Engineering Notebook

12B

Team Number

Puzzles¿

Team Name

The Potomac School

School

11/3/2023	00/00/0000	1 of 1
Start Date	End Date	Book #

v1.0.8.29.22

VEX ROBOTICS

REC FOUNDATION

Engineering Notebook

12B

Team Number

Puzzles¿

Team Name

The Potomac School

School

11/3/2023	00/00/0000	1 of 1
Start Date	End Date	Book #

v1.0.8.29.22

VEX ROBOTICS

REC FOUNDATION

Table of Contents

Table of Contents


12B

Foreword

Our team started our engineering process in September 2022, creating a rudimentary flywheel based robot with limited planning beyond game analysis and drawings. As we began to think about a new robot, we wanted to begin documenting our work; first, to effectively engineer, design, and build a robot to excel in the game (not just to only complete the necessary tasks); second, to push ourselves to build and document a complex robot for our own learning and understanding. This Engineering and Design notebook begins with a brief section on initial planning and the first robot and then progresses through our mid-season approach for a new complex robot with a unique transmission, and finally will feature our cyclical engineering process throughout the designing, building, and testing, and competition stages. We hope you enjoy reading about our journey. Thank you.

Team Intro

Our Team

Ben Runde - Co-Leader, Lead Builder

Arav Bhargava - Co-Leader, Driver, Lead Strategist, Assistant Programmer, Builder

Our Team

David Gardner - Assistant Programmer, Builder, Drive Team Member, Lead Scout

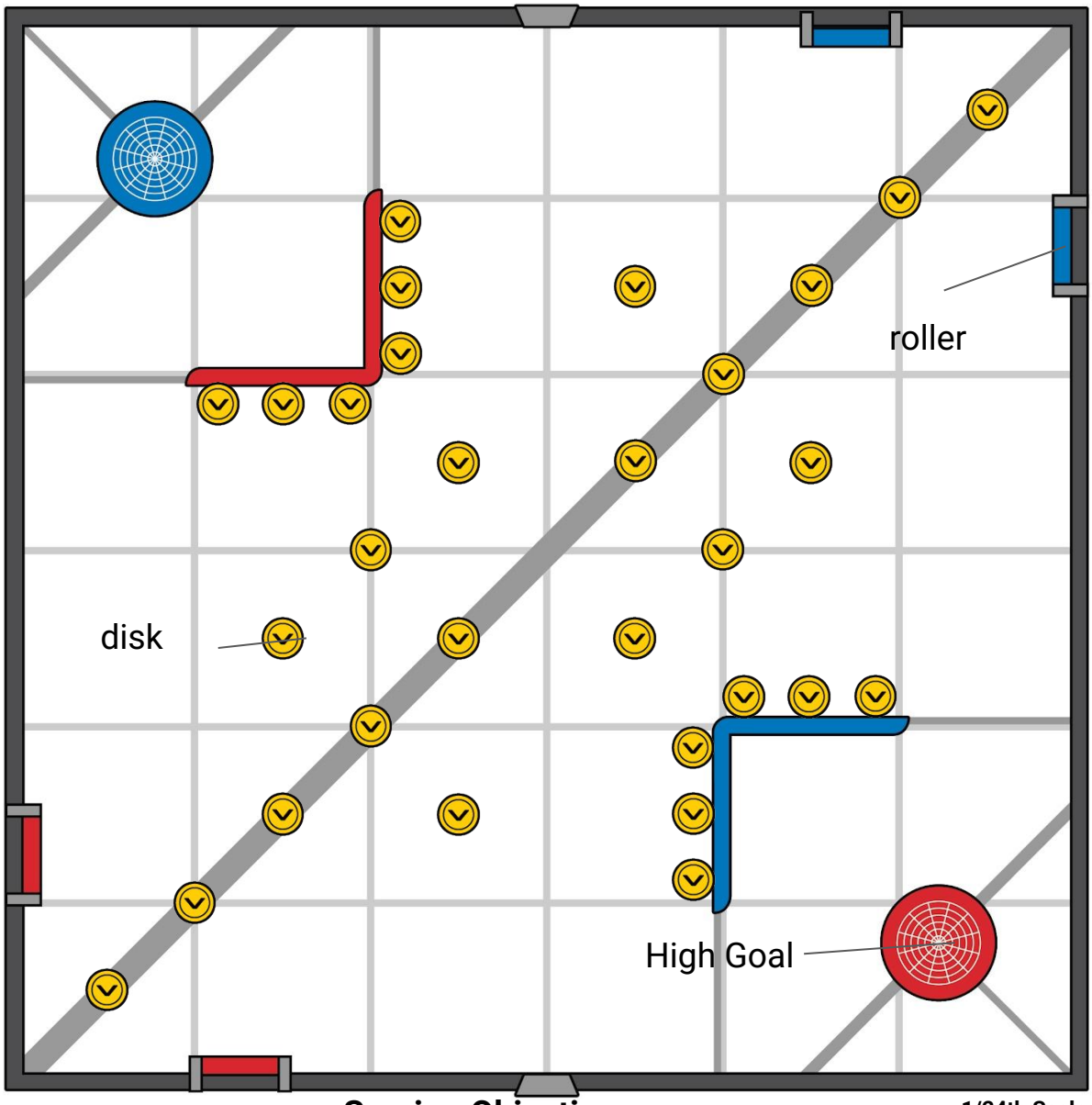
Devin Gaines - Builder, Assistant Strategist, Assistant Programmer, Drive Team Member

Our Team

Daniel Overdeck - Lead Programer, Assistant Builder

Previous Robot And Old Analysis

Game Analysis



1/24th Scale

Scoring Objectives

-60 disks
- 2 preloads
- 7 match loads
- 38 on fields
- +5 points in high goal
- +1 in low zone

-4 rollers
- +10 if owned by alliance

- +3 for every tile touched at end of round

Match Rules

- 15 second autonomous period
- 1:45 driver control period for head to head matches
- 2 disks preloads
- Horizontal Expansion **is limited** until the **endgame**
- Trapping - Cannot pin a robot for **more than 5 seconds**
- Expanding early results in the robot having to park to the side, stop actions for the rest of the match, and expansion doesn't count
- Any part of the robot that expands **outside or touches the perimeter of the field** results in a **DQ**

Scoring Priorities

Maximum Points: 60 x 5 high goal disks + 10 x 4 owned rollers +10 autonomous bonus + 28 x 3 tiles = **434 points**

1) **Autonomous bonus:** Awards a win point which helps in standings, also gives us the equivalent of a roller in points

2) **High goal disks:** Expect to make up the majority of scoring, once scored, they cannot be unscored making them the most reliable form of points in the game

3) **Rollers:** Good way to apply pressure during the mid game, due to being a contested objective they should be prioritized near the end of the match, essentially worth 20 points b/c they add 10 to an alliance and remove 10 from the other

4) **Expansion:** Inconsistent yet potent way of scoring points, can max with 84 points if done perfectly yet risks an early expansion or a DQ

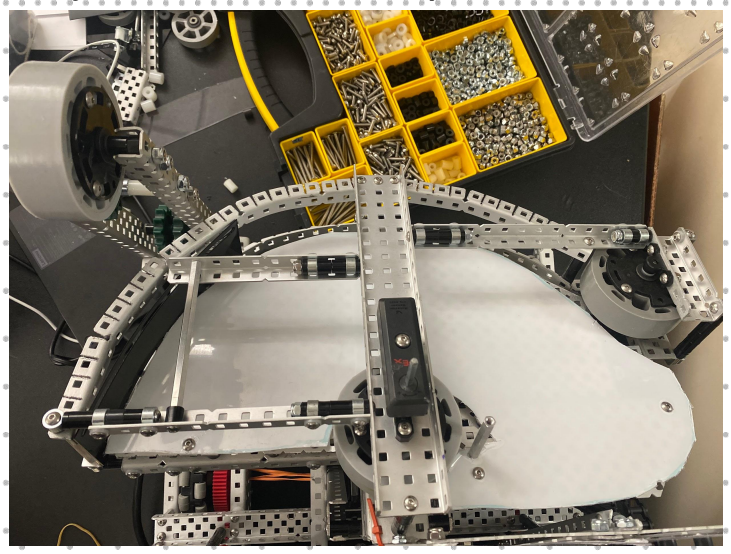
5) **Low Goal Disks:** Lowest priority, only worth 1 point.

First Robot (Flywheel)

During the first half of the season, our team used a different robot. This robot was a side-shooting flywheel robot based off of a team called 77000X from August's Mall of America Signature Event. This bot contained a unique flywheel shooting with a circular motion, a 4 motor drivebase, a roller mechanism, an angled intake system, inertial sensor, rotational sensor and three endgame string launchers.

-Flywheel

For our flywheel, we cut into one big C channel and slowly bent it to create a circular shape for the disk to travel through. Our goal was to be able to shoot from anywhere on the field. In order to fulfill this goal, we designed a flywheel with the many different contact points. By creating a flywheel with numerous contact points, the disk can speed up more resulting in a larger range of shooting.



Pros:

- We were successful in having a large range of shooting

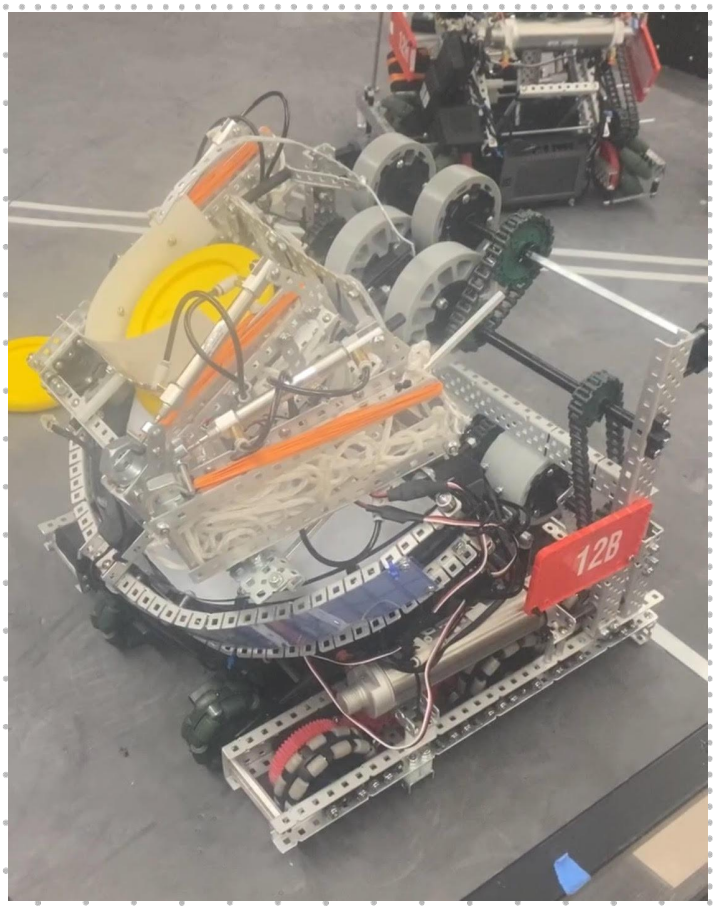
Cons:

- Very difficult to aim
  - One of the main reasons it was hard to aim was because of where the release point was on the robot. The disk was set to release on the right side of the robot instead of the center so aiming was slightly off at times.
- Slow shooting rate
  - This is because of the area the disk has to cover before being launched. The disk needs to go up the intake system, and then revolve around the flywheel adding more time to the launch

First Robot (Flywheel) Cont.

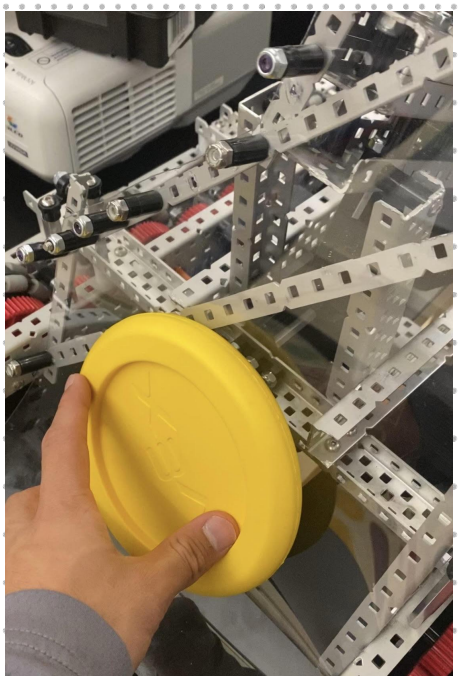
-End Game

Our end game consisted of three different string launchers and each launcher was released by different pneumatic pistons, none of them shared the same one. We averaged about 15 tiles per end game at the time which is not as good as other teams with six different launchers. But, even though it was not as many launchers as other teams, it was still efficient and got our team an adequate amount of points.



-Angled Intake

Regarding our intake, it was very successful. We connected our intake to our roller with a chain so they both moved simultaneously. We used small wheels to pull the disk up and Into the flywheel. To ensure that the disk does not get offcentered while traveling throughout the system, we used spacers and standoffs to keep it centered.



Mid Season Game Analysis and Preliminary Robot Design

Mid-Season Game Analysis

As the season is progressing towards states our team has begun analyzing the most optimal game strategies and build design aspects. From our discussion here are our takeaways:

- Spin up is dominated by teams with speed and power — the **fastest intake to triple shot** (the time from when the first disk enters the robot to the time when three disks have been shot), **fastest robots to evade defense**, and **powerful drivebases that can overpower opponents** attempting to take shots on the high goal.
    - We believe that a ***slim 6M drivebase with a transmission*** to switch the robot between a 'speed' mode and a 'power' mode would allow us to play fast offense and powerful defense in a single match.
    - Also, by incorporating a **wedge on the back of the robot,** this may help with playing defense and messing up opponent shots.
    - We were debating between slingshot and flywheel, both accurate strategies for shooting, but **we chose slingshot** because it was three shots at the same time which is inherently faster than 1 at a time (flywheel). *This is discussed more on the following page.*
    - Finally, 6 string launchers on the top of the robot would allow us to cover every tile on the field but not risk shooting too wide and being disqualified

Using these considerations we designed the robot using Computer Aided Design (CAD) which will be displayed further in the notebook.

Why Slingshot?

Slingshot fires three shots simultaneously on top of one another is more efficient than one after another (flywheel/turret)

When a slingshot shoots disks, the disks are all **grouped directly on top of each other** creating incredible accuracy

Contrasts with **catapult based robots** that often have **bad grouping** due to the arc of the shots.

Why Transmission?

Our transmission uses pneumatic pistons to switch our gear ratio to allow our drive base to move from 450 rpm to 135 rpm

With a transmission, we can play unique offense and defense in a variety of ways.

For offense, 450 rpm allows us to to be efficient on shots, specifically quantity of shots, while also evading opponents

For defense, 450 rpm allows us to stay with other opponents and not let them shoot, while 135 rpm allows us to push and pin any robot (even 8 motor pto robots)

Effective Driving Techniques - From Video Reviewing

From hours of youtube video reviewing and our own driving practice we have identified several unique strategies that can be applied to different in-game situations.

Listed below are the specific strategies we will explain in depth:

Offensive:

1. “Rapid Fire” Continuous Shots on Goal
2. “Fake-Out” Anti-defense Method
3. “Spin Move” Anti-Defense Method
4. “Early Roller Claim” Advantage on Rollers

Defensive:

1. “In-goal Defense” - Dual Robot Defense
2. “Brick Wall” - Roller Defense
3. “1 on 1” - Targeting One ‘better’ Robot
4. “Robot to Jail” - Continuous Wedge Pin
5. “In-Transit Defense” - Disk Removal

Offensive Moves:

Rapid Fire: Offensive Strategy - In more competitive matches, disks must be scored quickly at the beginning in order to clear the field quickly and claim rollers early. To do this, shooting a large quantity of shots instead of worrying about defenders and misses can be the most effective strategy to outscore opponents early.

Fake Out: Offensive Strategy - When defense is being played on the robot, fake-outs, where the robot begins to move to one side of the goal to shoot but then turns back and shoots on a different side, messes up the opponent's defense often leaving them stuck in the low goal.

Spin Move: As you and a defensive opponent are moving towards each other, keep both joysticks forward, switch one backwards until robot is at 180 degrees from its starting position, then switch the second joystick backwards. This move will maintain momentum while dodging the opponent allowing several seconds to shoot offensively.

Early Roller Claim: After using rapid fire strategy, if your alliance is ahead of opponents by >3 disks and majority of the disks are cleared, move to rollers earlier than opponents and have both you and your alliance partner guard two each. If executed correctly, this should leave an advantage on rollers and allow easy endgame release

Defensive Moves:

In Goal Defense: Defensive Strategy - As your alliance partner plays offense, move to the opponents low goal to play defense on both opponents. As the opponents move to shoot on the sides, drive out towards them and make contact with their robot before their shot. Once they shoot, move back in the low goal to guard the second robot. Continue to alternate defense between both opponents. This strategy can be used if both robots shoot from white lines on low goal. Most effective strategy for playing dual robot defense.

Brick Wall: Defensive Strategy - In 'power mode' position the robot with a 45 degree angle to corner rollers. As a robot attempts to steal a roller, cut off their attempt by driving straight forward. Power mode will ensure the opponent cannot pass.

1 on 1: Defensive Strategy - Target only one opponent robot that is playing offense and strike from the side to restrict their movement and deter their shooting.

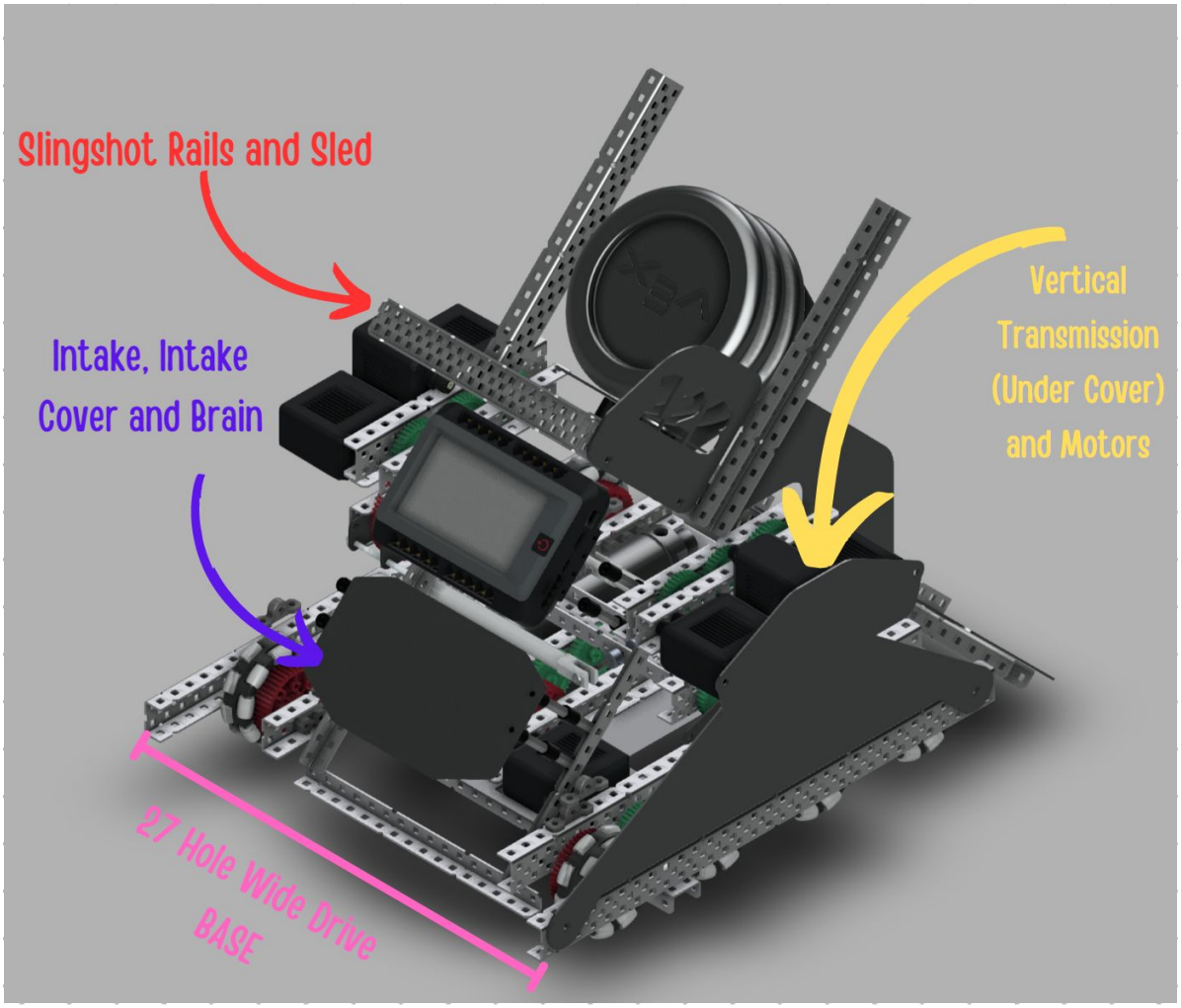
Defensive Moves Cont.

Robot to Jail: Defensive Strategy - While in ‘power mode,’ push opponent robot onto the corner of the low goal to completely remove them from the game. Only will work if our robot intercepts the opponent robot as they are moving across the field.

In-Transit Defense: Defensive Strategy - Inspired by the Las Vegas Signature event finalist team *In-Transit*, this defensive move only works with a very good alliance shooter. Our robot will stay on the opponent half of the field, firing disks into our own low goal to remove them from the use of our opponents. Our teammate will take the disks and put them into our own high goal. This should only be used against two very high scoring opponents.

Basic Design Concept

- To summarize the design of the robot we are planning on creating a **27 hole wide robot** with **omni and traction wheels,** with a **vertical transmission** to preserve central space, and a **slingshot firing mechanism.**
  - In addition to the main functions of the robot, driving and slingshot, we will have **endgame string launchers** to make contact with as many squares as possible during endgame, and will have a **wedge** to lessen the traction of our opponents robots

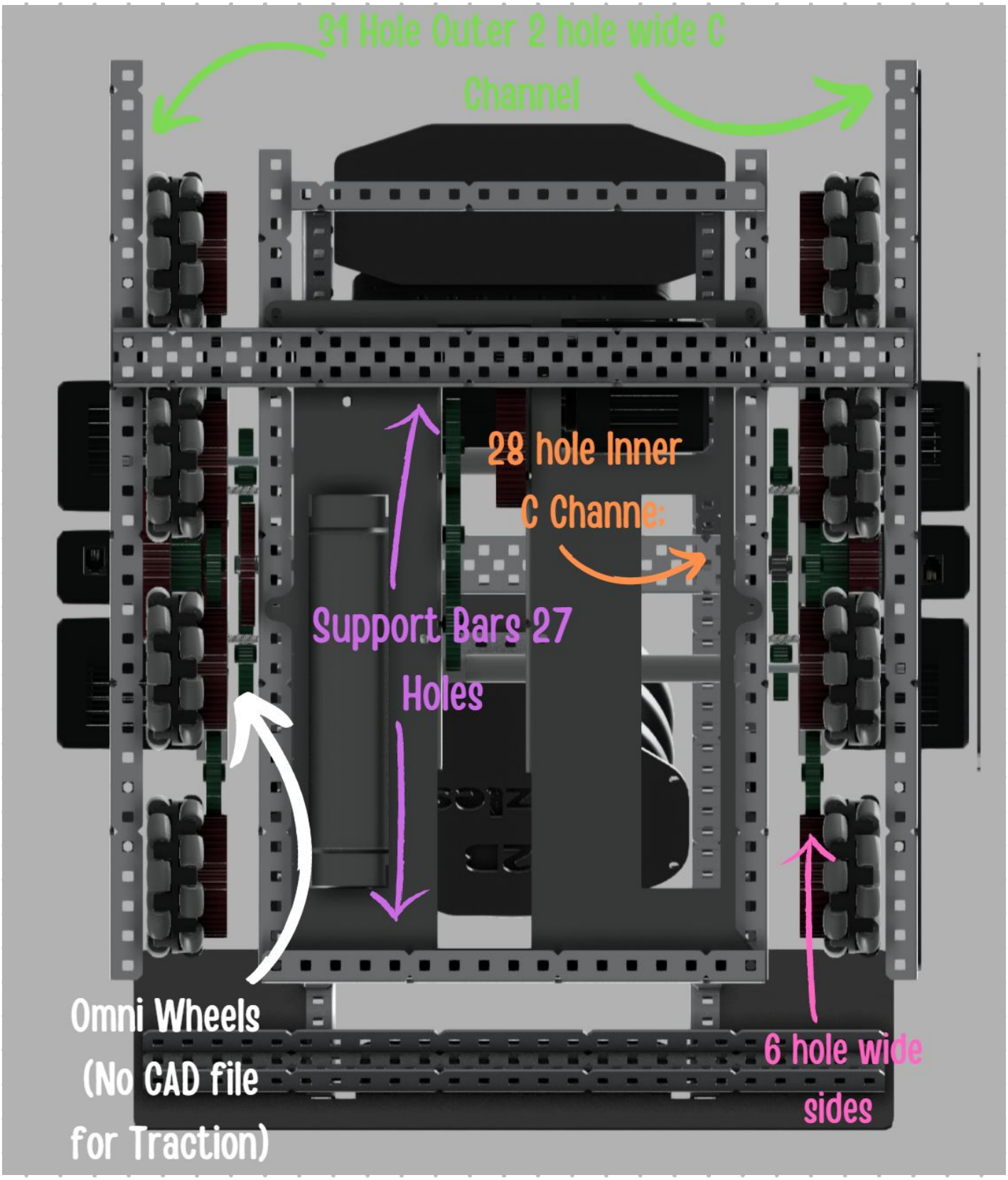


Drive Base Design

Main Design Features:

- **27 Holes wide (each side is 6 holes wide)**
  - We chose 27 holes for the width of our robot because we thought it would be a good combination between internal space (makes slingshot easier) and narrowness (improves driving performance)
  - We made each 6 holes wide (rather than 5 holes the most common and slimmest inner drive base width) to allow for the mounting of one more gear to connect to the drive base and grant the transmission gears more space slide laterally
- **31 outer drive base length and 28 hole inner drive base length**
  - We chose 31 holes for total drive base length somewhat arbitrarily, its long yet still allows for some space inside the 18x18x18 dimension requirements
    - We then decided to make the inner drive base length slightly shorter to create a funnel shape to help guide discs into the intake
- **Outer Omni Wheels and Inner Traction Wheels**
  - We Chose to do a combination of both omni and traction wheels since they would allow us to both turn tightly while stopping enemy teams from pushing us latterly (Only Omni → Sharp Turns/No Lateral Friction || Only Traction → Lateral Friction/Slow Turns)
- **Central and Rear Support Bars**
  - To support and connect both sides of the drive base and robot (27 Holes wide as mentioned before)
  - We mounted one centrally as to not interfere with the intake one end then mounted another on the opposite end of the intake
    - No Support on the Front, But Yes in the Center and the Back

Drive Base Design



Transmission Design

Design Process: Preliminary Design:

- While at first we imagined that the transmission would be placed in the bottom of the robot, in between both sides of the drive base, we realized that we were going to have to expand the robots width to do so (also the bottom was a convenient place for the pull back mechanism of the slingshot)
    - Therefore I decided to move it vertically aboth each side of the drive base
        - While unorthodox placement, we thought it was very **innovative** and **unorthodox,** as we have yet to see another team (through thorough research of current and past designs on The Forums, Reddit, Discords, and Mentors)
            - Once I decided that the transmission was going to be vertical I created a basic design concept: The Transmission would be made up of three distinct layers (all of which would be distinguished by small c channels perpendicular and on top of the drive base c channels)
                1. **The Bottom Layer → would hold the transmitting gears and connect to the drivebase**
                2. **The Middle Layer → Would hold the appropriate gears to create the various gear ratios of the transmission in both positions**
                3. **The Top Layer → Would hold the motors and the gears connecting them to the middle layer**

Transmission Design

Design Process: CAD Creation

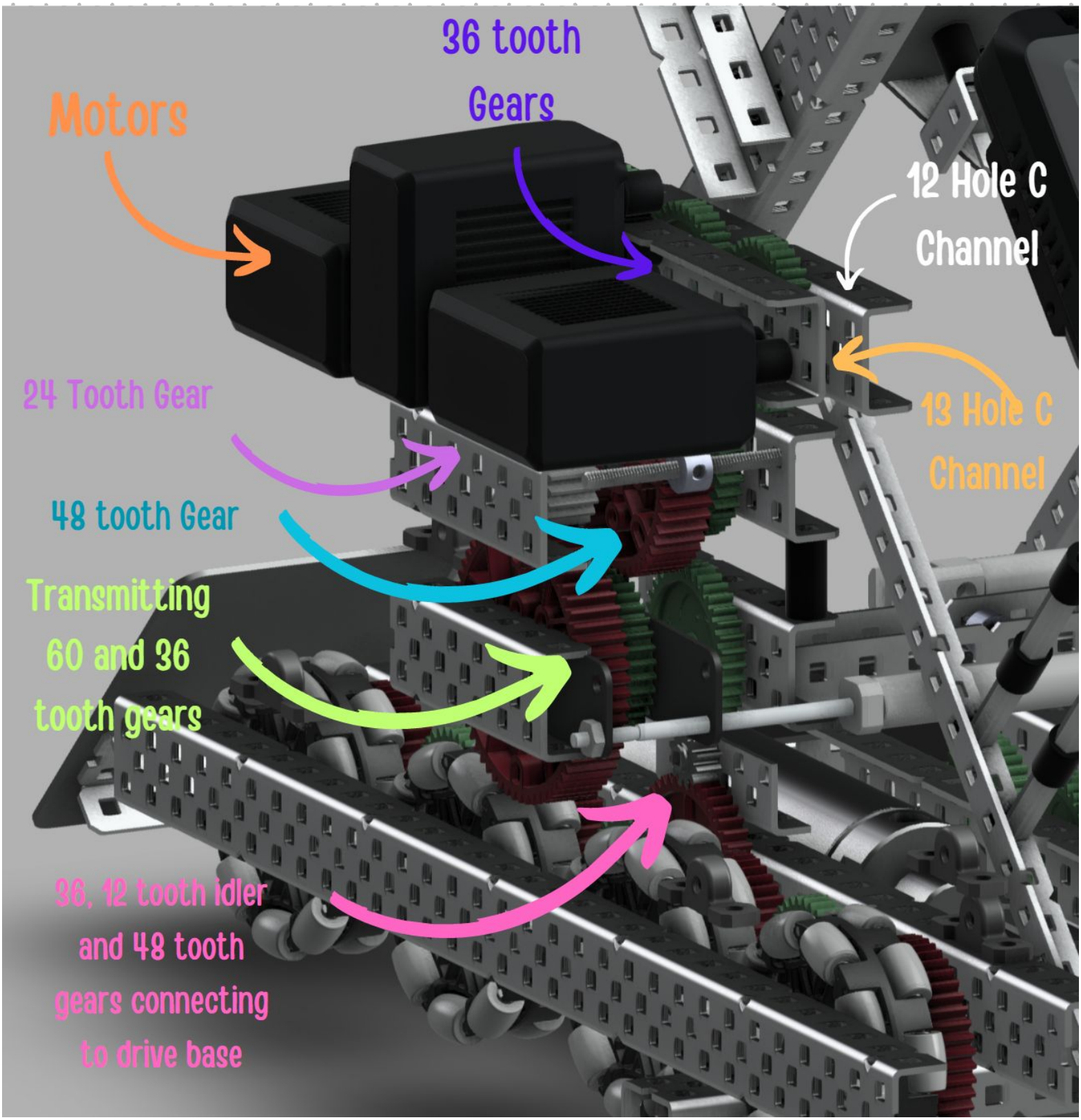
- Once we knew the basic design of the Transmission, I began to explore various gear ratios that would have matching heights for the transmission and what their output RPM's would be when interfaced with the robot
  - After exploring options we decided to go with the combination that would produce both an **output RPM of 135 (Would be quite slow but have tremendous torque) and 450 RPM (Would be Very Fast)**
- **For the bottom 2 layers, the 2 hole wide c channel was 6 holes wide (a bearing block was mounted in the exact middle, and therefore the axles/gears will be in the middle)**
  - In addition to the bottom layers we also created a layer 0.5 (aka "the special assembly") to hold a 12 tooth idler so that the transmitting gears would not interfere with the transmission wheels (**More details later**)
  - On the Drive base we mounted a 48 tooth gear, this gear connects to the 12 tooth idler on the special assembly, and then to a 36 tooth low profile gear (Layer 1) → this gear is on the same axle as the transmitting gears → in speed mode, a 36 tooth gear (layer 1) interfaces with a 48 tooth gear (layer 2) **this cancels out the gear ratio between the gears connecting layer 1 and the drive base** → in Torque mode, a 60 tooth gear (layer 1) interfaces with a 24 tooth (layer 2) **this works in conjunction to the gear ratio connecting the transmission to the drive base creating a final gear ratio of 3/10**
    - **Both the 1/1 and 3/10 gear ratios are applied to the gear ratio of the drive base (¾) and then multiplied by 600 to find the final output RPM's of 450 and 135**

Transmission Design

Design Process: CAD Creation

- **For layer 3 we mounted a 12 hole long 2 wide C Channel on the inner side → we then mounted 3 bearing blocks in the central row spaced apart appropriately for 3 36 tooth gears**
  - Then on the other side of the gears we mounted an offset **13 hole long c channe**l 0.5 inches away (enough space for the LP gears) and placed the bearing blocks on the bottom row
    - While quite odd, mounting the bearing blocks on the bottom from the second c channel and then offsetting it allowed us to attach the central motor more securely
      - **Motors were placed on the side of the 13 hole wide c channel**
        - While they do stick out over the drive base, they are high enough that it shouldn't be a major issue → additionally we can create a cage later to protect the motors
- **Special Assembly**
  - To mount the 12 tooth idler gear helping connect the transmission we mounted some angle pieces on each side of the drive base which connected via standoffs to a especially cut flat piece with space for a bearing block and holes for the standoffs
    - **This will be seen in more details in the pictures when the robot is constructed**
- **We also designed the custom pieces that would connect to the pneumatics to push the transmission gears back and forth**
  - **To mount these pistons we measured in the cad the correct standoff lengths**

Transmission Design



Project Transmission Design

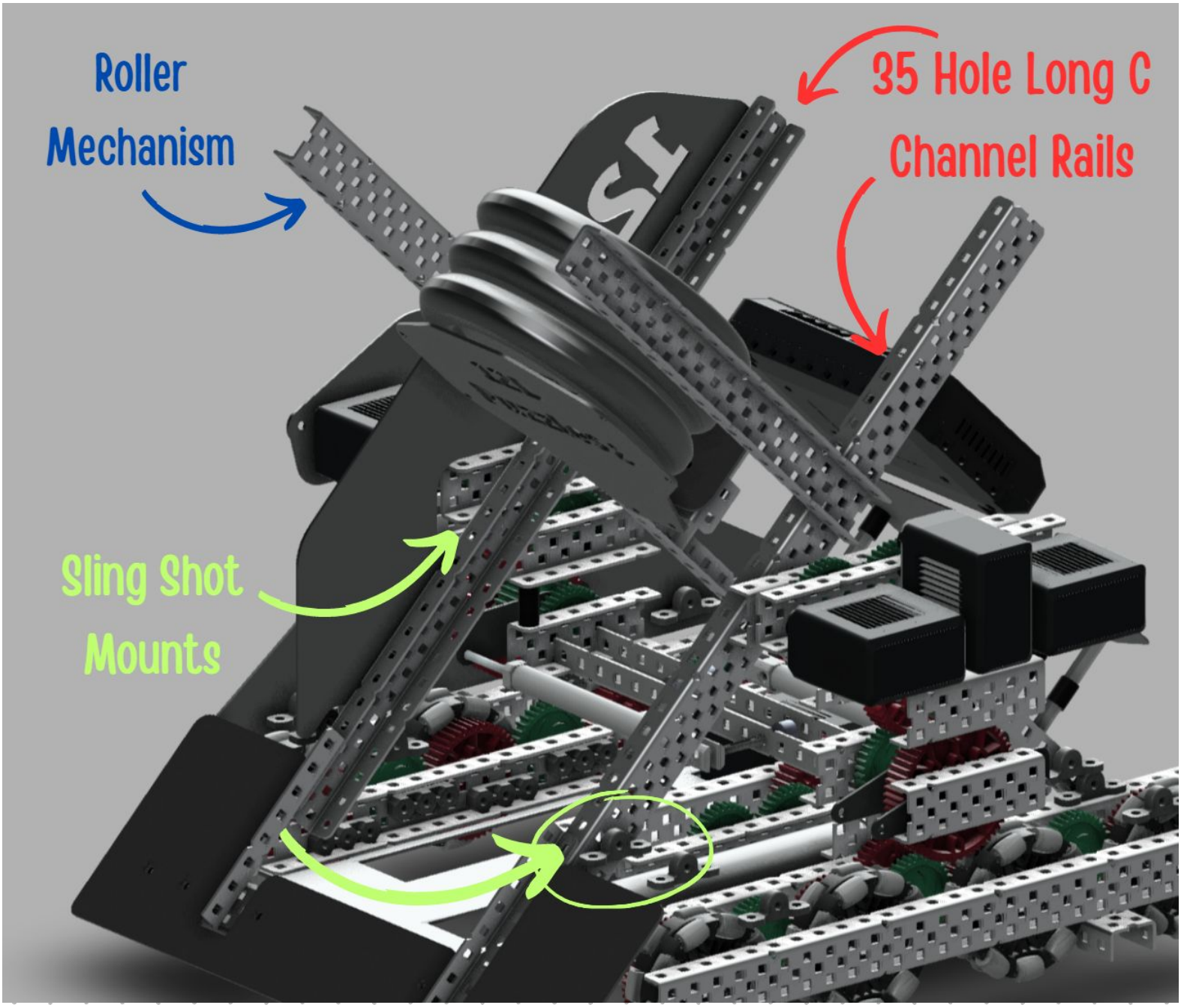
Name Ben Runde Date 1/3 Page 24

Basic Sling Shot Plan and Design

Design Process: Preliminary Design

- Currently, from the slingshots at Potomac, there seem 2 be 2 design philosophies for the slingshots, specifically the rails
  1. **Friction optimized (12X)** → This design philosophy uses several connected standoffs for the rails that are attached to a c channel pillar at the front, the sled has 2 holes drilled into it and has 2 high strength bearing blocks installed so that there is no metal on metal contact
  2. **Weight/Space Optimized (12H)** → this design uses a long c channel as the rail (allowing for easy mounting of accessories and no extra supports
     a. **We decided to follow this approach as we already had the excess weight from the transmission**
- **Once We knew the design Philosophy we began designing**
  - We mounted 2 30 hole long 2 wide c channel at approximately 45 degrees for optimal range
    - These c channels are mounted to a pillow block at back and bottom of the robot and additionally are connected to one of the Transmission c channels
- **We decided not to CAD the sled as it is not exact enough to accurately create the geometry and precision for it**
- **In addition to the Slingshot rails we also estimated and mounted some c channels in the primary position to hold the roller mechanism for the 4 wallr rollers on the field.**
  - **They will allow us to make good contact with the rollers**

Basic Sling Shot Plan and Design



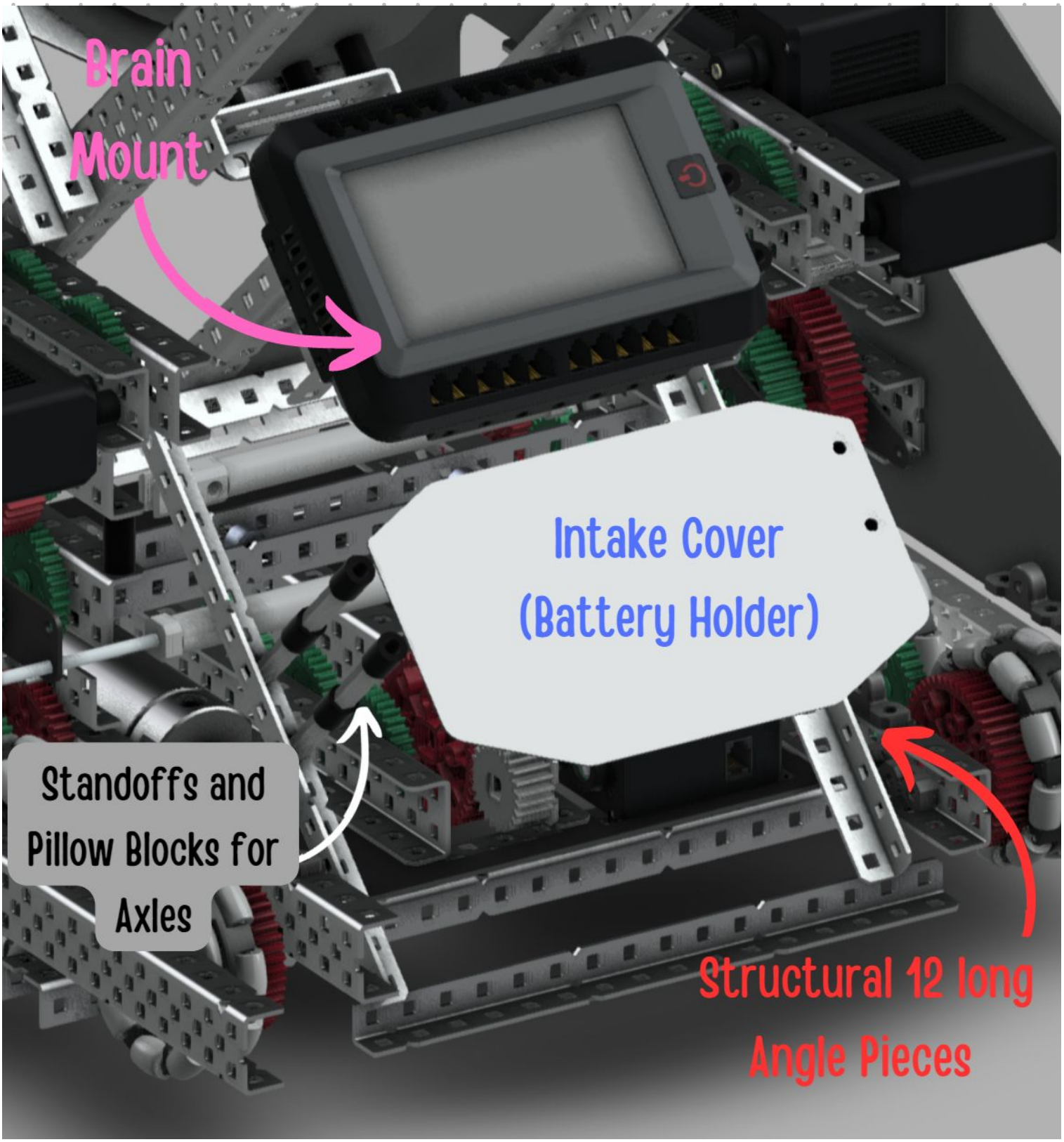
Project Sling Design
Name Ben Runde
Date 1/3

Intake Design

Design Process: Design and CAD

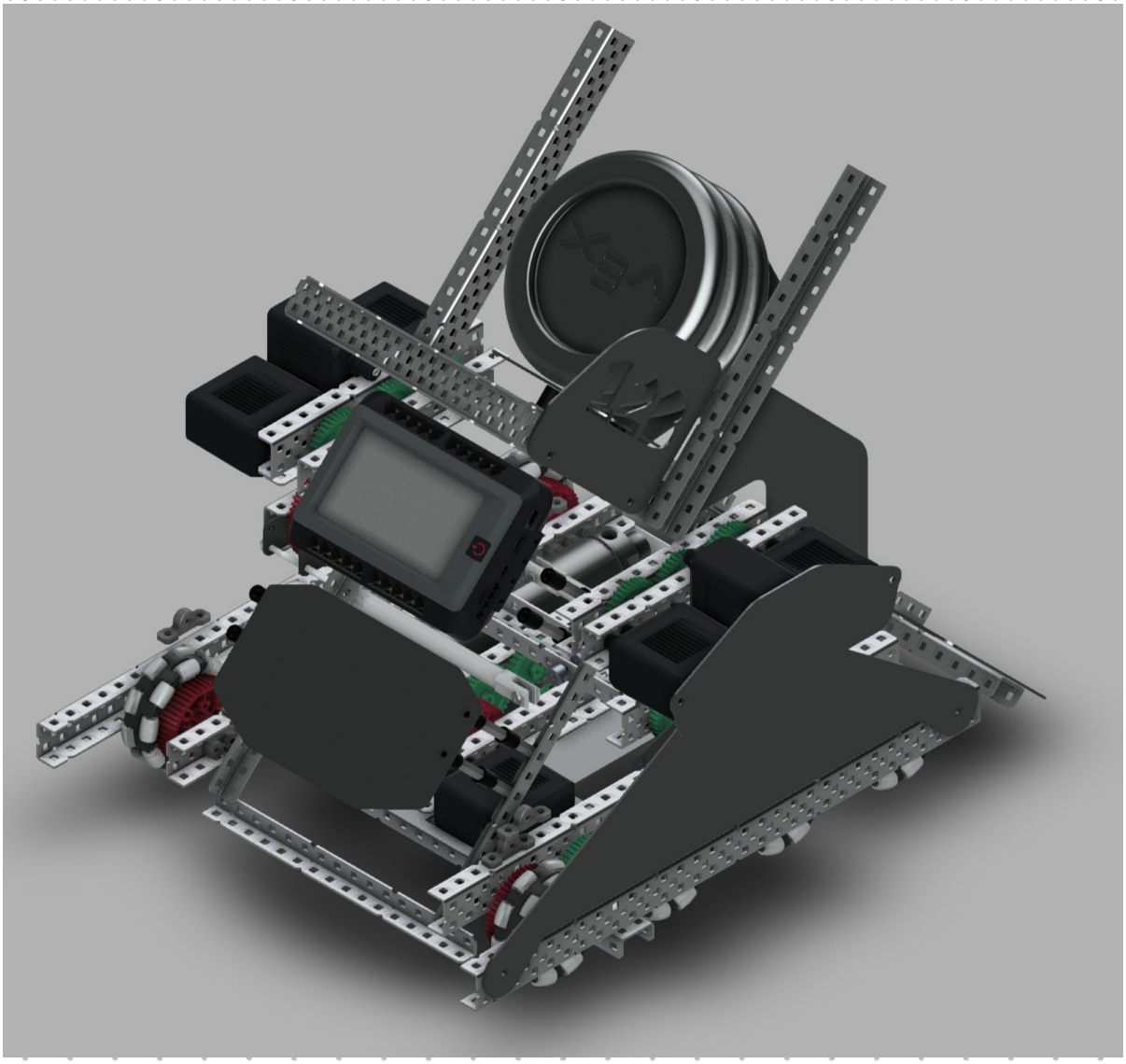
- Once we had the drive base, transmission, and slingshot rails mounted on the cad we began the construction of our intake
  - The Intake is quite simple, its based on 2 12 long 1 wide angle pieces on each end of the intake
    - We decided to use various sets of 1 inch flex wheels for our intake with a pivoting first level to make the initial intake easier
      - These flex wheels will be mounted on axles attached to pillow blocks connected to standoffs on the angle pieces
        - We chose to pick arbitrary spacing and pressure for the flex wheel for now, and we will adjust as we build to optimize
- In addition we mounted 2 other angle pieces on top of the standoffs and pillow blocks so we can mount our brain and battery on top of the intake and use space efficiently

Intake Design



General Design Overview

- **In summary we have designed a slingshot based robot, with a medium size width, an omni/traction combo drive base, and a transmission with a high torque (135 RPM) and a high speed mode (450 RPM)**



Mid Season Kick-Off/Start

January

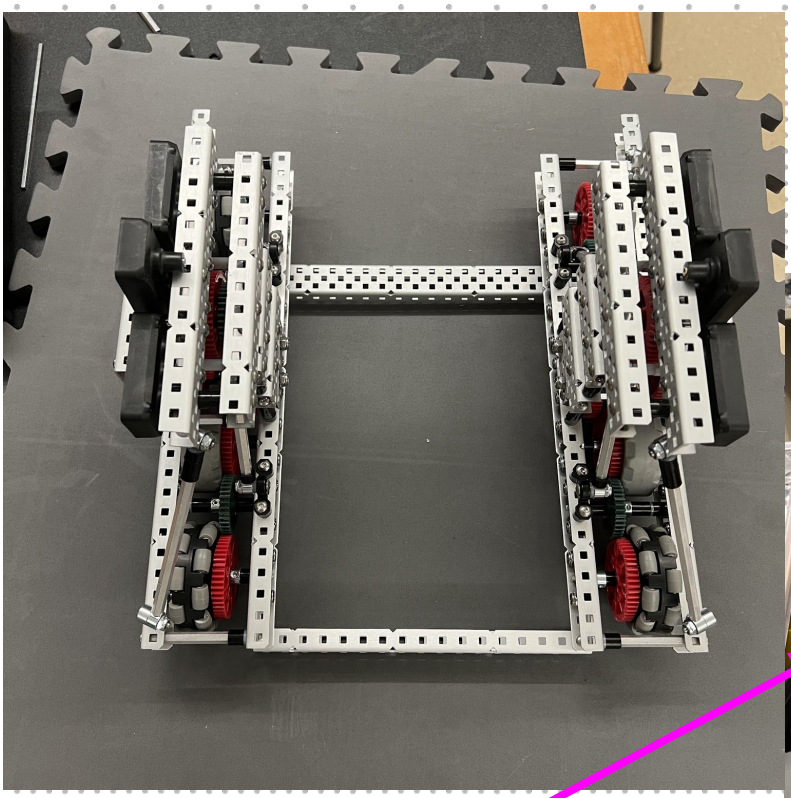
1	N/A
2	N/A
3	Drive Base
4	Drive Base
5	Drive Base
6	Transmission
7	N/A
8	N/A
9	Transmission
10	Transmission
11	Transmission
12	Intake
13	Intake and Corrected Transmission
14	N/A
15	Finishing Intake and Starting Slingshot
16	N/A
17	N/A
18	Sled and Roller Mechanism
19	N/A (Kalahari Signature Event)
20	N/A (Kalahari Signature Event)
21	N/A (Kalahari Signature Event)
22	N/A (Kalahari Signature Event)
23	N/A
24	Planning Skills Route; Sled and Baseplate
25	N/A
26	N/A (Practiced Canceled)
27	N/A (Practiced Canceled)
28	N/A
29	Sled Testing and Winch
30	Cleanup and Organization
31	Finishing Up Prototype Slingshot

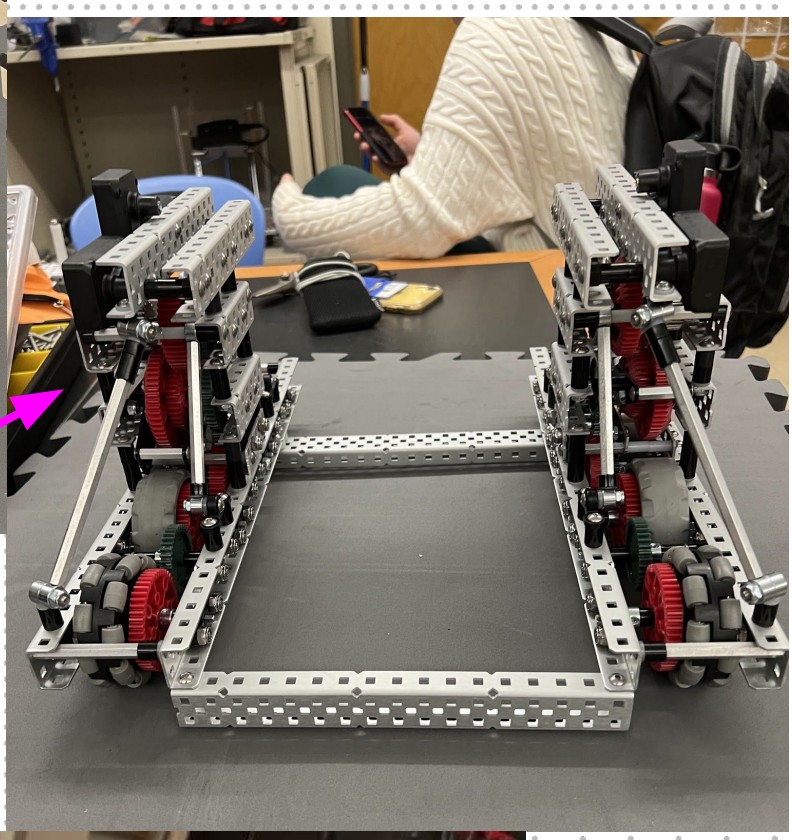
Drive Base Construction

- **The construction process for the drive base was very efficient, only taking 2 hours to complete.**
  - Thanks to the CAD, I was able to quickly and efficiently screw parts together without the need for lots of testing
  - The wheels are practically free spinning (only a very minor amount of friction) which will be helpful in reducing the total amount of friction between the motors and the wheel due to the extra gears required for a transmission

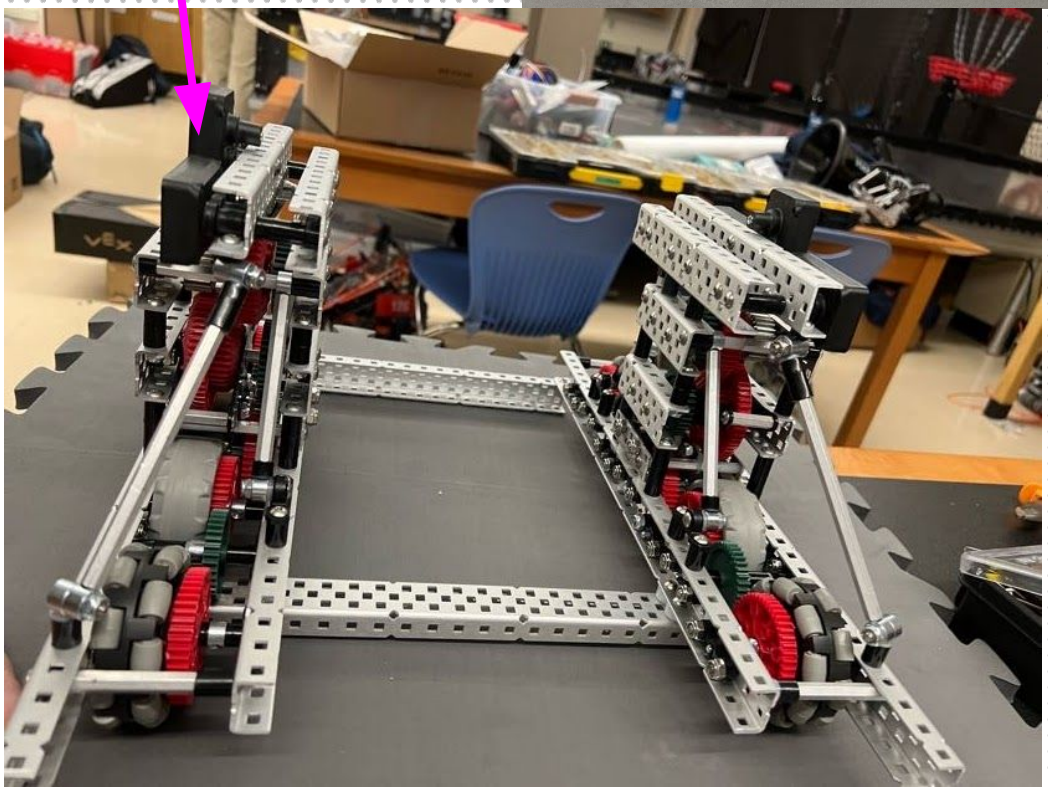
Transmission Construction

- The Construction of the Transmissions structure was completed soon after…
  - Once again, the CAD was extremely for getting the right spacing without requiring multiple different attempts (all of which would have required assembly and disassembly) saving lots of time
  - Similarly the transmission gears and spacing also was very easy because it was already figured out in the CAD → **the CAD makes assembly as easy as building a LEGO set**



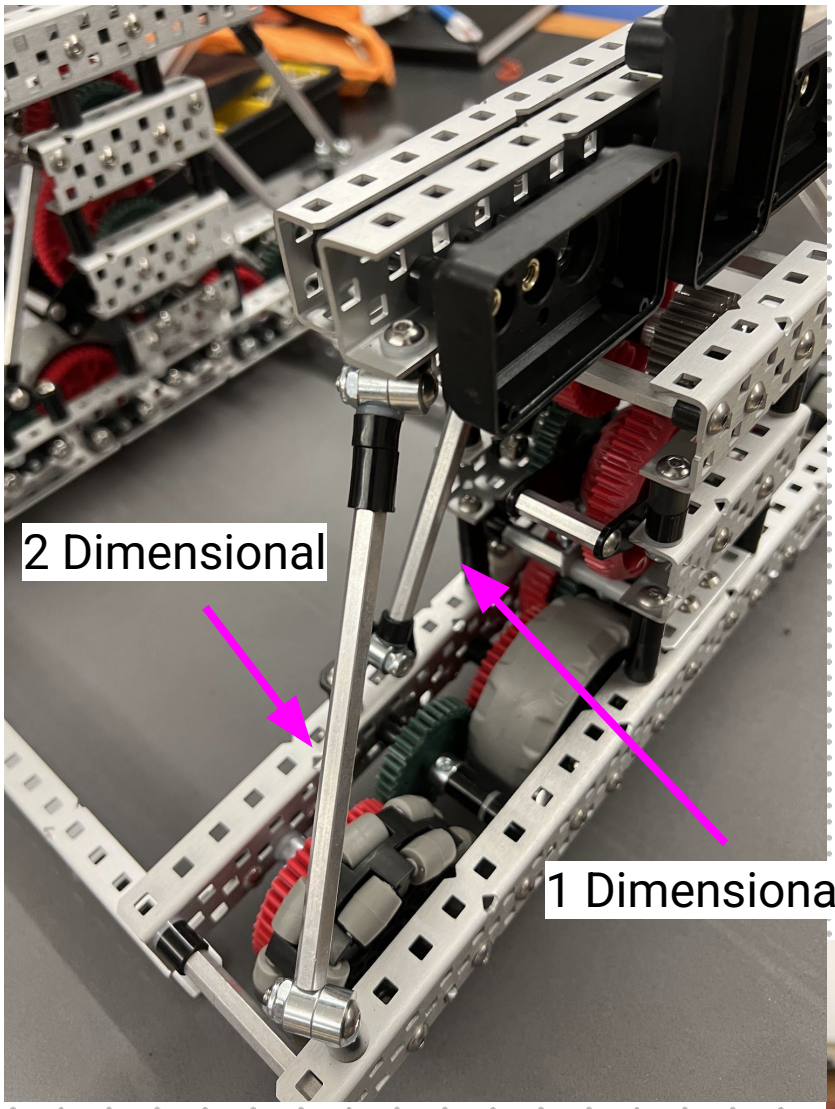


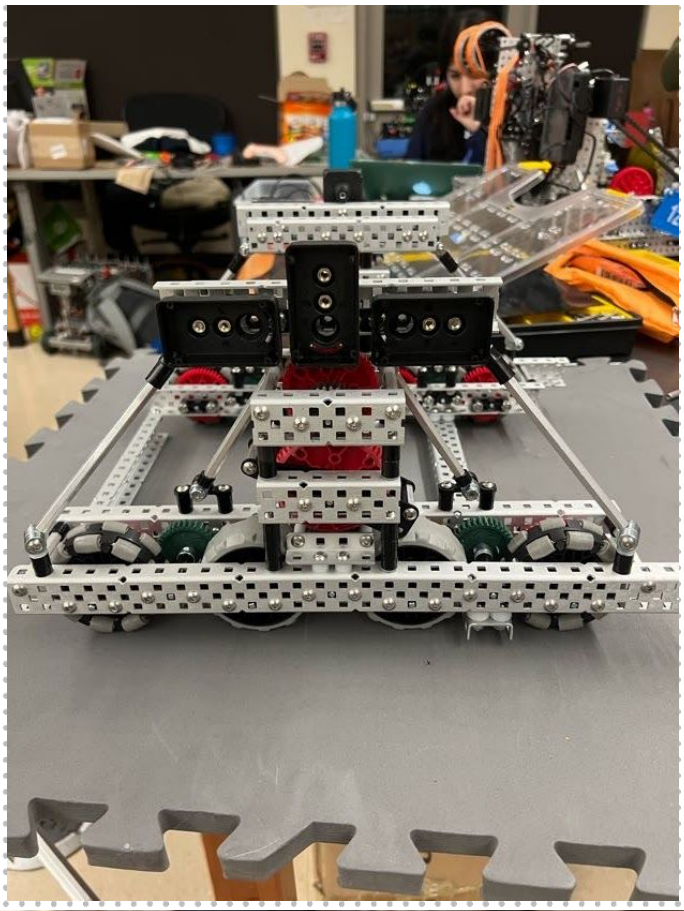
Transmission Towers

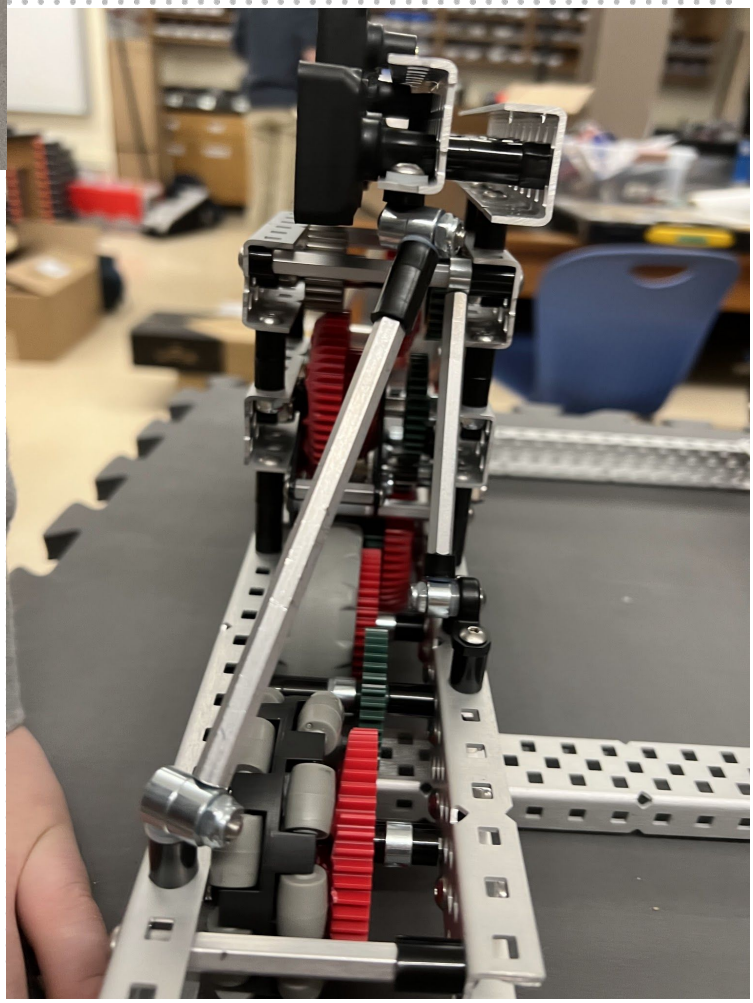


Project Drive Base and Transmission Structure Done

Name Ben Date 1/10 Page 32





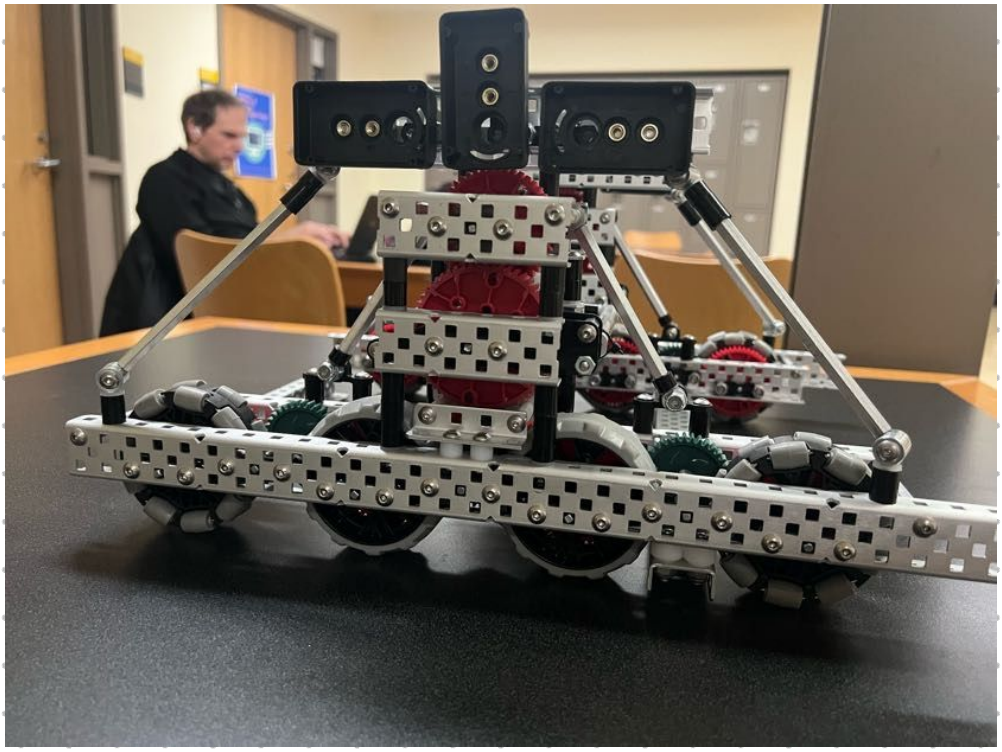


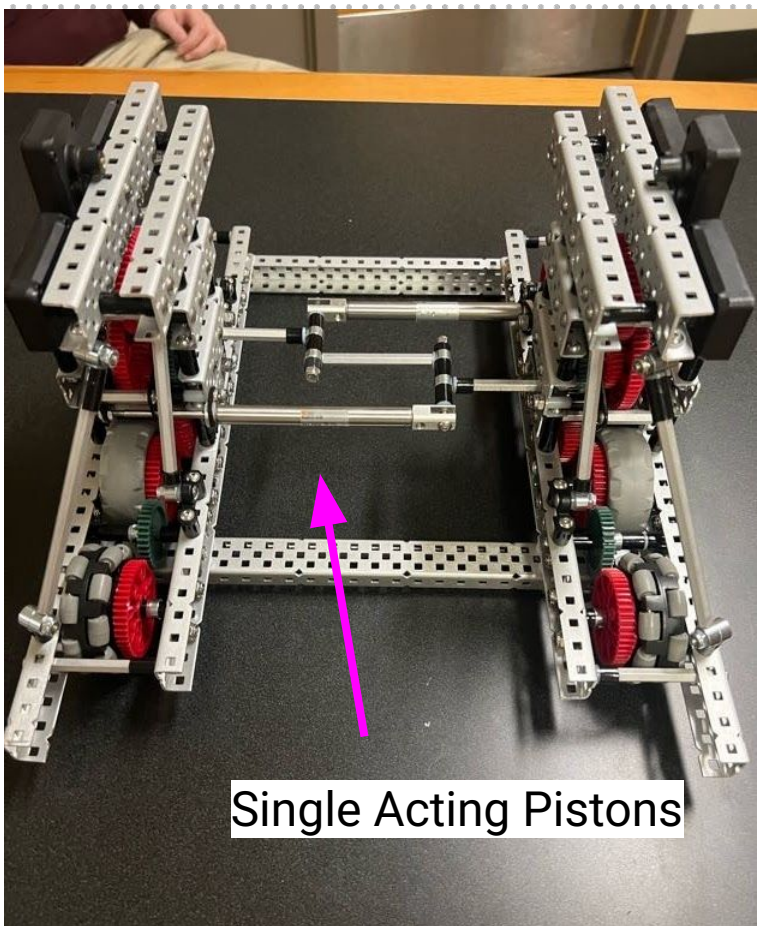
Transmission Tower Supports

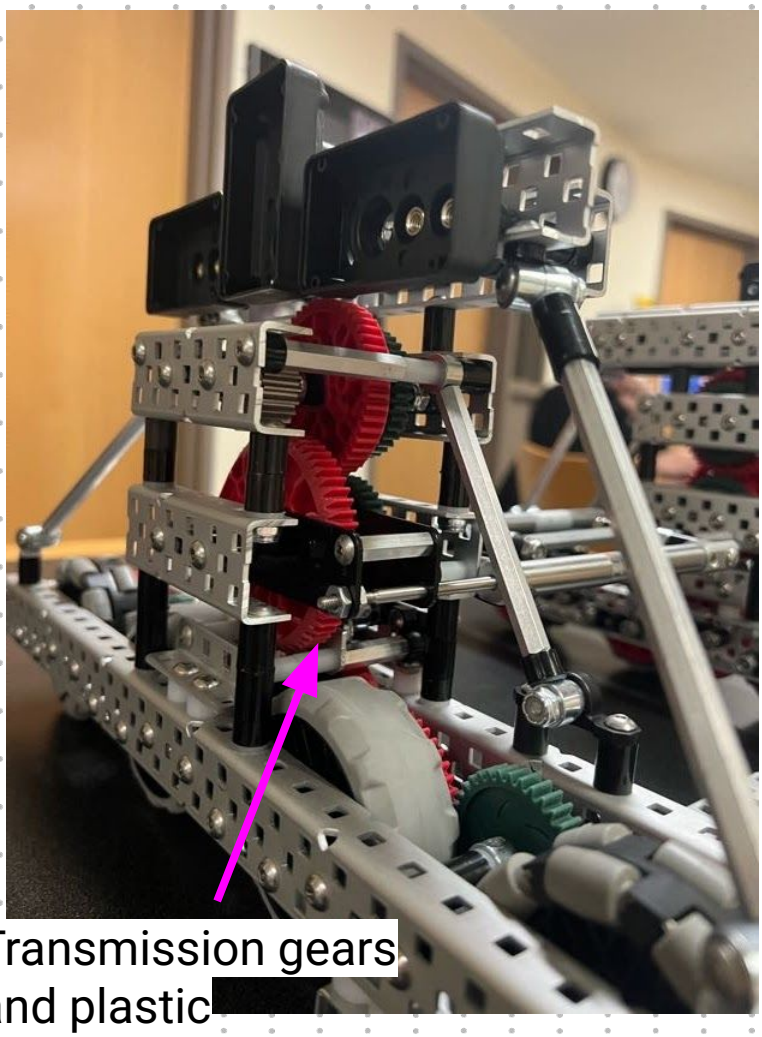
Due to the fact that the towers are only attached vertically by 2 screws they were inherently weak in both lateral directions.To fix this we put 2 types of lateral supports (1 Dimensional and 2 dimensional). With the supports the transmission tower is extremely solid.

Transmission Completed

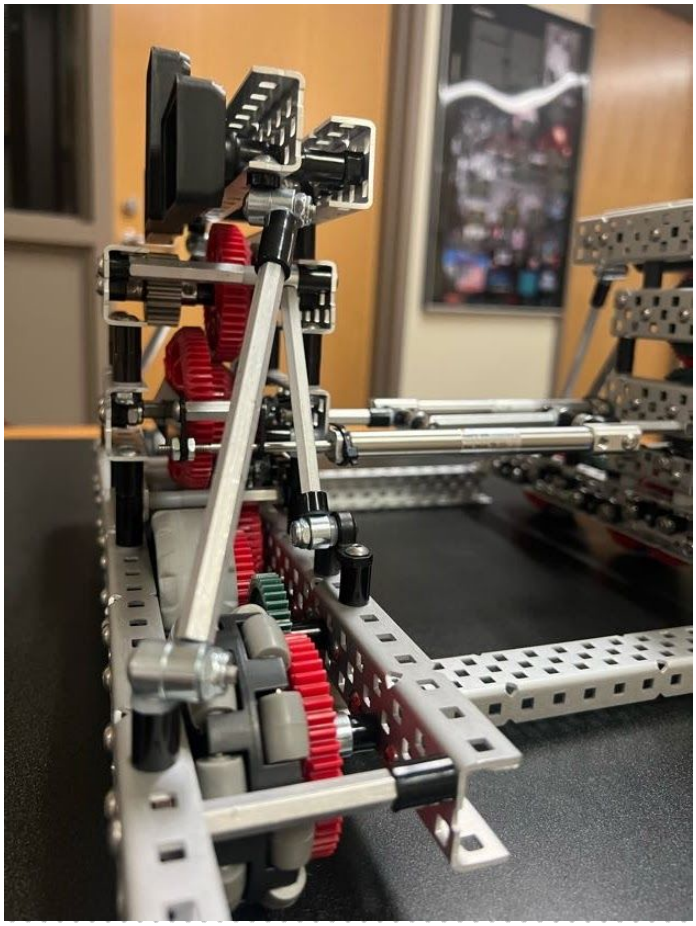
- Today we completed the installation of the pneumatic single acting pistons for the transmission
  - While we did include these pistons in the CAD the real life dimensions were slightly different and required some fine tuning
  - While preliminary tests showed that the pistons would work effectively with the pneumatics some more thorough examination showed that the single acting pistons tended to get a little stuck
    - **This means that further examination of the Transmission will be required**

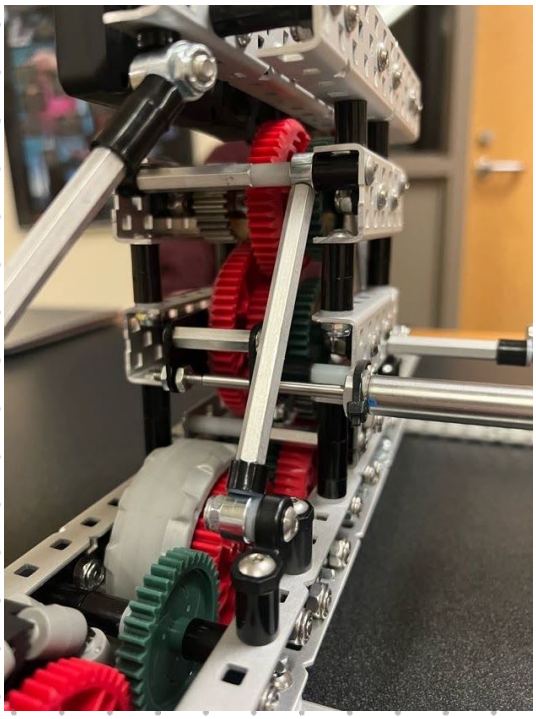






Transmission gears and plastic [redacted] transmission “piece”



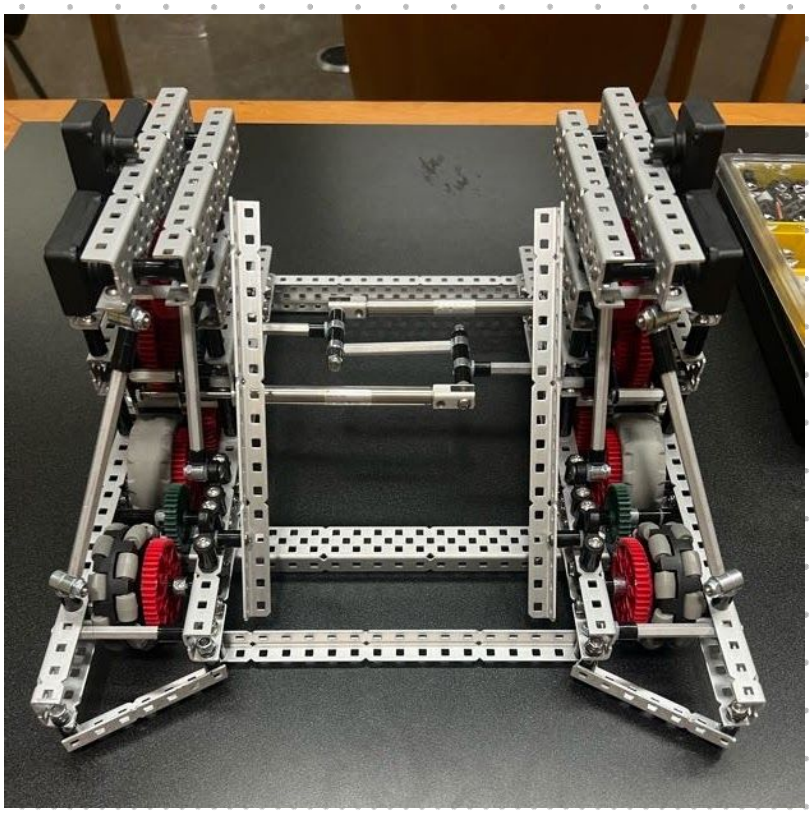


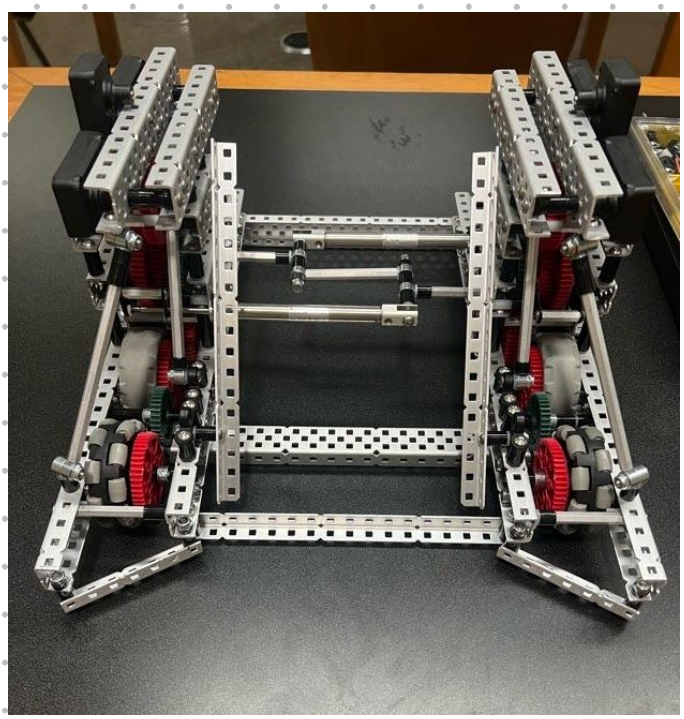
Project Transmission Done

Name Ben Date 1/11 Page 35

Starting The Intake

- “Switching gears” (figuratively speaking of course ;) ;) since we will be literally switching gears with this robot) we decided to begin working on the intake
- The first thing we did was start the “funnel” assembly
  - This is made up of the angle pieces at the front of the robot
    - Its function is to funnel disks into the intake
- Then we began working on the actual intake structure
  - We slightly adjusted its position from the cad due to interference with the transmission structure but it still is in roughly the same position



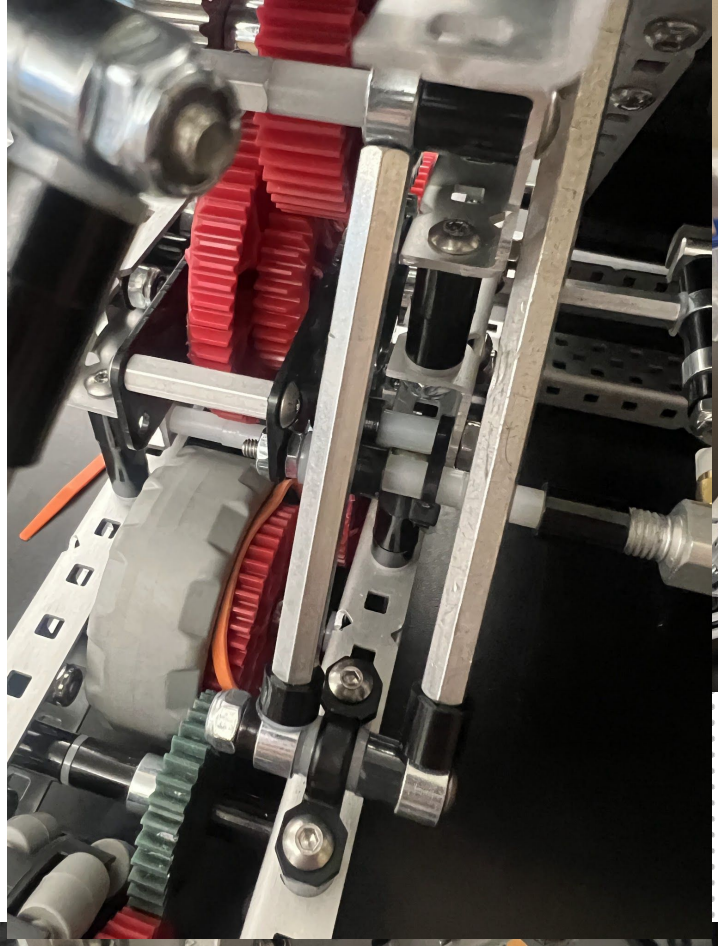


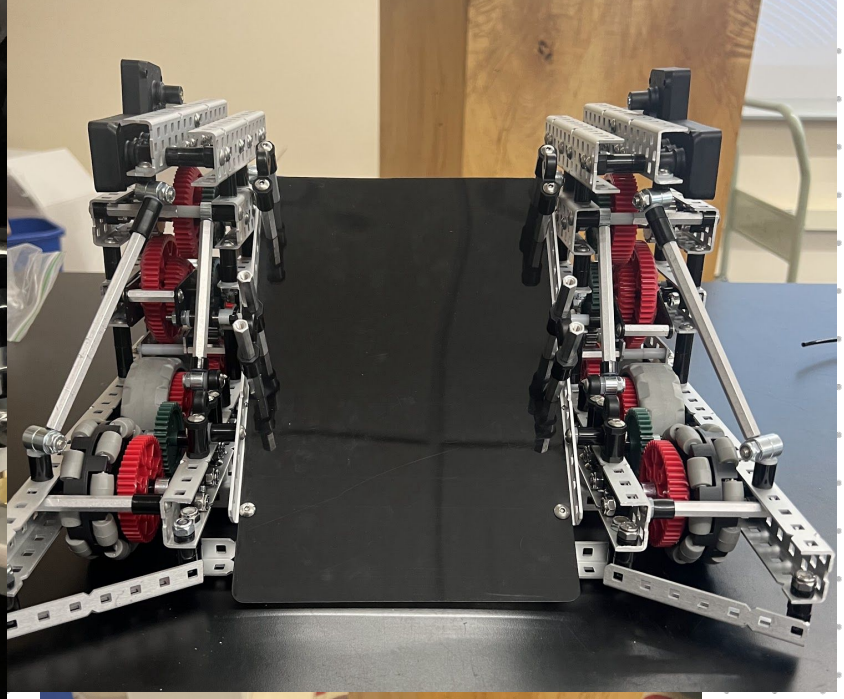
Intake Construction

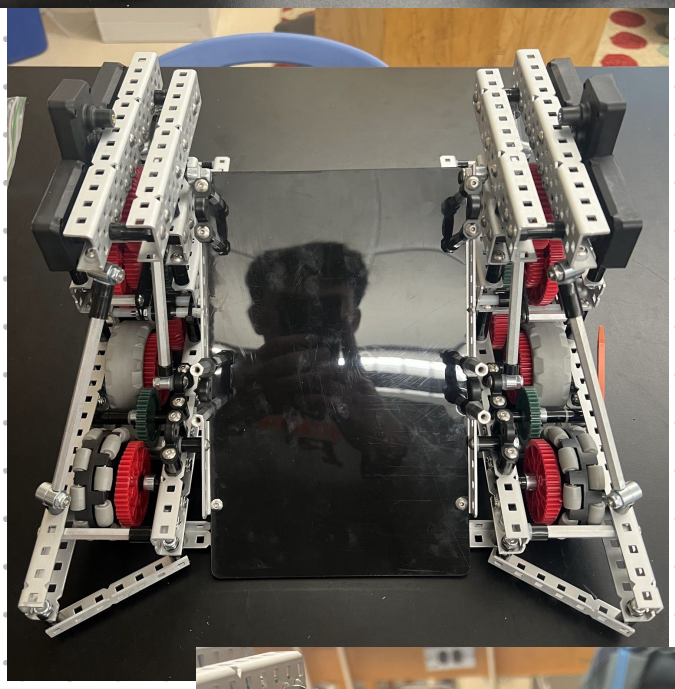
- We continued working on the intake today. We laser cut a delrin piece to serve as the base and screwed in some supports
  - Additionally we mounted pillow blocks for the intake rollers. These pillow blocks are mounted on spacers to allow custom spacings to optimize

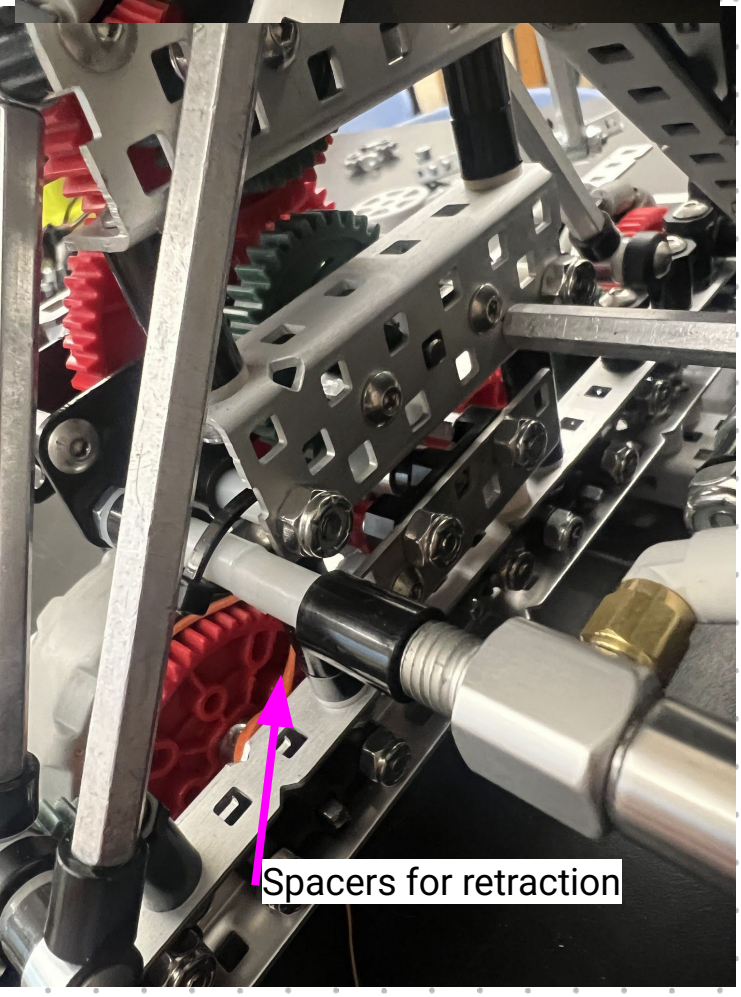
Transmission Testing

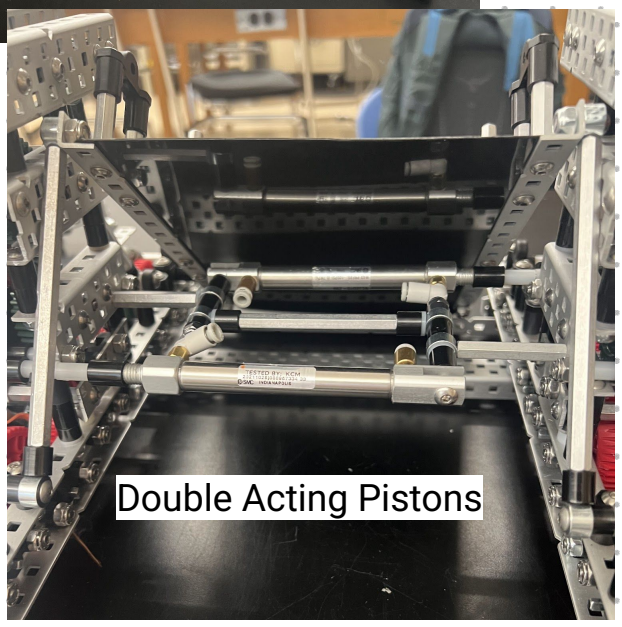
- **As stated in a previous entry more rigorous testing is required to further examine the consistency and effectiveness of the single acting pneumatic piston transmission**
- We first started by wiring up the pneumatic pistons to a makeshift air source
  - This test showed that the transmission was to shift outwards but not inwards → in other words it was able to go from a retracted position to an extended one, but the springs inside the piston could not bring it back into a retracted position
- **We then tried applying rubber bands to the transmission, but this was unsuccessful**
  - It could return to a retracted position but the plastic piece would bend on the side with the rubber band when extended, which did not allow the gears to fully engage in that position
- **This forced us to use double acting pistons**
  - We would have preferred to sue single acting pistons, due to them requiring left air, but we were left with no options
  - We then changed the spacing to factor in the smaller bore length of double acting pistons and placed spacers over a decent length of the “extendy” (for lack of a better term) part of the piston to stop it from retracting too far and breaking the plastic transmission pieces











Project Intake Construction and Transmission Testing

Name David Gardner Date 1/13 Page 38

Team Meeting:
Week In Review Takeaways

The first week working on a new robot was an incredible success. So far, each element of the CAD has worked as an effective blueprint. For the drive base and transmission towers, we were able to complete them within three days (6 hours), while the intake and double acting pistons took two days (3 hours) to complete. The robot looks very sleek and has not caused any significant problems so far. However, as we continue to add in parts and testing functionality we assume some problems may arise.

Next week, our goal is to complete the intake and begin the slingshot (the bulk of the project). Additionally, we will compete at Kalahari Signature event with our old robot. This will be used as a learning experience.

Time in Lab:
Monday - Friday, Sunday; 12 Hours

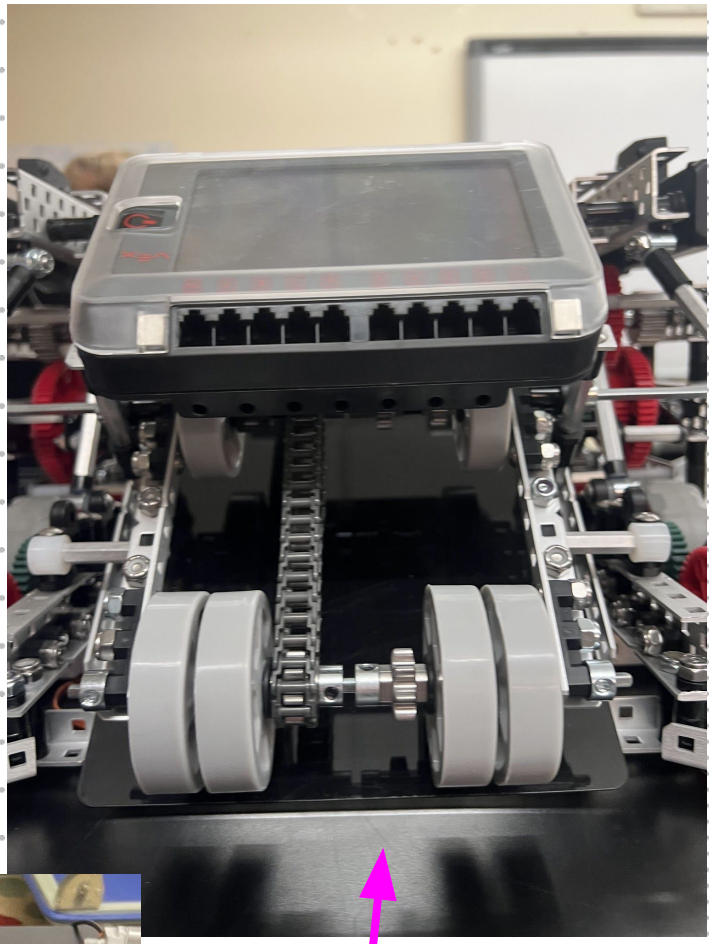
Intake Completion

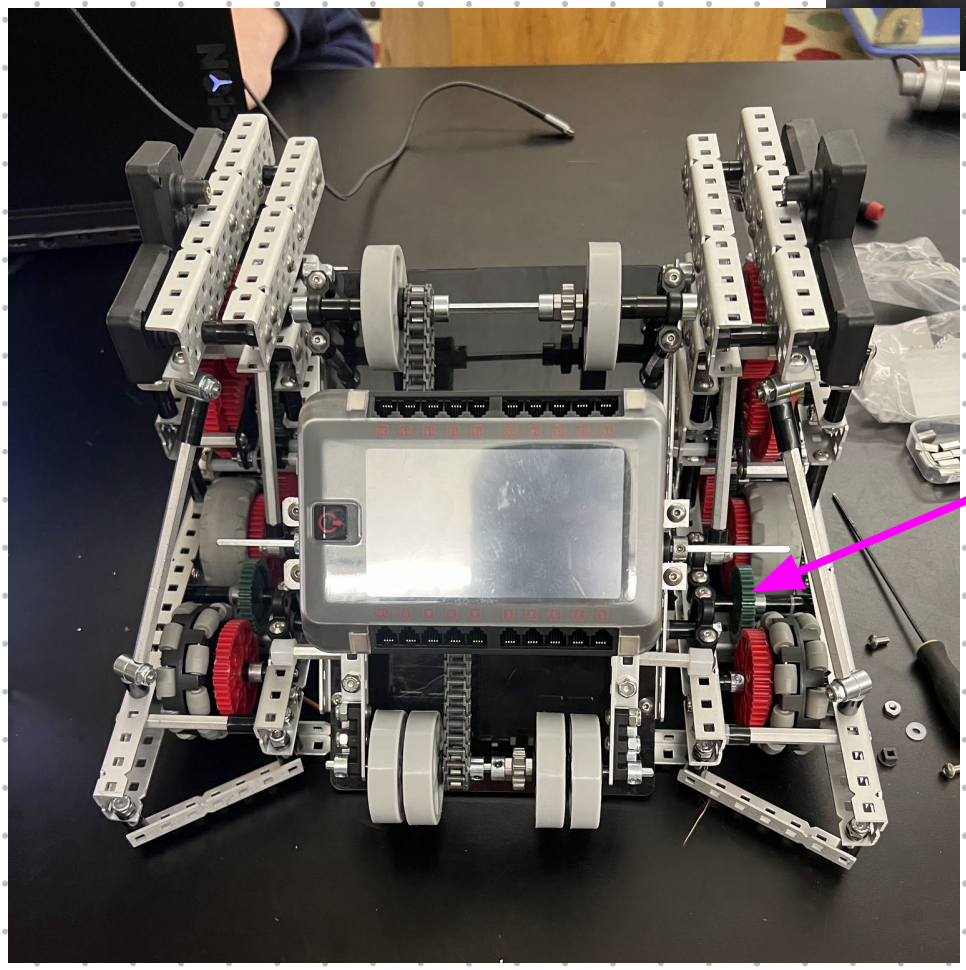
- Today we finished the construction of the intake
  - All that is left is to connect it to a motor
    - We are waiting on finishing the roller mechanism (which itself requires the completion of the slingshot since it is mounted upon it) to install this motor since they will be connected together
  - We tested it manually and it appears to work
    - If it does not work with an actual motor… we could easily change the spacing of the rubber wheels because its mounted on pillow blocks rathan a separate c-channel

Starting the Slingshot

- Following the instructions of the CAD, we attached 2 c-channel in their designated positions to act as rails for the slingshot
- We then rapidly constructed a simple sled for testing purposes
  - Additionally we have a signature event next week (kalahari) and do not have enough time to perfect it yet
- In order to reduce friction between the sled and the “rails” (c-channel) we attempted to screw screws with large freespining spacers to reduce almost all friction
  - This however was unsuccessful do to the uneven surface created by all the round spaces and have decided to lay a flat strip of plastic instead to reduce friction
- On another note, we attached the c-channel that will hold our roller mechanism to the robot

Simple Sled

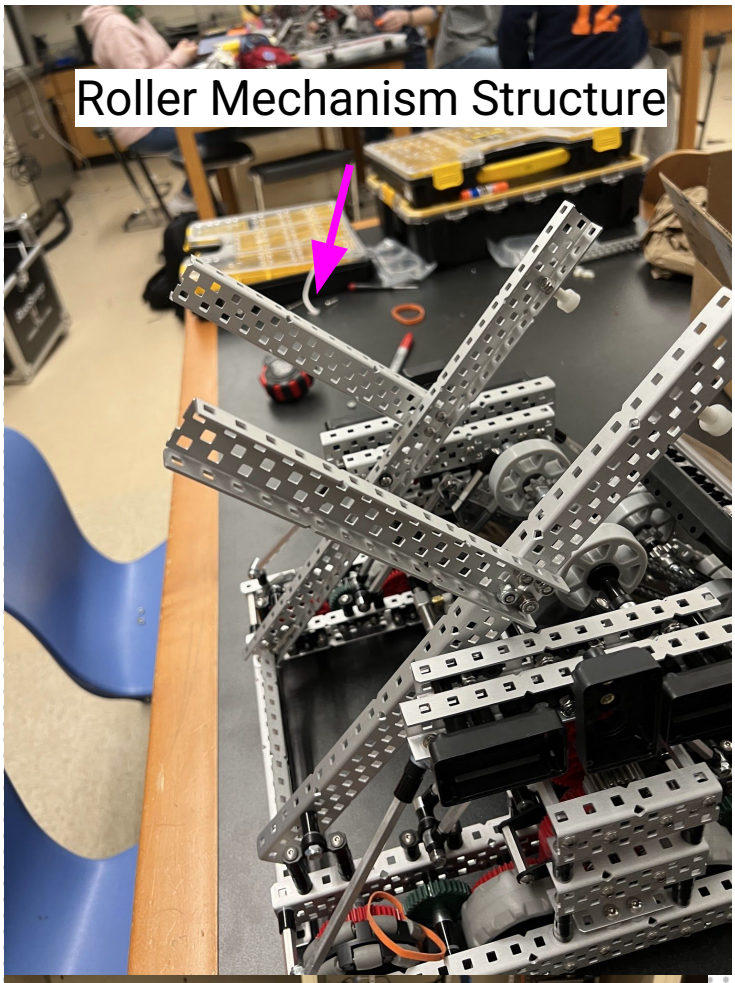


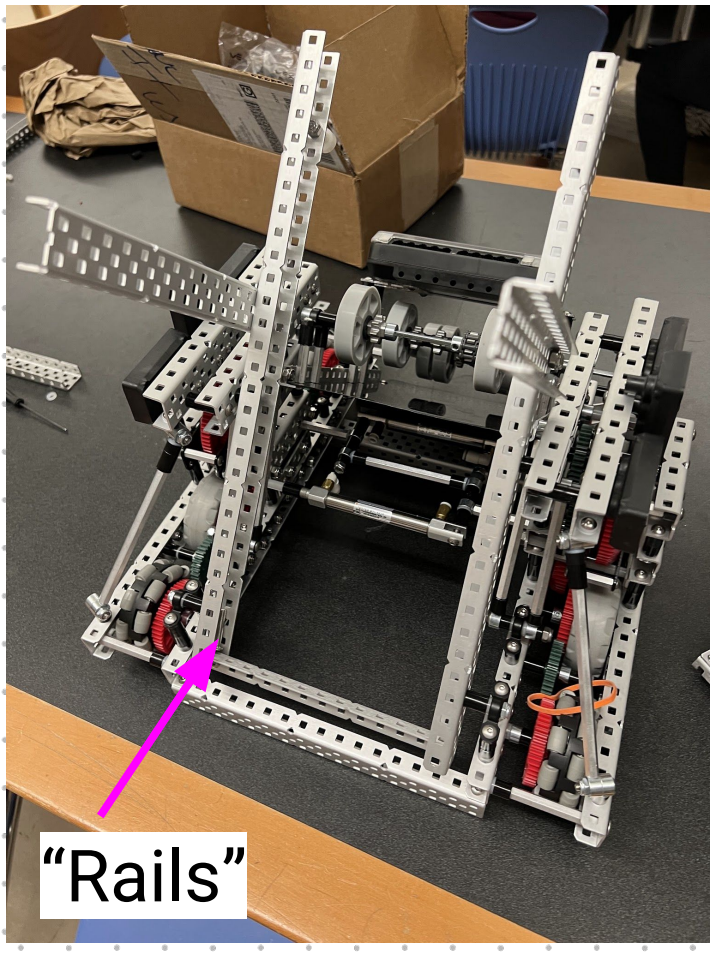


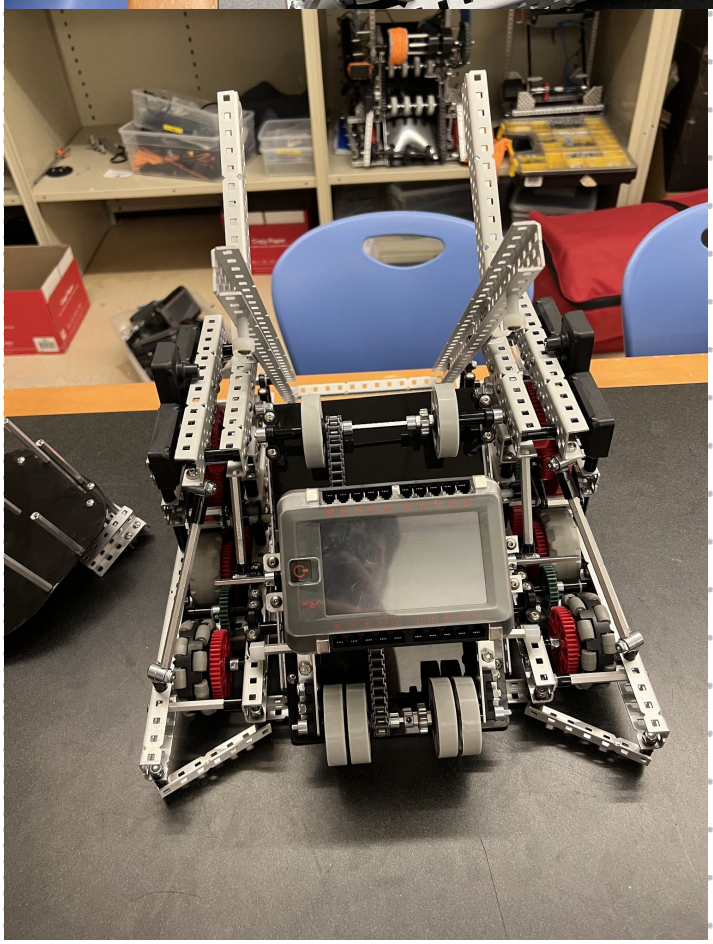
Intake

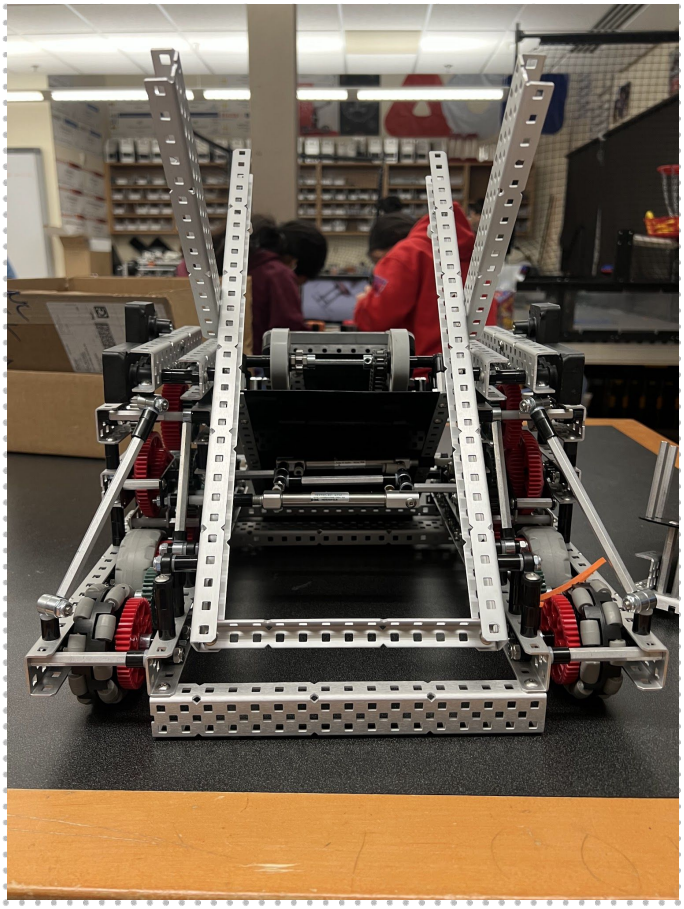
Project Finishing Intake and Starting Slingshot

Name David Gardner Date 1/15 Page 41









Project Finishing Intake and Starting Slingshot

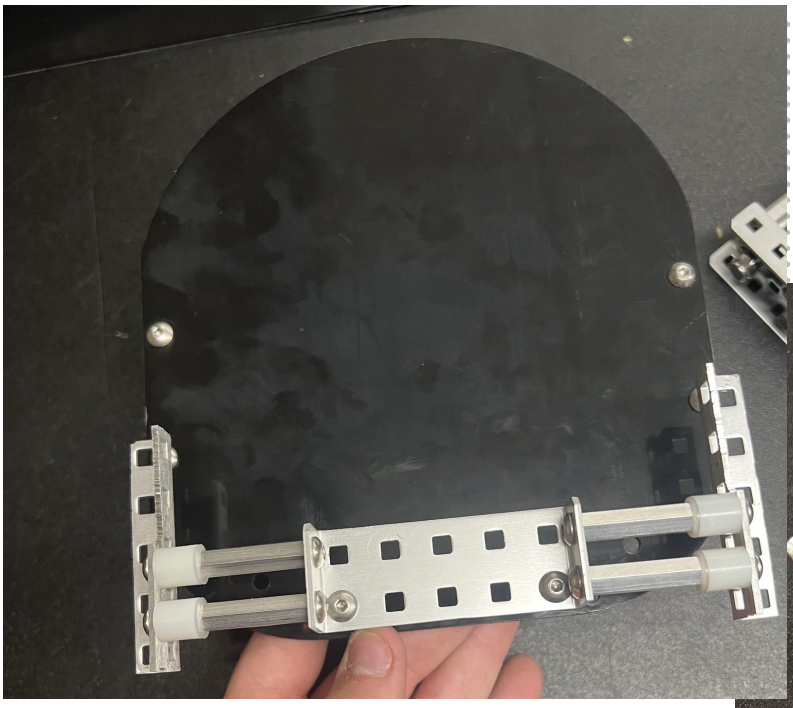
Name Ben Runde Date 1/15 Page 42

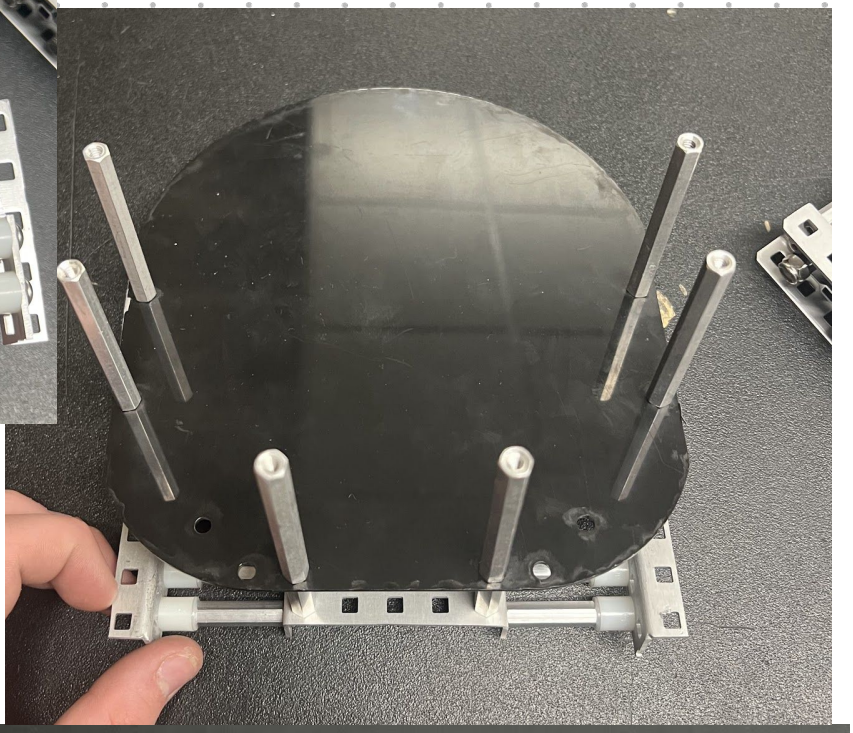
New Sled

- Due to complications with the intake, we have decided to change our sled and make its rail part (as in the part that slides on the long c-channel) lower profile and slimmer
  - Now the new rail mechanism for the sled does not stick out under the c channel unlike before
  - It uses a angle pieces on the side, in conjunction with a 2 hole long, 5 wide channel in the middle to mount the delrin disk holder.
  - Due to it only having only one side holding onto the rail we will have to modify it slightly in the future so that it doesn't derail

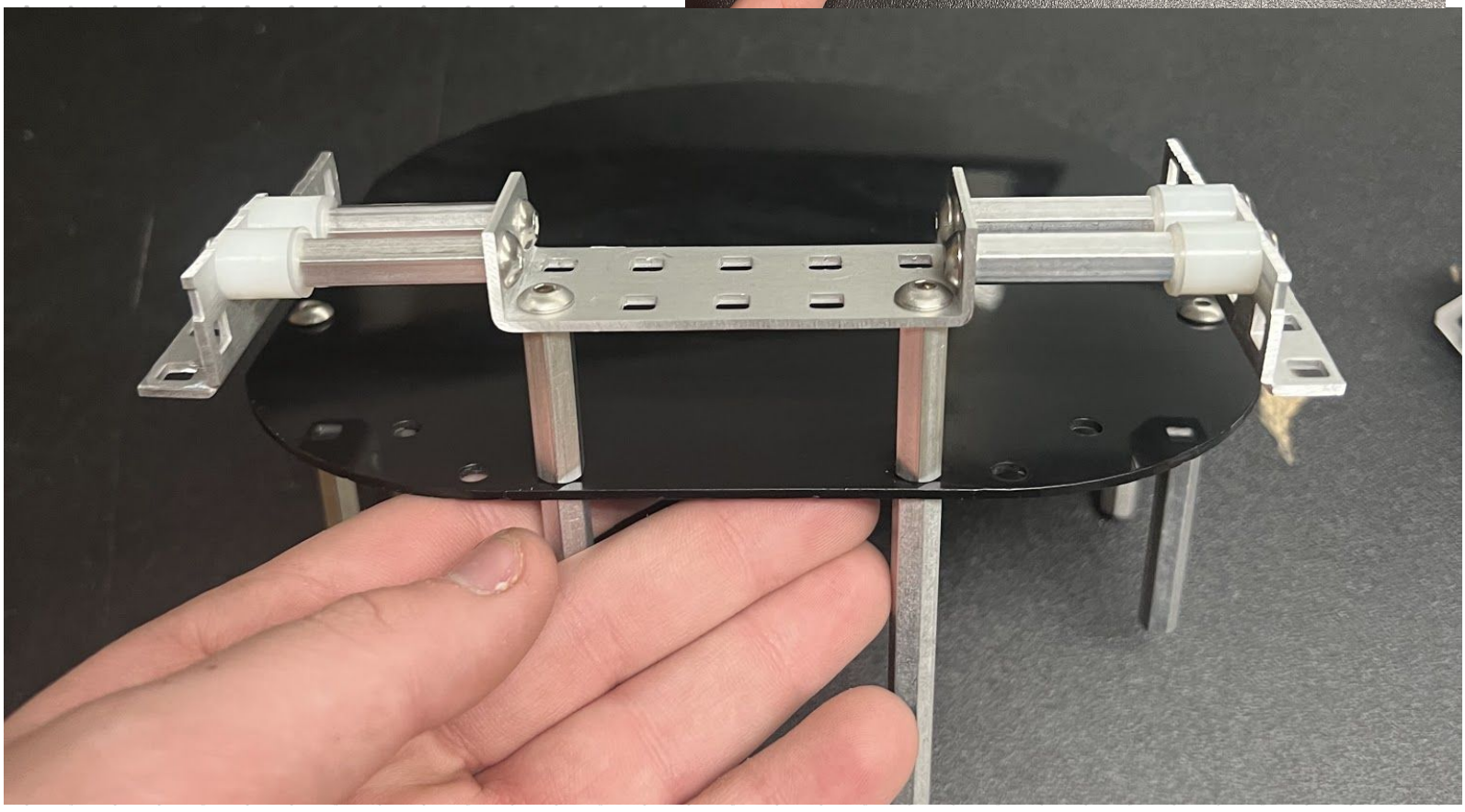
Starting Roller Mechanism

- Additionally we planned out and began installing pillow blocks for out roller mechanism and the motor connected to it and the intake
  - Through a combinations of axles and sprockets we plan to mount the motor in between the intake and roller mechanism and have it connected to both.



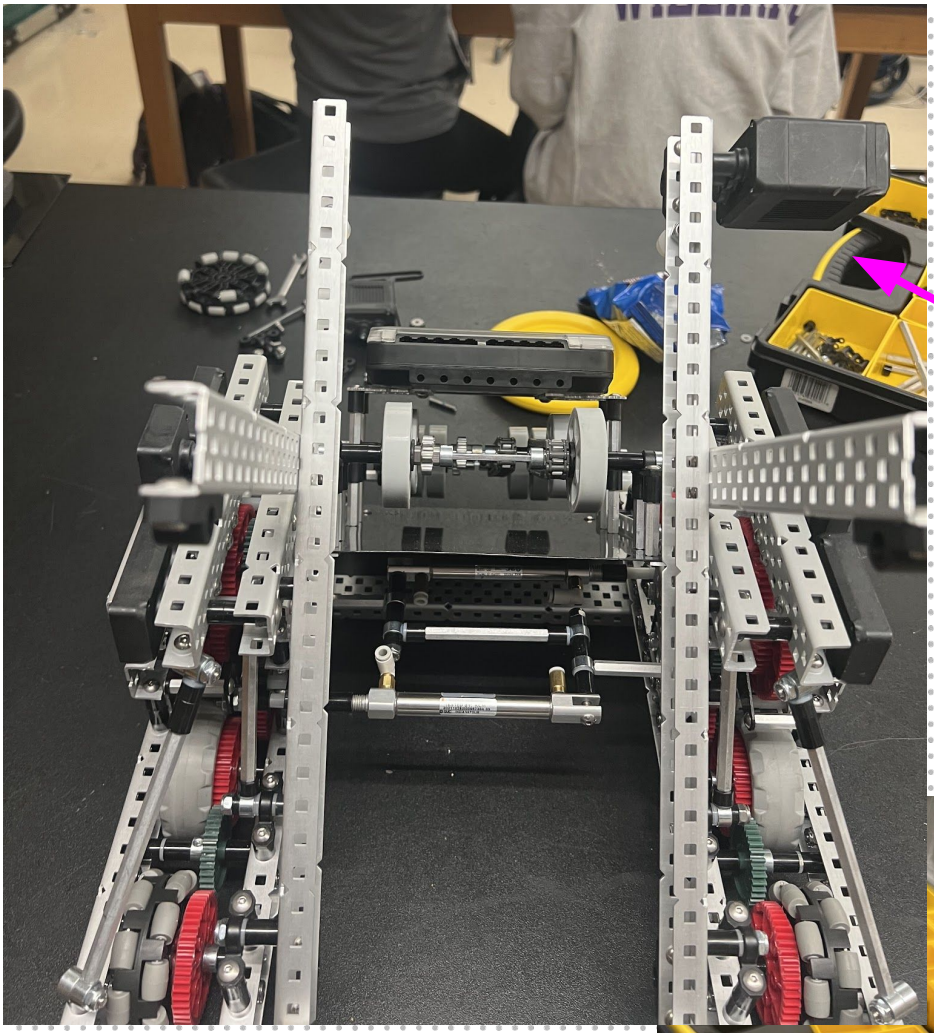


Sled/Rail Mechanism



Project Sled and Roller Mechanism

Name Ben Date 1/18 Page 44



Intake and Roller Mechanism Motor

Project Sled and Roller Mechanism

Name Ben Date 1/18

Kalahari Classic

Kalahari was an incredible experience. As our first signature event, we went in not knowing what to expect. Despite having to take our old robot, an outdated BLZ-I flywheel, we had an amazing time at the waterpark, and learned an incredible amount about the strategies and builds that the top-tier teams are using.

Team Meeting:
Week In Review Takeaways

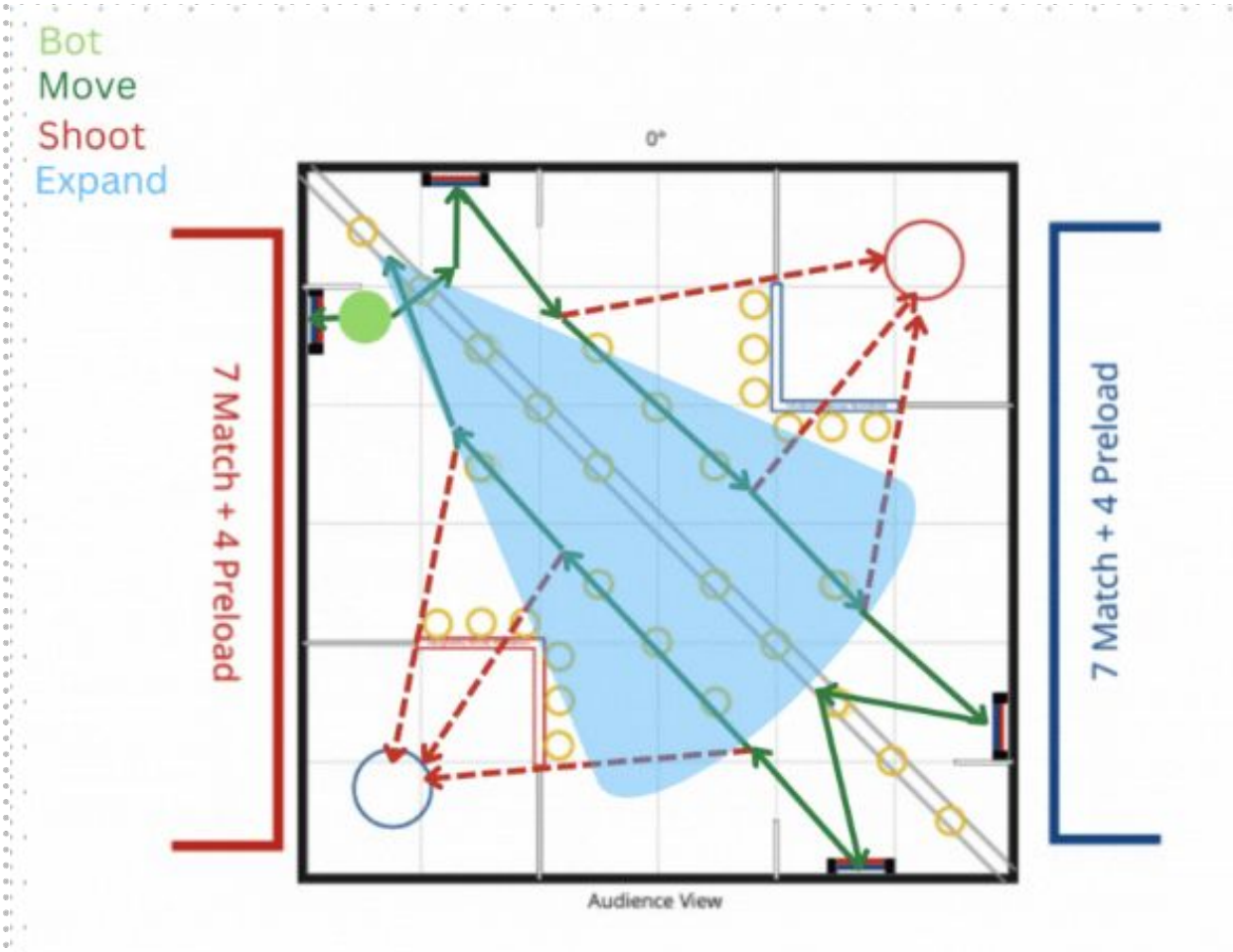
This week we had an incredible experience at the Kalahari Signature event. With the old robot we were able to play defense on top teams, learning commonalities between teams. This will be beneficial towards our future strategic planning. Additionally, we completed our first attempt at our slingshot sled and an intake roller.

Next week, our goal is to build the cocking mechanism, tune the rail system, and if we have time, begin planning programming skills.

Time in Lab:
Monday, Tuesday, Friday, Sunday; 11 Hours

Planning Programming Skills

Given that our robot will be a slingshot, it is most efficient to shoot **3 disks at a time** in order to score as quickly as possible. I optimized our route to ensure that our bot always had **3 disks before shooting**. This route gets the 4 rollers and shoots 6 full cycles, totaling for 130 points without expansion and 214 with perfect I also created pseudocode of the route to make our code easier to implement when the bot is finished.



New Rail System

- We have decided to once again redo our sleds rail/sliding system
  - The previous version which used a modified c-channel still produces too much friction…
  - Therefore we created a new sliding mechanism using spacers
    - Now the sled has 4 spacers acting as wheels on each side. These spacers rotate freely to eliminate friction.
- Additionally we have redesigned the sled to hold the discs perpendicular to the rails rather than parallel

Design of Base Plate

- Additionally, We began the assembly of the baseplate
  - **Due to the complexity and not knowing which cocking mechanism to use we decided to not to design the baseplate and cocking mechanism till now**
  - The baseplate is a custom laser cut piece for the bottom of the robot, designed to effectively use the empty space under the slingshot and intake (the space where motors go on most robots)
  - It has a space to attach c-channels for the cocking mechanism, an inertial sensor and both air tanks

Cocking Mechanism

- To select the appropriate type of cocking and releasing method for our slingshot we examined all 4 types (Slip Gear, PTO style release gear, ‘pinch’ and unwind, and reverse ratchet). Here are their Advantages and Disadvantages:
  - **Slip Gear**
    - The Slip Gear is the most conventional method. It works by sanding off the teeth of a gear and allowing another gear to slip (releasing the slingshot)
    - **Pros:**
      - Simple and Easy
        - When done correctly it is practically impossible for it to fail since it has almost no fail points (no extraneous moving parts)
      - Easily Modifiable
        - Can be done to almost any gear meaning that many different torque combinations can be tested quickly
    - **Cons:**
      - Hard to get right
        - Takes a lot of time and guesswork before the right combination can work
          - For example 12X has spent over 2 week straight trying to get their slip gear to work to no avail
            - **Therefore for the sake of time and to mitigate the risk of a total failure (which would cost weeks of time) we have decided not to use a slip gear**

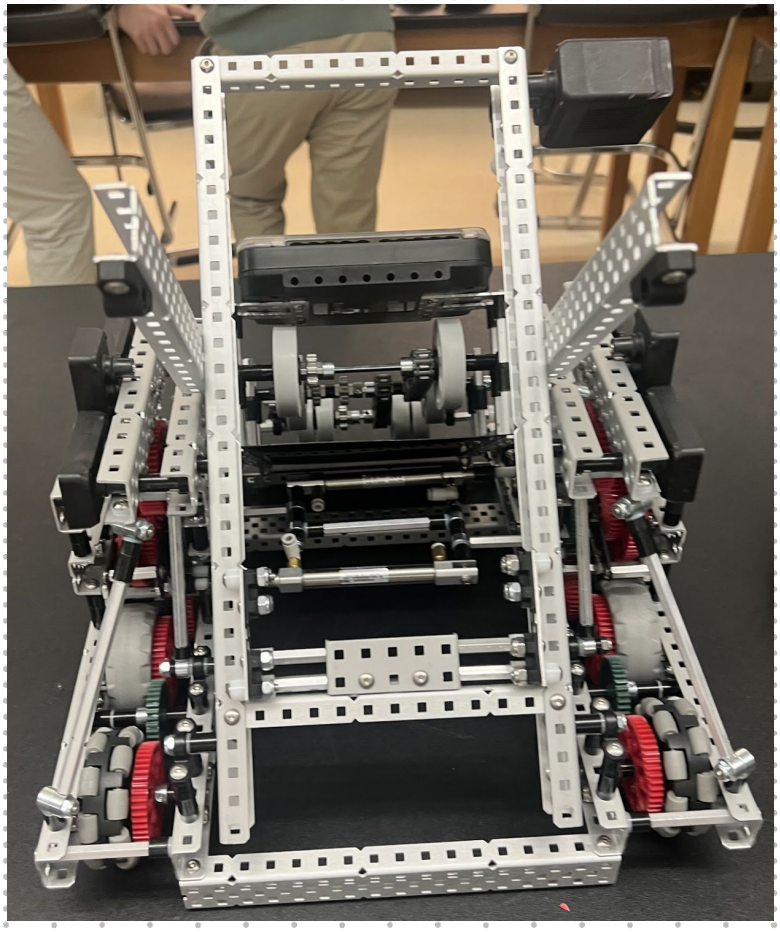
Cocking Mechanism

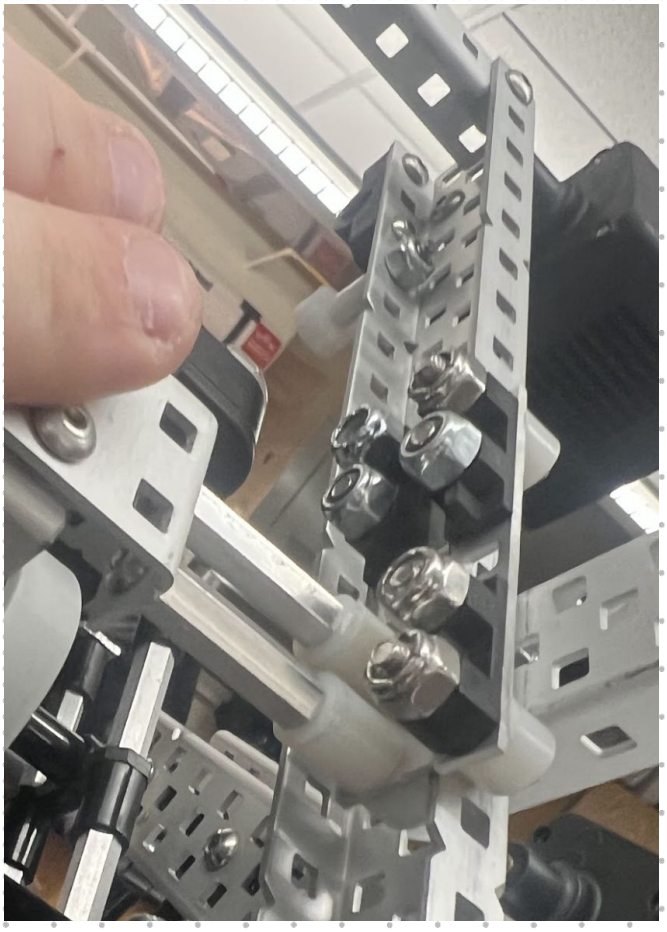
- **PTO style release gear**
  - The PTO style release works by using motors to directly cock back the slide, then holds the slide back using the motors and their resistance. To shoot, a piston removes a gear from the gearbox, like a PTO, removing any motor tension from the cocking mechanism allowing it to shoot
  - **Pros:**
    - Once Designed there is no need to tweak or test it like a slip gear → it should work “out of box”
    - Highly controllable and reliable
  - **Cons:**
    - Requires some planning (not a big deal)
      - Lots of tension and stress on PTO parts which can cause them to warp or break very easily
    - Poses some coding challenges
      - 12H for example tested this type of mechanism and experienced some coding issues that took a long time to correct. Furthermore their custom PTO pieces warped under pressure requiring them to be recreated.
        - **Therefore we will also not be using the PTO style release gear due to time concerns related to the tweaking required to make it work**

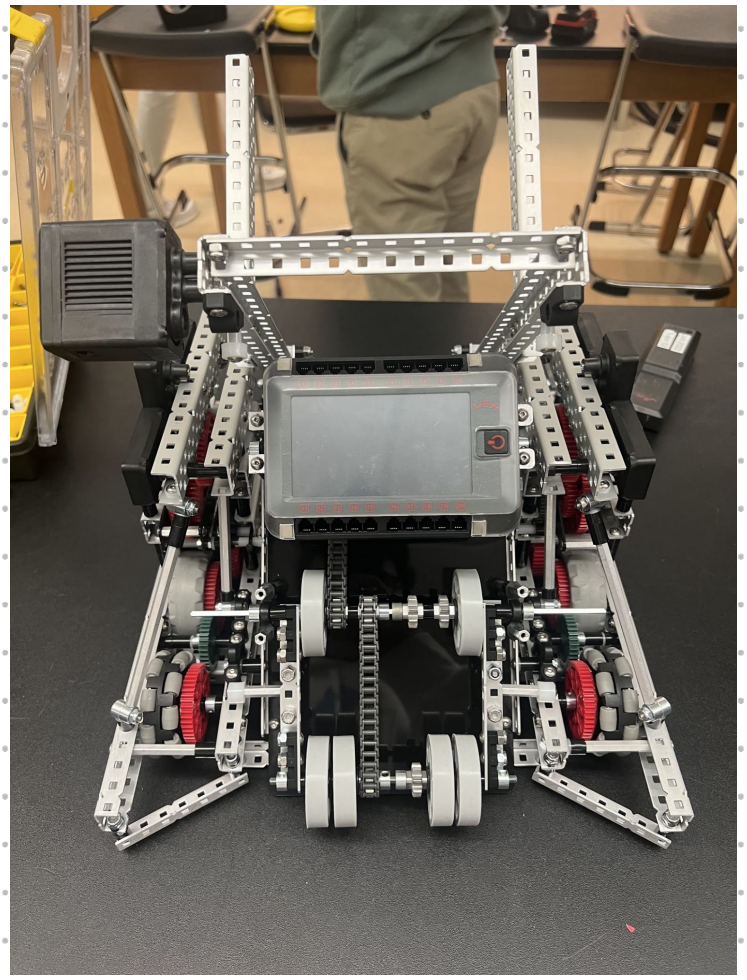
Cocking Mechanism

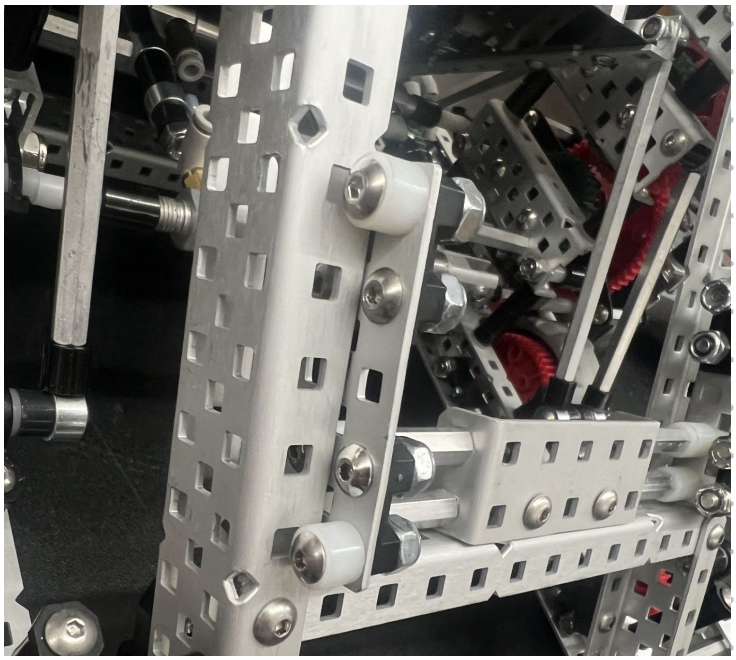
Potential Solutions:‘

- **Pinch’ and unwind**
  - A ‘Pinch’ and unwind system work as described. It ‘pinches’ and holds the sled into place when full cocked back, and then unravels and its winch allowing the sled to freely move forward when unpinched
  - **Pros:**
    - Will not fail…. Like ever
      - Only gears (and a pincher) without any modifications
    - Super easy to set up
      - Requires not tweaking and testing… just simple and easy
  - **Cons:**
    - Slow → requires the winch to completely unwind first
  - **Due to its consistent performance this will likely be our fallback method if the preferred method does not work**
- **Reverse Ratchet**
  - A reverse ratchet works by engaging a ratchet (as in spinning the axle) when cocking the sled back, then pinching it in place. To shoot, the motor will spin in reverse at full speed disengaging the ratchet and allowing the sled to roughly move freely when unpinching
  - **Pros:**
    - Quick
    - Does not require a lot of testing and set up
  - **Cons:**
    - Has not been proven to work yet by a Potomac Team (2145Z thought appears to be using this method upon examination of their robot but it is unconfirmed)
  - **This is our first choice, its quick and effective and should not require much tweaking when it is completed.**









Project Rails/Baseplate

Name Devin Gaines Date 1/24

Team Meeting:
Week In Review Takeaways

This week we continued to make progress on our build. We identified and improved on issues with our rail system, analyzed the best approach to pullback for the slingshot, and began to install the pullback system. All of the progress made this week will likely need tuning due to the complexity of the design. We will continue to examine potential solutions mentioned here.

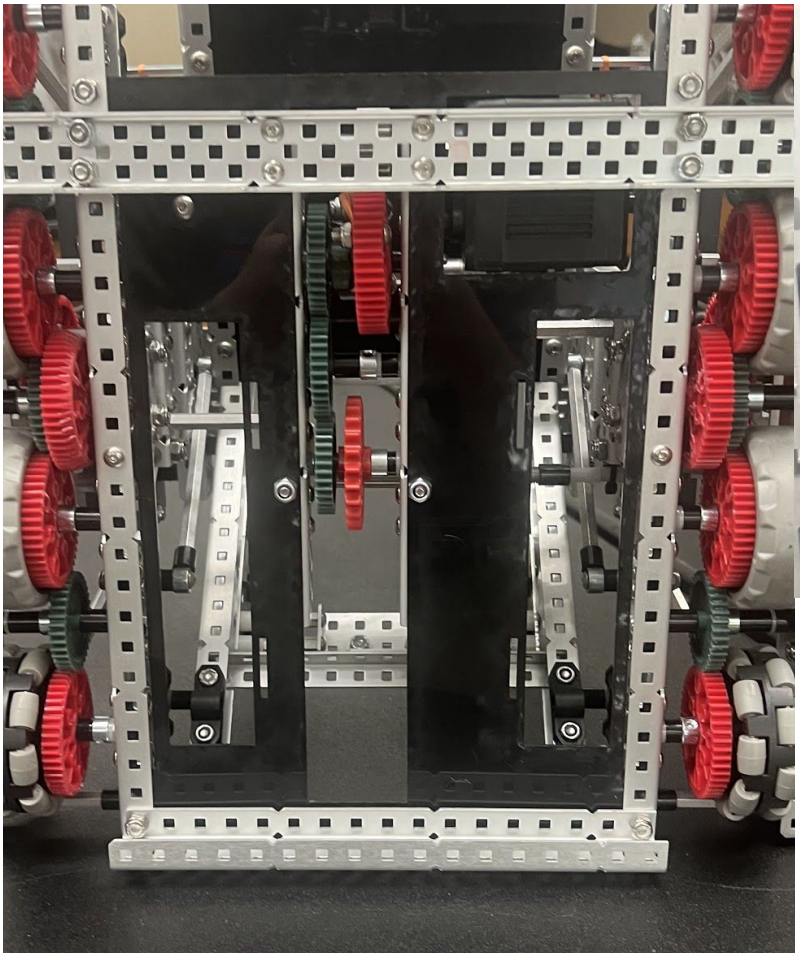
Next week, our goal is to continue optimizing each part, install winch and sled rail, and develop the slingshot.
Time in Lab:
Monday - Friday, Sunday; 18 Hours

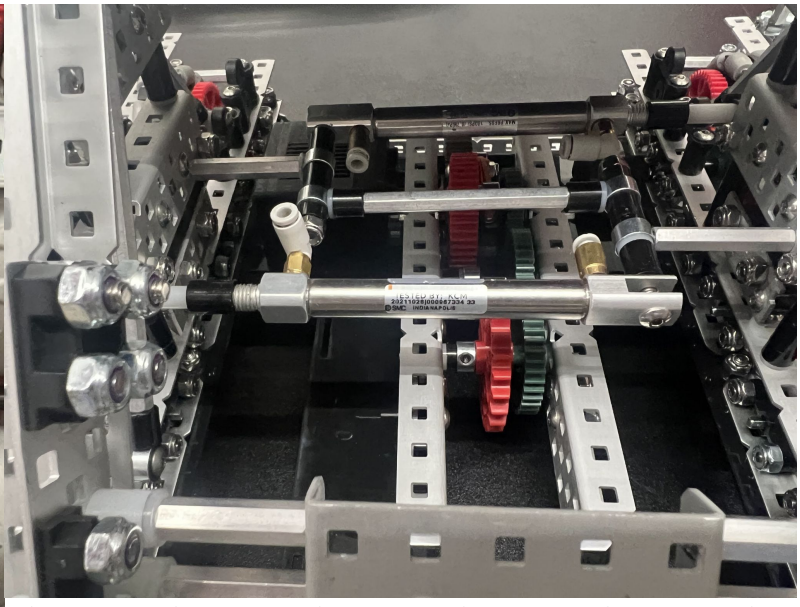
Base Plate and Winch Install

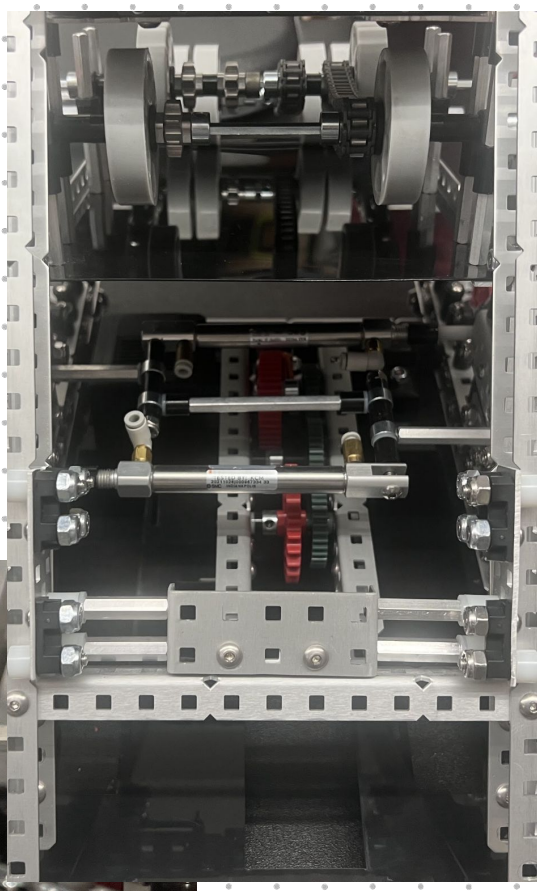
- Today we finished assembling the base plate and cocking mechanism
  - We then installed the base plate on the bottom of the robot
    - **It fit well thanks to the CAD**
- Following that we drilled holes on the “rail” c-channels for high strength axles
  - Once the holes and bearing blocks were installed, we measure out spaces and put a sprocket and the plastic high strength collars
    - The colares are there to hold the string that the Cocking mechanism will use

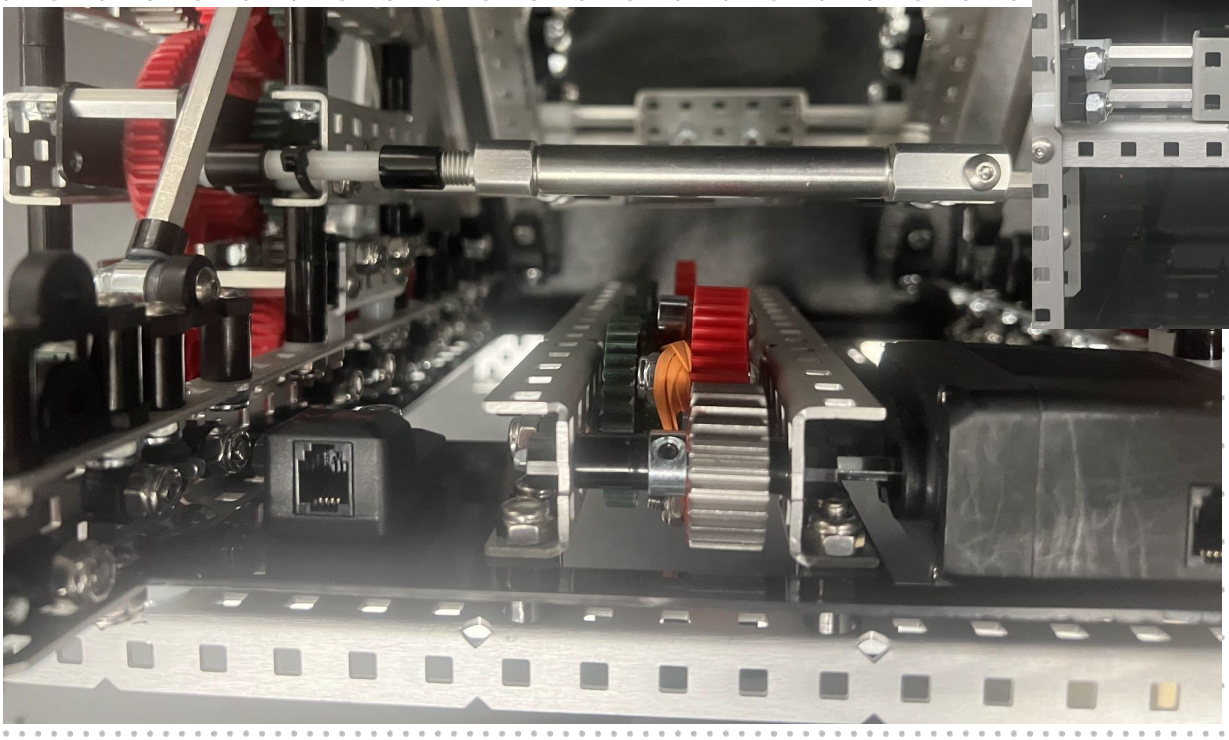
Delrin Rail Installation and Sled Testing

- We also installed custom cut delrin strips on top of the “rail” c-channel
  - These should lower some of the friction and play within the launcher
- Once that was done we installed the back and base of the sled onto the sled assembly
  - We then manually tested the sled with varying rubber band tension









Project Cocking Mech

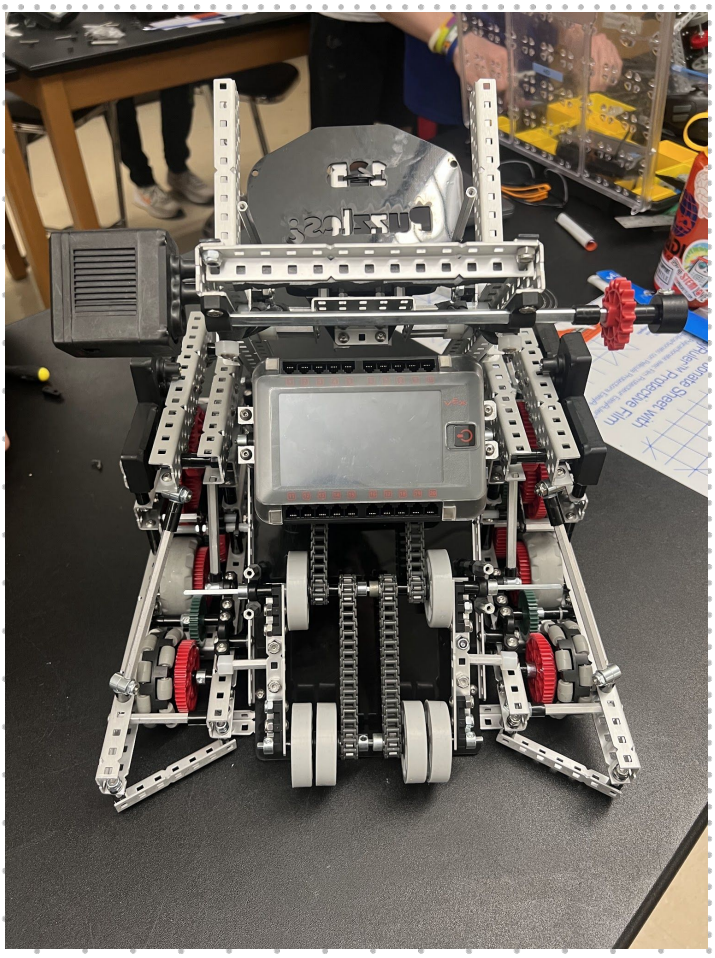
Name Devin Gaines Date 1/29 Page 56

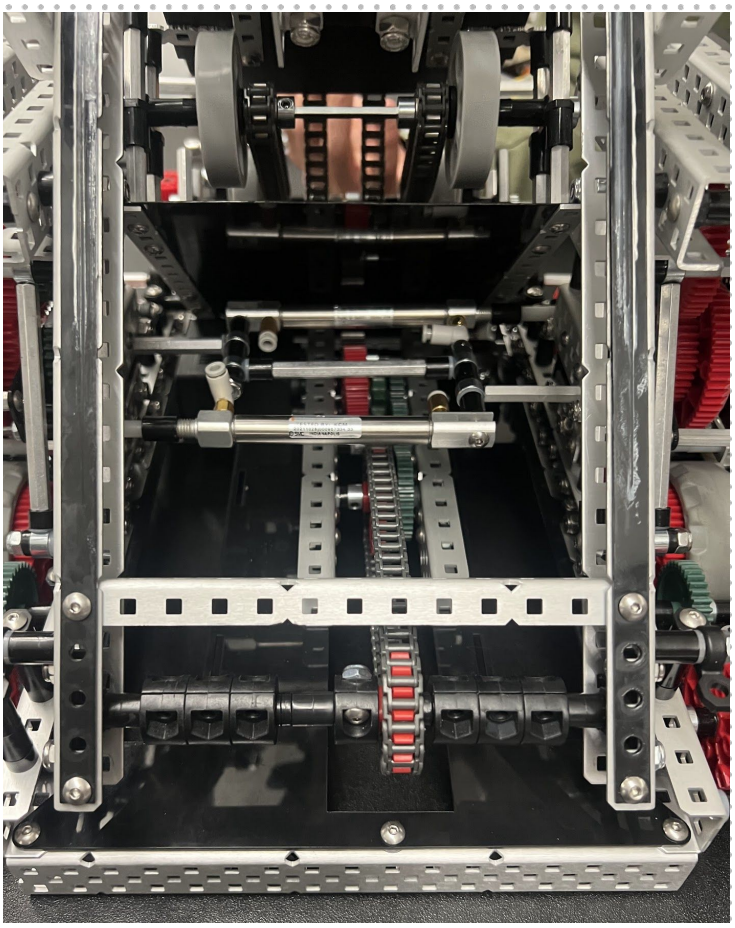
Original Sled Testing

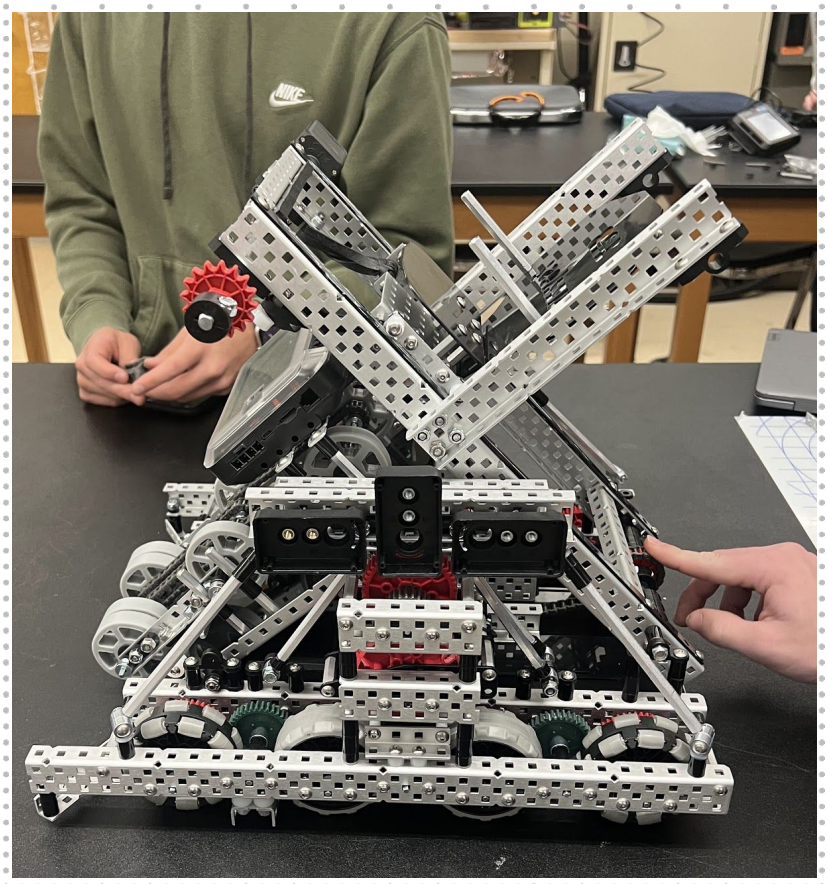
- The first thing we noticed is that the back of the sled was very flimsy made out of delrin
- Despite this flimsiness we decided to test it out anyway
  - When we pulled back the sled we noticed several problems
    - **First**, The grouping was horrible and the discs flew vertically because the back was not rigid
    - **Second,** The base c-channel of the sled kpt on getting loose and rocking back and forth

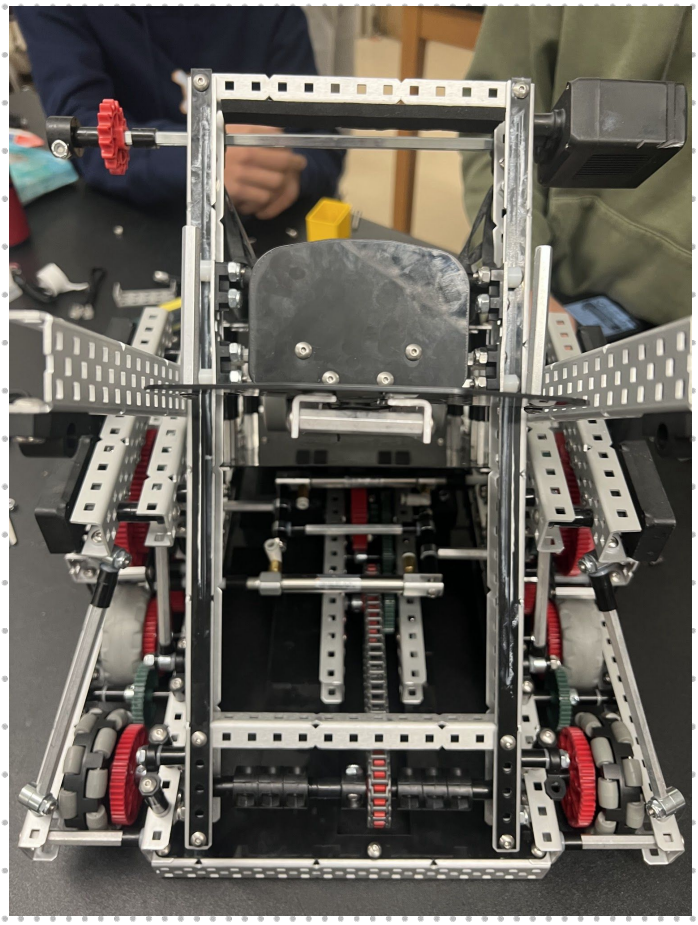
Sled Modifications

- To fix the first problem we tested a few different ways of making the back of the sled more rigid
  - The first thing we did is put some bar pieces on the back to make it more rigid
    - This helped with grouping and trajectory but still wasn't good enough
  - The second thing we did was install some more bar pieces to make it more rigid
    - This also helped but wasn't enough
  - The final thing we did was install some angle pieces to make it more rigid
    - **This worked**
- To fix the rocking we decided to expand the 2 hole long 5 wide c-channel into a 5 long 5 wide channel
  - Then we installed some standoffs on the other end of the c -channel to stop the rocking
    - **This Worked**



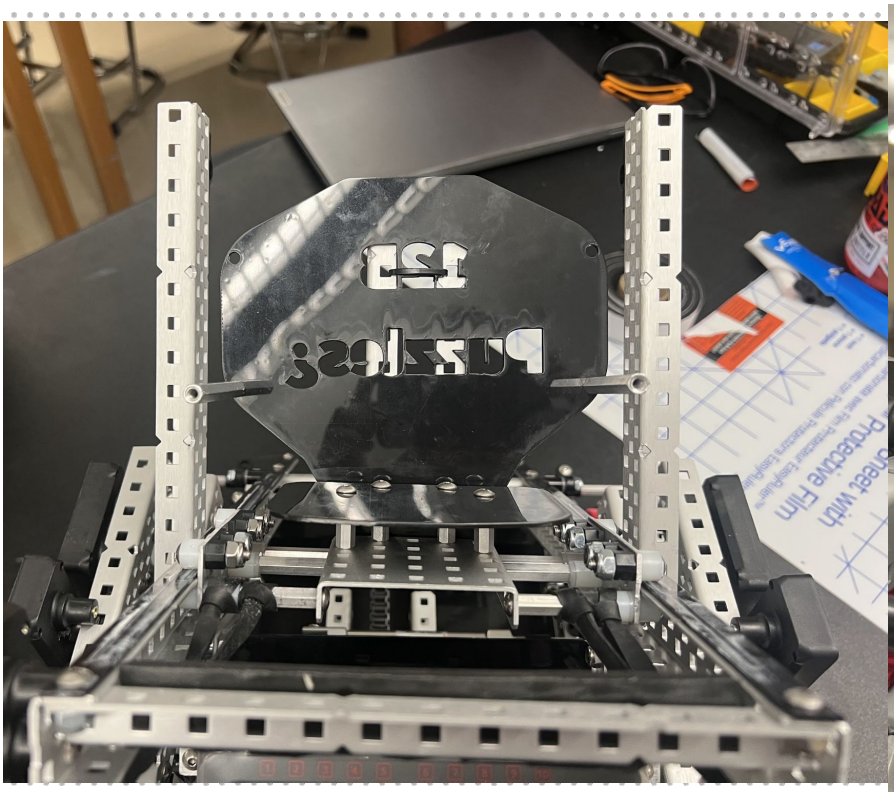


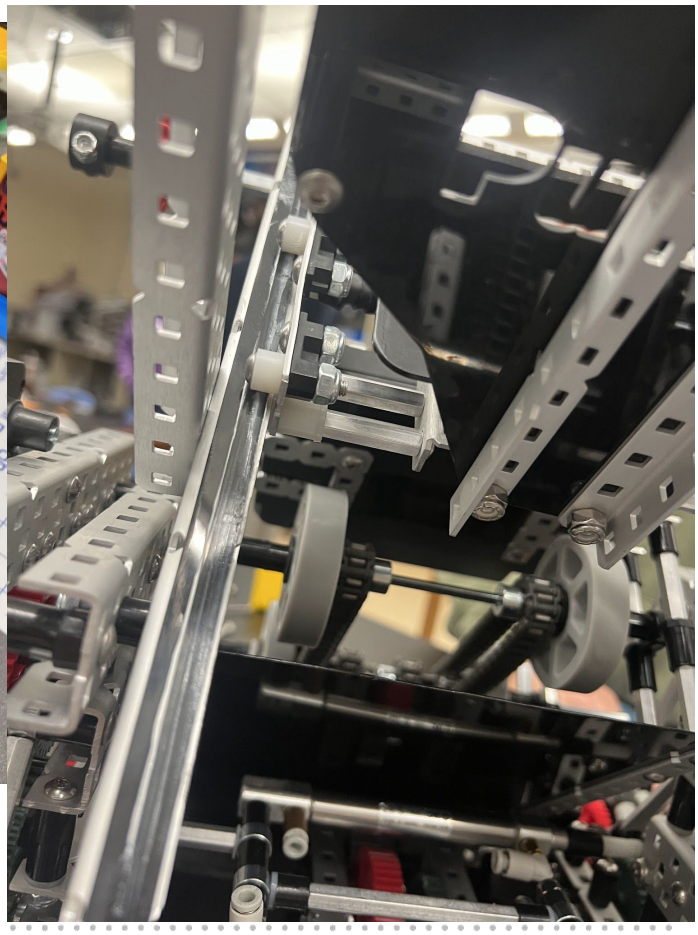




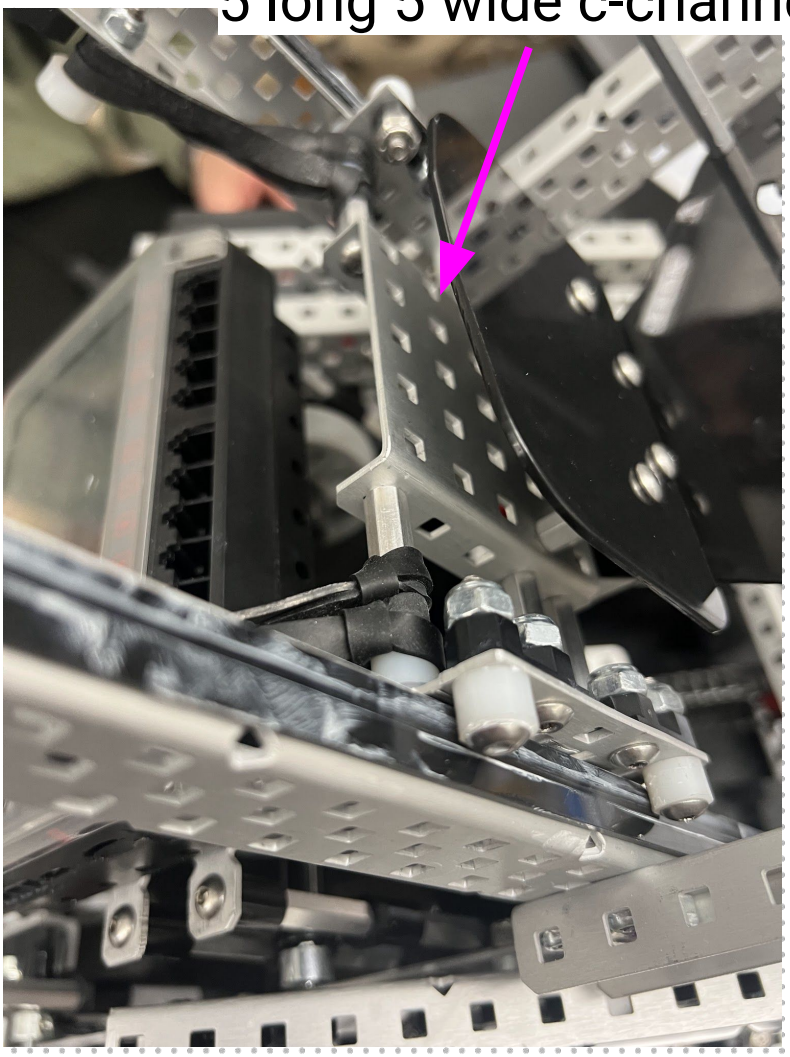
Project Sled Mods

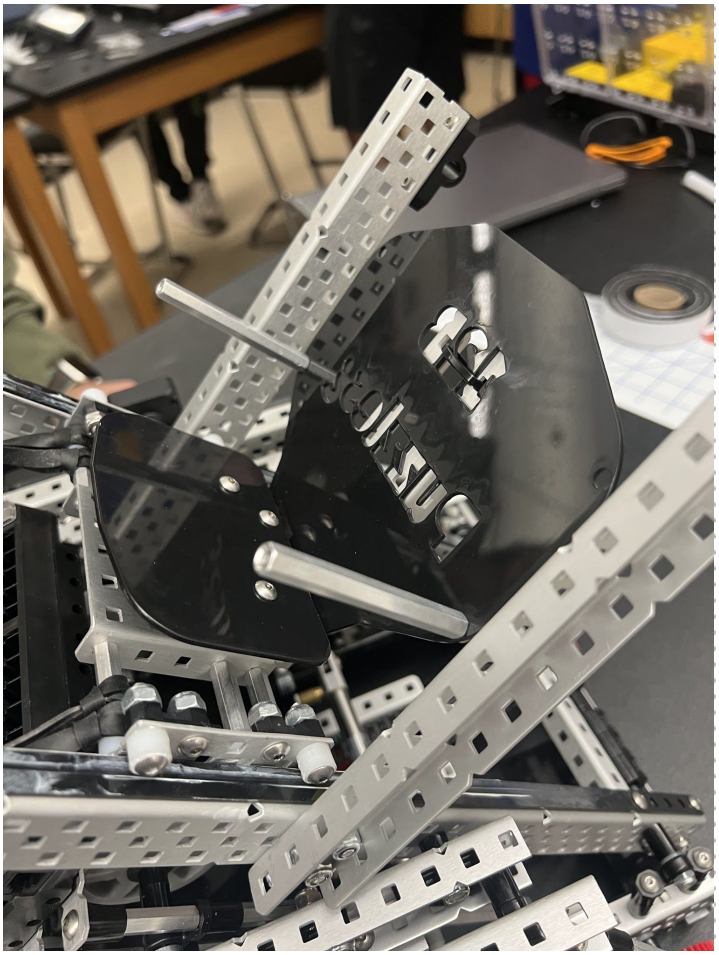
Name Ben Date 1/29 Page 58





5 long 5 wide c-channel





Project Sled Mods

Name David Gardner Date 1/29 Page 59

To Do List:

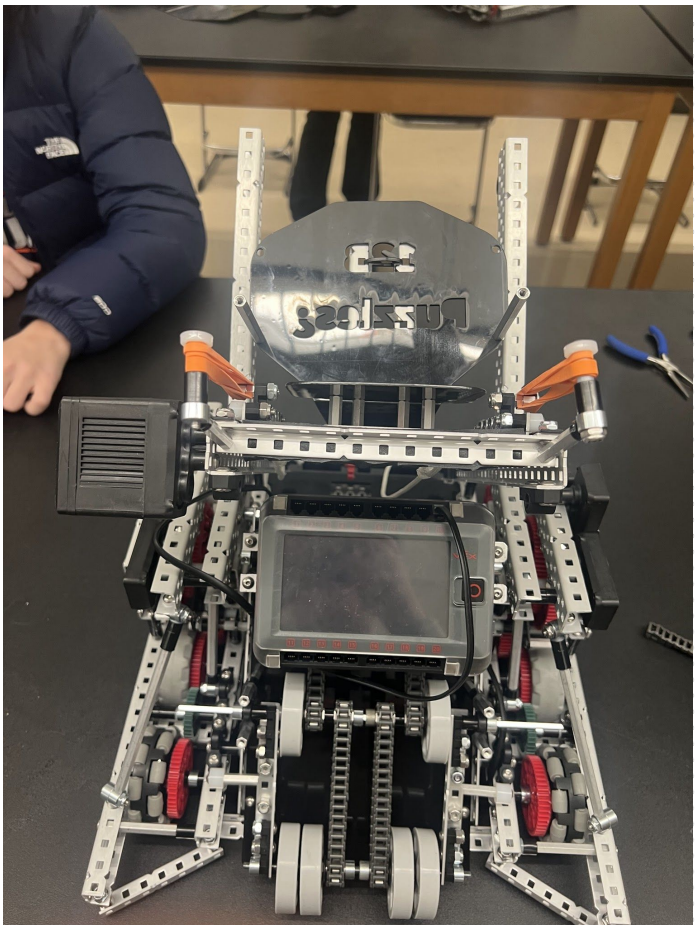
1. Finish Winch
    a. Test with different rubber band strengths
    b. Change rubber band placement
    c. Replace backstop with spacers to fix string clearance
    d. Install “back shield”
    e. Add locking mechanism
2. Extend and Replace Roller Mechanism and Link to intake
3. Install gears for drive base Motors
    a. Install motor shield
4. Wire Up Motors and Pneumatics
    a. Install air tanks
        i. Test for leaks
5. Endgame Launchers
6. Install Decorative Delrin
7. Code driver program
8. Code Autons
    a. Skills
    b. Right Side

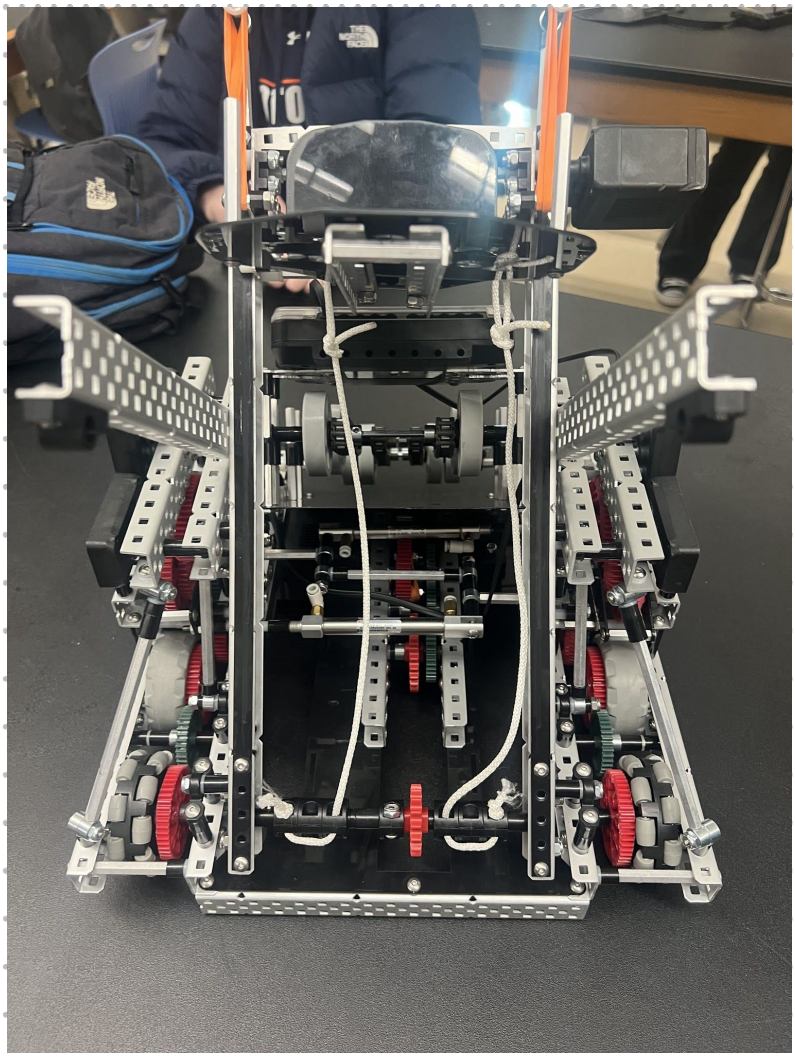
Finishing Up the Slingshot

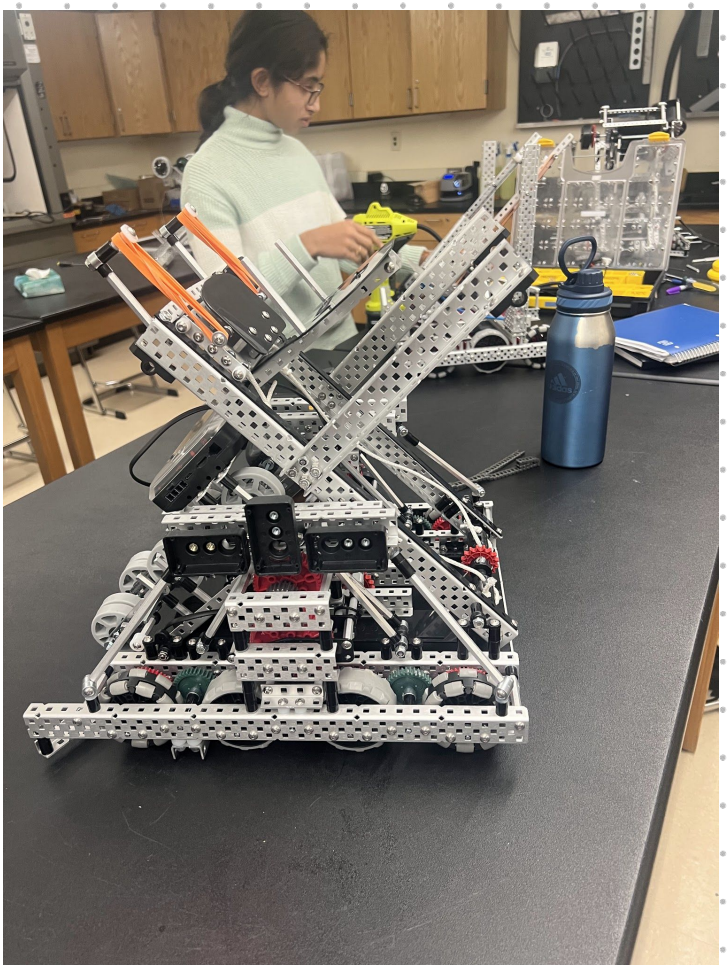
- Today we finished the prototype slingshot **its ready for powered testing,** In order finish the slingshot prototype we completed the following tasks:
    1. We changed the rubber band placement both on the sled and the structure of the slingshot itself
        - Currently the bands were falling off the slingshot everytime we released the sled and they were in the way of the intake, **Therefore we moved the rubber bands to be on top of the rails as to not block the intake and extended where they connected to the robot to stop them from falling off**
    2. Additionally we replaced the rear bar with spacers to stop the sled because the bar would have gotten in the way of the string
    3. We installed both strings that the cocking mechanism will use, (These will have to be optimized to ensure that they both pull back straight)
    4. We Changed the lengths of the roller mech arms to make sure that they do not collide with discs (went from 15 to 18 holes long)
    5. Finally we roughly selected the correct amount of rubber bands for our sling to test
    6. Additionally we decided not to continue with a back shield as a large wedge would be enough protection for the sled

To Do List Update:

1. Finish Winch
    a. ~~Test with different rubber band strengths~~
    b. ~~Change rubber band placement~~
    c. ~~Replace backstop with spacers to fix string clearance~~
    d. ~~Install “back shield”~~
    e. Add locking mechanism
2. ~~Extend and Replace Roller Mechanism~~ and Link to intake
3. Install gears for drive base Motors
    a. Install motor shield
4. Wire Up Motors and Pneumatics
    a. Install air tanks
        i. Test for leaks
5. Endgame Launchers
6. Install Decorative Delrin
7. Code driver program
8. Code Autons
    a. Skills
    b. Right Side







Project Finishing Sling

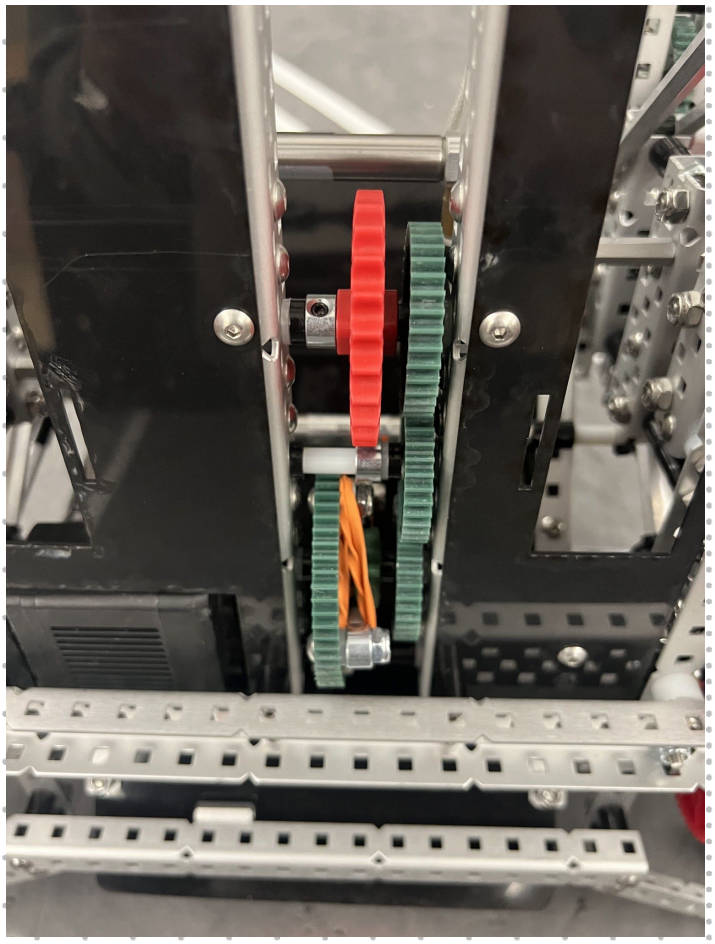
Name Arav Bhargava Date 1/31 Page 63

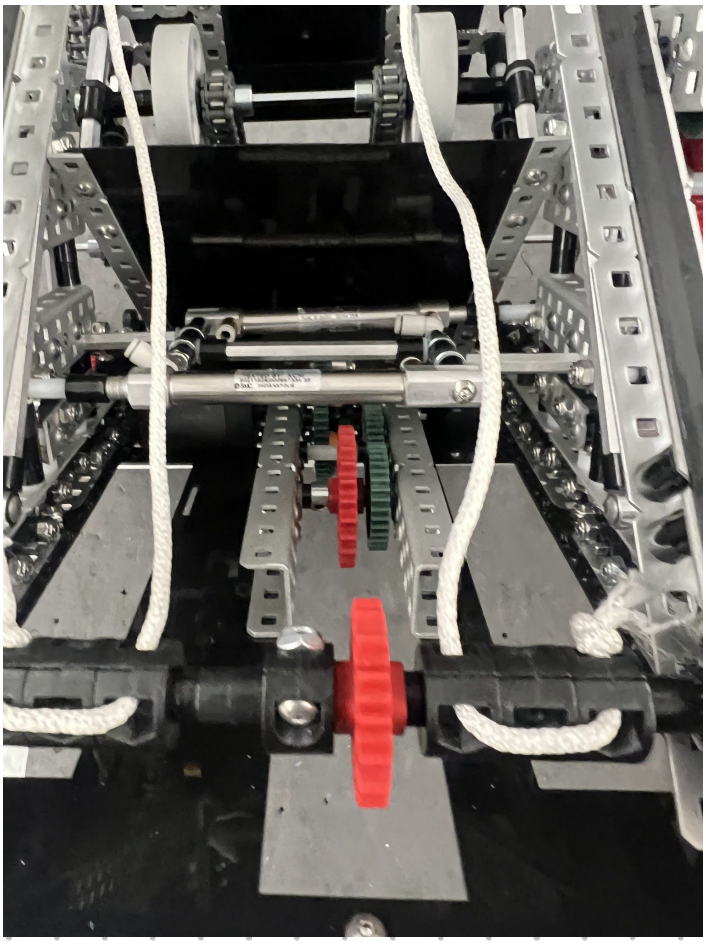
February

1	Rebuilding Cocking Ratchet
2	Installing Motor Gears and intake testing
3	Intake and Slingshot Modification
4	N/A
5	Locking Mech
6	Drive Base Testing
7	Transmission Testing
8	N/A
9	Transmission Repairs
10	N/A
11	N/A
12	N/A
13	Slingshot Testing
14	Drive Base Testing
15	Fixing Drive Base
16	Norcal Prep
17	Flight Repairs
18	Norcal Sig
19	Norcal Sig
20	N/A
21	Rebuilding Claw
22	Rebuilding Sled
23	Rebuilding Sled
24	Code
25	Skills Auton
26	Intake Inconsistency
27	Intake Mods
28	Endgame + Auton
29	N/A
30	N/A
31	N/A

Rebuilding the Cocking Ratchet

- Today we started testing the slingshot with a powered motor → **this revealed that the low profile ratchet that we had created using a 48 tooth gear was slipping and was not strong enough to cock back the slingshot**
  - Therefore we replaced the 24 and 48 tooth gears with a 12 and a 60 to allow for the creation of a double ratchet
    - Not only will this ratchet be stronger, it will also use a low profile gear allowing for more clearance and a more consistent mechanism



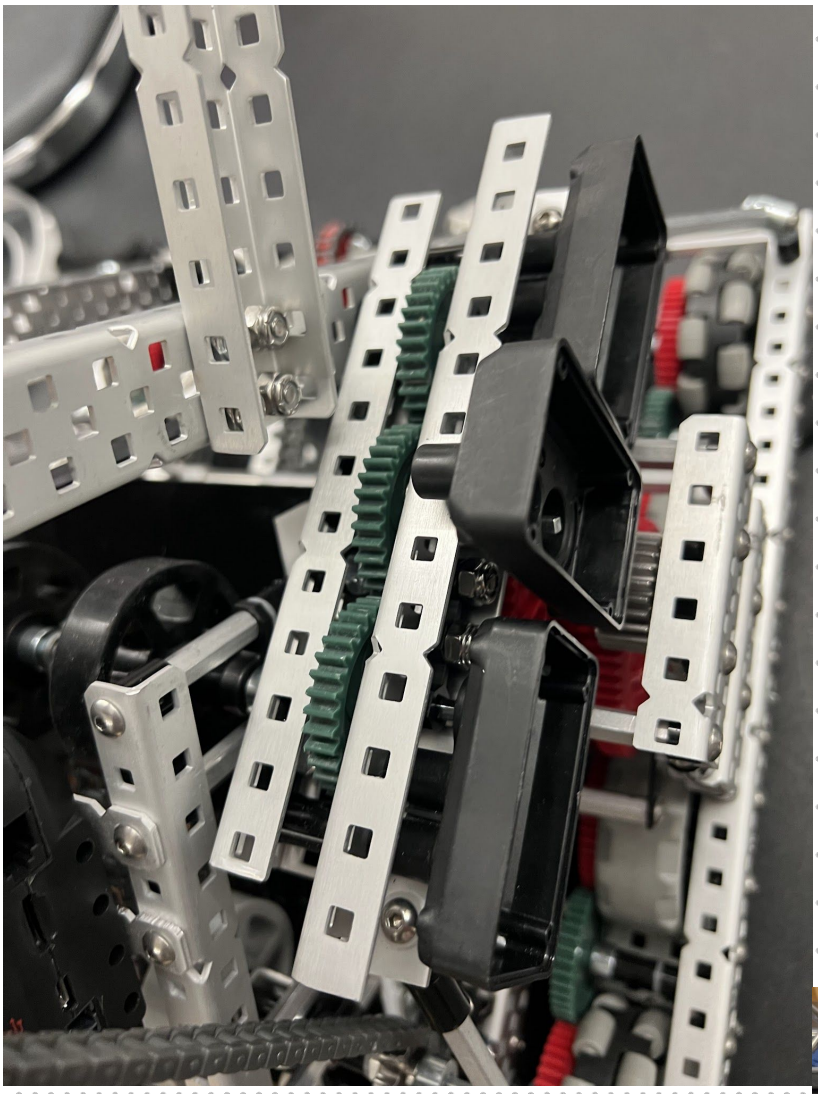


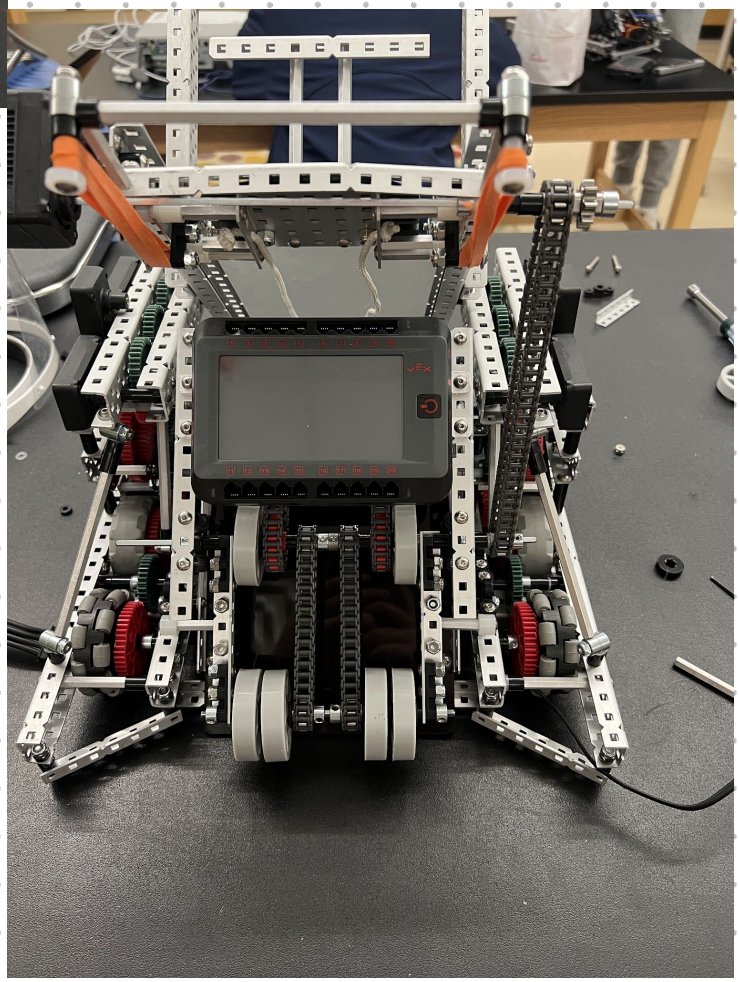
Installing motor gears

- Today we began the installation of the motor gears in order to begin testing driving
    - Due to the tight spacing and clearance the installation of the gears was harder than expected
        - To install the gears we decided to remove one c channel put an axle with the right spacings onto all the bearing blocks and then put back the c channel... Sandwiching close the housing.

Connecting Intake/ Preliminary testing

- Additionally we chained the intake onto the intake/roller motor and began testing
    - What we noticed is that while the disk was clearing the intake it would not go travel any distance once it left, meaning that it was not falling into the sled correctly.
        - **We theorized that this could be due to multiple reasons, these include:**
            - Hitting the standoff on the intake where the wheels are mounted on
            - Incorrect spacing for the intake wheels
            - Excessive friction
- **Due to these complications we have decided to start testing the intake**

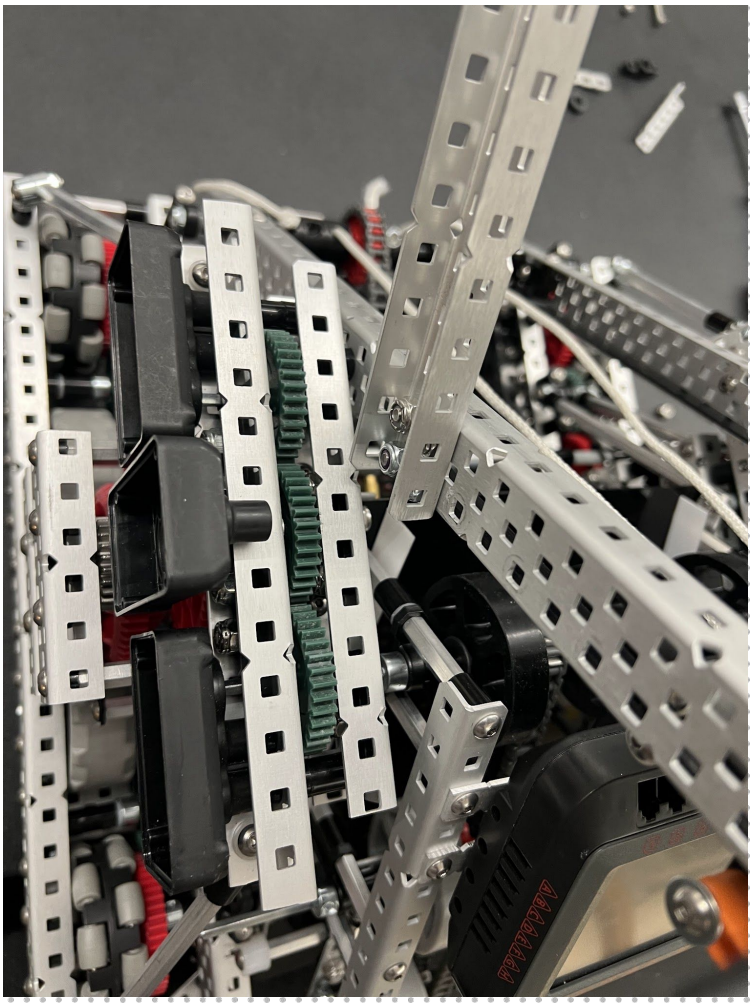


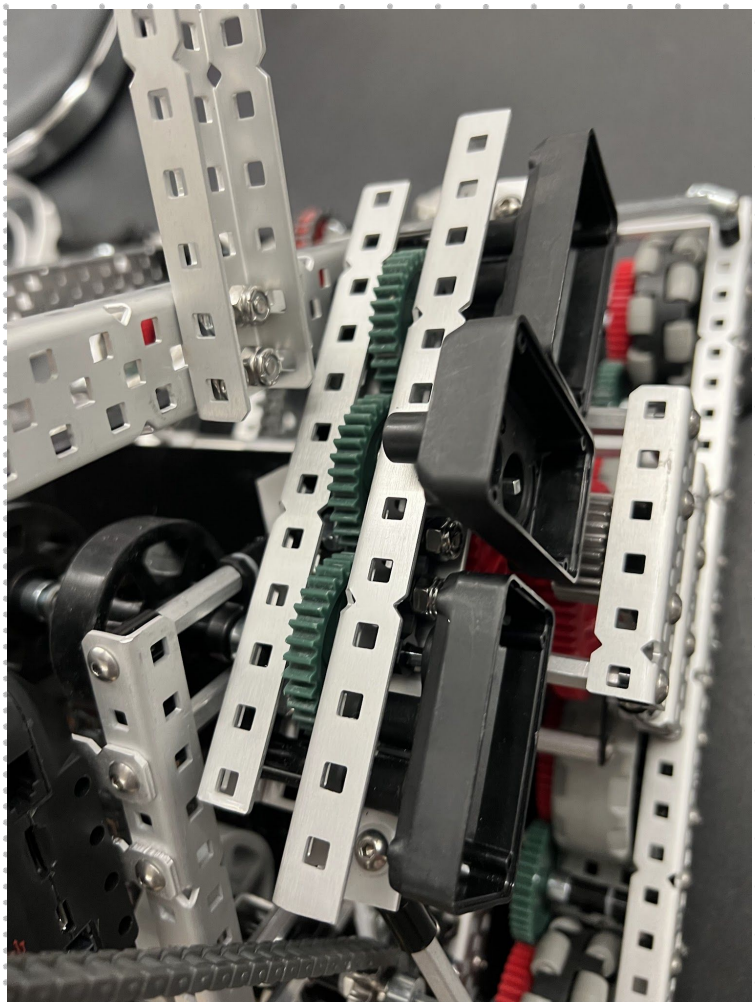


Project Motor Gears

Name Arav Bhargava

Date 2/2





Project Installing Motor Gears

Name Daniel Overdeck

Date 2/2

Creating Driver Program + Transmission

- For the driver we copied several concepts from our previous code such as the deadband and the intake, however much of it was new for this robot.

- This meant we had to create new codes for toggling the transmission solenoids as well as for changing the direction of the robot and the pullback macro for the robot

- It also meant changing the ports and adding new motors/solenoids onto the new bot and making sure the code matched where they were actually plugged into the brain

- We had to add two new motors onto the drivebase portion of the robot so we had to group 3 motors together instead of just two in our old robot.

- This left the remaining two motors to be for intake and pulling the slingshot back.

```
brain Brain;
motor FlywheelMotorA = motor(PORT17, ratio18_1, true);
controller Controller1 = controller(primary);
motor leftMotorA = motor(PORT18, ratio6_1, true);
motor leftMotorB = motor(PORT19, ratio6_1, false);
motor leftMotorC = motor(PORT20, ratio6_1, true);
motor_group LeftDriveSmart = motor_group(leftMotorA, leftMotorB, leftMotorC);
motor rightMotorA = motor(PORT11, ratio6_1, false);
motor rightMotorB = motor(PORT12, ratio6_1, true);
motor rightMotorC = motor(PORT13,  ratio6_1, false);
motor_group RightDriveSmart = motor_group(rightMotorA, rightMotorB, rightMotorC);
limit BumperA = limit(Brain.ThreeWirePort.H);
motor Intake = motor(PORT1, ratio6_1, false);
bool intakeToggle = false;
int counter=0;
bool reverseIntakeToggle= false;
inertial imu= inertial(PORT7);
smartdrive Drivetrain = smartdrive(LeftDriveSmart, RightDriveSmart, imu,  319.19, 295, 40
```

```
extern smartdrive Drivetrain;
extern brain Brain;
extern motor FlywheelMotorA;
extern controller Controller1;
extern motor leftMotorA;
extern motor leftMotorB;
extern motor leftMotorC;
extern motor_group LeftDriveSmart;
extern motor rightMotorA;
extern motor rightMotorB;
extern motor rightMotorC;
extern motor_group RightDriveSmart;
extern bool flywheelToggle;
extern bool button_toggle;
extern bool reverseIntakeToggle;
extern limit BumperA;
extern bool intakeToggle;
extern int counter;
```

Creating Driver Program + Transmission

-Because we were using C++ we had to write several initializations for different pieces whether it be solenoids, motors, sensors, booleans or anything else.

-For the motors we had to get the exact ratio for it depending on if it was a blue, green or red motor (most of our motors are blue except for the slingshot motor which is green for more torque)

-We placed the ports of our drivebase directly in the groups they were, we also had to change the direction of the middle and side motors so that they wouldn't go in the wrong direction and prevent the drivebase from moving by working against the other motors or cause the drivbebase to move in the wrong direction

-Another key part of the driver program was the methods needed to change boolean values and example being shown below.

```
void toggle_flywheel() { flywheelToggle = !flywheelToggle; }
void toggle_flywheel1() { flywheelReverse=!flywheelReverse; }
void buttonToggle() { button_toggle = !button_toggle; }
void toggle_intake() { intakeToggle = !intakeToggle; }
void toggle_reverse_intake() { reverseIntakeToggle = !reverseIntakeToggle; }
void rightToggle() { right_toggle = !right_toggle; }
```

Transmission Code

```
void rightToggle() { right_toggle = !right_toggle; }
```

This line initializes our transmission function which flips the boolean **right_toggle.**

```
if (right_toggle){
 solenoid2.set(0);
  solenoid3.set(0);
}
else {
   solenoid2.set(1);
  solenoid3.set(1);
}
```

This code checks the status of the boolean **right_toggle**. If **right_toggle** is true, then the solenoids are activated, changing the gear ratio. If **right_toggle** is false, the solenoids are pushed back in, returning the gear ratio back to normal.

```
Controller1.ButtonR1.pressed(rightToggle);
```

This line of code sets our transmission button to R1

Deadband For Bot Pictures+Speed Button

```
while (1) {

    int rightdrivespeed = Controller1.Axis2.value();
  int leftdrivespeed = Controller1.Axis3.value();
  int deadband = 5;
  //THE LINE
  if (button_toggle)
  {
  if (abs(rightdrivespeed) > deadband) // If the left joystick is greater than
                                       // or less than the threshold:
  {
    rightdrivespeed =Controller1.Axis2.value()*7/10; // Left Joystick Y value / 2.
  } else // If the left joystick is within the threshold:
  {
    // Stop the left motor (cancel noise)
    rightdrivespeed = 0;
  }

  if (abs(leftdrivespeed) > deadband) // If the right joystick is greater than
                                      // or less than the threshold:
if (abs(rightdrivespeed) > deadband) // If the left joystick
{
  rightdrivespeed =Controller1.Axis2.value(); // Left Joysti
} else // If the left joystick is within the threshold:
{
  // Stop the left motor (cancel noise)
  rightdrivespeed = 0;
}

if (abs(leftdrivespeed) > deadband) // If the right joystick
                                    // or less than the thre
{
  leftdrivespeed = Controller1.Axis3.value(); // Right Joyst
} else // If the right joystick is within the threshold:
{
  leftdrivespeed = 0;
}
}
LeftDriveSmart.spin(reverse, leftdrivespeed, percent);
RightDriveSmart.spin(reverse, rightdrivespeed, percent);
```

Deadband For Bot Pictures+Speed Button

-Calculates the given axis on the controller, does not run unless past a certain point (in this case 5)

-Reduces Controller Sensitivity of the robot so it doesn't move with slightest controller movement.

-Speed button adjusts speed when clicked.
-Hooked up to a void statement that adjusts a boolean and is activated when a button is pressed.

-Goes from 100 percent speed to 70 percent speed

-Useful for old robot as it would commonly overheat the motors, running them at lower speeds would make it so it didn't heat up as fast.

To Do List Update:

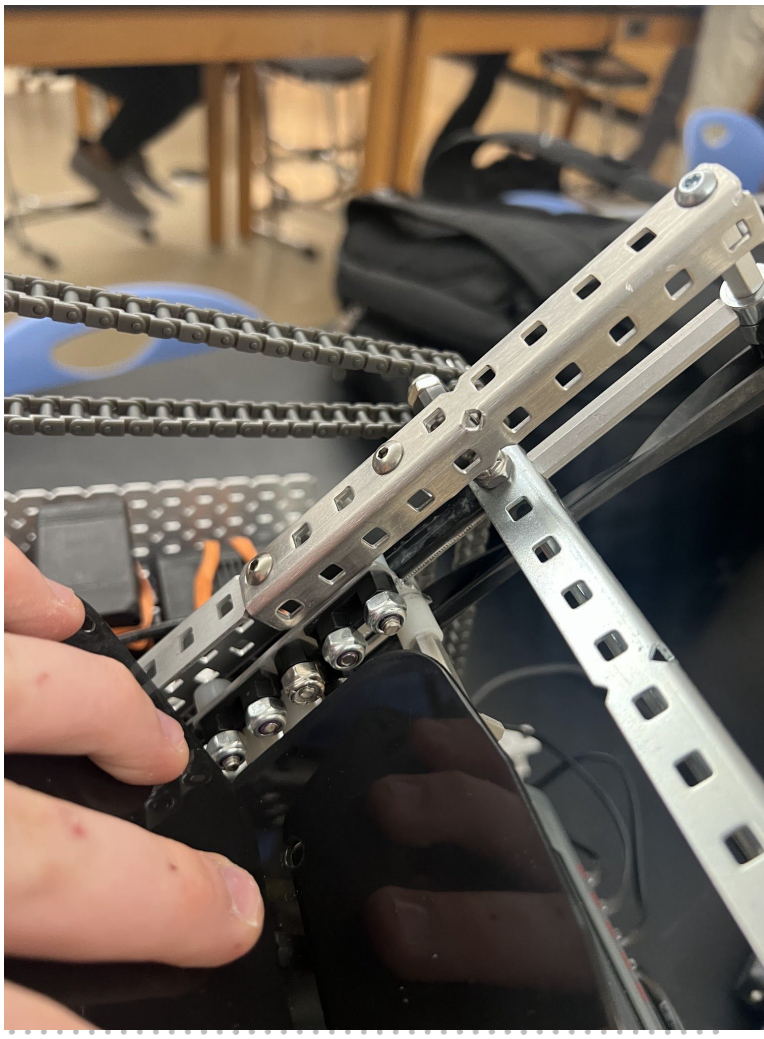
1. Finish Winch
    a. ~~Test with different rubber band strengths~~
    b. ~~Change rubber band placement~~
    c. ~~Replace backstop with spacers to fix string clearance~~
    d. ~~Install “back shield”~~
    e. Add locking mechanism
2. ~~Extend and Replace Roller Mechanism and Link to intake~~
3. ~~Install gears for drive base Motors~~
    a. Install motor shield
4. Wire Up Motors and Pneumatics
    a. Install air tanks
        i. Test for leaks
5. Endgame Launchers
6. Install Decorative Delrin
7. ~~Code driver program~~
8. Code Autons
    a. Skills
    b. Right Side

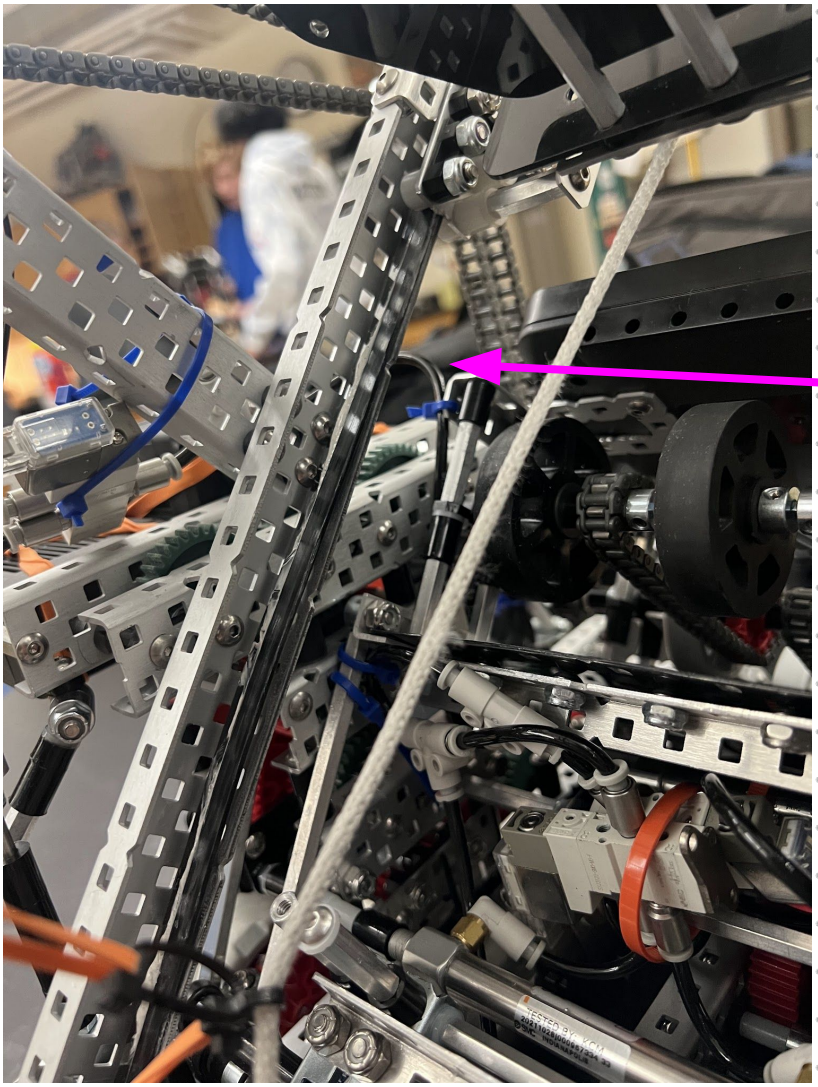
Slingshot Modification

- Continuing yesterday's testing, we have decided to lower the sled in order to make the job of the intake easier
  - To do this we moved the sled to slide on the bottom lip of the rail c-channel
    - This required iu to move the delrin rail piece as well as flipping the rubber band assembly upside down to to accommodate the new location
    - Additionally this required us change the high strength axle that connects to the intake and roller mech into a low strength. Hopefully it will not bend too much.
      - If it does bend we will have to somehow attach a high strength bearing without screwing it to the rail part
- At the same time we tested the new slingshot ratchet.
  - **It snapped in half due to the stress. Therefore we will have to use the pinch and unwind method**

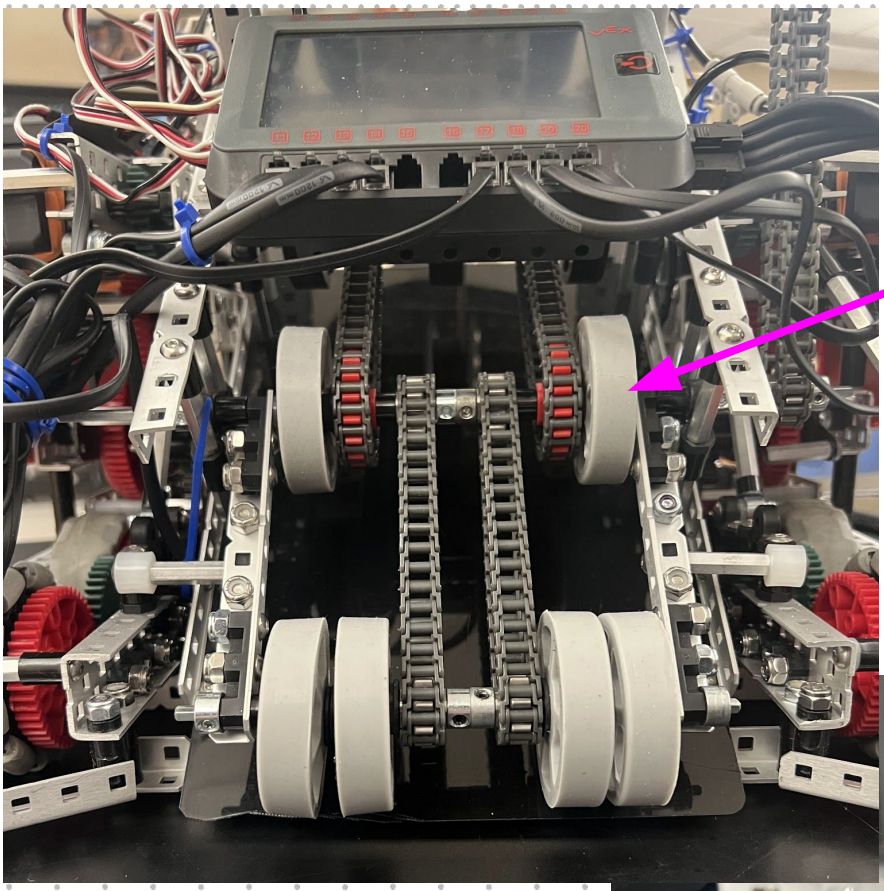
Intake Modifications

- To say that we thoroughly tested the intake today would be an understatement
  - We changed the pressure/compression of the wheels, the type of wheels, number of wheel, and the speed of the wheels before we found a combo that was able to launch the discs far enough
    - **The combination we landed on was 3 black flex wheels rotating at 1200 rpm with minimal compression for the last row. Now it works**

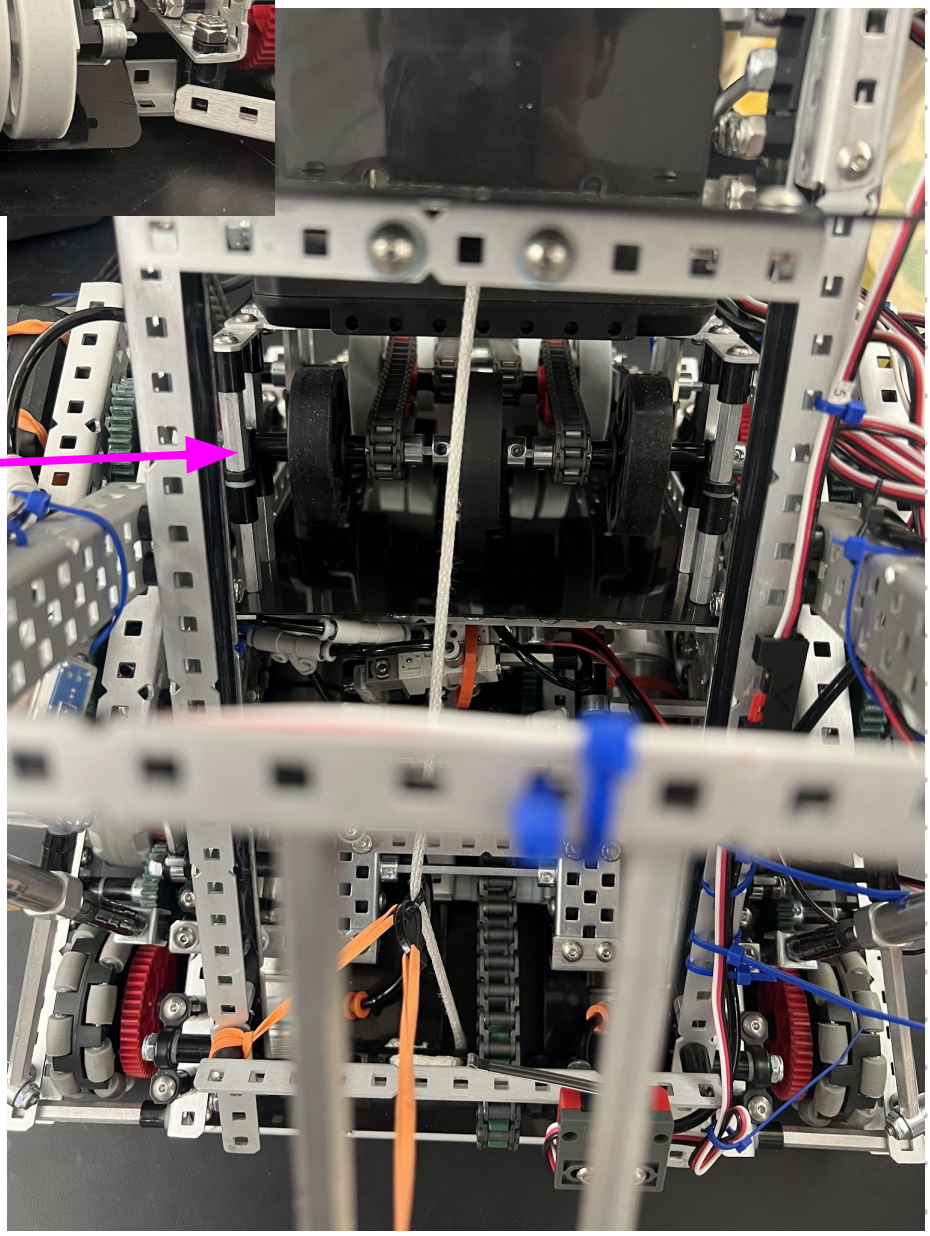




Lower Position



2 x 1 gear ratio (600 rpm for grey wheels 1200 for black)



1200 RPM black wheels

Black Wheels == Firmer Better transferring speed when optimized

Team Meeting:
Week In Review Takeaways

This week we worked on several programming initiatives, the intake gear ratio (which will be crucial to ensuring the disks are positioned correctly in the slingshot), and several slingshot modifications. We are starting to run into more issues; however, we are addressing them in a very analytical manner which is helping us make the most effective decisions for solutions.

Next week, we will focus solely on fixing the transmission and ensuring the robot drives properly in both power and speed modes.

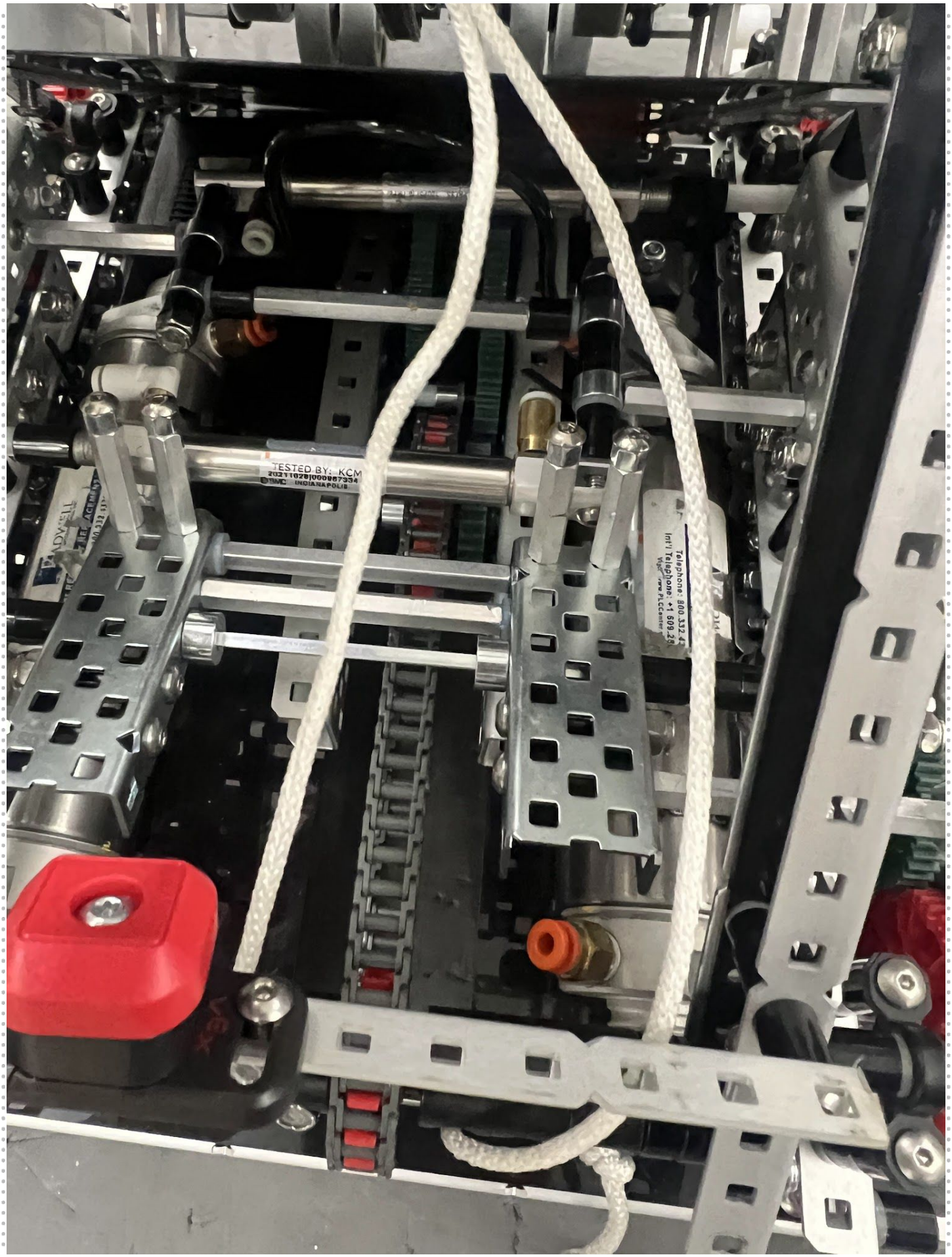
Time in Lab:
Tuesday - Friday, Sunday; 16 Hours

Drive Base Gear "Fiasco"

- Today we set up the robot for testing. Unfortunately we were unable to drive in "fast" mode.
  - Therefore we began troubleshooting the problem
    - What we tried today was to swap the big gears out with ones that were beveled better because we thought they might be interfacing with its complementary gear when not in that specific mode
      - This did not fix the problem
    - We continued troubleshoot the problem and found that one of the drive base gears was damaged

Creation of Locking Mechanism

- Additionally we constructed the locking mechanism for the slingshot to allow for it to unwind
  - Using steel c channel, some standoff and an axle we created a joint below the rails and placed it in an optimal position and to use space effectively
    - Additionally we mounted some double acting pneumatics on the outside of the rails to control the mechanism
      - To lock back the sled the pneumatics extend, extending standoffs into the struts of the sled



Project Locking Mech

Name Arav Bhargava Date 2/5/2023 Page 81

To Do List Update:

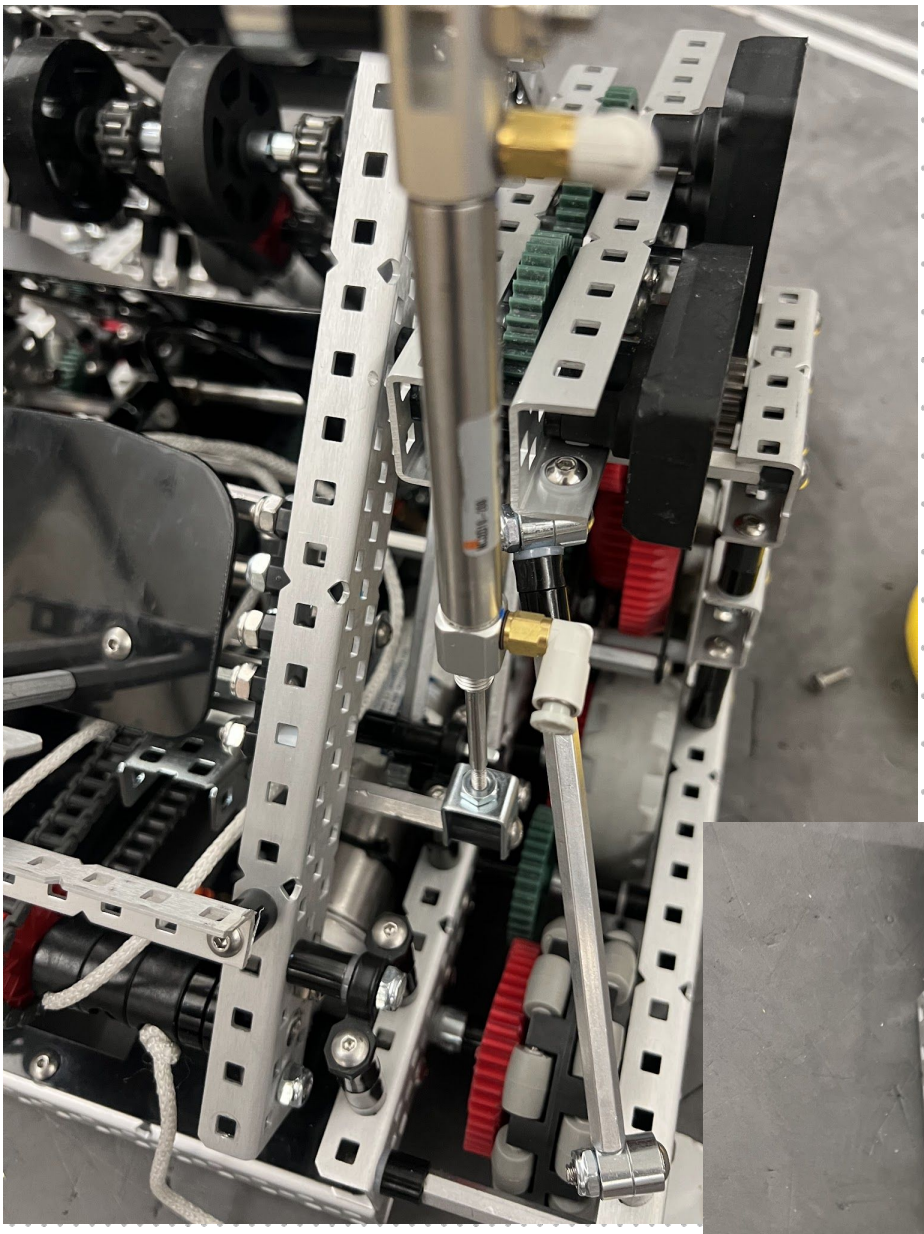
1. ~~Finish Winch~~
    a. ~~Test with different rubber band strengths~~
    b. ~~Change rubber band placement~~
    c. ~~Replace backstop with spacers to fix string clearance~~
    d. ~~Install “back shield”~~
    e. ~~Add locking mechanism~~
2. ~~Extend and Replace Roller Mechanism and Link to intake~~
3. ~~Install gears for drive base Motors~~
    a. Install motor shield
4. Wire Up Motors and Pneumatics
    a. Install air tanks
        i. Test for leaks
5. Endgame Launchers
6. Install Decorative Delrin
7. ~~Code driver program~~
8. Code Autons
    a. Skills
    b. Right Side

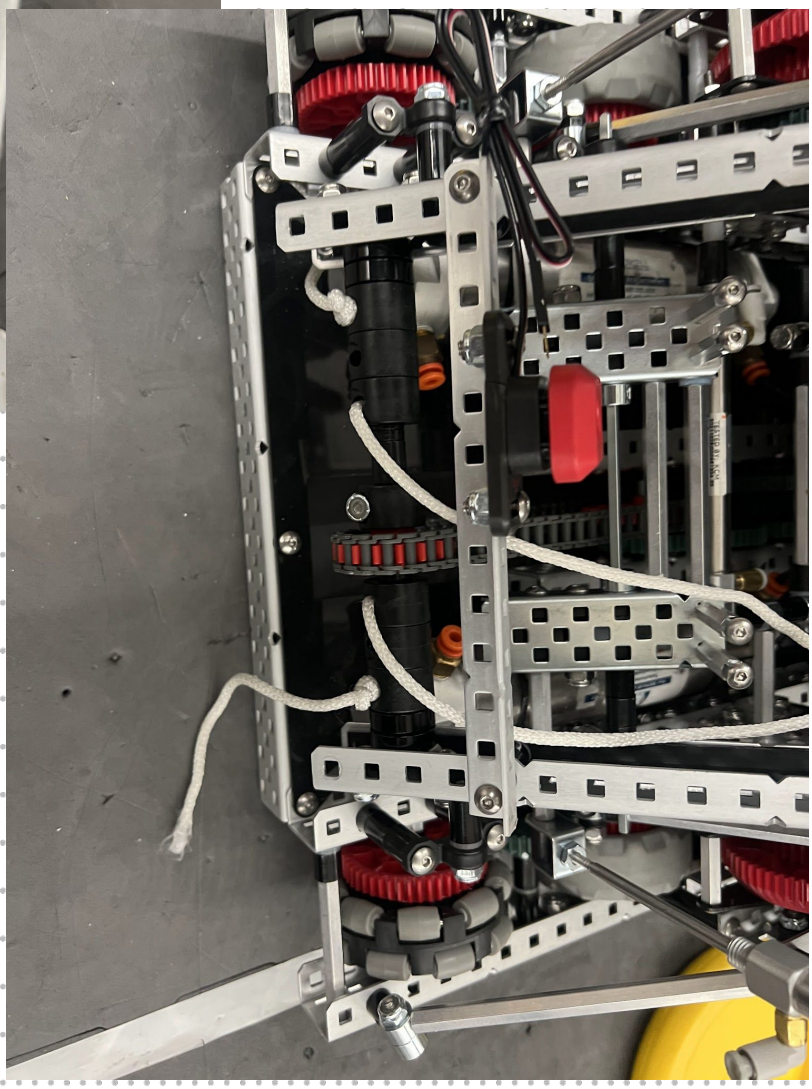
Drive Base Testing

- Again today we attempted to test the drive base but we were unsuccessful
  - The skipping was still occurring in the fast mode
    - Today we decided to replace some other gears and see if that was the issue
      - This time we're beveled and replaced the small moving gears to see if that would fix it.
        - While we did test with rubber bands to no success, we decided to prepare pneumatics and see if that could fix the issue next time

Misc... Prep for Full Test

- Additionally today we installed the the back stop and push button assembly to detect when to stop the sled and lock it into place
  - To do this we simply cut a angle piece placed it near the back and then mounted it to the rails
    - On this angle piece we then mounted the button
  - At the same time we installed the new backplate for the slingshot
    - We created a new backplate because we decided to further reinforce it by using standoffs directly attached to the sled base (5x 5 wide c-channel)





Project Prep

Name Daniel Overdeck

Date 2/6

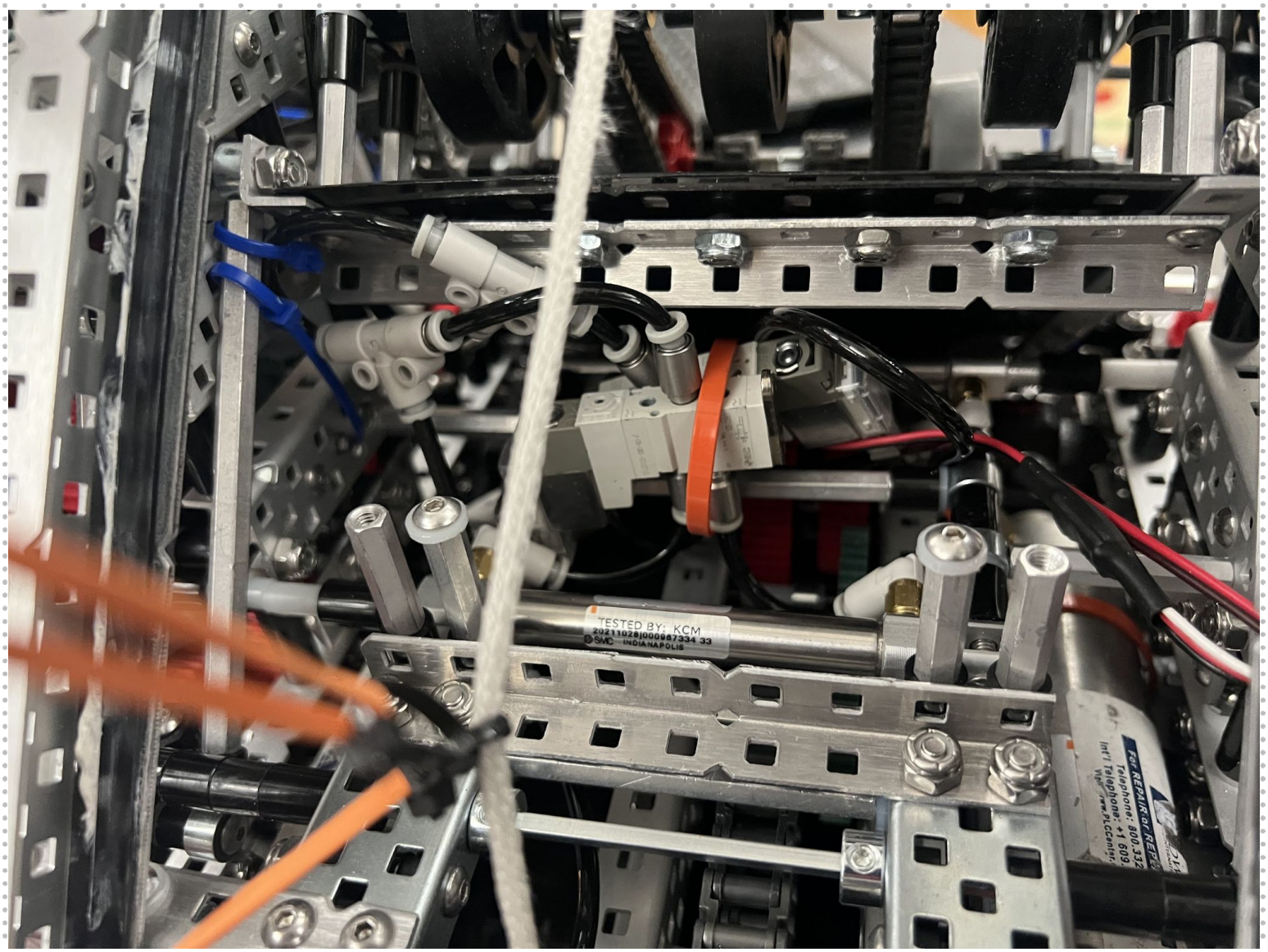
Basic Shooting Code

```
Controller1.ButtonR2.pressed(indexer);
void indexer() {
  solenoid.set(1);
      solenoid1.set(1);
      wait(.2, seconds);
      flywheelToggle=true;
      task pullback_task=task(pullback);
}
```

- Toggles two solenoids that are connected to the clamp that clamps it, is enabled by pressing R2 on the controller. Solenoids when set in 0 are unactivated but when set in 1 are activated therefore the set activates the solenoids causing the clamp to release itself.
- Waits .2 seconds and changes the the flywheelToggle to true as well as calls the pullback method
- By changing the boolean value of the flywheelToggle it allows the pullback macro to be run again as there was an if statement preventing it from automatically doing that when it was clamped down.
- If there was no if statement it would just toggle the task which would ram the sled into the back of our robot instead of staying still.
- Triggers pullback as after it needs to automatically pull it back so it can intake disks as it can only do that when fully pulled down and clamped.

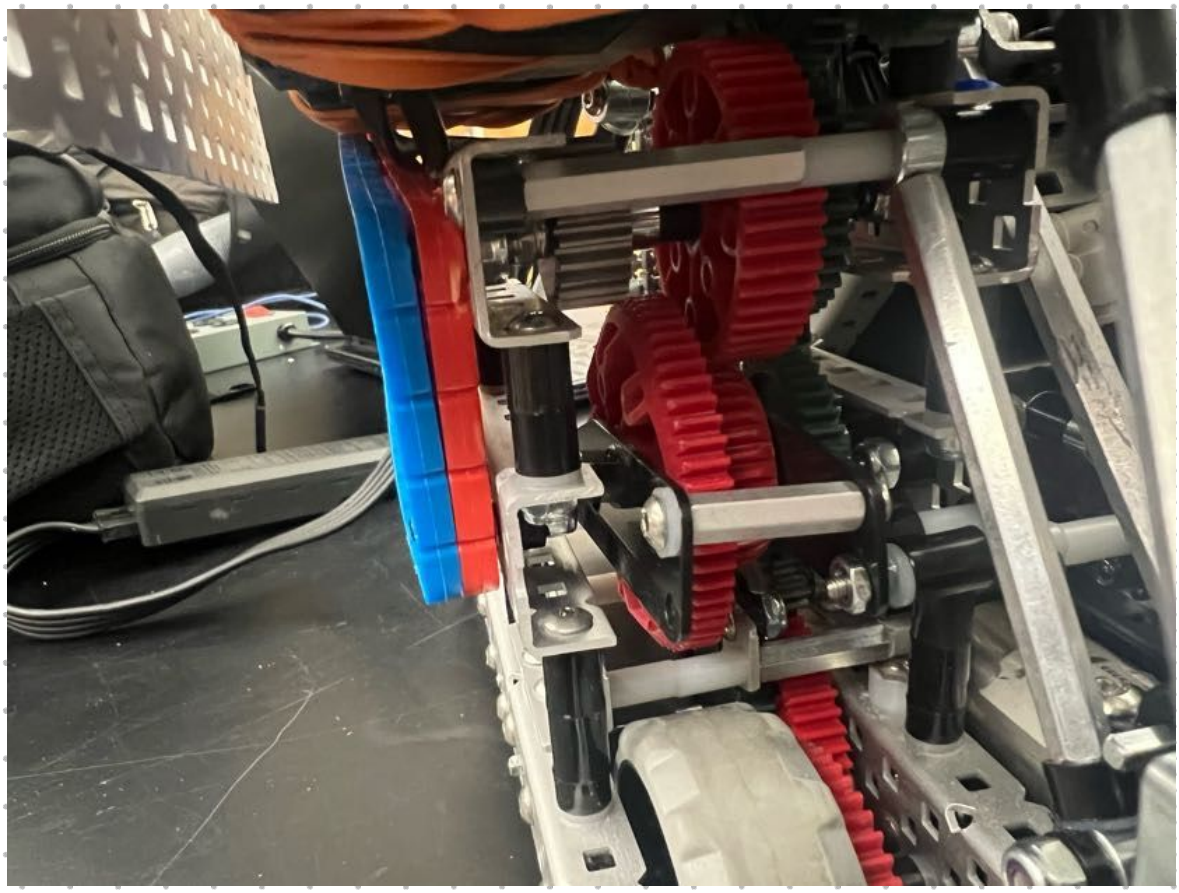
Transmission Testing

- As stated yesterday we decided to wire up pneumatics and install air tanks to test the transmission under pressure
  - We were able to quickly connect all air tanks and solenoids to their respective pistons but we spent a lot of time stopping air leaks
    - Because of this we weren't able to test it today and will have to wait until tomorrow



More Transmission Testing

- Today we finally tested the transmission under pressure
  - The drivebase still was skipping in the fast mode so we further examined it
    - What we noticed is that the transmission struggled to stay in the fast mode and the gears kept on getting pushed out
      - Therefore we decided to rebevel the moving small gears (gears connected to the fast mode) to see if they would stay in place this time
        - Unfortunately we were unable to test it today



Team Meeting:
Week In Review Takeaways

This week we worked mostly on the transmission and drive base. Overall much progress was made, but we need to optimize a few more things to ensure the transmission stays consistent. Additionally, we began the shooting code to set us up for testing (to get ready for Norcal).

Next week, our goal is to continue to optimize the drive base, prepare for Norcal, and finally compete :).

Time in Lab:
Monday - Friday, Sunday; 19 Hours

Slingshot Testing

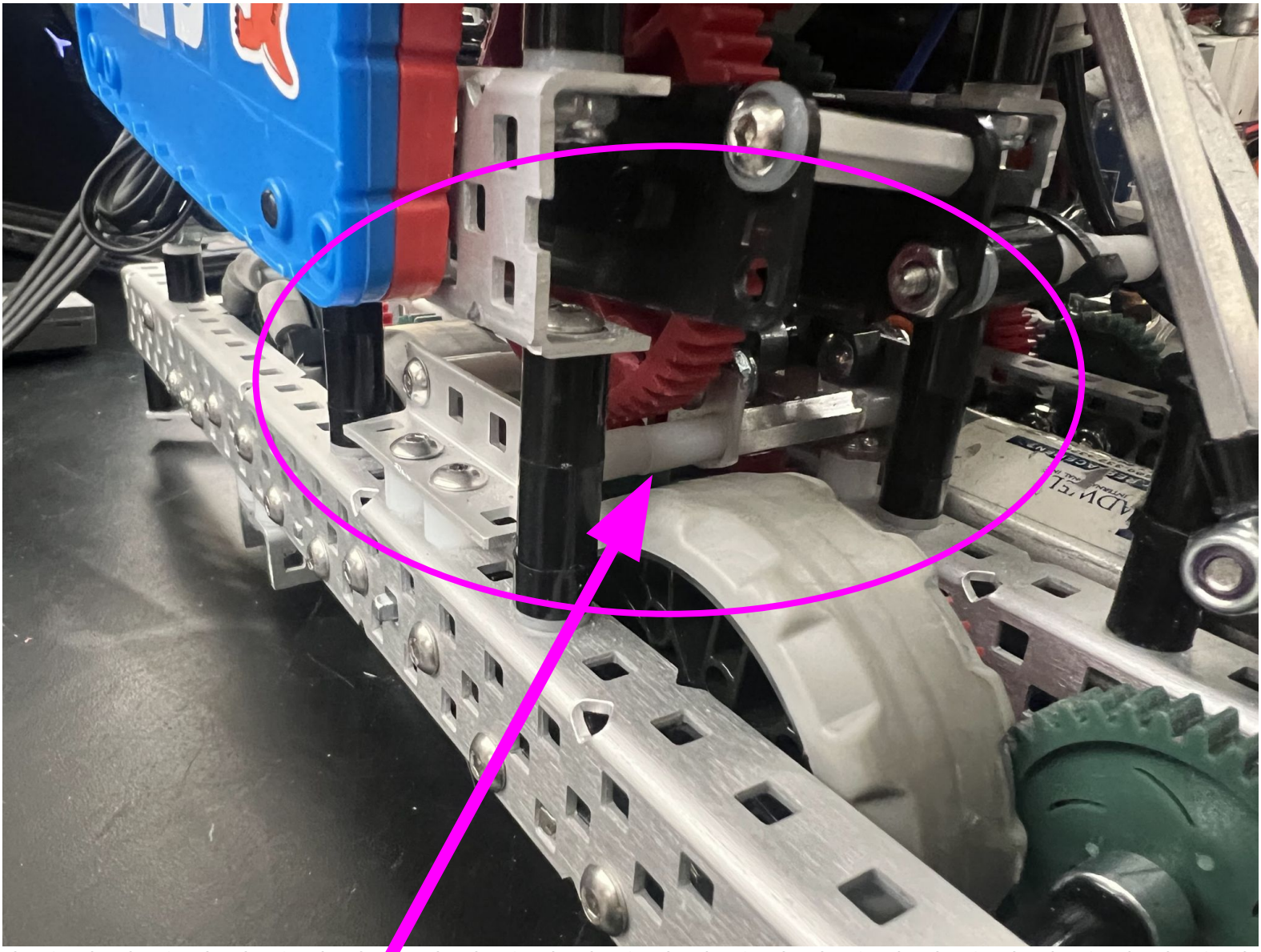
- Today we manually tested the effectiveness of the slingshot
  - The first thing we did was remove the pneumatic wiring from the transmission and install it onto the locking claw/mechanism
    - Even though we know it was counter productive, it was faster than creating a new pneumatic system just for simple testing
  - Then we made sure the pull back code and the claw worked
    - They did… therefore we are confident that it will work for the tournament

Wiring Up Pneumatics

- Following the testing of the slingshot we created the full Pneumatic system as well
  - We then tested for leaks, of course….
    - After detecting and fixing several, the system was complete and ready for operation

Drive Base Testing

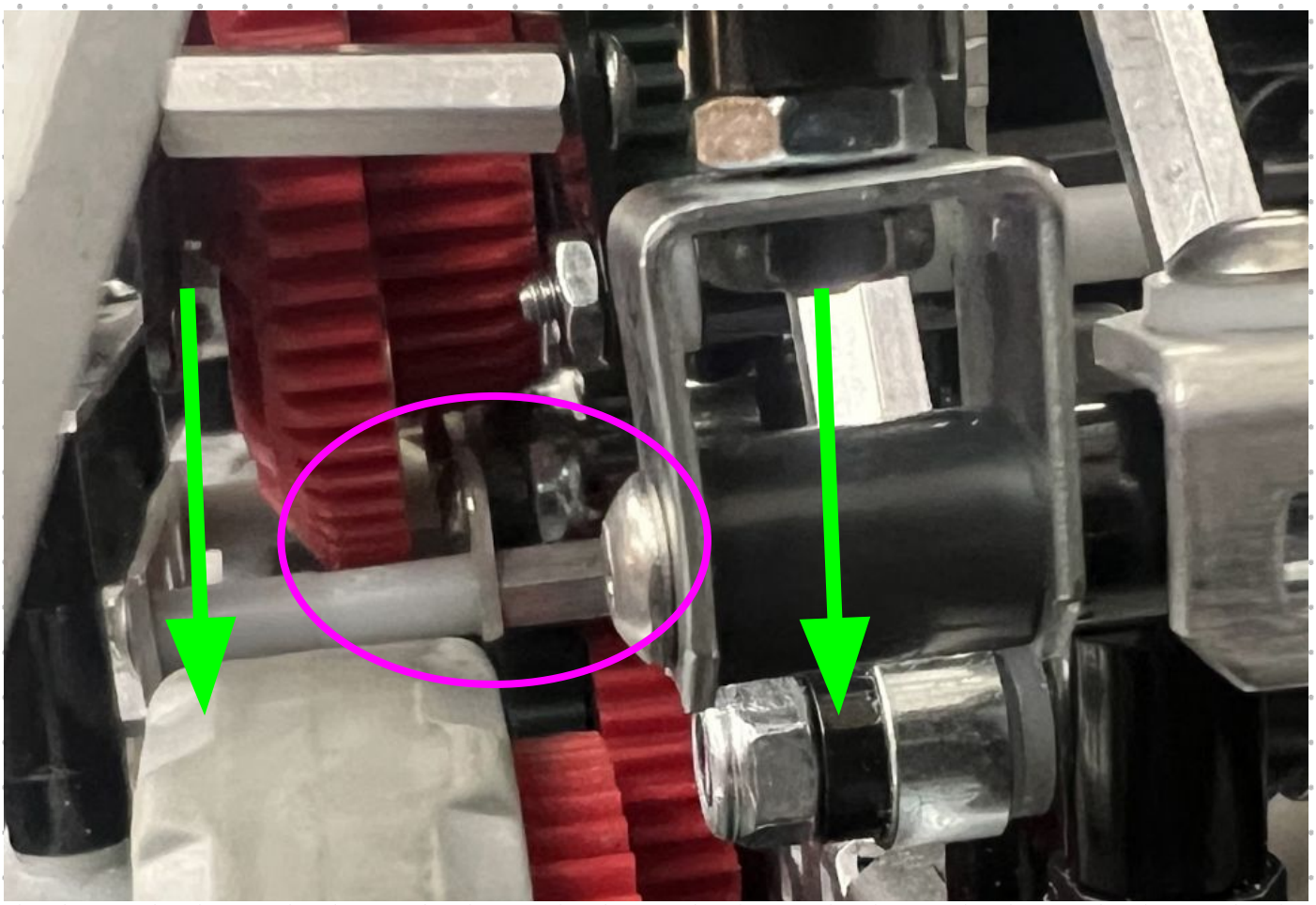
- Today we continued the drive base/transmission testing
  - **The skipping was not gone**
    - At this point we had switched every single transmission gear yet the problem still existed…
      - **therefore we we went black to the drawing board**
- We knew the problem was not the transmission (we had replaced all the transmission gears) and we knew it wasn't the drive base (we had replaced and rebeveled the gear where the drive base connects to the robot several times……..
  - The only thing that could have possible caused the issue at this point was the tiney 12 tooth idler gear connecting the drive base to the transmission….
    - **This turned out to be the issue**
      - When observing the the 12 tooth idler gear we noticed that the gear was moving a concerning amount…
        - While this could have been caused by “the special assembly” where it is housed (the one discussed in the CAD design) it appeared that the collar on the axle holding the gear was loose… therefore we removed the transmission gears to access the collar and secured the axle.
          - **While this helped** the right side of the drive base appeared to skip in only one direction, on closer inspection the bearing block we had bevelled to fit it in “the special assembly” was very rough and was not interfacing well with the spacers
            - We then removed the transmission gears once again, and removed the 12 tooth idler gear and installed a new bearing block **This also helped yet the problem persisted.**



“Special Assembly”

Fixing Drive Base

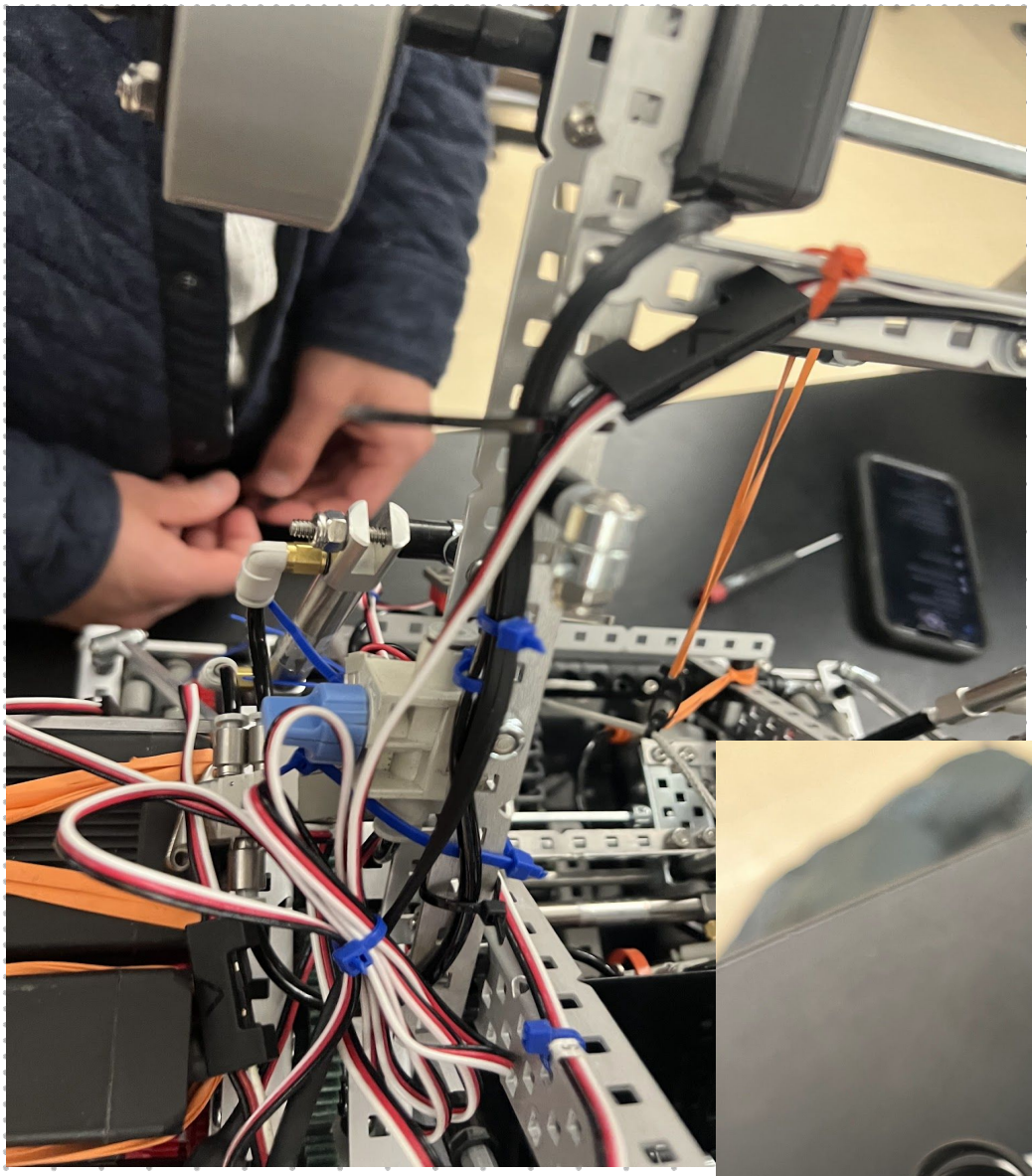
- Today we went back to the drawing board to fix the drive base
  - So far we've spent 2 weeks trying and failing to fix it... and NorCal/States are also getting closer therefore it was imperative we find and fix the problem once and for all today
    - While observing the drive base drive (and skip!! yay :/) we noticed that the transmission gears would stop spinning when the skipping occurred.
      - This led to us manually holding the transmission gears (not letting them spin) and powering the motors... when we did this there was no skipping (the motors only got hot)
        - Therefore we did the opposite. We held onto the transmission gears and spun the drive base... it did skip
- Upon further examination it appeared that the 12 tooth idler gears seemed to be meshing/were to close to the gear connecting to the transmission axle
  - Then the idea that the "special assembly" which held the axle holding the idler gear might have been pushed slightly up therefore we loosened it and tightened it while making sure to push it towards the drive base in order to ensure it meshed properly with the drive base gears.
    - Then we tested it.
      - **It worked!**
        - After 2 weeks we had finally fixed the drive base.. Just in time for NorCal as well!!!!

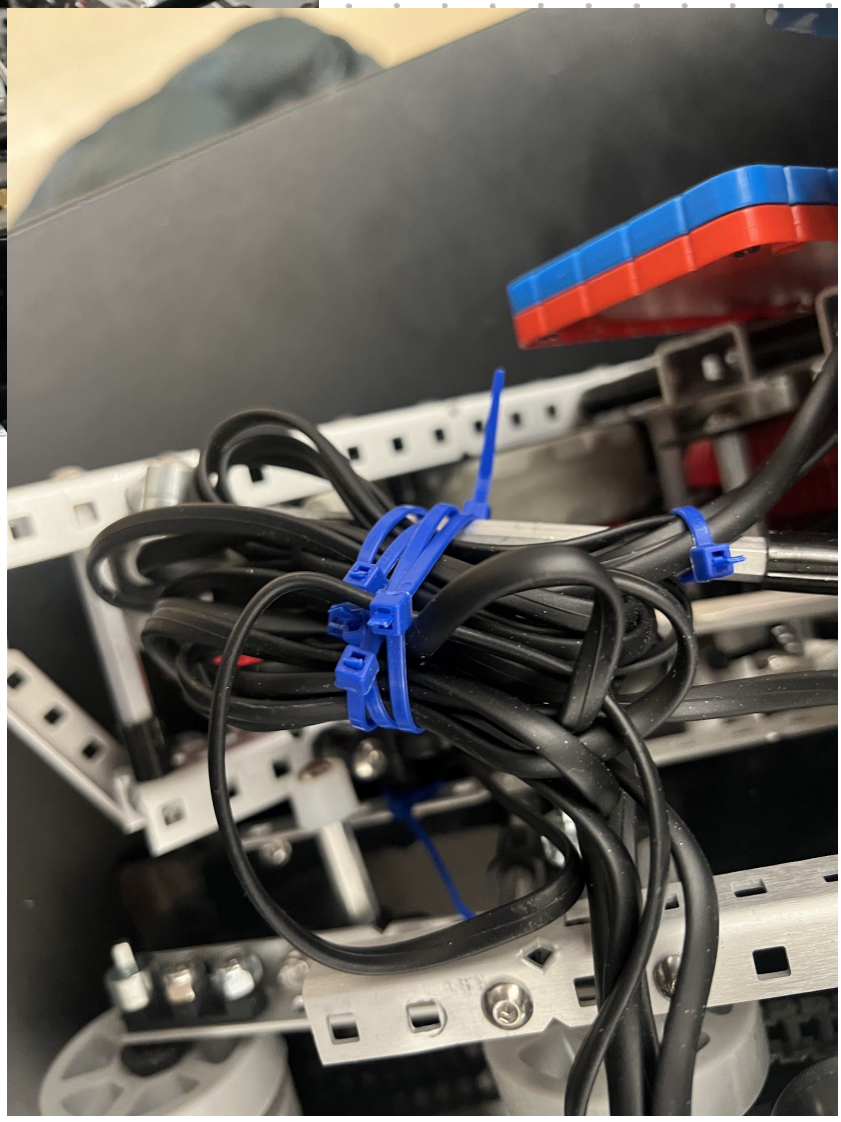


“Special Assembly” needed to be pushed down

NorCal Preparation

- To prepare for NorCal we decided to tie up all loose ends today
  - First we installed the roller mech which was very simple.
    - It was just a high strength axle, some flex wheels, and a sprocket. We then connected it to the intake and it was complete
  - Additionally we made sure to wire up and cable manage every electronic
    - This includes Motors, Sensors, and Solenoids
  - Furthermore we wired up and routed the pneumatic tubes and ran them throughout the robot in optimal locations
  - Finally we Finished the day by installing “rails” on the bottom of the robot to ensure the drive base axles do not get bent during the flight





Project Norcal prep

Name David Date 2/16

To Do List Update:

1. ~~Finish Winch~~
    a. ~~Test with different rubber band strengths~~
    b. ~~Change rubber band placement~~
    c. ~~Replace backstop with spacers to fix string clearance~~
    d. ~~Install “back shield”~~
    e. ~~Add locking mechanism~~
2. ~~Extend and Replace Roller Mechanism and Link to intake~~
3. ~~Install gears for drive base Motors~~
    a. ~~Install motor shield~~
4. ~~Wire Up Motors and Pneumatics~~
    a. ~~Install air tanks~~
        i. ~~Test for leaks~~
5. Endgame Launchers
6. Install Decorative Delrin
7. ~~Code driver program~~
8. Code Autons
    a. Skills
    b. Right Side

Fixing Drive Base Following Flight

- Following the flight to California, Various issues popped up in out robot, **the biggest issue is that the drive base is skipping due to the spacing on the transmission being pushed out of line**
  - To correctly diagnose where this skipping is originating from we placed the robot on the table and observed which gears were spinning and which where not when the skipping occurs
    - **We were able to see that it is the spacing between the gears connecting the motors and the transmitting gears was bigger than before**
      - Therefore to fix the skipping we unscrewed part of the transmission assembly and decreased the spacing
        - **This fixed the skipping :)**

NorCal Sig Event

Miscellaneous Repairs

- **As stated earlier the flight caused various issues due to the immense vibrations therefore we had to do some miscellaneous repairs**
  - One for example was retightening various collars and axles to make sure they were fine
  - Additionally we had to reinstall various screws that had gotten loose
  - We also had to remove new leaks that had formed in the pneumatics system

Pull Back Skipping

- **While trying to test pull back and get some good work done on the robot during the tournament** (currently it is only working as a push bot with a transmission since we can not shoot) we noticed that t**he sled seemed to be skipping 3 / 4 the way through pull back**
  - It would just kinda bounce up and down and not go back all the way
    - To diagnose the issue we decided to flip the robot over and examine the pull back gears, one of which noticed seemed to be toasting individually of its axle

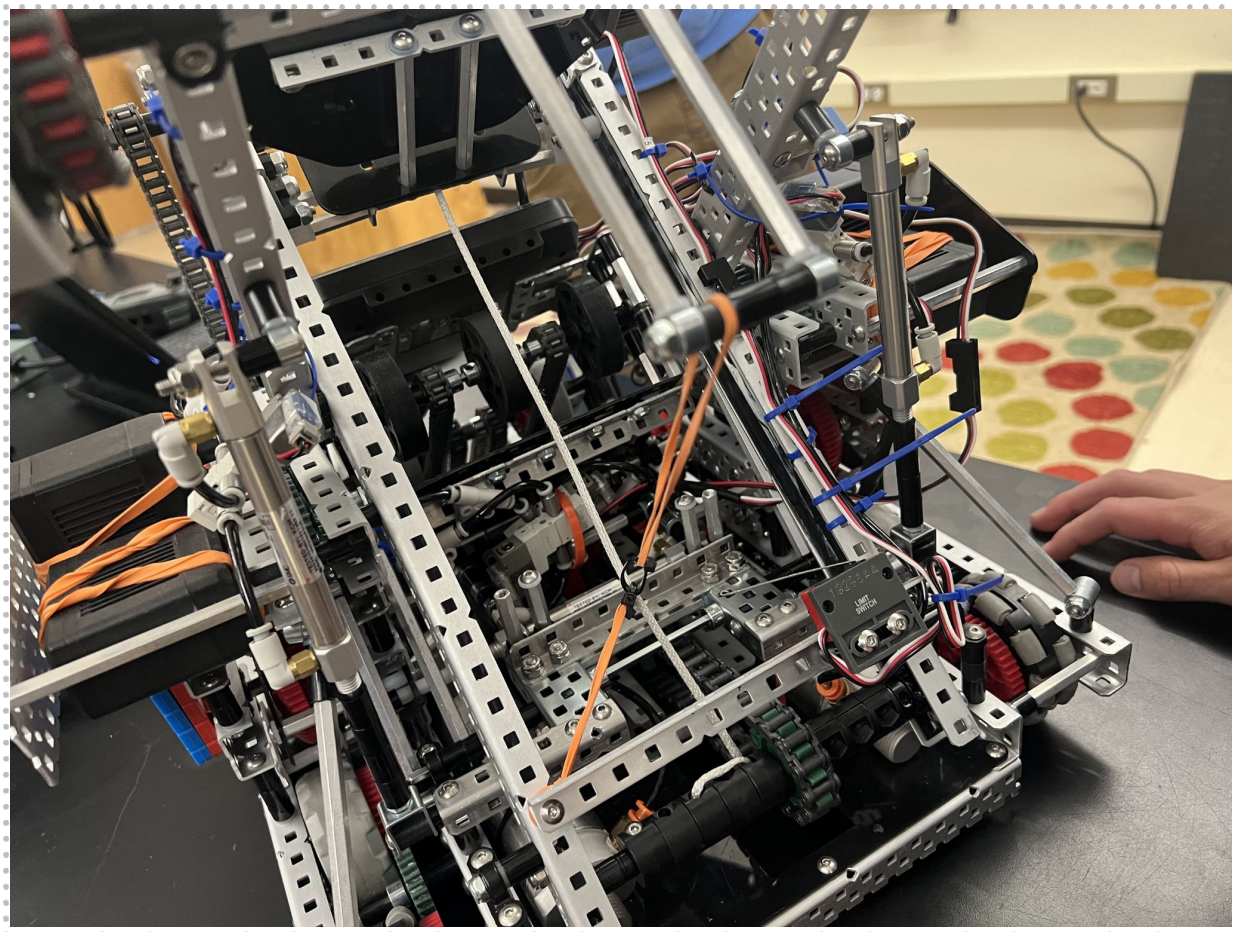
NorCal Sig Event

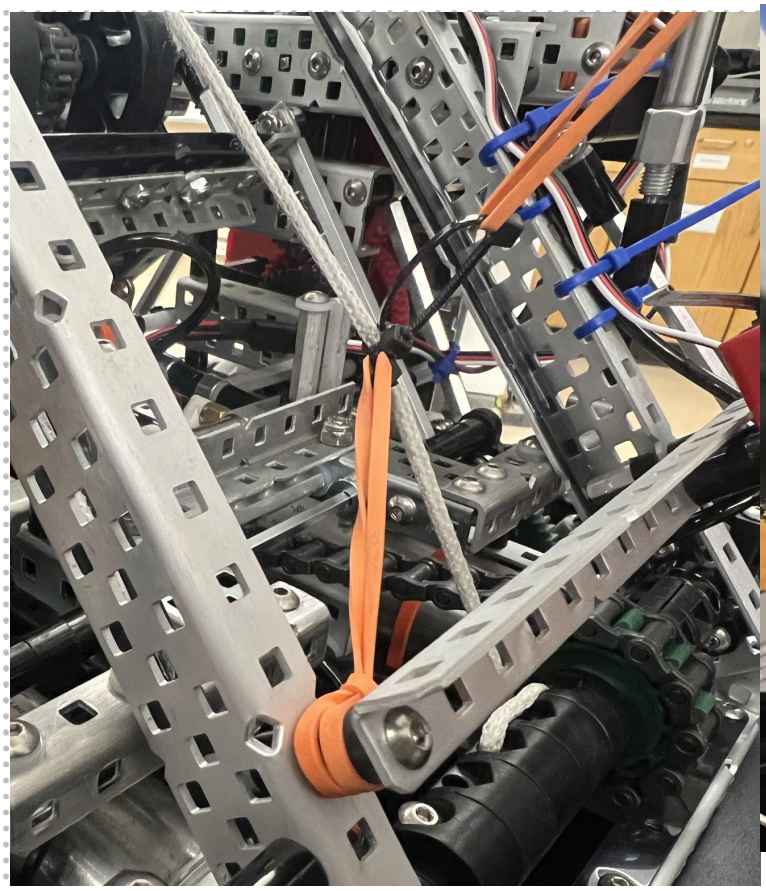
Fixing Rounded Pull Back Gears

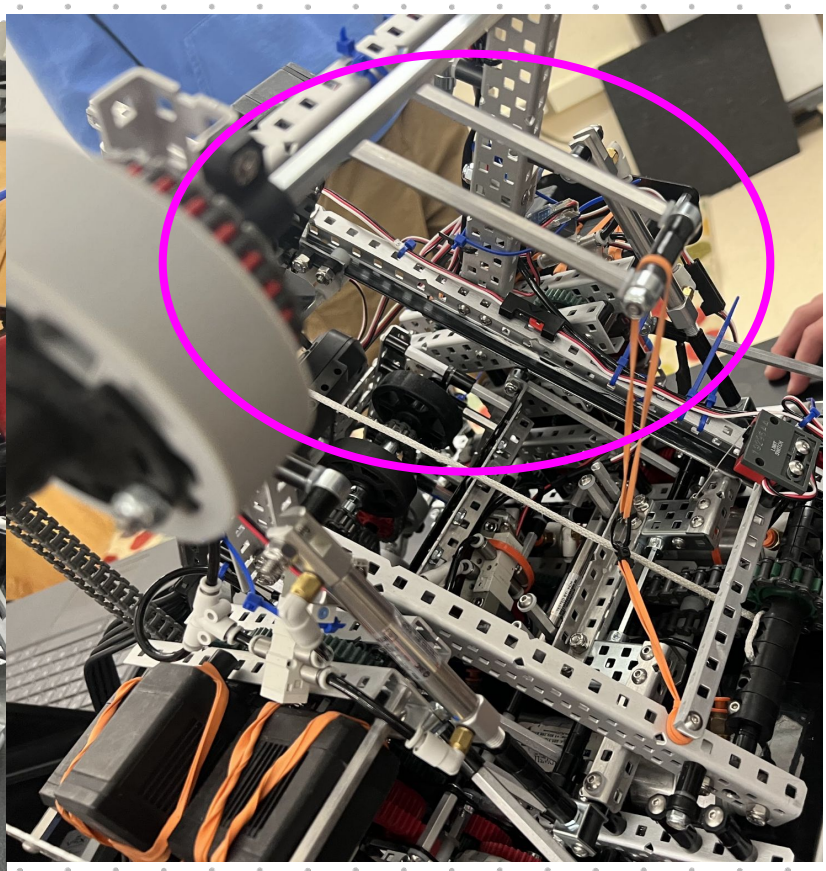
- **As stated earlier one of the gears seemed to be rounded (**in other words it would slip around its axle and hold the tension of the sling shot
  - To replace this gear we forced to remove the air tank to access the gears mounted on the c channels attached to the base place
    - **Once we replaced it, pull back worked again**

Pull Back Modification (pull back guide)

- **Another issue we noticed was that the chain pulling back the sled seemed to be skipping**
  - After careful observation we were able to deduce that the string attached to the sled appeared to be hitting the chain and causing it to skip
    - Therefore we attached rubber band to a zip tie around the string to hold it away from the chain
      - **A**dditionally when unwinding, the string sometimes got stuck **therefore we created a structure to hold another rubber band above the string so it could unwind freely as it would be lifted up by the band**







Project Norcal

Name Ben Date 2/18 Page 100

NorCal Sig Event

Remapping Controller Layout

Ben communicated that the transmission would work best when the motors are moving. Our transmission button was on **‘B’** which was hard to reach while driving since the thumb would need to leave the left stick to activate transmission. Therefore, we decided to assign the transmission to the **L1** button, which was more accessible.

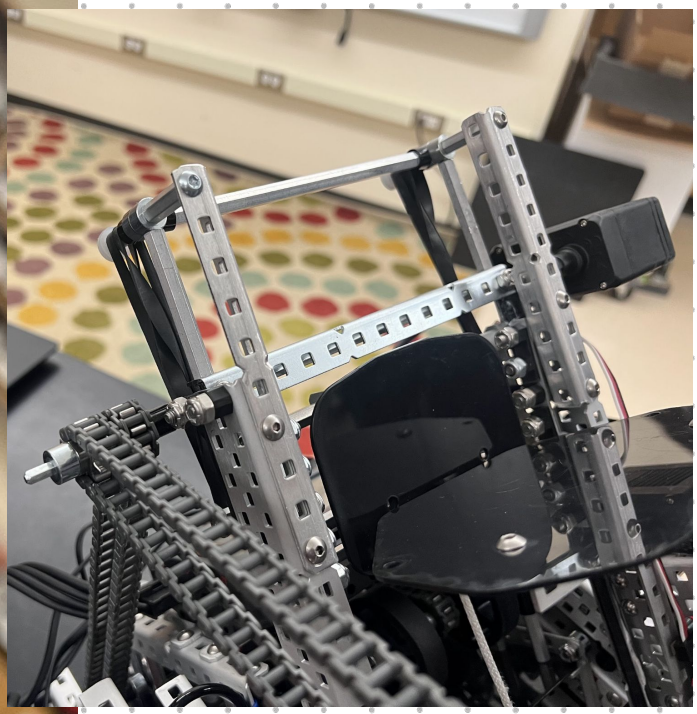
NorCal Sig Event

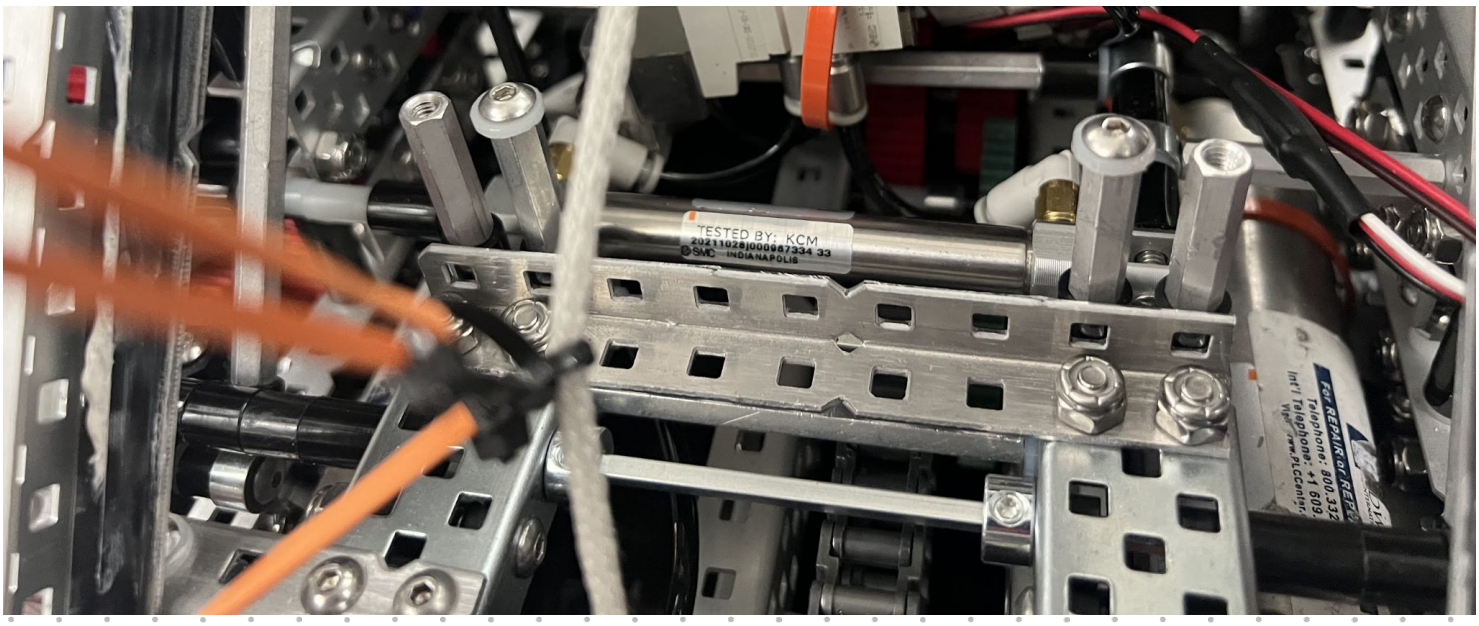
Reinforcing Rubber Band Holder

- Another issue we noticed is that the rubber band holder, which was attached to a angle piece connected to the end of the slingshot rail was bending downwards when pulled back, therefore causing the groupings of the sling shot to be quite as the bending changed the direction of the tension as the sled was being shot causing discs to fly directly upwards
  - **To fix this we installed some angle pieces on the top of the slingshot rails so and added standoffs connecting it to the old assembly reinforcing it**
    - **Now it works and does not bend**

Intake modifications

- **Additionally we noticed that the intake seemed to rub the ground when driving therefore we decided to trim the intake so that it would no longer rub**





Project Norcal

Name Ben Date 2/19 Page 103

NorCal Sig Event

Testing Claw/Clamp

- **Today we also began testing the clamp**
  - What we noticed is that while the clamp seemed to be able to hodm the sled back at first, after a few shots it would stop releasing and after a few more with us manually releasing the sled it could not hold the sled back
    - **Therefore we decided to observe the clamp carefully**
      - What we noticed is that the standoffs at the **end** of the standoffs were acting as hooks on the end of the clamp allowing it to hold but not release

Clamp Modifications

- Therefore we removed the spacers on the end of the standoff and tested the clamp
  - What we noticed at first was that the standoffs were now not long enough to make contact with the claw and hold it back
    - Therefore we added standoffs to the bottom of the standoffs to extend them while also not hooking the sled
      - Then we tested it again but the sheer force of the sling would bend the clamp out of place
        - **Therefore we decided to try to change the cla mo to a screw joint but this did not help → we will have to redesign it back in Virgnia**

NorCal Sig Event

Auto Reload Program (Problem)

In between our matches at NorCal, I attempted to create a shoot and reload macro that would work all as one button. For reference, current control scheme had two separate buttons for winding and unwinding to control our shooting mechanism. In addition, our drivebase would stop working when we pressed that command and we didn't disable our intake, which could lead to jamming in the future.

```
if (flywheelToggle){ // wind up
WindingMotorA.spinFor(reverse,5,seconds, 100, velocityUnits::pct);
flywheelToggle=false;
}


if (flywheelReverse){ // wind down
  WindingMotorA.spinFor(forward,5,seconds,100, velocityUnits::pct);
  flywheelReverse=false;
}
```

Our original code had us using the **SpinFor** method to wind and unwind, which presumably caused our drivebase to freeze since the method would override the drivebase code, disabling it for a short period of time

NorCal Sig Event

Auto Reload Program (Solution)

My attempted solution to this problem was to utilize both **SpinFor** methods in the our shooting function and create a boolean to disable our intake.

```
void shooter() {
  solenoid.set(1); //release sled
  solenoid1.set(1); //release sled
  intakeToggle = false;
  wait(2, sec);
  WindingMotorA.spinFor(reverse, 5, sec, 100, velocityUnits::pct);

  solenoid.set(0); // clamp sled
  solenoid1.set(0); // clamp sled
  WindingMotorA.spinFor(fwd, 5, sec, 100, velocityUnits::pct);
  intakeToggle = true;
```

```
// Intake
    if (Controller1.ButtonL2.pressing()&& intakeToggle==true) {
      Intake.spin(forward,100, percent);
```

My attempted solution to this problem was to utilize both **SpinFor** methods in the our shooting function and create a boolean to disable our intake. This will prevent us from jamming our slingshot by intaking too early.

NorCal Sig Event

Auto Reload Program (Reflection)

I was only able to test the code once. When I tested it, the result was not what I expected. The shooter worked, but the claw clamped before the sled fully rewinded. The drivebase was also disabled during this process. Overall this attempt was a **failure.**

Looking back, I probably should have utilized a **task** to run the shooting mechanism simultaneously with the drivebase. I also could have tried using a **for loop** to automate the winding sequence.

NorCal Sig Event

Change Direction Program

In addition to an auto reload program, I attempted to code a macro to change the direction of the motors while driving. This macro would make it easier to switch between our intake side and wedge side without having to drive backwards.

```
void changeDirection() {
    leftMotorA.setReversed(true);
    leftMotorB.setReversed(true);
    leftMotorC.setReversed(true);
    rightMotorA.setReversed(true);
    rightMotorB.setReversed(true);
    rightMotorC.setReversed(true);
    }
```

My method for making this was to make a function that would call the motor.**SetReversed** method to reverse the current direction of the motors when pressed. When I tested the program, it didn't work since our center wheel was set to reverse by default, so the side motors were working against the middle motors. In addition, after swapping direction, it wouldn't change back.

Looking back, I should have utilized a boolean variable as a parameter and have it after each function call.

NorCal Sig Event Conclusion

What Went Well

- Building
  - Drivebase fixed
  - Transmission works
- Won several matches
- Scouting system established

What Did Not Go Well

- Building
  - Shooter/pneumatic claw did not work
- Programs
  - Auto reload did not work
  - Change Direction did not work
  - Roller auton did not work
- Driving
  - Got stuck on disks
  - Got stuck on barricade

NorCal Sig Event Conclusion

Driving Conclusion

Norcal was a tough tournament. There were many skilled teams we played against and we were without our lead driver, Arav. I volunteered to drive as his replacement since I drove last year on 12Z and I was familiar with game strategy. My driving skills, however were subpar without practice. Our robot was limited to defense and rollers, without an endgame, close matches didn't go in our favor. For what we had, I think I did the best that I could.

Initially I played strictly defense the whole match, and even then I would overextend or be outmaneuvered. After a couple matches I learned to prioritize rollers in the last 45 seconds.

Our driving taught us lessons about the robots potential issues during match. For example, during our 4th match, I attempted to drive over the barrier and got stuck. Some games our bot got stuck under disks as well. From these experiences, we are more prepared for when we will encounter them in the future.

Team Meeting:
Week In Review Takeaways

This week we had an incredible experience at the NorCal Signature event as recapped in the previous slides.

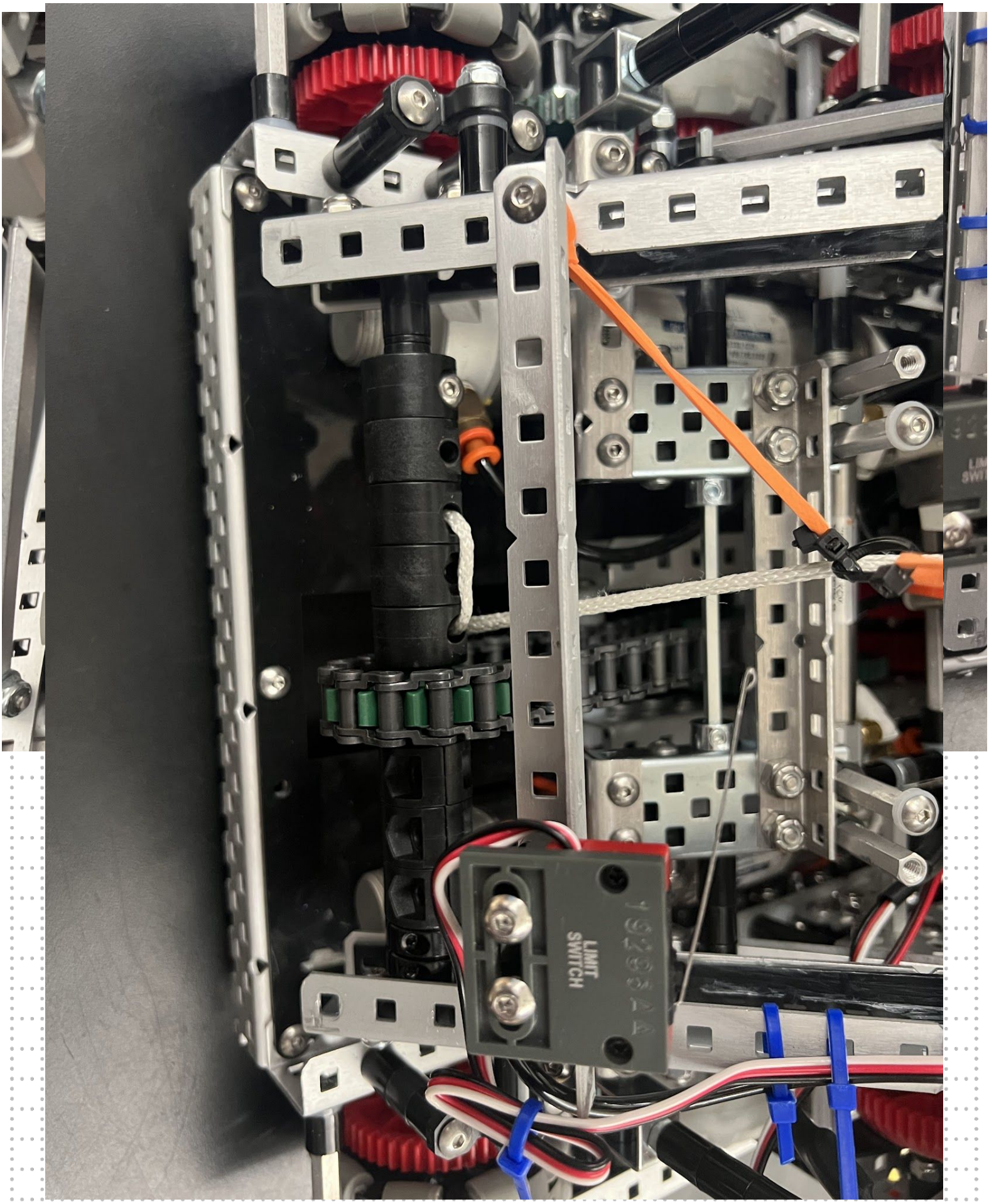
Next week, our goal is to rebuild claw/clamp, continue testing, and continue coding.

Time in Lab:
Monday - Wednesday, Thursday - Sunday (in NorCal); 28 Hours

Rebuilding Claw

- Following NorCal rebuilding the claw was the only thing necessary left to repair before the robot is complete
  - As stated earlier, the play which came from the “flimsiness” of the claw/clamp was allowing the sled to slip out before we actually shot
    - This flimsiness stemmed from the fact that the claw was largely made up of standoffs.
      - The first repair we attempted was to install an angle piece brace across the central standoffs
        - While this did help a little bit, **It was not Enough** when we re-installed the new claw on the Robot
  - The next thing we tried is to replace the side standoffs that connect to the pistons themselves. To lessen the play in between the pistons and the claw we replaced the standoffs with angle pieces with screwed on collars to create the screw joint.
    - While we were unable to test it today we are optimistic that this new claw will work well

- At the same time we replaced the push button with a limit switch since the force required to push the button was affecting the sled and the rubber band tensioners were getting in the way as well



Project Rebuilding Claw

Name Ben Date 2/21

Rebuilding Sled

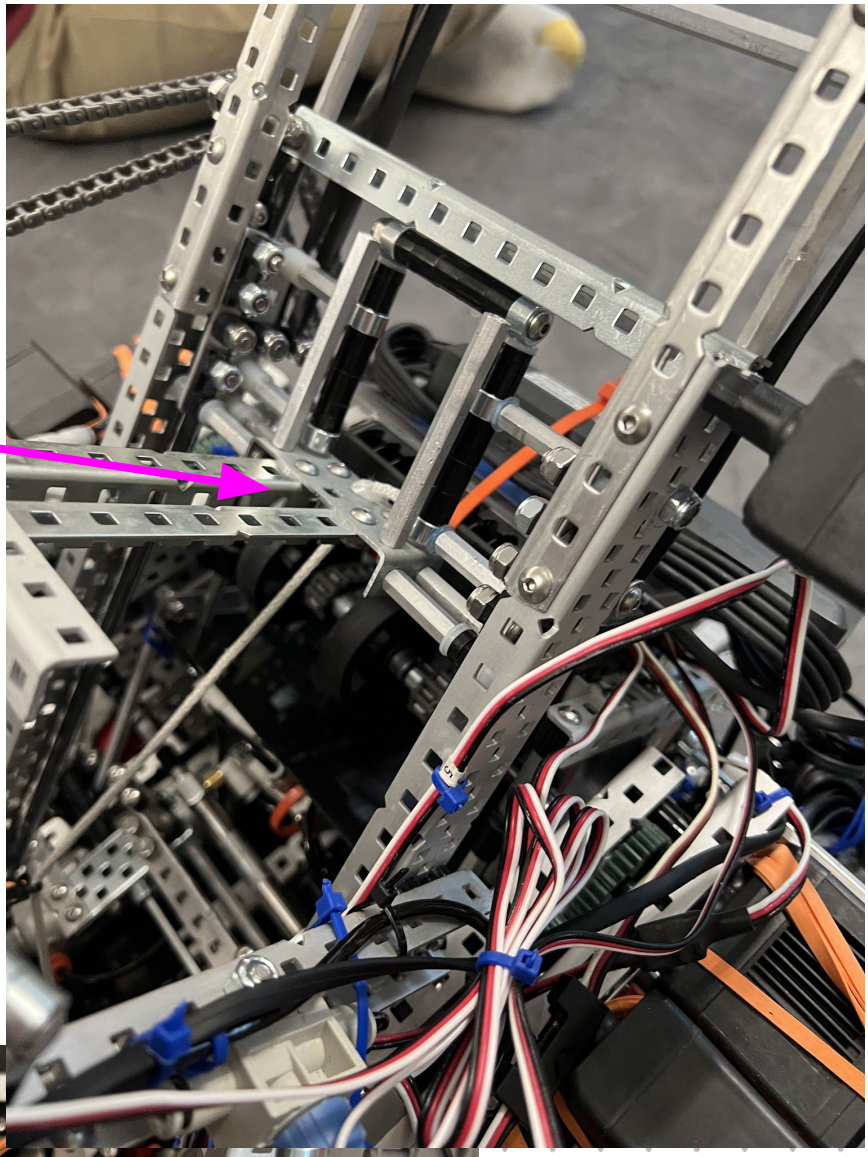
- Today we began by rebuilding the sled once again **(version 4 at this point)**
  - We did this because our groupings were extremely bad when testing manually
    - This is because the way the standoffs were attached to the base, they kept on getting loose allowing the discs to travel separately/unpredictably
  - Rather than using a 5 long 5 wide C channel sliding lengthwise across the rails, we used a 2 long 5 wide channel mounted perpendicular to the rails in order to have 2 contact points with the discs thanks to standoff (more stable when compared to a sled where the discs only touched at the bottom).

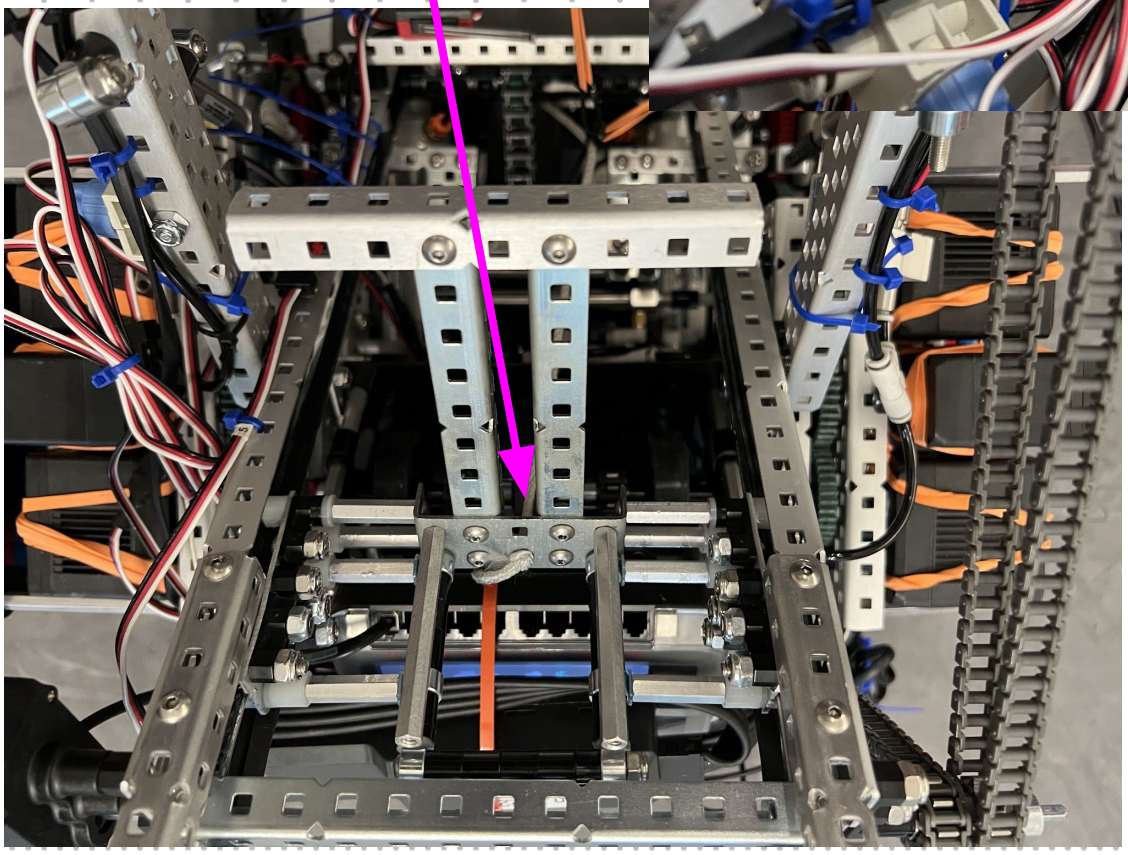
Testing Claw/Shooting

- Once the new claw was created and installed, we began to test shooting
  - The first problem we faced is that the new code for the limit switch (which replaced the push button) did not work for some reason. **More details in next entry**
  - One problem we noticed is that the clamp was clamping immediately after it hit the switch → therefore we delayed it by a few seconds. **More details in next entry**
  - The third problem we noticed is that the discs slide a lot when driving around → therefore we will design and cut inner side panels to ensure that the discs don't slide out when in the loading position

Sled V4

New Perpendicular C channel





Pullback Macro for Bot

```
task display_task = task(display);
if (!flywheelToggle){
task pullback_task=task(pullback);
}
BumperA.pressed(stopPullback);
int pullback(){
    FlywheelMotorA.spin(forward, 100, pct);
     return 1;
}
void stopPullback(){
  solenoid.set(0);
    solenoid1.set(0);
    FlywheelMotorA.stop(coast);
    wait(.2, seconds);
    FlywheelMotorA.spinFor(reverse, 5, seconds, 100, velocityUnits::pct);
}
```

-When a limit switch is toggled in the “BumperA.pressed” portion it activates the stop pullback portion of the slingshot which clamps the sled down and unwinds the string. Afterwards the sled can shoot discs by using a button to release the clamp

-Other part automatically pulls back the Sling until it hits the limit switch.

-It does this by spinning the sled in one direction constantly in a method that is constantly running aka a macro.

Transfer Limit Switch Issues

```
if (BumperA.pressing()) {
  solenoid.set(0);
  solenoid1.set(0);
   FlywheelMotorA.stop(coast);
   wait(.2,seconds);
   if (flywheelToggle){
   FlywheelMotorA.spinFor(reverse,6,seconds, 100, velocityUnits::pct);
   flywheelToggle=false;
   }
   if (flywheelReverse){
     FlywheelMotorA.spinFor(forward,6,seconds,100, velocityUnits::pct);
     flywheelReverse=false;
   }

  }
  else if (flywheelToggle) {
  FlywheelMotorA.spin(forward,100,pct);
  }
  else if (flywheelReverse){
   FlywheelMotorA.spin(reverse,100,pct);
 }
else {
  FlywheelMotorA.stop(coast);
 }
```

Transfer Limit Switch Issues Continued

-Because our old program for the Sled used a Bumper as shown to stop the sled if it went to far we had to search up the API For the Limit Switch sensor

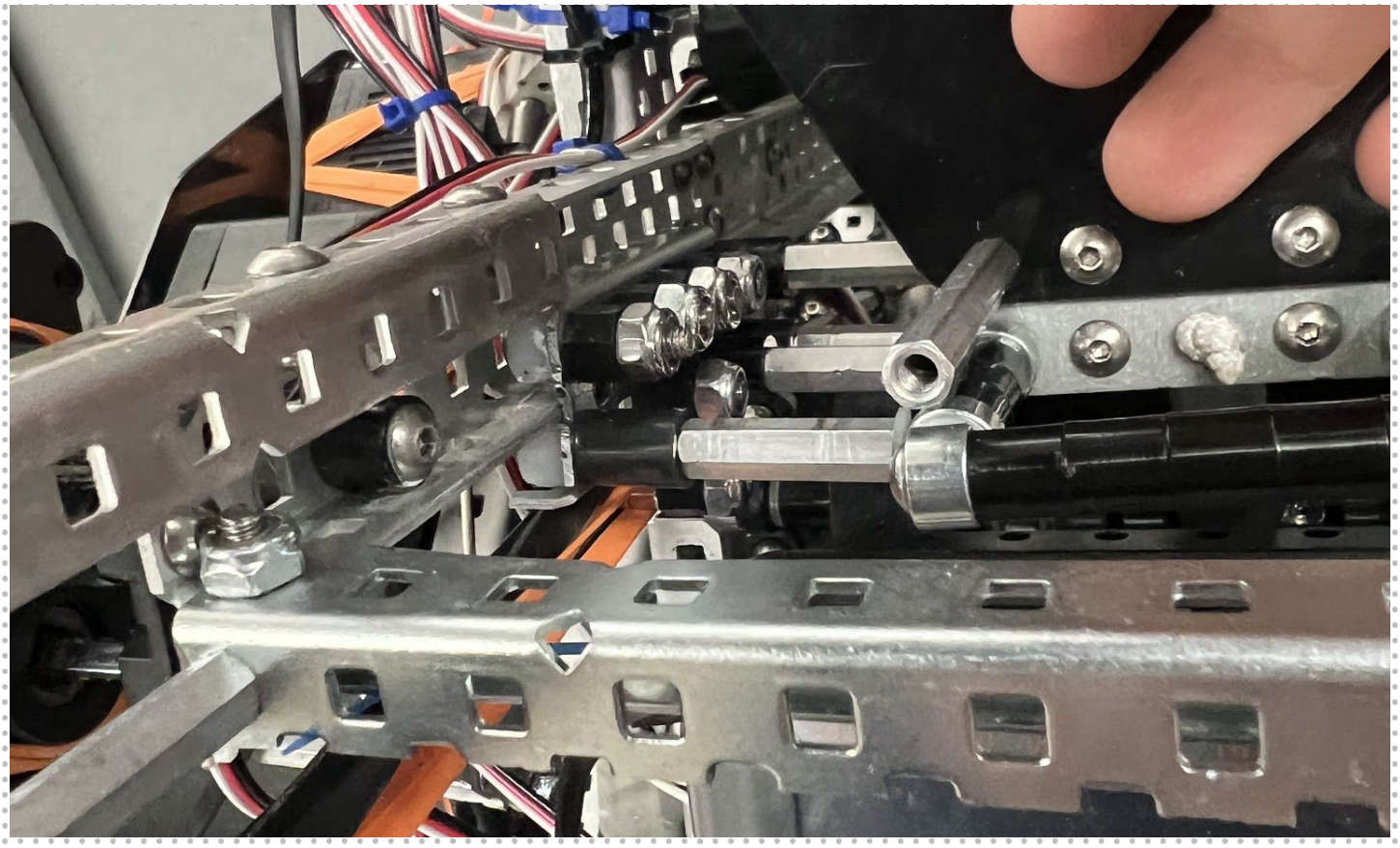
-We tried using the same code but with redefining Bumper as a limit sensor and using “pressing” the limit switch as an indicator that it was touching. This did not end up working so we had to find a new approach.

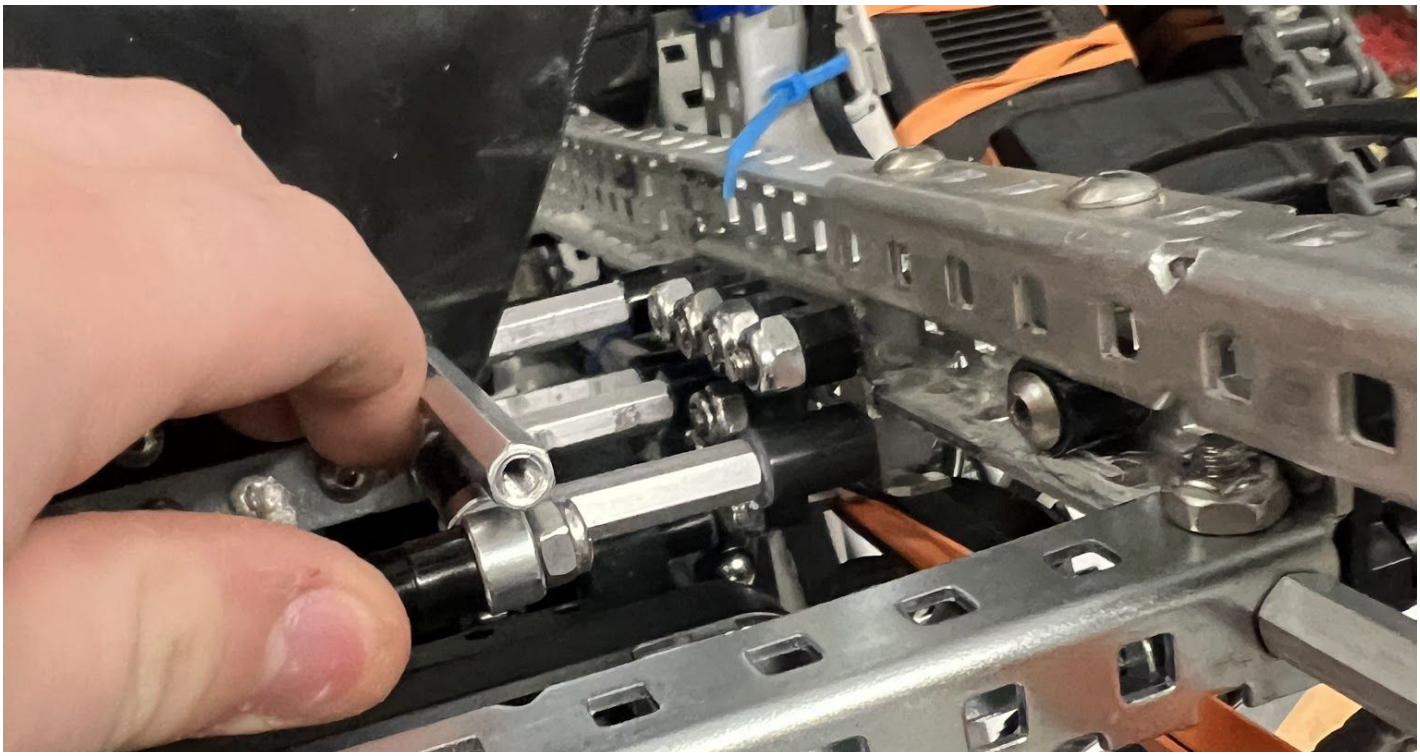
-Instead of that I used limit switch pressed and had it callback the method that would stop the sled from moving as well as clamp down, which resulted in the stop pullback method.

-We decided on using a limit switch as it required less tension to actually push it down with the limit switch generally being more consistent and easier to stop in the case of it not triggering as the point of impact is farther away.

Rebuilding Sled

- **Due to extensive testing yesterday,** when the timing of the clamping program was wrong → **causing the pneumatics to crush the sled against the backstop** since it was in the incorrect position (the clamp is supposed to activate when it is under the halfway point of the sled, any farther back and the pneumatic will just catch but will crush the sled)
    - This **caused the sled to deform and not slide easily**
        - Therefore we decided to rebuild the sled (the **only difference being that we used some modified 2 wide steel c channel rather than aluminum bar pieces since they would be less likely to deform**
            - For more detail check next page
- When we reinserted the sled, we noticed that it now was not sliding as easily
    - We then looked back at the deformed claw and noticed that the aluminum bar was bent out increasing the space between the spacers (acted as sliders) which allowed it to slide freely
        - Therefore → **we removed the delrin rail piece since it was too thick** and instead put the sled on the c-channel directly
            - While we theorized that this might be bad for friction, the sled was just fine
- **Additionally today we installed the inner side panels**
    - These side panels make sure that the discs stay in place when driving around and don't fall of the sled
- **Finally we also created the structure of the wedge**
    - The wedge exists to help lessen the traction of another teams robot to help with pushing by lifting the enemy robots front wheels → to create it we just attached some angle to the end of the rail c-channels and then attached a lengthwise c channel to support the plastic





Delaying Clamping

```
void stopPullback(){
  FlywheelMotorA.spin(reverse, 100, pct);
  wait(.4,seconds);
  flywheelToggle=false;
  solenoid.set(0);
    solenoid1.set(0);
    wait(.2, seconds);
    FlywheelMotorA.stop(coast);
```

-We had to add a wait in between the flywheel reversing in the opposite direction from the pullback so that it would start to move up the railing so that the clamp would clamp down on it just before it reached it

- This was created so that the clamp could use less air in clamping it down as it would now have something to clamp down onto.

-This required adjusting the time of the wait so it wouldn't overpull or under pull the sled into the wrong position

-After it was done clamping the code would then stop the flywheel and wait for the indexer to be used to shoot the discs from.

Remapping Controller Layout

```
Controller1.ButtonR1.pressed(toggle_flywheel);
Controller1.ButtonR2.pressed(toggle_flywheel1);
Controller1.ButtonUp.pressed(buttonToggle);
Controller1.ButtonL2.pressed(toggle_intake);
Controller1.ButtonL1.pressed(toggle_reverse_intake);
Controller1.ButtonB.pressed(rightToggle);
Controller1.ButtonDown.pressed(indexer);
```

-For the code we had to remap where we were going to put the buttons for the new robot.

-We decided that R1 and R2 would both be for changing the direction of the direction of the slingshot in either unwinding or winding it, the Up button would change the speed of the Robot, L1 and L2 would both be for the intake with L2 intaking and L1 outaking and B would be for transmission while the Down button would be to shoot the discs.

-Later on we decided that a flywheel macro would be better than toggling it in one direction to the other direction and having it stop so we changed the code to do that

-By pressing the button the code will callback as specific method, whether that be changing a boolean value or calling an independent method for the robot in the main.

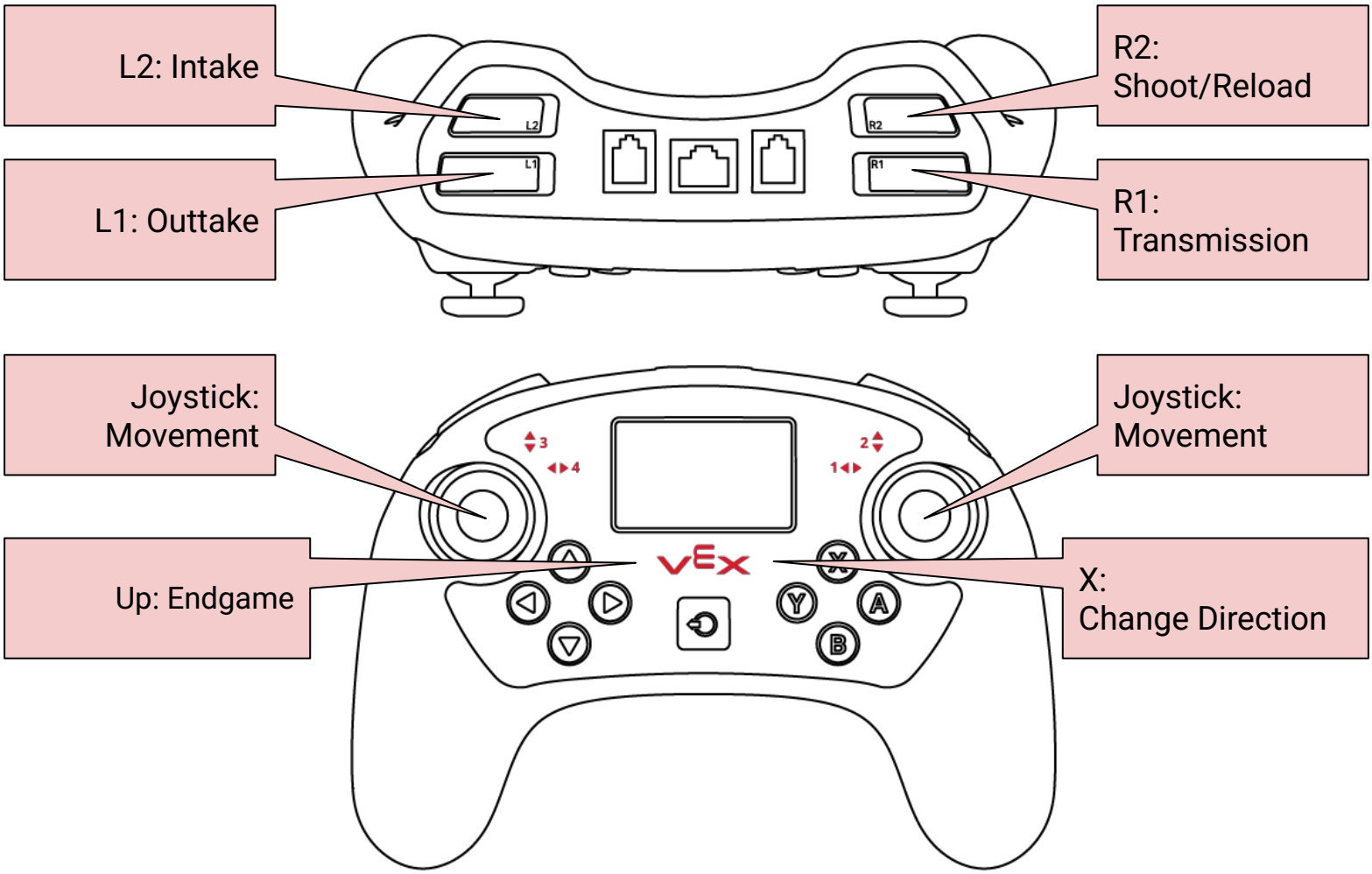
Remapping Controller Layout

```
Controller1.ButtonUp.pressed(buttonToggle);
Controller1.ButtonL2.pressed(toggle_intake);
Controller1.ButtonL1.pressed(toggle_reverse_intake);
Controller1.ButtonR1.pressed(rightToggle);
Controller1.ButtonR2.pressed(indexer);
Controller1.ButtonX.pressed(swtich_motor);
```

-Because the slingshot pulls back by itself it frees up both the R1 and R2 buttons which we like using as they require much less effort for the driver to press as they are placed in a more natural position.

-We kept the intake Buttons where they were as well as the slow down button in the same location as where they previously were

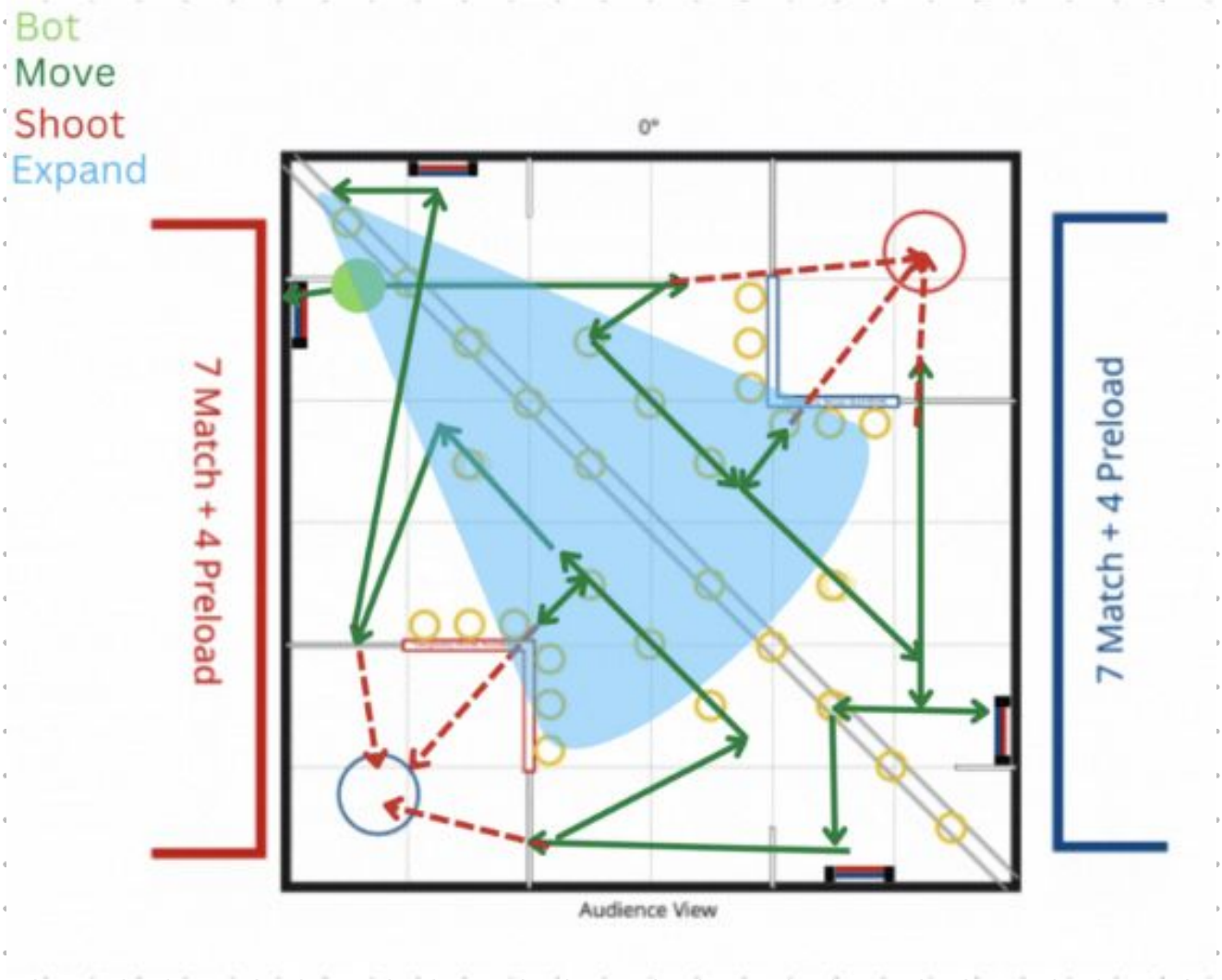
-We also added in a program to change the direction of the Robot so that it would move in the direction of the wedge which would make it easier on the driver during the competition.



Project Final Controller layout

Name David G Date 2/24 Page 126

Auton Skills Route - States



Working On Skills Auton

-Plan for skills Auton: get 5 rounds of disks (15 disks, 3 rounds of 5 disks), shoot each individual round, get all four rollers, shoot endgame from corner of field into all the tiles of the field.

-First point of action, making a skeleton program with all the separate parts of it, non exact distances, turns and commented out intake and shooter programs.

-Run that program and start to adjust the distances and turn angles inside of it as well as separate wait times and intake lengths.

Starting States Auton

We have made considerable progress so far. From our skeleton program, we created a consistent 2 rollers and 3 rounds of disks program just by adjusting angles and distances. The first shot was always consistent however we had to optimize the wait times between shots to allow our slingshot to reload. While programming, we have run into some intake problems where the disks do not fall into the slingshot as we would like.

We still need to program 2 more rounds, 2 rollers, and endgame; however, we believe that refining the robot to eliminate final issues is a more important allocation of our time. We will continue to program as we are at states to ensure a quality skills run.

Changing Default Starting Position for Transmission

- Due to the inherent nature of a transmission, switching gears while in Auton would lead to lots of inconsistency when driving in auton mode (this is because of meshing when switching gears, sometimes the meshing is better and it happens quickly while sometimes it doesn't)
    - Therefore the default mode for the pneumatics will have to be the default mode of the root (fast or torque)
        - Currently the default position is torque, which limits our ability to run auton quickly and effectively since the output rpm is 135 (extremely slow)
            - **Therefore will we replace the port of the solenoids so that the speed mode will be the default**
- To do this we removed the solenoids and reinstalled the tubing entering the pistons in the opposite ports
    - While this ran smoothly at first on of the fittings on the pneumatics got extremely loose causing a leak
        - To fix this elak we had to remove the solenoids again and screw in the fitting to make sure it was secure
            - We did this but the fitting got loose again so we decided to replace it
                - **This was a mistake since it took us a long time since its positioning was extremely awkward and cramped**
                    - Despite this after many attempts we were able to put it in and it worked

Team Meeting:
Week In Review Takeaways

This week was spent working on match and skills driving and auton. We made several optimizations on transmission, intake, and slingshot to be more effective in match while also repairing several components. Though, we had some issues we were able to solve a majority of them and are looking to continue to address them before states.

Next week, our goal is to do the bulk of the skills autonomous, prepare for states, and continue to optimize each aspect of the robot.

Time in Lab:
Monday - Sunday; 20 Hours

Intake inconsistency and repairs

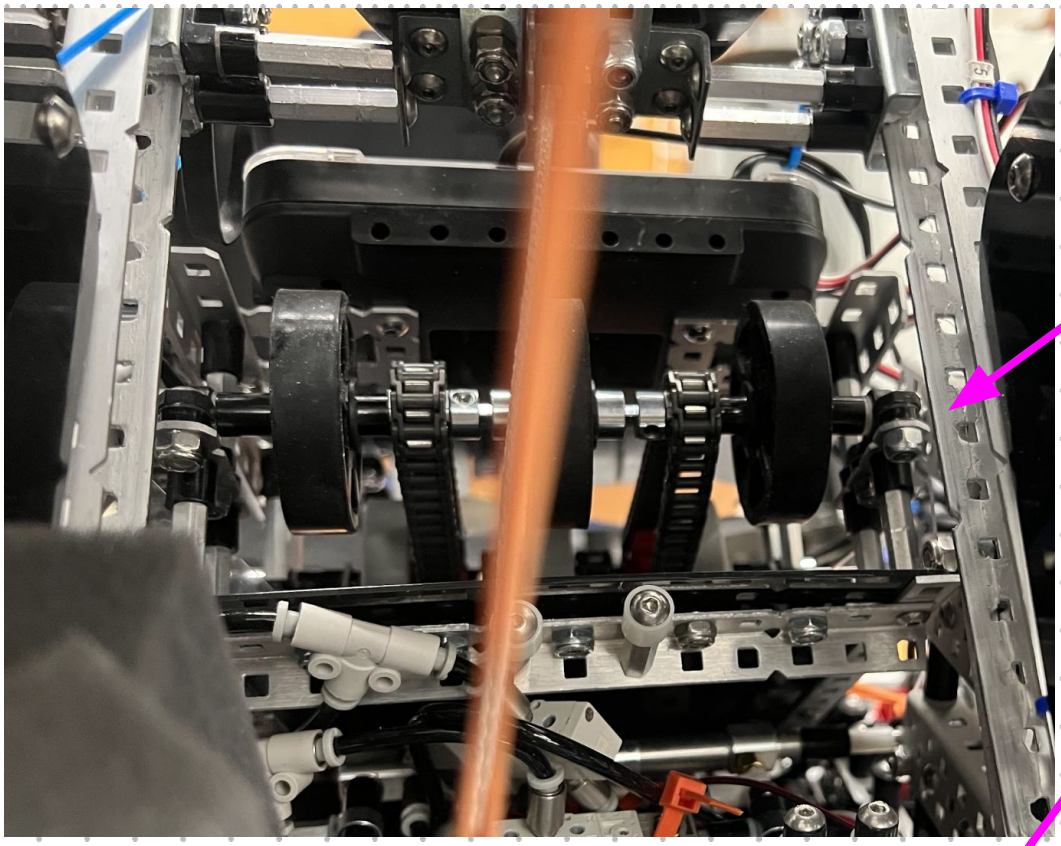
- Once we fixed the transmission yesterday, **we decided to test the robot in a competition format**
    - While driving worked, and we could shoot discs, the majority of discs were still not traveling far enough into the sled and were getting stuck between it and the intake and the sled
        - **Therefore we thought it was imperative to make the intake more consistent so that we could continue working on the auton and improve performance**
            - The first blatant issue we noticed was that part of the intake floor plate was broken
                - To replace it we **were forced to completely disassemble the intake** due to the fact that it is mounted on top of the floor plate
                - Once we completely rebuilt it we began to work on the first challenge with our intake → **Initial pick up of disc**
- Additionally, the intake seemed to struggle to transfer discs from the floor of the field and into the intake itself
    - **To fix this issue we brainstormed several solutions**
        - The first was applying downwards pressure on the freemoving part of the intake (front wheels are on a hinged assembly to adapt to the different sizes of discs)
            - We tried many different tensions by attaching rubber bands to all the convenient spots yet the tenson still was too much
                - **Then we noticed we could loop a rubber band around one of the transmission supports → this worked well**
        - The second solution was creating a slight curve on the base of the intake by mounting screws on the bottom support of the intake plate → this woulds stop the intake from rubbing on the ground and would make intake easier
            - While at first the curve made the gap between the intake floor and filed too big too intake we were able to adjust it to be perfect → **now initial intake was near perfect, now all that was left was top make the placement of discs into the sled more consistent**

Intake inconsistency and repairs

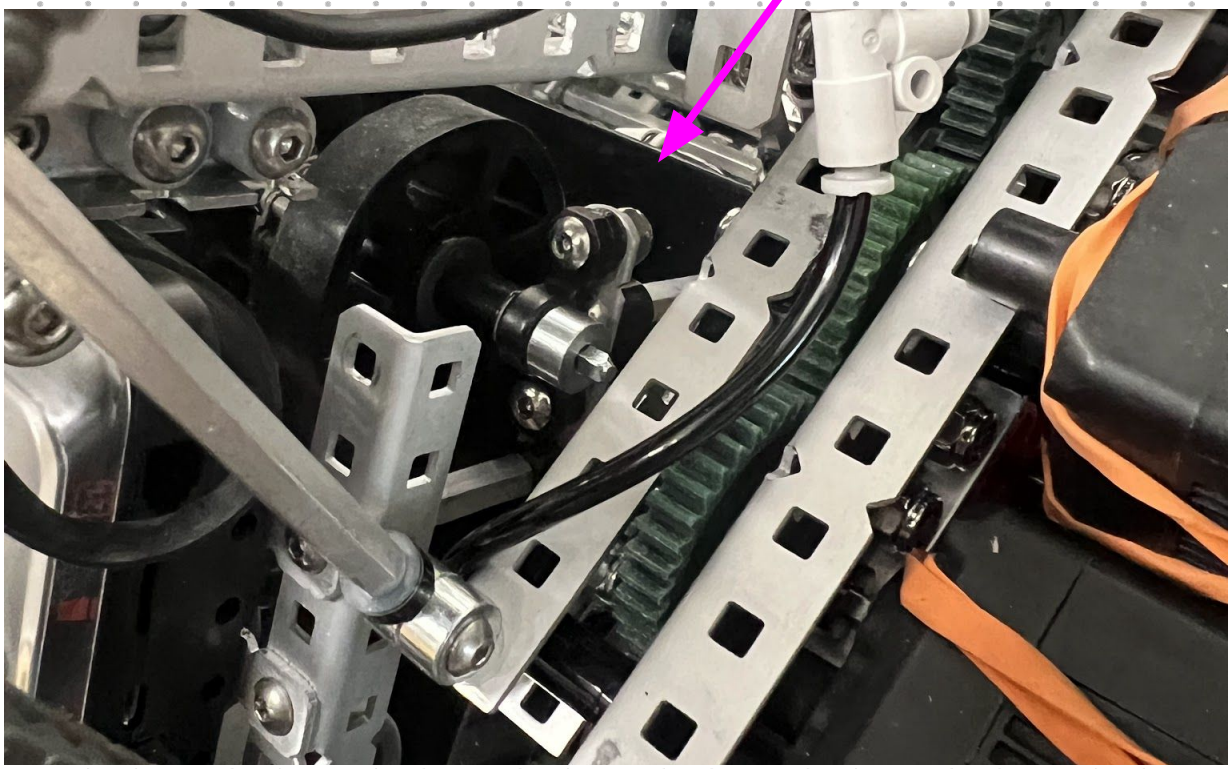
- To troubleshoot this we decided to remove the brain and the intake cover to get a better view
  - **The first things we did was attempt to pinpoint the issue**
    - To do this we put the entire intake in frame (without cover) and recorded a slow motion video of multiple successful and unsuccessful intakes
      - When observing the videos, we noticed that when the disc didn't go far enough it seemed to gain enough speed but "bobble" on the c channel rails of the slingshot
- **Therefore we reflected on how to prevent the bobbling**
  - There were 2 main ideas 1, sand down the rails, and 2 "center" the discs near perfectly
    - Option 1 did run the risk of causing issue with slingshot pull back which would then require us to remove the c channel rail (and therefore would cause further delays at a time when time is a priority) so **we decided to proceed with 2 out of caution**
- **While we attempted various different centering methods (**zip tie funnel, plastic strips acting as a funnel) in dozens of different configurations the problem improved only slightly, **still containing inconsistency** which would be fatal for skills auton
  - **We will have to go back to the drawing board and try again**

Intake Modifications

- Continuing yesterday's work we **went back to the drawing board**
  - To retest and modify the intake and re verify the intake "bobbling" problem we removed the intake cover and brain again
    - **The first thing we did was to retake the slow motion video**
      - **This corroborated what we saw yesterday, the discs were "bobbling"**
        - Therefore we decided to double down on the centering solution
          - We tried so many different combinations of pressures and widths for the funnel created by the plastic guides we mounted on the sides **yet nothing seemed to work**
            - Simply we were stuck, we knew that we had to improve centering yet we had no clue what we could do (we had tried every pressure and size that came to mind, everything either was too little or too much [would center yet also slow down the discs]
              - **Then a member of another team at our school suggested we move the the last set of wheels up**
- While at first we were hesitant since we thought that we would have to redesign the intake floor plate to install more holes, we realised that we could mount a piece of aluminum bar to offset the pillow blocks the axle holding the last set of wheels was mounted to.
  - Once we installed the offset set of wheels close to the sled, we decided to test it out
    - **While some discs made it through some discs were not making enough contact with the last set of wheels and were getting stuck in the intake** → therefore we adjusted the height of the wheel various times until it made enough contact to send the discs through while also not burning out the motors
      - **Now the intake works consistently**



Off Set Pillow Blocks



Starting the Endgame

- Due to issues with the main parts and various delays we were unable to start the endgame launchers until today
  - The design we picked has been dubbed (at least at our school) as the “potomac standard”, simply every team at Potomac (our school) has eventually switched to it due to its simplicity and effectiveness
    - The basic design is a angle pieced (rigid yet light) with 2 standoffs on each end
      - 1 standoff (the “barrel” side) has a ramp on it to guide the weight over it rather than hitting it
        - The reason this standoff is there is to tension rubber bands the weights are attached to so the string can be launched
      - The other standoff is designed to hold the weight
      - A string is then wound around these standoffs and the rubber band and weights (which are attached to the string) are stretched across
        - **To fire 2 pistons will lift up a bar and release the rubber bands from the standoffs launching the weight and therefore string**
- The clearest place to mount the endgame was on top of the roller mechanism support bar which helps hold the c channels holding the roller mech and allows us to mount the launchers perpendicular to the slingshot
  - **While this worked we noticed that the angle (45%) of the shooter, which the endgame was perpendicular to, made dqing more likely so we decided to mount it with standoffs so it could be flat**

Working On Skills Auton Pt 2

```
void auto_skills(){
  solenoid.set(1);
      solenoid1.set(1);
  pullback();
  solenoid2.set(0);
      solenoid3.set(0);
      wait(1, seconds);
  Drivetrain.driveFor(reverse, 10, inches, 100, velocityUnits::pct);
  Intake.spinFor(.35, seconds, 100, velocityUnits::pct);
  Drivetrain.driveFor(fwd, 20, inches, 100, velocityUnits::pct);
  wait(1,seconds);
  Drivetrain.turn(left, 80, velocityUnits::pct);
  imu.resetRotation();
   waitUntil((fabs(imu.rotation())>=35.00));
   Drivetrain.stop();
   Intake.spin(fwd,100,pct);
 Drivetrain.driveFor(fwd, 149, inches, 100, velocityUnits::pct);

   wait(1,seconds);
   Intake.stop(coast);
   Drivetrain.turn(right, 80, velocityUnits::pct);
   imu.resetRotation();
```

Working On Skills Auton Pt 2

-Though the photo does not fully encapsulate our full Auton it gets to the key pieces of it which are using the imu sensor to detect the length and the rotation degree of the robot

-We can then use that to get exact measurements and more consistent autons as the robot will move to the same position everytime if the IMU is calibrated correctly (IMU is the inertial sensor on our robot)

-For our turns we reset the IMU and then turn so we get the same degree value everytime

-We also use the wait until to stop everything else in the auton until the turn is completed

-We use the slow mode for our auton by setting the solenoids to it at the start as well as pulling back the slingshot so it can intake disks. It also unclamps the clamp.

-We decided to put it in slow mode because it provides for more consistent autos as its movements are more exact rather than have it be in full speed which comes at the expense of time

-However this time was made up with the careful choosing of how our Auton would move across the field.

State of the ~~Union~~ Robot

Currently, the Robot is now complete. The drive base has been reliable, the transmission has become more consistent as we tweaked it, our slingshot can pull back and clamp consistently, and now our intake is working well and consistently. We truly have an incredible and well functioning robot for states. However, due to the various delays, we have not had the opportunity to work on the skills auton as much as we have liked, therefore it will be imperative that we use the time before and during the tournament to get a good autonomous skills score to help use skills as a back. Therefore, much of the pressure and our main ~~new~~ hope for qualifying for worlds will now rest on Arav's (our driver) shoulders. Additionally we believe that the incredible innovation and creativity of our robot sets us up to be in a good position to possibly win a judged award and qualify for worlds that way.

Pre States Meeting

In our pre states we discussed our goals and plans for states. Since various members of the team have attended worlds in previous years we are coming into the tournament from a competitive perspective. But, this isn't stopping us from still approaching the tournament as a learning and educational experience. We would really like to go to worlds, but we understand that not all teams will get that opportunity, **yet we will still try and persevere.** At states our priority is to come into the tournament with a working winpoint auton, which will be beneficial for our qualification ranking. We would like to get our driver skills done on friday, and once we are done with our autonomous, we will do out autonomous runs for skills on Saturday morning. While it might be difficult to secure our spots through skills, due to the fact that we started our program quite late, we still believe that it is very much possible we make it to the semi finals, and or possibly win a judged award that will qualify us for worlds. **Even if we don't we were all in agreement that this season has taught us an incredible amount about the engineering design process and robotics in general.** We learned how to create new mechanism, new coding techniques, and how to be good engineers that use the engineering design method and document their work.