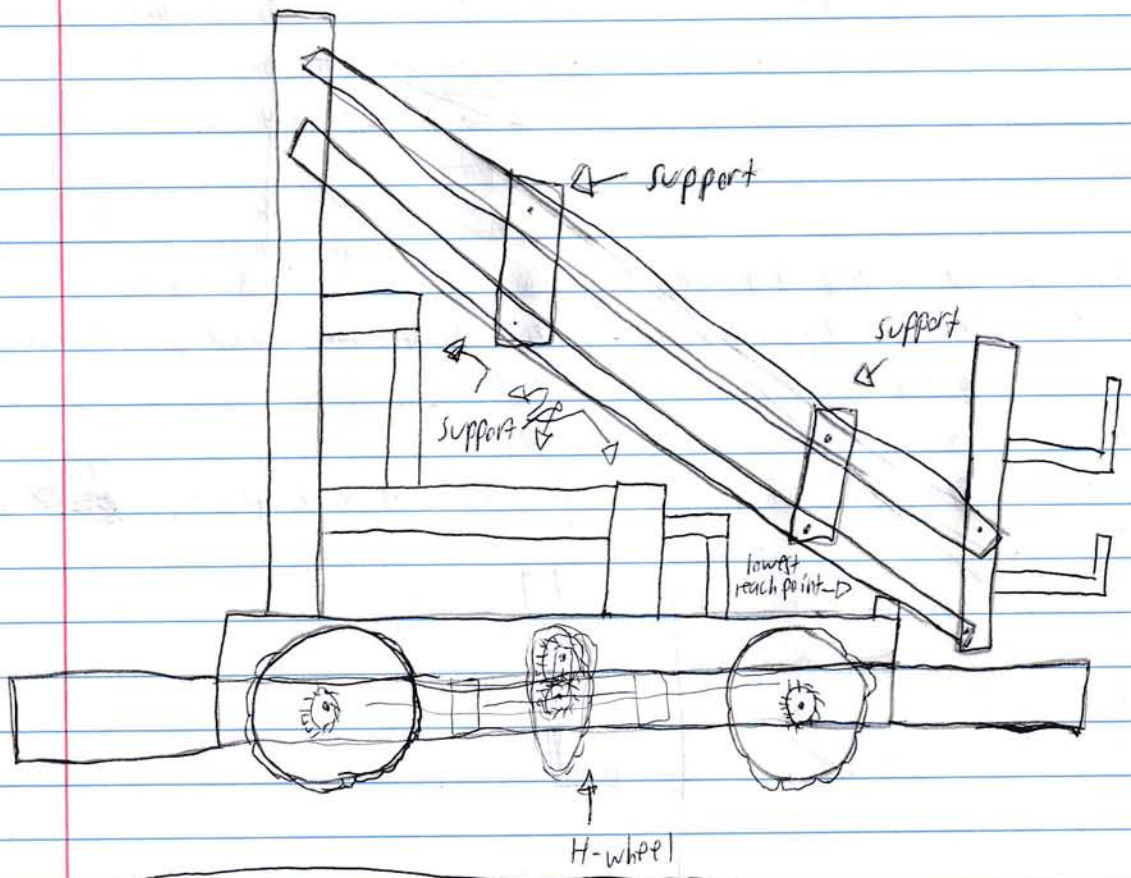


- The two hooks ensure a better pick up of the riser, especially when two risers are being lifted.
- This robot can stack all of the risers.
- For there to be less friction, anywhere where there is a contact between a beam and an axle there is at least one washer.
- The rubber collars create more friction (plastic on plastic is better than rubber on plastic)
- For directing risers better, there are prongs on the front and back:



- It is very easy to only partially place a riser, but to be scored it needs to be fully in the goal. The prongs help with this.

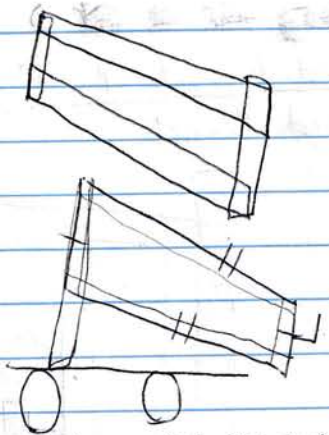
First Major Robot, Final



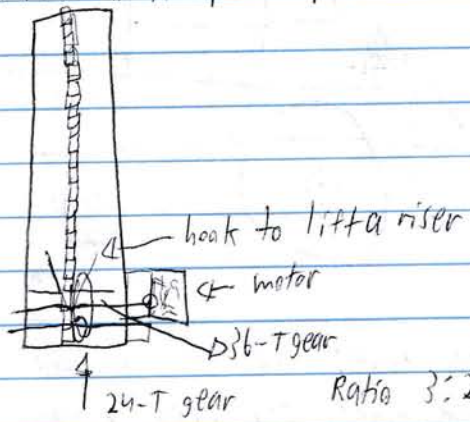
Left Drive -	2.5	ratio	for speed
Right Drive -	2.5	ratio	for speed
H-Drive -	2:1	ratio	for speed
Arm -	1:9	ratio	for torque

Second Major Robot

- Gear ratio [redacted] 5:2 for driving, as before. It is very beneficial to have a very fast robot. The wheels connected to the driving motors are on the front.
- The arm is shorter than the arm [redacted] of the first robot. The gear ratio is 1:3 for torque, with two motors to provide even more power for lifting.
- The [redacted] arm is still a parallelogram



- The H-drive was not very useful in the previous robot, and I decided to remove it. The only use it really has is in programming.
- The robot has a chain-arm to pick up another riser on the back

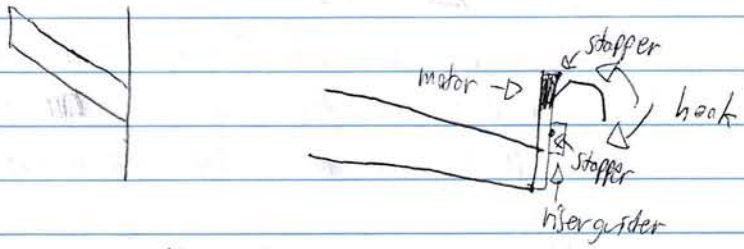


hook (back):
not very efficient

- The main difference [redacted] from ours the previous robot is that the hook on the front for getting the risers is different
- It has a motor connected to it, with a 1:1 ratio.
- The hook does a better job at picking up the risers, but it is not stellar.

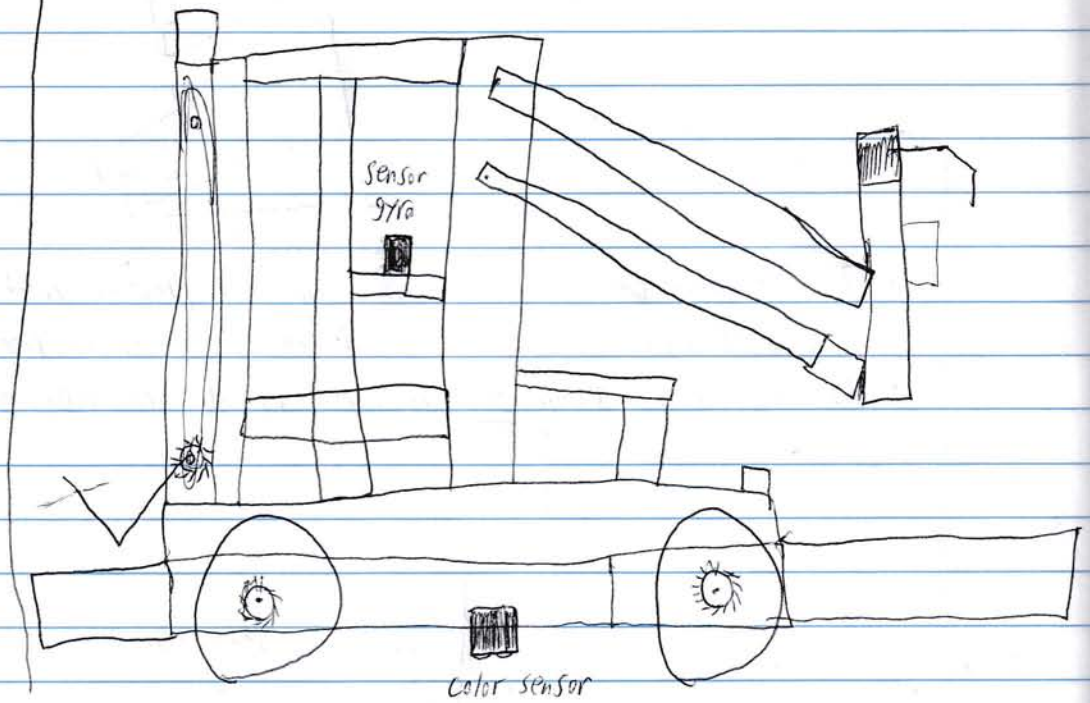
Second Major Robot, Continued

- The hook:



- The robot does not tip like before, since the wheels connected to the motors are in the front

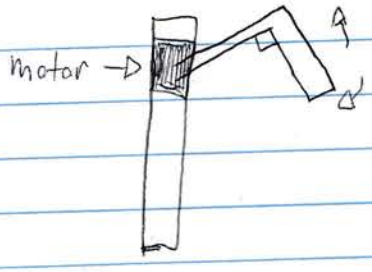
Robot Picture:



Left Drive: 5:2 ratio for speed
Right Drive: 5:2 ratio for speed
Front-Hook: 1:1 ratio
Back Arm: 3:2 ratio for speed
Left-Motor-Front×Arm: 1:8 ratio for torque
Right-Motor-Front×Arm: 1:8 ratio for torque

Third Major Robot

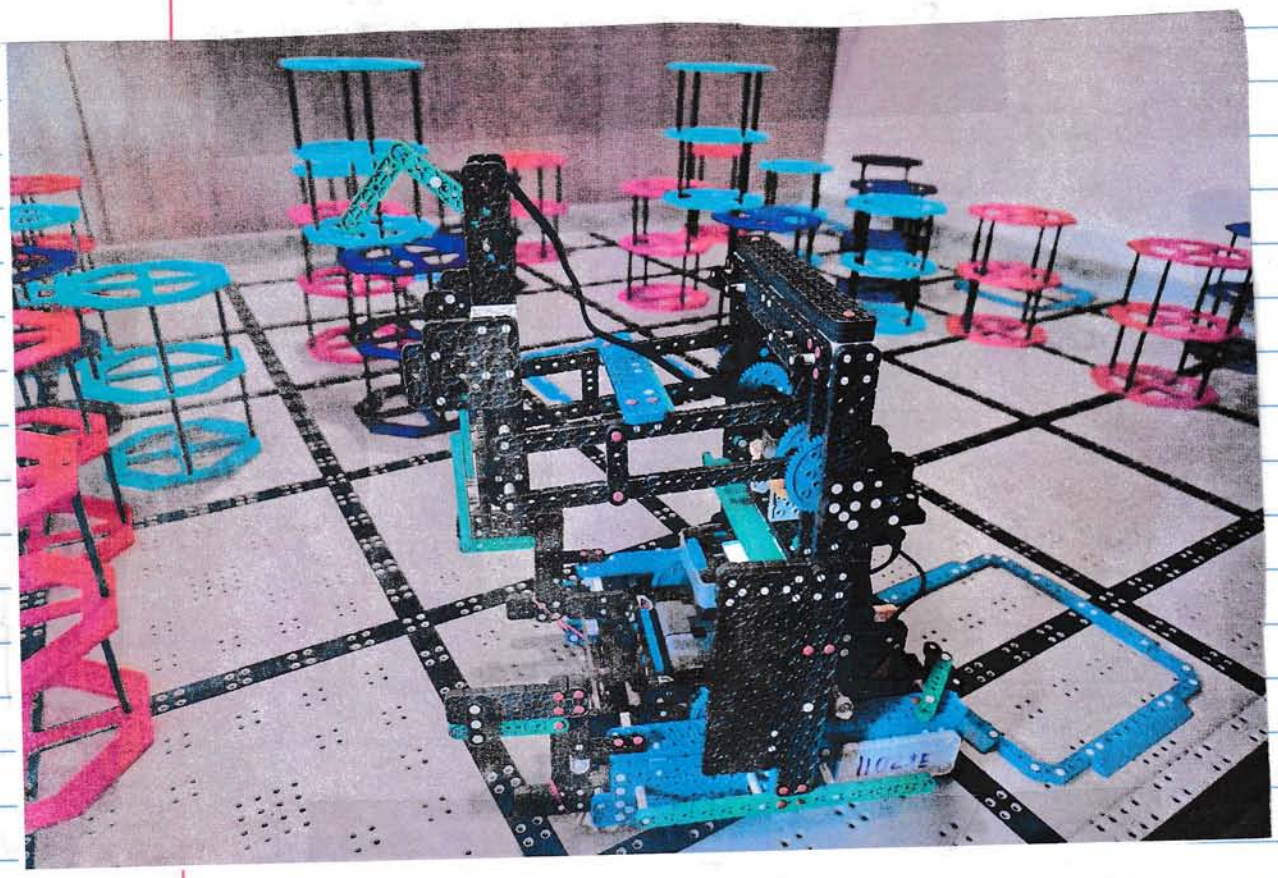
- Driving ratio 5:2 for speed, ~~of~~ the same as for the previous two major robots.
- The driving wheels connected to the driving motors are in the front as last time, so that the robot still drives even when it is slightly tipped.
- I put back the H-Drive to help out the programming. In programming, ~~it is~~ it is beneficial to have the H-Drive, so that the robot can align itself to the black lines using the two color sensors.
- The arm was slow before, and I had to wait for it to fully go up. Now, the ratio is 1:4, with two motors, and there is still enough power. Before, the ratio was 1:8, now it is 1:4, still ~~for~~ for torque.
- The front hook of the robot is similar, but not the ~~same~~ same as before.



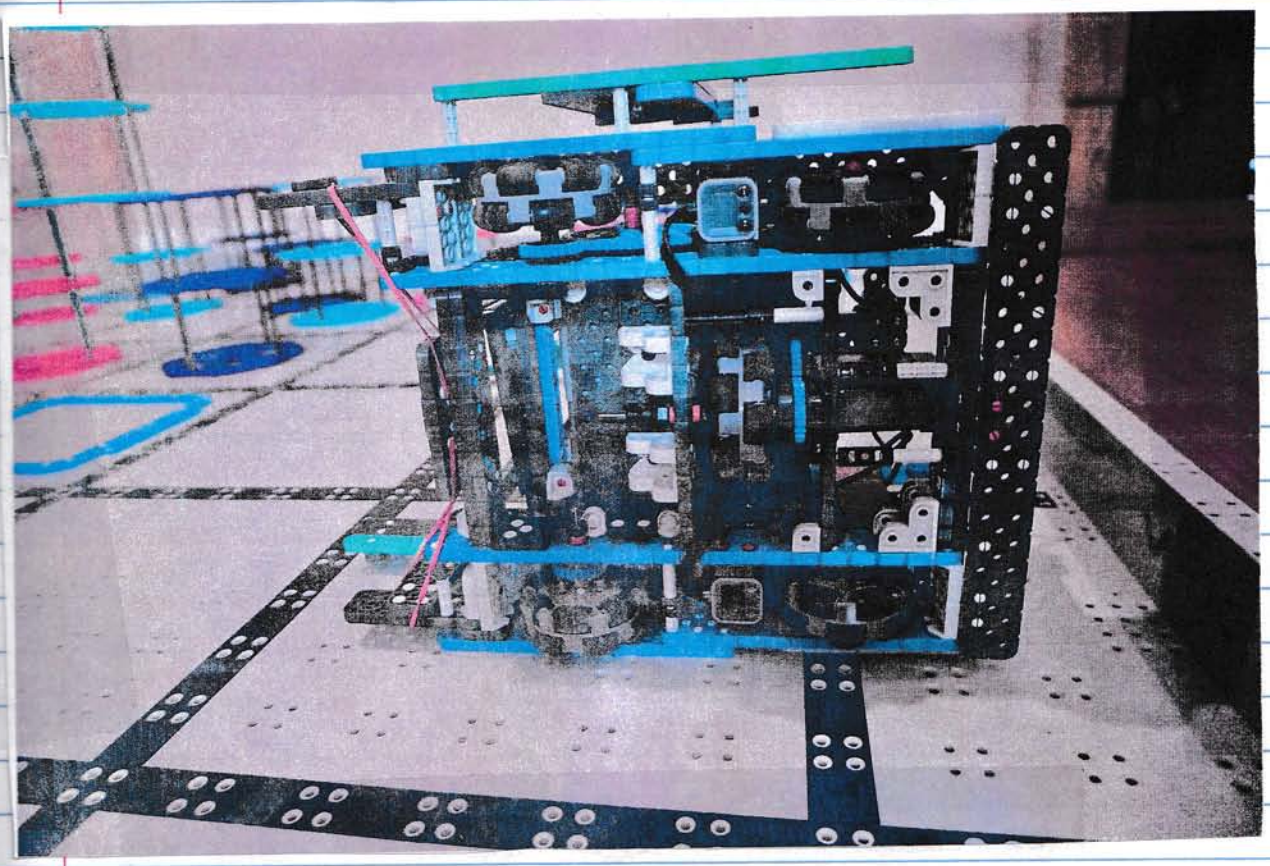
  - Also, the hook is not centered, to help with picking up ~~the~~ the risers, but this did not turn out very beneficial.
  - For the next ~~robot~~, the hook will probably be different, and centered.
- I removed the the chain arm that was on the previous main robot, since it was not very useful. It ~~was~~ was impractical, and I ~~could~~ barely used it.

Left Drive: 2.5 ratio for speed
Right Drive: 2.5 ratio for speed
Front-Hook: 1:1 ratio
H-Drive: 2:1 ratio for speed, kind of sluggish (needs to be changed for the next robot)
Left Motor- Front Arm: 1:4 ratio for ~~torque~~ torque and some speed
Right Motor- Front Arm: 1:4 ratio for torque and some speed

Third Major Robot, Pictures

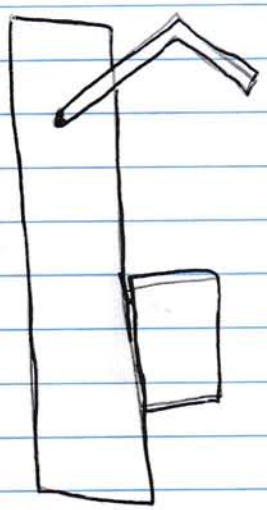


Third Major Robot, Pictures, Continued



Third Major Robot, Pictures, Final

Hooks the risers:



12-30-2020

Trying a 3:1 Gear Ratio

- The driving gear ratio of 2.5 B changed to a chain and sprocket ratio of 3.0. The robot moved slightly faster, but the acceleration was slow, and there was very little torque. I believe acceleration with chain and sprocket gears are slower in general than normal gears.
- There was a higher level of strain on the motors, and the battery was drained faster.
- due to this, the ratio has been changed to original 2.5, 60 tooth to 24 tooth gear ratio
- When the gear ratio was changed, one of the motors was sluggish, so it was replaced.
- It may have to consider reducing the weight, and trying a gear ratio of 2. The acceleration would be faster/greater, and torque would also increase, but I do not know if the overall speed will increase/decrease

1-16-2021
12-12-2020
11-21-2020

SRE Virtual skills only competition #4
Tampa Bay skills only competition
January 2021 Westwood Skills — only Competition

Westwood:
1/16/2021
Driving: 204 — 210
Programming: 6
204+6 = 210, 1st place, Skills winner

SRE Virtual competition:
12/12/2020
Driving: 204 — 212
Programming: 8
204+8 = 212, 3rd place

Tampa Bay Virtual skills:
11/21/2020
Driving: 173 — 177
Programming: 4
173+4 = 177, 2nd place
• The score was put in as 180, but it should have been 177.

1-28-2021

Making the H-Drive work Better

- Since the front of the robot is heavier, and the h-drive wheel is more towards the back, and the gear ratio of 2:1, there was not enough torque
- Removed the gears that made the ratio 2, and now it's 1:1
- There are still difficulties, but the changes improved it.
- The 2:1 gear ratio slowed the h-drive, but and now it is faster with 1:1
- Driving gear ratio changed from 2.5 to a chain and sprocket ratio of 3:1. This made it faster, but there was little acceleration, and very little torque
- Battery was also drained faster due to higher motor strain
- Due to this, ratio was changed back to 2.5, 60 tooth to 24 tooth gear ratio.
- One of the motors stoned, so it was replaced, improving the speed
- The weight may have to be reduced, and the gear ratio might also be reduced. There would be more acceleration, but not necessarily more torque or higher speeds.

2-6-2024

Robot Programming

- To make the robot more reliable, there are now gyro sensors for turns and two color sensors for detecting the black lines of the field.
- Making turns without a gyro sensor is very unreliable, especially for fast drive trains.
- A slight difference in the charge of the battery can ruin the program
- Unfortunately, turning with a gyro sensor is not as reliable as it should be. Because of this, the robot programming has failed at every single competition so far
- The color sensors can be used to align to the black lines on the field, which can clear up some of the accumulated errors
- In between most modules of the program, there is a short delay. Roughly 4 milliseconds to 25 milliseconds. This seems helpful, because the robot needs to pause to drive accurately. The momentum it acquired can throw the robot off course if it does not come to an almost complete stop.

2-13-2021

New Robot Claw

- In previous design, there was a need for high precision when hooking rises. With the new claw, there is not such a great need for exactness. this is very beneficial to programming, so it increases our chances of getting 70+ points

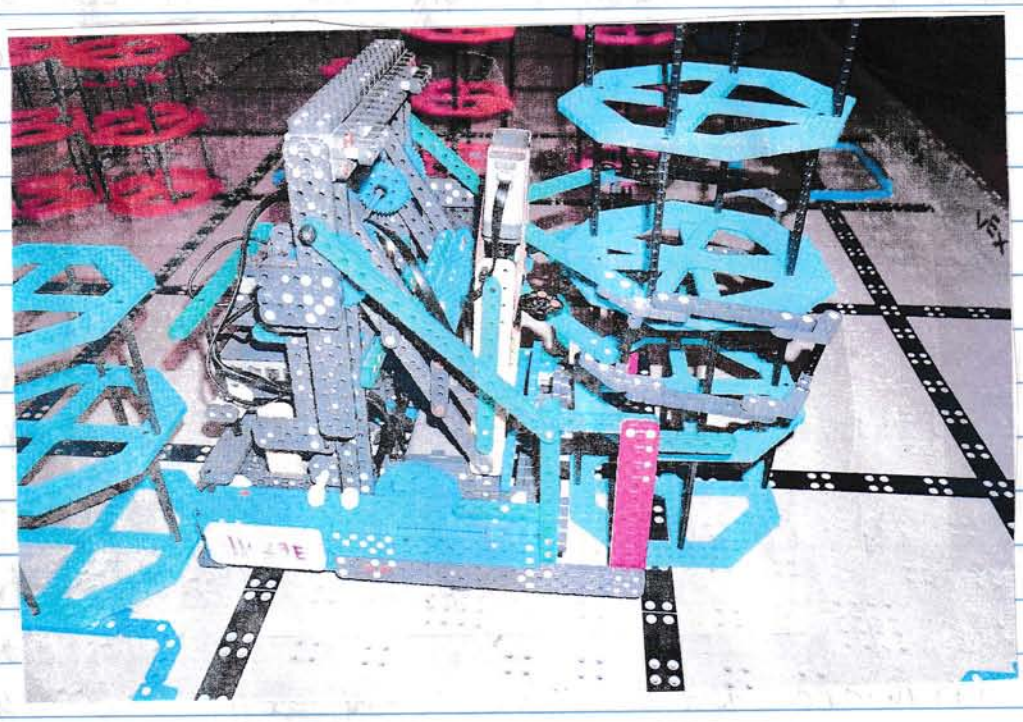
- The claw latches on outside of the rber, so there is a wider range of successful picking-up.

- The gear ratio is 1:3, for torque (12 tooth to 36 tooth) The left and right sides of the claw are connected by two 36-tooth gears. Since there are two gears adjacent, the direction is flipped. If the left gear spins clockwise, the right will spin counter-clockwise.

- So far, the robot has proven to be more consistent in programming with the new claw.

- The new claw is heavier, so it creates a weight imbalance and tips the robot. To solve this, the brain was moved back, and counter-balanced it.

New Robot Claw (Picture)



(GOOD

BAD

BAD

2-20-2021

Tampa Bay Skills Invitational

GOOD - We received the Design and Build Awards.

BAD - Programming has failed another time.

- The robot was sluggish, and Adrian could not score any points on driving.
- We earned 173 on driving, and 4 on programming for 177 points total. (still less than previous 212)

NVA	Driving	Programming
First	108	1
Second	105	1
Third	193	4

- Tom suceed in programming, it needs PID (proportional integral derivative)

BAD - The base of the bot has worn out, and the axles have spun so much that they have slightly damaged the beams that support them.

2-28-2021

Bot Camp Live Remote Skillsorly

- Although the ratio of the robot's gears is 2:1 2.5, it's responsive speed is more like [1.5, 1.67]
- The problem is probably with friction slowing it down.
- Programming failed again :(
- We were hoping for at [redacted] least 205 on driving, but Adrian scored 172 in driving, and 2 in programming, for a total of 174 points.
- This is still less than the Record: 212, which has been our goal to beat.

	Driving	Programming
First run	141	0
second run	108	0
third run	172	2

3-1-2021

Fixing The Robot

- Since the robot was so slow at the competition yesterday, Adrian has removed the a few side beams that on the drive base to look for a few issues.

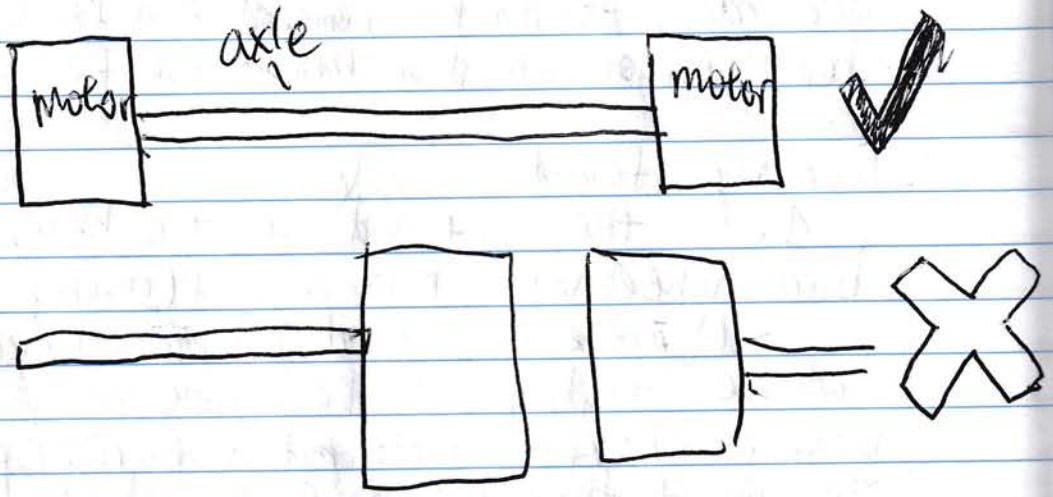
- Two were found:
  1. On the right side of the base, the back wheel was half a unit (roughly 1/4 of an inch) above the ground. The most likely cause would be it sliding off the beams, and when it was put back, it wasn't put in the right place. This meant that the H-Drive wheel was probably supporting much more weight than it should have.

  2. The same wheel was also touching an over head beam, creating friction. The force of friction was most likely slowing the robot down, especially on the right.

- After the wheel was fixed, and some unnecessary beams were removed, the robot moved a lot faster!

- Plus, the lift arm was not rising quickly, so it was changed. In general, it's better to have the two motors lifting the arm acting on one axel, instead of two small ones. When they are connected to two shorter axels, the faster motor tries to move it faster, while the weaker one can't, wasting available energy. When they are synchronized, they move faster.

Diagram



- After they were connected, and reinforced, the arm rises faster

- With some more practice, and with the new speed, 2:42 should be more achievable.

3-6-2024

Disco Dynamic Live VEX Remote T.

- First live remote tournament we participated in.
- It was a Canadian Event, so there were many good teams.
- Scored 9th out of 13th, avg. of 177 points, with the lowest discarded.
- There were three finals matches, meaning that the top 6 teams were in.
- 6th place was 20-25 points away.
- In this tournament, there were two teams who scored 306, a full score.
- In the second [redacted] qualification match, the port one driving motor and its wire malfunctioned, and so only one side could/was moving. It started spinning in circles.
- After [redacted] Adrian reset the robot, he could drive for a little and the it went in a straight line. He replaced the wire, and it fixed it.

Match	Score	Notes
Practice	42	The other Lincoln Middle team did not have the competition code.
Match 1	82	Adrian started a bit late and was not warmed up.
Match 2	13	Motor and wire malfunction, scoring 3 points
Match 3	173	Adrian scored 3, other 2
Match 4	211	Both scored 3 stacks
Match 5	242	Adrian scored 4 stacks, and the other team 3

3-14-2021

Change In Strategy

- To possibly get a better score, we are stacking 3 teals, one orange, and two/three purples.
- This checks all the requirements too 210+
- If this is too risky, the old strategy can still be used.
- Since the robot has improved, we can get more points, and the goal is 242 by states.

3-16-2021

Programming

- The program uses functions to result in less total code.

Function → `void DriveStraight(int power, int time)`

```
{
  setMotorSpeed(leftDrive, power);
  setMotorSpeed(rightDrive, power);
  sleep(time);
}
```

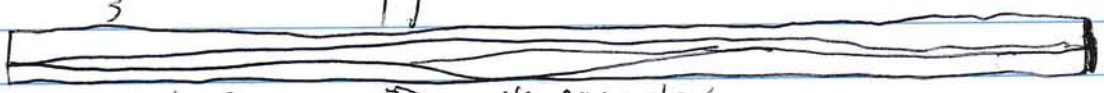
Body →

input parameters

semicolons to end line

- The program is split into Part 1 and Part 2.
- When I start the program from the brain, it automatically executes Part 1.
- To start part Two, which is made to get 3 points, I have to press a bumper sensor.
- The program is made to score 73 points.

```
void PartOne()
{
  Body with many functions and commands
}
```



```
void PartTwo()
{
  Body
}
```

No parameters