



ROVER RUCKUS

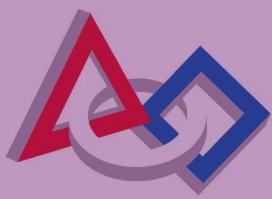


Presented By Qualcomm



Team 7341
F.R.E.N.C.H. F.R.I.E.S.

Engineering Notebook



FTC

FIRST® Tech Challenge

FRIENDLY
RELIABLE
ENERGETIC
NERDY
CLASSY
HEROES

PROFESSIONAL
GRACIOUS
ENGINEERING
TEAMWORK
FIRST

Signature : Shelby Greer

Date: Jan. 27, 2019

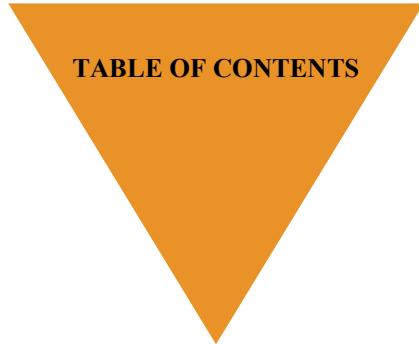
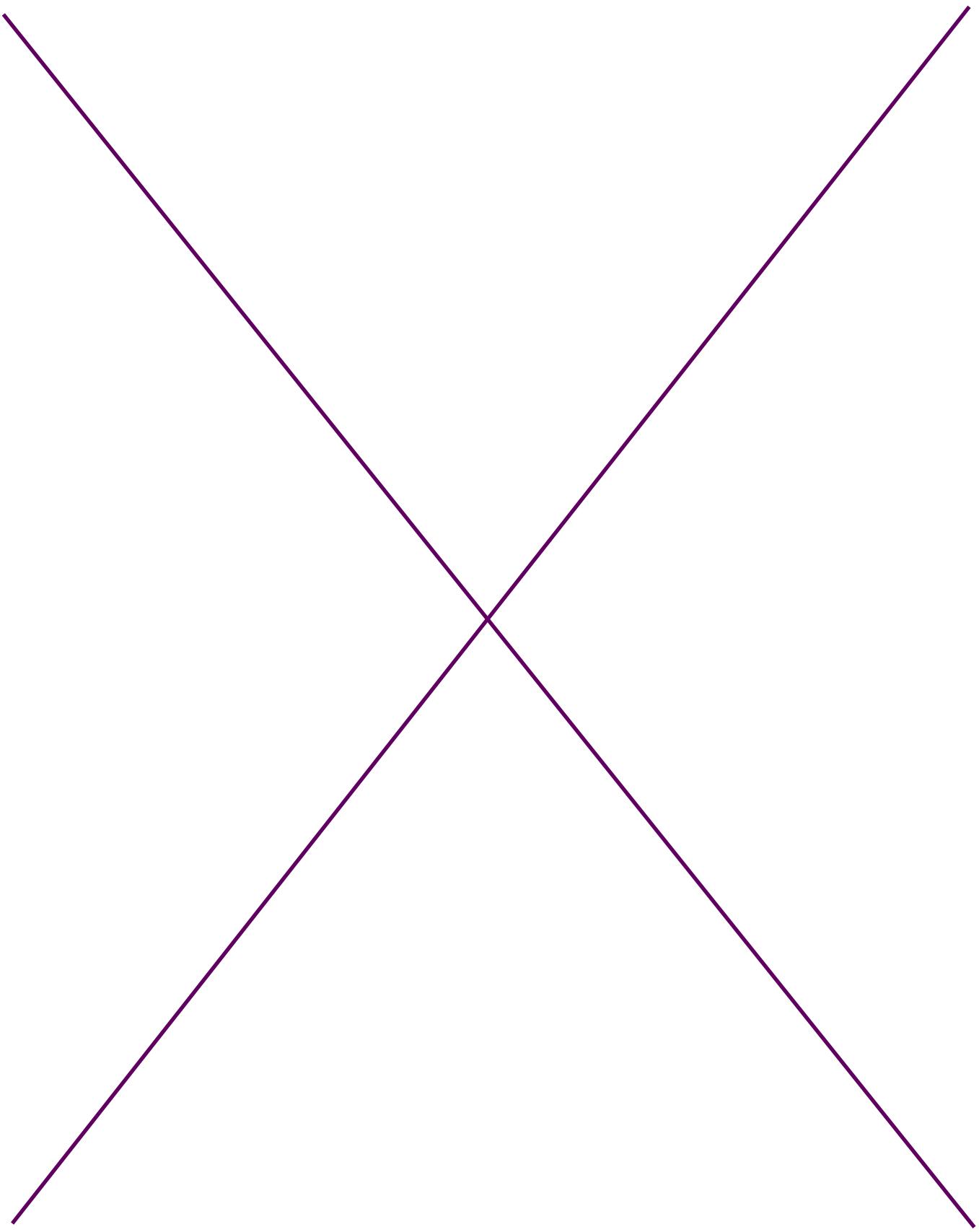


TABLE OF CONTENTS

Team Summary	5
Our Story	7
Our Team	13
Our Outreach Journey	19
Our Engineering Journey	51
Our Business Plan	441

Signature : Shelby Greer

Date: Jan. 27, 2019



Signature : Shelby Greer

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Team Summary

We are Team 7341 - F.R.E.N.C.H. F.R.I.E.S. (Friendly, Reliable, Energetic, Nerdy, Classy, Heros, proFessional, gRacious, engineering, tEamwork, FIRST). This is our six year participating in the FTC program and we have 2 veteran team members this year. We are excited and scared about this year's challenge and robot design as there are many new things to consider.

Our journey begins with a budget and the search for grants. We have been very lucky that NASA, DoD STEM, FPL, Rockwell Collins and Lockheed has provided us with funds and Girl Scouts allow us to use the FTC parts to support the team. This year we also received special grant from the sister's of a NASA Employee who work with robotics. Thank you to all our supporters (**Tab 1**). You can review our Business plan for our income and spending plan (**Tab 8**).

One of our ingenious designs is a phone box for the robot. The box help us to not accidentally push the button that will back you out of the program running (**Tab 3**). In designing the robot we thought that if we could have something help the robot while driving over the crater it would lessen the chance of tipping over. We created two small wheelie bars and added them to the robot. We think that our home made linear slide works well and is a combination element using both a TorqueNADO and servos (**Tab 5**).

While planning the autonomous program we found that the phone could not pick up all the minerals at the same time so we developed code to have the phone move to scan for the gold mineral (**Tab 7**). We were able to create a motorized element fetcher, but went back to a simpler end effector because the motor needed was too heavy.

We have created quite a few test programs to help us learn about how the sensors, motors and camera work and react to the robot's environment. The data we learn from our test program are transferred to the operational TeleOp and Autonomous programs (**Tab 6**).

We have set up our robot with the following motors, attachment and sensors:

Seven (7) drive motors, where the 4 Drive Motors have encoders for running autonomous, 2 to raise and lower the capture attachment and 1 to extend and retract the center lift mechanism.

Six (6) servos: 1 to Dump the marker, 1 to turn the phone and 2 for the second level of the lift mechanism and 2 for the distance arms.

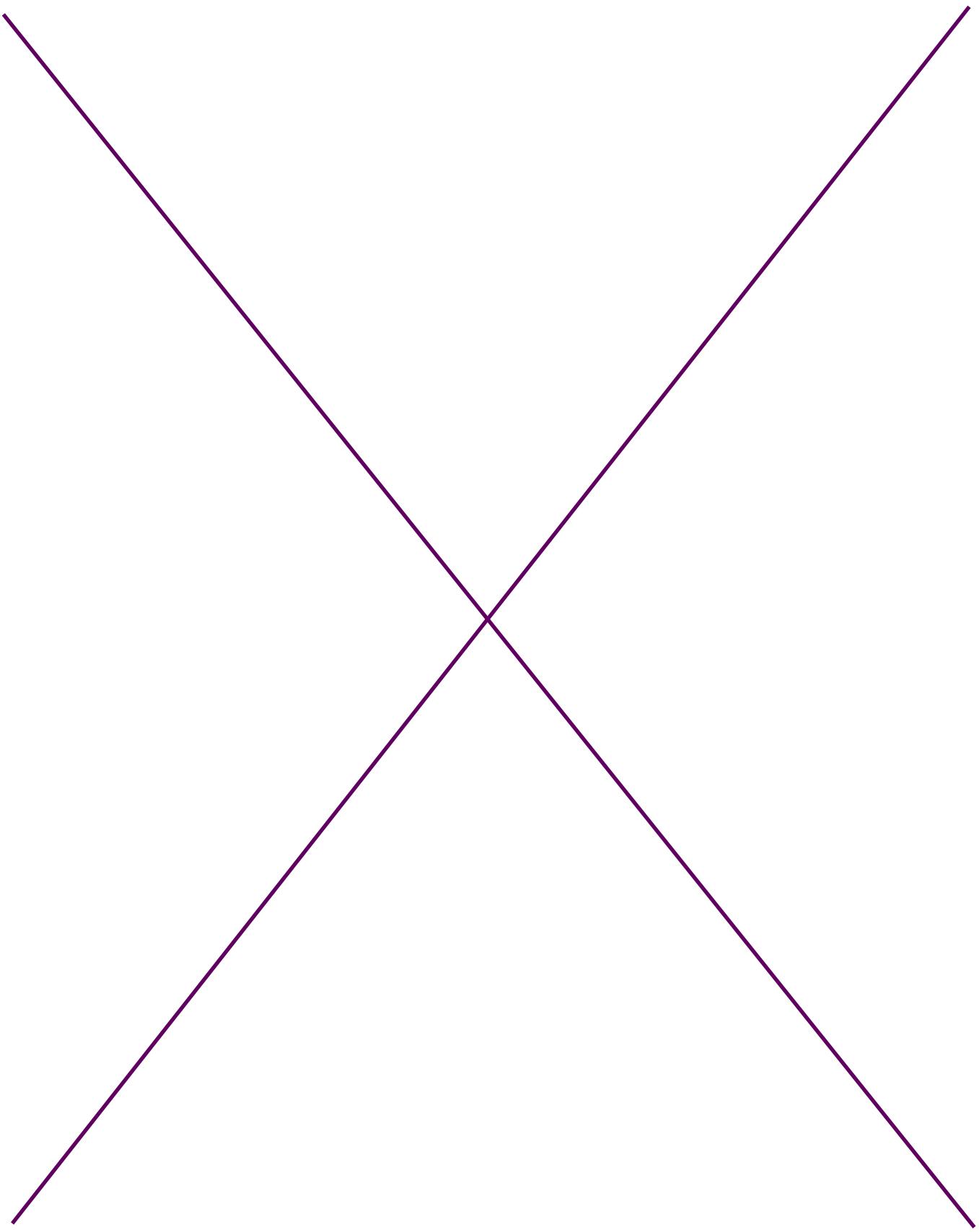
Six (6) touch sensor which are used to determine the upper and lower limits of our attachments and used to stop the motors from moving when the limit has been reached.

Outreach is one of the activities that we enjoy as we travel between Brevard and Orange Counties supporting the different Girl Scout and community event. We would like to highlight our work for our Veterans (**Tab 2**) and co-sponsoring the Space Jam Meet on Dec 15th (**Tab 4**).

Thank you for reviewing our Engineering Notebook!

Signature : Shelby Greer

Date: Jan. 27, 2019



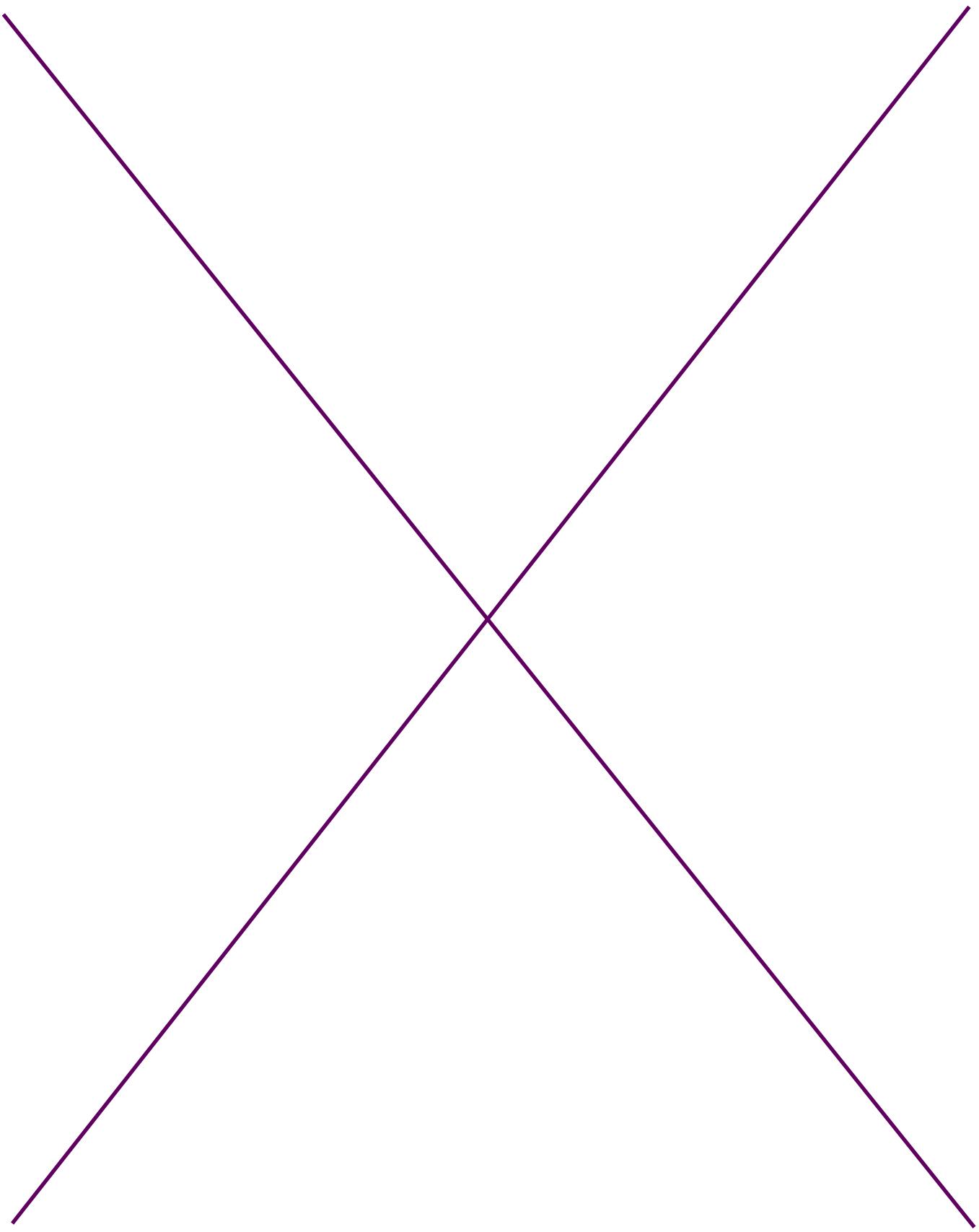
Signature : Shelby Greer

Date: Jan. 27, 2019



Signature : Shelby Greer

Date: Jan. 27, 2019



Signature : Shelby Greer

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Story

Introduction

We are Team 7341 - F.R.E.N.C.H. F.R.I.E.S. (Friendly, Reliable, Energetic, Nerdy, Classy, Heros, professional, gracious, engineering, teamwork, FIRST) and this is our sixth year participating in the FTC program. We were known as the "Minty Matrix" team in FLL and our last project was featured in the "Inventors" magazine (November/December 2013).

We are a Girl Scout team and our 2 members attend the same schools (Astronaut High School). As Girls Scouts we believe in sharing our knowledge with others and making the world a better place. We are asking our fellow Space Coast Teams to join us in this outreach activity.

Our team is BIG on outreach over the years we have traveled to the Orlando Science Center, Orlando Convention Center and KSC Visitors Complex for community events, attended FLL and FRC Tournaments sharing our knowledge of FIRST and our robot. Our outreach activities are year round reaching out to those who are wanting to learn more about FIRST.

This year our coach decided that her teams (FLL and FTC) would sponsor a FTC Tournament and we joined up with the Plaid Pelican and Plaid Piranha's planning the Theme, special activities and the location. We have involved our Sister FLL Team as many of them will be

Signature : Shelby Greer

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Story “Continued”

joining the team next year. We enjoyed coming up with the theme of “Space Jam” to get everyone in the spirit of giving. As part of the tournament we are asking teams donate non-perishable food items for St. Teresa’s Food Pantry. We came up with a few outreach activities that the team can join in during the day. Our coach is going to contact the local FLL teams to invite them to watch the tournament.

After all the pre-planning of the tournament was completed it was now time to get to work on the robot. It was a challenge for all to learn the new communications software and hardware. After watching the Challenge Video the gears in our brains started to turn finding that we need to get more communications and motor hardware when the “Grant” money came in. Not letting that issue get in the way we talked and designed a robot that would do everything and build a robot that we had the parts for.... So more changes will come as the tournament progresses.

We are lucky that our team has two coaches and two engineering mentors that help us by teaching us things from using equipment (saws, planers, drills and more), engineering concepts (math, fractions, angles and physics) , software (Android Studio (JAVA), Adobe Premiere (Video Editing) to speaking in front of a group of people. We have learned so much over the past few months.

Signature : Shelby Greer

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Story “Continued”

We would like to thank all our sponsors, because if they were not there we would not have been able to build the robot we have today. So, a BIG Thank-you goes out to them as well as the sponsors for the FIRST program: Qualcomm, Rockwell Collins, Virtual Reality and PTC.



Signature : Shelby Greer

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Story “Continued”

The following pages will describe our team and the journey we have taken with our robot “Princess Charlie”. One of the things we work at is showing others all about FIRST Core Values as these values are similar in words and the Promise and Law of Girl Scouts. We love the FIRST® philosophies of Gracious Professionalism® and Coopertition®. We use the Core Values:

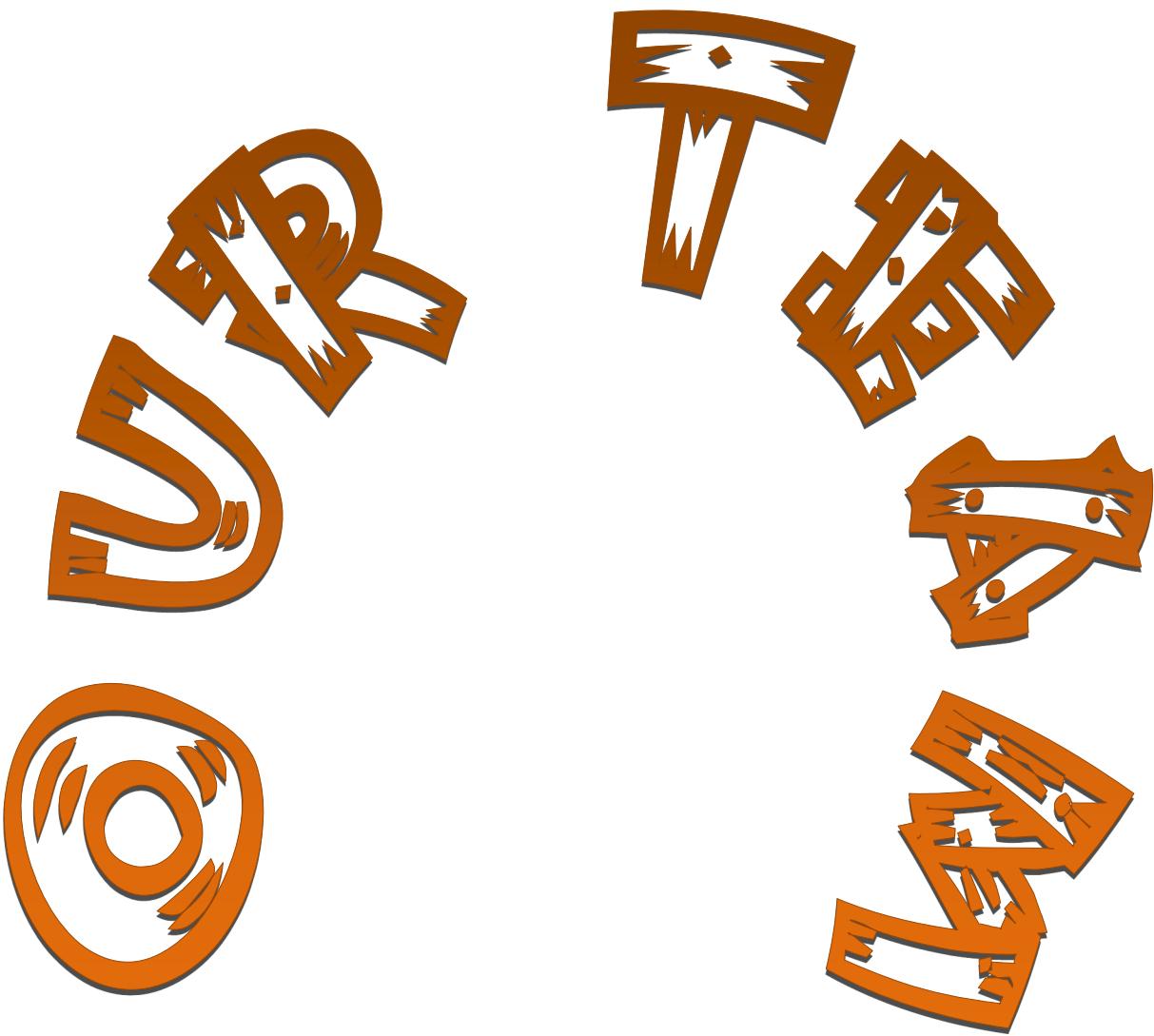
- ⇒ Discovery—is when we explore new skills and ideas.
- ⇒ Innovation—is when we use our creativity and persistence to solve problems.
- ⇒ Impact—is when we apply what we learn/know to improve our world.
- ⇒ Inclusion—is when we respect each other and embrace our differences.
- ⇒ Teamwork—is that we are stronger when we work together.
- ⇒ Fun—is when we enjoy and celebrate what we do!

Gracious Professionalism® FIRST® uses this term to describe our programs’ intent and Gracious Professionalism® is not clearly defined for a reason. For us Gracious Professionalism means:

- ⇒ You help each other out
- ⇒ You show respect and listen to all
- ⇒ You are sensitive everyone’s thoughts and ideas
- ⇒ You are always willing to learn more

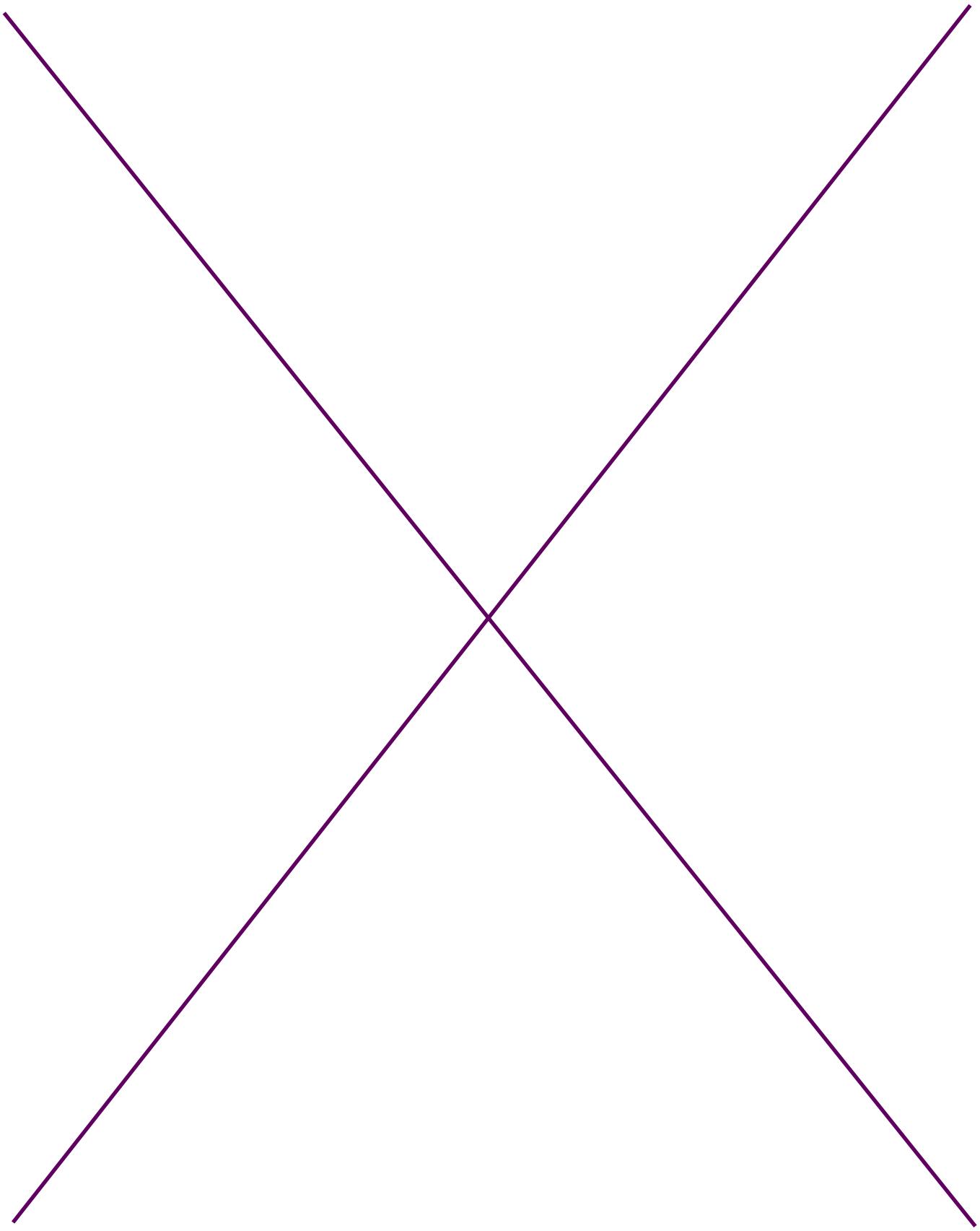
Signature : Shelby Greer

Date: Jan. 27, 2019



Signature : _____

Date: _____



Signature : _____

Date: _____

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Team Members

Jessica Anderson



Jessica Anderson is a senior at Astronaut High School. This is Jessica's 4th year in FTC Robotics and 2nd year on the FRC Team comBBat 21. Jessica is a member of Girl scout Troop 1703, Astronaut's band, joined Astronauts winter guard and was on the Astronauts Bowling Team. Jessica enjoys building things and contributes by building the attachment parts for the robot.

Shelby Greer



Shelby Greer is a sophomore at Astronaut High School. This is Shelby's 3rd year in FTC Robotics. Shelby is a member of a girl scouts troop 1703. Being in robotics is important because it teaches a lot about robotics and you can be a engineer when you get older if you want to or do something else enjoy building things and learn something new every day. Whenever I get to a robotics meeting.

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Team Members

Caroline Achee



Caroline Achee is a displaced KSC Shuttle worker who loves to build things and write software. Caroline has a passion of sharing what she knows with anyone who wants to learn. She has been a coach/mentor for the past 6 years and this year she is mentoring a Jr. FLL, FLL and FTC team. She also is the League's social media person. When Caroline is not playing with robots she builds Drupal websites, volunteers with her Girl Scout troop, volunteers at her church and sings in her church choir.

Chelsea Partridge



Chelsea Partridge is a Test Engineer with Lockheed Martin on the Orion Spacecraft. She is a Mentor for the FRENCH FRIES FTC. Chelsea graduated with her degree in Mechanical Engineering from the University of North Florida in May 2015. Volunteering and STEM outreach are two of Chelsea's biggest passions. Chelsea combines these passions by serving as President of the Missile, Space, and Range Pioneers, a social club that has been on the Space Coast since 1966.

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Team Members

Emily Worthington



I am a Sophomore in college. I am studying sign language at Valencia College. Back when I was in high school I was a F.I.R.S.T member of the F.R.E.N.C.H fries team. Last year with my schedule I wasn't able to help out as much as I wish I could've. But luckily this year I've moved closer to my home team and I have been able to assist them at the competitions reliving my high school experience in the joy that I had at that time I'm so grateful for this opportunity to continue with F.I.R.S.T. I look forward to the rest of the season and hopefully the seasons to come.

Jaylene Riddle



Jaylene Riddle works for the Navy keeping their Orlando Office building safe. She is a Coach/Mentor and loves all things mechanical . Jaylene continues the "Couch to Century" bike rides.

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

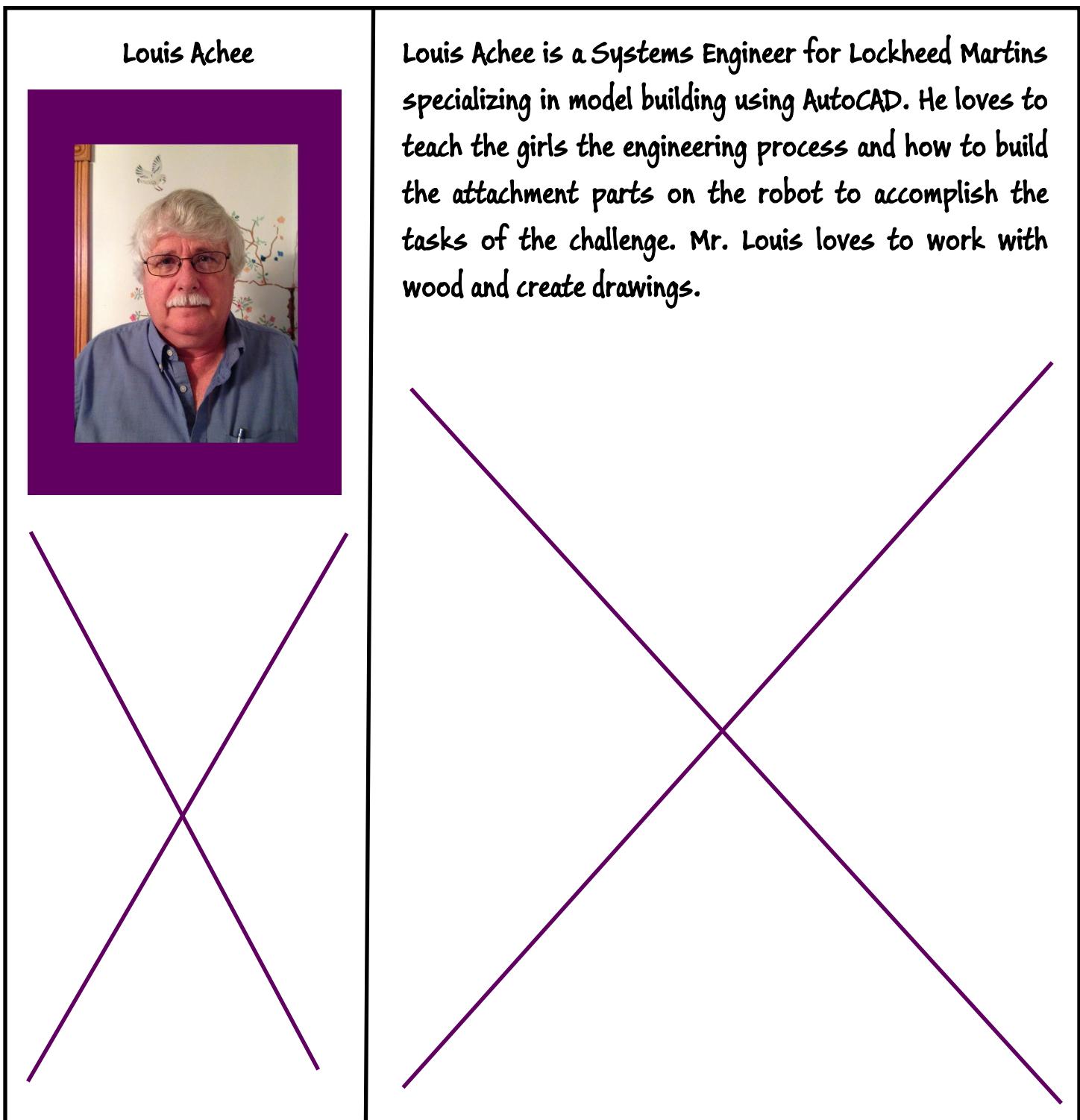
F.R.E.N.C.H. F.R.I.E.S.

Team Members

Louis Achee



Louis Achee is a Systems Engineer for Lockheed Martins specializing in model building using AutoCAD. He loves to teach the girls the engineering process and how to build the attachment parts on the robot to accomplish the tasks of the challenge. Mr. Louis loves to work with wood and create drawings.



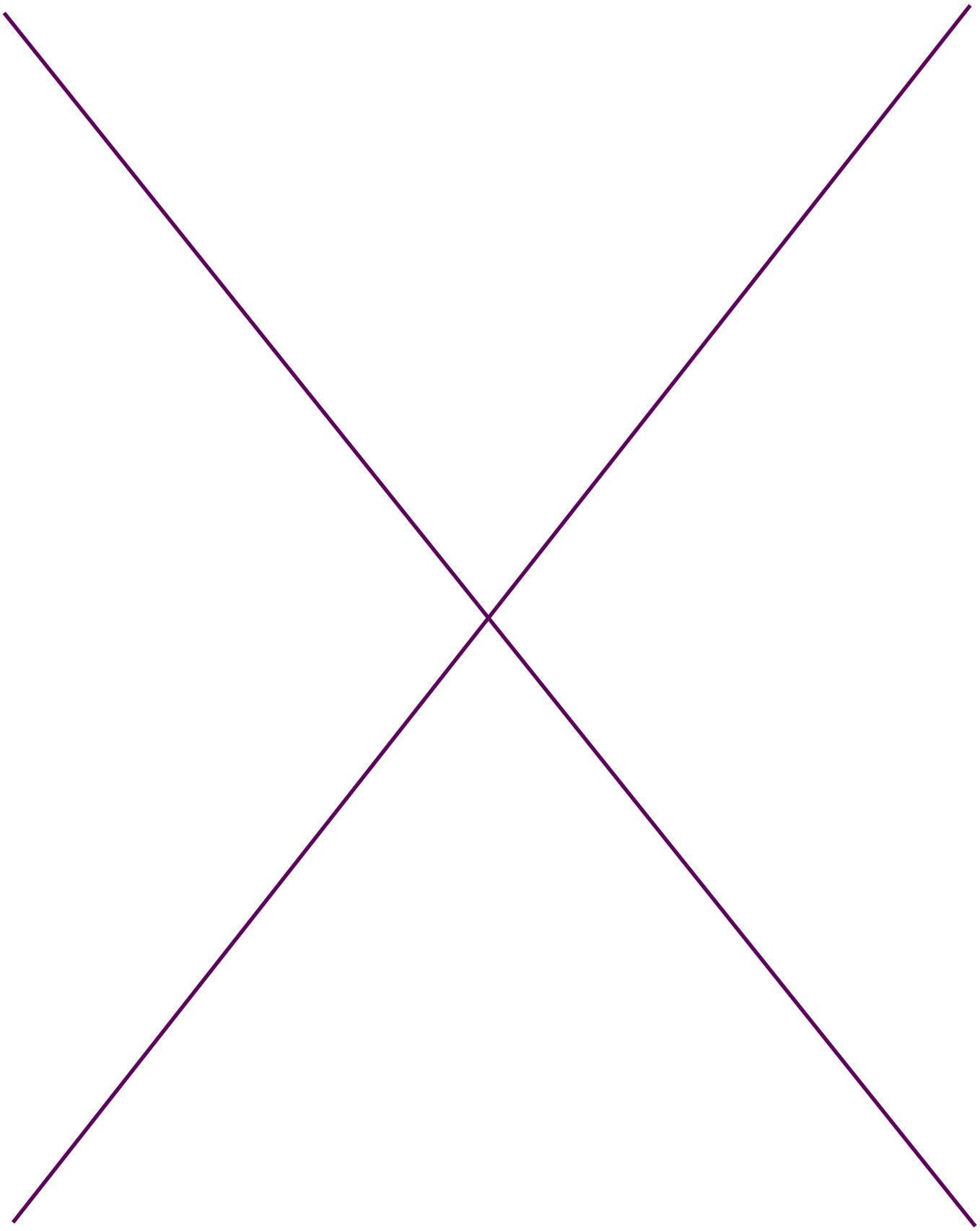
Signature : Jessica Anderson

Date: Jan. 27, 2019

**OUR OUTRAGE
TURNS TO HOPE**

Signature : _____

Date: _____



Signature : _____

Date: _____

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: September 2018

Purpose of the Activity:

We thank our coach for traveling to Melbourne several times to help the Prep School get their team going. The first trip they brought both Prince Charles and Princess Charlie down to their school so that they could drive the different robots since they had not build a robot yet.

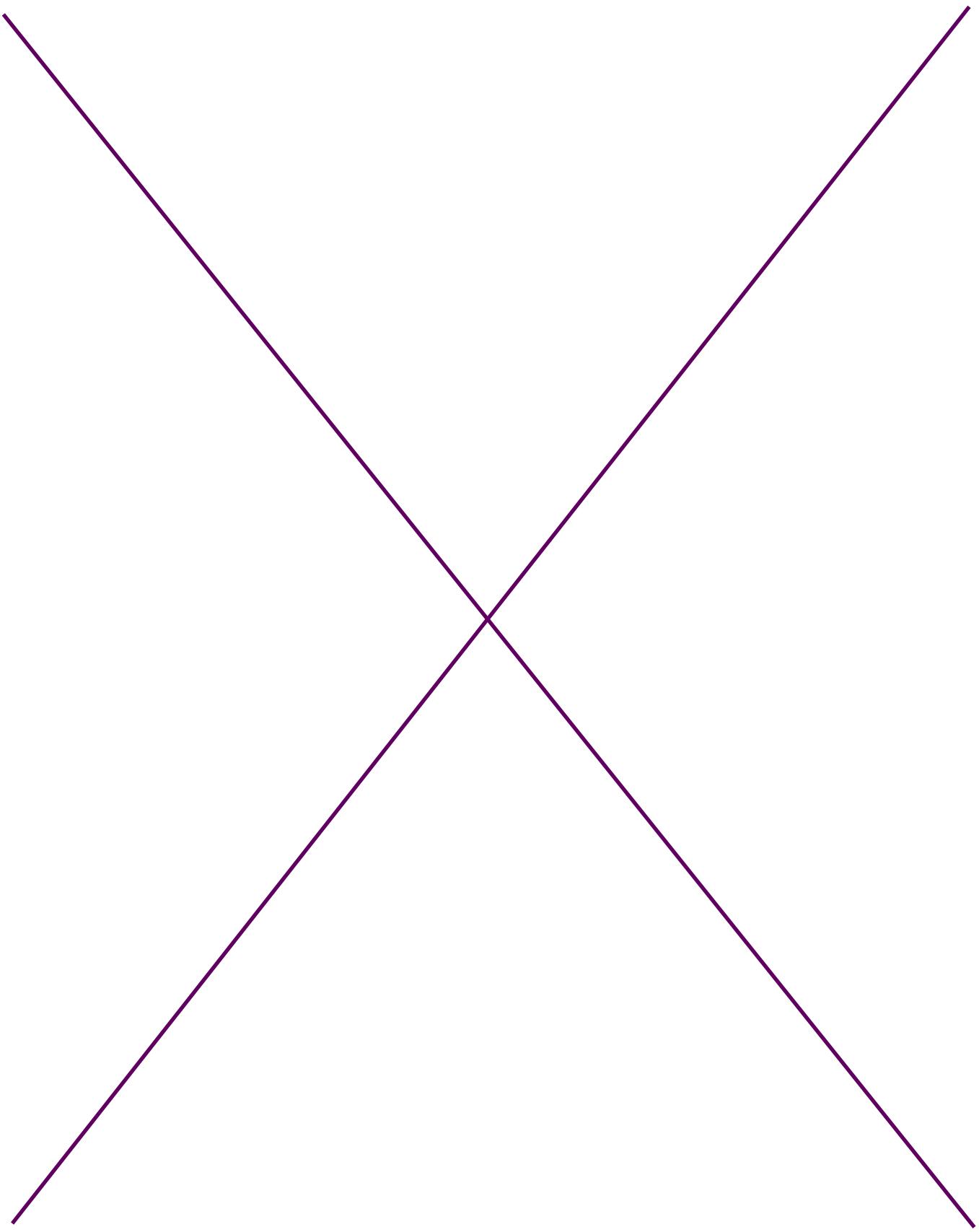
Images:



The students were able to test drive both Prince Charles our older robot and our new robot Prince Charlie. We went down once before the reveal and they did not have their kit yet and once after they received they kit and we talked about robot design and the communications connection

Signature : Jessica Anderson

Date: Jan 26, 2019



Signature : Jessica Anderson

Date: Jan 26, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: September 2018

Purpose of the Activity:

We thank our coach for helping out the Madison Middle School's team. She went to their school on many occasions to help them with robot programing as they started to use Android Studios. She helped them get the Teleop program going.

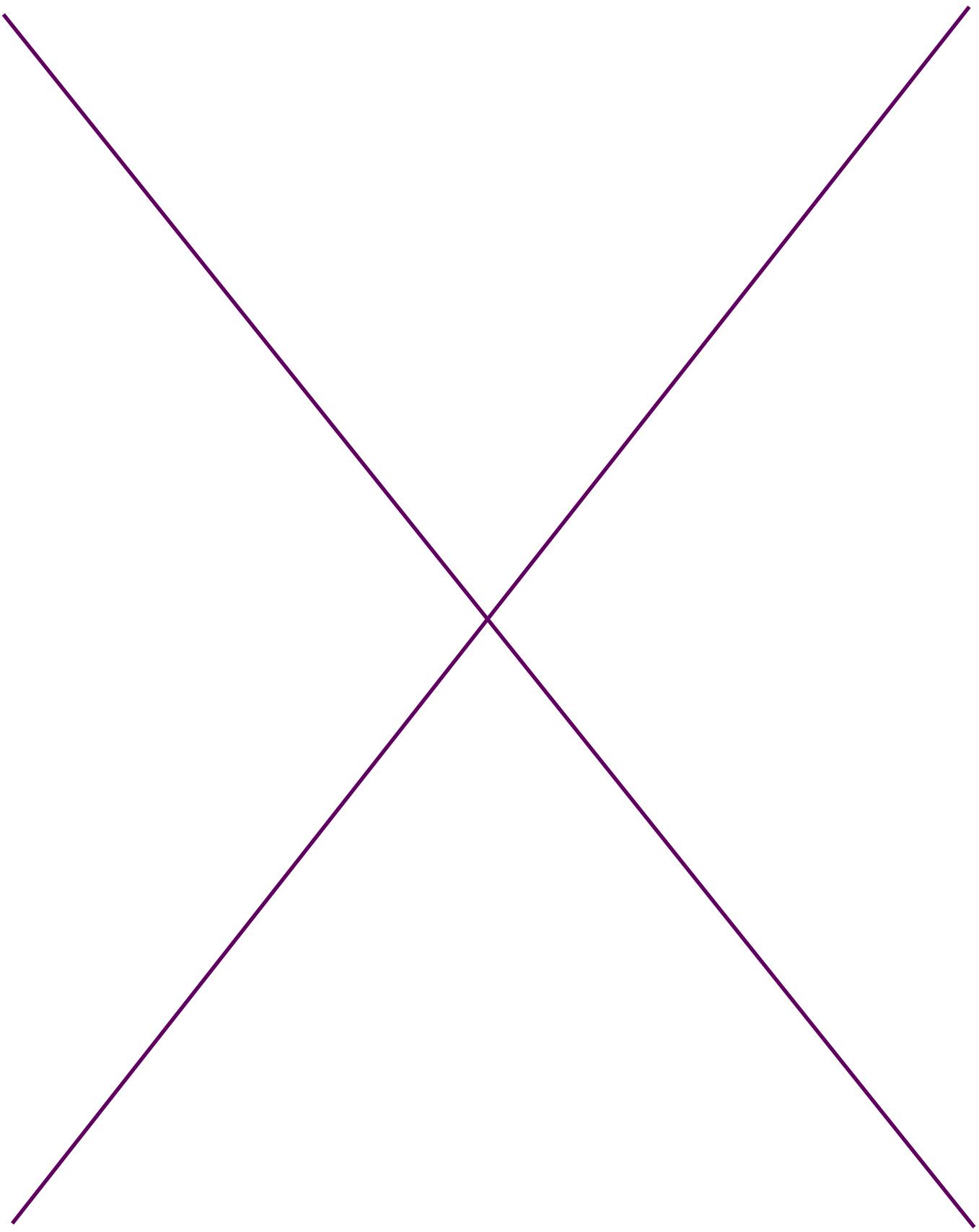
Images:



The coach brought a copy of our hardware configuration and Telop program for them to use to get started. After they had their robot build their were able to drive the robot around. The next week she brought a crater wall and they found that they could cross over the wall with no problem. Over then go to their meetings from time to time and help them when they had an issue that they could not solve.

Signature : Jessica Anderson

Date: Jan 26, 2019



Signature : Jessica Anderson

Date: Jan 26, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: September 8, 2018

Purpose of the Activity:

Field Trip day with our Sister FIRST LEGO League Jr and FIRST LEGO League Teams. Coach had a sleep over for all the girls attending the event. Breakfast in the morning was a little crazy.....

The day was filled with learning new things and having fun!!!!

Images:



Selfies with Barry Bonzac!!
Mr. FIRST himself



Eating breakfast before heading out to the KSC Visitors Center for a morning of learning and talking to experts about space.

The FIRST Tech Challenge team learn what their challenge was going to be for the season.

We enjoyed working together.

Signature : Jessica Anderson

Date: Sept 8, 2019

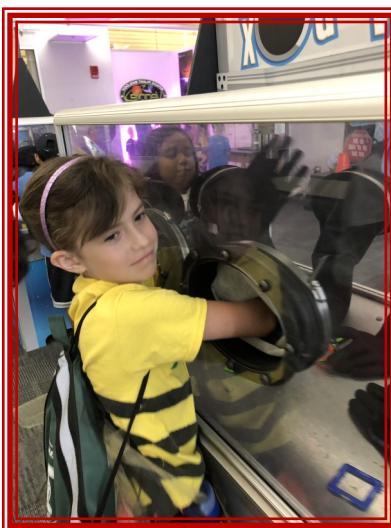
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity Continued

Date: September 8, 2018

Images:



We all had a great day at KSC Visitor Center, We are ready for this year's mission. Best of luck to all teams participating!!!

Signature : Jessica Anderson

Date: Sept 8, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

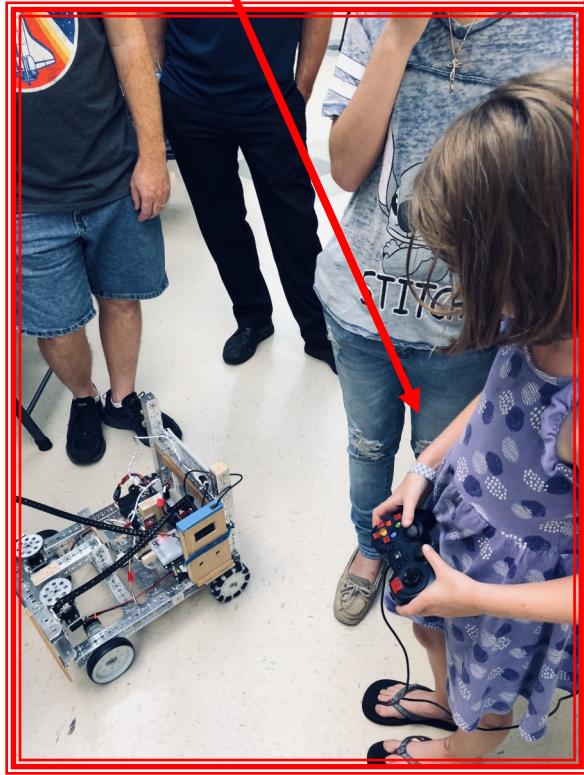
Date: September 8, 2019

Purpose of the Activity:

We participated in a GSDO NASA Robotics event at the First Baptist Church in Merritt Island. We talked to the NASA Engineers and FIRST FLL Program Managers about our robot and one of the NASA Engineers drove our robot. We also talked with other FTC teams attending the event about their robot design sharing things that we have learned.

Images:

Other students trying their hand at driving our robot.....



NASA Engineer driving our robot....



Signature : Shelby Greer

Date: Sept 9, 2018

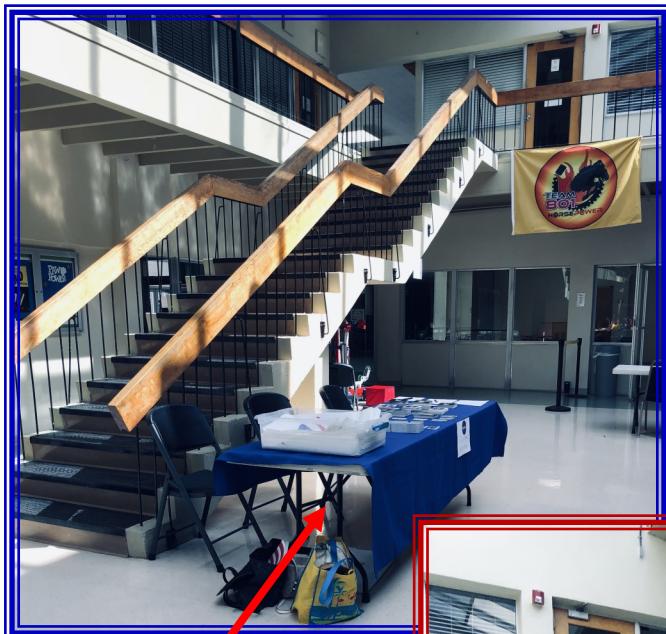
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

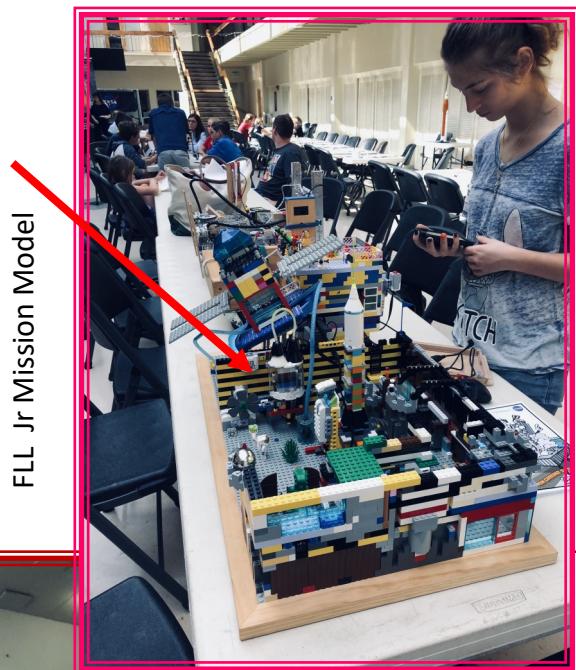
Outreach Activity Continued

Date: September 8, 2019

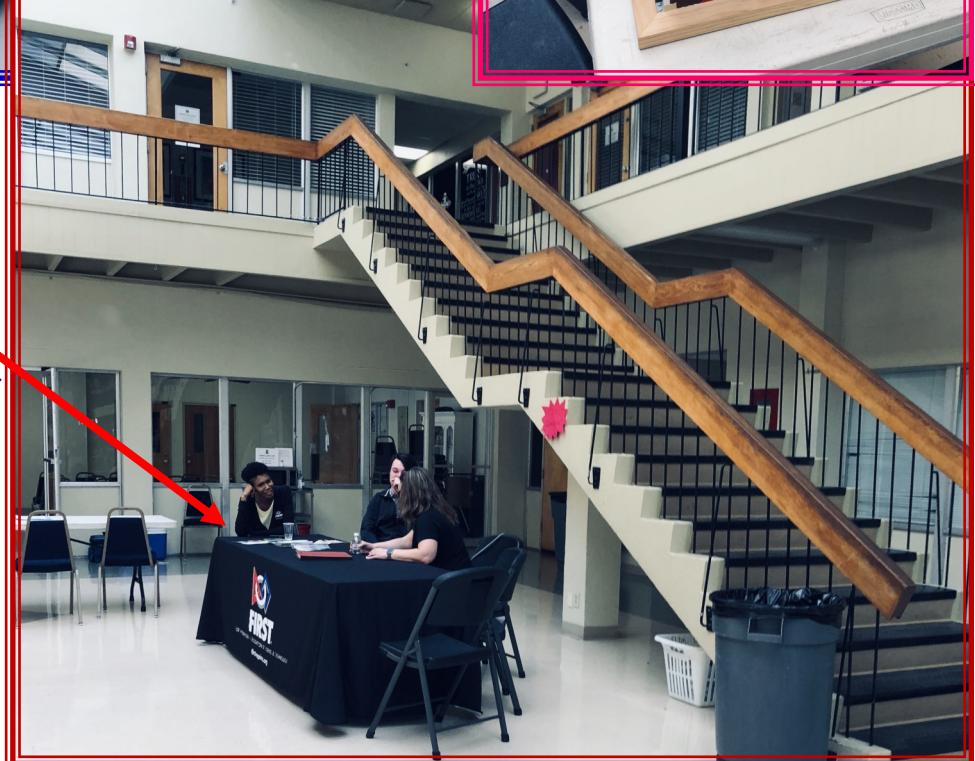
Images:



NASA Table



FLL Jr Mission Model



FIRST Table

Signature : Shelby Greer

Date: Sept 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: September 16, 2019

Purpose of the Activity:

Girl Scout Community Day where we shared all the things that you can do as a Girl Scout. We had several tables showing three aspects of FIRST. The FIRST LEGO League Jr. model was on display where the people interacted with the elements of Mission Moon. The FIRST LEGO League display board and robot was on display. The FIRST Tech Challenge Robot Prince Charles and the Engineering Notebook was available for review. We used this opportunity to recruit new members to all the teams.

Images:



Signature : Shelby Greer

Date: Sept 16, 2018

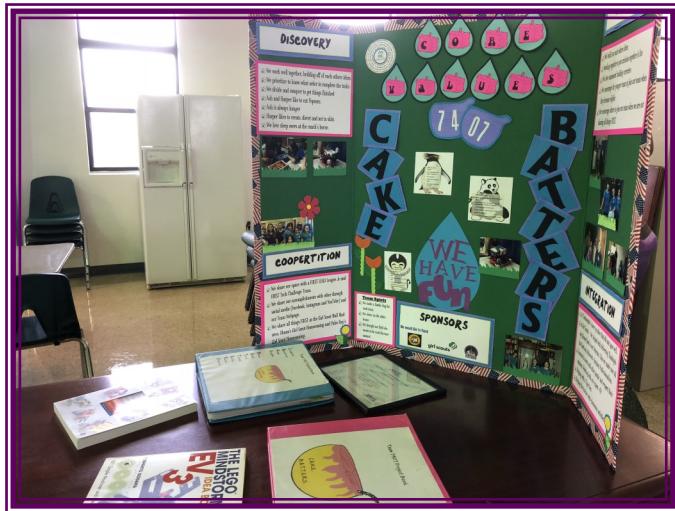
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity Continued

Date: September 16, 2019

Images:



Here along with other Girl Scout Troops, our group was showcasing the three FIRST Programs our coach coaches

The girls learned about our last year's project as we had not started on the 2018-2019 season yet.



Signature : Shelby Greer

Date: Sept 16, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: October 19, 2019

Purpose of the Activity:

Girl Scout STEM Night was all about the Science, Technology and Engineering. We had several tables showing three aspects of FIRST. The FIRST LEGO League Jr. model was on display where the people interacted with the elements of Mission Moon. The FIRST LEGO League display board and robot was on display. The FIRST Tech Challenge Robot Prince Charles and the Engineering Notebook was available for review. We used this opportunity to recruit new members to all the teams. We had a hands on build a rocket activity.

Images:



Prince Charles is ready to go and the FIRST LEGO Jr Model is also on display Ready set go.....



Signature : Jessica Anderson

Date: Oct 19, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity Continued

Date: October 19, 2019

Images:



The evening was filled with all types of different STEAM Activities for the girls to try out.

You never know where you will spark the interest.....



Working together and learning how to make straw rockets and how to drive Prince Charles.... Future FIRST Tech Challenge Team Members.

Signature : Jessica Anderson

Date: Oct 19, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: October 26, 2018

Purpose of the Activity:

Sharing all thing FIRST Tech Challenge at Portofino's while selling Girl Scout Fall Products. We were able to talk to the community and share information on our Robot Prince Charles and what the FIRST program is all about.

Images:

We were showcasing Prince Charles at our booths to let our community know all about FIRST Tech Challenge and the other programs of FIRST.

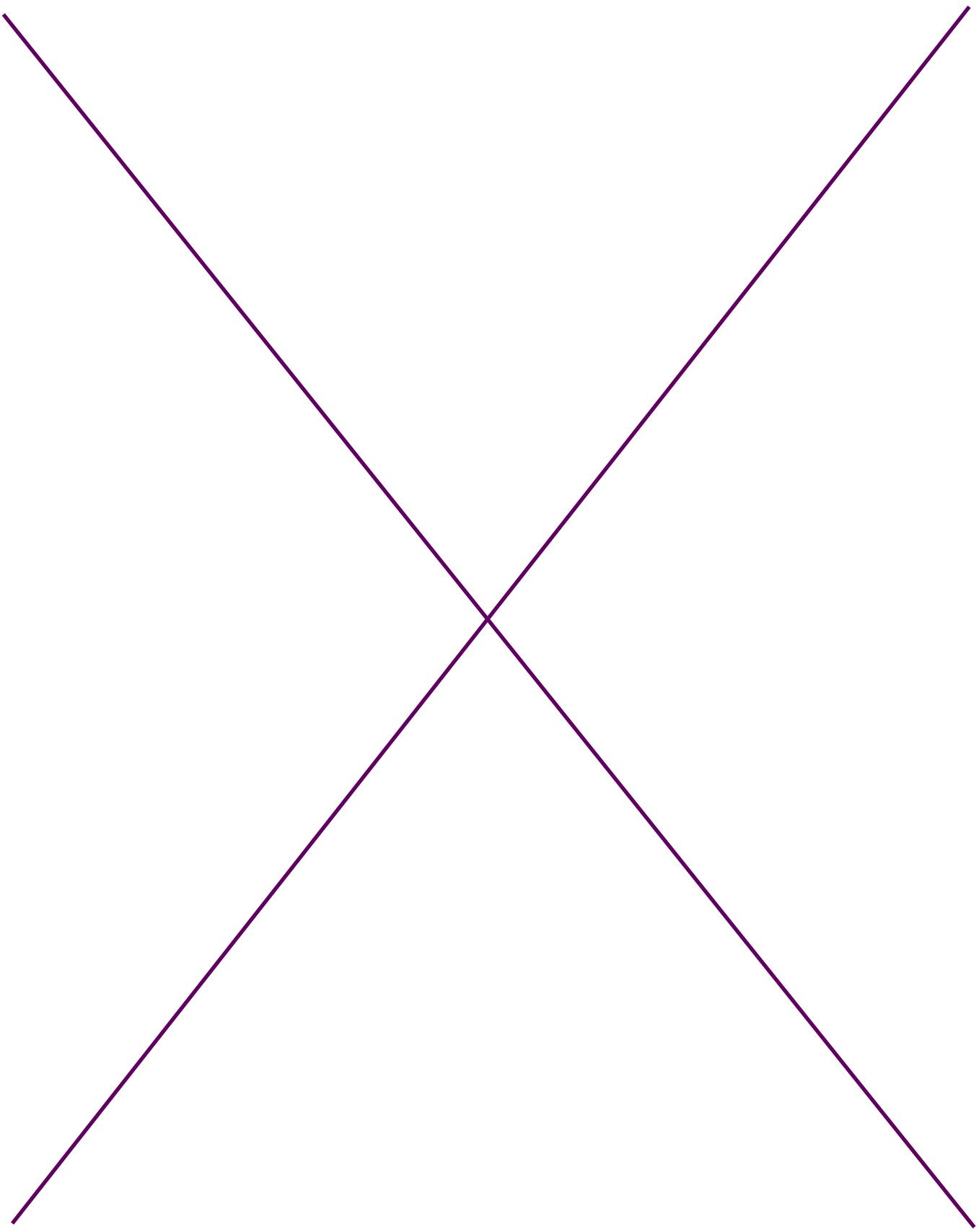
We talked to all the patrons over a 3 hour time span

Oh no.... Prince Charles where are you?????



Signature : Shelby Greer

Date: Oct. 26, 2018



Signature : Shelby Greer

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: October 27, 2018

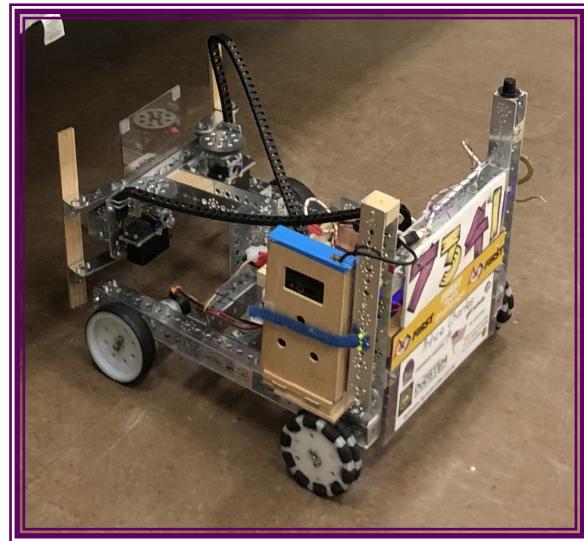
Purpose of the Activity:

Sharing all thing FIRST Tech Challenge at Portofino's while selling Girl Scout Fall Products. We were able to talk to the community and share information on our Robot Prince Charles and what the FIRST program is all about.

Images:

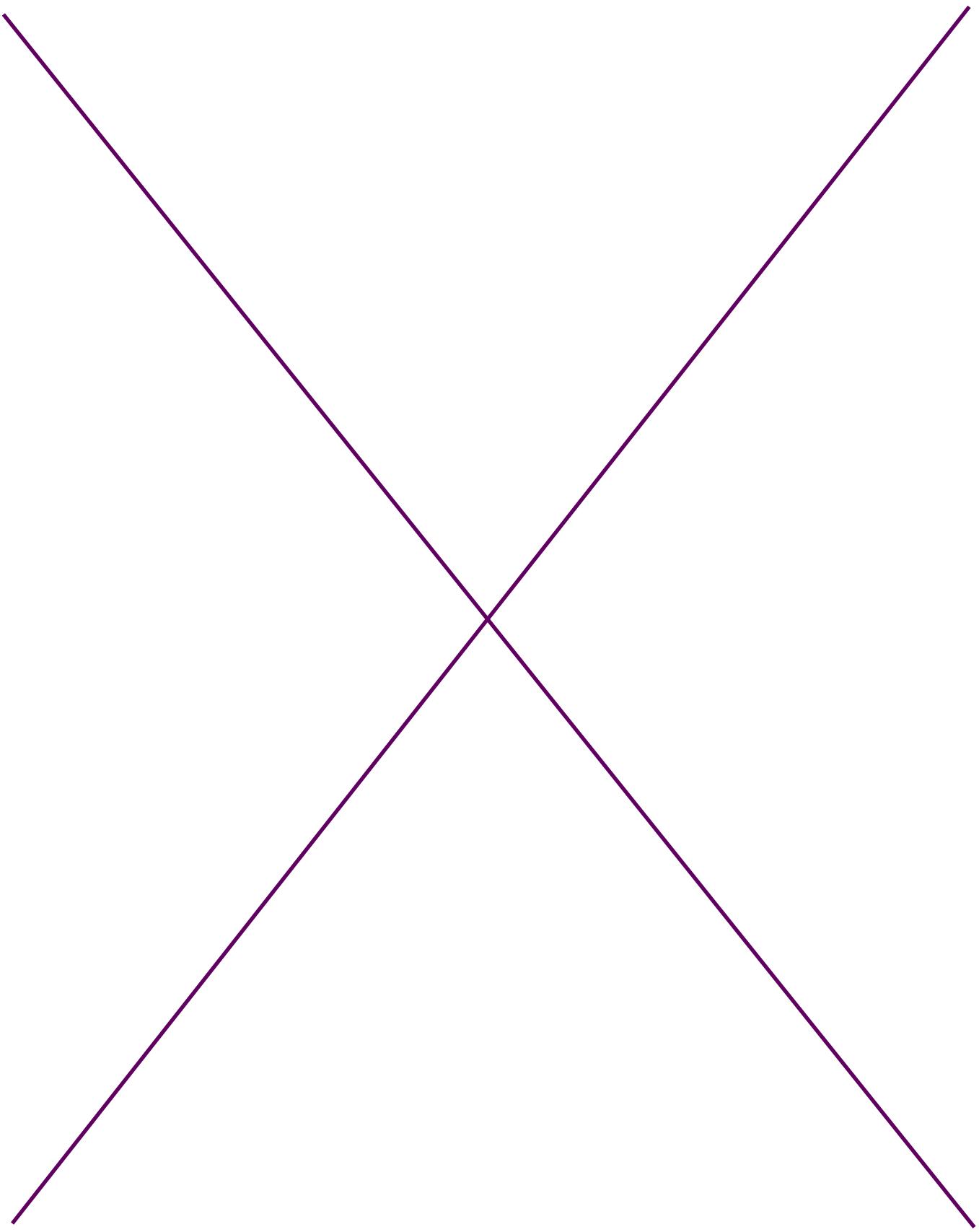


We were showcasing Prince Charles at our booths to let our community know all about FIRST Tech Challenge and the other programs of FIRST.
We talked to all the patrons over a 3 hour time span



Signature : Jessica Anderson

Date: Oct. 27, 2018



Signature : Jessica Anderson

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

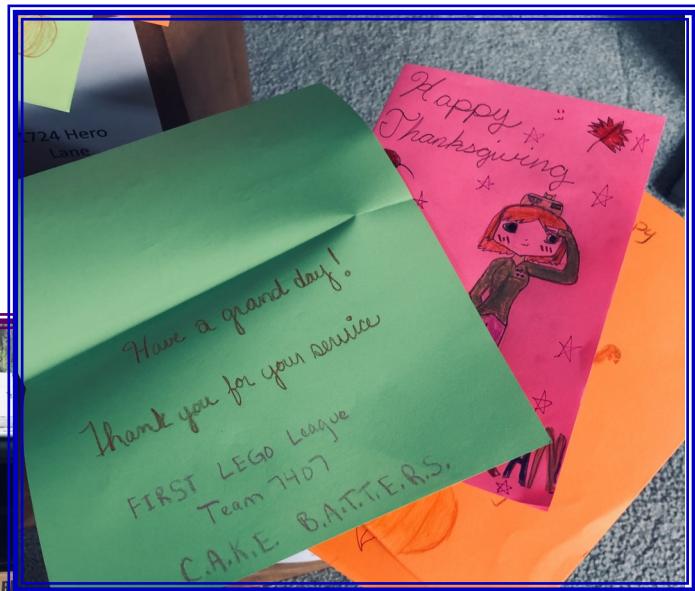
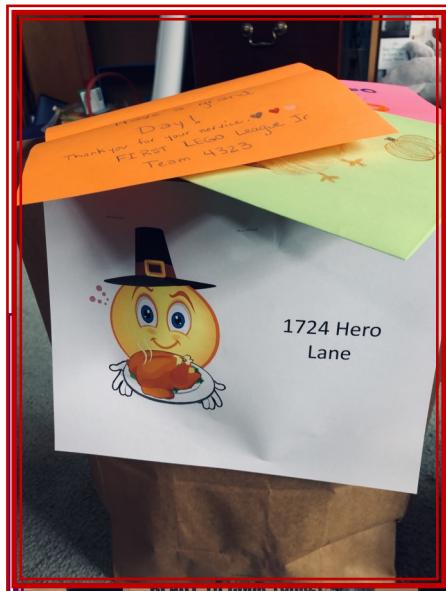
Outreach Activity

Date: November 20, 2018

Purpose of the Activity:

We were at the WinVet Community distributing food to the Veterans who lived there. We helped with the Girl Scout Community Service Project "Dinner in a Bag". Each troop that supports the project will fill a bag with all the fixing that you would need for a Thanksgiving Dinner. Our coach finds the funds to purchase the 25 turkeys that are needed to go along with the food.

Images:

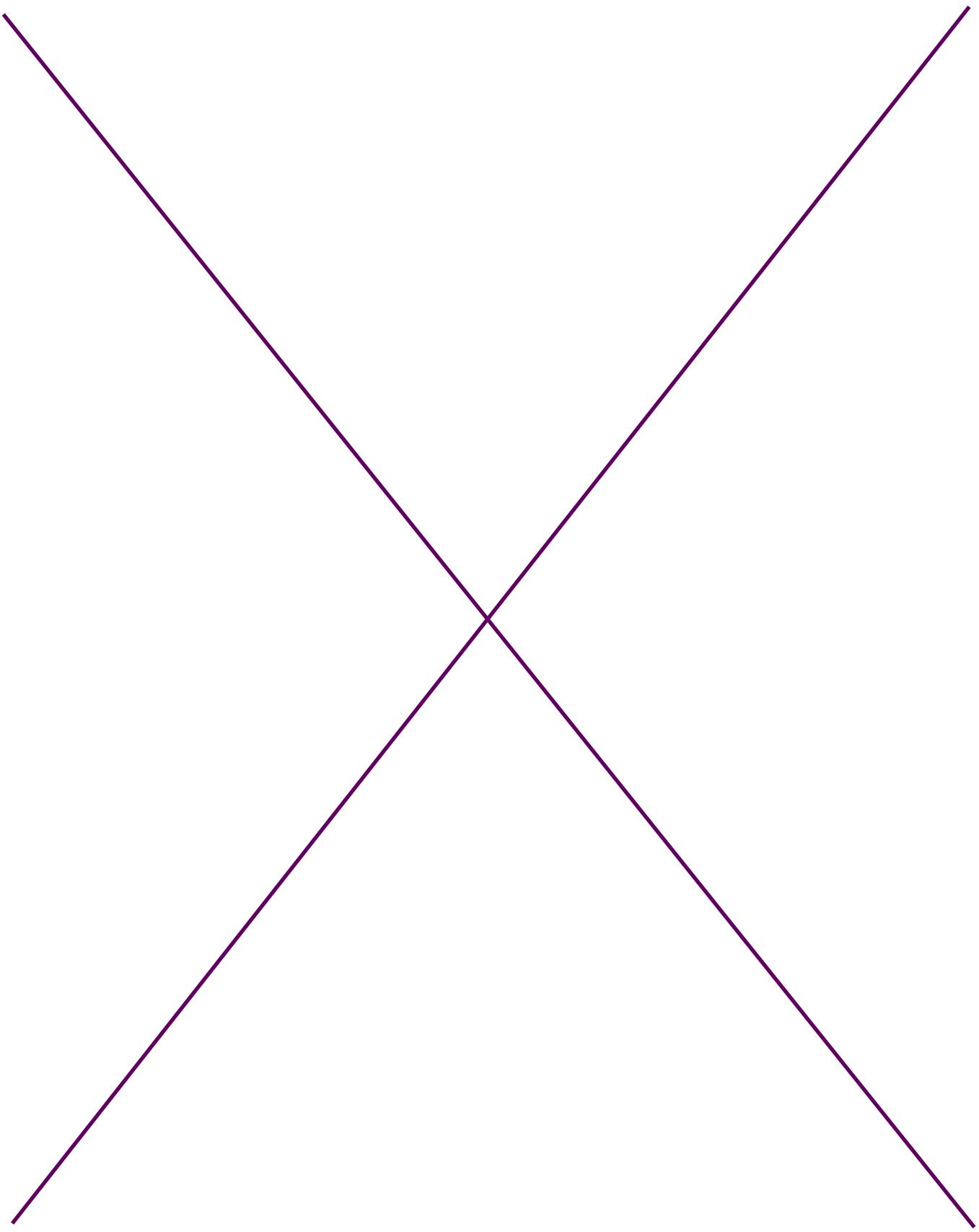


With the help of our Sister Team we made cards to put in the bags of food we put together.

It is great to make the world a better place for others.

Signature : Shelby Greer

Date: Nov. 20, 2018



Signature : Shelby Greer

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: Dec. 6, 2018

Purpose of the Activity:

We attended a Lockheed Martin “Lunch and Learn” event to get employees to come volunteer for the FIRST program. We were able to get a employees to help out.

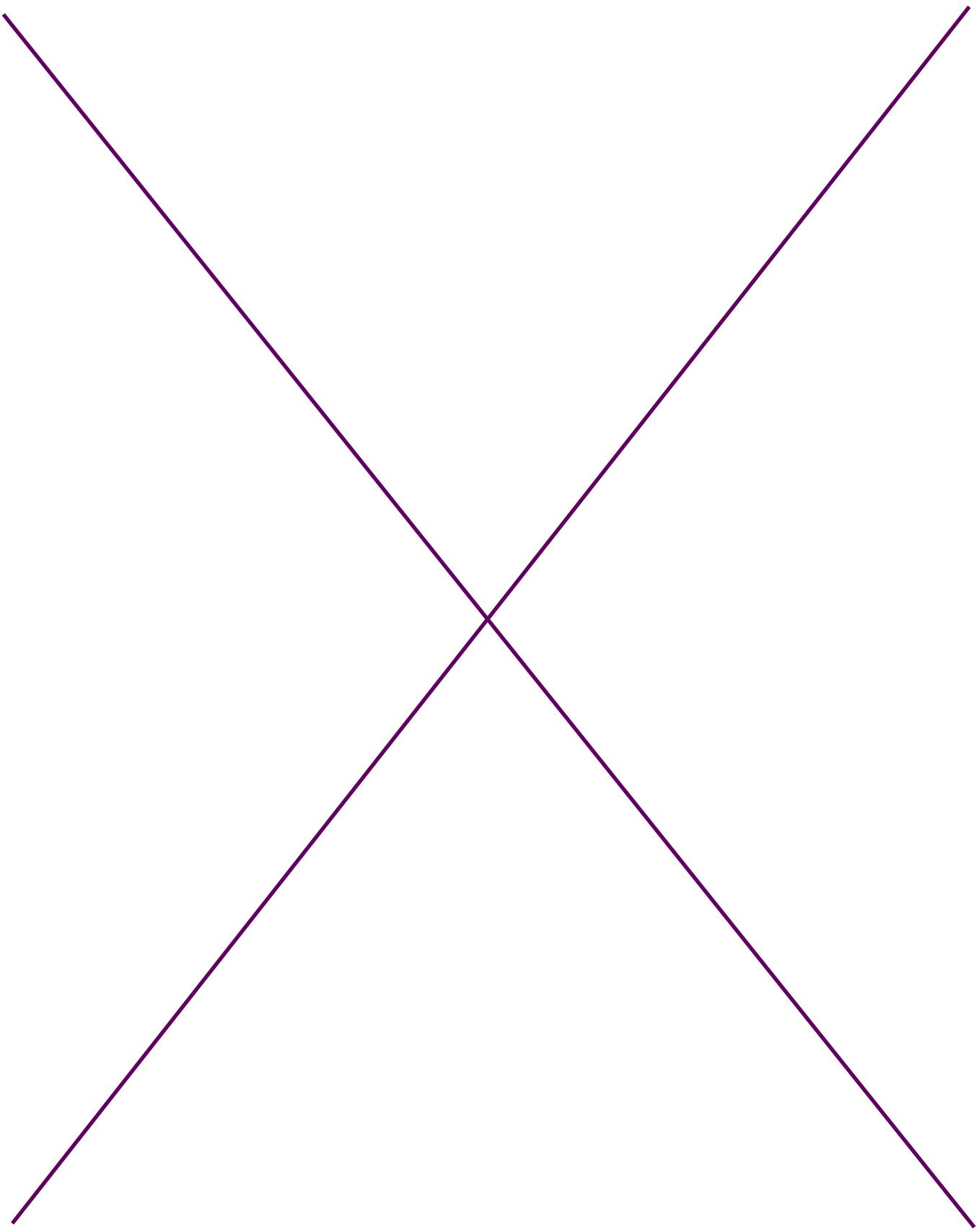
Images:



Barry Bonzack hosted the event at the Lockheed Martin Cape Canaveral Building.

Signature : Shelby Greer

Date: Dec. 6, 2018



Signature : Shelby Greer

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: December 15, 2018

Purpose of the Activity:

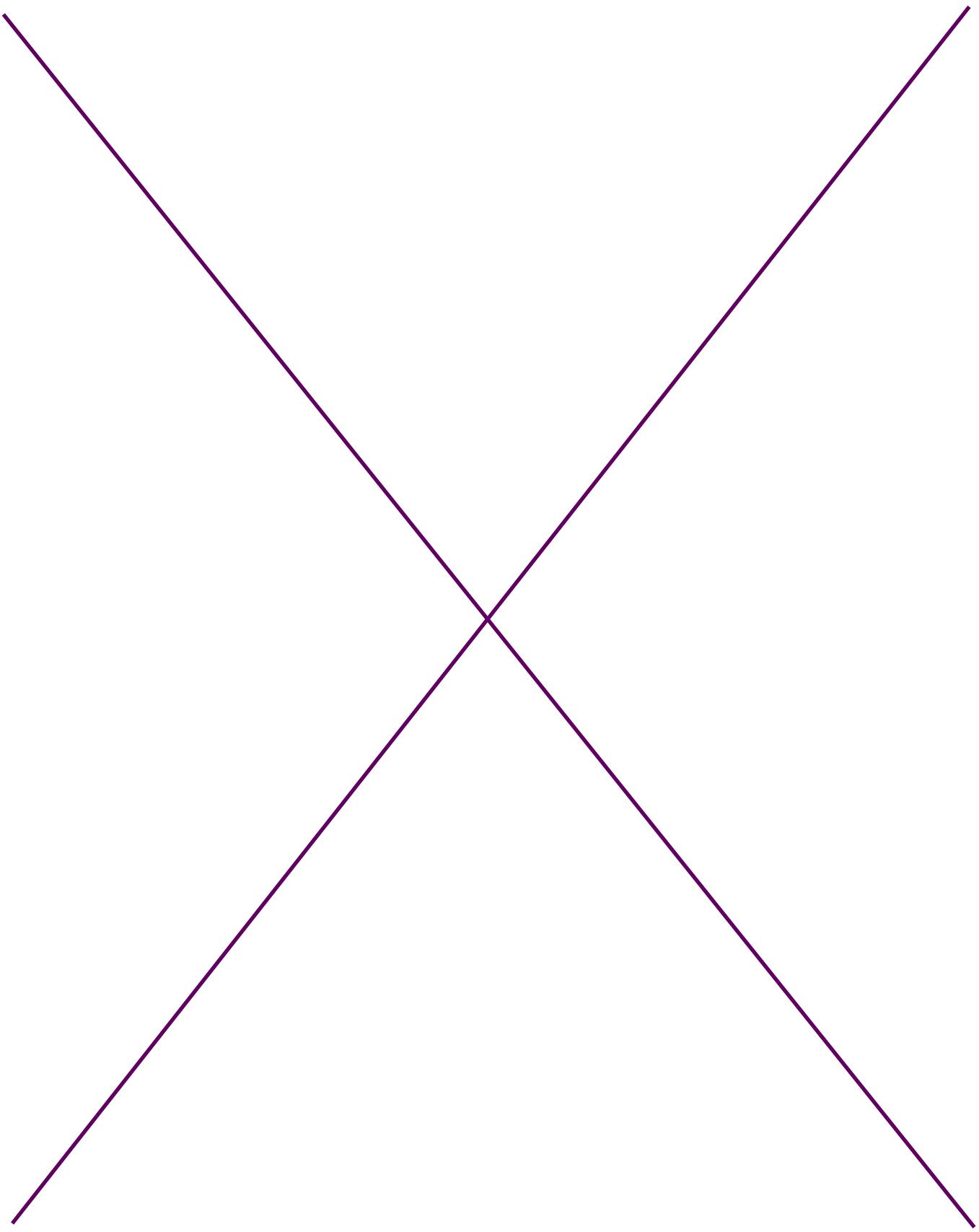
We were giving back to the community by supporting the Toys for Tots program. We stopped by a location in Titusville after our December meet. While there we enjoyed a celebration dinner.

Images:



Signature : Shelby Greer

Date: Dec. 15, 2018



Signature : Shelby Greer

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: December 20, 2018

Purpose of the Activity:

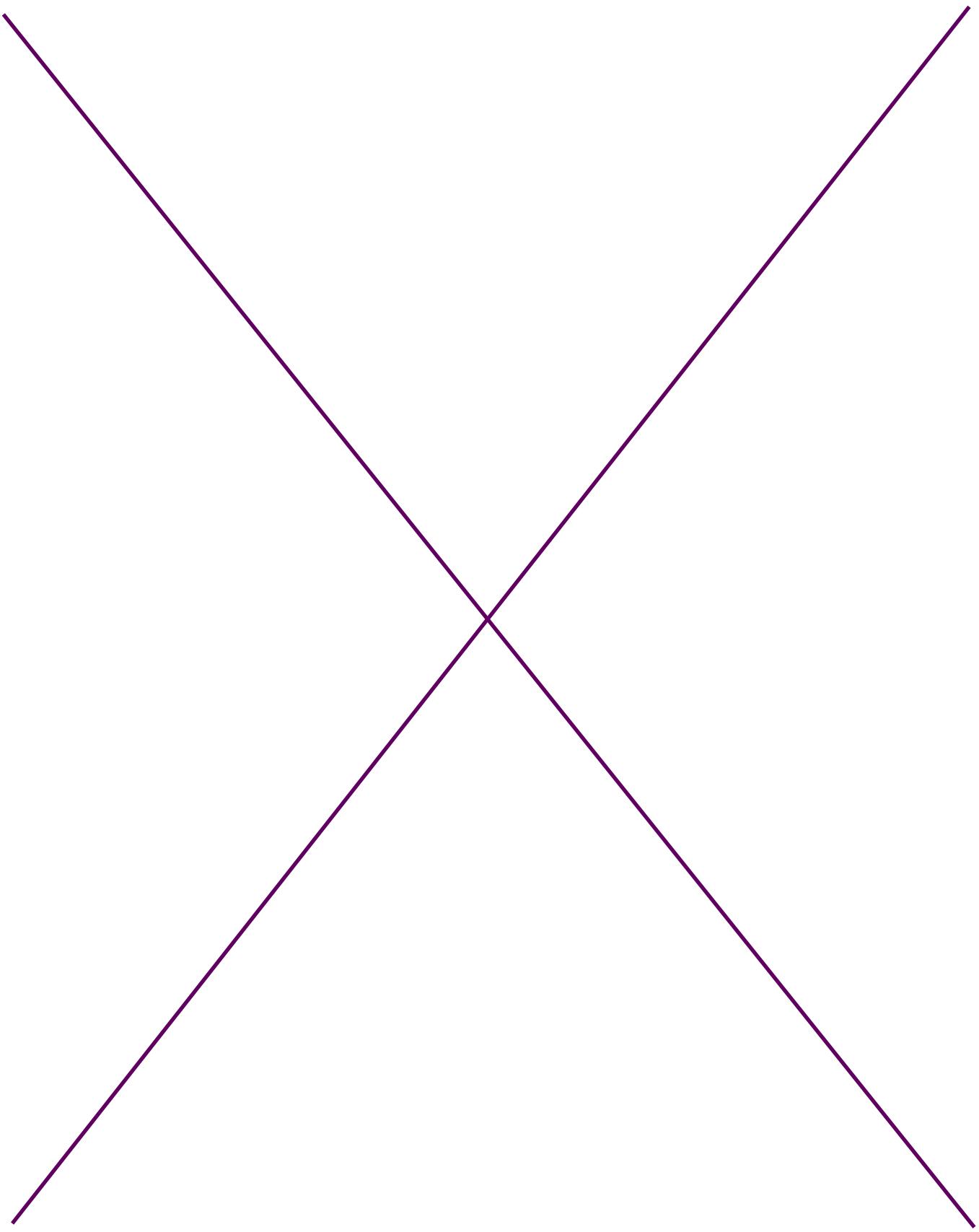
We were at the WinVet Community distributing food to the Veterans who lived there. We helped with the Girl Scout Community Service Project “Dinner in a Bag”. Each troop that supports the project will fill a bag with all the fixing that you would need for a Christmas Dinner. Our coach finds the funds to purchase the 25 turkeys that are needed to go along with the food.

Images:



Signature : Shelby Greer

Date: Dec. 20, 2018



Signature : Shelby Greer

Date: Oct. 26, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: January 19, 2019

Purpose of the Activity:

We helped out at a Ohana Family Fun Day. This was a day about doing activities outside in the fresh air. Some of the stations were set up to work on teamwork skills. We were glad to help out others to learn about teamwork and how much fun it is to work together.

Images:



We were participating in a pre event briefing where we talked about the different activities. We discussed how activity worked and who was going to do which activity. It turned out that no one had a particular activity that

they wanted to work on, so it was decided that the girls would walk around and go to an activity that needed support.

We started the day eating lunch and worked until about 3:00 with the teardown on the event.

It was great to be in the outdoors.



Signature : Jessica Anderson

Date: Jan 19, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity Continued

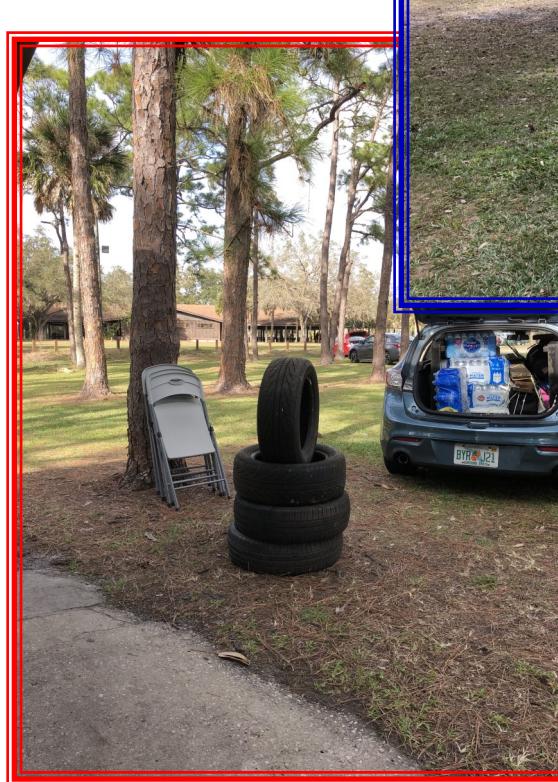
Date: January 19, 2019

Images:



Fun team work activities.

Left foot.... Right Foot... Left Foot...
Right Foot



Signature : Jessica Anderson

Date: Jan 19. 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: January 20, 2019

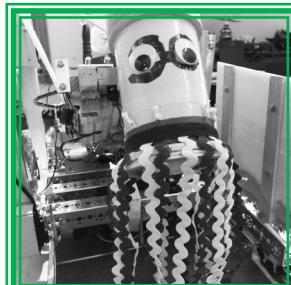
Purpose of the Activity:

We attended Otronicon at the Orlando Science Center. We had a table showing two aspects of FIRST. The FIRST LEGO League Jr. model was on display where the people interacted with the elements of Mission Moon. The FIRST Tech Challenge Robot Princess Charlie and the Engineering Notebook was available for review. We shared the room with other FIRST LEGO Jr. League, FIRST LEGO League, FIRST Tech Challenge and FIRST Robotics Competition teams. Hundreds of people stopped by to see the model and test drive the robot.

Images:



Working with our Sister Team the B.E.E.s and C.A.K.E. B.A.T.T.E.R.S.



We are on the road again.... Traveling to Orlando.....
About a 45 minute drive....



Signature : Jessica Anderson

Date: Jan 20. 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

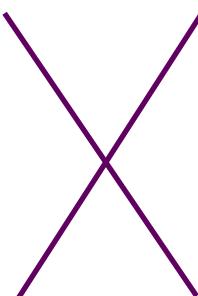
Outreach Activity Continued

Date: January 20, 2019

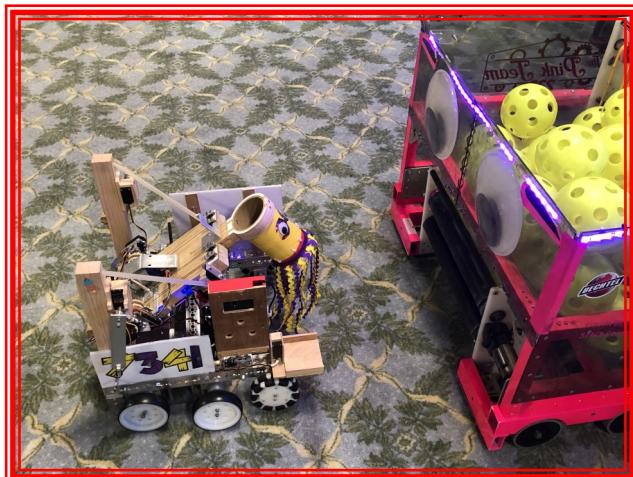
Images:



We were learning new things from other teams as well as sharing our design with others.



Made a new friend!! We just have to be careful while we play together.....



Signature : Jessica Anderson

Date: Jan 20. 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity

Date: January 26, 2019

Purpose of the Activity:

We attended the Citrus of Girl Scout Annual Meeting to showcase the robotics activites our team and two sister teams what accomplished. There were members from all 6 counties (Brevard, Lake, Orange, Osceola, Seminole, and Volusia). Which has a potenial of reaching 15,000 girls. The Leaders and Girls come from all walks of life each having a potential to learn from the FIRST program.

Images:



Jessica was also a Community Delegate for the meeting. We had one of our up and coming team member help with our showcase. Tempy will be moving up from the FIRST LEGO League Team C.A.K.E. B.A.T.T.E.R.S. next season.

We also showcased the FIRST LEGO League robot and project as well as the FIRST LEGO League Jr. model. Everyone enjoyed the LEGOsssss.



Signature : Jessica Anderson

Date: Jan 27. 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Outreach Activity Continued

Date: January 26, 2019

Images:



Maryann Barry (CEO of Citrus of Girl Scouts) checking out the FIRST LEGO League Jr. model, FIRST LEGO League robot and project and the FIRST Tech Challenge robot.

We had several young ladies come and test drive Princess Charlie.



Signature : Jessica Anderson

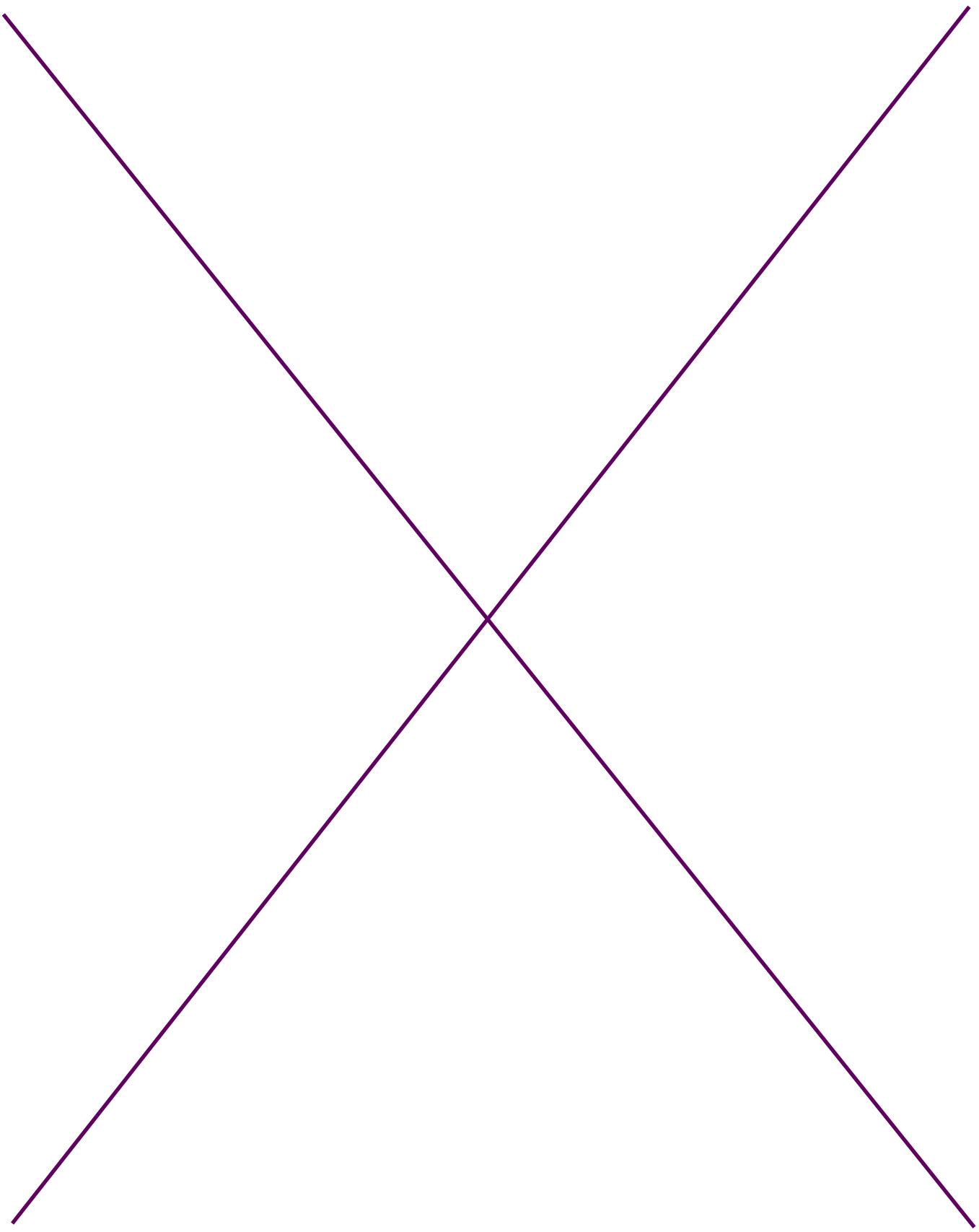
Date: Jan 27. 2019

OPEN ENGINEERING

TECHNOLOGY

Signature : Jessica Anderson

Date: Jan 27, 2019



Signature : Jessica Anderson

Date: Jan 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: September 2, 2018

Purpose of the Activity:

Build the new chassis

Create a telop

Test the new motors

Process:

Today's meeting started with the decision to keep Prince Charles and build a new robot. By the end of the meeting we named her Princess Charlie. We will be using the following:

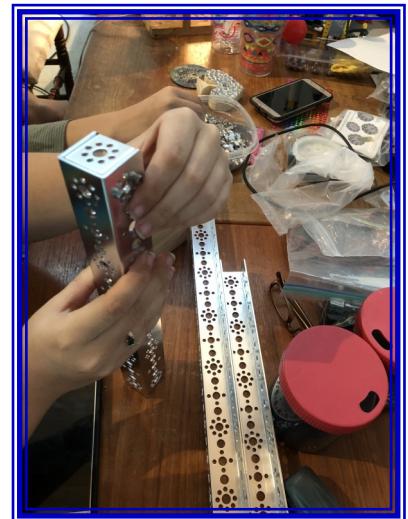
1. 2 Rev controller since we anticipating having to have 8 motors to support all the functions
2. 4 Standard wheels
3. 2 Omni wheels for steering
4. 4 AndyMark 60 motors for driving, since we do not know what the mission will be this year....

Using the drawing on the next two pages and all our new parts, we put our base chassis together. We are taking the approach that this part of the

robot will not change (very little if any) as are using the nuts with the nylon locking feature. Also we are using 3 bolts and nuts in a triangular pattern for a stronger hold. The connecting beams will be less likely to twist under pressure. This approach was taken con-



nnecting all the different beams together-



Signature : Jessica Anderson

Date: Sept. 2. 2018

Team 7341

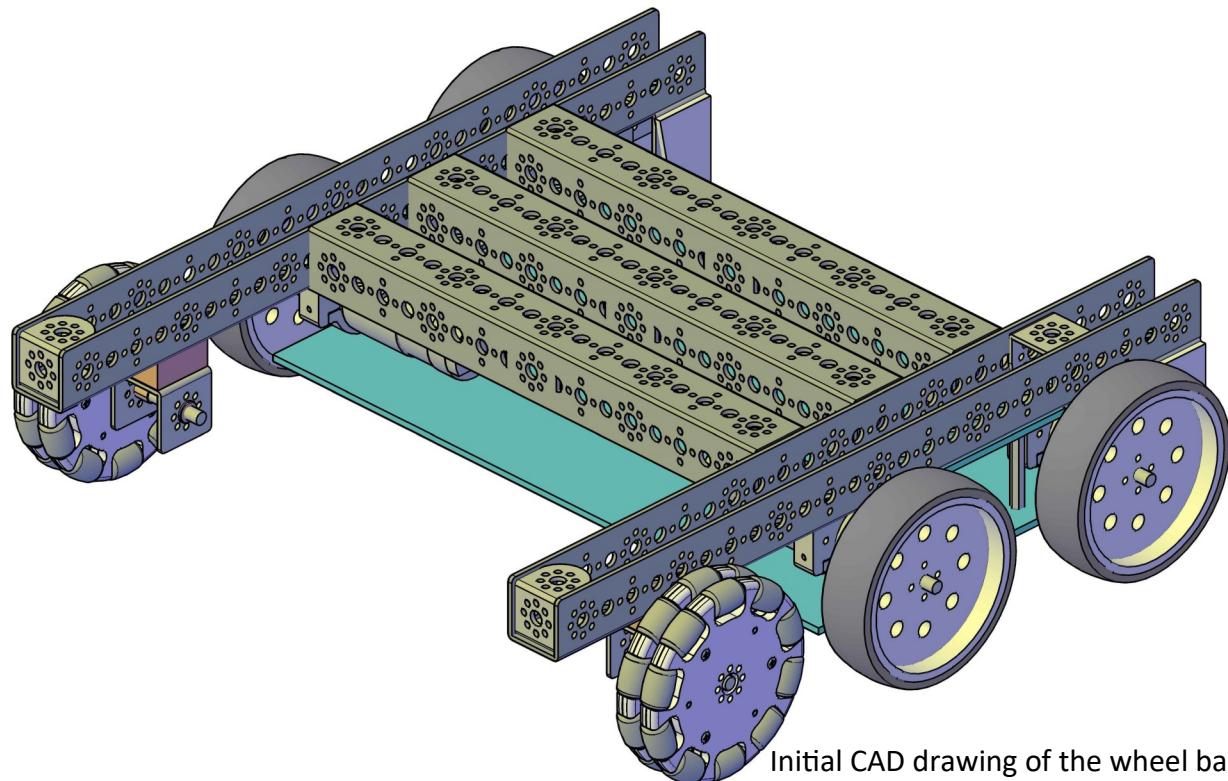
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 2, 2018

Process:

er. We added the electronic hardware from the bottom of the robot leaving the top to support any type of mission that might come in the future. We are ready to bring Princess Charlie to the reveal if we can....



Initial CAD drawing of the wheel base.

Signature : Jessica Anderson

Date: Sept. 2. 2018

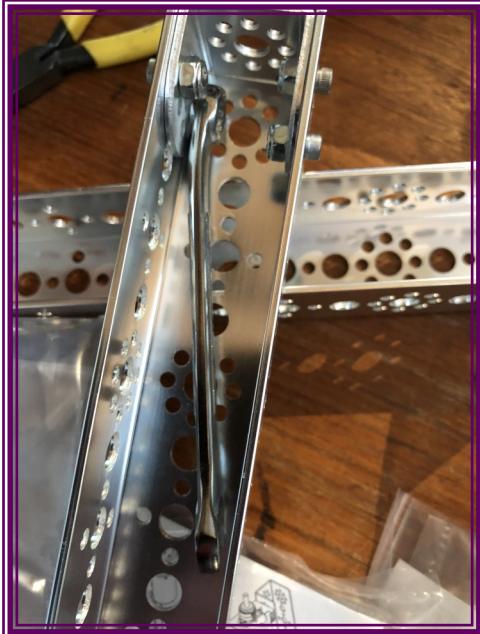
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 2, 2018

Process:



Oh...no.... the wrench is stuck. We worked hard and at getting all the bolts and nuts tight so the robot does not fall apart. When we lifted up the chassis after getting all the pieces on We found some leftover parts.

Oh no Princess Charlie threw up. What a mess!!



Signature : Jessica Anderson

Date: Sept. 2. 2018

Team 7341

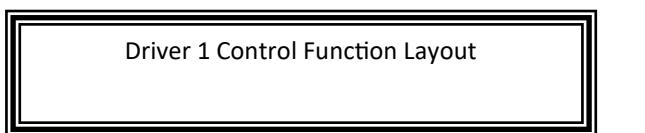
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 2, 2018

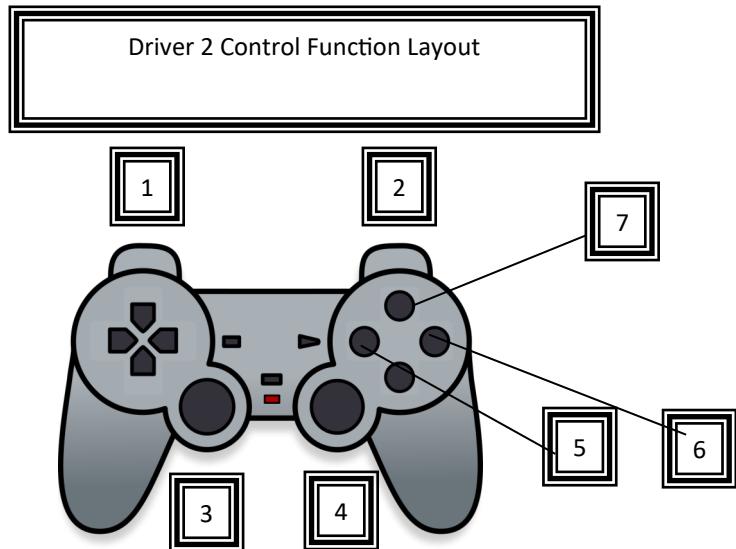
Process:

Described below is the functional layout for the robot controller buttons.



Driver 1 Control Function Definition

1. Left Wheels control
2. Right Wheels control
3. Outer Lift control down
4. Outer Lift control up



Driver 2 Control Function Definition

1. Move the phone left
2. Move the phone right
3. Lift and lower grabber
4. Extend and retract grabber
5. Make Grabber go backwards
6. Make Grabber pick up items
7. Stop Grabber

Signature : Jessica Anderson

Date: Sept. 2. 2018

Team 7341

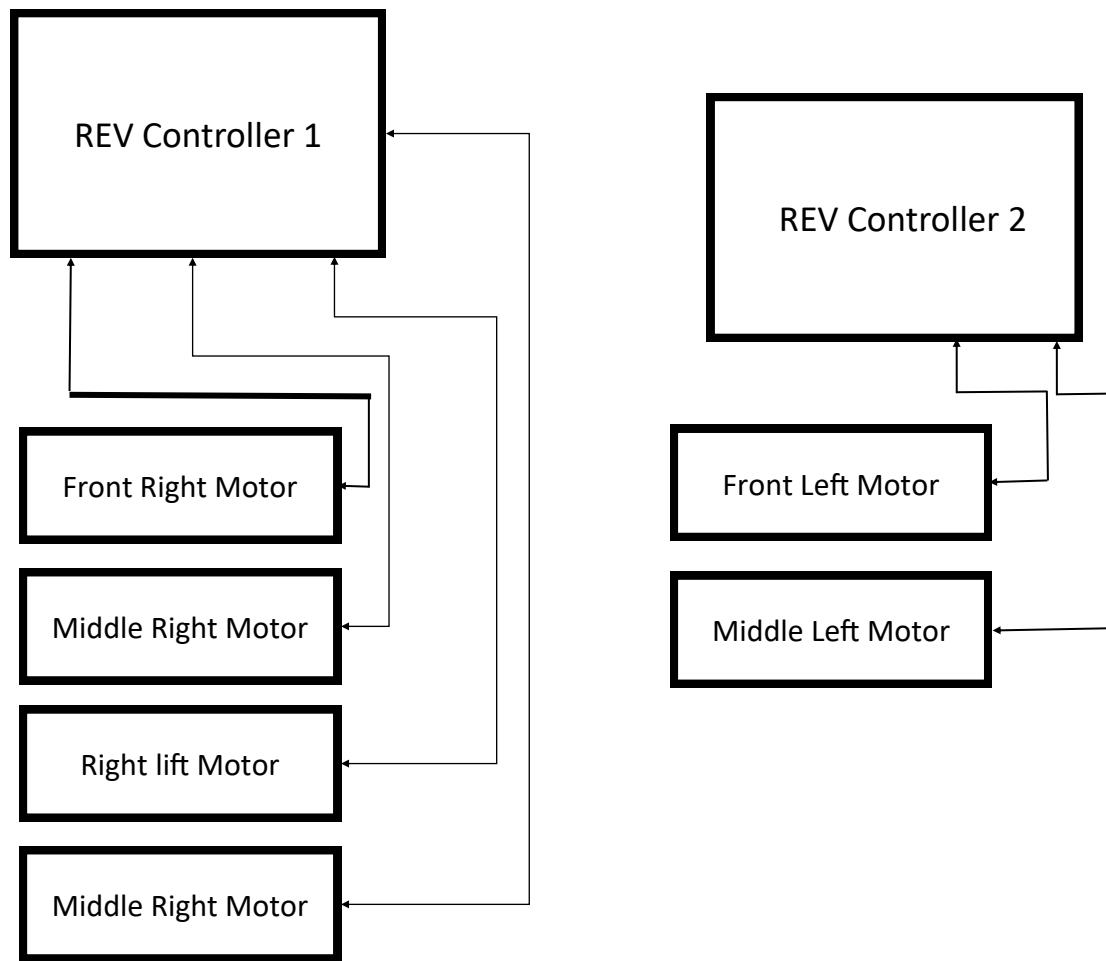
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 2, 2018

Process:

Following describes the hardware layout of the robot



Signature : Jessica Anderson

Date: Sept. 2. 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: September 15, 2018

Purpose of the Activity:

Put the lander together

Process:

We worked the entire meeting building the lander. We had several small issues in getting it together. One was that the brackets that we were to hang from were just a little too narrow, so we used a vise to push them out a little to make it easier to put on the lander.

Another issue was with the stanchion and that they did not have threads in the end so we decided the pre-tap them to make it easier to put everything together.
Our first attempt

was to use the small vise and a hand tool to screw in the bolt. This took a lot of pressure and our hands started to hurt. We next tried using the drill, but the vice moved around too much and we could not hold it still. Finally we took the stanchion to a big vise and we were able to tap



Signature : Jessica Anderson

Date: Sept. 15. 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 15, 2018

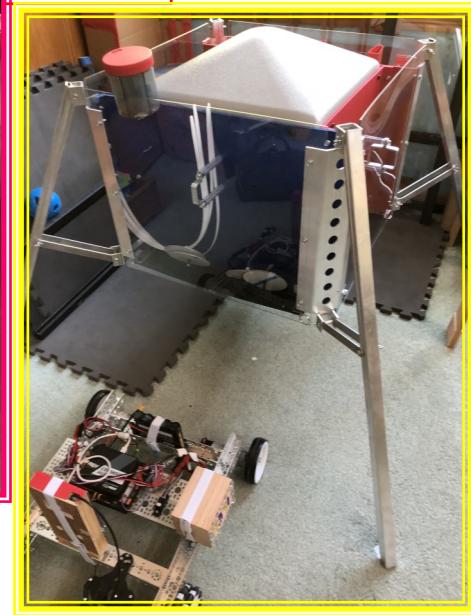
Process:

the rest of the stanchion were tapped in no time at all. The last part that was a little tuff to complete is

putting in the pin for the top of the lander. We had to find something to help put the plastic pins.



After putting the lander together we placed Princess Charlie under and verified that she would fit with no issues.

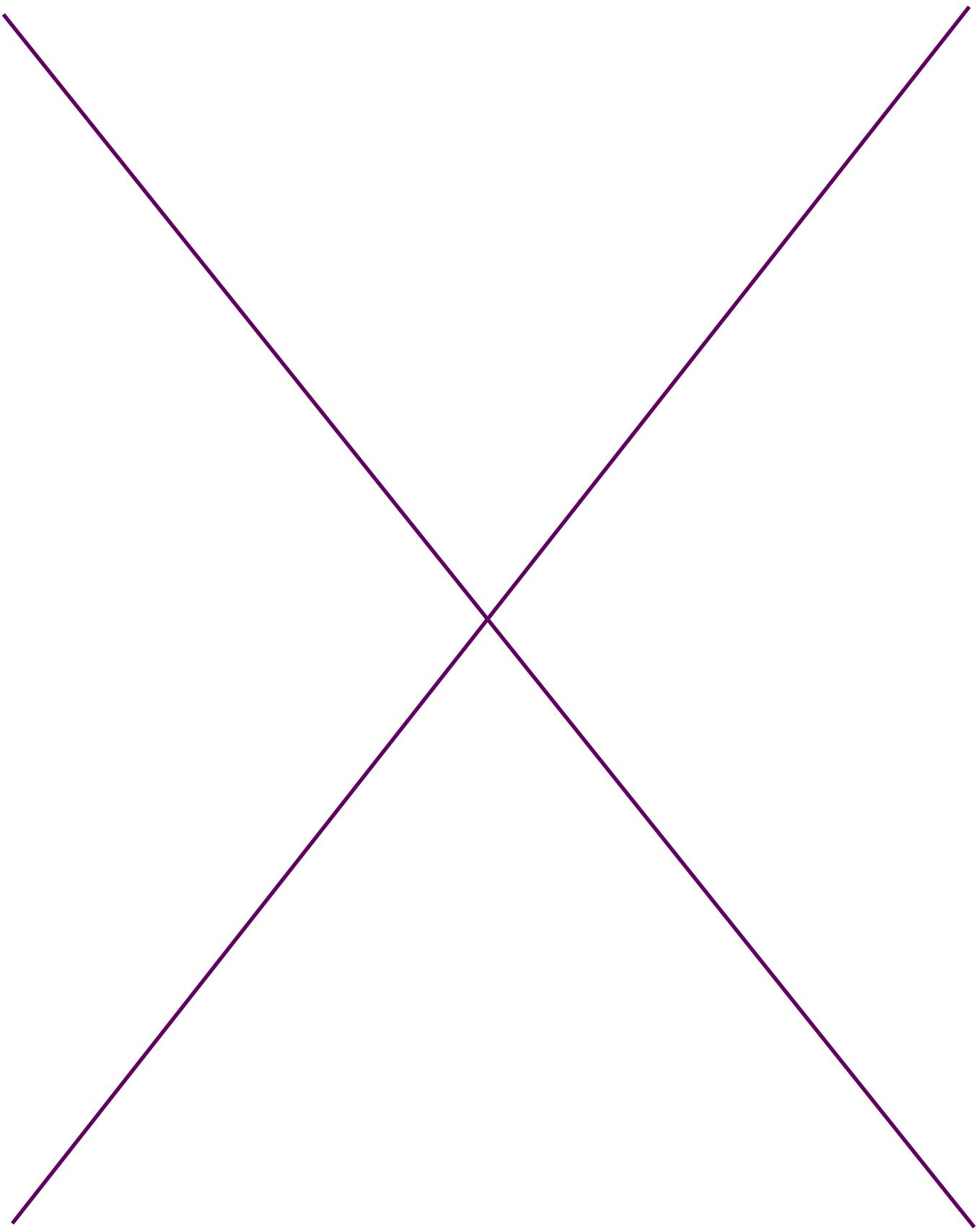


We were finished for the day and next time we will be working on how to move the minerals into the lander or the depot.

(Handwritten signature: Jessica Anderson)

Date: Sept. 15. 2018

Signature : Jessica Anderson



Signature : Jessica Anderson

Date: Sept. 15. 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: September 18, 2018

Purpose of the Activity:

We are one of the winners!!!!

Winners of the #TextrixCausesARuckus contest.

Process:

Wow!

Winners of a torqueNATO motor. We cannot wait to use it on our robot!!!

Congrats to all the winners of this first contest.

Best of luck!

We cannot wait until the motor comes in the mail. We have the perfect spot for our new motor.

Congrats to our 1st round of **#TETRIXCausesARuckus** winners:
#3565 Westford Academy Ghost Robotics, [@theDotsRobotics](#), [@FTC_FRENCHFRIES](#), #5309 FTC Plan B Robotics, & [@Supersonic11931](#)!
➡ bit.ly/2NRlvmr to get bi-weekly tips / enter our out-of-this-world giveaway.

[@FTCTeams](#)

TETRIX CAUSES A RUCKUS GIVEAWAY
Week 1 Winners: TorqueNADO

Team #3565 Ghost Robotics

Team #14779 The Dots

Team #7341 F.R.E.N.C.H. F.R.I.E.S.

Team #5308 Plan B

Team #11931 Supersonic

PITSCO

1

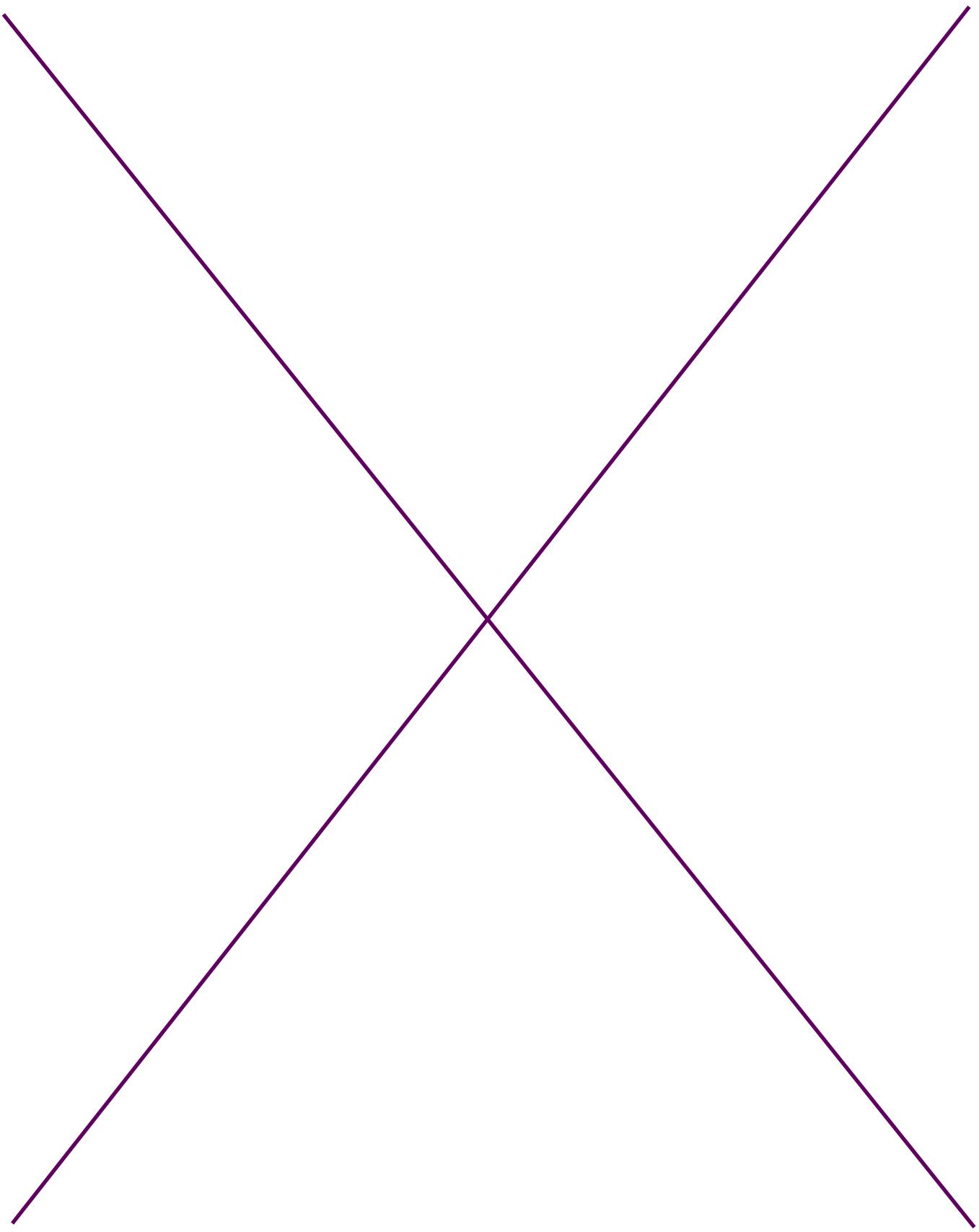
1

10



Signature : Shelby Geer

Date: Sept. 18, 2018



Signature : Shelby Geer

Date: Sept. 18, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: September 23, 2018

Purpose of the Activity:

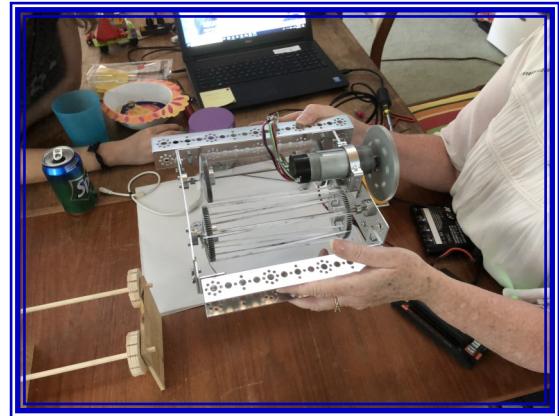
Build and test the prototype of the mineral grabber

Thank you to Pitsco!

Process:



We have most the prototype together, there still an issue with the gears on the side. Things are still binding.



Signature : Shelby Geer

Date: Sept. 28, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 23, 2018

Process:

We saw on social media that we were one of the winners in the Pitsco drawing and when the motor came in we could not believe it!!! We knew where to use the motor.... On the center part of our lift mechanism. Thank you PITSCO!!!!



Signature : Shelby Geer

Date: Sept. 28, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: September 30, 2018

Purpose of the Activity:

Finish prototype grabber

Check out old linear slide to see how many parts we need to get to the top of the lander

Process:

In the spirit of saving money we took parts off of Prince Charles to build up Princess Charlie. Taping the

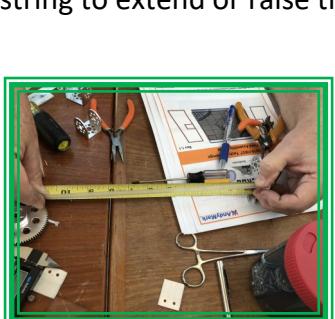
wooden linear slide we discussed that we needed the outside plus 2 levels in order to reach the top of the lander. A wooden prototype was created as a proof of concept for the collector. The collector would be able to pick up either the gold or silver mineral (block and ball).

The next discussion was on how are we going to move the linear slide up and down. Since the slide uses gravity to lower the parts it would not allow for hanging at the start of the match. So we looked into having tubing to hold the slide part down (or compressed) and

string to extend or raise the linear slide. In working with the tubing we could purchase 7" tubing would not stretch enough so we could reach our goal. We did create special stops to hold the tubing.



Parts from Prince Charles



Signature : Shelby Geer

Date: Sept. 30, 2018

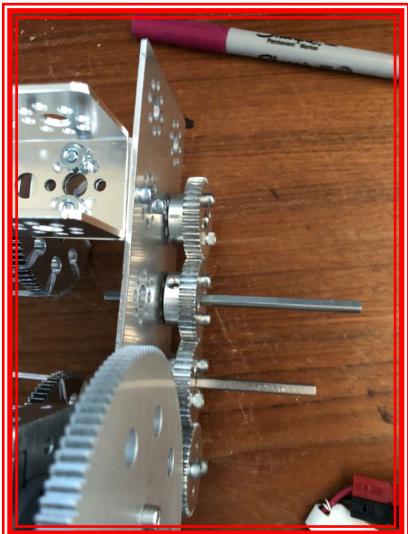
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: September 30, 2018

Process:

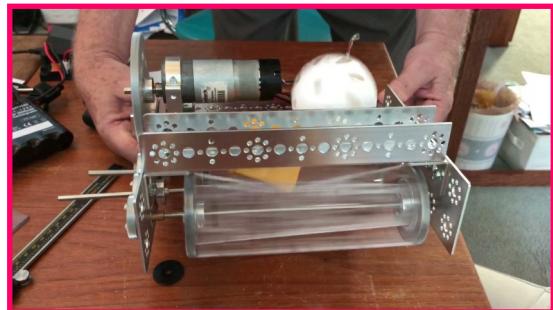


Using the wooden prototype we created a functional prototype using parts we had in our box. We did need to cut down axle bushings and make Delrin washers when creating our prototype. After we cut all the pieces we assembled the grabber and had a successful test. We made use of elastic for the grabbing surface.

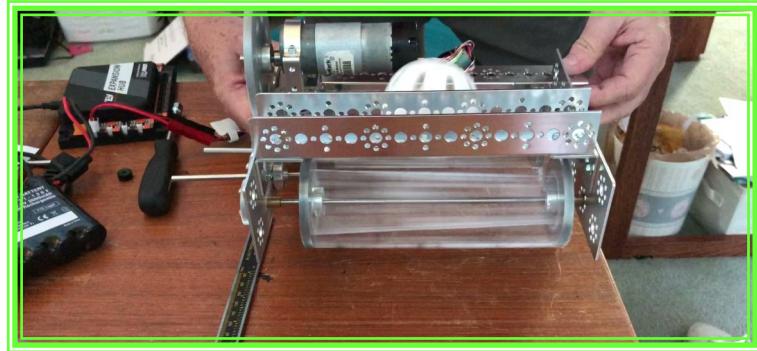
The prototype used 120 tooth gears where are working model will be made using birch plywood. The side



will be constructed using part of last year's game element. We will also house the gears inside the



grabber/collector so that they don't get entangled with anything.



Signature : Shelby Geer

Date: Sept. 30, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

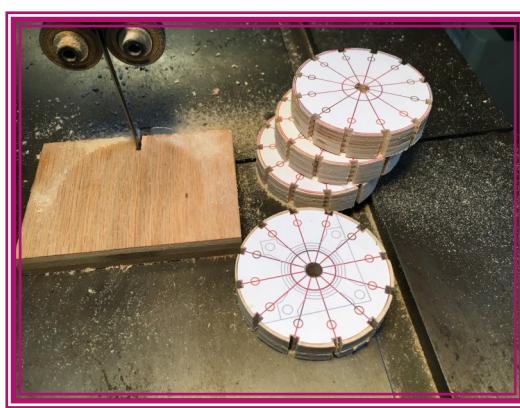
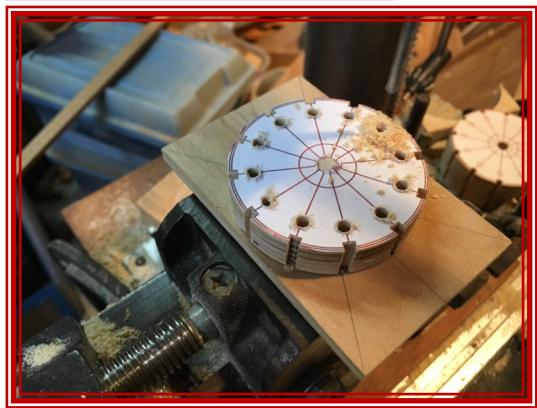
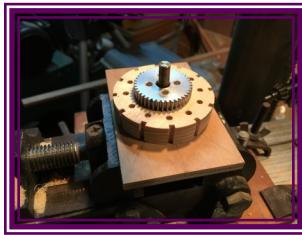
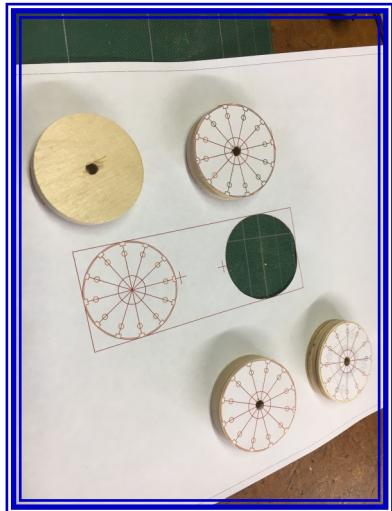
Date: October 14, 2018

Purpose of the Activity:

Creating of the first article of the mineral grabber

Process:

The cogs for the mineral grabber are designed and made out of wood.



Signature : Shelby Geer

Date: Oct. 14, 2018

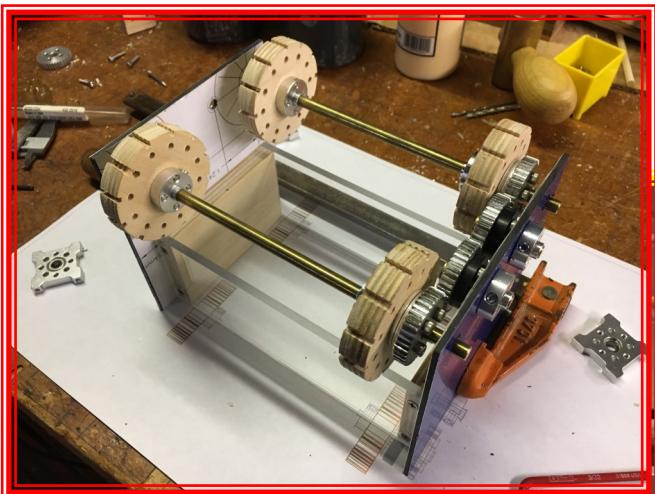
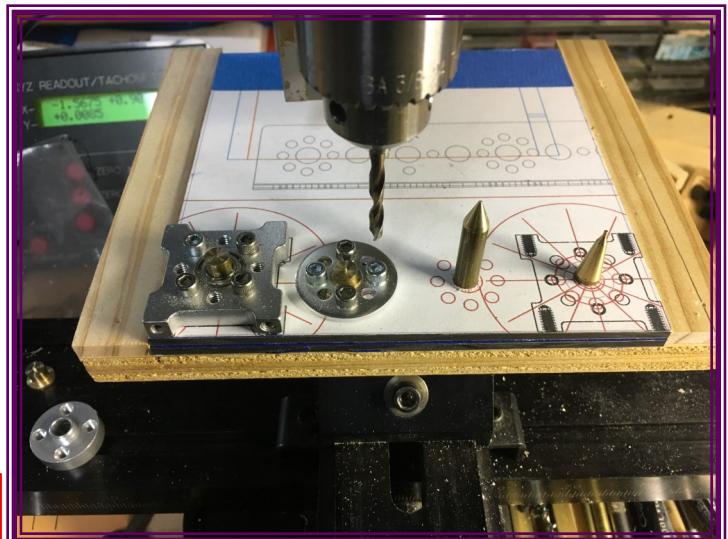
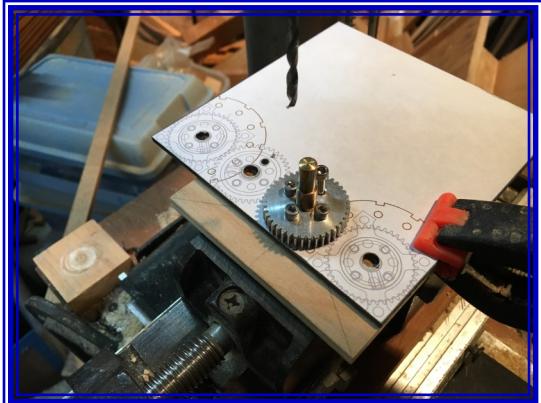
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 14, 2018

Process:



Various stages of the motorized end effector

Signature : Shelby Geer

Date: Oct. 14, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: October 21, 2018

Purpose of the Activity:

Adding the phone to Princess Charlie in the proper place

Stowing robot control wires

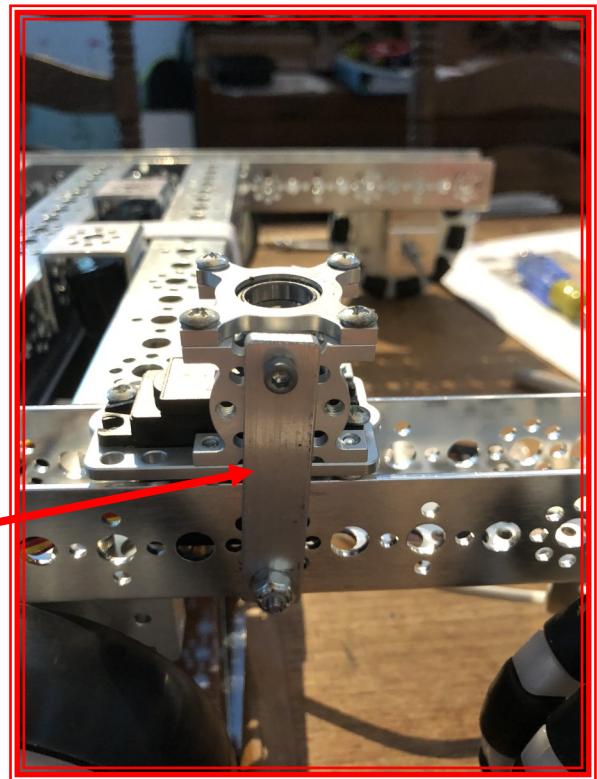
Process:

We have determined that we will be using a servo under the phone box so that we can move the phone to find the targets since when the phone is in the portrait mode you cannot see all the minerals.

The ServoCity servo block was added with brackets we made. This block help to keep the servo safe from being damaged.



Special Bracket



Signature : Shelby Geer

Date: Oct. 21, 2018

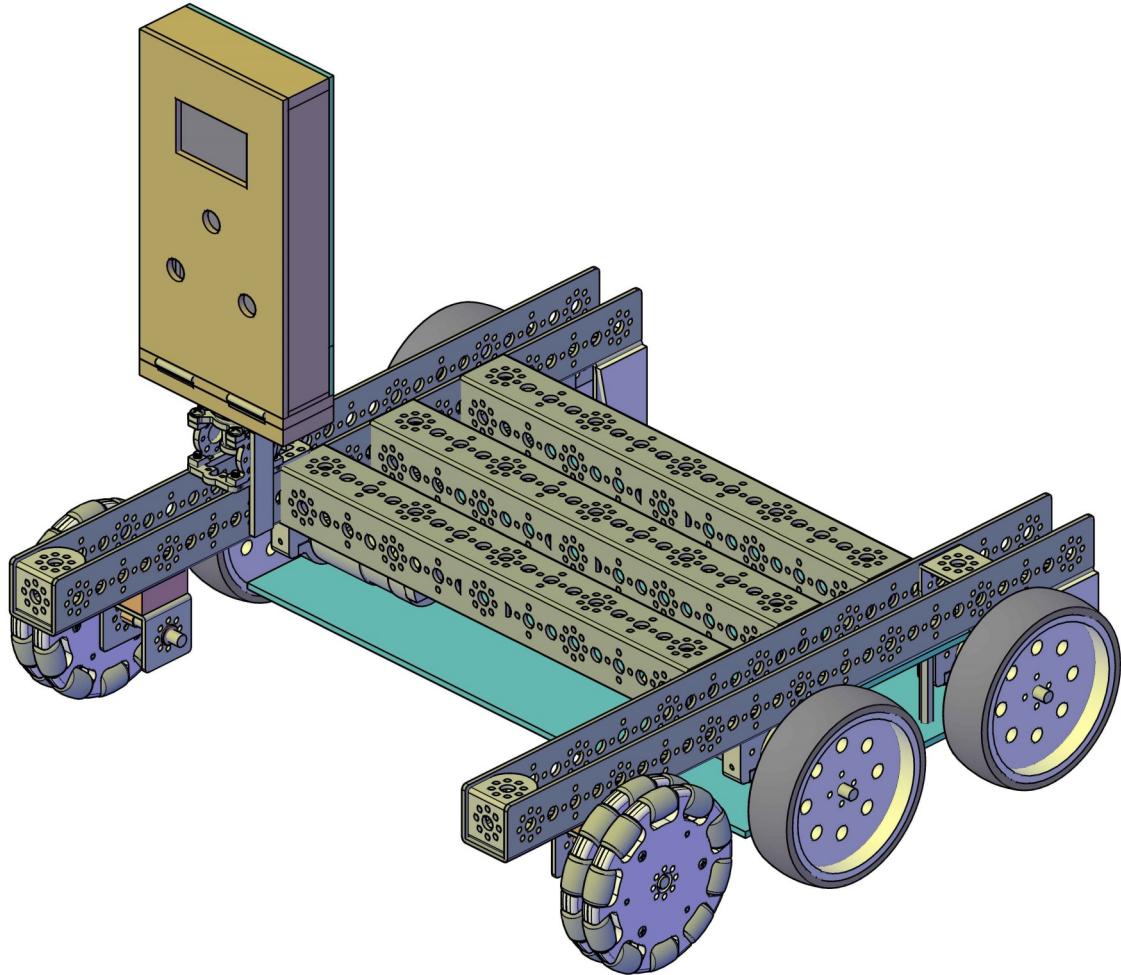
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 21, 2018

Process:



Using the phone box from last year we added a servo bracket underneath the box allowing the phone to rotate to find the mineral during autonomous. This is needed because we left the phone in a vertical position and not landscape where the camera could have seen all three elements.

Signature : Shelby Geer

Date: Oct. 21, 2018

Team 7341

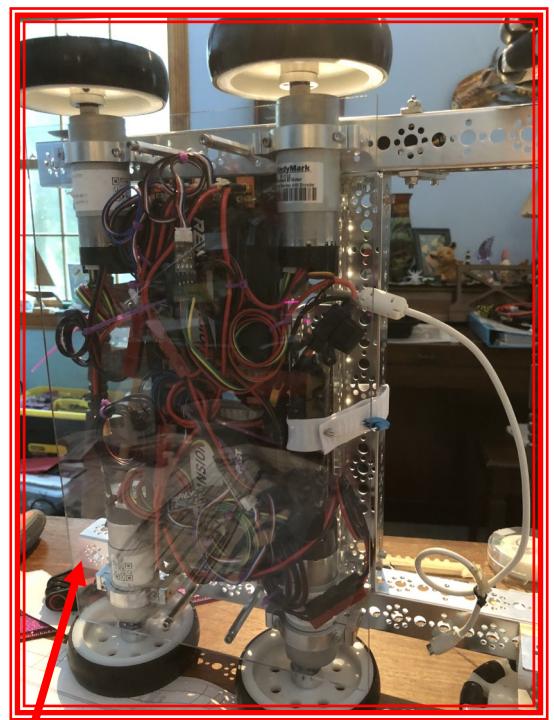
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 21, 2018

Process:

We obtained the REV mini cab to phone cable to have a secure way of attachment between the phone and the REV electronics.

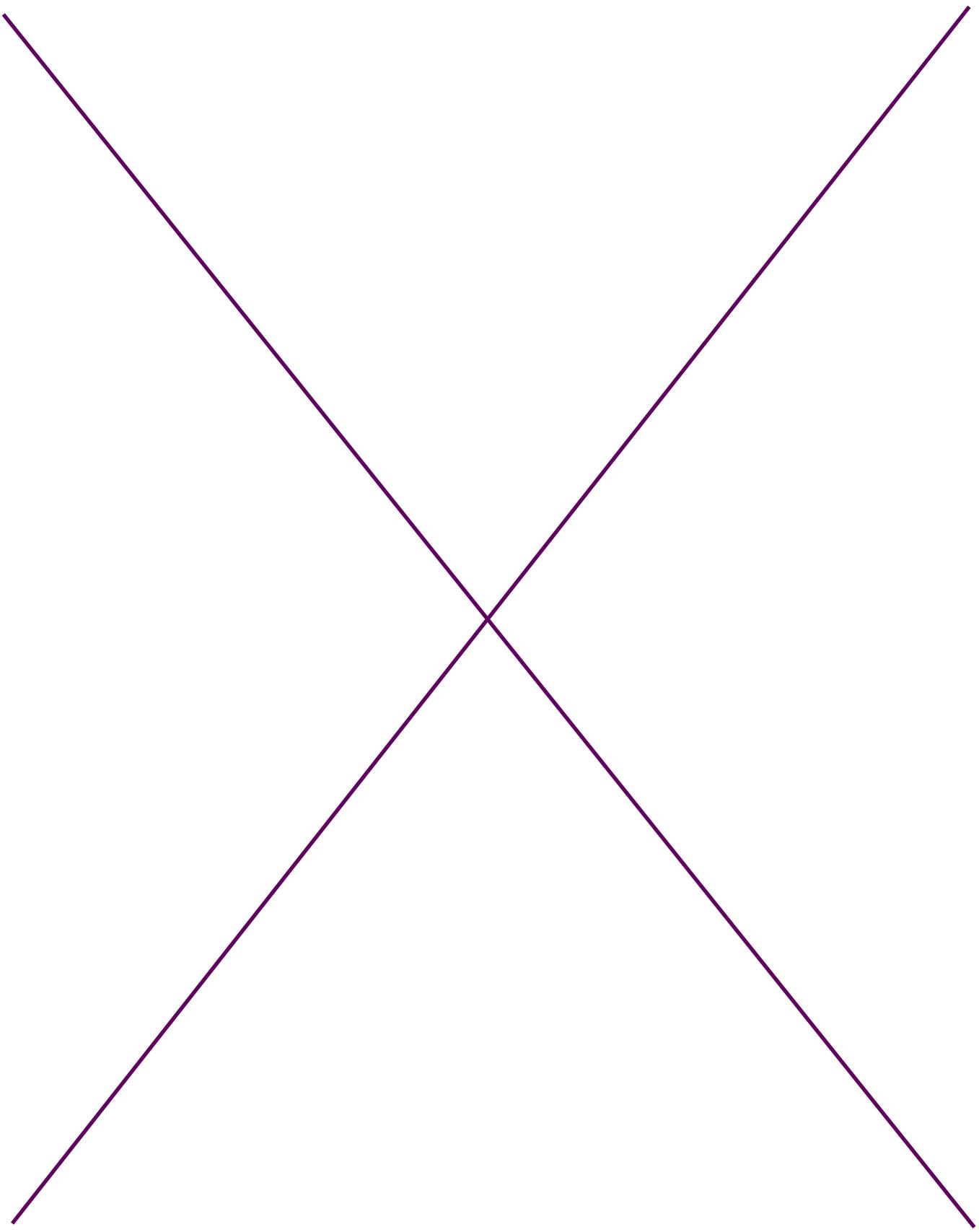


We added a piece of plexiglass to the bottom of the robot to keep our cable secure. This also allows us to see the state of the REV controller boxes.

Once we finished tightening all the bolts we practiced driving the robot. The next item to add to the robot is the mineral grabber.

Signature : Shelby Geer

Date: Oct. 21, 2018



Signature : Shelby Geer

Date: Oct. 21, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: October 22, 2018

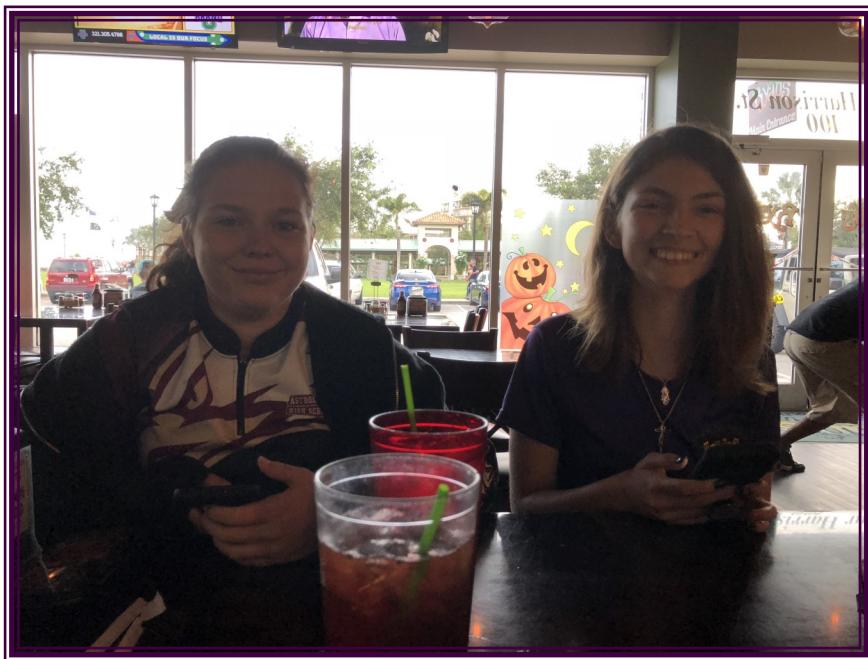
Purpose of the Activity:

We went out to dinner for some team bonding followed by attending the Plaid Pelican Team Meeting to talk about the upcoming Meet we are hosting together. The event theme is "Space Jam". Go Buggs Bunny!!!

Process:

Our outing brought us to Ryan's Pizzeria where we enjoyed a wonderful meal.

So before meeting with the Plaid Pelican team about the December Meet, we stopped for some dinner and team bonding. Yummy food....



Signature : Jessica Anderson

Date: Oct. 22, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 22, 2018

Process:

We discussed several options for the theme of our joint Meet. The final winner was "Space Jams" with an emphasis on Buggy Bunny and a jar of jam.....

We will be making Drivers and Coach badges with the characters. The team table tags will be ajar shape with the team numbers on the label space. The Plaid Pelican team will come up with a flyer to be distributed via email.

We are also planning a Food Drive for the Church's pantry as a Outreach project.

We are planning on a dinner afterwards to celebrate our success.



Plaid Pelican Meeting

Signature : Jessica Anderson

Date: Oct. 22, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

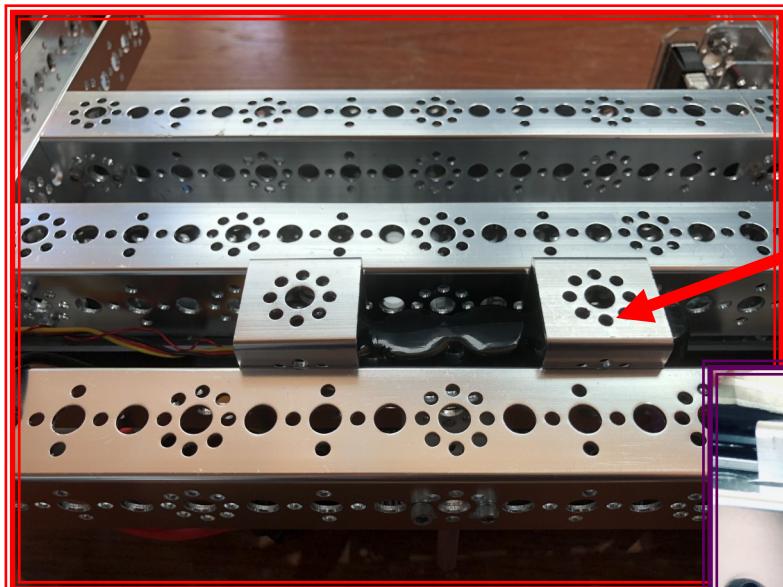
Date: October 28, 2018

Purpose of the Activity:

Adjustments to safe keep the battery Team Marker

Adding addition support for the plexiglass for the
bottom of the robot Phone Boxes

Process:



Added brackets to keep the battery in
when the robot was upside-down.

Battery is safe from falling out now.



Signature : Jessica Anderson

Date: Oct. 28, 2018

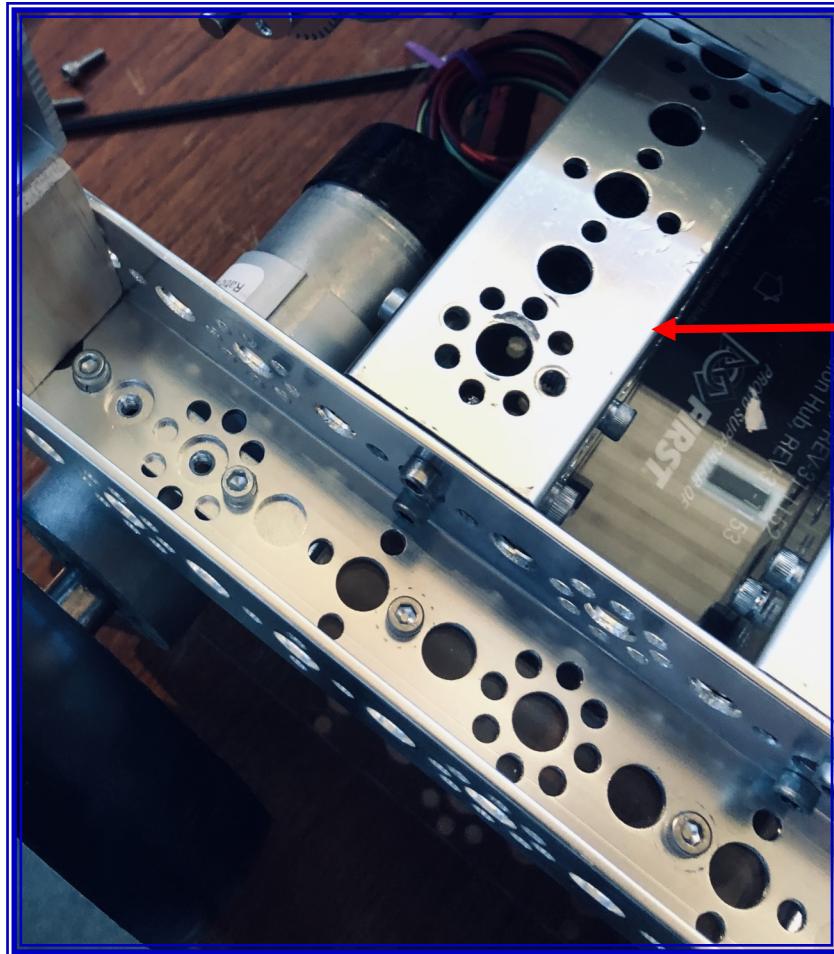
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

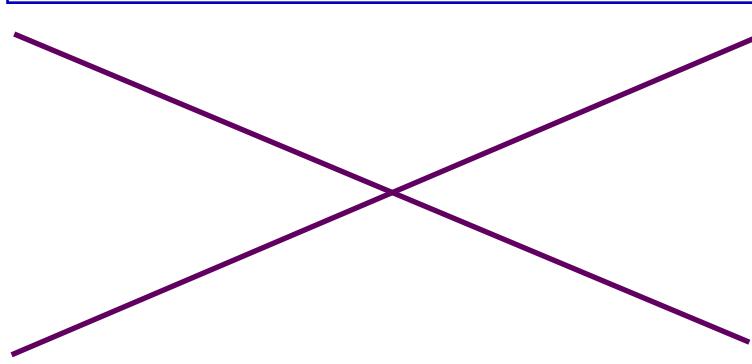
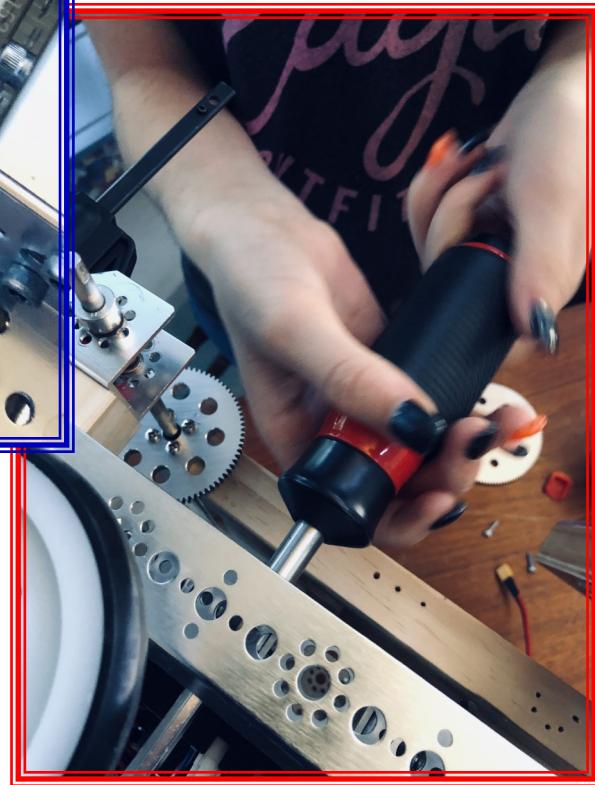
Engineering Activity Continued

Date: October 28, 2018

Process:



We need to move one of the support brackets so that we could put in the supports and motors for the lift mechanism holding our mineral grabber.



Signature : Jessica Anderson

Date: Oct. 28, 2018

Team 7341

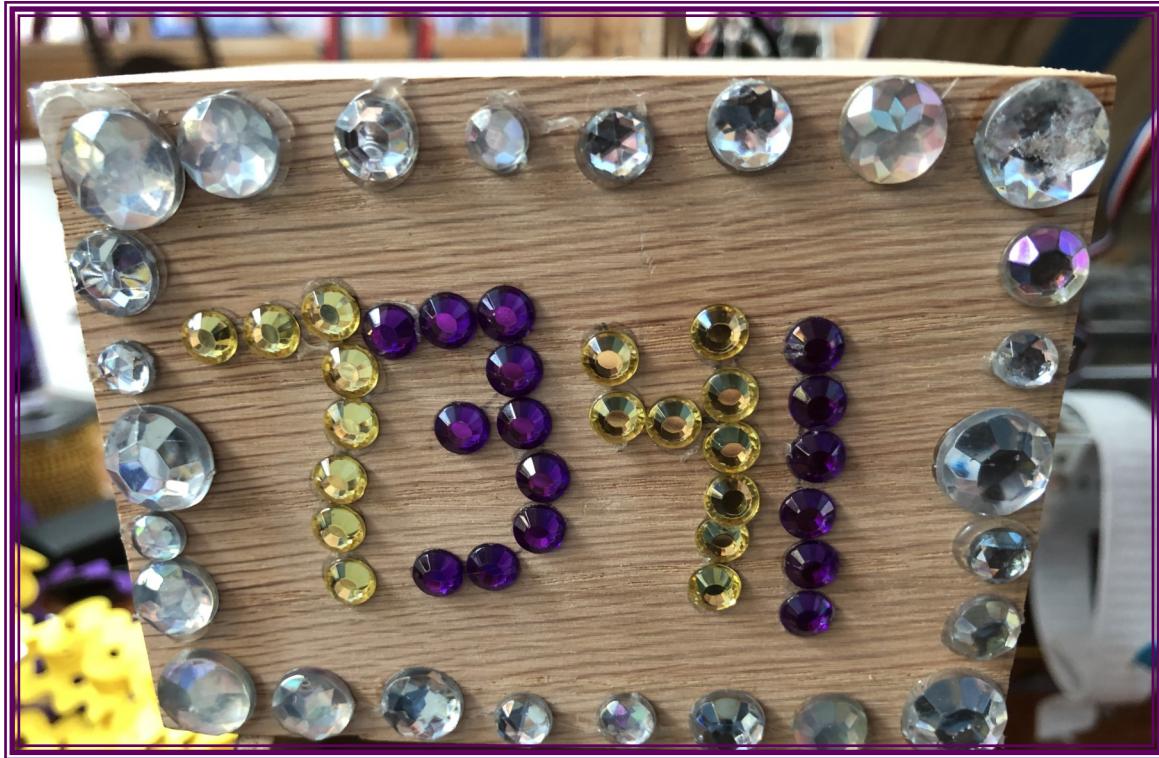
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:

We have completed our team marker, a Oak Block sporting bling for team numbers. We then refurbished the tape on the Driver Control phone box and moved one for the Robot phone boxes from Prince Charles as he only needed one in his current configuration.



Signature : Jessica Anderson

Date: Oct. 28, 2018

Team 7341

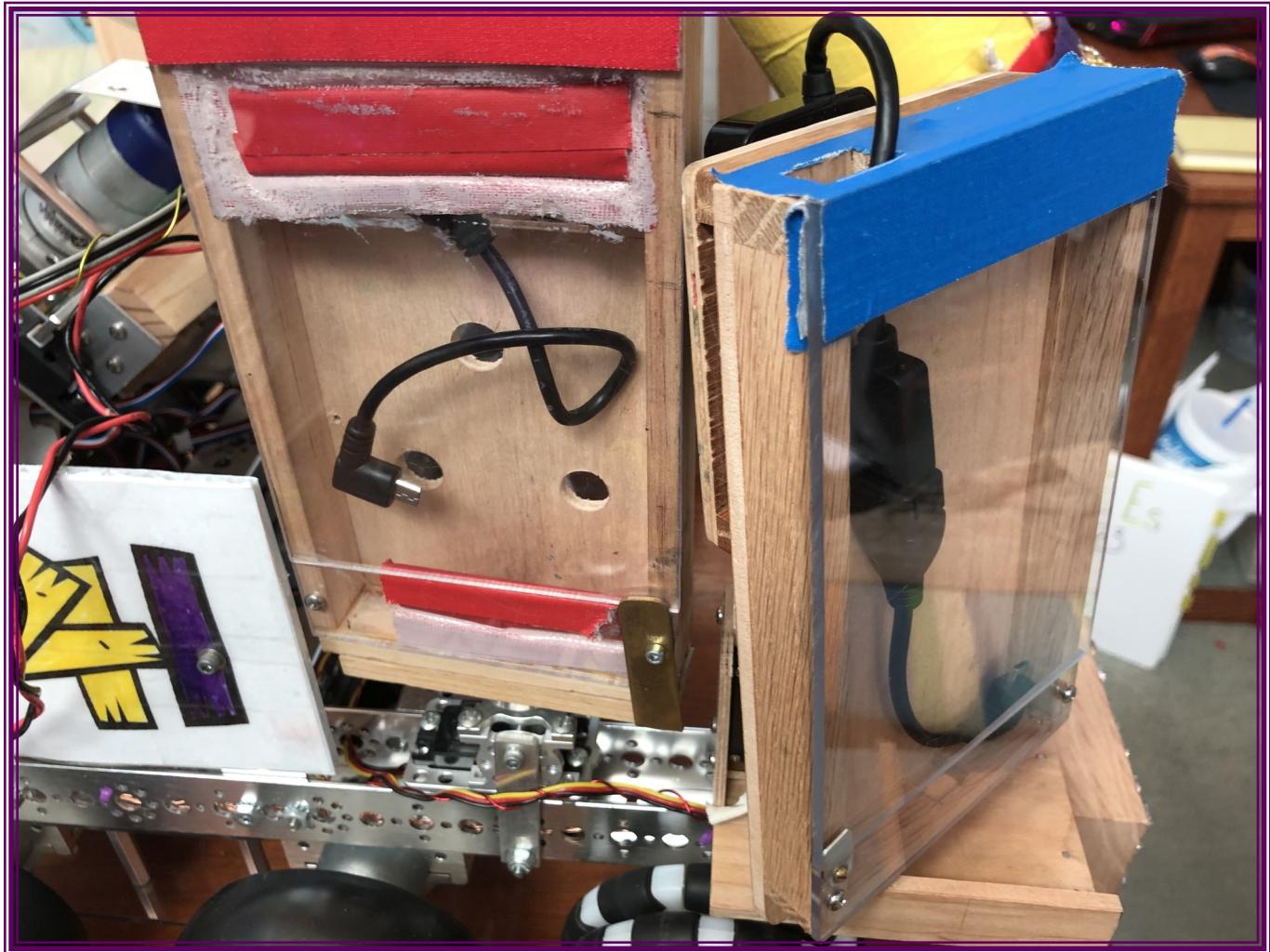
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:

We also added a new latching mechanism to the Robot phone so it would also hold up the phone when moving around.



Signature : Jessica Anderson

Date: Oct. 28, 2018

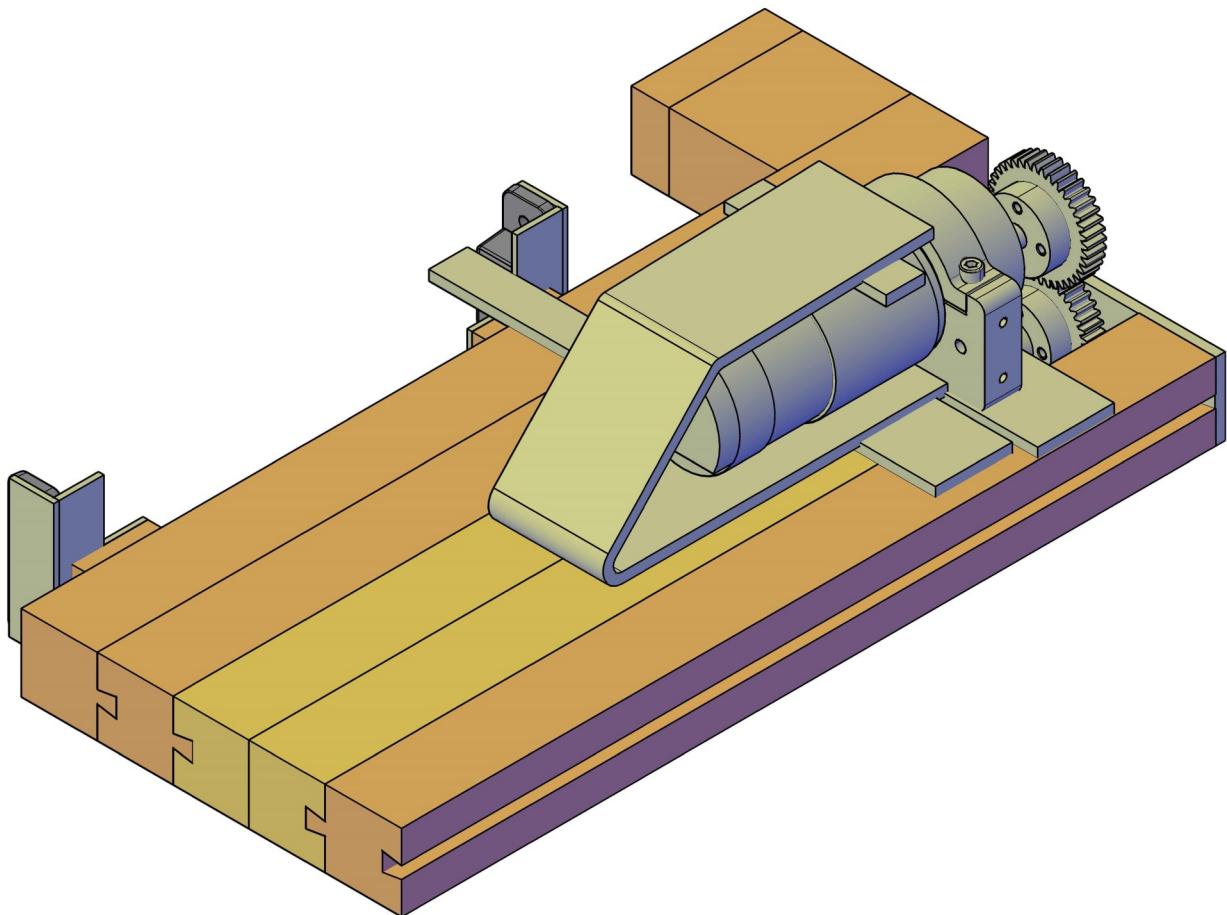
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:



Using the linear slide design part previously created we modified the center to use a linear actuator mechanism to raise and lower the center section of the linear slide.

Signature : Jessica Anderson

Date: Oct. 28, 2018

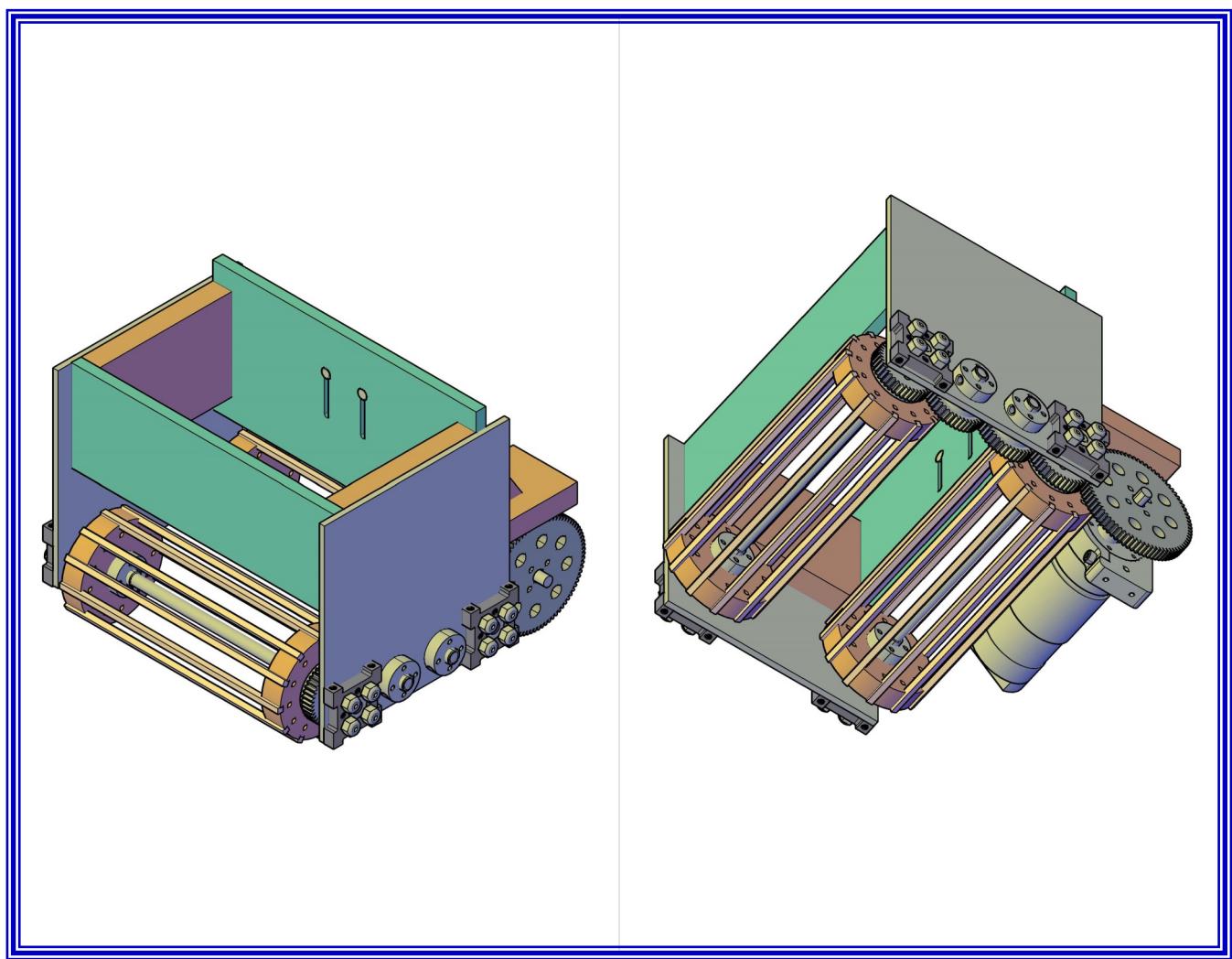
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:



This year we decided to try making a mechanism that used a motorized method in picking of the elements of the game. We have completed the first article to test at the first competition.

Signature : Jessica Anderson

Date: Oct. 28, 2018

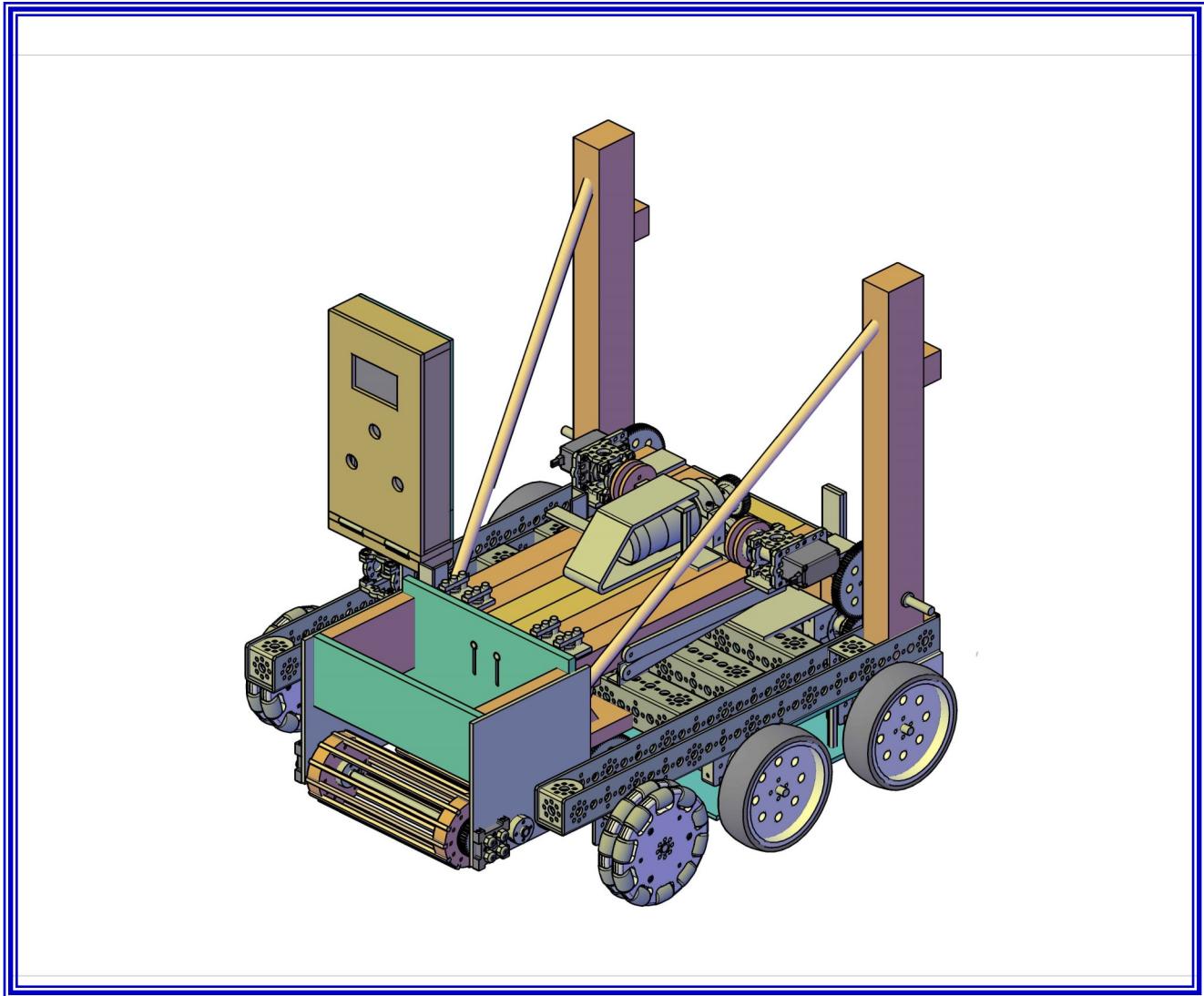
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:



This is the total design of the robot - first phase.

Signature : Jessica Anderson

Date: Oct. 28, 2018

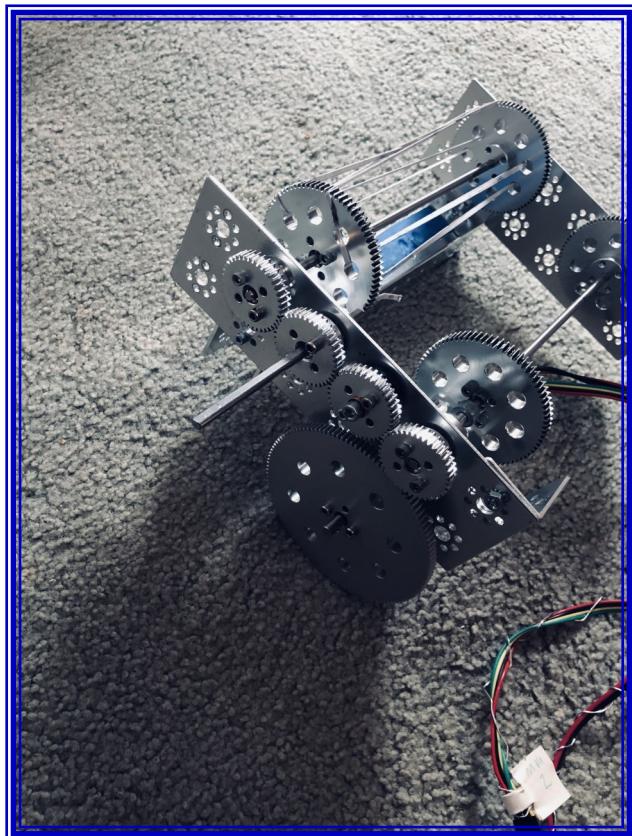
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: October 28, 2018

Process:



Here we have a picture of our prototype and our biggest concern is that the mechanism is too heavy.

After deciding that we could build the mechanism we wanted to calculate the work needed to move the end-effector and the calculation is as follows:

Work = weight of the object x gravity X distance traveled

Gravity = 9.8

Weight of the object = .0517 (Gold Mineral)

Reference is in meters

Distance traveled = .889

Work to pickup is .45042074J

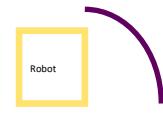
Work to lower is -.45042074J

Work to the top of the lander

Distance traveled = 1.27

Work to pickup is 6.434J

Work to pickup is -6.434J



Weight of the object = .0517 (Silver Mineral)

Distance traveled = .889

Work to pickup is .44940742J

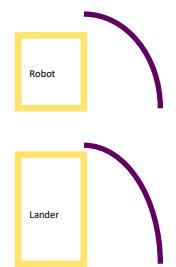
Work to lower is -.44940742J

Work to the top of the lander

Distance traveled = 1.27

Work to pickup is .6435J

Work to pickup is -.6435J



Signature : Jessica Anderson

Date: Oct. 28, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

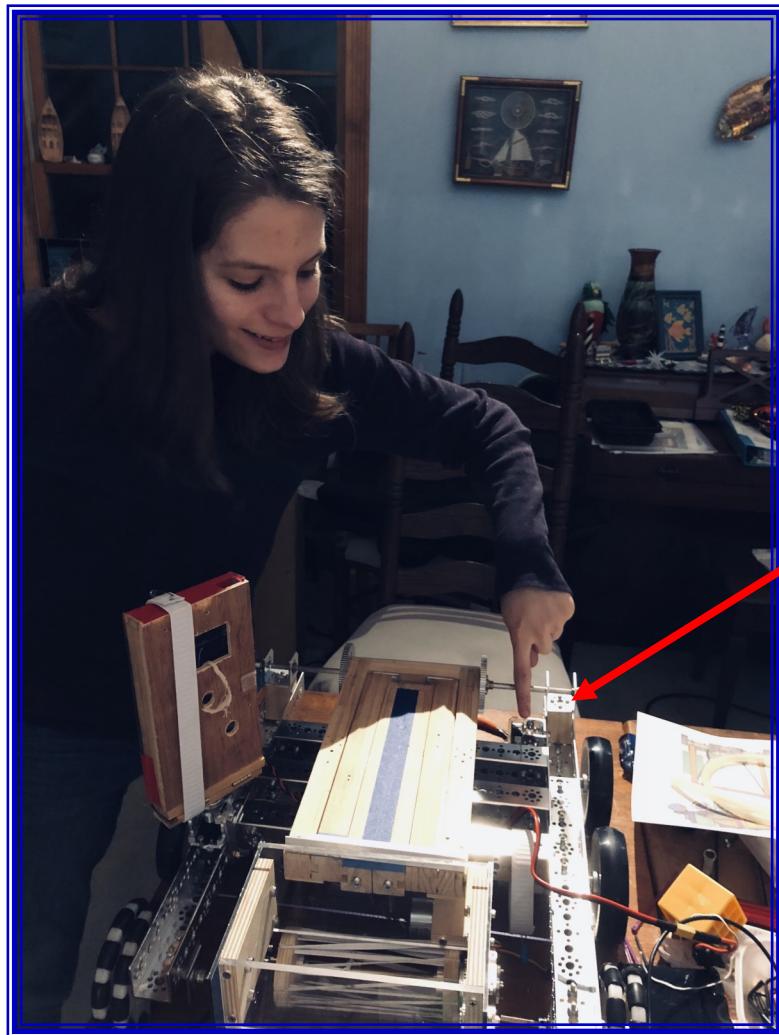
Engineering Activity

Date: October 29, 2018

Purpose of the Activity:

Alumni Emily Worthington stopped by and provided some insight to where to put our “power switch”.

Process:

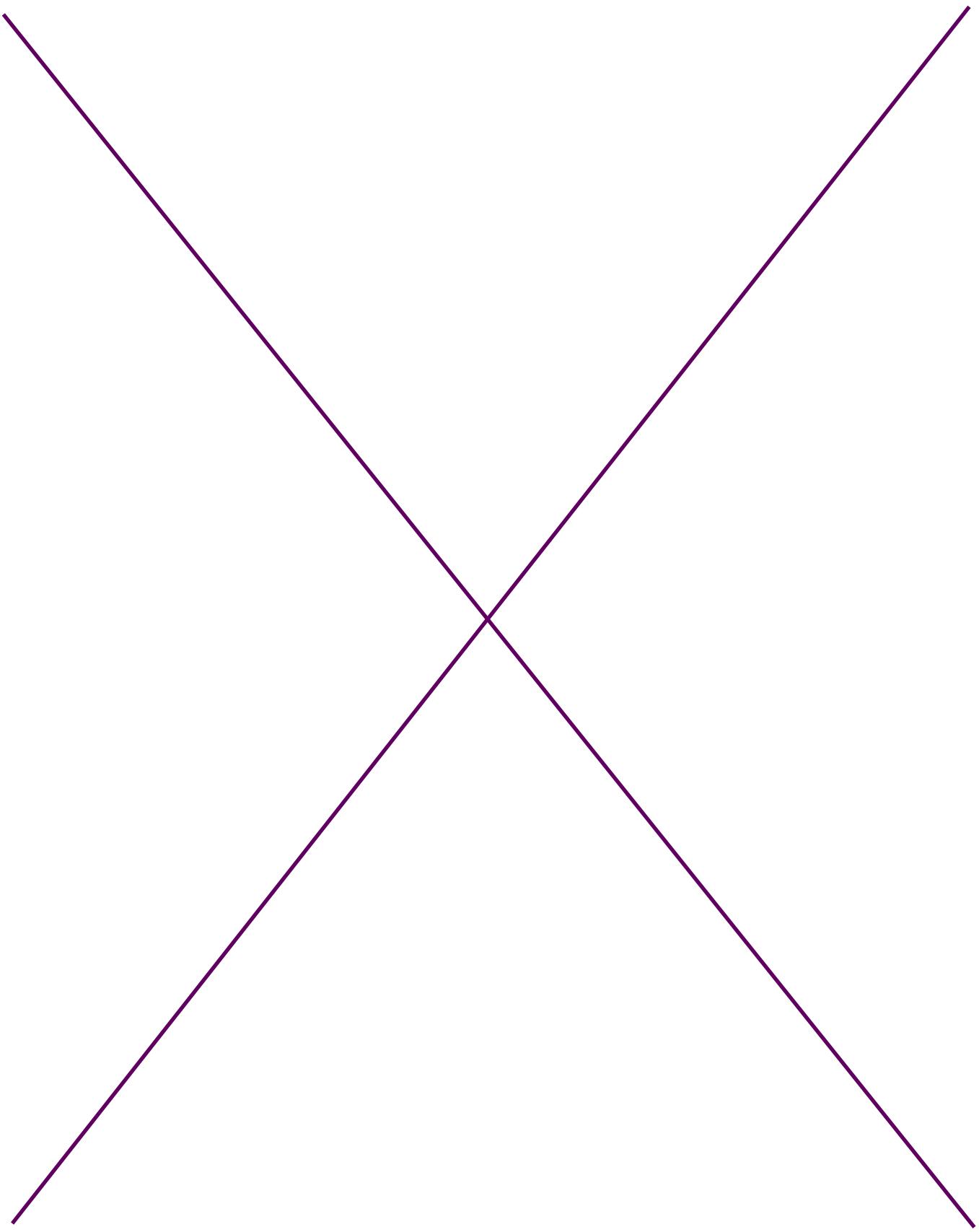


As an alumni Emily brings new and different insight to life and when she stopped by she offered a suggestion on where to place the power switch so the robot would not accidentally get powered off.... Thanks Emily!!!

We will be grateful when she moves closer and can visit more often.

Signature : Jessica Anderson

Date: Oct. 29, 2018



Signature : Jessica Anderson

Date: Oct. 29, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

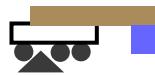
Date: November 4, 2018

Purpose of the Activity:

- | | |
|------------------------------|------------------------------|
| Adding the linear slide | Adding robot safety features |
| Attaching the grabber | Adding the flag holder |
| Adjusting the wheel position | |

Process:

Today we moved our middle wheels so that they were closer to the back wheels so we can get over the crater wall and not get stuck.



Before



After



After adjusting the wheels we verified the construction of the center section of the lift with the drawing and found that things should line up. New



Signature: Shelby Greer

Date: Nov. 4, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

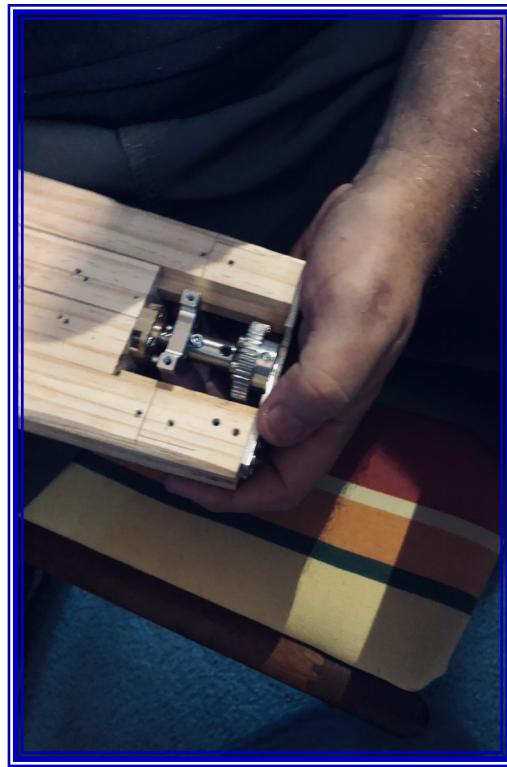
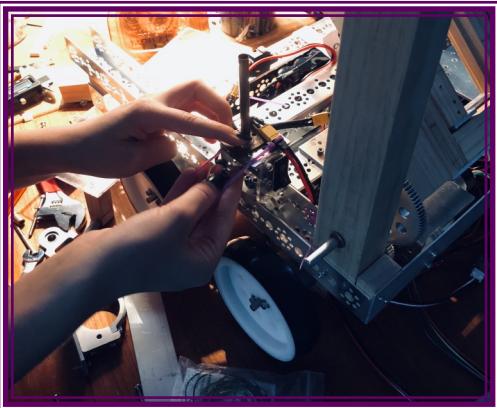
Engineering Activity Continued

Date: November 4, 2018

Process:

wheels for the second level lift were added and tested to make sure that they worked as needed.

We also added our special flag holder on top of the power switch.



Signature : Shelby Greer

Date: Nov. 4, 2018

Team 7341

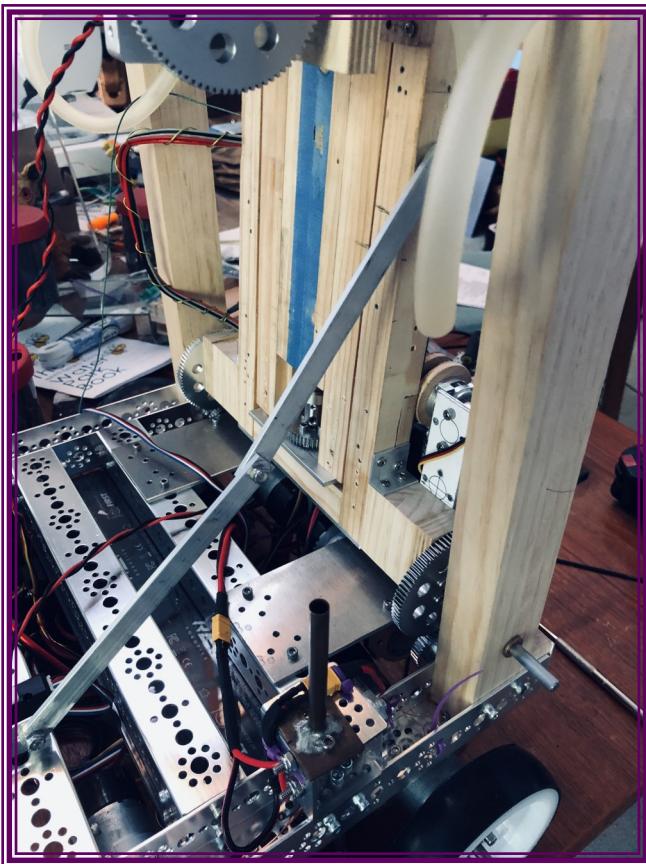
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

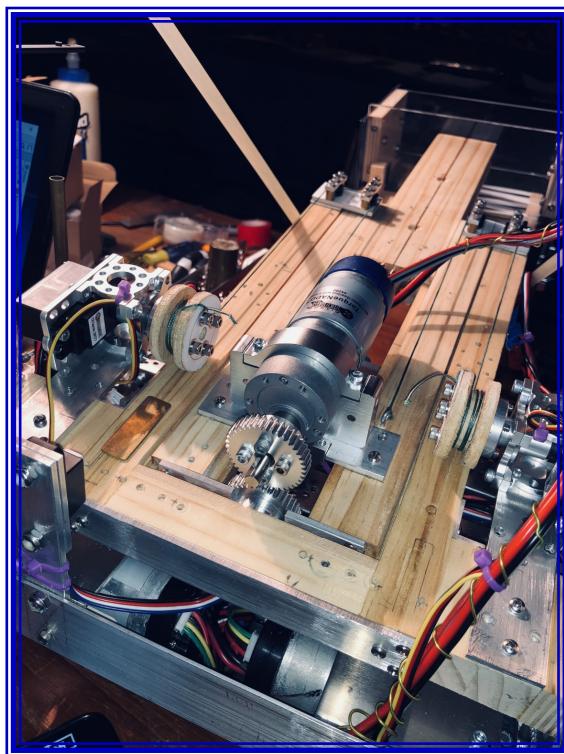
Date: November 4, 2018

Process:

Below and at the bottom left are the images showing how Shelby and Mr. Louis verified that the center section of the lift was correct. After the verification was completed the lift was tested for it's functionality.



We added an extra safety feature so that when we lifted the grabber it would not want to go to far



Signature : Shelby Greer

Date: Nov. 4, 2018

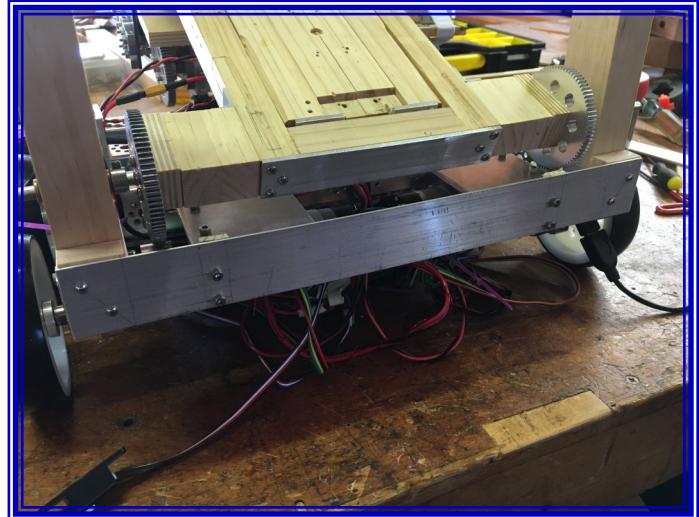
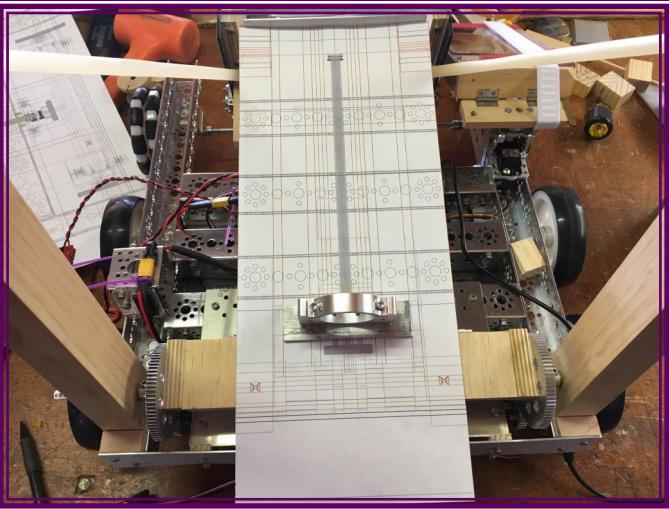
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

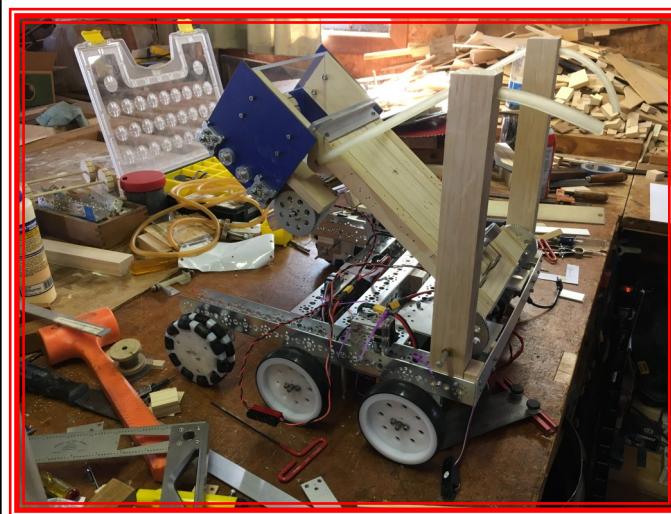
Engineering Activity Continued

Date: November 4, 2018

Process:



This is the build up stages for the center of the linear slide.



Signature: Shelby Greer

Date: Nov. 4, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

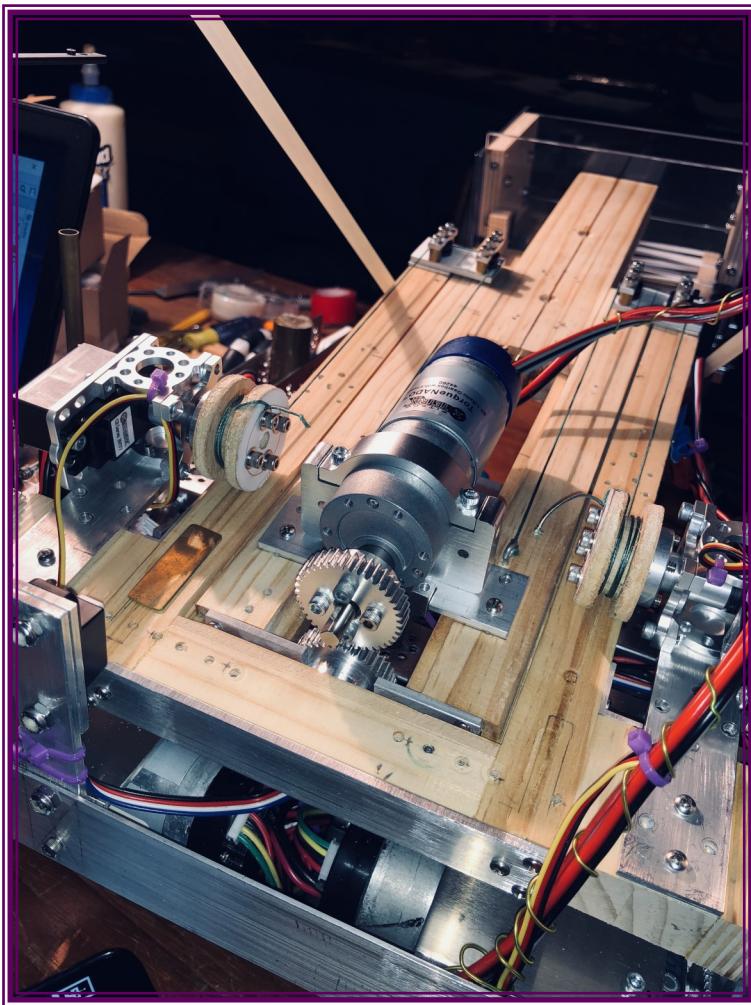
Engineering Activity Continued

Date: November 4, 2018

Process:

past the 92 degrees. This arm helps insure that the touch sensor being used to stop the lift motor stops when we have reached our top position. This will also help us from having to much force that we tip over backwards.

Below is a photo of the backside of the lift mechanism. The motor is a TETRIX® MAX TorqueNADO® Motor from Pitsco which we won from a national drawing... It was our lucky day! Thank you Pitsco!



After driving the robot around we found that our robot phone did not stay upright so we add a little elastic to keep it in place. We also



Signature: Shelby Greer

Date: Nov. 4, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

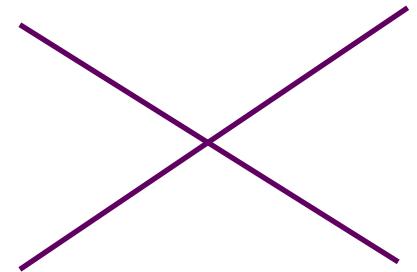
Engineering Activity Continued

Date: November 4, 2018

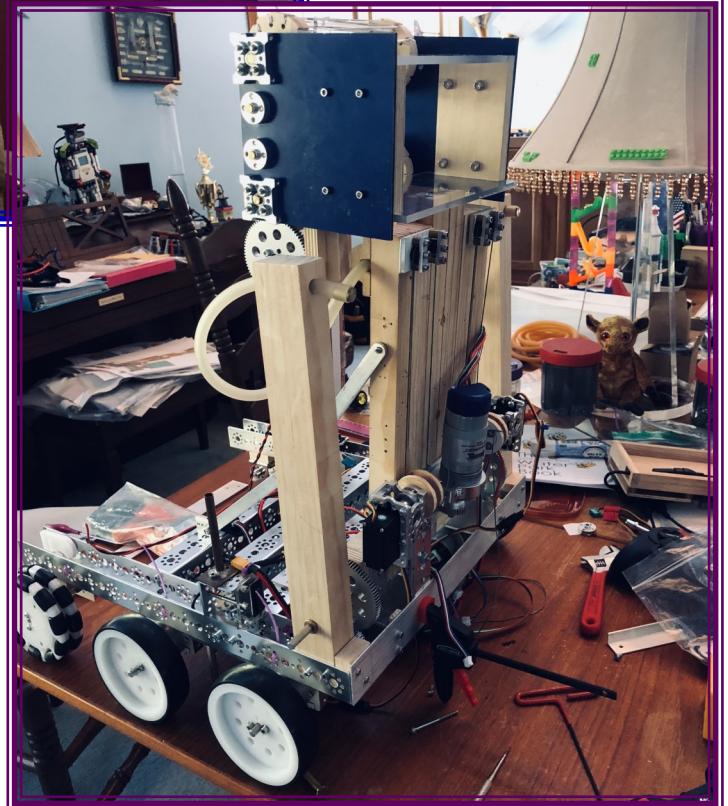
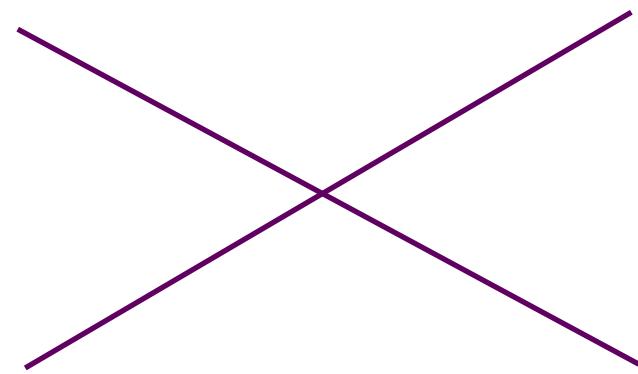
Process:



It looks like the end effector reaches the top of the lander so we can put gold and silver mineral in the pockets.



We just have a few minor adjustment to make on the robot and it is ready for our first meet of the season.



Signature : Shelby Greer

Date: NOV. 4, 2018

Team 7341

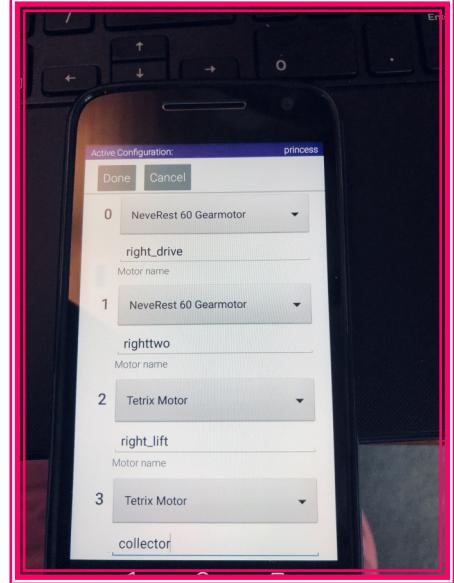
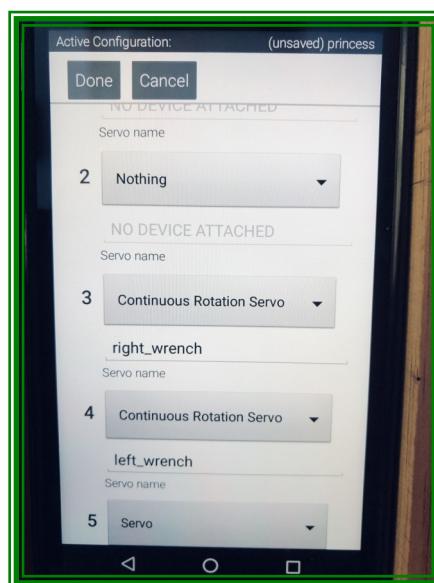
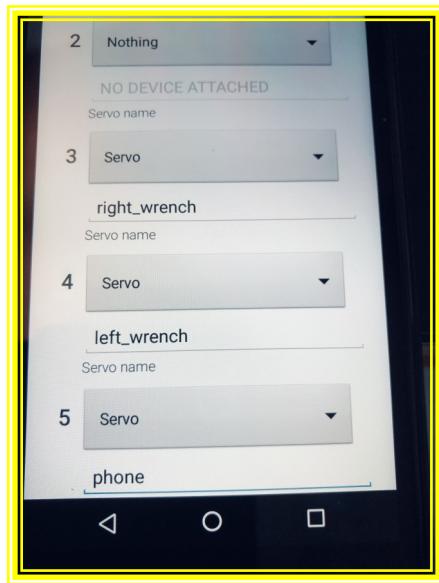
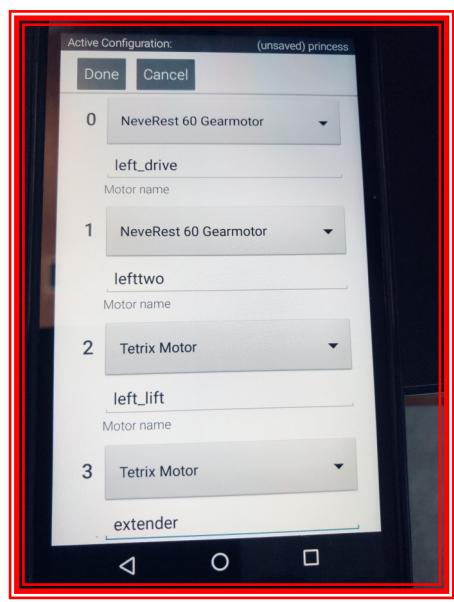
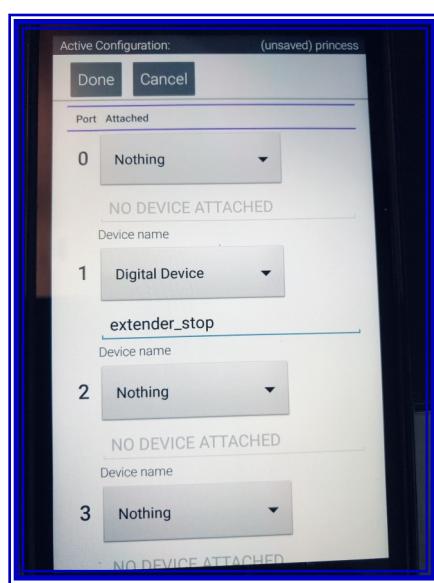
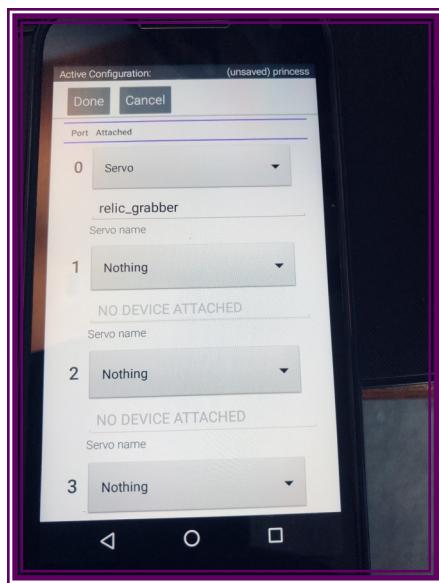
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 4, 2018

Process:

screen captured our robot's configuration incase the phone resets.



Signature : Shelby Greer

Date: Nov. 4, 2018

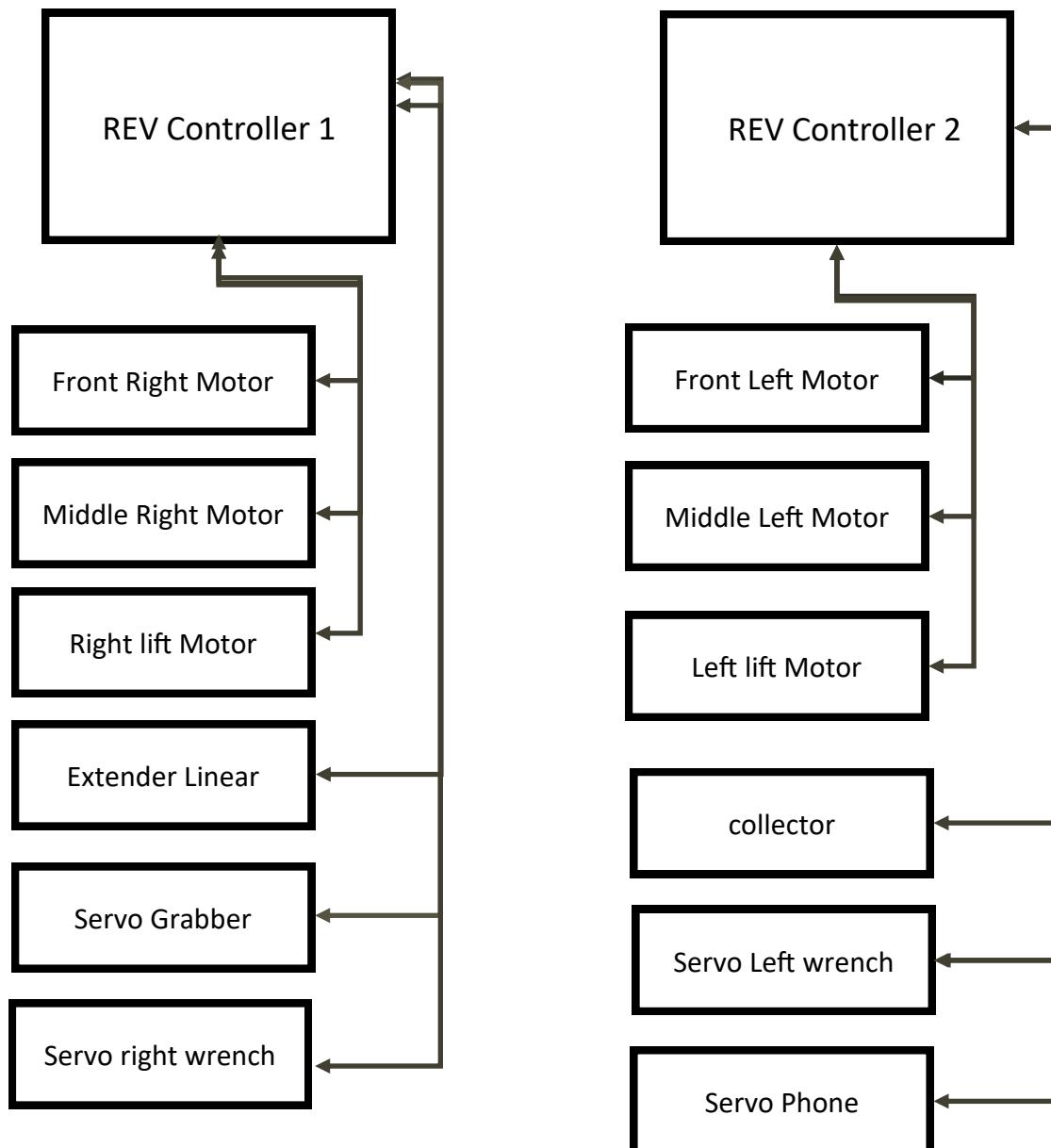
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 4, 2018

Process:



Signature: Shelby Greer

Date: Nov. 4, 2018

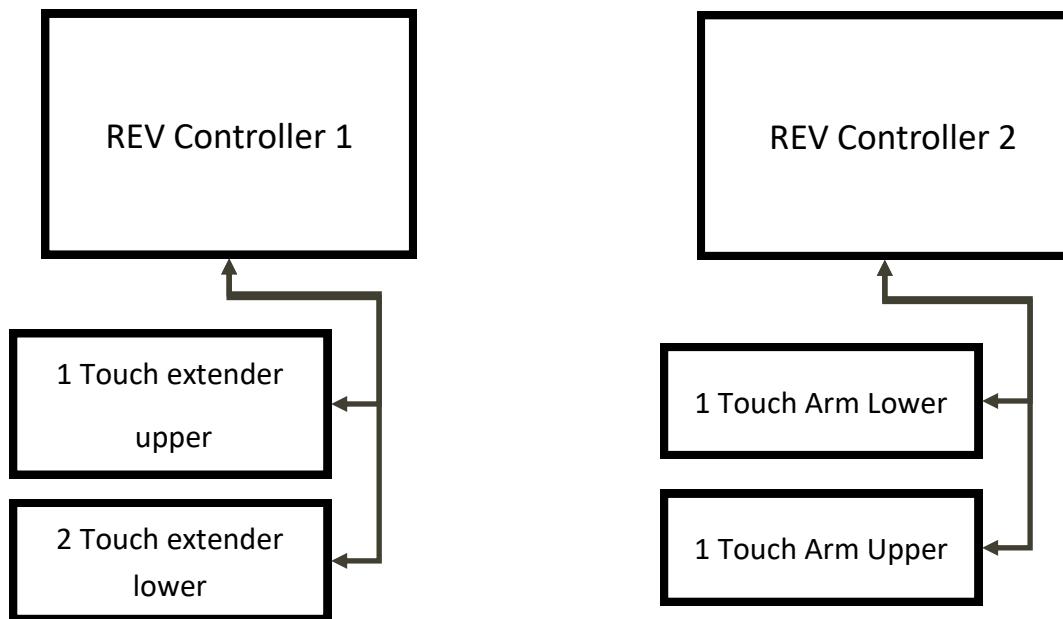
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

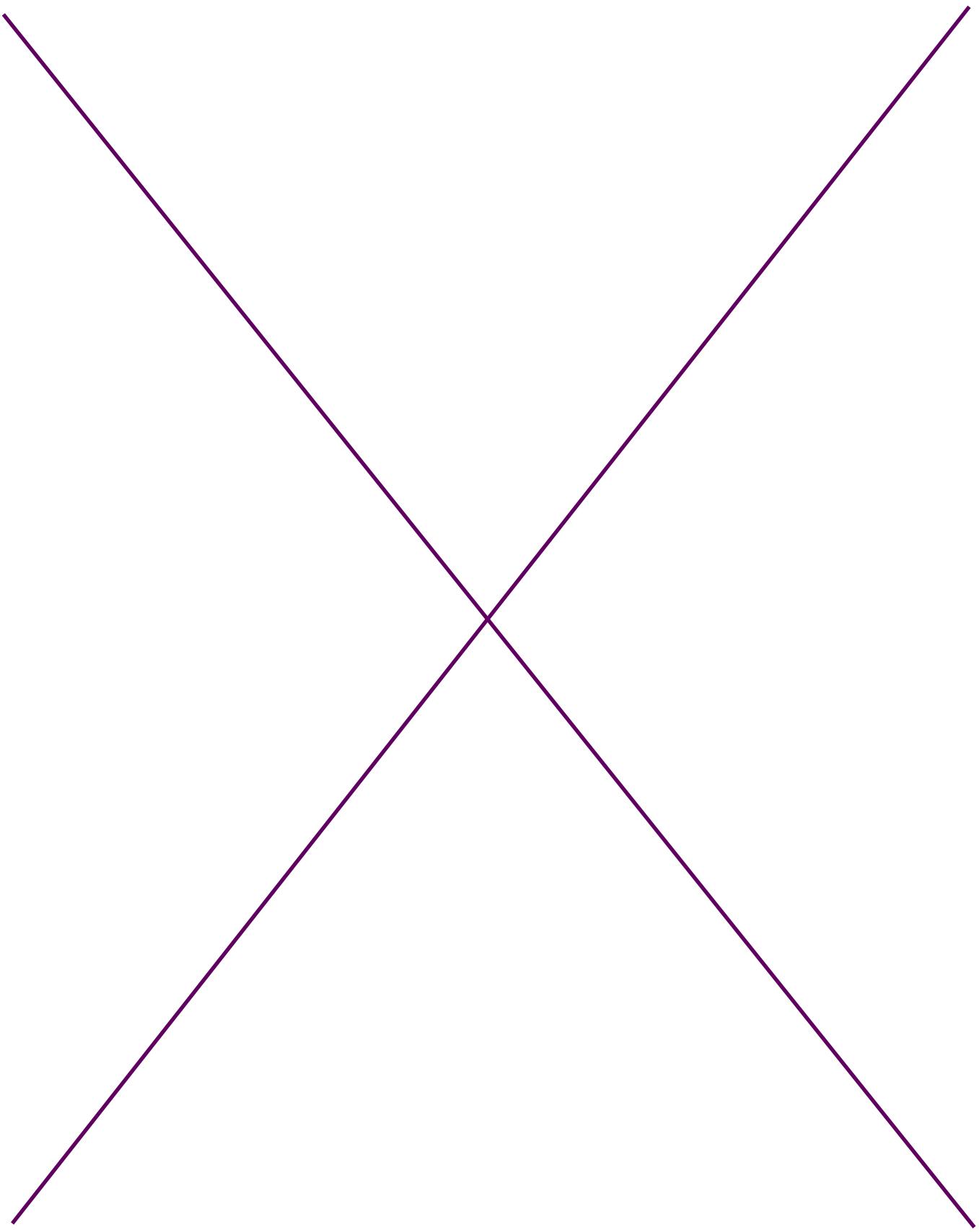
Date: November 4, 2018

Process:



Signature : Shelby Greer

Date: Nov. 4, 2018



Signature : Shelby Geer

Date: Nov. 4, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: November 9-10, 2018

Purpose of the Activity:

Preparing for our first meet of the season	Packing
Practice driving	Sleepover
Downloading programs	

Process:

We have made final preparation for the meet by charging the batteries and driving collecting the minerals. We did not run the autonomous, so we will skip it this time around.

Following are the programs prepared for the robot:

Hardware Definition Function

```
package org.firstinspires.ftc.Team7341;
```

```
import com.qualcomm.hardware.bosch.BNO055IMU;
import com.qualcomm.hardware.bosch.JustLoggingAccelerationIntegrator;
import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cRangeSensor;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.CRServo;
import com.qualcomm.robotcore.hardware.ColorSensor;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.DigitalChannel;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.hardware.I2cAddr;
import com.qualcomm.robotcore.hardware.Servo;
import com.qualcomm.robotcore.hardware.TouchSensor;
import com.qualcomm.robotcore.util.ElapsedTime;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;

import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotorSimple.Direction.REVERSE;
import static java.lang.Thread.currentThread;
import static java.lang.Thread.sleep;

/**
 * This is NOT an opmode.
 * <p>
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
 * <p>
 * This hardware class assumes the following device names have been configured on the robot:
 * Note: All names are lower case and some have single spaces between words.
 */
public class PrinceCharlesBaDazzle {

    // setup for calculation of the how far to move
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
static final double P_TURN_COEFF      = 0.1; // Larger is more responsive, but also less stable
static final double HEADING_THRESHOLD = 1 ;   // As tight as we can make it with an integer
gyro

static final double COUNTS_PER_MOTOR_REV = 1680 ; //eg: AndyMark Motor Encoder
static final double DRIVE_GEAR_REDUCTION = 1; // This is < 1.0 if geared UP
static final double WHEEL_DIAMETER_INCHES = 4.0 ; // For figuring circumference
static final double GEAR_DIAMETER_INCHES = 1.0 ; // For figuring circumference

static final double COUNTS_PER_INCH      = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
(WHEEL_DIAMETER_INCHES * 3.1415);
static final double LIFT_COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
(GEAR_DIAMETER_INCHES * 3.1415);

static final double DRIVE_SPEED        = 0.5;
static final double DRIVE_SPEED1       = 0.9;
static final double STONE_DRIVE_SPEED = 0.2;
static final double TURN_SPEED         = 0.3 ;
static final double TOUCH_SPEED        = 0.1;
static final double LIFT_SPEED         = 0.2;
static final double STOP_SPEED         = 0;

int floor_color_value;
int position_option = 0;
int position_side = 0;
int turn_option = 0;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
int path_option = 0;
int find_target = -1;
int hanging = -1;
int gold_position = 0;
int count = 0;
// Driver motors
DcMotor rightdrive;
DcMotor lefdrive;
DcMotor righttwo;
DcMotor lefttwo;
float right = 0;
float left = 0;

// Distance control

ModernRoboticsI2cRangeSensor distance;

//motors to lift and lower
DcMotor right_liftdrive;
DcMotor left_liftdrive;
float lift_power = 0;

// move the collector in/out up/down
DcMotor extenderdrive;
// pick up the minerals
DcMotor collectordrive;
float extender_power = 0;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
float collector_power = 0;

// open position
double right_position = 0;
double left_position = .5;

// Side phone
Servo phone;
double phone_position = 0.26;
CRServo left_wrench;
CRServo right_wrench;
double wrench_power = 0;

DigitalChannel lower_stop;
DigitalChannel upper_stop;

// Our sensors, motors, and other devices go here, along with other long term state
BNO055IMU imu;

// State used for updating telemetry
Orientation angles;
Acceleration gravity;

/* local OpMode members. */
HardwareMap hardwareMap = null;
private ElapsedTime period = new ElapsedTime();

// Private Members
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
private LinearOpMode myOpMode;

/* Constructor */
public PrinceCharlesBaDazzle() {

}

/* Initialize standard Hardware interfaces */
public void init(HardwareMap ahwMap, int option) {
    // Save reference to Hardware map
    hardwareMap = ahwMap;

    /*
     * Use the hardwareMap to get the dc motors and servos by name.
     * Note that the names of the devices must match the names used
     * when you configured your robot and created the configuration file.
     */
    // if option is one define hardware... otherwise put it start position..
    if (option == 1) {
        // start of drive train definitions
        rightdrive = hardwareMap.dcMotor.get("right_drive");
        leftdrive = hardwareMap.dcMotor.get("left_drive");
        leftdrive.setDirection(REVERSE);
        righttwo = hardwareMap.dcMotor.get("righttwo");
        lefttwo = hardwareMap.dcMotor.get("lefttwo");
        lefttwo.setDirection(REVERSE);
        // end of drive train definitions
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
right_liftdrive = hardwareMap.dcMotor.get("right_lift");
right_liftdrive.setDirection(REVERSE);
left_liftdrive = hardwareMap.dcMotor.get("left_lift");

collectordrive = hardwareMap.dcMotor.get("collector");

extenderdrive = hardwareMap.dcMotor.get("extender");
extenderdrive.setDirection(REVERSE);
// stop the lifter when down flat
lower_stop = hardwareMap.get(DigitalChannel.class, "lower_stop");
upper_stop = hardwareMap.get(DigitalChannel.class, "upper_stop");

// set the digital channel to input.
lower_stop.setMode(DigitalChannel.Mode.INPUT);
upper_stop.setMode(DigitalChannel.Mode.INPUT);

phone = hardwareMap.servo.get("phone");
left_wrench = hardwareMap.get(CRServo.class, "left_wrench");
right_wrench = hardwareMap.get(CRServo.class, "right_wrench");

// set up the guidance hardware

// Set up the parameters with which we will use our IMU. Note that integration
// algorithm here just reports accelerations to the logcat log; it doesn't actually
// provide positional information.
BNO055IMU.Parameters parameters = new BNO055IMU.Parameters();
parameters.angleUnit      = BNO055IMU.AngleUnit.DEGREES;
parameters.accelUnit      = BNO055IMU.AccelUnit.METERS_PERSEC_PERSEC;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
parameters.calibrationDataFile = "BNO055IMUCalibration.json"; // see the calibration sample op-
mode
parameters.loggingEnabled = true;
parameters.loggingTag = "IMU";
parameters.accelerationIntegrationAlgorithm = new JustLoggingAccelerationIntegrator();

// Retrieve and initialize the IMU. We expect the IMU to be attached to an I2C port
// on a Core Device Interface Module, configured to be a sensor of type "AdaFruit IMU",
// and named "imu".
imu = hardwareMap.get(BNO055IMU.class, "imu");
imu.initialize(parameters);

} else {

    // set the phones
    phone_position = .26;
    phone.setPosition(phone_position);

}
}

public void wait(int sec) {
    for (int i = 0; i < 2 * sec; i++) {
        try {
            sleep(500);
        } catch (InterruptedException e) {
            currentThread().interrupt();
            break;
        }
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
        }
    }
}

/**
*
* waitForTick implements a periodic delay. However, this acts like a metronome with a regular
* periodic tick. This is used to compensate for varying processing times for each cycle.
* The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
*
* @param periodMs Length of wait cycle in mSec.
*/
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long) period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            sleep(remaining);
        } catch (InterruptedException e) {
            currentThread().interrupt();
        }
    }

    // Reset the cycle clock for the next pass.
    period.reset();
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

The Teleop Program (ChickenNoodleFrenchFry) which allows the robot to be controlled using the gamepads.

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

The function of this program is to run teleop.

Neither the name of Qualcomm Technologies Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackables;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotor.RunMode;
import static com.qualcomm.robotcore.util.Range.clip;
import static java.lang.Math.abs;
import static java.lang.String.format;
import static org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer.CameraDirection.BACK;

@TeleOp(name = "FF: ChickenNoodleFrenchFry", group = "FF")
//@Disabled
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
public class ChickenNoodleFrenchFry extends LinearOpMode {  
    /* Declare OpMode members. */  
  
    private ElapsedTime runtime = new ElapsedTime();  
  
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle();  
    private static final String VUFORIA_KEY = "AW/Sw13////  
    AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-  
    Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/  
    MZz7Zvp35rYD6IavfkgCMZ0DVAXHBv4J5LlrGIVXYfhhs1NkITGPDqVRW2aBmKLwctHzaztzycau3g//  
    QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/  
    rEsoZMaPnPfkyGcE-  
    Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/  
    FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here  
    /**  
     * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
     * localization engine.  
     */  
    private VuforiaLocalizer vuforia;  
    /**  
     * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
     * Detection engine.  
     */  
    private TFObjectDetector tfod;  
  
    // Leave argument list empty if you want to disable the camera monitor view.  
    TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();  
    private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";
private boolean targetVisible = false;
// private boolean targetVisible = false;Valid choices are: BACK or FRONT

private static final VuforiaLocalizer.CameraDirection CAMERA_CHOICE = BACK;
// Variables to be used for later
private VuforiaLocalizer vuforiaLocalizer;
private VuforiaLocalizer.Parameters parameters;
private VuforiaTrackables visionTargets;
@Override
public void runOpMode() throws InterruptedException {

    // data for moving the phone
    boolean phone_locked;
    // set to be at the bottom
    int extender_position1 = 0;
    float extender_power1 = 0;
    int last_extender_position = -999;
    int extender_position = 2;
    int extender_direction = 0;
    int attop = 0;
    int atbottom = 1;
    int location = 0;
    int phonecyclecount;
    phone_locked = false;
    phonecyclecount = 0;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// set phone data
// State used for updating telemetry
// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Setting Up ChickenNoodleFrenchFry"); //
telemetry.update();

/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */

/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

// end of the front-end of the robots definitions

telemetry.addData("Text", "Waiting to start ChickenNoodleFrenchFry");
telemetry.update();
// this is to initialize the camera for target viewing
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// initVufor();    telemetry.update();
/*
if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3","setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3","back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}
*/
waitForStart();

// put servos in start position

robot.init(hardwareMap, 2);
robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
//move phone out of the way
robot.phone_position = .84;
robot.phone.setPosition(robot.phone_position);

robot.lock_power = .5;
robot.lock.setPosition(robot.lock_power);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// Set up our telemetry dashboard
composeTelemetry();
// setting up timer
runtime.reset();
// while the op mode is active, loop and read the RGB data.
// Note we use opModelsActive() as our loop condition because it is an interruptible method.
while (opModelsActive()) {

/*
 * Gamepad 1 controls the motors via the left/right stick
 */

// this is for the motor control function
// forward is negative power value
// backwards is positive power value
int drive_mode = 2;
if (gamepad1.right_stick_y != 0 || gamepad1.left_stick_y != 0 ||
    gamepad1.right_stick_x != 0 || gamepad1.left_stick_x != 0) {
    if (gamepad1.right_stick_y <= 0) {
        robot.right = ((-gamepad1.right_stick_y - (abs(gamepad1.right_stick_x))) / (float) .5);
    } else {
        robot.right = ((-gamepad1.right_stick_y + abs(gamepad1.right_stick_x))) / (float) .5;
    }
    if (gamepad1.left_stick_y <= 0) {
        robot.left = ((-gamepad1.left_stick_y - (abs(gamepad1.left_stick_x))) / (float) .5);
    } else {
        robot.left = ((-gamepad1.left_stick_y + abs(gamepad1.left_stick_x))) / (float) .5;
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// clip the right/left values so that the values never exceed +/- 1
robot.right = clip(robot.right, -1, 1);
robot.left = clip(robot.left, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
robot.right = (float) scaleInput(robot.right);
robot.left = (float) scaleInput(robot.left);

// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
} else {
    robot.right = 0;
    robot.left = 0;
// clip the right/left values so that the values never exceed +/- 1
robot.right = clip(robot.right, -1, 1);
robot.left = clip(robot.left, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
robot.right = (float) scaleInput(robot.right);
robot.left = (float) scaleInput(robot.left);

// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// Gamepad 2

// Lift (Raise and Lower) the end effector
// You should not go much past 90degrees
// up is negative value on the controller
// down is positive value on the controller
// lower_stop is to make sure the motor stop when touching the floor
// lift_top is to stop when just past being vertical
// if the digital channel returns true it's HIGH and the button is unpressed.

if ((gamepad2.left_stick_y < 0 && robot.upper_stop.getState() == true ) ||
    (gamepad2.left_stick_y > 0 && robot.lower_stop.getState() == true )) {
    // set lift speed to a constant value use the negative
    // negate the power value to get the motors going in the direction desired

    if (gamepad2.left_stick_y < -.75) gamepad2.left_stick_y =(float) -.75;
    if (gamepad2.left_stick_y > .75) gamepad2.left_stick_y = (float) .75;

    robot.lift_power = -gamepad2.left_stick_y;

    // clip the right/left values so that the values never exceed +/- 1
    robot.lift_power = clip(robot.lift_power, -1, 1);

    // scale the joystick value to make it easier to control
    // the robot more precisely at slower speeds.

    robot.lift_power = (float) scaleInput(robot.lift_power);

    robot.right_liftdrive.setPower(robot.lift_power);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.left_liftdrive.setPower(robot.lift_power);

} else {

    // stop the lift motor

    robot.lift_power = 0;
    robot.lift_power = (float) scaleInput(robot.lift_power);
    robot.right_liftdrive.setPower(robot.lift_power);
    robot.left_liftdrive.setPower(robot.lift_power);

}

// raise the mineral grabber up/down
// up is negative power value
// going extender position = 2 we are at the bottom
extender_position1 = robot.extenderdrive.getCurrentPosition();
if (extender_position1 > last_extender_position) extender_direction = 2;
if (extender_position1 < last_extender_position) extender_direction = 1;

if ((gamepad2.right_stick_y < 0 )
|| (gamepad2.right_stick_y > 0)) {

    last_extender_position = extender_position1;

    // speed to a constant value

    robot.extender_power = gamepad2.right_stick_y;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// clip the right/left values so that the values never exceed +/- 1
extender_power1 = clip(robot.extender_power, -1, 1);
robot.extender_power = extender_power1;

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.

if (!robot.extender_lower.getState() && extender_direction == 1 && atbottom == 0) {
    attop = 1;
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
    location = robot.extenderdrive.getCurrentPosition();
}

if (!robot.extender_lower.getState() && extender_direction == 2 && attop == 0){
    atbottom = 1;
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
    location = robot.extenderdrive.getCurrentPosition();
}

if (robot.extender_lower.getState()){
    atbottom = 0;
    attop = 0;
}

if (!robot.extender_lower.getState() && extender_direction == 1 && robot.extender_power < 0 /* && attop == 1 */ ) robot.extender_power = 0;
if (!robot.extender_lower.getState() && extender_direction == 2 && robot.extender_power > 0 /* && atbottom == 1 */ ) robot.extender_power = 0;
// over ride because you are at the top or bottom and you need to move
if (!robot.extender_lower.getState() && atbottom == 1 && extender_power1 < 0) ro-
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
bot.extender_power = extender_power1;
    if (!robot.extender_lower.getState() && attop == 1 && extender_power1 > 0) ro-
bot.extender_power = extender_power1;

    robot.extender_power = (float) scaleInput(robot.extender_power);
    robot.extenderdrive.setPower(robot.extender_power);

} else {

    // stop the extension motor

    robot.extender_power = 0;
    robot.extender_power = (float) scaleInput(robot.extender_power);
    robot.extenderdrive.setPower(robot.extender_power);

}

// wrench up/down
if (gamepad2.right_bumper ) {
    //going up
    robot.wrench_power = .75;
    robot.left_wrench.setPower(robot.wrench_power);
    robot.right_wrench.setPower(-robot.wrench_power);
} else if (gamepad2.right_trigger != 0 ) {
    //going down
    robot.wrench_power = .75;
    robot.left_wrench.setPower(-robot.wrench_power);
    robot.right_wrench.setPower(robot.wrench_power);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
 } else {  
 //stop wrench  
 robot.wrench_power = 0;  
 robot.left_wrench.setPower(robot.wrench_power);  
 robot.right_wrench.setPower(robot.wrench_power);  
 }  
  
//pick the minerals up  
if (gamepad2.b) {  
 robot.collector_power = 1;  
 robot.collector_power = (float) scaleInput(robot.collector_power);  
 robot.collectordrive.setPower(robot.collector_power);  
}  
  
if (gamepad2.x) {  
 robot.collector_power = 0;  
 robot.collector_power = (float) scaleInput(robot.collector_power);  
 robot.collectordrive.setPower(robot.collector_power);  
}  
  
//spit the minerals out  
if (gamepad2.y) {  
 robot.collector_power = -1;  
 robot.collector_power = (float) scaleInput(robot.collector_power);  
 robot.collectordrive.setPower(robot.collector_power);  
}  
  
if (gamepad2.a) {  
 robot.collector_power = 0;  
 robot.collector_power = (float) scaleInput(robot.collector_power);  
 robot.collectordrive.setPower(robot.collector_power);  
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

}

// right bumper move out the phone
// left bumper move in the phone

```
if (gamepad2.right_bumper && robot.phone.getPosition() >= 0 && !phone_locked) {  
    //move phone (bottom is 1)  
    robot.phone_position += .1;  
    robot.phone.setPosition(robot.phone_position);  
    phone_locked = true;  
  
} else if (gamepad2.left_bumper && robot.phone.getPosition() <= 1 && !phone_locked) {  
    // phone (top position is 0)  
    robot.phone_position -= .1;  
    robot.phone.setPosition(robot.phone_position);  
    phone_locked = true;  
}  
if (phone_locked) {  
    phonecyclecount++;  
    if (phonecyclecount == 100){  
        phone_locked = false;  
        phonecyclecount = 0;  
    }  
}
```

```
telemetry.addData("0", "FF ChickenNoodleFrenchFry - %2.5f S Elapsed", runtime.seconds());  
//get the value of the light
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
telemetry.addData("D1",format("Rx: %.2f Lx: %.2f Ry: %.2f Ly: %.2f", gamepad1.right_stick_x,  
gamepad1.left_stick_x, gamepad1.right_stick_y, gamepad1.left_stick_y));  
telemetry.addData("D2",format("Rx: %.2f Lx: %.2f ", gamepad2.right_stick_x,  
gamepad2.left_stick_x));  
telemetry.addData("1",format("R - %.2f L - %.2f - Lift - %.2f - Extender - %.2f", robot.right, ro-  
bot.left, robot.lift_power,robot.extender_power));  
  
// telemetry.addData("3",format("Floor Color %2d Blue %2d Red %2d", robot.floor_color.alpha(),  
robot.floor_color.blue(), robot.floor_color.red()));  
telemetry.addData("4",format("Lower %s upper %s", robot.lower_stop.getState(), ro-  
bot.upper_stop.getState()));  
telemetry.addData("Extender state is ", robot.extender_lower.getState());  
telemetry.addData("5",format("phone position %.2f - %d",robot.phone.getPosition  
(),robot.extenderdrive.getCurrentPosition()));  
telemetry.addData("6",format("extnder direction %d",extender_direction));  
telemetry.addData("9",format("atbottom %d atop %d",atbottom, attop));  
telemetry.addData("9",format("location %d ",location));  
telemetry.update();  
idle(); // Always call idle() at the bottom of your while(opModelsActive()) loop  
}  
}  
  
/*  
* This method scales the joystick input so for low joystick values, the  
* scaled value is less than linear. This is to make it easier to drive  
* the robot more precisely at slower speeds.  
*/
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
double scaleInput(double dVal) {
    double[] scaleArray = {0.0, 0.05, 0.09, 0.10, 0.12, 0.15, 0.18, 0.24,
        0.30, 0.36, 0.43, 0.50, 0.60, 0.72, 0.85, 1.00, 1.00};

    // get the corresponding index for the scaleInput array.
    int index = (int) (dVal * 16.0);
    if (index < 0) {
        index = -index;
    } else if (index > 16) {
        index = 16;
    }

    double dScale = 0.0;
    if (dVal < 0) {
        dScale = -scaleArray[index];
    } else {
        dScale = scaleArray[index];
    }

    return dScale;
}

public void setDrivePower(float rightPower, float leftPower, int power_mode) {
    // telemetry.addData("2", format("PM - %d", power_mode));
    // telemetry.update();
    if (power_mode == 2) {
        // set front and back motors full power
        robot.righthdrive.setPower(rightPower);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.leftdrive.setPower(leftPower);
robot.righttwo.setPower(rightPower);
robot.lefttwo.setPower(leftPower);

} else if (power_mode == 1) {
    // set front motors
    robot.rightdrive.setPower(0);
    robot.leftdrive.setPower(0);
    robot.righttwo.setPower(0);
    robot.lefttwo.setPower(0);

} else {
    // set front motors
    robot.rightdrive.setPower(0);
    robot.leftdrive.setPower(0);
    robot.righttwo.setPower(0);
    robot.lefttwo.setPower(0);
}

}

public void setDriverMode(RunMode mode) {

    if (robot.leftdrive.getMode() != mode) {
        robot.leftdrive.setMode(mode);
    }
    if (robot.rightdrive.getMode() != mode) {
        robot.rightdrive.setMode(mode);
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
}

//-----
// Telemetry Configuration
//-----



void composeTelemetry() {

    telemetry.addData("ball color red","ball red color ");
    telemetry.update();

    // At the beginning of each telemetry update, grab a bunch of data
    // from the IMU that we will then display in separate lines.

    telemetry.addAction(new Runnable() { @Override public void run()

    {
        // Acquiring the angles is relatively expensive; we don't want
        // to do that in each of the three items that need that info, as that's
        // three times the necessary expense.

        robot.angles = robot imu.getAngularOrientation(AxesReference.INTRINSIC, AxesOrder.ZYX, AngleUnit.DEGREES);
    }
});


    telemetry.addLine()
        .addData("status", new Func<String>() {
            @Override public String value() {
                return robot imu.getSystemStatus().toShortString();
            }
        });
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
})
.addData("calib", new Func<String>() {
    @Override public String value() {
        return robot imu.getCalibrationStatus().toString();
    }
});

telemetry.addLine()
.addData("heading", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot angles.angleUnit, robot angles.firstAngle);
    }
})
.addData("roll", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot angles.angleUnit, robot angles.secondAngle);
    }
})
.addData("pitch", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot angles.angleUnit, robot angles.thirdAngle);
    }
});

});
}

//-----
// Formatting
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

//-----

```
String formatAngle(AngleUnit angleUnit, double angle) {
    return formatDegrees(AngleUnit.DEGREES.fromUnit(angleUnit, angle));
}

String formatDegrees(double degrees){
    return String.format(Locale.getDefault(), "%.1f", AngleUnit.DEGREES.normalize(degrees));
}
private void initVuforia() {
    /*
     * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
     */
    VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

    parameters.vuforiaLicenseKey = VUFORIA_KEY;
    parameters.cameraDirection = BACK;

    // Instantiate the Vuforia engine
    vuforia = ClassFactory.getInstance().createVuforia(parameters);

}
/**
 * Initialize the Tensor Flow Object Detection engine.
 */
private void initTfod() {

    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(

```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
"tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());  
  
TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters  
(tfodMonitorViewId);  
tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);  
tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LABEL_SILVER_MINERAL);  
  
}  
}
```

Following are the functions to be used during autonomous which is not ready for a meet, but has been written. The first function is the DriveDef2 which provides a method for driving forward and backwards:

```
package org.firstinspires.ftc.Team7341;  
  
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;  
import com.qualcomm.robotcore.hardware.DcMotor;  
import com.qualcomm.robotcore.hardware.HardwareMap;  
import com.qualcomm.robotcore.util.ElapsedTime;  
  
/**  
 * This is NOT an opmode.  
 *  
 * This class can be used to define all the specific hardware for a single robot.  
 * In this case that robot is PrinceCharles.  
 * See AutoBlue and others classes starting with "FF" for usage examples.  
 */
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

* This hardware class assumes the following device names have been configured on the robot:

* Note: All names are lower case and some have single spaces between words.

*/

```
public class DriveDef2
{
    /* Declare OpMode members. */
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Prince Charles's hardware

    private ElapsedTime runtime = new ElapsedTime();

    // Private Members
    private LinearOpMode myOpMode;

    /* local OpMode members. */
    HardwareMap hardwareMap = null;
    private ElapsedTime period = new ElapsedTime();

    /* Constructor */
    public DriveDef2(){

    }

    /* Initialize standard Hardware interfaces */
    public void init(HardwareMap ahwMap, LinearOpMode opMode) {
        // Save reference to Hardware map
        hardwareMap = ahwMap;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
myOpMode = opMode;  
/*  
 * Initialize the drive system variables.  
 * The init() method of the hardware class does all the work here  
 */  
robot.init(hardwareMap, 1);  
  
/*  
 * Use the hardwareMap to get the dc motors and servos by name.  
 * Note that the names of the devices must match the names used  
 * when you configured your robot and created the configuration file.  
 */  
}  
  
/**/  
*  
* waitForTick implements a periodic delay. However, this acts like a metronome with a regular  
* periodic tick. This is used to compensate for varying processing times for each cycle.  
* The function looks at the elapsed cycle time, and sleeps for the remaining time interval.  
*  
* @param periodMs Length of wait cycle in mSec.  
*/  
public void waitForTick(long periodMs) {  
  
    long remaining = periodMs - (long)period(milliseconds());  
  
    // sleep for the remaining portion of the regular cycle period.  
    if (remaining > 0) {
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
try {
    Thread.sleep(remaining);
} catch (InterruptedException e) {
    Thread.currentThread().interrupt();
}

// Reset the cycle clock for the next pass.
period.reset();

/*
 * Method to perform a relative move, based on encoder counts.
 * Encoders are not reset as the move is based on the current position.
 * Move will stop if any of three conditions occur:
 * 1) Move gets to the desired position
 * 2) Move runs out of time
 * 3) Driver stops the opmode running.
*/
public void encoder2Drive(double speed, double inches,
    double timeoutS) {
    int newLeftTarget;
    int newRightTarget;
    int newLefttwoTarget;
    int newRighttwoTarget;

    robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
    robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.righthdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righthdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
```

// Determine new target position, and pass to motor controller

```
newLeftTarget = robot.leftdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
newRightTarget = robot.righthdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
robot.leftdrive.setTargetPosition(newLeftTarget);
robot.righthdrive.setTargetPosition(newRightTarget);
newLefttwoTarget = robot.lefttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
newRighttwoTarget = robot.righttwo.getCurrentPosition() + (int)(Inches * ro-
bot.COUNTS_PER_INCH);
robot.lefttwo.setTargetPosition(newLefttwoTarget);
robot.righttwo.setTargetPosition(newRighttwoTarget);
```

// Turn On RUN_TO_POSITION

```
robot.leftdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.righthdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.righttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

// reset the timeout time and start motion.

```
runtime.reset();
robot.leftdrive.setPower((float)(speed));
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.righthdrive.setPower((float)(speed));
robot.lefttwo.setPower((float)(speed));
robot.righttwo.setPower((float)(speed));

//telemetry.addData("Path1", "Running to %7d :%7d", newLeftTarget, newRightTarget);
// keep looping while we are still active, and there is time left, and both motors are running.
// keep looping while we are still active, and there is time left, and both motors are running.
while ((runtime.seconds() < timeoutS) ||
    ( robot.leftdrive.isBusy() && robot.righthdrive.isBusy() &&
    robot.lefttwo.isBusy() && robot.righttwo.isBusy() )) {
    myOpMode.telemetry.addData("3", "new left %d", newLeftTarget);
    myOpMode.telemetry.addData("3", "new right %d", newRightTarget);
    myOpMode.telemetry.addData("3", "new left %d", newLefttwoTarget);
    myOpMode.telemetry.addData("3", "new right %d", newRighttwoTarget);
    myOpMode.telemetry.addData("3", "curr left %d", robot.lefttwo.getCurrentPosition());
    myOpMode.telemetry.addData("3", "curr right %d", robot.righttwo.getCurrentPosition());
    myOpMode.telemetry.update();
    waitForTick (5);
}

// Stop all motion;
robot.lefttwo.setPower(robot.STOP_SPEED);
robot.righttwo.setPower(robot.STOP_SPEED);
robot.leftdrive.setPower(robot.STOP_SPEED);
robot.righthdrive.setPower(robot.STOP_SPEED);

// Turn off RUN_TO_POSITION
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.lefttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);

}
}
```

The second function is the TurnDef2 which provides a method for turning left or right in place (one set of wheels go forward while the other set goes backwards):

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cGyro;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.util.ElapsedTime;
import com.qualcomm.robotcore.util.Range;

/**
 * This is NOT an opmode.
 *
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
 *
 * This hardware class assumes the following device names have been configured on the robot:
 * Note: All names are lower case and some have single spaces between words.

```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
/*
public class TurnDef2
{
    /* Declare OpMode members. */
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Prince Charles's hardware

    private ElapsedTime runtime = new ElapsedTime();

    // Private Members
    private LinearOpMode myOpMode;

    /* local OpMode members. */
    HardwareMap hardwareMap = null;
    private ElapsedTime period = new ElapsedTime();

    /* Constructor */
    public TurnDef2(){

    }

    /* Initialize standard Hardware interfaces */
    public void init(HardwareMap ahwMap, LinearOpMode opMode) {
        // Save reference to Hardware map
        hardwareMap = ahwMap;

        myOpMode = opMode;
        /*
        * Use the hardwareMap to get the dc motors and servos by name.
        */
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
* Note that the names of the devices must match the names used
* when you configured your robot and created the configuration file.
*/
robot.init(hardwareMap, 1);

}

/**
*
* waitForTick implements a periodic delay. However, this acts like a metronome with a regular
* periodic tick. This is used to compensate for varying processing times for each cycle.
* The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
*
* @param periodMs Length of wait cycle in mSec.
*/
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period(milliseconds());

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            Thread.sleep(remaining);
        } catch (InterruptedException e) {
            Thread.currentThread().interrupt();
        }
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// Reset the cycle clock for the next pass.  
    period.reset();  
}  
  
/*  
 * Method to perform a relative move, based on encoder counts.  
 * Encoders are not reset as the move is based on the current position.  
 * Move will stop if any of three conditions occur:  
 * 1) Speed negative is forward and positive is backwards  
 * 2) leftInches - pass a negative to go forward and positive to backwards  
 * 3) rightInches - pass a negative to go forward and positive to backwards  
 * 4) direction - 1 - right, 2 - left  
 * 5) timeoutS - number of seconds before the function timeout  
 */  
public void encoder2DriveTurn2(double speed, double Inches,  
                                int direction, double timeoutS) {  
    int newLeftTarget;  
    int newRightTarget;  
    int newLefttwoTarget;  
    int newRighttwoTarget;  
    float rightdirection = 1;  
    float leftdirection = 1;  
  
    // 1 - turn right  
    // 2 - turn left  
    // minus goes forward... matches the controller  
    // Determine new target position, and pass to motor controller
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
if (direction == 2){  
    leftright = -1;  
} else {  
    rightdirection = -1;  
}  
  
robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);  
robot.rightdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);  
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);  
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);  
  
// Determine new target position, and pass to motor controller  
  
newLeftTarget = robot.leftdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH *  
leftright);  
newRightTarget = robot.rightdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH *  
rightdirection);  
  
robot.leftdrive.setTargetPosition(newLeftTarget);  
robot.rightdrive.setTargetPosition(newRightTarget);  
  
// set the front motors  
newLefttwoTarget = robot.lefttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH *  
leftright);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.righttwo.setPower(robot.STOP_SPEED);
robot.leftdrive.setPower(robot.STOP_SPEED);
robot.rightdrive.setPower(robot.STOP_SPEED);

// Turn off RUN_TO_POSITION
robot.lefttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);

}

// turn right or left
// 1 - left
// 2 - right

public void encoder2DriveTurn(double speed, double angle, int direction) {

    double direction_value;

    // Determine new target position, and pass to motor controller
    if (direction == 1){
        direction_value = 1;
    } else {
        direction_value = -1;
    }

    // Turn On RUN_Without encoder
    robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
newRighttwoTarget = robot.righttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH
```

```
* rightdirection);
```

```
robot.lefttwo.setTargetPosition(newLefttwoTarget);
```

```
robot.righttwo.setTargetPosition(newRighttwoTarget);
```

```
// Turn On RUN_TO_POSITION
```

```
robot.leftdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

```
robot.rightdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

```
robot.lefttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

```
robot.righttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

```
// reset the timeout time and start motion.
```

```
runtime.reset();
```

```
robot.leftdrive.setPower((float)(speed) * leftright);
```

```
robot.rightdrive.setPower((float)(speed) * rightleft);
```

```
robot.lefttwo.setPower((float)(speed) * leftright);
```

```
robot.righttwo.setPower((float)(speed) * rightleft);
```

```
// keep looping while we are still active, and there is time left, and both motors are running.
```

```
while ((runtime.seconds() < timeoutS) ||
```

```
      (robot.leftdrive.isBusy() && robot.rightdrive.isBusy() &&
```

```
      robot.lefttwo.isBusy() && robot.righttwo.isBusy() ) ) {
```

```
    waitForTick (5);
```

```
}
```

```
// Stop all motion;
```

```
robot.lefttwo.setPower(robot.STOP_SPEED);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.righthdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// robot.gyro.resetZAxisIntegrator();

while (!onHeading(speed, direction_value*angle, robot.P_TURN_COEFF)) {
    // Update telemetry & Allow time for other processes to run.

}

// Stop all motion;
robot.leftdrive.setPower(robot.STOP_SPEED);
robot.righthdrive.setPower(robot.STOP_SPEED);

}

/**
 * Perform one cycle of closed loop heading control.
 *
 * @param speed Desired speed of turn.
 * @param angle Absolute Angle (in Degrees) relative to last gyro reset.
 *              0 = fwd. +ve is CCW from fwd. -ve is CW from forward.
 *              If a relative angle is required, add/subtract from current heading.
 * @param PCoeff Proportional Gain coefficient
 * @return
 */
boolean onHeading(double speed, double angle, double PCoeff) {
    double error;
    double steer;
    boolean onTarget = false;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
double leftSpeed;
double rightSpeed;

// determine turn power based on +/- error
error = getError(angle);

if (Math.abs(error) <= robot.HEADING_THRESHOLD) {
    steer = 0.0;
    leftSpeed = 0.0;
    rightSpeed = 0.0;
    onTarget = true;
}
else {
    steer = getSteer(error, PCoeff);
    rightSpeed = speed * steer;
    leftSpeed = -rightSpeed;
}

// Send desired speeds to motors.
robot.leftdrive.setPower(leftSpeed);
robot.righthdrive.setPower(rightSpeed);

return onTarget;
}

/**
 * returns desired steering force. +/- 1 range. +ve = steer left

```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
* @param error Error angle in robot relative degrees
* @param PCoeff Proportional Gain Coefficient
* @return
*/
public double getSteer(double error, double PCoeff) {
    return Range.clip(error * PCoeff, -1, 1);
}

/**
 * getError determines the error between the target angle and the robot's current heading
 * @param targetAngle Desired angle (relative to global reference established at last Gyro Reset).
 * @return error angle: Degrees in the range +/- 180. Centered on the robot's frame of reference
 *         +ve error means the robot should turn LEFT (CCW) to reduce error.
*/
public double getError(double targetAngle) {

    double robotError;

    // calculate error in -179 to +180 range (
    // robotError = targetAngle - robot.gyro.getIntegratedZValue();
    // while (robotError > 180) robotError -= 360;
    // while (robotError <= -180) robotError += 360;
    // return robotError;
    return 0;
}

}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

The Autonomous function (AutoPosition) provides instructions to the robot to act on it's own. Following is the code:

```
/* Copyright (c) 2015 Qualcomm Technologies Inc
```

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;
import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.tfod.Recognition;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

@Autonomous(name = "FF: AutoPosition", group = "Auto")
//@Disabled
public class AutoPosition extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
    TurnDef2 turn = new TurnDef2();

    LiftDef2 lift = new LiftDef2();
    DriveDistanceDef distance1 = new DriveDistanceDef();
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
boolean phone_locked;
boolean phone_in;
boolean phone_out;
int phonecyclecount;

private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";

/*
 * IMPORTANT: You need to obtain your own license key to use Vuforia. The string below with which
 * 'parameters.vuforiaLicenseKey' is initialized is for illustration only, and will not function.
 * A Vuforia 'Development' license key, can be obtained free of charge from the Vuforia developer
 * web site at https://developer.vuforia.com/license-manager.
 *
 * Vuforia license keys are always 380 characters long, and look as if they contain mostly
 * random data. As an example, here is a example of a fragment of a valid key:
 * ...
 * Once you've obtained a license key, copy the string from the Vuforia web site
 * and paste it in to your code onthe next line, between the double quotes.
 */

// Variables to be used for later

private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-
Num6rS90mBDJwrv1CJMcf5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/"
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

MZz7Zvp35rYD6lavfkgCMZ0DVAXHBv4J5LlrGlVXYfhhS1NkITGPDqVRW2aBmKLwctHZaztzycau3g//
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnFkYGcE-
Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here

```
/**  
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
 * localization engine.  
 */  
private VuforiaLocalizer vuforia;  
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFObjectDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();  
  
private ElapsedTime runtime = new ElapsedTime();  
  
@Override public void runOpMode() throws InterruptedException {  
    String print_val;  
    print_val = "Test";  
    String target_print_val;  
    target_print_val = "Target";  
    String turn_print_val;
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
turn_print_val = "Turn";
String hanging_print_val;
hanging_print_val = "Not Hanging";
/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);
// lift.init(hardwareMap, this);

distance1.init(hardwareMap, this);
drive.init(hardwareMap, this);
turn.init(hardwareMap, this);
// touch.init(hardwareMap, this);

// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Autonomous Position");

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righthdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righthdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Send telemetry message to indicate successful Encoder reset
telemetry.addData("1", "Starting drive position at Left %7d - Right %7d Lefttwo %7d - Righttwo %
7d",
    robot.leftdrive.getCurrentPosition(),
    robot.righthdrive.getCurrentPosition(),
    robot.lefttwo.getCurrentPosition(),
    robot.righttwo.getCurrentPosition());

robot.position_option = 0;

while (robot.position_side == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.position_side = 1;
    }else if (gamepad2.y) {
        robot.position_side = 2;
    }else {
        telemetry.addData("5","Select D2 - x for Red Side cnt - %d", robot.count);
        telemetry.addData("6","Select D2 - y for Blue Side");
        telemetry.update();
    }
    robot.wait(1);
}
robot.wait(1);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
robot.count = 0;
while (robot.position_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.position_option = 3;
        print_val = "Crater";
    } else if (gamepad2.b) {
        robot.position_option = 1;
        print_val = "Depot";
    } else {
        telemetry.addData("5", "Select D2 - a for position 1 crater cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - b for position 2 depot");
        telemetry.update();
        robot.wait(1);
    }
}
// turning right or left
robot.wait(1);
robot.count = 0;
while (robot.turn_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.turn_option = 1;
        turn_print_val = "Right";
    } else if (gamepad2.y) {
        robot.turn_option = 2;
        turn_print_val = "Left";
    } else {
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
telemetry.addData("5", "Select D2 - x for turning right cnt - %d", robot.count);
telemetry.addData("6", "Select D2 - y for turning left");
telemetry.update();
robot.wait(1);

}

}

// robot hanging or not
robot.wait(1);
robot.count = 0;
while (robot.hanging == -1 && (!isStopRequested() || opModelsIsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.hanging = 1;
        hanging_print_val = "Hanging";
    } else if (gamepad2.y) {
        robot.hanging = 0;
        hanging_print_val = "Not Hanging";
    } else {
        telemetry.addData("5", "Select D2 - x for Hanging cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - y for NOT Hanging");
        telemetry.update();
        robot.wait(1);
    }
}
telemetry.addData("3", "before setting up tfod");

initVufor();
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {  
    telemetry.addData("3", "setting up tfod");  
    telemetry.update();  
    initTfod();  
    telemetry.addData("3", "back from setting up tfod");  
    telemetry.update();  
    robot.wait(5);  
  
} else {  
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");  
}  
  
// Send telemetry message to indicate successful Encoder reset  
if (robot.position_side == 1) {  
    telemetry.addData("1", "All setup at Position Red Side %s - %s - %s - %s",  
        print_val, turn_print_val, target_print_val, hanging_print_val);  
} else {  
    telemetry.addData("1", "All setup at Position Blue Side %s - %s - %s - %s",  
        print_val, turn_print_val, target_print_val, hanging_print_val);  
}  
telemetry.addData("1", "Position %d - %d - %d",  
    robot.turn_option + robot.position_option, robot.turn_option, robot.position_option);  
telemetry.addData("2", "Waiting to start");  
telemetry.update();  
  
// wait for the start button to be pressed.
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
waitForStart();

int location = 0;

// init phone
robot.phone_position = .08;
robot.phone.setPosition(robot.phone_position);

robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

if (robot.hanging == 1) {
    // get down from the lander
    // lift.liftmove(1, .25);
}

robot.path_option = robot.turn_option + robot.position_option;
if (robot.find_target == 1) {
    // put phone servo in start position
telemetry.addData("6", " moving phone");
    robot.init(hardwareMap, 2);
    //get the target value
    if (tfod != null) {
        tfod.activate();
    }
    phone_locked = true;
    findMineral();
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
telemetry.addData("7", "Location is %d", robot.gold_position);

telemetry.addData("6", "Option going to move the gold mineral");
if (robot.gold_position == 1){
    // turn to the left and move forward
    // turn left
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 2, 2.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 48in");
        // forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 37, 2.5);
        // turn right
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 1, 2.5);
        // forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.0);

    } else {
        // on the crater side
        telemetry.addData("6", "Option going forward 31in");
        // forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
    }
} else if (robot.gold_position == 2) {
    // going straight
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 48in");
        // forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 37, 2.5);
        // turn right
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 1, 2.5);
        // forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.0);
    }
}
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
telemetry.addData("6", "Option going forward 48in");
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 52, 4.5);
} else {
//on the crater side
telemetry.addData("6", "Option going forward 28in");
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 39, 5.5);
}
} else if (robot.gold_position == 3) {
// turning right and then going straight
// turn right
drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5.1, 3.5);
if (robot.position_option == 1 ){
//on the depot side
telemetry.addData("6", "Option going forward 35in ");
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 35, 2.5);
//turn left
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11.2, 2.5);
//forward towards the depot
drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.5);
} else {
//on the crater side
telemetry.addData("6", "Option going forward 31in");
//forward towards the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
    }  
}  
  
telemetry.update();  
  
robot.marker_power = .7;  
robot.marker.setPosition(robot.marker_power);  
robot.wait(2);  
robot.marker_power = .4;  
robot.marker.setPosition(robot.marker_power);  
  
} else if (robot.path_option == 2){  
    telemetry.addData("6", " Depot Side turning Right");  
    telemetry.update();  
    //forward and get past the lander legs  
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);  
    // turn right  
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);  
    //forward  
    drive.encoder2Drive(robot.DRIVE_SPEED1, 32, 5.5);  
    // turn left  
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);  
    //forward  
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);  
    // turn left  
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);  
    //forward  
    drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
// dump off the marker

// backwards towards the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);

} else if (robot.path_option == 3) {
    telemetry.addData("6", " Depot Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,1, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,1, 3.5);
    //forward toward the depot
    drive.encoder2Drive(robot.DRIVE_SPEED, 12, 3.5);
    // dump off the marker

    //forward into the crater
    drive.encoder2Drive(robot.DRIVE_SPEED, 5, 3.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,1, 3.5);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
//forward toward the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 96, 10.5);

} else if (robot.path_option == 4) {
    telemetry.addData("6", " Crater Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
    //forward going past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 29, 5.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,2, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 26, 5.5);

}else if (robot.path_option == 5) {
    telemetry.addData("6", " Crater Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward and go to the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 60, 8.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,1, 3.5);
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

```
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
// dump off the marker

// backwards toward the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);

} else {
    // nearest to the audience
    // the distance is 23 15/16 but allow for role from stone to parking zone
    telemetry.addData("6", " No Valid option picked");
    telemetry.update();

}

if (tfod != null) {
    tfod.shutdown();
}
robot.wait(20);
telemetry.addData("Path", "Autonomous Complete");
telemetry.update();

idle(); // Always call idle()
}

public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period.milliseconds();
```

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

// sleep for the remaining portion of the regular cycle period.

```
if (remaining > 0) {  
    try {  
        Thread.sleep(remaining);  
    } catch (InterruptedException e) {  
        Thread.currentThread().interrupt();  
    }  
}
```

// Reset the cycle clock for the next pass.

```
period.reset();  
}  
}
```

During the meet today we were able to get one ball in the lander. We do need more practice and some minor adjustments so we can get to more minerals. Additional work needs to be done as we did run the Autonomous program and the robot went in circles.... So sad something is connected wrong.

Following are the scores from the meet:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
1	14534 12090	4227 9013	2	27	0	2	0	0	27	0
2	8392 14989	7592 7477	19	228	0	19	0	110	118	0
3	7341 6323	15401 14976	50	17	0	40	10	0	17	0

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

Process:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
4	4717 14215	14856 14538	25	39	0	25	0	10	29	0
5	14976 11499	7592 8945	23	92	0	23	0	30	62	0
6	8392 4717	12090 15401	25	104	10	15	0	25	79	0
7	14856 4227	7477 7341	170	148	95	75	0	55	93	0
8	14215 11499	6323 14534	58	106	0	58	0	70	36	0
9	9013 8945	14538 14989	63	36	30	33	0	0	36	0
10	14534 14976	4717 7477	25	136	0	25	0	55	81	0
11	11499 14856	9013 8392	104	71	15	89	0	10	61	0
12	14538 7592	12090 6323	136	194	55	81	0	80	114	0
13	15401 4227	14989 14215	109	82	70	39	0	0	32	50
14	7341 7592	8945 14534	161	119	80	81	0	0	69	50
15	7477 15401	11499 14538	144	139	65	79	0	0	79	60
16	4227 8945	6323 4717	78	76	45	23	10	0	76	0
17	14989 12090	14976 14856	112	80	30	82	0	40	40	0
18	14215 9013	7341 8392	50	20	15	35	0	0	20	0
19	14856 14534	8945 15401	104	67	15	89	0	15	52	0

Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

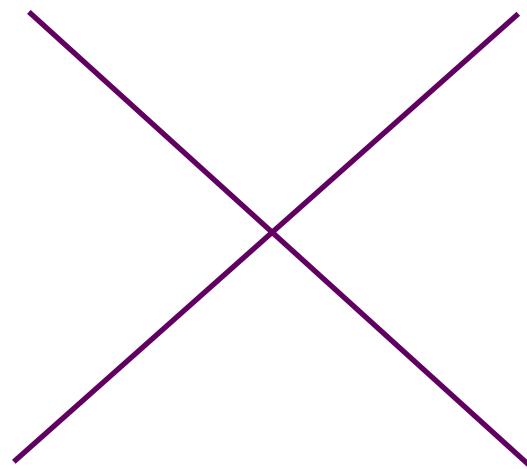
Engineering Activity Continued

Date: November 9-10, 2018

Process:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
20	8392 14538	14976 4227	35	109	0	35	0	55	54	0
21	12090 7341	14989 11499	78	80	45	33	0	0	80	0
22	7477 6323	9013 4717	182	92	75	107	0	0	62	30
23	7592 14976	14215 14856	144	128	80	54	10	75	53	0
24	4717 14538	11499 7341	48	15	0	48	0	0	15	0
25	14215 7477	8945 12090	134	64	45	89	0	60	4	0
26	15401 9013	4227 7592	61	132	15	46	0	70	62	0
27	14534 14989	6323 8392	130	75	55	75	0	30	45	0

Now it's time to start tweaking our autonomous program and adding enhancing features to our robot.



Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 9-10, 2018

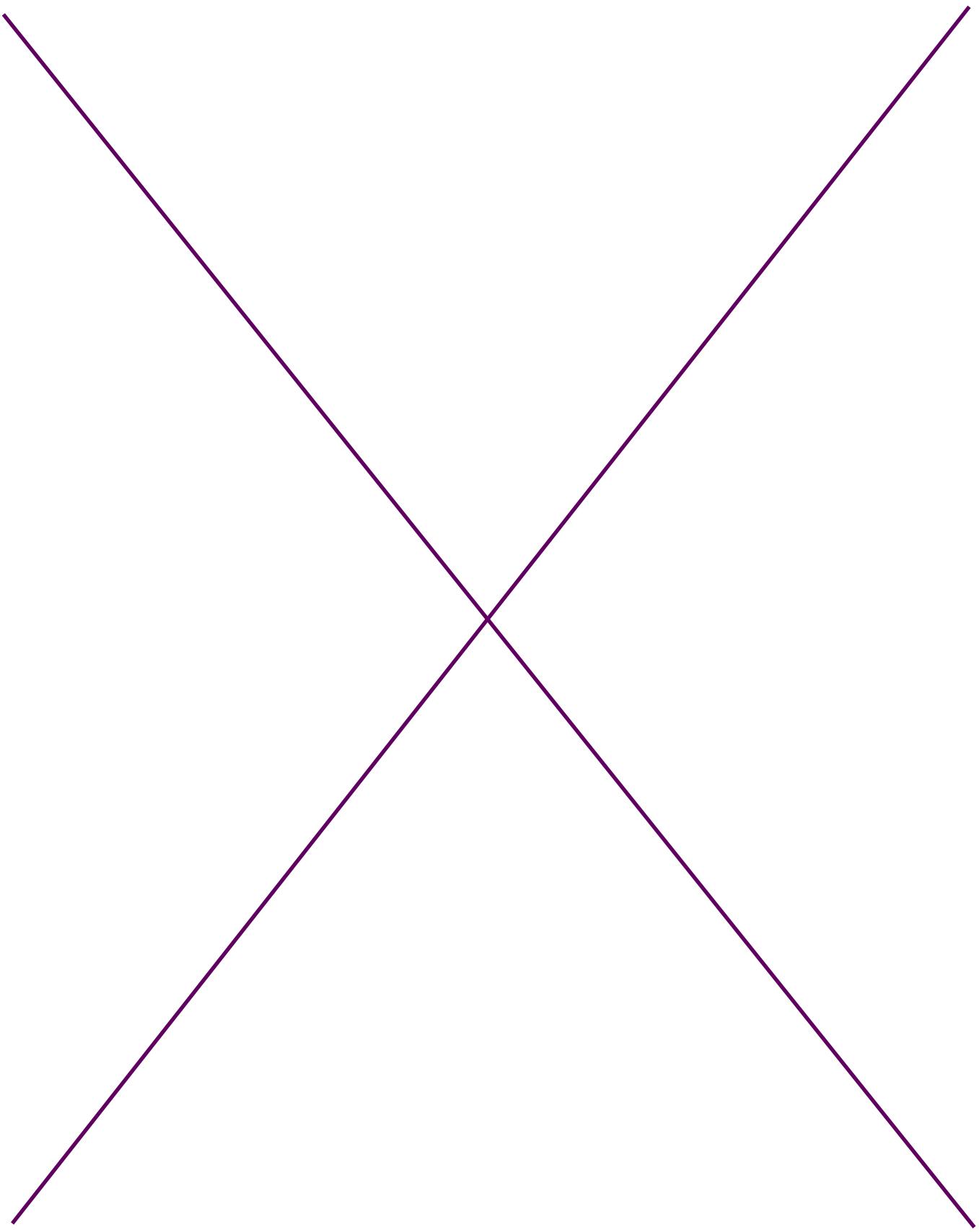
Process:

The robot is ready for competition with our inspection sheets.



Signature : Jessica Anderson

Date: Nov. 10, 2018



Signature : Jessica Anderson

Date: Nov. 10, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: November 25, 2018

Purpose of the Activity:

Determining what happened during Autonomous

Seeing if we can get closer to the crater wall

Create a new team marker that will go through the collector.

Process:

We needed to determine why the Autonomous program was not running correctly at all. When commanded to travel 5 inches, it did not move the correct amount or the proper direction. After removing the bottom plate it was determined that all the motor control wire were not connected correctly and that the motor definition on the robot was switched. The middle motors were in port 0 and not 1. The configuration was changed and the control wires were connected. We tried the Autonomous program again and it worked better. Now we will layout all the possible options for us to program.

We will add following five questions to the Initialization of the Autonomous program:

1. Red or Blue side
2. Turning Right or Left
3. Crater or depot side
4. Finding the Gold Mineral
5. Start hanging

From the answers to these question we will pick a path through the Autonomous Program. We will need to verify speed and distance timing. Can the robot move 25 inches in 4 or 6 seconds. This was determined to be under 2 seconds so we have set our process timeout time to 1.5.

signature : Shelby Greer

Date: Nov. 25, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 25, 2018

Process:

After moving the connections to the correct ports and connecting all the encoder wires. We ran the test autonomous program and the robot move the expected amount a distance. The code was then added to support the new questions added to the start of the program. We had the following paths to take:

1. First determine if you are on the red or blue side
2. Determine if you are on the crater or depot side
3. Determine if you would like to turn right or left to get to the final spot.

Following is the calculation used to determine the number of rotations are needed to travel. We found that we had the wrong number for the “COUNTS_PER_MOTOR_REV” which cause incorrect calculations for the length needed ... it was too short. The counts_per_motor was 1120. After reviewing the information on the motor we determine the count was 1680. You take the counts_per_inch and multiple the number of inches you would like to move to set the motors encoder.

```
static final double COUNTS_PER_MOTOR_REV = 1680; //eg: AndyMark Motor Encoder  
static final double DRIVE_GEAR_REDUCTION = 1; //This is < 1.0 if geared UP  
static final double WHEEL_DIAMETER_INCHES = 4.0; //For figuring circumference  
static final double GEAR_DIAMETER_INCHES = 1.0; //For figuring circumference  
  
static final double COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *  
DRIVE_GEAR_REDUCTION) / (WHEEL_DIAMETER_INCHES * 3.1415);
```

signature : Shelby Greer

Date: Nov. 25, 2018

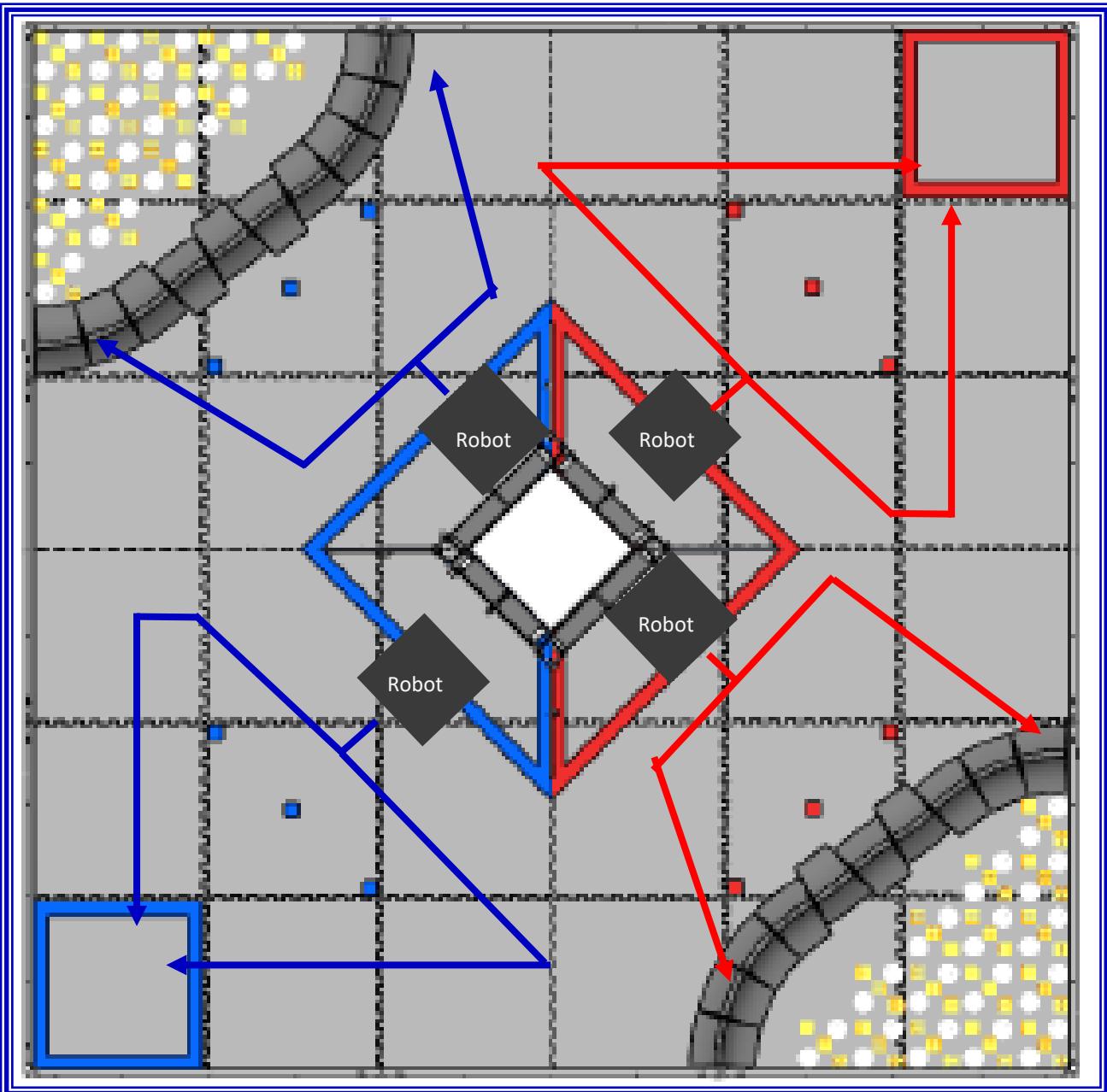
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 25, 2018

Process: These are the travel pattern before determining the mineral position.



Signature: Shelby Greer

Date: Nov. 25, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 25, 2018

Process:

We observed that the front wheels were preventing us from getting close to the crater wall which was needed in order to pick up the minerals on the other side. We determined that if we moved the front wheels back we would gain enough space allowing us to get more minerals.



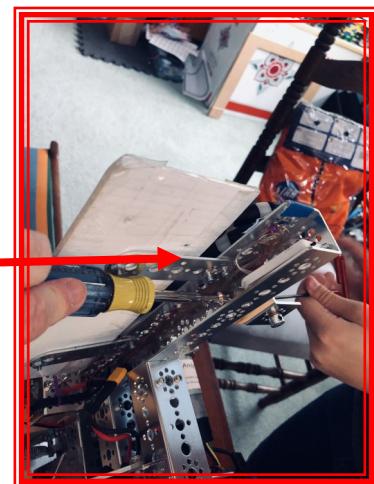
Before



After

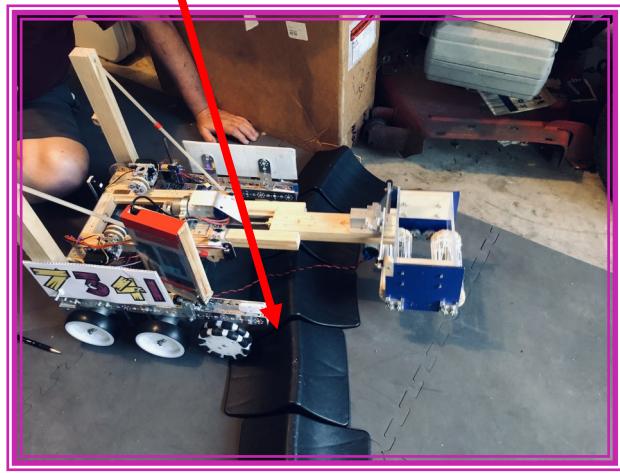
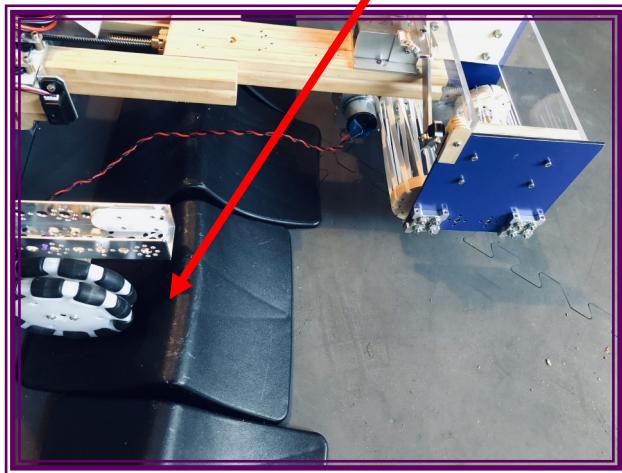


Before moving
the wheels



Moving the
wheels

After moving
the wheels



Signature: Shelby Greer

Date: Nov. 25, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 25, 2018

Process:

After the last meet we also talked about adding wheelie bar wheels to the back of the robot. Our engineer created something for us to try, so we added them to the robot to see if it will help prevent us from tipping over when trying to go over the crater wall.



But when we went out to test the robot we found that the wheelie bar wheels were too tall and the prevented the robot from moving around. We needed the engineer to cut the bars down allowing the robot to tip up some to get over the crater.

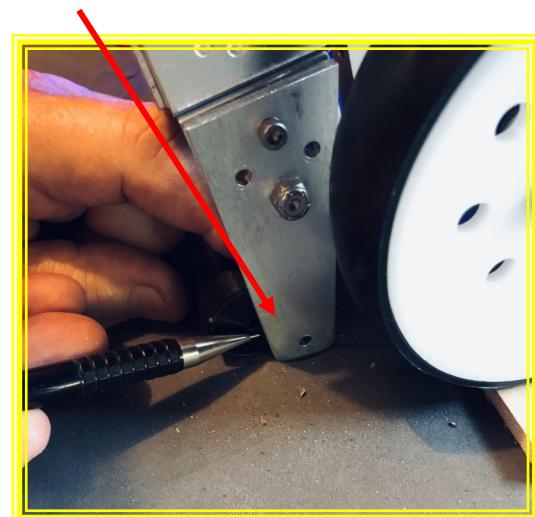


We will try out the adjusted wheelie bar wheels next time. To see if they will prevent the robot from tipping over when going over the



crater wall.

The wheelie bar wheels will be adjusted about 1 and 1/2 inches off the ground allowing for the tilting of the robot about 45 degrees.



signature : Shelby Greer

Date: Nov. 25, 2018

Team 7341

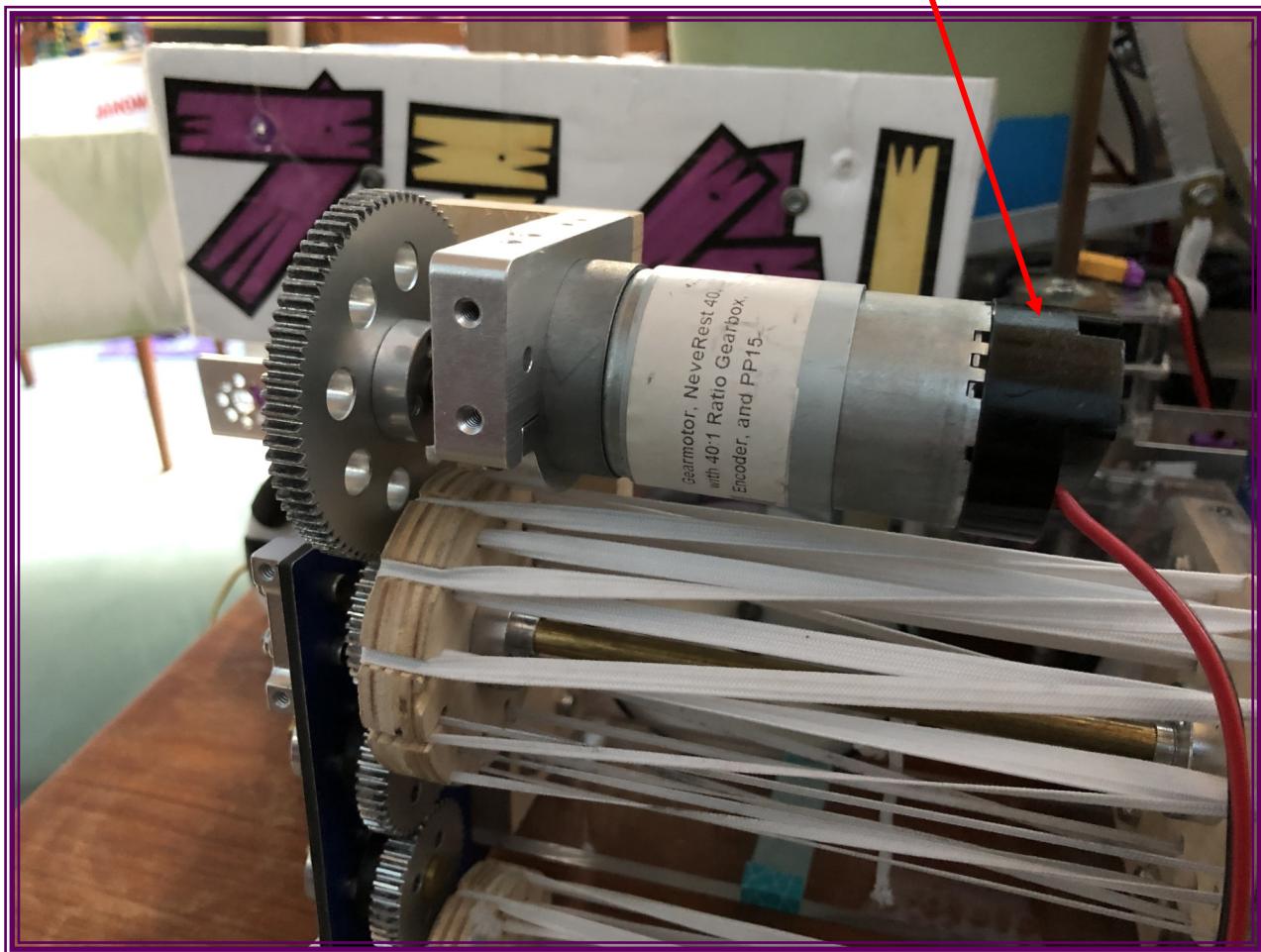
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: November 25, 2018

Process:

We completed additional testing on the grabber or mineral collector and found that the motor we were using did not have enough stall torque and it was stopping when something was jammed in the collector. The motor that we were using had a stall torque of 16.8 oz-in with 303 rpm for the speed a ServoCity 303 rpm Gear Motor. We have changed the motor to a AndyMark NeveRest 40 which has a stall torque of 350 oz-in and a speed of 6,600 rpm.



Signature: Shelby Greer

Date: Nov. 25, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: December 9, 2018

Purpose of the Activity:

We will be testing the following:

Marker dumper

Worm Gear for the outer lift

Autonomous Testing for going to the depot

Process:

We have added a new function that will hold our Team Marker and dump it into the depot once the robot has entered the depot. We created a tray and servo box for this function.



The tray and servo box are made out of 1/4 pine plywood. The plywood is better to use than solid pine because the plywood will not warp as much as the solid wood will.

After adding the new function a test program was created to make sure that we could control the servo.



Signature : Jessica Anderson

Date: Dec. 9, 2018

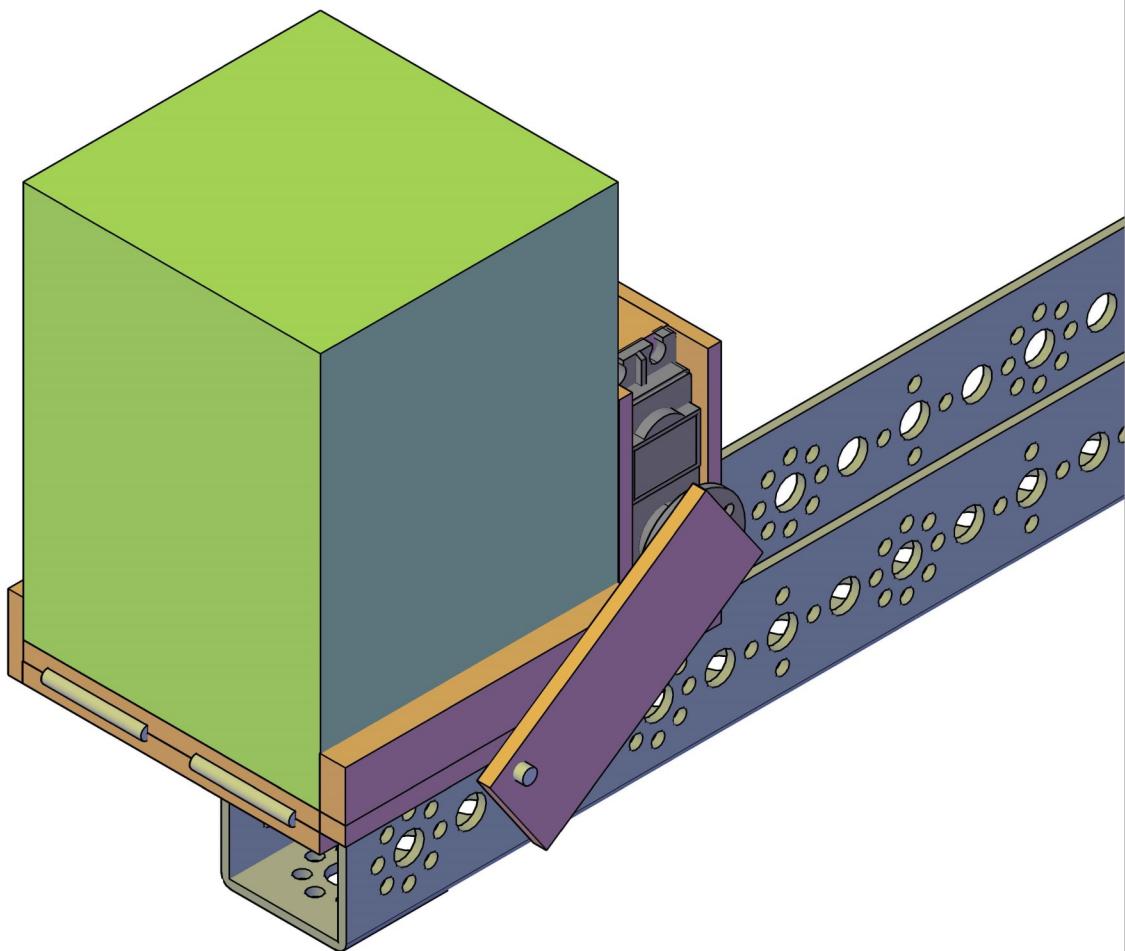
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:



This is the drawing of our marker dumper.

Signature : Jessica Anderson

Date: Dec. 9, 2018

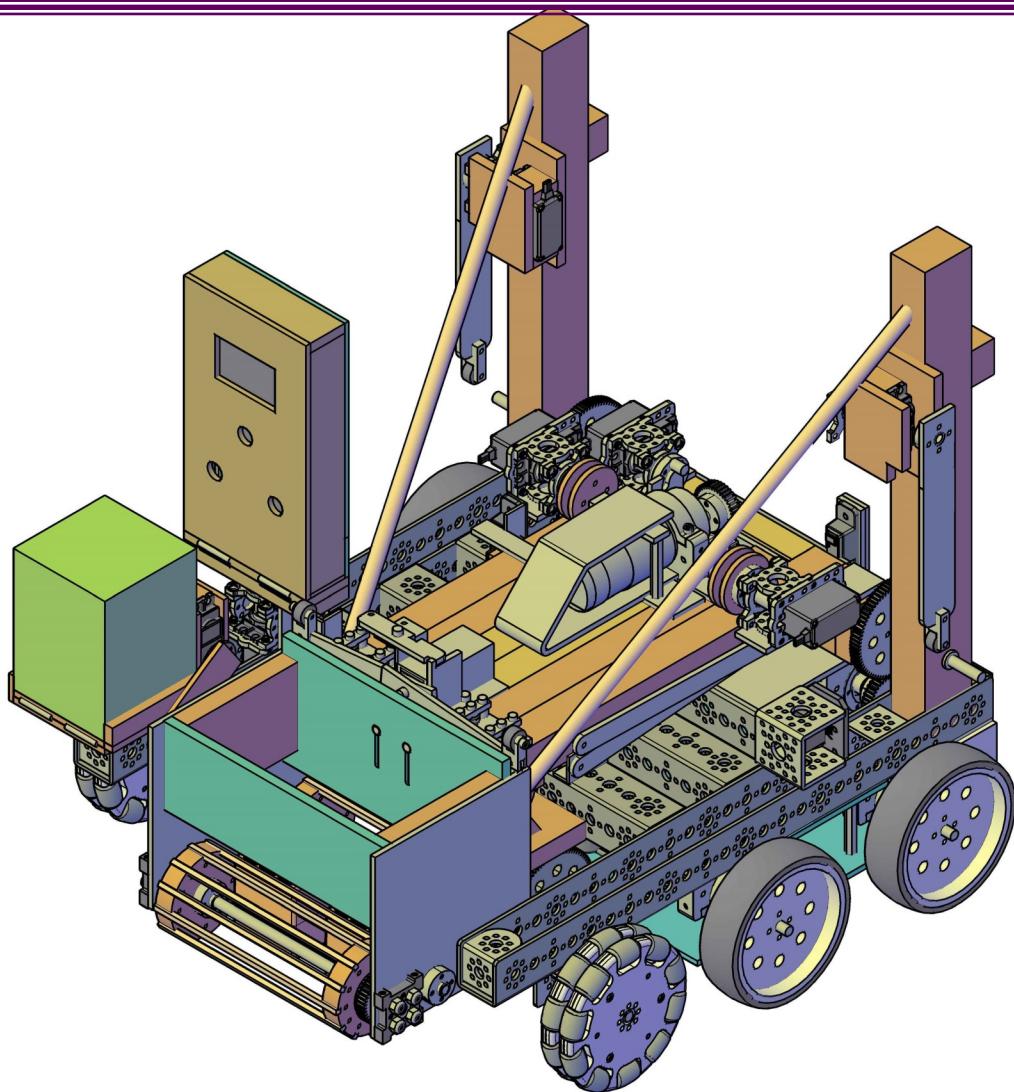
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:



Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

The following code is our completed Marker Dump Test Program

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;
import com.qualcomm.robotcore.eventloop.opmode.**Autonomous**;

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;

import java.text.SimpleDateFormat;
import java.util.Date;

@Autonomous(name = "FF: MarkerTest", group = "Auto")
//@Disabled
public class MarkerTest extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
    TurnDef2 turn = new TurnDef2();

    LiftDef2 lift = new LiftDef2();
    DriveDistanceDef distance1 = new DriveDistanceDef();

    private ElapsedTime runtime = new ElapsedTime();

    @Override public void runOpMode() throws InterruptedException {

        /*
         * Initialize the drive system variables.
        */

        // Set the start time and the end time of the routine.
        runtime.reset();
        runtime.setTimeout(10000);
        
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
* The init() method of the hardware class does all the work here
*/
robot.init(hardwareMap, 1);

robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);
// wait for the start button to be pressed.

waitForStart();

int location = 0;

telemetry.addData("6", " Dump Marker");
telemetry.update();

robot.wait(1);
robot.marker_power = .7;
robot.marker.setPosition(robot.marker_power);
robot.wait(2);
robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

telemetry.addData("Path", "Marker " + " Test Complete");
telemetry.update();

idle(); // Always call idle()
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

}

```
public void waitForTick(long periodMs) {  
  
    long remaining = periodMs - (long)period.currentTimeMillis();  
  
    // sleep for the remaining portion of the regular cycle period.  
    if (remaining > 0) {  
        try {  
            Thread.sleep(remaining);  
        } catch (InterruptedException e) {  
            Thread.currentThread().interrupt();  
        }  
    }  
  
    // Reset the cycle clock for the next pass.  
    period.reset();  
}  
}
```

This useful program showed us that we could dump the marker if we were able to get to the depot.

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

After testing out the Marker dump we began testing out the Autonomous program designed from the previous meeting. We determined that the phone was detecting two targets at the same time cause the robot to not move to the correct position. It was determined that using the phone's servo position was not the correct method of determining the position of the GOLD mineral. We are going to change to using the mineral recognition position.

We have created a test program to determine if the change to the code is correct. Following is the code snippet:

```
if (goldMineralX != -1 && silverMineral1X != -1 && silverMineral2X != -1) {  
    if (silverMineral2X > goldMineralX && silverMineral1X > goldMineralX) {  
        telemetry.addData("Gold Mineral Position", "Left");  
        robot.gold_position = 1;  
    } else if (silverMineral2X < goldMineralX && silverMineral1X < goldMineralX) {  
        telemetry.addData("Gold Mineral Position", "Right");  
        robot.gold_position = 3;  
    } else {  
        telemetry.addData("Gold Mineral Position", "Center");  
        robot.gold_position = 2;  
    }  
}
```

Testing continued to find that our initial thought were not correct and went back to using the servo's position, but we fine tuned the positions where the phone was pointing at as well as the time between each of the moves. The following code was the final outcome for determining the position:

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
while ((goldMineralX == -1 && silverMineral1X == -1 && silverMineral2X == -1) || (!isStopRequested())
&& opModelsActive())) {

List<Recognition> updatedRecognitions = tfod.getUpdatedRecognitions();

// if data - check it
if (updatedRecognitions != null) {
    // telemetry.addData("# Object Detected", updatedRecognitions.size());
    count++;
    // telemetry.addData("7", "track is %d", count);
    int item_count = 0;
    //phone_locked = false;
    if (updatedRecognitions.size() > 0) {

        for (Recognition recognition : updatedRecognitions) {
            label = recognition.getLabel();
            item_count++;
            telemetry.addData("7", "checking - %s found %d", label , found);
            telemetry.update();
            if (recognition.getLabel().equals(LABEL_GOLD_MINERAL) && goldMineralX == -1) {
                found++;
                telemetry.addData("7", "found gold");
                telemetry.update();
                goldMineralX = (int) recognition.getLeft();
                mineral_location = robot.phone_position;
                robot.wait(1);
            }
        }
    }
}
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral1X == -1) {  
    found++;  
    telemetry.addData("7", "found silver");  
    telemetry.update();  
    silverMineral1X = (int) recognition.getLeft();  
    mineral_location_silver1 = robot.phone_position;  
    robot.wait(1);  
} else if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral2X == -1 &&  
    mineral_location_silver1 != robot.phone_position) {  
    found++;  
    telemetry.addData("7", "found silver");  
    telemetry.update();  
    silverMineral2X = (int) recognition.getLeft();  
    mineral_location_silver2 = robot.phone_position;  
    robot.wait(1);  
}  
// telemetry.addData("4", "count %d - internal count %d", updatedRecognitions.size(),  
item_count);  
// telemetry.update();  
if (item_count == updatedRecognitions.size()) break;  
}  
}  
}  
telemetry.addData("er", "should move phone %b %b %b", phone_in, phone_out, phone_locked);  
telemetry.update();  
// move the phone box to find the target  
if (phone_in && !phone_locked) {
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
//move phone (bottom is 1)
if (robot.phone_position > .41) {
    phone_in = false;
    phone_out = true;
} else {
    robot.phone_position += .12;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;

}
} else if (phone_out && !phone_locked) {
    // phone (top position is 0)
    if (robot.phone_position < .19) {
        phone_in = true;
        phone_out = false;
    } else {
        robot.phone_position -= .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
}
if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 25000) {
        phone_locked = false;
        phonecyclecount = 0;
    }
}
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;
import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.tfod.Recognition;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

{@Autonomous(name = "FF: TargetTest", group = "Auto")}
//@Disabled
public class TargetTest extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
    TurnDef2 turn = new TurnDef2();

    LiftDef2 lift = new LiftDef2();
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

The following program was created to test the movement of the phone servo and the ability of the phone to recognize the different minerals:

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
DriveDistanceDef distance1 = new DriveDistanceDef();

boolean phone_locked;
boolean phone_in;
boolean phone_out;
int phonecyclecount;

private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";

/*
 * IMPORTANT: You need to obtain your own license key to use Vuforia. The string below with which
 * 'parameters.vuforiaLicenseKey' is initialized is for illustration only, and will not function.
 * A Vuforia 'Development' license key, can be obtained free of charge from the Vuforia developer
 * web site at https://developer.vuforia.com/license-manager.
 *
 * Vuforia license keys are always 380 characters long, and look as if they contain mostly
 * random data. As an example, here is a example of a fragment of a valid key:
 * ... ylglzTqZ4mWjk9wd3cZO9T1axEqzuhxoGlfOOI2dRzKS4T0hQ8kT ...
 * Once you've obtained a license key, copy the string from the Vuforia web site
 * and paste it in to your code onthe next line, between the double quotes.
 */

// Variables to be used for later

private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/
Mzz7Zvp35rYD6lavfkgCMZ0DVAXhBv4J5LlrGlVXYfhhS1NklTGPdqVRW2aBmKLwctHzaztzycau3g//
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnFkYGcE-
Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here

```
/**  
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
 * localization engine.  
 */  
private VuforiaLocalizer vuforia;  
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFOBJECTDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFOBJECTDetector.Parameters tfodParameters = new TFOBJECTDetector.Parameters();  
  
private ElapsedTime runtime = new ElapsedTime();  
  
@Override public void runOpMode() throws InterruptedException {  
    String print_val;  
    print_val = "Test";
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
String target_print_val;
target_print_val = "Target";
String turn_print_val;
turn_print_val = "Turn";
String hanging_print_val;
hanging_print_val = "Not Hanging";
/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);
// lift.init(hardwareMap, this);

distance1.init(hardwareMap, this);
drive.init(hardwareMap, this);
turn.init(hardwareMap, this);
// touch.init(hardwareMap, this);

// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Autonomous Position");

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

telemetry.addData("3", "before setting up tfod");
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
initVufor();

if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3", "setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3", "back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}

// Send telemetry message to indicate successful Encoder reset

telemetry.addData("2", "Waiting to start");
telemetry.update();

// wait for the start button to be pressed.

waitForStart();

int location = 0;

// init phone once
robot.phone_position = .07;
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
robot.phone.setPosition(robot.phone_position);

//get the target value
if (tfod != null) {
    tfod.activate();
}
phone_locked = true;
findMineral();
telemetry.addData("7", "Gold Location is %d", robot.gold_position);
telemetry.addData("6", "Option going to move the gold mineral");

if (tfod != null) {
    tfod.shutdown();
}

telemetry.addData("Path", "TargetTest Complete");
telemetry.update();
robot.wait(20);
idle(); // Always call idle()
}

//find the position of the gold mineral
private void findMineral() {
    double mineral_location;
    double mineral_location_silver1;
    double mineral_location_silver2;
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
phone_in = true;
phone_locked = false;
boolean just_switched = false;
robot.phone_position = .06;
String label = "outward";
robot.phone.setPosition(robot.phone_position);

telemetry.addData("2", "in findMineral");
if (tfod != null) {
    // getUpdatedRecognitions() will return null if no new information is available since
    // the last time that call was made.

    int count = 0;
    int found = 0;

    mineral_location = -1;
    mineral_location_silver1 = -1;
    mineral_location_silver2 = -1;
    int goldMineralX = -1;
    int silverMineral1X = -1;
    int silverMineral2X = -1;
    // loop through to find the three targets

    while ((goldMineralX == -1 && silverMineral1X == -1 && silverMineral2X == -1) || (!
isStopRequested() && opModelsActive())) {

        List<Recognition> updatedRecognitions = tfod.getUpdatedRecognitions();

        // if data - check it
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
if (updatedRecognitions != null) {
    // telemetry.addData("# Object Detected", updatedRecognitions.size());
    count++;
    // telemetry.addData("7", "track is %d", count);
    int item_count = 0;
    //phone_locked = false;
    if (updatedRecognitions.size() > 0) {

        for (Recognition recognition : updatedRecognitions) {
            label = recognition.getLabel();
            item_count++;
            telemetry.addData("7", "checking - %s found %d", label , found);
            telemetry.update();
            if (recognition.getLabel().equals(LABEL_GOLD_MINERAL) && goldMineralX == -1) {
                found++;
                telemetry.addData("7", "found gold");
                telemetry.update();
                goldMineralX = (int) recognition.getLeft();
                mineral_location = robot.phone_position;
                robot.wait(1);
            }
            if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral1X == -1) {
                found++;
                telemetry.addData("7", "found silver");
                telemetry.update();
                silverMineral1X = (int) recognition.getLeft();
                mineral_location_silver1 = robot.phone_position;
                robot.wait(1);
            }
        }
    }
}
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
    } else if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral2X ==  
-1 &&  
        mineral_location_silver1 != robot.phone_position) {  
    found++;  
    telemetry.addData("7", "found silver");  
    telemetry.update();  
    silverMineral2X = (int) recognition.getLeft();  
    mineral_location_silver2 = robot.phone_position;  
    robot.wait(1);  
}  
// telemetry.addData("4", "count %d - internal count %d", updatedRecognitions.size(),  
item_count);  
// telemetry.update();  
if (item_count == updatedRecognitions.size()) break;  
}  
}  
}  
}  
telemetry.addData("er","should move phone %b %b %b", phone_in,  
phone_out,phone_locked);  
telemetry.update();  
// move the phone box to find the target  
if (phone_in && !phone_locked) {  
    //move phone (bottom is 1)  
    if (robot.phone_position > .41) {  
        phone_in = false;  
        phone_out = true;  
    } else {
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
        robot.phone_position += .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;

    }

} else if (phone_out && !phone_locked) {
    // phone (top position is 0)
    if (robot.phone_position < .19) {
        phone_in = true;
        phone_out = false;
    } else {
        robot.phone_position -= .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
}

if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 25000) {
        phone_locked = false;
        phonecyclecount = 0;
    }
}

telemetry.addData("8", "1Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1, mineral_location_silver2);
// telemetry.update();
//robot.wait(5);
idle(); // Always call idle()
// if all items found then break out
if (found == 3) break;
} //end of while

if (mineral_location != -1) {
    if (mineral_location == .18) {
        telemetry.addData("Gold Mineral Position", "Left");
        robot.gold_position = 1;
    } else if (mineral_location == .42) {
        telemetry.addData("Gold Mineral Position", "Right");
        robot.gold_position = 3;
    } else if (mineral_location == .3){
        telemetry.addData("Gold Mineral Position", "Center");
        robot.gold_position = 2;
    }
}
// telemetry.addData("8", "2Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1, mineral_location_silver2);
telemetry.update();
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
    }
    // robot.wait(20);
}

public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period.currentTimeMillis();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            Thread.sleep(remaining);
        } catch (InterruptedException e) {
            Thread.currentThread().interrupt();
        }
    }

    // Reset the cycle clock for the next pass.
    period.reset();
}

private void initVufor() {
/*
 * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
 */
}
```

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:

```
VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

parameters.vuforiaLicenseKey = VUFORIA_KEY;
parameters.cameraDirection = VuforiaLocalizer.CameraDirection.BACK;

// Instantiate the Vuforia engine
vuforia = ClassFactory.getInstance().createVuforia(parameters);

}

/*
* Initialize the Tensor Flow Object Detection engine.
*/

private void initTfod() {

    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());

    TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters
(tfodMonitorViewId);
    tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LA-
BEL_SILVER_MINERAL);

}
}
```

This useful program showed that the mineral's position will be determined correctly.

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

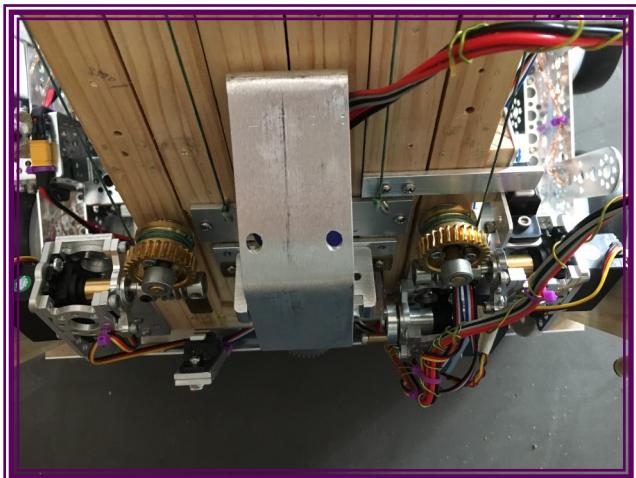
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

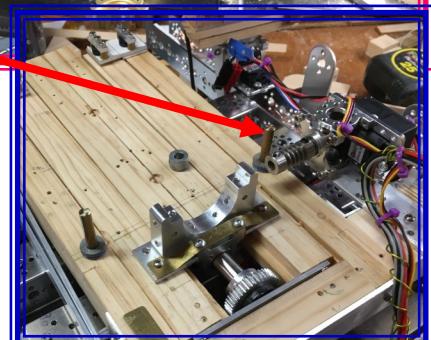
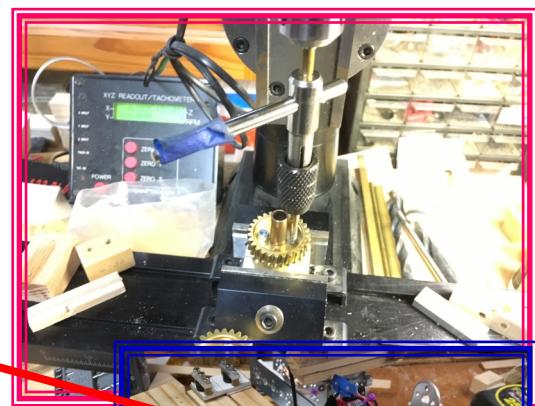
Date: December 9, 2018

Process:

We modified the second level of the lift to use worm gears. This will keep the lift at the position height that it was raised or lowered to. We had to make a new spool that will be attached to the gear that works with the worm gear. We hope that this will be helpful.



Pictured are the new worm gear and sprocket with spool. A machine lathe was used to create the new spool. A milling machine is used to drill the holes to connect the gear and spool together. (New worm gear)



During our test of the modifications to the second level of the lift we found that the worm gear did keep the lift from lowering when there is no power apply to servo. We were hoping that changing to the worm gear would be helpful, but it was way too slow in the raising and lowering of the lift so we quickly changed back to using the spools like a wrench.

We have added a lift lock which hold down the second level of lift so that we can hang as well as a latch-

Signature : Jessica Anderson

Date: Dec. 9, 2018

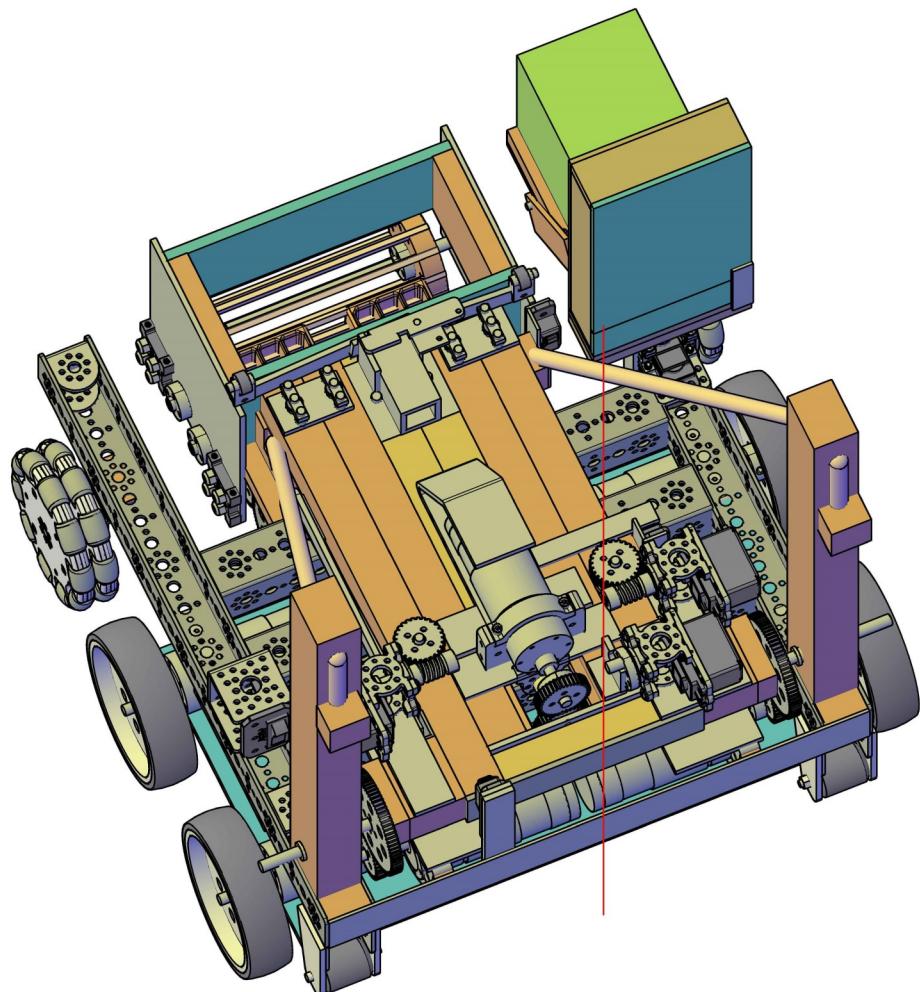
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:



This design is showing the use of a worm gear to raise and lower the outer section of the linear slide.

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

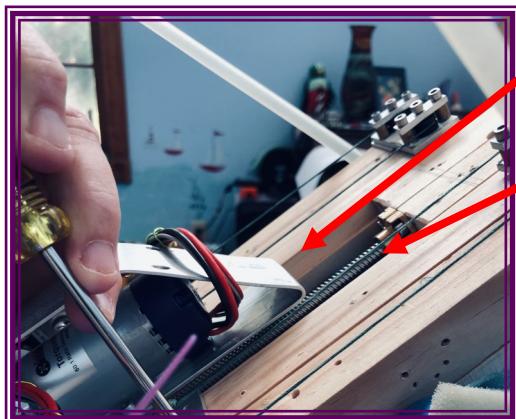
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

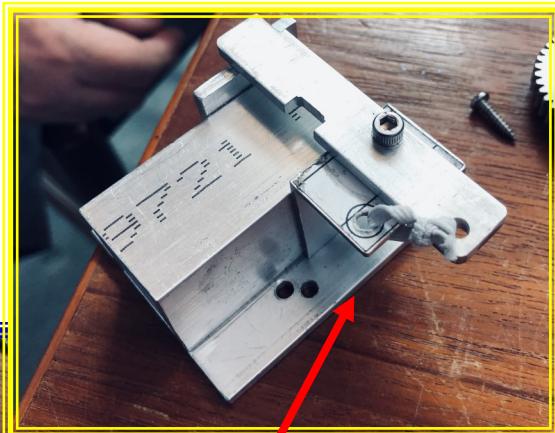
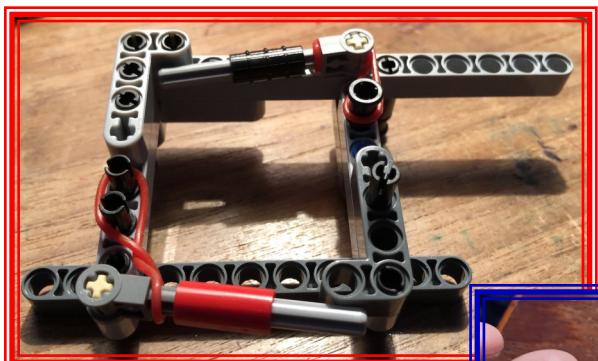
Process:

ing mechanism. If we are able to get latched to the lander we can lift at the end game. This will take some practice and strategic driving. In addition to adding the latch we added a skid over the worm gears motor so we do not hang-up on the lander while pulling the robot off the floor.

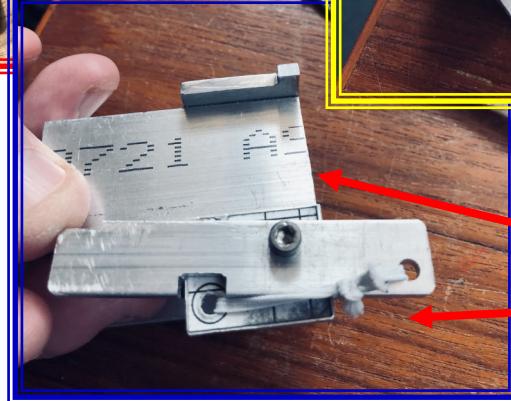


Skid so the robot does not hang up on the worm gear's motor.

A LEGO latch was created as a prototype to determine what was needed and the real one is made from aluminum with the help of our Mentor Louis Achee. Thank you Mr. Louis!!!



The prototype shows the top latch which will hold the robot. The bottom latch was found not to be required, so it was not included.



Latch closed

Latch open

Elastic holds the latch closed

Signature : Jessica Anderson

Date: Dec. 9, 2018

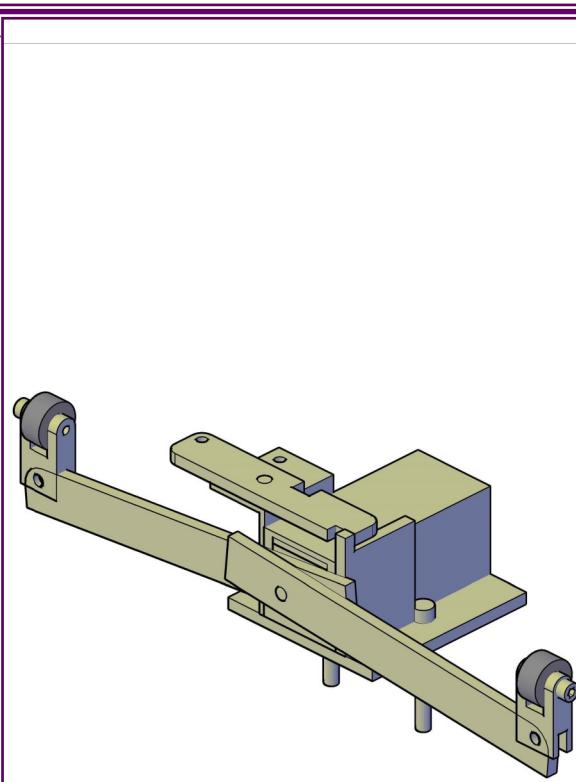
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

Process:



This is the latching device design where you push up to latch onto the lander

Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

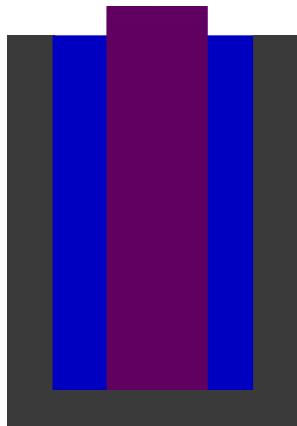
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 9, 2018

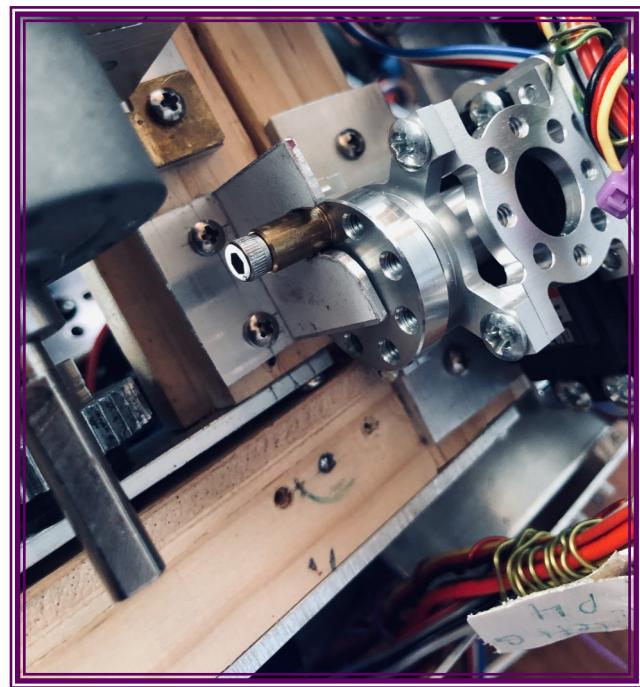
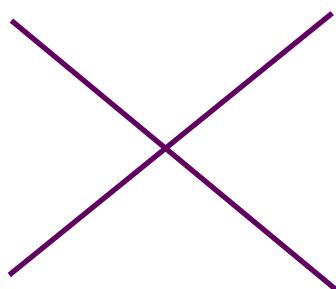
Process:

As mentioned before a lift lock was added to the out layer since the lower when there is no power being supplied to the servo. We needed a way to hold the second layer of the lift down while trying to pull the robot up off the ground.



- The gray item is the out part of the lift and is stationary
- The blue item is the second layer of the lift and is moved using a string and uses gravity to lower down into position. This is the layer that is locked when we need to pull the robot off the mat.
- The purple item is the middle layer of the lift and is moved using a linear actuator screw and gear attached to a motor.

If we did not have a latch then when the robot latched onto the lander the middle layer would pull everything up while it was pulling itself down again and the robot will stay on the ground.



Signature : Jessica Anderson

Date: Dec. 9, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: December 14-15, 2018

Purpose of the Activity:

Final touches for the Meet

Meet

Post Meet Discussion

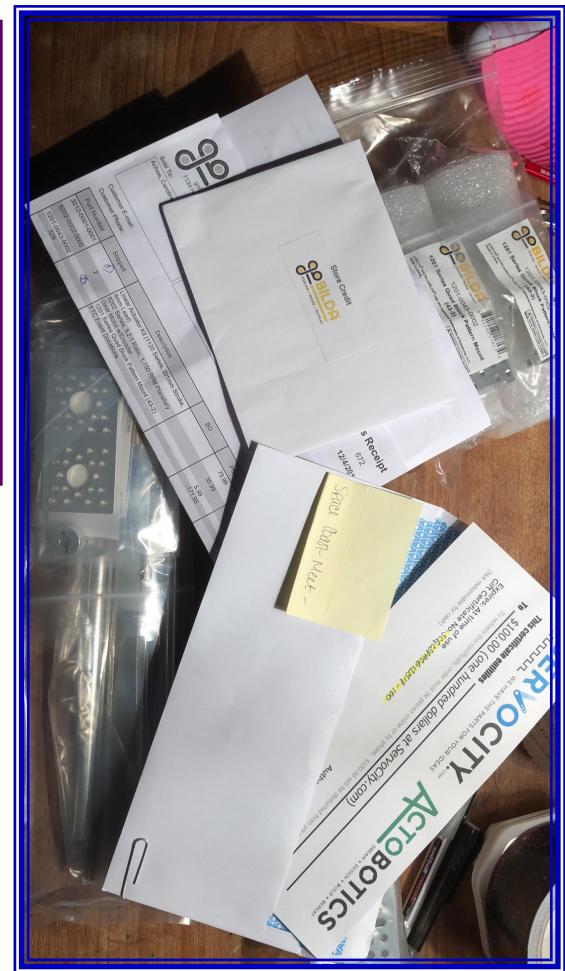
Process:

We would like to **thank** ServoCity and AndyMark for some cool giveaways.



The clips were used to create 6 packets to give to the first 6 teams that completed both Hardware and Software Inspections. We had 20 teams participate at the meet. It was an early day as we started inspections at 7:00am. We completed 5 matches per team and finished ahead of time.

We used the items from ServoCity as part of the door prizes. The best was the two \$100.00 gift certificates!! Yay!!



Signature: Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

We transformed the Parish Center from being empty... to filled with tables ready for teams to arrive:



Signature: Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

We had a sleepover after getting the gym and parish center ready for the meet as we were up at 5:00am to greet the teams starting at 6:30 that morning. The night before we charged all the robot batteries and updated the phones with the latest copy of the software. It was a lot of fun at the sleepover. Following are the programs we had to support the meet:

Our Hardware Definition program:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.hardware.bosch.BNO055IMU;
import com.qualcomm.hardware.bosch.JustLoggingAccelerationIntegrator;
import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cRangeSensor;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.CRServo;
import com.qualcomm.robotcore.hardware.ColorSensor;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.DigitalChannel;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.hardware.I2cAddr;
import com.qualcomm.robotcore.hardware.Servo;
import com.qualcomm.robotcore.hardware.TouchSensor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotorSimple.Direction.REVERSE;
import static java.lang.Thread.currentThread;
import static java.lang.Thread.sleep;

/**
 * This is NOT an opmode.
 * <p>
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
 * <p>
 * This hardware class assumes the following device names have been configured on the robot:
 * Note: All names are lower case and some have single spaces between words.
 */
public class PrinceCharlesBaDazzle {

    // setup for calculation of the how far to move

    static final double P_TURN_COEFF      = 0.1;    // Larger is more responsive, but also less stable
    static final double HEADING_THRESHOLD   = 1;     // As tight as we can make it with an integer
    gyro

    static final double COUNTS_PER_MOTOR_REV  = 1680; // eg: AndyMark Motor Encoder
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
static final double DRIVE_GEAR_REDUCTION = 1; // This is < 1.0 if geared UP  
static final double WHEEL_DIAMETER_INCHES = 4.0; // For figuring circumference  
static final double GEAR_DIAMETER_INCHES = 1.0; // For figuring circumference  
  
static final double COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *  
DRIVE_GEAR_REDUCTION) /  
(WHEEL_DIAMETER_INCHES * 3.1415);  
static final double LIFT_COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *  
DRIVE_GEAR_REDUCTION) /  
(GEAR_DIAMETER_INCHES * 3.1415);  
  
static final double DRIVE_SPEED = 0.5;  
static final double DRIVE_SPEED1 = 0.9;  
static final double STONE_DRIVE_SPEED = 0.2;  
static final double TURN_SPEED = 0.3;  
static final double TOUCH_SPEED = 0.1;  
static final double LIFT_SPEED = 0.2;  
static final double STOP_SPEED = 0;  
  
int floor_color_value;  
int position_option = 0;  
int position_side = 0;  
int turn_option = 0;  
int path_option = 0;  
int find_target = -1;  
int hanging = -1;  
int gold_position = 0;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int count = 0;  
// Driver motors  
DcMotor rightdrive;  
DcMotor leftdrive;  
DcMotor righttwo;  
DcMotor lefttwo;  
float right = 0;  
float left = 0;  
  
// Distance control  
  
ModernRoboticsI2cRangeSensor distance;  
  
//motors to lift and lower  
DcMotor right_liftdrive;  
DcMotor left_liftdrive;  
float lift_power = 0;  
  
// move the collector in/out up/down  
DcMotor extenderdrive;  
// pick up the minerals  
DcMotor collectordrive;  
float extender_power = 0;  
float collector_power = 0;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// open position
double right_position = 0;
double left_position = .5;

// Side phone
Servo phone;
double phone_position = 0.26;
CRServo left_wrench;
CRServo right_wrench;
double wrench_power = 0;

Servo lock;
double lock_power = 0;
Servo marker;
double marker_power = 0;

DigitalChannel lower_stop;
DigitalChannel upper_stop;
DigitalChannel extender_lower;
DigitalChannel extender_upper;
DigitalChannel wrench_lower;
DigitalChannel wrench_upper;
// Our sensors, motors, and other devices go here, along with other long term state
BNO055IMU imu;
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// State used for updating telemetry
Orientation angles;
Acceleration gravity;

/* local OpMode members. */
HardwareMap hardwareMap = null;
private ElapsedTime period = new ElapsedTime();

// Private Members
private LinearOpMode myOpMode;

/* Constructor */
public PrinceCharlesBaDazzle() {

}

/* Initialize standard Hardware interfaces */
public void init(HardwareMap ahwMap, int option) {
    // Save reference to Hardware map
    hardwareMap = ahwMap;

    /*
     * Use the hardwareMap to get the dc motors and servos by name.
     * Note that the names of the devices must match the names used
     * when you configured your robot and created the configuration file.
     */
}
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// if option is one define hardware... otherwise put it start position..  
if (option == 1) {  
    // start of drive train definitions  
    rightdrive = hardwareMap.dcMotor.get("right_drive");  
    leftdrive = hardwareMap.dcMotor.get("left_drive");  
    leftdrive.setDirection(REVERSE);  
    righttwo = hardwareMap.dcMotor.get("righttwo");  
    lefttwo = hardwareMap.dcMotor.get("lefttwo");  
    lefttwo.setDirection(REVERSE);  
    // end of drive train definitions  
  
    right_liftdrive = hardwareMap.dcMotor.get("right_lift");  
    right_liftdrive.setDirection(REVERSE);  
    left_liftdrive = hardwareMap.dcMotor.get("left_lift");  
    right_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);  
    left_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);  
  
    collectordrive = hardwareMap.dcMotor.get("collector");  
  
    extenderdrive = hardwareMap.dcMotor.get("extender");  
    extenderdrive.setDirection(REVERSE);  
    // stop the lifter when down flat  
    lower_stop = hardwareMap.get(DigitalChannel.class, "lower_stop");  
    upper_stop = hardwareMap.get(DigitalChannel.class, "upper_stop");  
    // Center piece  
    wrench_lower = hardwareMap.get(DigitalChannel.class, "wrench_lower");  
    wrench_upper = hardwareMap.get(DigitalChannel.class, "wrench_upper");
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// outer piece
extender_lower = hardwareMap.get(DigitalChannel.class, "extender_lower");
extender_upper = hardwareMap.get(DigitalChannel.class, "extender_upper");
// set the digital channel to input.
lower_stop.setMode(DigitalChannel.Mode.INPUT);
upper_stop.setMode(DigitalChannel.Mode.INPUT);
wrench_lower.setMode(DigitalChannel.Mode.INPUT);
wrench_upper.setMode(DigitalChannel.Mode.INPUT);
extender_lower.setMode(DigitalChannel.Mode.INPUT);
extender_upper.setMode(DigitalChannel.Mode.INPUT);

phone = hardwareMap.servo.get("phone");
left_wrench = hardwareMap.get(CRServo.class,"left_wrench");
right_wrench = hardwareMap.get(CRServo.class,"right_wrench");

lock = hardwareMap.servo.get("lock");
marker = hardwareMap.servo.get("marker");

// set up the guidance hardware
// Set up the parameters with which we will use our IMU. Note that integration
// algorithm here just reports accelerations to the logcat log; it doesn't actually
// provide positional information.
BNO055IMU.Parameters parameters = new BNO055IMU.Parameters();
parameters.angleUnit      = BNO055IMU.AngleUnit.DEGREES;
parameters.accelUnit      = BNO055IMU.AccelUnit.METERS_PERSEC_PERSEC;
parameters.calibrationDataFile = "BNO055IMUCalibration.json"; // see the calibration sample
opmode
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
parameters.loggingEnabled    = true;
parameters.loggingTag       = "IMU";
parameters.accelerationIntegrationAlgorithm = new JustLoggingAccelerationIntegrator();

// Retrieve and initialize the IMU. We expect the IMU to be attached to an I2C port
// on a Core Device Interface Module, configured to be a sensor of type "AdaFruit IMU",
// and named "imu".
imu = hardwareMap.get(BNO055IMU.class, "imu");
imu.initialize(parameters);

} else {

    // set the phones
    phone_position = .26;
    phone.setPosition(phone_position);

}

}

public void wait(int sec) {

    for (int i = 0; i < 2 * sec; i++) {
        try {
            sleep(500);
        } catch (InterruptedException e) {
            currentThread().interrupt();
            break;
        }
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
        }
    }
}

/**
*
* waitForTick implements a periodic delay. However, this acts like a metronome with a regular
* periodic tick. This is used to compensate for varying processing times for each cycle.
* The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
*
* @param periodMs Length of wait cycle in mSec.
*/
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long) period.currentTimeMillis();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            sleep(remaining);
        } catch (InterruptedException e) {
            currentThread().interrupt();
        }
    }

    // Reset the cycle clock for the next pass.
    period.reset();
}
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
    }  
}
```

Following is the Autonomous Program

The path we took was on the Depot side and looking for the gold mineral

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.
This will work on only the blue side.*

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE*

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

*OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;
import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.tfod.Recognition;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

@Autonomous(name = "FF: AutoPosition", group = "Auto")
//@Disabled
public class AutoPosition extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
TurnDef2 turn = new TurnDef2();

LiftDef2 lift = new LiftDef2();
DriveDistanceDef distance1 = new DriveDistanceDef();

boolean phone_locked;
boolean phone_in;
boolean phone_out;
int phonecyclecount;

private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";

/*
 * IMPORTANT: You need to obtain your own license key to use Vuforia. The string below with which
 * 'parameters.vuforiaLicenseKey' is initialized is for illustration only, and will not function.
 * A Vuforia 'Development' license key, can be obtained free of charge from the Vuforia developer
 * web site at https://developer.vuforia.com/license-manager.
 *
 * Vuforia license keys are always 380 characters long, and look as if they contain mostly
 * random data. As an example, here is a example of a fragment of a valid key:
 * ... ylgIzTqZ4mWjk9wd3cZO9T1axEqzuhxoGlfOOI2dRzKS4T0hQ8kT ...
 * Once you've obtained a license key, copy the string from the Vuforia web site
 * and paste it in to your code onthe next line, between the double quotes.
 */
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

// Variables to be used for later

```
private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-
Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/
MZZ7vp35rYD6lavfkgCMZ0DVAXhBv4J5LlrGlVXYfhhS1NkITGPDqVRW2aBmKLwctHZaztzycau3g// 
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnPnFkYGcE-
Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMijIrFtBH+Uj/vVQzkJDx8QqH2EEed+AM+WKq"; // Insert your own key here

/**
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia
 * localization engine.
 */
private VuforiaLocalizer vuforia;
/**
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object
 * Detection engine.
 */
private TFObjectDetector tfod;

// Leave argument list empty if you want to disable the camera monitor view.
TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
@Override public void runOpMode() throws InterruptedException {
    String print_val;
    print_val = "Test";
    String target_print_val;
    target_print_val = "Target";
    String turn_print_val;
    turn_print_val = "Turn";
    String hanging_print_val;
    hanging_print_val = "Not Hanging";
    /*
     * Initialize the drive system variables.
     * The init() method of the hardware class does all the work here
     */
    robot.init(hardwareMap, 1);
    // lift.init(hardwareMap, this);

    distance1.init(hardwareMap, this);
    drive.init(hardwareMap, this);
    turn.init(hardwareMap, this);
    // touch.init(hardwareMap, this);

    // Send telemetry message to signify robot waiting;
    telemetry.addData("Status", "Autonomous Position");

    String startDate;

    startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Send telemetry message to indicate successful Encoder reset
telemetry.addData("1", "Starting drive position at Left %7d - Right %7d Lefttwo %7d - Righttwo %7d",
    robot.leftdrive.getCurrentPosition(),
    robot.rightdrive.getCurrentPosition(),
    robot.lefttwo.getCurrentPosition(),
    robot.righttwo.getCurrentPosition());

robot.position_option = 0;

while (robot.position_side == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.position_side = 1;
    }else if (gamepad2.y) {
        robot.position_side = 2;
    }else {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.addData("5","Select D2 - x for Red Side cnt - %d", robot.count);
telemetry.addData("6","Select D2 - y for Blue Side");
telemetry.update();
}
robot.wait(1);
}
robot.wait(1);
robot.count = 0;
while (robot.position_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.position_option = 3;
        print_val = "Crater";
    } else if (gamepad2.b) {
        robot.position_option = 1;
        print_val = "Depot";
    } else {
        telemetry.addData("5", "Select D2 - a for position 1 crater cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - b for position 2 depot");
        telemetry.update();
        robot.wait(1);
    }
}
// turning right or left
robot.wait(1);
robot.count = 0;
while (robot.turn_option == 0 && (!isStopRequested() || opModelsActive())) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.count++;
if (gamepad2.x) {
    robot.turn_option = 1;
    turn_print_val = "Right";
} else if (gamepad2.y) {
    robot.turn_option = 2;
    turn_print_val = "Left";
} else {
    telemetry.addData("5", "Select D2 - x for turning right cnt - %d", robot.count);
    telemetry.addData("6", "Select D2 - y for turning left");
    telemetry.update();
    robot.wait(1);
}
// going for the target mineral or not
robot.wait(1);
robot.count = 0;
while (robot.find_target == -1 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.find_target = 1;
        target_print_val = "Find Target";
    } else if (gamepad2.b) {
        robot.find_target = 0;
        target_print_val = "Do Not Find Target";
    } else {
        telemetry.addData("5", "Select D2 - a for find mineral target cnt - %d", robot.count);
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.addData("6", "Select D2 - b for do not find mineral target");
telemetry.update();
robot.wait(1);

}

}

robot.find_target = 1;
target_print_val = "Find Target";
// robot hanging or not
robot.wait(1);
robot.count = 0;
while (robot.hanging == -1 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.hanging = 1;
        hanging_print_val = "Hanging";
    } else if (gamepad2.y) {
        robot.hanging = 0;
        hanging_print_val = "Not Hanging";
    } else {
        telemetry.addData("5", "Select D2 - x for Hanging cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - y for NOT Hanging");
        telemetry.update();
        robot.wait(1);
    }
}
telemetry.addData("3", "before setting up tfod");
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
initVufor();

if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3", "setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3", "back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}

// Send telemetry message to indicate successful Encoder reset
if (robot.position_side == 1) {
    telemetry.addData("1", "All setup at Position Red Side %s - %s - %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
} else {
    telemetry.addData("1", "All setup at Position Blue Side %s - %s - %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
}
telemetry.addData("1", "Position %d - %d - %d",
    robot.turn_option + robot.position_option, robot.turn_option, robot.position_option);
telemetry.addData("2", "Waiting to start");
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.update();

// wait for the start button to be pressed.

waitForStart();

int location = 0;

// init phone
robot.phone_position = .08;
robot.phone.setPosition(robot.phone_position);

robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

if (robot.hanging == 1) {
    // get down from the lander
    // lift.liftmove(1, .25);
}

robot.path_option = robot.turn_option + robot.position_option;
if (robot.find_target == 1) {
    // put phone servo in start position
    telemetry.addData("6", " moving phone");
    robot.init(hardwareMap, 2);
    //get the target value
    if (tfod != null) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
tfod.activate();
}
phone_locked = true;
findMineral();
telemetry.addData("7", "Location is %d", robot.gold_position);

telemetry.addData("6", "Option going to move the gold mineral");
if (robot.gold_position == 1){
    // turn to the left and move forward
    // turn left
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 2, 2.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 48in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 37, 2.5);
        //turn right
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 1, 2.5);
        //forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.0);

    } else {
        //on the crater side
        telemetry.addData("6", "Option going forward 31in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
    }
} else if (robot.gold_position == 2) {
    // going straight
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 48in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 52, 4.5);
    } else {
        // on the crater side
        telemetry.addData("6", "Option going forward 28in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 39, 5.5);
    }
} else if (robot.gold_position == 3) {
    // turning right and then going straight
    //turn right
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5.1, 3.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 35in ");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 35, 2.5);
        // turn left
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11.2, 2.5);
        //forward towards the depot
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.5);
} else {
    // on the crater side
    telemetry.addData("6", "Option going forward 31in");
    //forward towards the crater
    drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
}
}

telemetry.update();

robot.marker_power = .7;
robot.marker.setPosition(robot.marker_power);
robot.wait(2);
robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

} else if (robot.path_option == 2){
    telemetry.addData("6", "Depot Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
    //forward
    drive.encoder2Drive(robot.DRIVE_SPEED1, 32, 5.5);
    //turn left
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);
//forward
drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
// turn left
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);
//forward
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
// dump off the marker

// backwards towards the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);

} else if (robot.path_option == 3) {
    telemetry.addData("6", "Depot Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,1, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,1, 3.5);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
//forward toward the depot  
drive.encoder2Drive(robot.DRIVE_SPEED, 12, 3.5);  
//dump off the marker  
  
//forward into the crater  
drive.encoder2Drive(robot.DRIVE_SPEED, 5, 3.5);  
//turn right  
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);  
//forward toward the crater  
drive.encoder2Drive(robot.DRIVE_SPEED1, 96, 10.5);  
  
}  
else if (robot.path_option == 4) {  
    telemetry.addData("6", "Crater Side turning Right");  
    telemetry.update();  
    //forward and get past the lander legs  
    drive.encoder2Drive(robot.DRIVE_SPEED, 15, 1.5);  
    //turn right  
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);  
    //forward going past the lander  
    drive.encoder2Drive(robot.DRIVE_SPEED1, 29, 5.5);  
    //turn left  
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);  
    //forward toward the wall  
    drive.encoder2Drive(robot.DRIVE_SPEED1, 26, 5.5);  
  
}  
else if (robot.path_option == 5) {  
    telemetry.addData("6", "Crater Side turning Left");
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.update();
//forward and get past the lander legs
drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
// turn left
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
//forward and go to the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 60, 8.5);
// turn right
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,1, 3.5);
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
//dump off the marker

// backwards toward the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);
} else {
// nearest to the audience
// the distance is 23 15/16 but allow for role from stone to parking zone
telemetry.addData("6", " No Valid option picked");
telemetry.update();

}

if (tfod != null) {
tfod.shutdown();
}
robot.wait(20);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.addData("Path", "Autonomous Complete");
telemetry.update();

idle(); // Always call idle()
}

//find the position of the gold mineral
private void findMineral() {
    double mineral_location;
    double mineral_location_silver1;
    double mineral_location_silver2;
    phone_in = true;
    phone_locked = false;
    boolean just_switched = false;
    robot.phone_position = .06;
    String label = "outward";
    robot.phone.setPosition(robot.phone_position);

    telemetry.addData("2", "in findMineral");
    if (tfod != null) {
        // getUpdatedRecognitions() will return null if no new information is available since
        // the last time that call was made.
        int count = 0;
        int found = 0;

        mineral_location = -1;
        mineral_location_silver1 = -1;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
mineral_location_silver2 = -1;
int goldMineralX = -1;
int silverMineral1X = -1;
int silverMineral2X = -1;

// loop through to find the three targets

while ((goldMineralX == -1 && silverMineral1X == -1 && silverMineral2X == -1) || (!
isStopRequested() && opModelsActive())) {

List<Recognition> updatedRecognitions = tfod.getUpdatedRecognitions();

// if data - check it
if (updatedRecognitions != null) {
    // telemetry.addData("# Object Detected", updatedRecognitions.size());
    count++;
    // telemetry.addData("7", "track is %d", count);
    int item_count = 0;
    //phone_locked = false;
    if (updatedRecognitions.size() > 0) {

        for (Recognition recognition : updatedRecognitions) {
            label = recognition.getLabel();
            item_count++;
            telemetry.addData("7", "checking - %s found %d", label , found);
            telemetry.update();
            if (recognition.getLabel().equals(LABEL_GOLD_MINERAL) && goldMineralX == -1) {
                found++;
            }
        }
    }
}
}
```

Signature: Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.addData("7", "found gold");
telemetry.update();
goldMineralX = (int) recognition.getLeft();
mineral_location = robot.phone_position;
robot.wait(1);
}
if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral1X == -1) {
    found++;
    telemetry.addData("7", "found silver");
    telemetry.update();
    silverMineral1X = (int) recognition.getLeft();
    mineral_location_silver1 = robot.phone_position;
    robot.wait(1);
} else if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral2X == -1
&&
        mineral_location_silver1 != robot.phone_position) {
    found++;
    telemetry.addData("7", "found silver");
    telemetry.update();
    silverMineral2X = (int) recognition.getLeft();
    mineral_location_silver2 = robot.phone_position;
    robot.wait(1);
}
// telemetry.addData("4", "count %d - internal count %d", updatedRecognitions.size(),
item_count);
// telemetry.update();
if (item_count == updatedRecognitions.size()) break;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
        }
    }
}

telemetry.addData("er","should move phone %b %b %b", phone_in, phone_out, phone_locked);
telemetry.update();
// move the phone box to find the target
if (phone_in && !phone_locked) {
    //move phone (bottom is 1)
    if (robot.phone_position > .41) {
        phone_in = false;
        phone_out = true;
    } else {
        robot.phone_position += .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
} else if (phone_out && !phone_locked) {
    // phone (top position is 0)
    if (robot.phone_position < .19) {
        phone_in = true;
        phone_out = false;
    } else {
        robot.phone_position -= .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
}

if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 25000) {
        phone_locked = false;
        phonecyclecount = 0;
    }
}

telemetry.addData("8", "1Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1,
mineral_location_silver2);
// telemetry.update();
//robot.wait(5);
idle(); // Always call idle()
// if all items found then break out
if (found == 3) break;
}//end of while

if (mineral_location != -1) {
    if (mineral_location == .18) {
        telemetry.addData("Gold Mineral Position", "Left");
        robot.gold_position = 1;
    } else if (mineral_location == .42) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.addData("Gold Mineral Position", "Right");
robot.gold_position = 3;
} else if (mineral_location == .3){
    telemetry.addData("Gold Mineral Position", "Center");
    robot.gold_position = 2;
}
// telemetry.addData("8", "2Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1,
mineral_location_silver2);
telemetry.update();
}
// robot.wait(20);
}

public void waitForTick(long periodMs) {
    long remaining = periodMs - (long)period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            Thread.sleep(remaining);
        }
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
    } catch (InterruptedException e) {
        Thread.currentThread().interrupt();
    }
}

// Reset the cycle clock for the next pass.
period.reset();
}

private void initVuforia() {
/*
 * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
 */
VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

parameters.vuforiaLicenseKey = VUFORIA_KEY;
parameters.cameraDirection = VuforiaLocalizer.CameraDirection.BACK;

// Instantiate the Vuforia engine
vuforia = ClassFactory.getInstance().createVuforia(parameters);

}
/***
 * Initialize the Tensor Flow Object Detection engine.
 */
private void initTfod() {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(
    "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());

TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters
(tfodMonitorViewId);
tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);
tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LA-
BEL_SILVER_MINERAL);

}
```

The Autonomous Program uses several function to accomplish it's task. Following are the different functions. First is DefDrive which is used to drive the robot forward or backwards:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.util.ElapsedTime;

/**
 * This is NOT an opmode.
 *
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
*  
* This hardware class assumes the following device names have been configured on the robot:  
* Note: All names are lower case and some have single spaces between words.  
  
*/  
public class DriveDef2  
{  
    /* Declare OpMode members. */  
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Prince Charles's hardware  
  
    private ElapsedTime runtime = new ElapsedTime();  
  
    // Private Members  
    private LinearOpMode myOpMode;  
  
    /* local OpMode members. */  
    HardwareMap hardwareMap = null;  
    private ElapsedTime period = new ElapsedTime();  
  
    /* Constructor */  
    public DriveDef2(){  
  
    }  
  
    /* Initialize standard Hardware interfaces */  
    public void init(HardwareMap ahwMap, LinearOpMode opMode) {  
        // Save reference to Hardware map
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
hardwareMap = ahwMap;

myOpMode = opMode;
/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);

/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */
}

/**
 *
 * waitForTick implements a periodic delay. However, this acts like a metronome with a regular
 * periodic tick. This is used to compensate for varying processing times for each cycle.
 * The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
 *
 * @param periodMs Length of wait cycle in mSec.
 */
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period.currentTimeMillis();
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// sleep for the remaining portion of the regular cycle period.  
if (remaining > 0) {  
    try {  
        Thread.sleep(remaining);  
    } catch (InterruptedException e) {  
        Thread.currentThread().interrupt();  
    }  
}  
  
// Reset the cycle clock for the next pass.  
period.reset();  
}  
  
/*  
 * Method to perform a relative move, based on encoder counts.  
 * Encoders are not reset as the move is based on the current position.  
 * Move will stop if any of three conditions occur:  
 * 1) Move gets to the desired position  
 * 2) Move runs out of time  
 * 3) Driver stops the opmode running.  
 */  
public void encoder2Drive(double speed, double inches,  
                         double timeoutS) {  
    int newLeftTarget;  
    int newRightTarget;  
    int newLefttwoTarget;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int newRighttwoTarget;

robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Determine new target position, and pass to motor controller
newLeftTarget = robot.leftdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
newRightTarget = robot.rightdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
robot.leftdrive.setTargetPosition(newLeftTarget);
robot.rightdrive.setTargetPosition(newRightTarget);
newLefttwoTarget = robot.lefttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
newRighttwoTarget = robot.righttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH);
robot.lefttwo.setTargetPosition(newLefttwoTarget);
robot.righttwo.setTargetPosition(newRighttwoTarget);

// Turn On RUN_TO_POSITION
robot.leftdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.righttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);

// reset the timeout time and start motion.
runtime.reset();
robot.leftdrive.setPower((float)(speed));
robot.rightdrive.setPower((float)(speed));
robot.lefttwo.setPower((float)(speed));
robot.righttwo.setPower((float)(speed));

//telemetry.addData("Path1", "Running to %7d :%7d", newLeftTarget, newRightTarget);
// keep looping while we are still active, and there is time left, and both motors are running.
// keep looping while we are still active, and there is time left, and both motors are running.
while ((runtime.seconds() < timeoutS) ||
       ( robot.leftdrive.isBusy() && robot.rightdrive.isBusy() &&
         robot.lefttwo.isBusy() && robot.righttwo.isBusy() )) {
    myOpMode.telemetry.addData("3", "new left %d", newLeftTarget);
    myOpMode.telemetry.addData("3", "new right %d", newRightTarget);
    myOpMode.telemetry.addData("3", "new left %d", newLefttwoTarget);
    myOpMode.telemetry.addData("3", "new right %d", newRighttwoTarget);
    myOpMode.telemetry.addData("3", "curr left %d", robot.lefttwo.getCurrentPosition());
    myOpMode.telemetry.addData("3", "curr right %d", robot.righttwo.getCurrentPosition());
    myOpMode.telemetry.update();
    waitForTick (5);
}

// Stop all motion;
```

Signature: Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.lefttwo.setPower(robot.STOP_SPEED);
robot.righttwo.setPower(robot.STOP_SPEED);
robot.leftdrive.setPower(robot.STOP_SPEED);
robot.rightdrive.setPower(robot.STOP_SPEED);

// Turn off RUN_TO_POSITION
robot.lefttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);

}

}
```

The next function is used to turn the robot left or right:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cGyro;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.util.ElapsedTime;
import com.qualcomm.robotcore.util.Range;

/**
 * This is NOT an opmode.
 *
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
* See AutoBlue and others classes starting with "FF" for usage examples.  
*  
* This hardware class assumes the following device names have been configured on the robot:  
* Note: All names are lower case and some have single spaces between words.  
  
*/  
public class TurnDef2  
{  
    /* Declare OpMode members. */  
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Prince Charles's hardware  
  
    private ElapsedTime runtime = new ElapsedTime();  
  
    // Private Members  
    private LinearOpMode myOpMode;  
  
    /* local OpMode members. */  
    HardwareMap hardwareMap = null;  
    private ElapsedTime period = new ElapsedTime();  
  
    /* Constructor */  
    public TurnDef2(){  
    }  
  
    /* Initialize standard Hardware interfaces */  
    public void init(HardwareMap ahwMap, LinearOpMode opMode) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// Save reference to Hardware map
hardwareMap = ahwMap;

myOpMode = opMode;
/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */
robot.init(hardwareMap, 1);

}

/**
 *
 * @param periodMs Length of wait cycle in mSec.
 */
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
if (remaining > 0) {
    try {
        Thread.sleep(remaining);
    } catch (InterruptedException e) {
        Thread.currentThread().interrupt();
    }
}

// Reset the cycle clock for the next pass.
period.reset();
}

/*
 * Method to perform a relative move, based on encoder counts.
 * Encoders are not reset as the move is based on the current position.
 * Move will stop if any of three conditions occur:
 * 1) Speed negative is forward and positive is backwards
 * 2) leftInches - pass a negative to go forward and positive to backwards
 * 3) rightInches - pass a negative to go forward and positive to backwards
 * 4) direction - 1 - right, 2 - left
 * 5) timeoutS - number of seconds before the function timeout
*/
public void encoder2DriveTurn2(double speed, double Inches,
                               int direction, double timeoutS) {
    int newLeftTarget;
    int newRightTarget;
    int newLefttwoTarget;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
newRightTarget = robot.rightdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH * rightdirection);

robot.leftdrive.setTargetPosition(newLeftTarget);
robot.rightdrive.setTargetPosition(newRightTarget);

// set the front motors
newLefttwoTarget = robot.lefttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH * leftdirection);
newRighttwoTarget = robot.righttwo.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH * rightdirection);

robot.lefttwo.setTargetPosition(newLefttwoTarget);
robot.righttwo.setTargetPosition(newRighttwoTarget);

// Turn On RUN_TO_POSITION
robot.leftdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);
robot.righttwo.setMode(DcMotor.RunMode.RUN_TO_POSITION);

// reset the timeout time and start motion.
runtime.reset();
robot.leftdrive.setPower((float)(speed) * leftdirection);
robot.rightdrive.setPower((float)(speed) * rightdirection);
robot.lefttwo.setPower((float)(speed) * leftdirection);
robot.righttwo.setPower((float)(speed) * rightdirection);
```

Signature : _____

Date: _____

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int newRighttwoTarget;
float rightdirection = 1;
float leftdirection = 1;

// 1 - turn right
// 2 - turn left
// minus goes forward... matches the controller
// Determine new target position, and pass to motor controller
if (direction == 2){
    leftdirection = -1;
} else {
    rightdirection = -1;
}

robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Determine new target position, and pass to motor controller

newLeftTarget = robot.leftdrive.getCurrentPosition() + (int)(Inches * robot.COUNTS_PER_INCH *
leftdirection);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// keep looping while we are still active, and there is time left, and both motors are running.  
while ((runtime.seconds() < timeoutS) ||  
    (robot.leftdrive.isBusy() && robot.rightdrive.isBusy() &&  
     robot.lefttwo.isBusy() && robot.righttwo.isBusy()) ) {  
    waitForTick (5);  
}  
  
// Stop all motion;  
robot.lefttwo.setPower(robot.STOP_SPEED);  
robot.righttwo.setPower(robot.STOP_SPEED);  
robot.leftdrive.setPower(robot.STOP_SPEED);  
robot.rightdrive.setPower(robot.STOP_SPEED);  
  
// Turn off RUN_TO_POSITION  
robot.lefttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);  
robot.righttwo.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);  
robot.leftdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);  
robot.rightdrive.setMode(DcMotor.RunMode.RUN_WITHOUT_ENCODER);  
}  
  
// turn right or left  
// 1 - left  
// 2 - right  
  
public void encoder2DriveTurn(double speed, double angle, int direction) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
double direction_value;

// Determine new target position, and pass to motor controller
if (direction == 1){
    direction_value = 1;
} else {
    direction_value = -1;
}

// Turn On RUN_Without encoder
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// robot.gyro.resetZAxisIntegrator();

while (!onHeading(speed, direction_value*angle, robot.P_TURN_COEFF)) {
    // Update telemetry & Allow time for other processes to run.

}

// Stop all motion;
robot.leftdrive.setPower(robot.STOP_SPEED);
robot.rightdrive.setPower(robot.STOP_SPEED);

}

/**
 * Perform one cycle of closed loop heading control.
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
*  
* @param speed  Desired speed of turn.  
* @param angle  Absolute Angle (in Degrees) relative to last gyro reset.  
*          0 = fwd. +ve is CCW from fwd. -ve is CW from forward.  
*          If a relative angle is required, add/subtract from current heading.  
* @param PCoeff  Proportional Gain coefficient  
* @return  
*/  
boolean onHeading(double speed, double angle, double PCoeff) {  
    double error ;  
    double steer ;  
    boolean onTarget = false ;  
    double leftSpeed;  
    double rightSpeed;  
  
    // determine turn power based on +/- error  
    error = getError(angle);  
  
    if (Math.abs(error) <= robot.HEADING_THRESHOLD) {  
        steer = 0.0;  
        leftSpeed = 0.0;  
        rightSpeed = 0.0;  
        onTarget = true;  
    }  
    else {  
        steer = getSteer(error, PCoeff);  
        rightSpeed = speed * steer;  
    }
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
leftSpeed = -rightSpeed;  
}  
  
// Send desired speeds to motors.  
robot.leftdrive.setPower(leftSpeed);  
robot.rightdrive.setPower(rightSpeed);  
  
  
return onTarget;  
}  
  
/**  
 * returns desired steering force. +/- 1 range. +ve = steer left  
 * @param error Error angle in robot relative degrees  
 * @param PCoeff Proportional Gain Coefficient  
 * @return  
 */  
public double getSteer(double error, double PCoeff) {  
    return Range.clip(error * PCoeff, -1, 1);  
}  
  
/**  
 * getError determines the error between the target angle and the robot's current heading  
 * @param targetAngle Desired angle (relative to global reference established at last Gyro Reset).  
 * @return error angle: Degrees in the range +/- 180. Centered on the robot's frame of reference  
 *      +ve error means the robot should turn LEFT (CCW) to reduce error.  
 */
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
/*
public double getError(double targetAngle) {

    double robotError;

    // calculate error in -179 to +180 range (
    // robotError = targetAngle - robot.gyro.getIntegratedZValue();
    // while (robotError > 180) robotError -= 360;
    // while (robotError <= -180) robotError += 360;
    // return robotError;
    return 0;
}
```

Following is the lift test function:

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.I2cAddr;
import com.qualcomm.robotcore.util.ElapsedTime;
import com.vuforia.HINT;
import com.vuforia.Vuforia;

import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.matrices.OpenGLMatrix;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
import org.firstinspires.ftc.robotcore.external.navigation.RelicRecoveryVuMark;
import org.firstinspires.ftc.robotcore.external.navigation.VuMarkInstanceId;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackable;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackableDefaultListener;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackables;
import org.firstinspires.ftc.teamcode.R;

import java.text.SimpleDateFormat;
import java.util.Date;

import static java.lang.Boolean.FALSE;
import static java.lang.Boolean.TRUE;

@Autonomous(name = "FF: TestLift", group = "Auto")
//@Disabled
public class TestLift extends LinearOpMode {

    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();

    LiftDef2 lift = new LiftDef2();
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
private ElapsedTime runtime = new ElapsedTime();

@Override public void runOpMode() throws InterruptedException {

    /*
     * Initialize the drive system variables.
     * The init() method of the hardware class does all the work here
     */
    robot.init(hardwareMap, 1);

    lift.init(hardwareMap, this);

    drive.init(hardwareMap, this);

    // Send telemetry message to signify robot waiting;
    telemetry.addData("Status", "Lift Test Position");

    String startDate;

    startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

    // wait for the start button to be pressed.
    waitForStart();

    telemetry.addData("9", "Lift Test %d", robot.position_option);
    telemetry.update();
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.wait(1);
// if the floor color is blue use the left arm
// if the floor color is red use the right arm

lift.liftmove( 1, 15);

robot.wait(1);

telemetry.addData("Path", "Test Lift Complete");
telemetry.update();
robot.wait(20);
idle(); // Always call idle()
}

public void waitForTick(long periodMs) {

long remaining = periodMs - (long)period.milliseconds();

// sleep for the remaining portion of the regular cycle period.
if (remaining > 0) {
try {
Thread.sleep(remaining);
} catch (InterruptedException e) {
Thread.currentThread().interrupt();
}
}
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
    }  
}  
  
// Reset the cycle clock for the next pass.  
period.reset();  
}  
}
```

The Final program used during the meet is our Teleop function and our program name is ChickenNoodleFrenchFry (we were hunger when we came up with the name):

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

The function of this program is to run teleop.

Neither the name of Qualcomm Technologies Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

*SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackables;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
import static com.qualcomm.robotcore.hardware.DcMotor.RunMode;
import static com.qualcomm.robotcore.util.Range.clip;
import static java.lang.Math.abs;
import static java.lang.String.format;
import static org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer.CameraDirection.BACK;

{@TeleOp(name = "FF: ChickenNoodleFrenchFry", group = "FF")
//@Disabled
public class ChickenNoodleFrenchFry extends LinearOpMode {
    /* Declare OpMode members. */

    private ElapsedTime runtime = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle();
    private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNywpbJlmrEGBajczWU1Oi-
Num6rS90mBDJwrv1CJM5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/
MZh7Zvp35rYD6lavfkgCMZ0DVAXHBv4J5LlrGIVXYfhhS1NklTGPDqVRW2aBmKLwctHZaztzycau3g// 
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnPfKgC-E-
Zuz1DStPn6ZriIRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMjjIrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here
    /**
     * {@link #vuforia} is the variable we will use to store our instance of the Vuforia
     * localization engine.
     */
    private VuforiaLocalizer vuforia;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFObjectDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFOBJECTDETECTOR.Parameters tfodParameters = new TFObjectDetector.Parameters();  
private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";  
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";  
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";  
private boolean targetVisible = false;  
// private boolean targetVisible = false; Valid choices are: BACK or FRONT  
  
private static final VuforiaLocalizer.CameraDirection CAMERA_CHOICE = BACK;  
// Variables to be used for later  
private VuforiaLocalizer vuforiaLocalizer;  
private VuforiaLocalizer.Parameters parameters;  
private VuforiaTrackables visionTargets;  
@Override  
public void runOpMode() throws InterruptedException {  
  
    // data for moving the phone  
    boolean phone_locked;  
    // set to be at the bottom  
    int extender_position1 = 0;  
    float extender_power1 = 0;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int last_extender_position = -999;
int extender_position = 2;
int extender_direction = 0;
int attop = 0;
int atbottom = 1;
int location = 0;
int phonecyclecount;
phone_locked = false;
phonecyclecount = 0;

// set phone data
// State used for updating telemetry
// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Setting Up ChickenNoodleFrenchFry"); //
telemetry.update();
/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */

/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

// end of the front-end of the robots definitions

telemetry.addData("Text", "Waiting to start ChickenNoodleFrenchFry");
telemetry.update();
// this is to initialize the camera for target viewing
// initVufor(); telemetry.update();
/*
if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3","setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3","back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}
*/
waitForStart();
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// put servos in start position

robot.init(hardwareMap, 2);
robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
//move phone out of the way
robot.phone_position = .84;
robot.phone.setPosition(robot.phone_position);

robot.lock_power = .5;
robot.lock.setPosition(robot.lock_power);
// Set up our telemetry dashboard
composeTelemetry();
// setting up timer
runtime.reset();
// while the op mode is active, loop and read the RGB data.
// Note we use opModelsActive() as our loop condition because it is an interruptible method.
while (opModelsActive()) {

/*
 * Gamepad 1 controls the motors via the left/right stick
 */

// this is for the motor control function
// forward is negative power value
// backwards is positive power value
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
int drive_mode = 2;
if (gamepad1.right_stick_y != 0 || gamepad1.left_stick_y != 0 ||
    gamepad1.right_stick_x != 0 || gamepad1.left_stick_x != 0) {
    if (gamepad1.right_stick_y <= 0) {
        robot.right = ((-gamepad1.right_stick_y - (abs(gamepad1.right_stick_x))) / (float) .5);
    } else {
        robot.right = ((-gamepad1.right_stick_y + abs(gamepad1.right_stick_x)) / (float) .5);
    }
    if (gamepad1.left_stick_y <= 0) {
        robot.left = ((-gamepad1.left_stick_y - (abs(gamepad1.left_stick_x))) / (float) .5);
    } else {
        robot.left = ((-gamepad1.left_stick_y + abs(gamepad1.left_stick_x)) / (float) .5);
    }

    // clip the right/left values so that the values never exceed +/- 1
    robot.right = clip(robot.right, -1, 1);
    robot.left = clip(robot.left, -1, 1);

    // scale the joystick value to make it easier to control
    // the robot more precisely at slower speeds.
    robot.right = (float) scaleInput(robot.right);
    robot.left = (float) scaleInput(robot.left);

    // write the values to the motors
    setDrivePower(robot.right, robot.left, drive_mode);
} else {
    robot.right = 0;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.left = 0;
// clip the right/left values so that the values never exceed +/- 1
robot.right = clip(robot.right, -1, 1);
robot.left = clip(robot.left, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
robot.right = (float) scaleInput(robot.right);
robot.left = (float) scaleInput(robot.left);

// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
}

// Gamepad 2

// Lift (Raise and Lower) the end effector
// You should not go much past 90degrees
// up is negative value on the controller
// down is positive value on the controller
// lower_stop is to make sure the motor stop when touching the floor
// lift_top is to stop when just past being vertical
// if the digital channel returns true it's HIGH and the button is unpressed.
if ((gamepad2.left_stick_y < 0 && robot.upper_stop.getState() == true) ||
    (gamepad2.left_stick_y > 0 && robot.lower_stop.getState() == true)) {
    // set lift speed to a constant value use the negative
    // negate the power value to get the motors going in the direction desired
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
if (gamepad2.left_stick_y < -.75) gamepad2.left_stick_y = (float) -.75;
if (gamepad2.left_stick_y > .75) gamepad2.left_stick_y = (float) .75;

robot.lift_power = -gamepad2.left_stick_y;

// clip the right/left values so that the values never exceed +/- 1
robot.lift_power = clip(robot.lift_power, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.

robot.lift_power = (float) scaleInput(robot.lift_power);

robot.right_liftdrive.setPower(robot.lift_power);
robot.left_liftdrive.setPower(robot.lift_power);

} else {

// stop the lift motor

robot.lift_power = 0;
robot.lift_power = (float) scaleInput(robot.lift_power);
robot.right_liftdrive.setPower(robot.lift_power);
robot.left_liftdrive.setPower(robot.lift_power);
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// raise the mineral grabber up/down
// up is negative power value
// going extender position = 2 we are at the bottom
extender_position1 = robot.extenderdrive.getCurrentPosition();
if (extender_position1 > last_extender_position) extender_direction = 2;
if (extender_position1 < last_extender_position) extender_direction = 1;

if ((gamepad2.right_stick_y < 0 )
|| (gamepad2.right_stick_y > 0)) {

last_extender_position = extender_position1;

// speed to a constant value

robot.extender_power = gamepad2.right_stick_y;
// clip the right/left values so that the values never exceed +/- 1
extender_power1 = clip(robot.extender_power, -1, 1);
robot.extender_power = extender_power1;
// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
if (!robot.extender_lower.getState() && extender_direction == 1 && atbottom == 0) {
    attop = 1;
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
    location = robot.extenderdrive.getCurrentPosition();
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
if (!robot.extender_lower.getState() && extender_direction == 2 && attop == 0){
    atbottom = 1;
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
    location = robot.extenderdrive.getCurrentPosition();
}

if (robot.extender_lower.getState()){
    atbottom = 0;
    attop = 0;
}

if (!robot.extender_lower.getState() && extender_direction == 1 && robot.extender_power < 0 /* && attop == 1 */)
    robot.extender_power = 0;
if (!robot.extender_lower.getState() && extender_direction == 2 && robot.extender_power > 0 /* && atbottom == 1 */)
    robot.extender_power = 0;
    // over ride because you are at the top or bottom and you need to move
    if (!robot.extender_lower.getState() && atbottom == 1 && extender_power1 < 0)
        robot.extender_power = extender_power1;
    if (!robot.extender_lower.getState() && attop == 1 && extender_power1 > 0)
        robot.extender_power = extender_power1;

    robot.extender_power = (float) scaleInput(robot.extender_power);
    robot.extenderdrive.setPower(robot.extender_power);

} else {
    // stop the extension motor
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.extender_power = 0;
robot.extender_power = (float) scaleInput(robot.extender_power);
robot.extenderdrive.setPower(robot.extender_power);

}

// wrench up/down
if (gamepad2.right_bumper && robot.wrench_upper.getState() == true ) {
    //going up
    //make sure you are unlocked
    //unlocking
    robot.lock_power = .5;
    robot.lock.setPosition(robot.lock_power);

    robot.wrench_power = .75;
    robot.left_wrench.setPower(robot.wrench_power);
    robot.right_wrench.setPower(-robot.wrench_power);
} else if (gamepad2.right_trigger != 0 && robot.wrench_lower.getState() == true) {
    //going down
    robot.wrench_power = .75;
    robot.left_wrench.setPower(-robot.wrench_power);
    robot.right_wrench.setPower(robot.wrench_power);
} else {
    //stop wrench
    robot.wrench_power = 0;
    robot.left_wrench.setPower(robot.wrench_power);
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.right_wrench.setPower(robot.wrench_power);
}

// set the lock so we can lift the robot
if (gamepad2.left_bumper) {
    //locking
    robot.lock_power = .16;
    robot.lock.setPosition(robot.lock_power);
} else if (gamepad2.left_trigger != 0) {
    //unlocking
    robot.lock_power = .5;
    robot.lock.setPosition(robot.lock_power);
}

// pick the minerals up
if (gamepad2.b) {
    robot.collector_power = 1;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
if (gamepad2.x) {
    robot.collector_power = 0;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
// spit the minerals out
if (gamepad2.y) {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
robot.collector_power = -1;
robot.collector_power = (float) scaleInput(robot.collector_power);
robot.collectordrive.setPower(robot.collector_power);

}
if (gamepad2.a) {
    robot.collector_power = 0;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}

// right bumper move out the phone
// left bumper move in the phone

if (gamepad1.right_bumper && robot.phone.getPosition() >= 0 && !phone_locked) {
    //move phone (bottom is 1)
    robot.phone_position += .1;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;

} else if (gamepad1.left_bumper && robot.phone.getPosition() <= 1 && !phone_locked) {
    // phone (top position is 0)
    robot.phone_position -= .1;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;
}
if (phone_locked) {
    phonecyclecount ++;
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
if (phonecyclecount == 100){  
    phone_locked = false;  
    phonecyclecount = 0;  
}  
  
telemetry.addData("0", "FF ChickenNoodleFrenchFry - %2.5f S Elapsed", runtime.seconds());  
// get the value of the light  
  
telemetry.addData("D1", format("Rx: %.2f Lx: %.2f Ry: %.2f Ly: %.2f",  
gamepad1.right_stick_x, gamepad1.left_stick_x, gamepad1.right_stick_y, gamepad1.left_stick_y));  
    telemetry.addData("D2", format("Rx: %.2f Lx: %.2f ", gamepad2.right_stick_x,  
gamepad2.left_stick_x));  
    telemetry.addData("1", format("R - %.2f L - %.2f - Lift - %.2f - Extender - %.2f", robot.right,  
robot.left, robot.lift_power, robot.extender_power));  
  
// telemetry.addData("3", format("Floor Color %2d Blue %2d Red %2d", robot.floor_color.alpha()  
, robot.floor_color.blue(), robot.floor_color.red()));  
    telemetry.addData("4", format("Lower %s upper %s", robot.lower_stop.getState(), ro-  
bot.upper_stop.getState()));  
    telemetry.addData("Extender state is ", robot.extender_lower.getState());  
    telemetry.addData("5", format("phone position %.2f - %d", robot.phone.getPosition()  
(), robot.extenderdrive.getCurrentPosition()));  
    telemetry.addData("6", format("extnder direction %d", extender_direction));  
    telemetry.addData("9", format("atbottom %d atop %d", atbottom, attop));  
    telemetry.addData("9", format("location %d ", location));
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
telemetry.update();
idle(); // Always call idle() at the bottom of your while(opModelsActive()) loop
}

/*
 * This method scales the joystick input so for low joystick values, the
 * scaled value is less than linear. This is to make it easier to drive
 * the robot more precisely at slower speeds.
 */
double scaleInput(double dVal) {
    double[] scaleArray = {0.0, 0.05, 0.09, 0.10, 0.12, 0.15, 0.18, 0.24,
        0.30, 0.36, 0.43, 0.50, 0.60, 0.72, 0.85, 1.00, 1.00};
;

    // get the corresponding index for the scaleInput array.
    int index = (int) (dVal * 16.0);
    if (index < 0) {
        index = -index;
    } else if (index > 16) {
        index = 16;
    }

    double dScale = 0.0;
    if (dVal < 0) {
        dScale = -scaleArray[index];
    } else {
        dScale = scaleArray[index];
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
}

return dScale;

}

public void setDrivePower(float rightPower, float leftPower, int power_mode) {
    // telemetry.addData("2", format("PM - %d", power_mode));
    // telemetry.update();
    if (power_mode == 2) {
        // set front and back motors full power
        robot.rightdrive.setPower(rightPower);
        robot.leftdrive.setPower(leftPower);
        robot.righttwo.setPower(rightPower);
        robot.lefttwo.setPower(leftPower);
    } else if (power_mode == 1) {
        // set front motors
        robot.rightdrive.setPower(0);
        robot.leftdrive.setPower(0);
        robot.righttwo.setPower(0);
        robot.lefttwo.setPower(0);
    } else {
        // set front motors
        robot.rightdrive.setPower(0);
        robot.leftdrive.setPower(0);
        robot.righttwo.setPower(0);
        robot.lefttwo.setPower(0);
    }
}
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

}

```
public void setDriverMode(RunMode mode) {
```

```
    if (robot.leftdrive.getMode() != mode) {
```

```
        robot.leftdrive.setMode(mode);
```

```
    }
```

```
    if (robot.rightdrive.getMode() != mode) {
```

```
        robot.rightdrive.setMode(mode);
```

```
    }
```

```
}
```

```
//-----
```

```
// Telemetry Configuration
```

```
//-----
```

```
void composeTelemetry() {
```

```
    telemetry.addData("ball color red","ball red color ");
```

```
    telemetry.update();
```

```
    // At the beginning of each telemetry update, grab a bunch of data
```

```
    // from the IMU that we will then display in separate lines.
```

```
    telemetry.addAction(new Runnable() { @Override public void run()
```

```
    {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
// Acquiring the angles is relatively expensive; we don't want
// to do that in each of the three items that need that info, as that's
// three times the necessary expense.

    robot.angles = robot imu.getAngularOrientation(AxesReference.INTRINSIC, AxesOrder.ZYX, AngleUnit.DEGREES);
}

});

telemetry.addData("status", new Func<String>() {
    @Override public String value() {
        return robot imu.getSystemStatus().toShortString();
    }
})
.addData("calib", new Func<String>() {
    @Override public String value() {
        return robot imu.getCalibrationStatus().toString();
    }
});

telemetry.addData("heading", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot angles.angleUnit, robot angles.firstAngle);
    }
})
.addData("roll", new Func<String>() {
```

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
@Override public String value() {
    return formatAngle(robot.angles.angleUnit, robot.angles.secondAngle);
}
})
.addData("pitch", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot.angles.angleUnit, robot.angles.thirdAngle);
    }
};

//-----
// Formatting
//-----
```

```
String formatAngle(AngleUnit angleUnit, double angle) {
    return formatDegrees(AngleUnit.DEGREES.fromUnit(angleUnit, angle));
}

String formatDegrees(double degrees){
    return String.format(Locale.getDefault(), "%.1f", AngleUnit.DEGREES.normalize(degrees));
}

private void initVuforia() {
    /*
     * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
     */
}
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

```
VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();  
  
parameters.vuforiaLicenseKey = VUFORIA_KEY;  
parameters.cameraDirection = BACK;  
  
// Instantiate the Vuforia engine  
vuforia = ClassFactory.getInstance().createVuforia(parameters);  
  
}  
/**  
 * Initialize the Tensor Flow Object Detection engine.  
 */  
private void initTfod() {  
  
    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(  
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());  
  
    TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters  
(tfodMonitorViewId);  
    tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);  
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LA-  
BEL_SILVER_MINERAL);  
  
}
```

signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

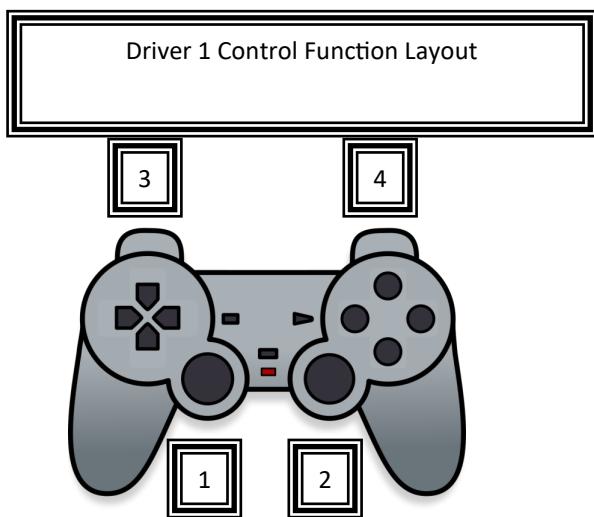
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

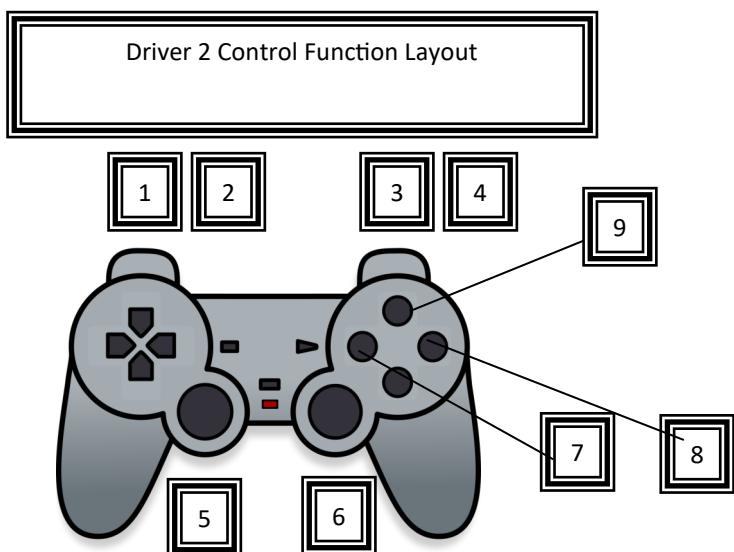
Process:

Described below is the functional layout for the robot controller buttons.



Driver 1 Control Function Definition

1. Left Wheels control
2. Right Wheels control
3. Move the phone left
4. Move the phone right



Driver 2 Control Function Definition

1. Lock the second layer of the slide
2. Unlock the second layer of the slide
3. Raise the second layer of the slide
4. Lower the second layer of the slide
5. Lift and lower grabber
6. Extend and retract grabber
7. Make Grabber go backwards
8. Make Grabber pick up items
9. Stop Grabber

Signature : Shelby Greer

Date: Dec. 15, 2018

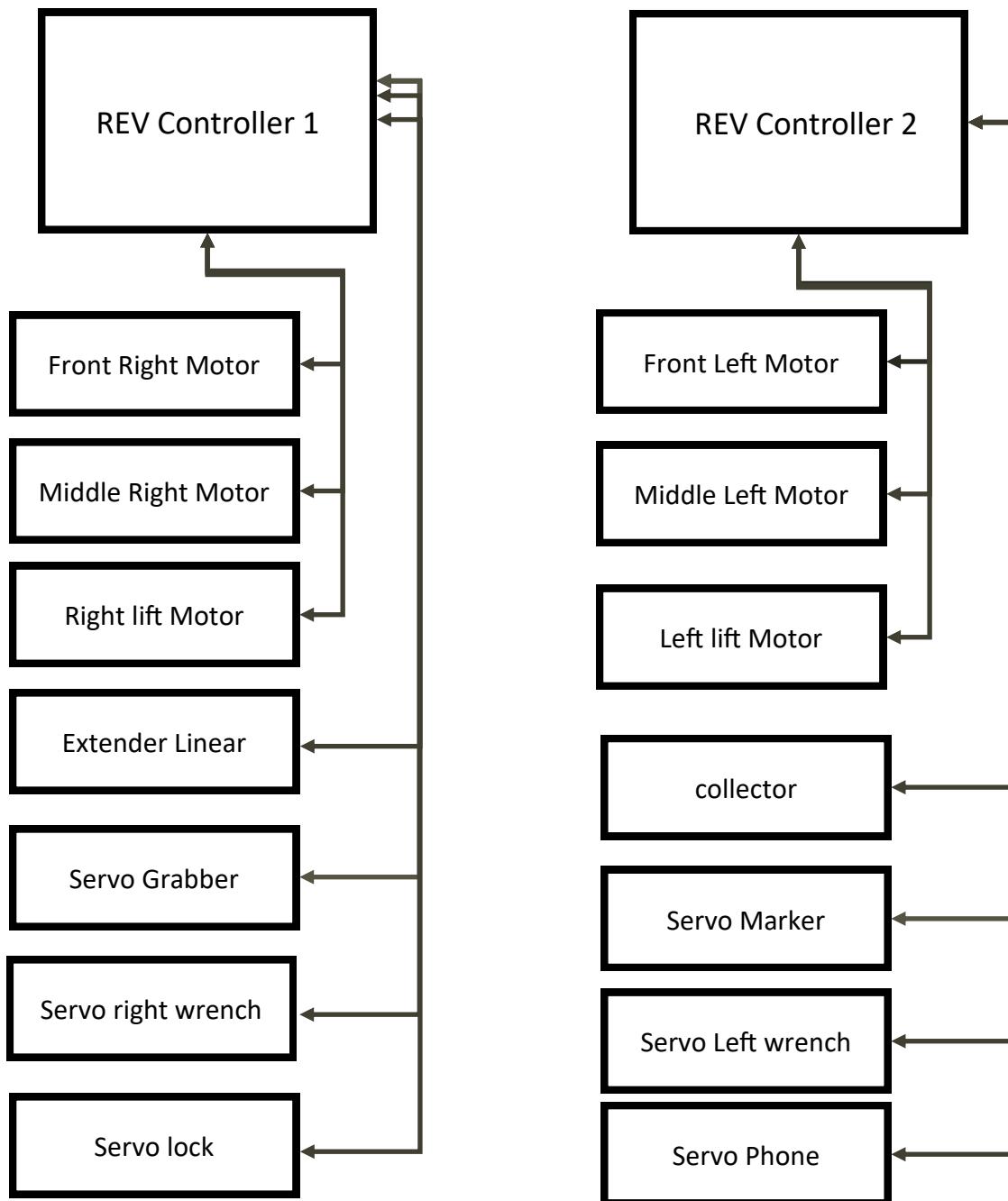
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:



Signature : Shelby Greer

Date: Dec. 15, 2018

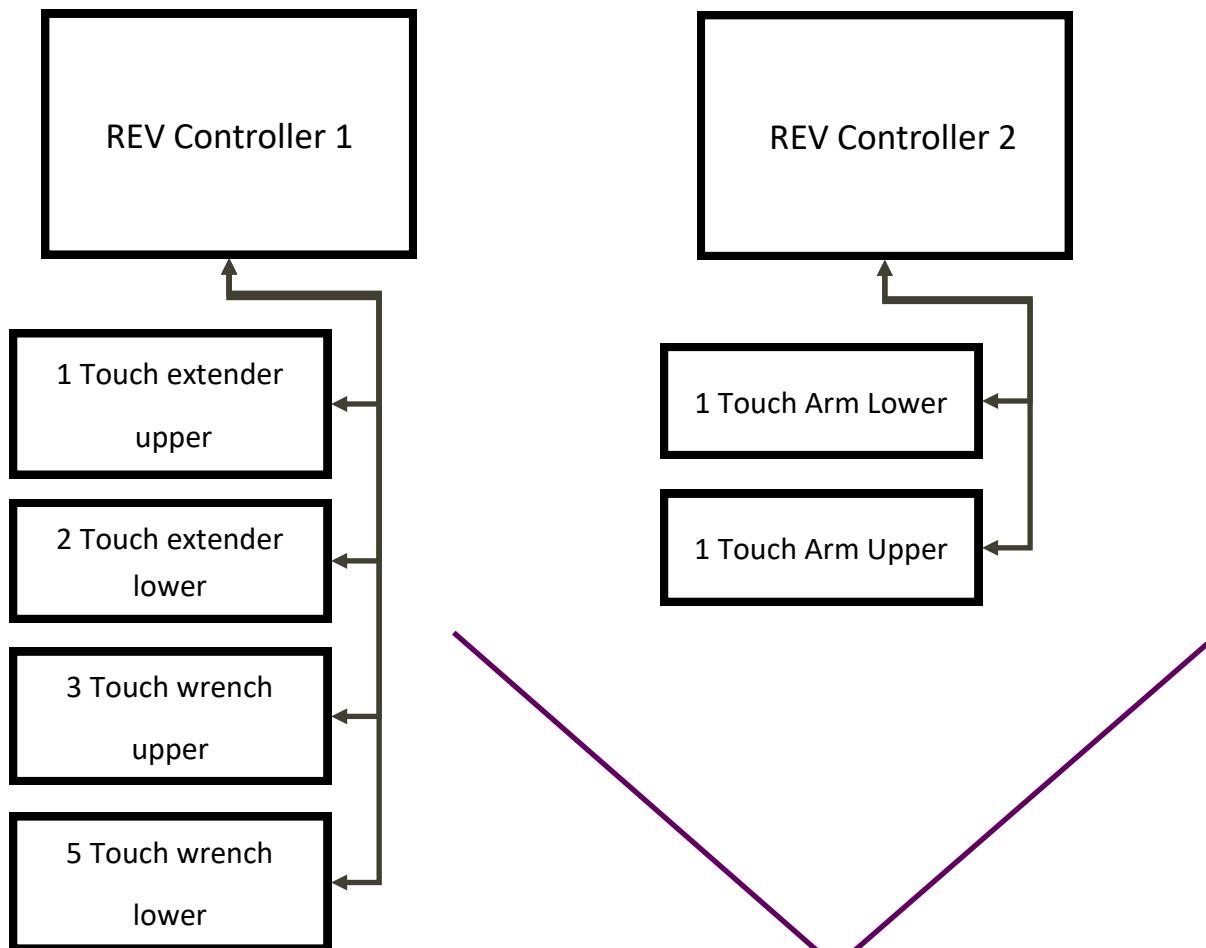
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:



Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

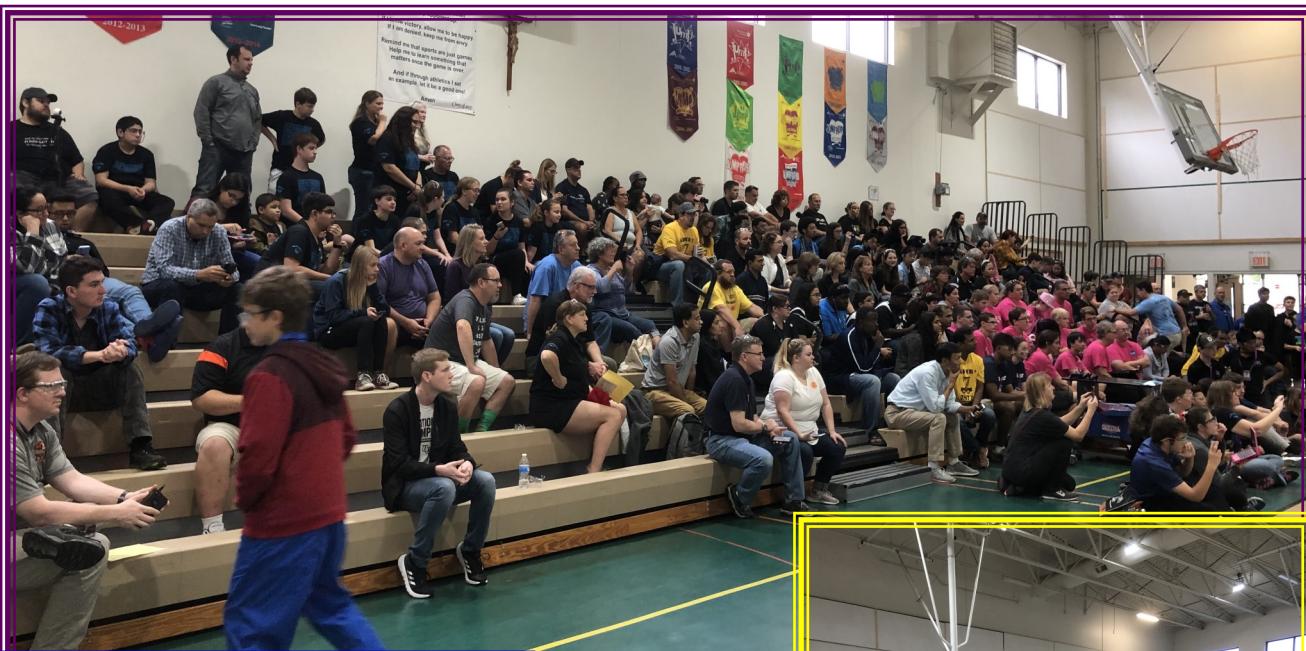
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

We also change the gym to support our game and the house was packed most all day! We thank all our



volunteers for
making this
event a great
success!!

X



Signature : _____

Date: _____

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

The meet went well as we were able to run our autonomous program where it worked 4 out of 5 times. After our second match we had to quickly remove most of the setup questions as it took too long to answer. We were able to change it to have only one question which was to find out if we were on the deposit or crater side of the lander.

Our team was able to join the volunteers for lunch.



Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

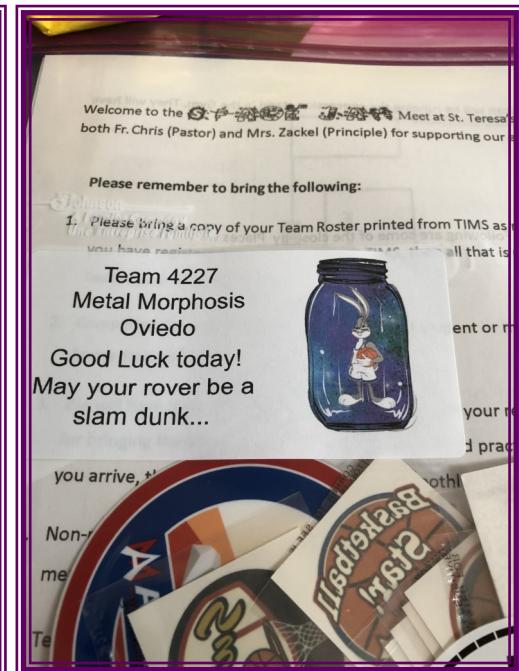
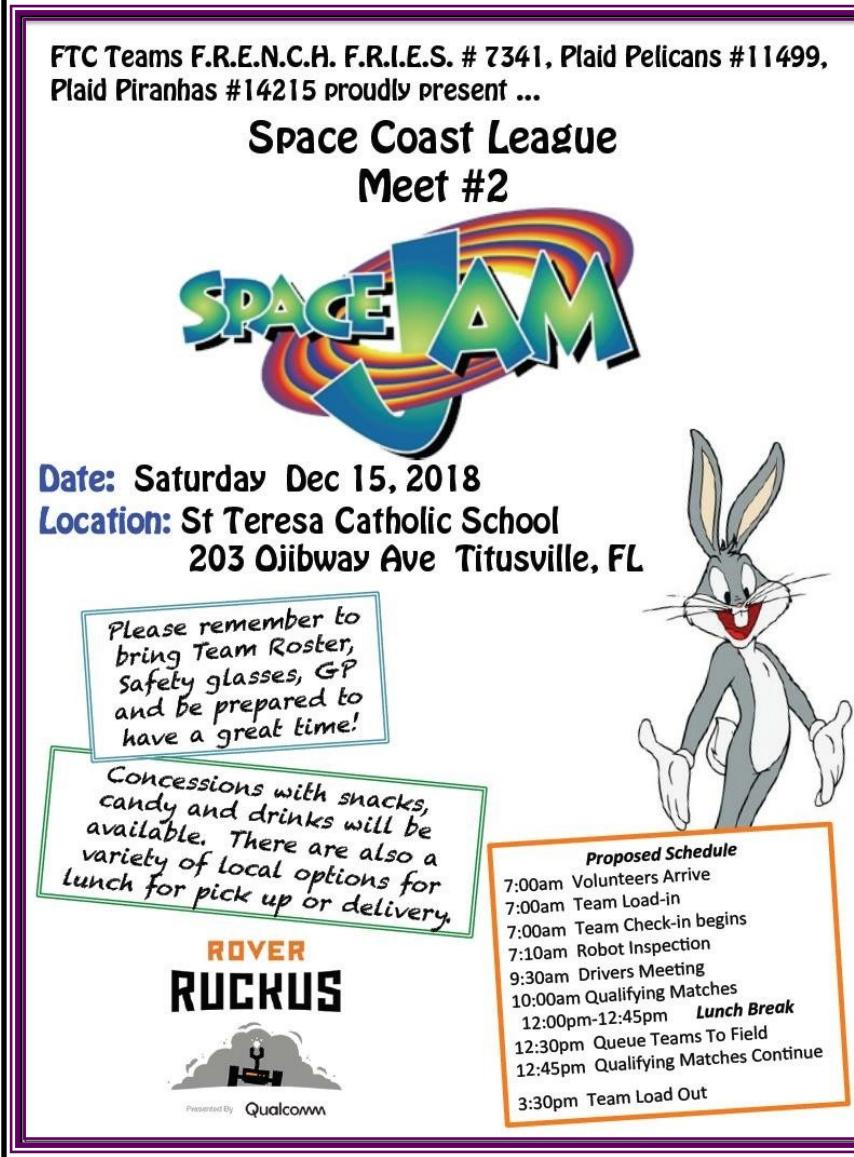
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

We would like to thank **Plaid Pelican** and **Plaid Piranha's** for helping with our Meet... Space Jam. We were happy to have a great items to share with all the other teams in the league. Pictured is the flyer



and team info bag, attached is the badges used during the meet.

Signature: Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

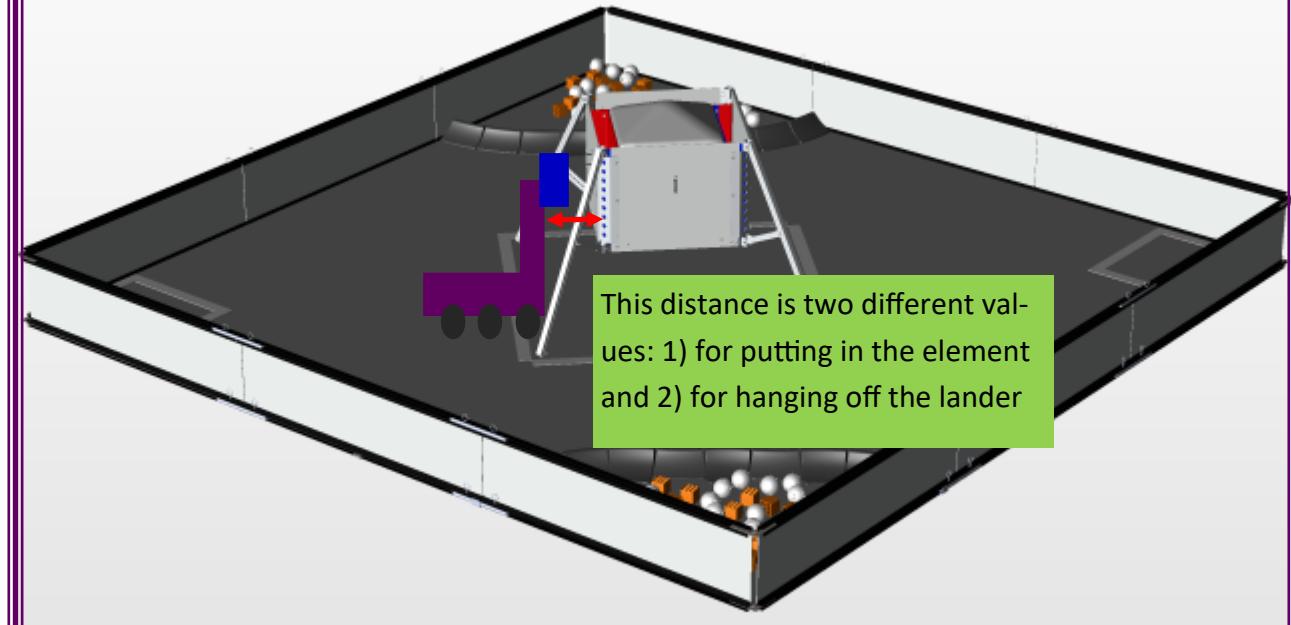
Engineering Activity Continued

Date: December 14-15, 2018

Process:

After we finished cleanup up after the meet we went to dinner at Portofino's and discussed some improvement to the robot as well as completed an outreach event. We were every happy to have Chelsea and Emily join us for the meet.

We discussed changing the grabber to something lighter and that you could see around to make it easier to hang at the end of the meet. We did try a few times to hang, but found it hard to know that we were lined-up. Also during testing at the meet we found that the robot was going too close to the lander making it impossible to latch the robot. It was determined that we could add a three position arm that will stop the robot so it does not get too close during the placement of the mineral in the lander or to hang on the lander. The third position for the arm would be stowed so it is not sticking out the back of the robot.



signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

Following is the scores from the matches of this meet:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
1	12090 14989	14215 14976	142	47	65	77	0	15	32	0
2	7477 12245	15401 11499	7	25	0	7	0	0	25	0
3	14673 8392	4717 9013	15	65	0	15	0	30	35	0
4	7341 6323	9671 7592	170	152	65	105	0	70	82	0
5	14765 14534	8945 4227	121	30	50	71	0	30	0	0
6	14215 14856	11499 6323	108	88	55	53	0	30	58	0
7	9671 14989	7477 14673	19	76	0	19	0	45	31	0
8	4717 7592	12245 8945	155	89	70	85	0	0	89	0
9	4227 15401	14976 8392	84	55	65	19	0	0	55	0
10	12090 14765	14856 9013	152	144	65	87	0	60	84	0
11	14673 7341	14534 15401	55	136	40	15	0	55	81	0
12	12245 7592	14765 14215	206	65	80	126	0	25	40	0
13	12090 8945	11499 7341	167	61	55	112	0	40	21	0
14	9013 4227	6323 14989	200	194	85	115	0	65	129	0
15	14976 4717	14856 7477	97	80	30	67	0	55	25	0

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

Process:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
16	8392 9671	14534 12090	25	178	25	0	0	100	78	0
17	7592 14673	14856 4227	177	115	65	112	0	45	70	0
18	9013 14976	7341 12245	110	59	30	80	0	40	19	0
19	6323 7477	14765 8392	212	34	40	172	0	10	24	0
20	14215 15401	8945 9671	111	118	75	36	0	60	58	0
21	14989 11499	4717 14534	46	156	0	46	0	100	56	0
22	6323 12245	12090 14673	178	131	40	138	0	45	86	0
23	14534 7477	9013 7592	216	219	95	81	40	75	144	0
24	14976 9671	11499 14765	34	83	0	34	0	0	83	0
25	8392 14856	14989 8945	81	81	10	71	0	0	81	0
26	4227 14215	7341 4717	150	147	80	70	0	65	82	0
27	15401 14765	7592 12090	95	252	35	60	0	115	137	0

Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 14-15, 2018

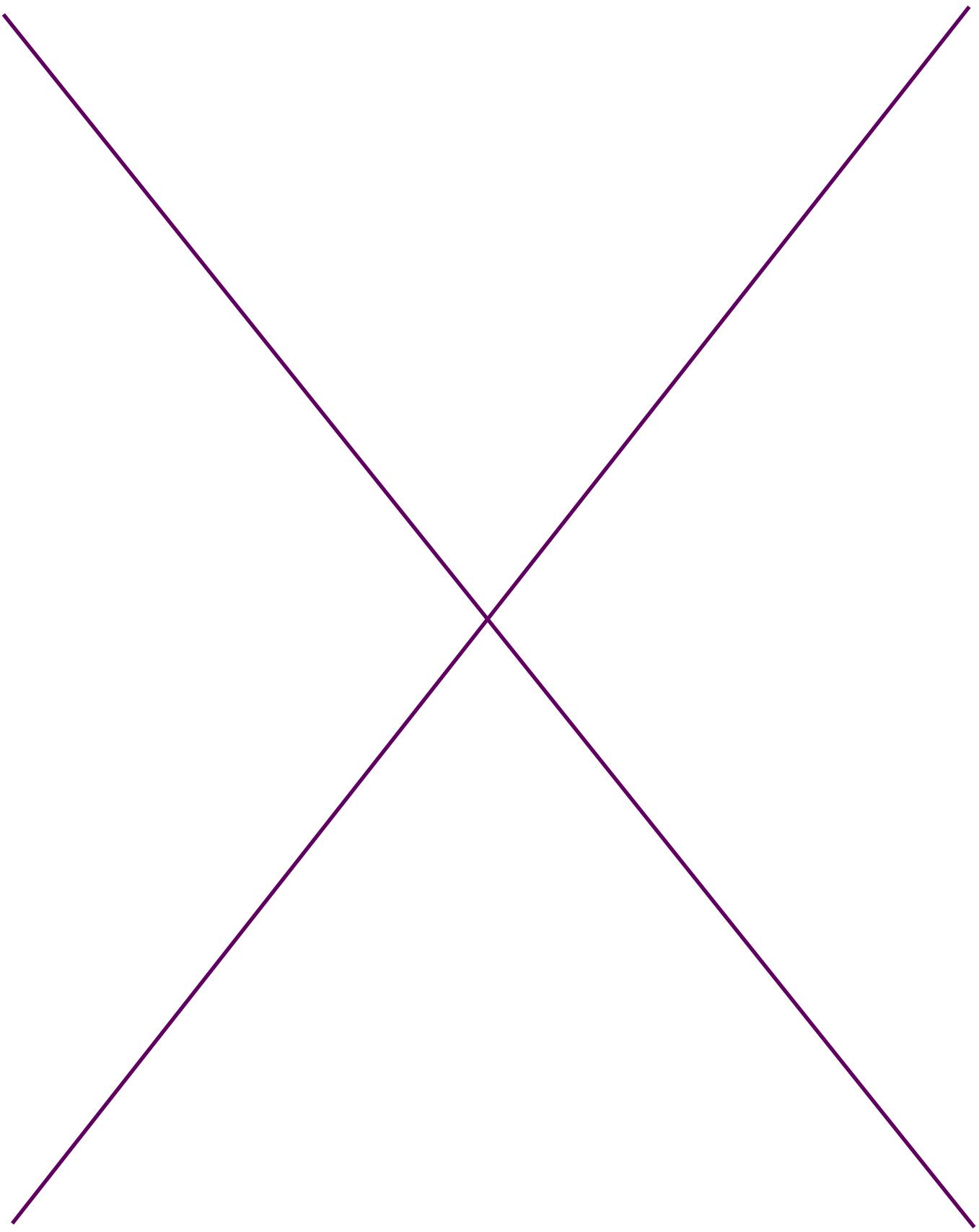
Process:

Following is the League ranking after the first 10 matches. We hope to due much better in the next meet.

Team Number	Team Name	QP	RP	Ranked Matches	Matches
7592	Roarbots	18	927	10	11
4227	Metal Morphosis	18	865	10	11
6323	The Pink Team	14	948	10	11
12090	STEM-Punk	14	825	10	11
9013	Merritt Island Matrix	14	779	10	11
7477	Super 7	14	706	10	11
14534	Robo Peggz	14	626	10	11
15401	CyberNoles	12	587	10	11
4717	Mechromancers	12	580	10	11
14856	Wandering Wildcats	11	947	10	11
8945	Touch-Down!	11	677	10	11
14989	Florida Prep Academy	9	688	10	11
14215	Plaid Piranhas	8	760	10	11
14976	PAC RATS	8	566	10	11
11499	Plaid Pelicans	8	545	10	11
7341	F.R.E.N.C.H. F.R.I.E.S.	6	806	10	11
14765	Mechatronic Mustangs	6	337	5	5
12245	Cobalt	4	351	5	5
14673	Odyssey One	4	335	5	5
14538	VCS Panthers Robotics - Crimson Blaze	4	326	6	6
9671	Tomahawk Technauts	2	341	5	5
8392	Lift-Off!	1	440	10	11

Signature : Shelby Greer

Date: Dec. 15, 2018



Signature : Shelby Greer

Date: Dec. 15, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: December 22, 2018

Purpose of the Activity:

Create and test a new grabber. We are looking to duplicate one we did a few years ago

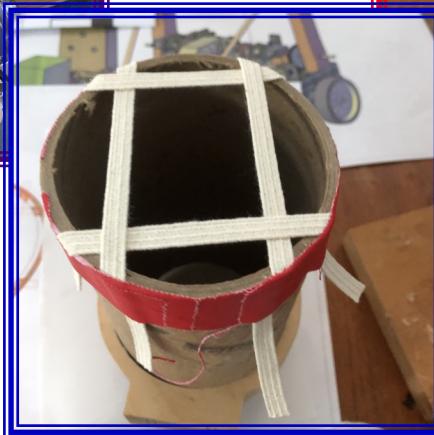
Create servo boxes for the robot distance guide

Process:

Our mentor Mr. Louis cut the round shipping tube to a length of xx inches which is long enough to hold two items of any combination. We experiment with various elastics on the bottom of the robot to determine which one we would like to try. We ended up using the four pieces of elastic.



tic style allowed either the gold or silver we found that when capturing the gold the gold mineral to make sure that the ele- tube. There has been a discussion whether grabber on a hinge so we could possible match. We would then need to be able to complete the search for the element in a shorter time so we



First we had to remove the initial grabber so we can add the



new grabber. Either elastic or mineral. Hand capturing mineral you needed to tap ment goes fully into the or not to have the new hang at the start of the

signature : Shelby Greer

Date: Dec. 22, 2018

Team 7341

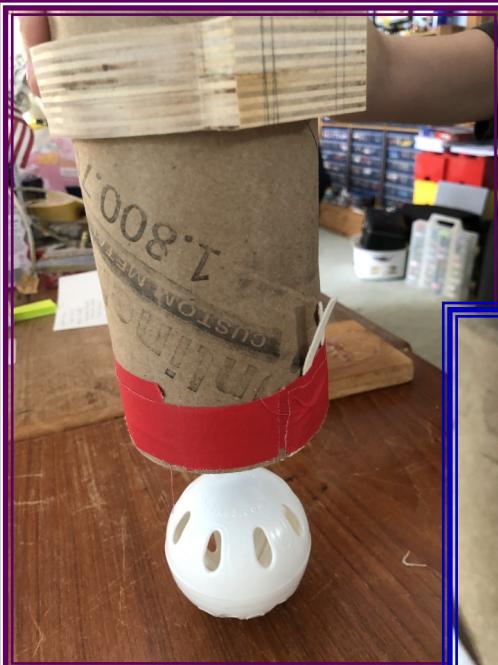
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 22, 2018

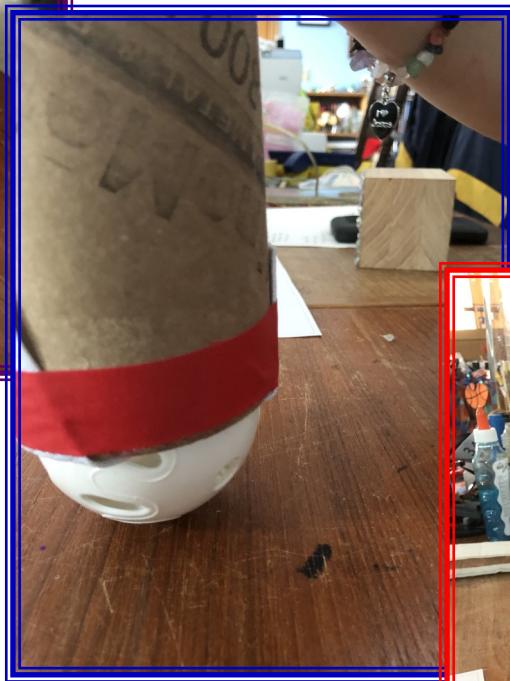
Process:

could lower the robot and find our target of the gold mineral.



Following is a capture sequence:

1. First line up the with the mineral
2. Lower the grabber over the mineral
3. Pick up the grabber
4. Go to the depot or lander
5. Dump the mineral
6. Repeat the sequence



Signature : Shelby Greer

Date: Dec. 22, 2018

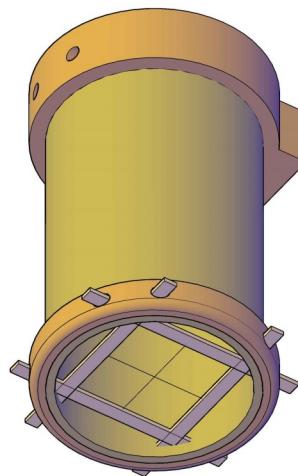
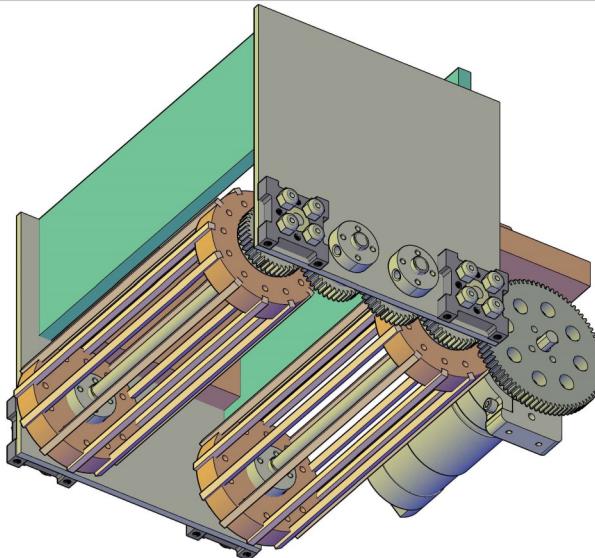
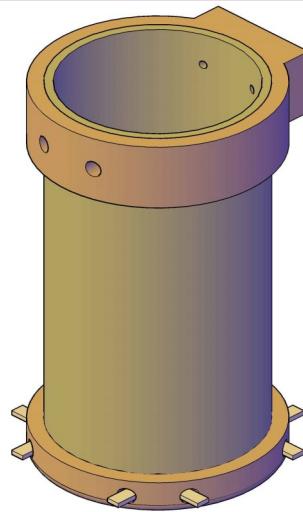
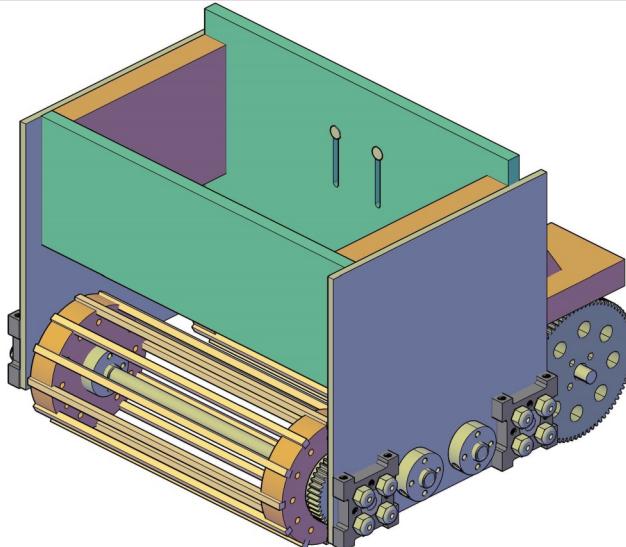
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 22, 2018

Process:



Comparison of the two end effectors for picking up the minerals.

Signature : Shelby Greer

Date: Dec. 22, 2018

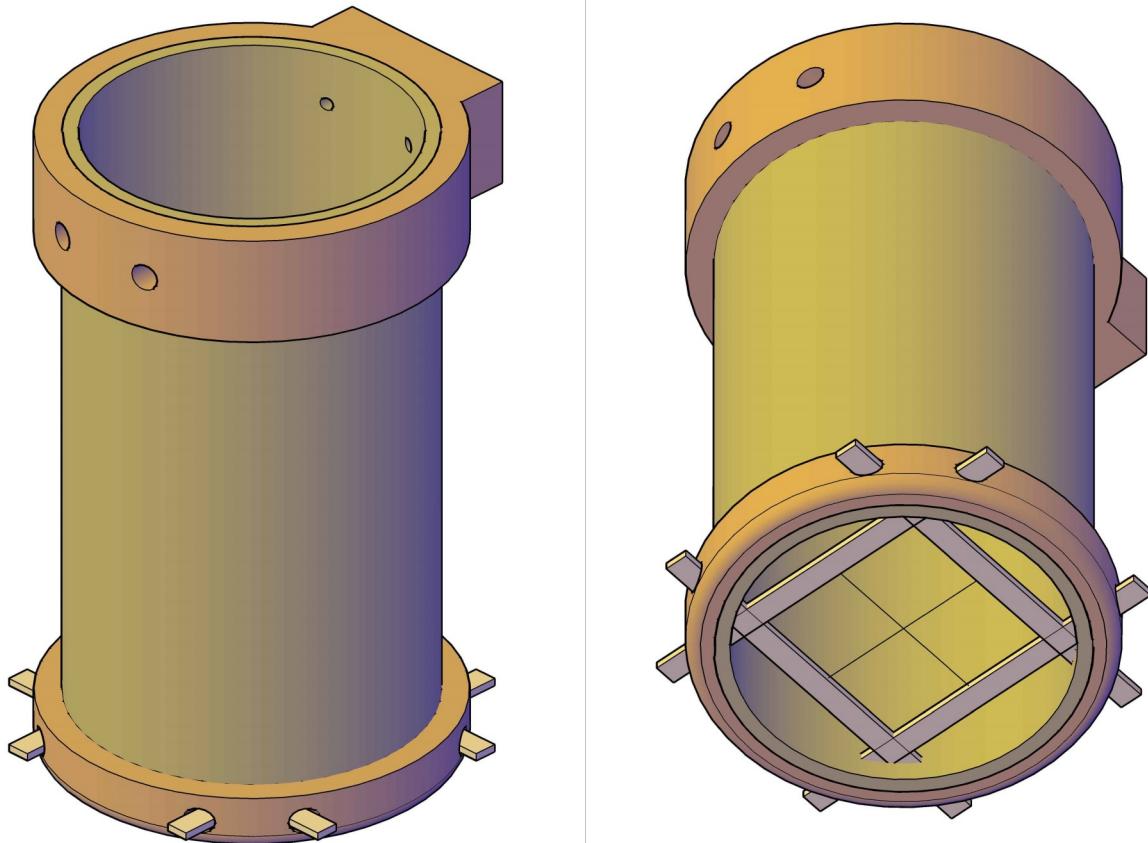
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: December 22, 2018

Process:



Sometimes the simplest design is a better item.

Signature : Shelby Greer

Date: Dec. 22, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

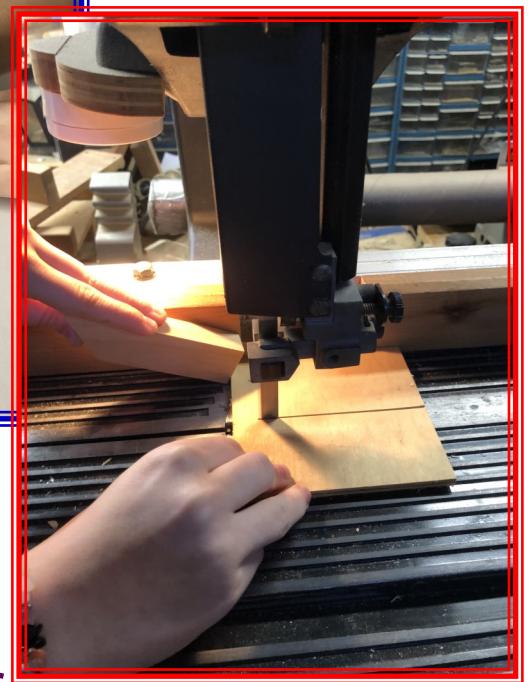
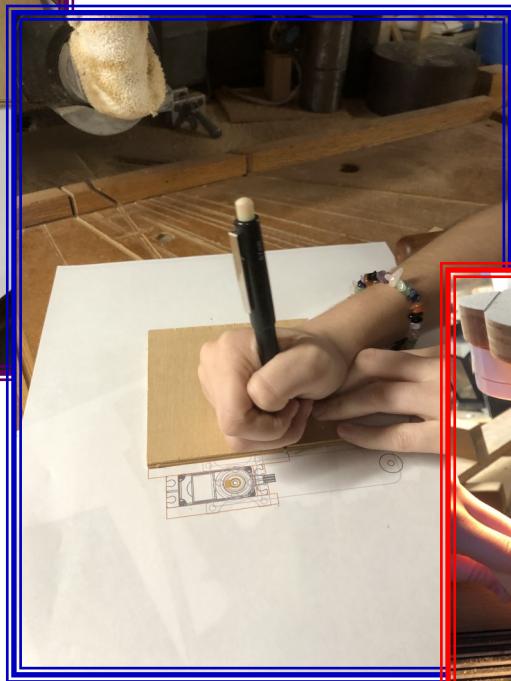
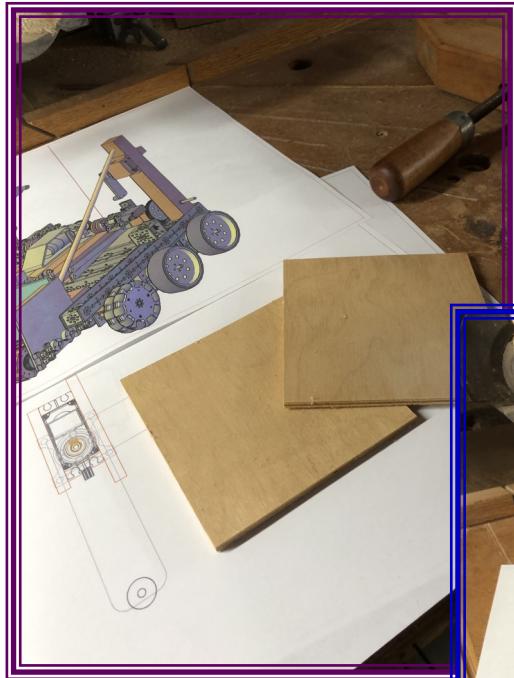
Engineering Activity Continued

Date: December 22, 2018

Process:

After testing the new grabber we went and started building the new servo boxes. We needed the box to be low profile, but it need to support the distance arm for the two different level.

1. For dumping the minerals in the lander
2. For hanging from the lander.



Signature : Shelby Greer

Date: Dec. 22, 2018

Team 7341

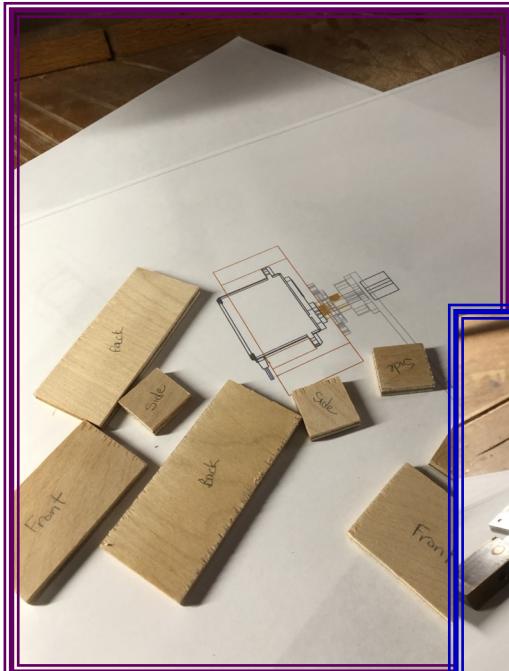
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

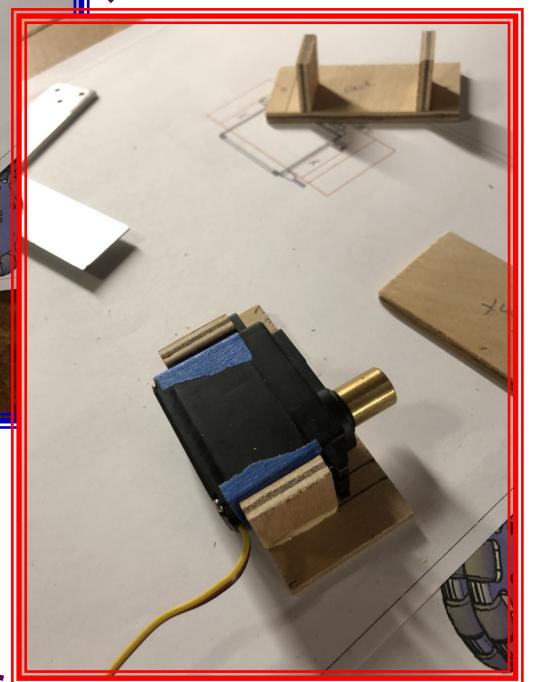
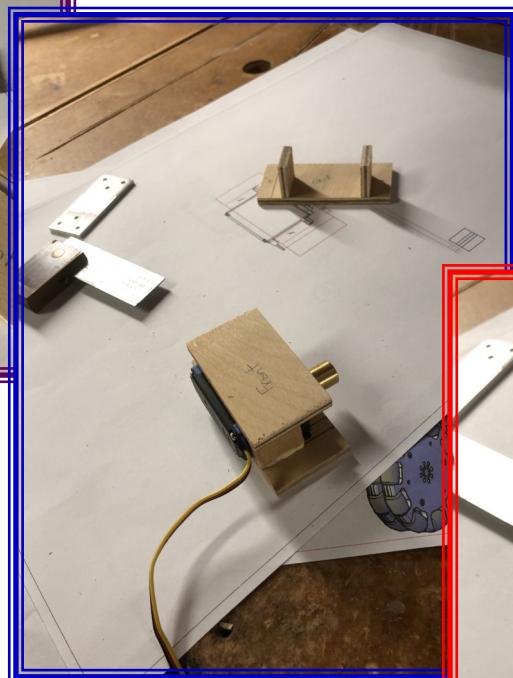
Date: December 22, 2018

Process:

1. We will need to test the structure to determine if the first one we have created will work correctly.



2. We measured and cut each piece separately
3. The building process is that you glue the one side then using screws for the other side so you can remove the servo if needed.



Signature: Shelby Greer

Date: Dec. 22, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

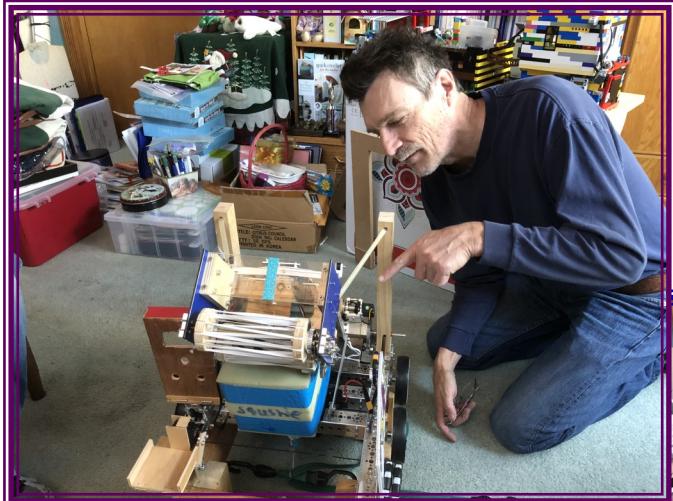
Date: December 23, 2018

Purpose of the Activity:

A special engneeting stopped by our build space.

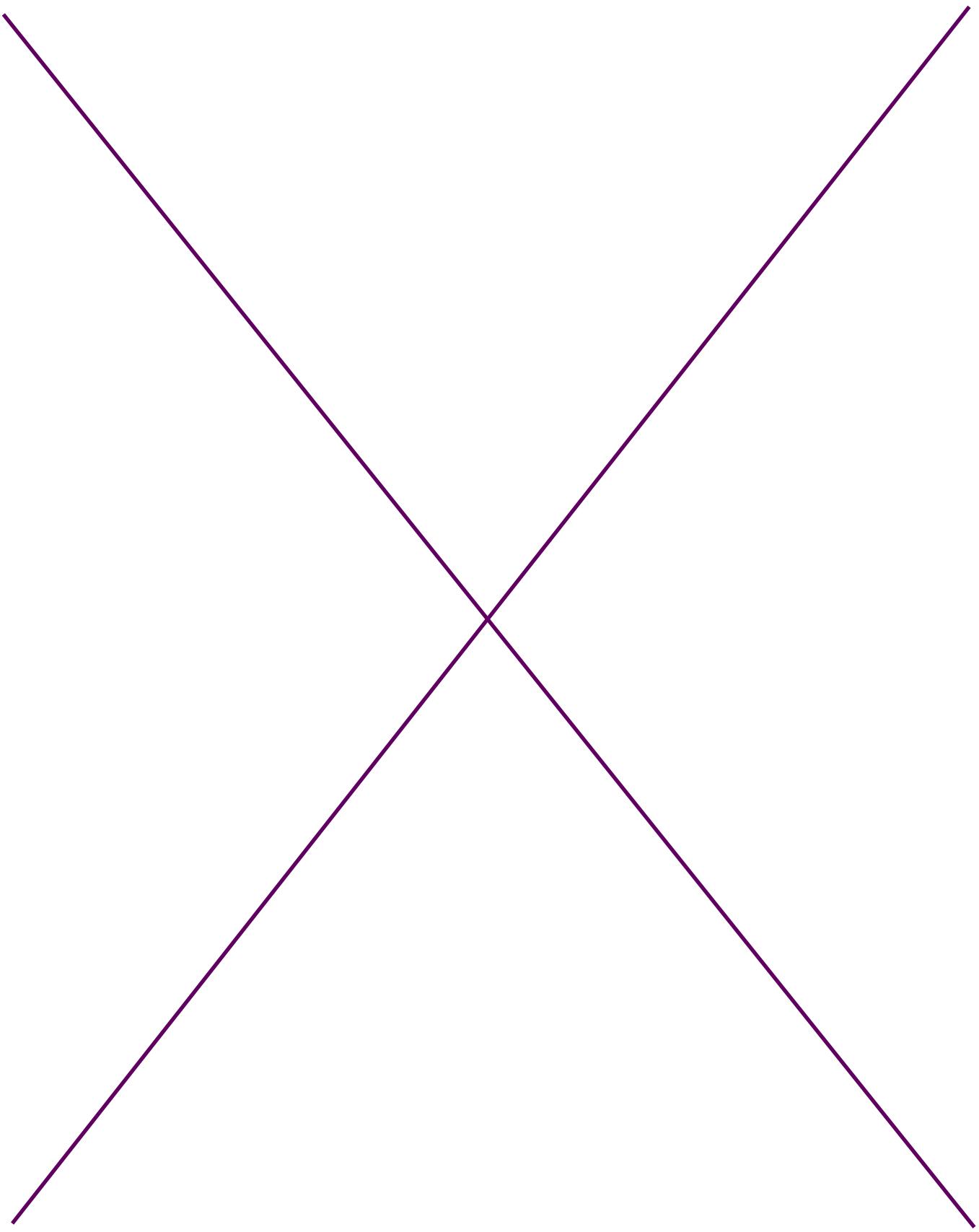
Process:

Former NASA and now a Sierra Nevada Corporation Hardware Engineer stopped by our build space to check out our robot. Mike Garske has worked both on the Space Shuttle at KSC and JSC and now is a Chief Engineer for the Dream Chaser.



Signature: Shelby Greer

Date: Dec. 23, 2018



Signature : Shelby Greer

Date: Dec. 23, 2018

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: January 5, 2019

Purpose of the Activity:

Update the teleop code to support the new arms

Build the second servo box for the arm

Updated the autonomous program to work correctly

Process:

We started the meeting by updating the hardware configuration to have our two safety arms. The purpose of the arms is to make sure that we do not get too close to the lander for hanging at the end or dumping minerals into the lander. We also determined that the control should be on the drivers controller and we will be using button “a” and “x” to raise and lower the arm.

Configuration code:

```
Servo right_arm;  
double right_arm_power = 0.5;  
Servo left_arm;  
double left_arm_power = 0.5;  
  
right_arm = hardwareMap.servo.get("right_arm");  
left_arm = hardwareMap.servo.get("left_arm");
```

Teleop Code:

```
// MOVE THE ARMS  
if (gamepad1.y) {  
    robot.right_arm_power = .5;  
    robot.right_arm.setPosition(robot.right_arm_power);  
    robot.left_arm_power = .5;  
    robot.left_arm.setPosition(robot.left_arm_power);  
}  
if (gamepad1.a) {
```

Signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:

```
robot.right_arm_power = 1;  
robot.right_arm.setPosition(robot.right_arm_power);  
robot.left_arm_power = 0;  
robot.left_arm.setPosition(robot.left_arm_power);  
}
```

After updating the code we connected the wires to the hubs and end connected the servos so we can determine which values to set to the servos when they are to be deployed.



Signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:

So, where is the servo ports????

After we found the ports we connected the robot to the computer and used the REV Interface function and determined that we will use the value of "0" and "1" to extend the arms to the proper position since their start position is "0.5". The interface function moved the one servo and we determined the proper values to use. After it showed the ports being used where ports 0 and 1, and when we went back to our program configuration they were supposed to be ports 2 and 3. The robot would not come up and run so we needed to troubleshoot this issue after we built the second servo box.

This is the third iteration of the servo box. The first two had the following errors:

1. The wood used was not thick enough to properly put the screws in to hold it together
2. The construction was such that the servo was being squished not allowing it to move freely

Following is the construction of the final or third iteration of the box.

1. Because the plywood available is not 1/4 inch it was necessary for us to mold the wood to the proper size. Using the planer we created the piece of wood the correct thickness.
2. Each piece was traced on the piece of wood and cut out.
3. We used a wood super glue to glue the box side sections onto the back piece.
4. We then completed the box by putting screws in the top side to close the box.
5. We made the two servo boxes mirror images of each other



signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

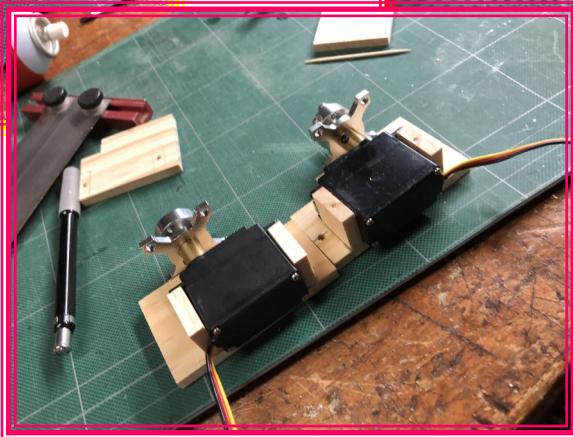
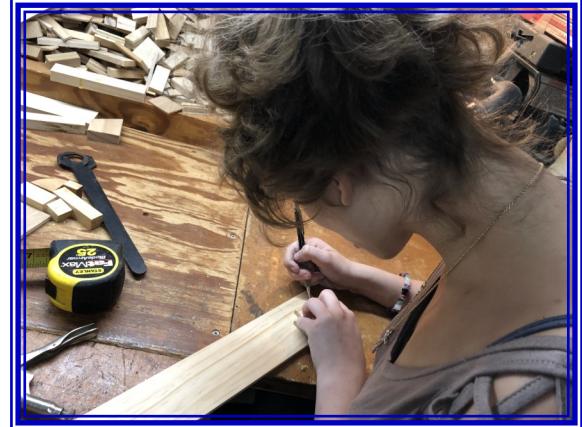
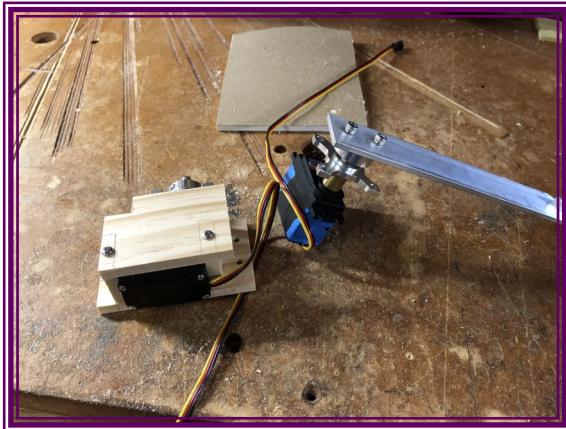
Engineering Activity Continued

Date: January 5, 2019

Process:

since they go on opposite side of the robot. We used the radial arm and band saws to cut out the parts to

the servo boxes. Pictured is one completed and some of the process taken to create the second servo box. A drill press was used to drill the



holes for mounting the
After mounting the new
we started troubleshooting
bot's phones could not find
the REV Hub Interface pro-

servo boxes on the robot.
servo boxes for our arms
the issue where the ro-
the REV hubs after using
gram.

Signature: Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

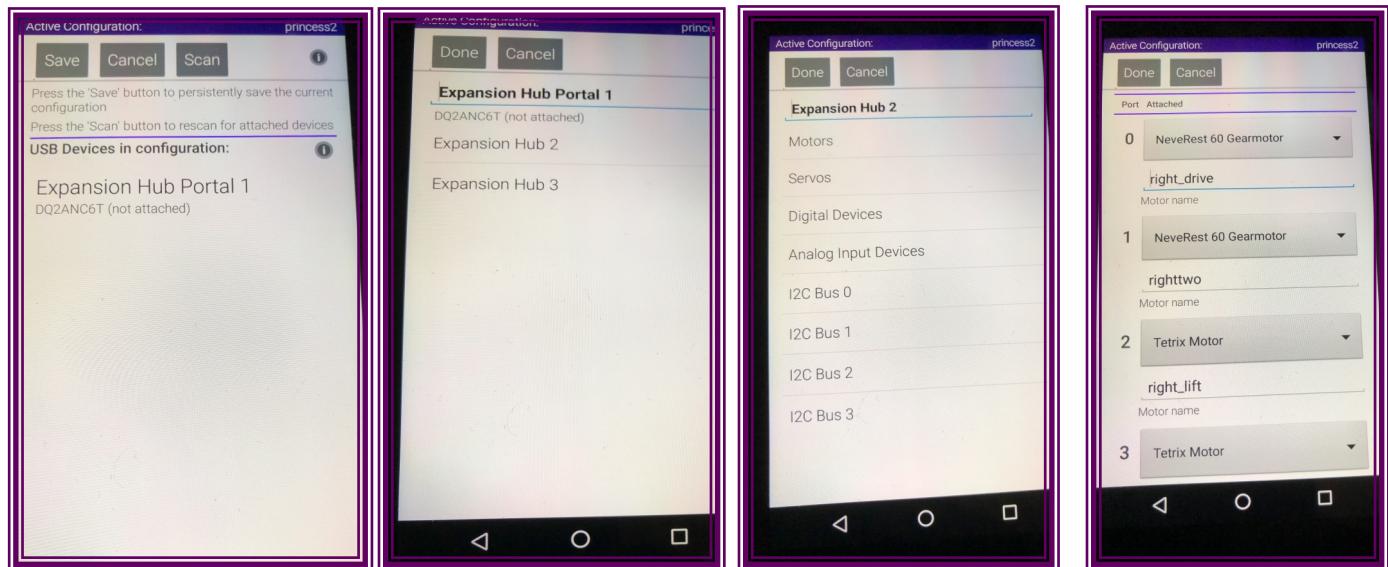
Process:

After changing the hub names on the phones several times not being successful. It was determined that we needed to reset the port identifications for each of the two hubs using the procedure provided here:

https://github.com/ftctechnh/ftc_app/wiki/Using-Two-Expansion-Hubs

In doing this we needed to re-scan the hardware wiping out the current configuration. We needed to do this on both of our phones. Noting that we had not captured this information in our document in case this happened again we are adding it here:

We have two hub numbering them Port 2 and Port3. Following is the information for the hardware configuration:



signature : Shelby Greer

Date: Jan. 5, 2019

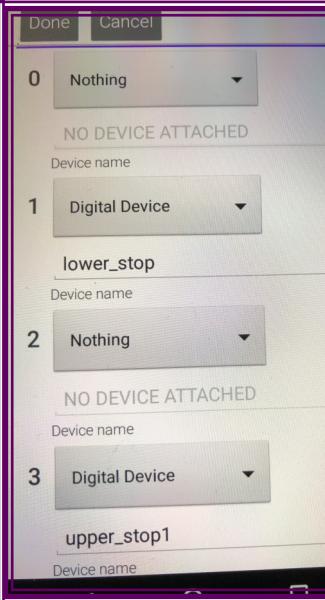
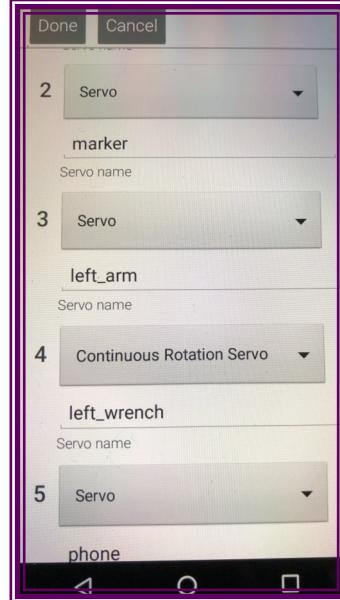
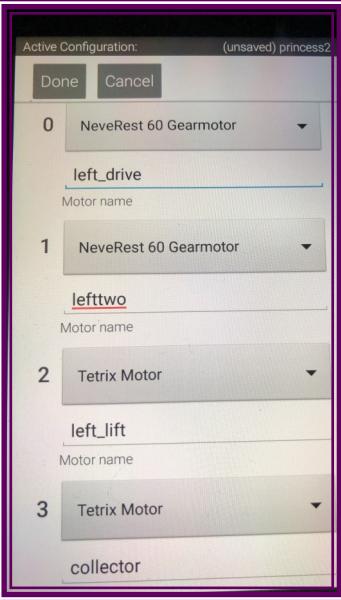
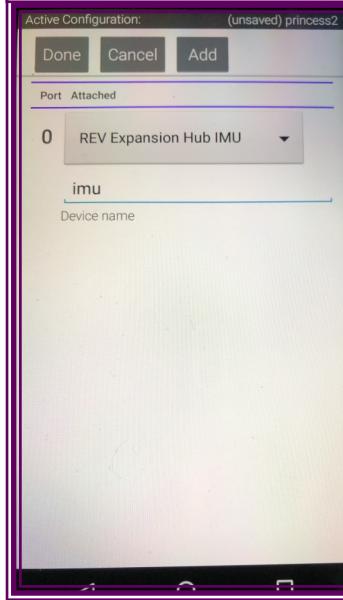
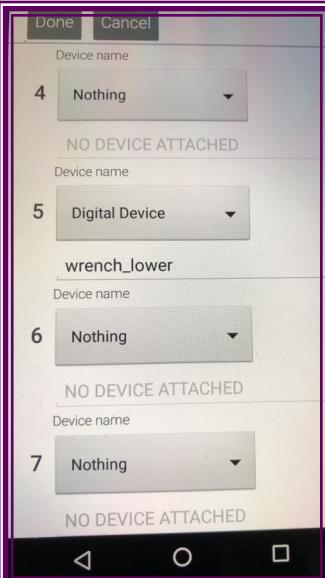
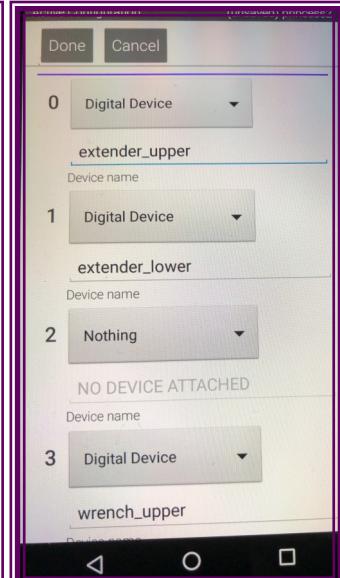
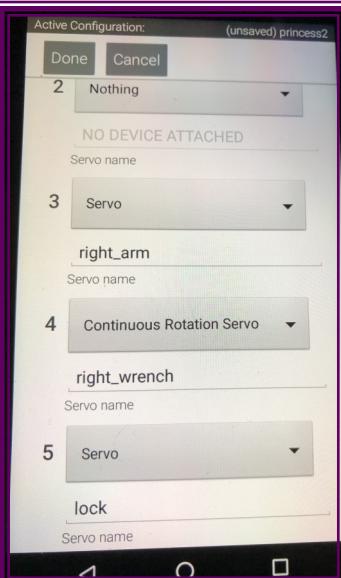
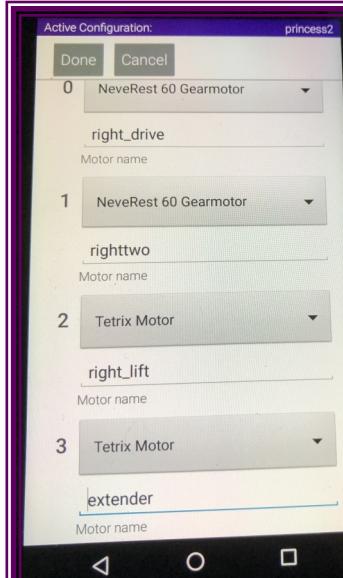
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:



Port 2 is hardware device—DQ2ANC6T and Port 3 is hardware device DQ1033OJ. After we corrected the configuration on the

Signature: Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

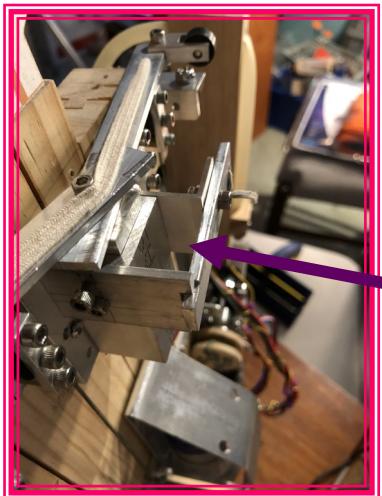
Date: January 5, 2019

Process:

robot phone we finished added the new pieces to the robot. The new arms and new end effector. Minor adjustments were made to the arms so that the start position was straight down allowing the up position to be 90 degrees from the edge of the robot. Now we can approach the lander and know that we are not too close. Additional testing is needed to checkout the



adjusted
margin of
know
der!!!!



locking mechanism. We added a 1/2 inch to the depth to allow for a larger error for hooking on to the lander. Never where you are going to find a robot spi-

Latches onto the lander



Signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

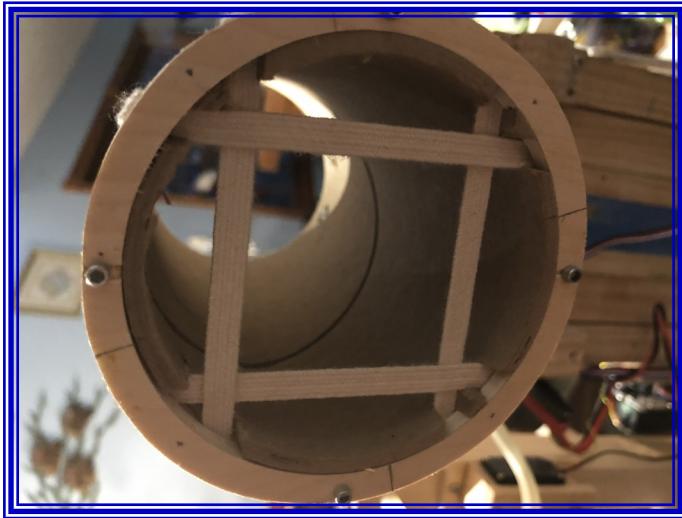
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:

The new end effector has been styled off of one our team did 4 years ago where we used a cardboard shipping tube and instead of two pieces of elastic and a trap door we are using four pieces of elastic that crisscross each other where you can pick up either a gold or silver mineral.



Another important thing that we learned that it is best to use a drill press if you want your holes you drilled to be straight.



Signature: Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

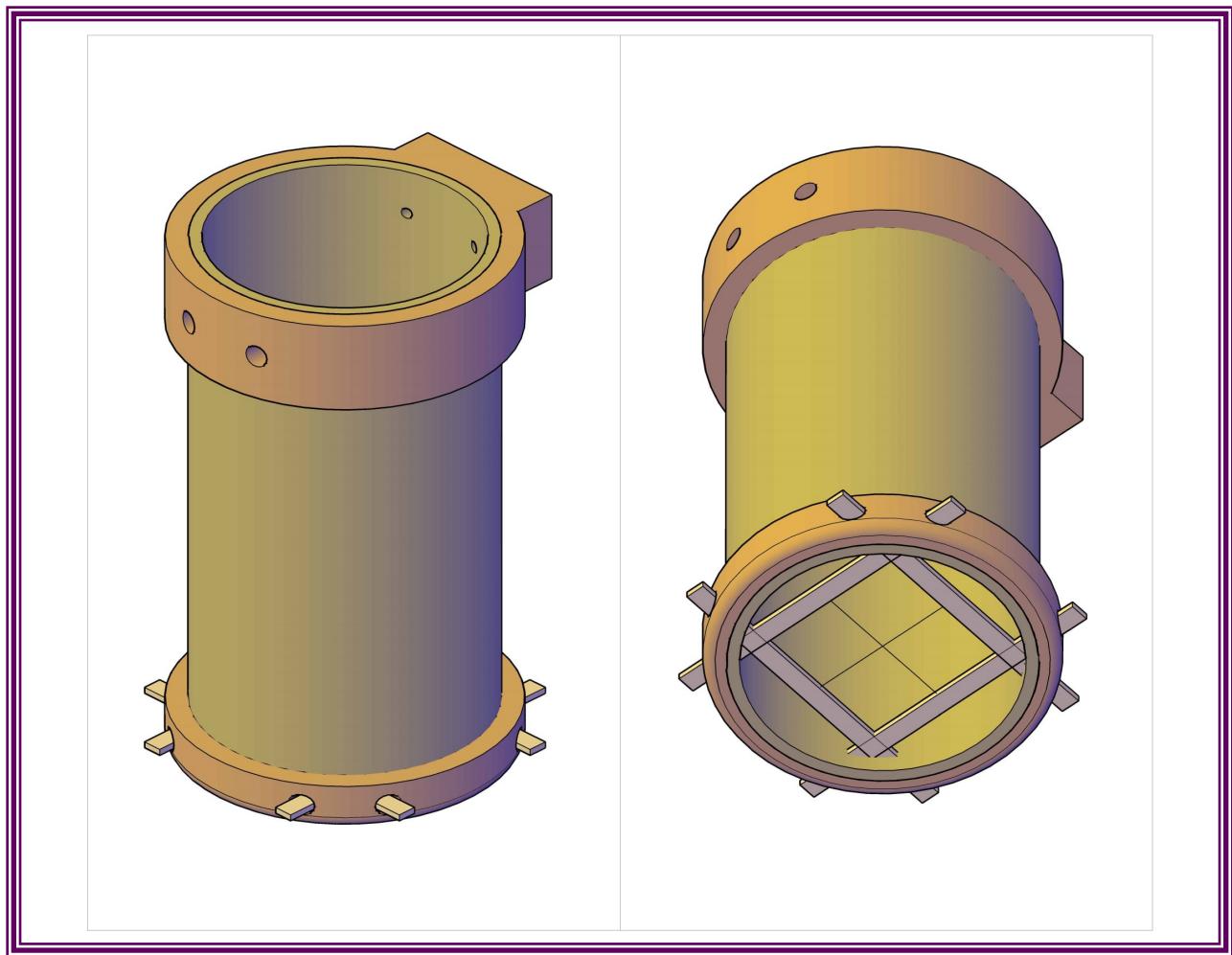
Engineering Activity Continued

Date: January 5, 2019

Process:

Following are the different Engineering Drawings that helped us with the modifications to our robot.

The Tube End Effector which is much lighter than our first End Effector which we wanted to if we could



design and create a mechanical device to pick up the minerals.

Signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

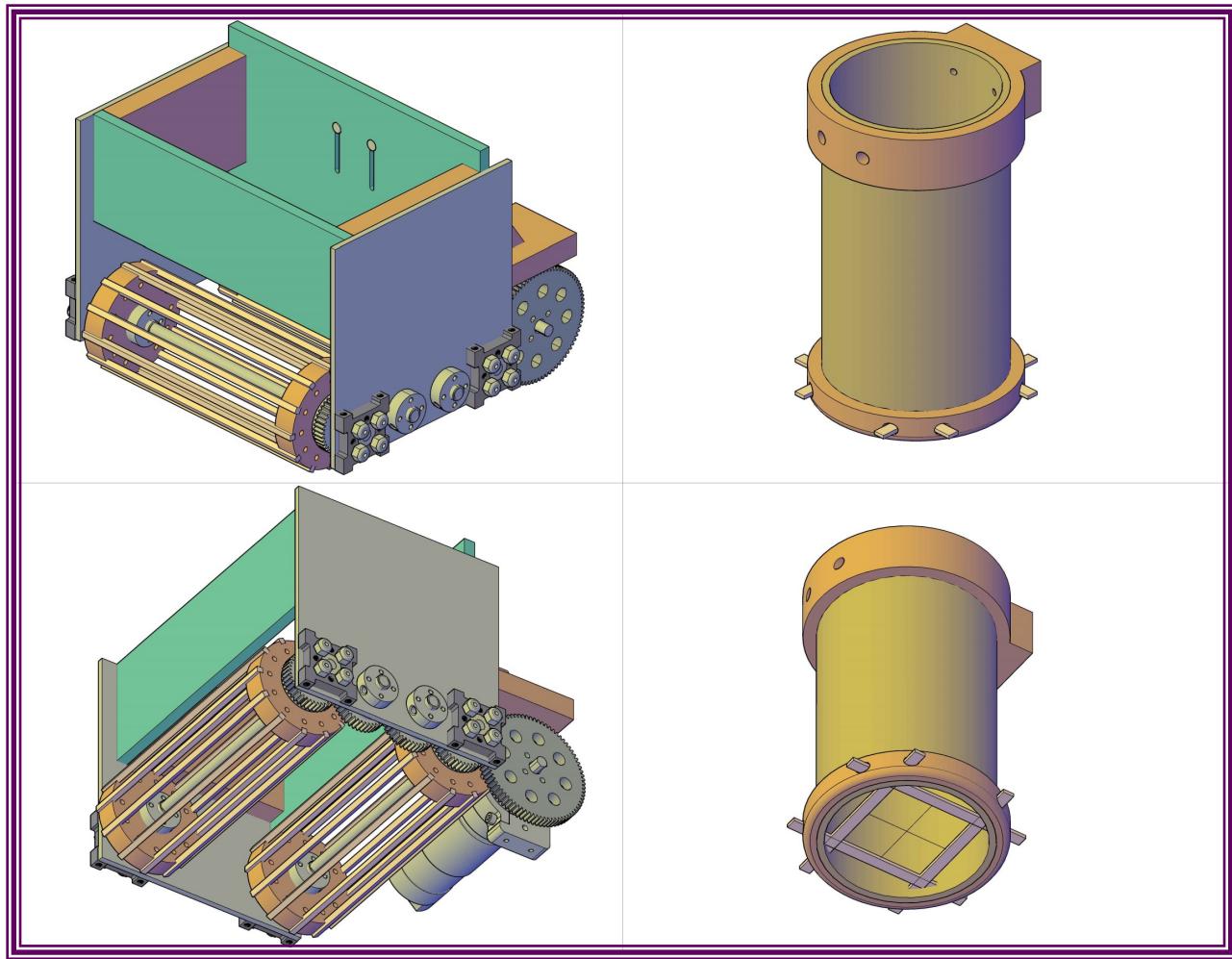
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:

Seen in the design is the AndyMark Motor that added the weight to the End Effector. The old End Effector work well to pickup the items but the weight made it difficult in controlling the lift into the lander.



Signature : Shelby Greer

Date: Jan. 5, 2019

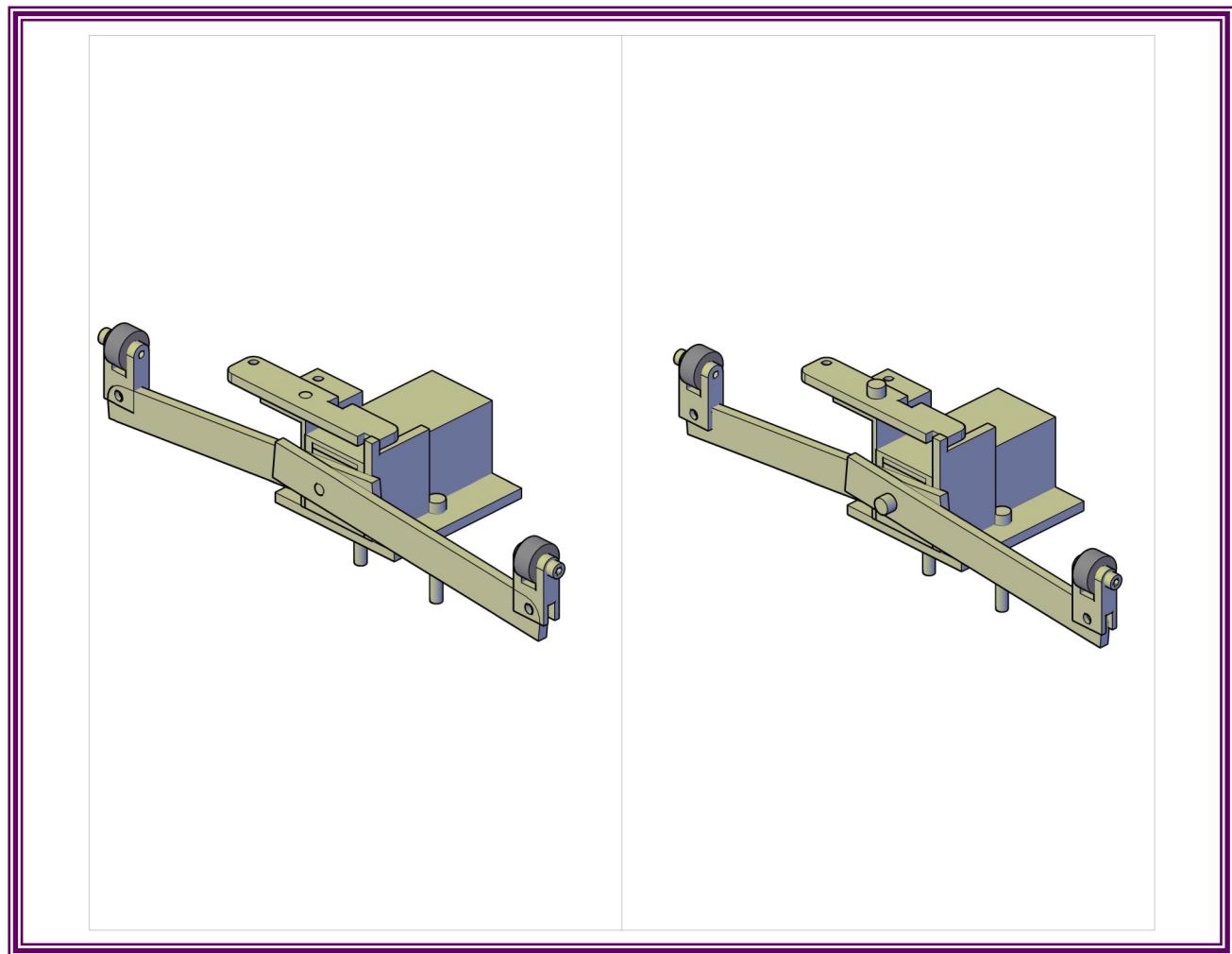
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 5, 2019

Process:



This is the old and new latch so we can hang from the lander at "End Game"

Signature: Shelby Greer

Date: Jan. 5, 2019

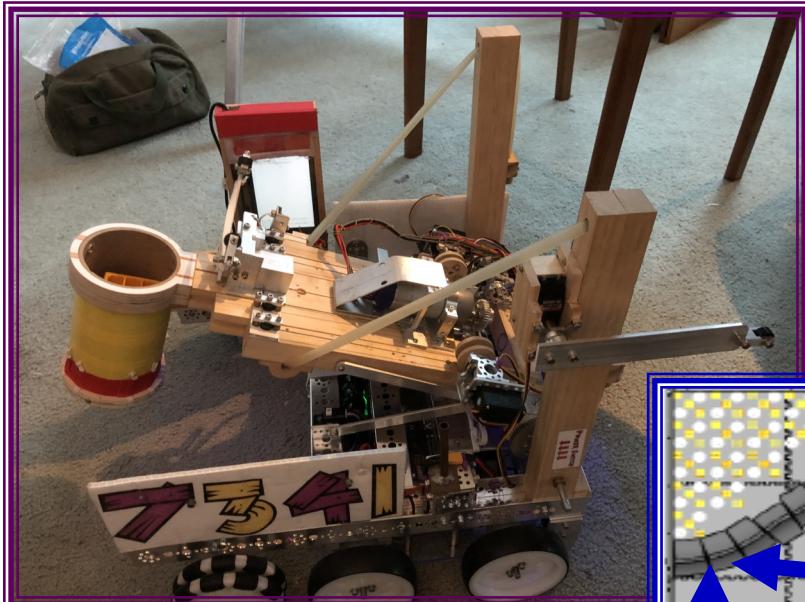
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

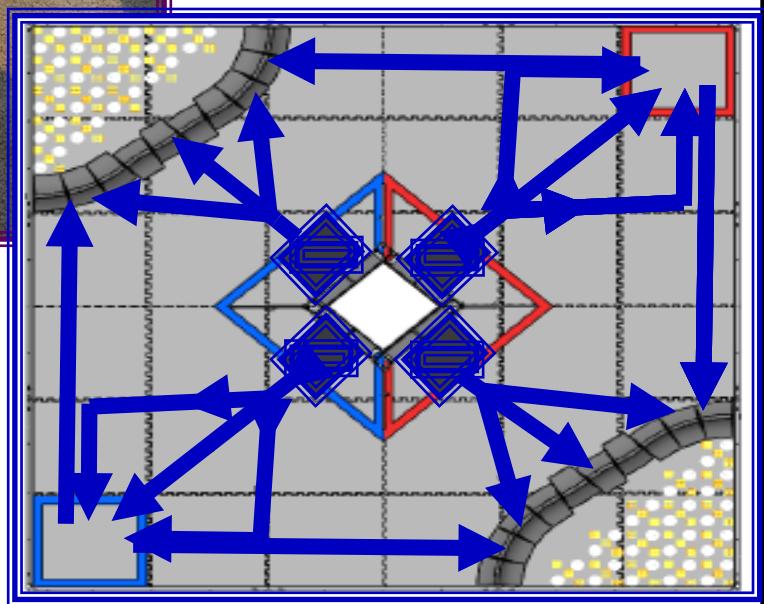
Date: January 5, 2019

Process:



Princess Charlie is ready for final testing before our next Meet and we are now creating goodie bags for all the teams attending.

We have laid out the new paths for our Autonomous program which will be using the depot side so we can drop the marker.



Working on our **6 FIRST Core Value...**

We express the FIRST® philosophies of Gracious Professionalism® and Coopertition® through our Core Values:

- **Discovery:** We explore new skills and ideas.
- **Innovation:** We use creativity and persistence to solve problems.
- **Impact:** We apply what we learn to improve our world.
- **Inclusion:** We respect each other and embrace our differences.
- **Teamwork:** We are stronger when we work together.
- **Fun:** We enjoy and celebrate what we do!

Signature : Shelby Greer

Date: Jan. 5, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: January 11-12, 2019

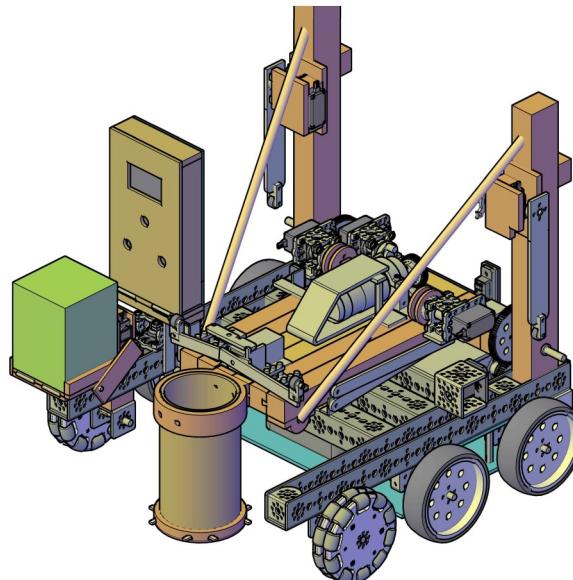
Purpose of the Activity:

Preparing for Meet 3	Compete in the meet
Sleep over with the FIRST LEGO League Team	Have a ton of fun
Drive to Sanford	Prepare for the League Championship

Process:

We now have a new configuration for our robot and as we head to Sanford our Sister FLL team heads to Melbourne for the competition.... Good Luck to them.

We have completed the upgrades to our robot to use a tube collector, and our autonomous has been tweek so that it should find the mark each time.... We are so ready for a GREAT day.



Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

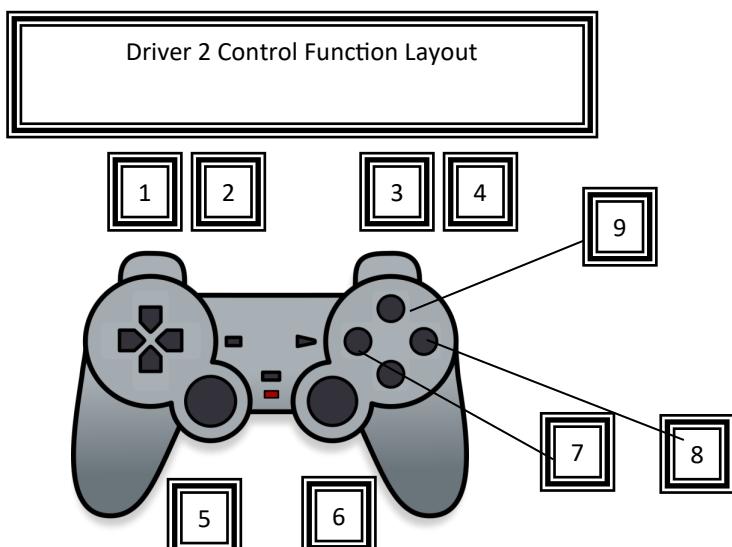
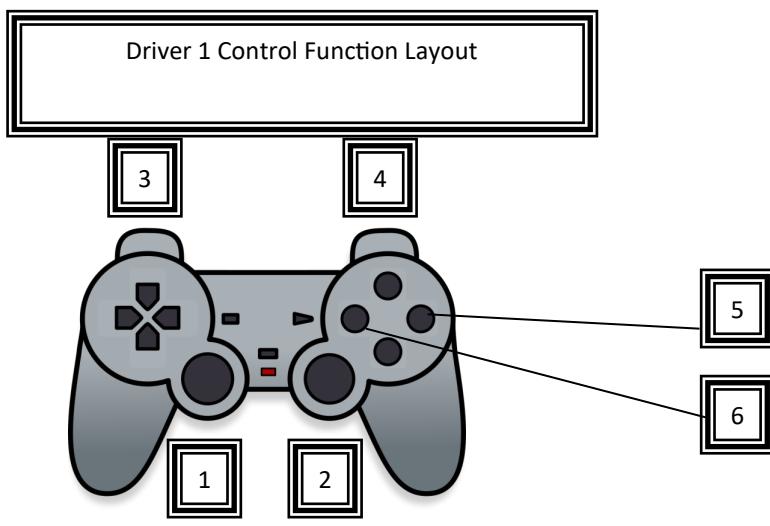
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Described below is the functional layout for the robot controller buttons.



Driver 1 Control Function Definition

1. Left Wheels control
2. Right Wheels control
3. Move the phone left
4. Move the phone right
5. Arms up
6. Arms down

Driver 2 Control Function Definition

1. Lock the second layer of the slide
2. Unlock the second layer of the slide
3. Raise the second layer of the slide
4. Lower the second layer of the slide
5. Lift and lower grabber
6. Extend and retract grabber
7. Make Grabber go backwards
8. Make Grabber pick up items
9. Stop Grabber

Signature : Jessica Anderson

Date: Jan. 12, 2019

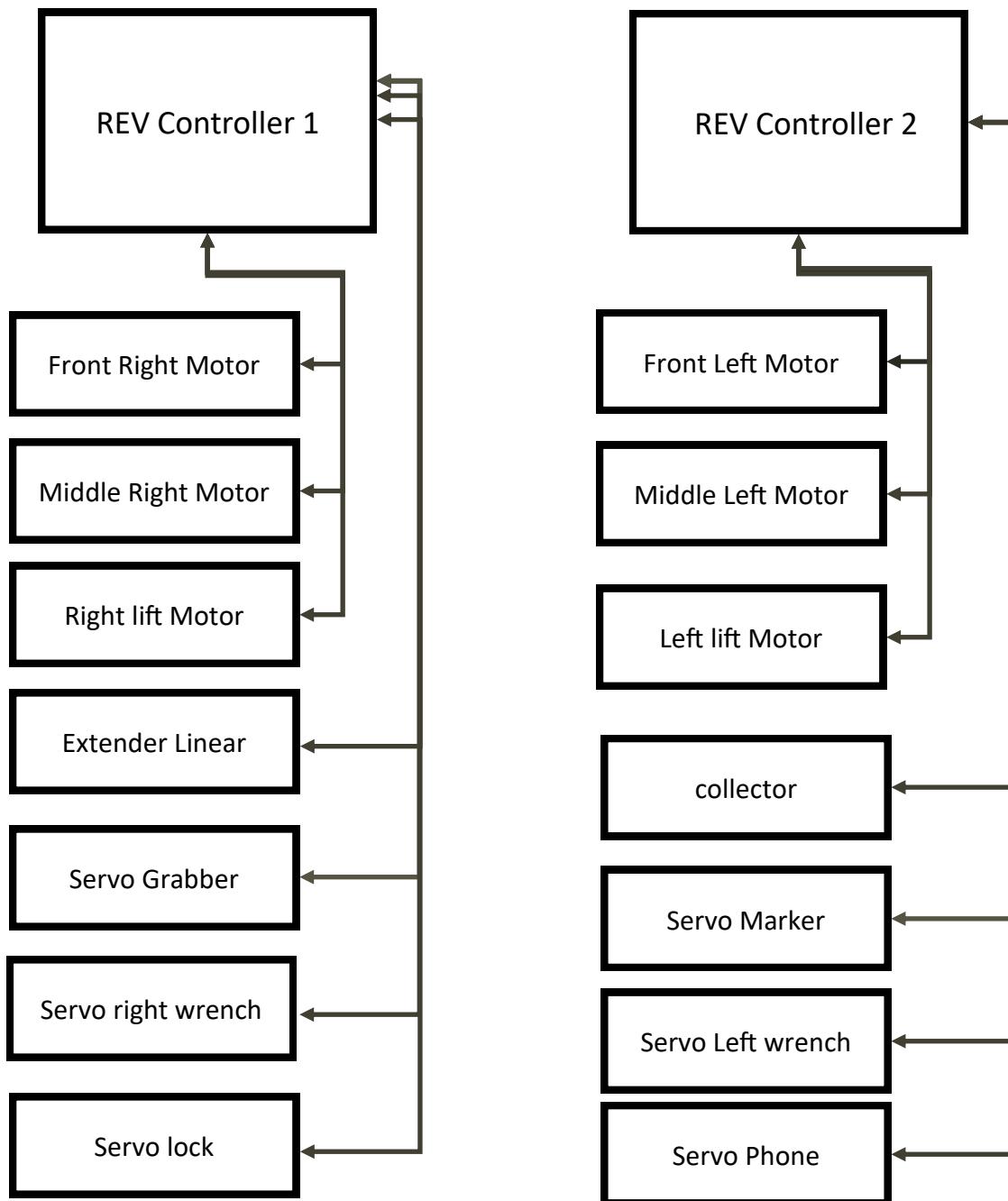
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:



Signature : Jessica Anderson

Date: Jan. 12, 2019

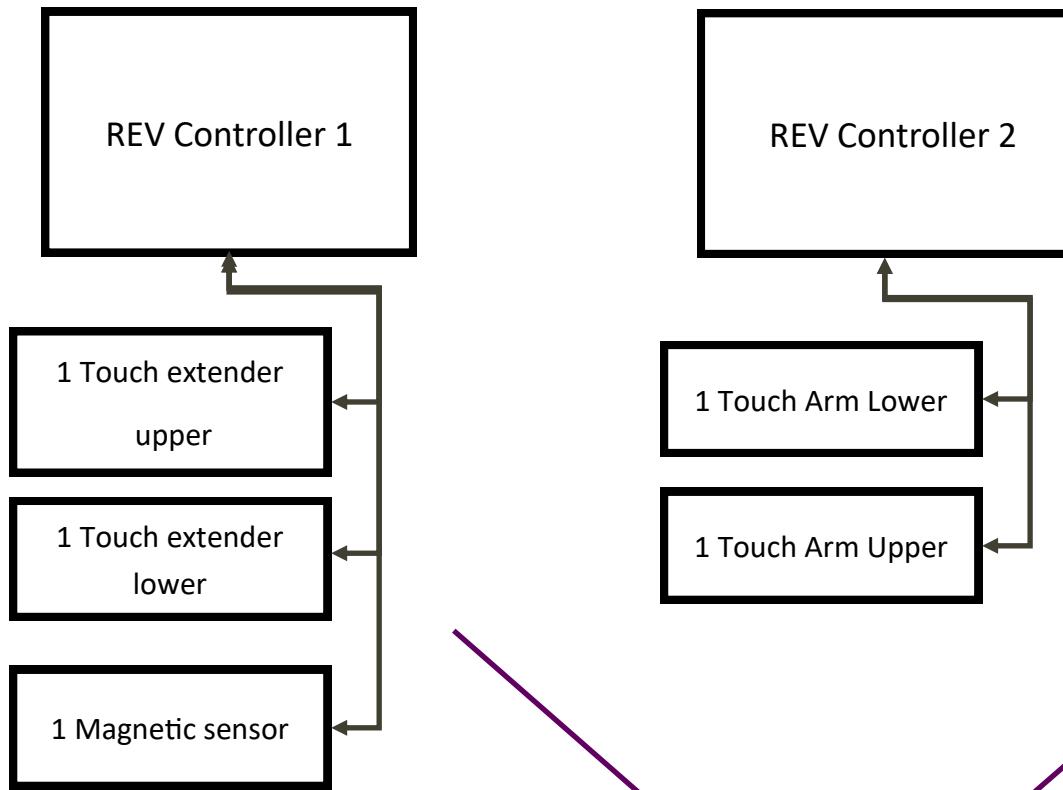
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:



Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Following are the programs used during Meet 3

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

The function of this program is to run teleop.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackables;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotor.RunMode;
import static com.qualcomm.robotcore.util.Range.clip;
import static java.lang.Math.abs;
import static java.lang.String.format;
import static org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer.CameraDirection.BACK;

@TeleOp(name = "FF: ChickenNoodleFrenchFry", group = "FF")
//@Disabled
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
public class ChickenNoodleFrenchFry extends LinearOpMode {  
    /* Declare OpMode members. */  
  
    private ElapsedTime runtime = new ElapsedTime();  
  
    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle();  
    private static final String VUFORIA_KEY = "AW/Sw13////  
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-  
Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/  
MZz7Zvp35rYD6lavfkgCMZ0DVAXHBv4J5LlrGIVXYfhhS1NkITGPDqVRW2aBmKLwctHZaztzycau3g//  
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/  
rEsoZMaPnPfKYGcE-  
Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/  
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here  
/**  
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
 * localization engine.  
 */  
private VuforiaLocalizer vuforia;  
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFObjectDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";
private boolean targetVisible = false;
// private boolean targetVisible = false;Valid choices are: BACK or FRONT

private static final VuforiaLocalizer.CameraDirection CAMERA_CHOICE = BACK;
// Variables to be used for later
private VuforiaLocalizer vuforiaLocalizer;
private VuforiaLocalizer.Parameters parameters;
private VuforiaTrackables visionTargets;
@Override
public void runOpMode() throws InterruptedException {

    // data for moving the phone
    boolean phone_locked;
    // set to be at the bottom
    int extender_position1 = 0;
    float extender_power1 = 0;
    int last_extender_position = -999;
    int extender_position = 2;
    int extender_direction = 0;
    int attop = 0;
    int atbottom = 1;
    int location = 0;
    int phonecyclecount;
    phone_locked = false;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
phonecyclecount = 0;

// set phone data
// State used for updating telemetry
// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Setting Up ChickenNoodleFrenchFry"); // 
telemetry.update();

/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */

/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

// end of the front-end of the robots definitions
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
telemetry.addData("Text", "Waiting to start ChickenNoodleFrenchFry");
telemetry.update();
// this is to initialize the camera for target viewing
// initVufor();    telemetry.update();
/*
if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3", "setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3", "back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}
*/
waitForStart();

// put servos in start position

robot.init(hardwareMap, 2);
robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
//move phone out of the way
robot.phone_position = .84;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
robot.phone.setPosition(robot.phone_position);

robot.lock_power = .5;
robot.lock.setPosition(robot.lock_power);
// Set up our telemetry dashboard
composeTelemetry();
// setting up timer
runtime.reset();
// while the op mode is active, loop and read the RGB data.
// Note we use opModelsActive() as our loop condition because it is an interruptible method.
while (opModelsActive()) {

/*
* Gamepad 1 controls the motors via the left/right stick
*/
// this is for the motor control function
// forward is negative power value
// backwards is positive power value
int drive_mode = 2;
if (gamepad1.right_stick_y != 0 || gamepad1.left_stick_y != 0 ||
    gamepad1.right_stick_x != 0 || gamepad1.left_stick_x != 0) {
    if (gamepad1.right_stick_y <= 0) {
        robot.right = ((-gamepad1.right_stick_y - (abs(gamepad1.right_stick_x))) / (float) .5);
    } else {
        robot.right = ((-gamepad1.right_stick_y + abs(gamepad1.right_stick_x)) / (float) .5);
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
if (gamepad1.left_stick_y <= 0) {  
    robot.left = ((-gamepad1.left_stick_y - (abs(gamepad1.left_stick_x))) / (float) .5);  
} else {  
    robot.left = ((-gamepad1.left_stick_y + abs(gamepad1.left_stick_x)) / (float) .5);  
}  
  
// clip the right/left values so that the values never exceed +/- 1  
robot.right = clip(robot.right, -1, 1);  
robot.left = clip(robot.left, -1, 1);  
  
// scale the joystick value to make it easier to control  
// the robot more precisely at slower speeds.  
robot.right = (float) scaleInput(robot.right);  
robot.left = (float) scaleInput(robot.left);  
  
// write the values to the motors  
setDrivePower(robot.right, robot.left, drive_mode);  
} else {  
    robot.right = 0;  
    robot.left = 0;  
    // clip the right/left values so that the values never exceed +/- 1  
    robot.right = clip(robot.right, -1, 1);  
    robot.left = clip(robot.left, -1, 1);  
  
    // scale the joystick value to make it easier to control  
    // the robot more precisely at slower speeds.  
    robot.right = (float) scaleInput(robot.right);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
robot.left = (float) scaleInput(robot.left);

// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
}

// Gamepad 2

// Lift (Raise and Lower) the end effector
// You should not go much past 90degrees
// up is negative value on the controller
// down is positive value on the controller
// lower_stop is to make sure the motor stop when touching the floor
// lift_top is to stop when just past being vertical
// if the digital channel returns true it's HIGH and the button is unpressed.
if ((gamepad2.left_stick_y < 0 && robot.upper_stop.getState() == true ) ||
    (gamepad2.left_stick_y > 0 && robot.lower_stop.getState() == true )) {
    // set lift speed to a constant value use the negative
    // negate the power value to get the motors going in the direction desired

    if (gamepad2.left_stick_y < -.95) gamepad2.left_stick_y =(float) -.95;
    if (gamepad2.left_stick_y > .75) gamepad2.left_stick_y =(float) .75;

    robot.lift_power = -gamepad2.left_stick_y;

    // clip the right/left values so that the values never exceed +/- 1
    robot.lift_power = clip(robot.lift_power, -1, 1);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.

robot.lift_power = (float) scaleInput(robot.lift_power);

robot.right_liftdrive.setPower(robot.lift_power);
robot.left_liftdrive.setPower(robot.lift_power);

} else {

// stop the lift motor

robot.lift_power = 0;
robot.lift_power = (float) scaleInput(robot.lift_power);
robot.right_liftdrive.setPower(robot.lift_power);
robot.left_liftdrive.setPower(robot.lift_power);

}

// raise the mineral grabber up/down

// up is negative power value

// going extender position = 2 we are at the bottom

extender_position1 = robot.extenderdrive.getCurrentPosition();
if (extender_position1 > last_extender_position) extender_direction = 2;
if (extender_position1 < last_extender_position) extender_direction = 1;

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
if ((gamepad2.right_stick_y < 0 )
 || (gamepad2.right_stick_y > 0)){

    last_extender_position = extender_position1;

    // speed to a constant value

    robot.extender_power = gamepad2.right_stick_y;
    // clip the right/left values so that the values never exceed +/- 1
    extender_power1 = clip(robot.extender_power, -1, 1);
    robot.extender_power = extender_power1;
    // scale the joystick value to make it easier to control
    // the robot more precisely at slower speeds.

    if (!robot.extender_lower.getState() && extender_direction == 1 && atbottom == 0) {
        attop = 1;
        robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
        robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
        location = robot.extenderdrive.getCurrentPosition();
    }
    if (!robot.extender_lower.getState() && extender_direction == 2 && attop == 0){
        atbottom = 1;
        robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
        robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);
        location = robot.extenderdrive.getCurrentPosition();
    }
    if (robot.extender_lower.getState()){
        atbottom = 0;
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
attop = 0;
}

if (!robot.extender_lower.getState() && extender_direction == 1 && robot.extender_power < 0 /* && attop == 1 */ ) robot.extender_power = 0;
if (!robot.extender_lower.getState() && extender_direction == 2 && robot.extender_power > 0 /* && atbottom == 1 */ ) robot.extender_power = 0;
// over ride because you are at the top or bottom and you need to move
if (!robot.extender_lower.getState() && atbottom == 1 && extender_power1 < 0) robot.extender_power = extender_power1;
if (!robot.extender_lower.getState() && attop == 1 && extender_power1 > 0) robot.extender_power = extender_power1;

robot.extender_power = (float) scaleInput(robot.extender_power);
robot.extenderdrive.setPower(robot.extender_power);

} else {

// stop the extension motor

robot.extender_power = 0;
robot.extender_power = (float) scaleInput(robot.extender_power);
robot.extenderdrive.setPower(robot.extender_power);

}

// wrench up/down
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
if (gamepad2.right_bumper && robot.wrench_upper.getState() == true ) {  
    //going up  
    //make sure you are unlocked  
    //unlocking  
    robot.lock_power = .5;  
    robot.lock.setPosition(robot.lock_power);  
  
    robot.wrench_power = .75;  
    robot.left_wrench.setPower(robot.wrench_power);  
    robot.right_wrench.setPower(-robot.wrench_power);  
} else if (gamepad2.right_trigger != 0 && robot.wrench_lower.getState() == true) {  
    //going down  
    robot.wrench_power = .75;  
    robot.left_wrench.setPower(-robot.wrench_power);  
    robot.right_wrench.setPower(robot.wrench_power);  
} else {  
    //stop wrench  
    robot.wrench_power = 0;  
    robot.left_wrench.setPower(robot.wrench_power);  
    robot.right_wrench.setPower(robot.wrench_power);  
}  
  
// set the lock so we can lift the robot  
if (gamepad2.left_bumper) {  
    //locking  
    robot.lock_power = .16;  
    robot.lock.setPosition(robot.lock_power);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
 } else if (gamepad2.left_trigger != 0) {  
     //unlocking  
     robot.lock_power = .5;  
     robot.lock.setPosition(robot.lock_power);  
 }  
  
// pick the minerals up  
if (gamepad2.b) {  
    robot.collector_power = 1;  
    robot.collector_power = (float) scaleInput(robot.collector_power);  
    robot.collectordrive.setPower(robot.collector_power);  
}  
if (gamepad2.x) {  
    robot.collector_power = 0;  
    robot.collector_power = (float) scaleInput(robot.collector_power);  
    robot.collectordrive.setPower(robot.collector_power);  
}  
// spit the minerals out  
if (gamepad2.y) {  
    robot.collector_power = -1;  
    robot.collector_power = (float) scaleInput(robot.collector_power);  
    robot.collectordrive.setPower(robot.collector_power);  
}  
if (gamepad2.a) {  
    robot.collector_power = 0;  
    robot.collector_power = (float) scaleInput(robot.collector_power);  
    robot.collectordrive.setPower(robot.collector_power);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
}

// MOVE THE ARMS

if (gamepad1.y) {
    robot.right_arm_power = .5;
    robot.right_arm.setPosition(robot.right_arm_power);
    robot.left_arm_power = .5;
    robot.left_arm.setPosition(robot.left_arm_power);
}

if (gamepad1.a) {
    robot.right_arm_power = 1;
    robot.right_arm.setPosition(robot.right_arm_power);
    robot.left_arm_power = 0;
    robot.left_arm.setPosition(robot.left_arm_power);
}

// right bumper move out the phone
// left bumper move in the phone

if (gamepad1.right_bumper && robot.phone.getPosition() >= 0 && !phone_locked) {
    //move phone (bottom is 1)
    robot.phone_position += .1;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;

} else if (gamepad1.left_bumper && robot.phone.getPosition() <= 1 && !phone_locked) {
    // phone (top position is 0)
    robot.phone_position -= .1;
    robot.phone.setPosition(robot.phone_position);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
    phone_locked = true;
}
if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 100){
        phone_locked = false;
        phonecyclecount = 0;
    }
}

telemetry.addData("0", "FF ChickenNoodleFrenchFry - %2.5f S Elapsed", runtime.seconds());
// get the value of the light

telemetry.addData("D1", format("Rx: %.2f Lx: %.2f Ry: %.2f Ly: %.2f", gamepad1.right_stick_x,
gamepad1.left_stick_x, gamepad1.right_stick_y, gamepad1.left_stick_y));
telemetry.addData("D2", format("Rx: %.2f Lx: %.2f ", gamepad2.right_stick_x,
gamepad2.left_stick_x));
telemetry.addData("1", format("R - %.2f L - %.2f - Lift - %.2f - Extender - %.2f", robot.right, ro-
bot.left, robot.lift_power,robot.extender_power));

// telemetry.addData("3", format("Floor Color %2d Blue %2d Red %2d", robot.floor_color.alpha
(), robot.floor_color.blue(), robot.floor_color.red())));
telemetry.addData("4", format("Lower %s upper %s", robot.lower_stop.getState(), ro-
bot.upper_stop.getState()));
telemetry.addData("Extender state is ", robot.extender_lower.getState());
telemetry.addData("5", format("phone position %.2f - %d", robot.phone.getPosition
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
()>robot.extenderdrive.getCurrentPosition());  
    telemetry.addData("6",format("extnder direction %d",extender_direction));  
    telemetry.addData("9",format("atbottom %d atop %d",atbottom, attop));  
    telemetry.addData("9",format("location %d ",location));  
    telemetry.update();  
    idle(); // Always call idle() at the bottom of your while(opModelsActive()) loop  
}  
}  
  
/*  
 * This method scales the joystick input so for low joystick values, the  
 * scaled value is less than linear. This is to make it easier to drive  
 * the robot more precisely at slower speeds.  
 */  
double scaleInput(double dVal) {  
    double[] scaleArray = {0.0, 0.05, 0.09, 0.10, 0.12, 0.15, 0.18, 0.24,  
        0.30, 0.36, 0.43, 0.50, 0.60, 0.72, 0.85, 1.00, 1.00};  
;  
    // get the corresponding index for the scaleInput array.  
    int index = (int) (dVal * 16.0);  
    if (index < 0) {  
        index = -index;  
    } else if (index > 16) {  
        index = 16;  
    }  
  
    double dScale = 0.0;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
if (dVal < 0) {  
    dScale = -scaleArray[index];  
} else {  
    dScale = scaleArray[index];  
}  
  
return dScale;  
}  
  
public void setDrivePower(float rightPower, float leftPower, int power_mode) {  
    // telemetry.addData("2", format("PM - %d", power_mode));  
    // telemetry.update();  
    if (power_mode == 2) {  
        // set front and back motors full power  
        robot.rightdrive.setPower(rightPower);  
        robot.leftdrive.setPower(leftPower);  
        robot.righttwo.setPower(rightPower);  
        robot.lefttwo.setPower(leftPower);  
    } else if (power_mode == 1) {  
        // set front motors  
        robot.rightdrive.setPower(0);  
        robot.leftdrive.setPower(0);  
        robot.righttwo.setPower(0);  
        robot.lefttwo.setPower(0);  
    } else {  
        // set front motors  
        robot.rightdrive.setPower(0);  
    }
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
robot.leftdrive.setPower(0);
robot.righttwo.setPower(0);
robot.lefttwo.setPower(0);
}

}

public void setDriverMode(RunMode mode) {

    if (robot.leftdrive.getMode() != mode) {
        robot.leftdrive.setMode(mode);
    }
    if (robot.rightdrive.getMode() != mode) {
        robot.rightdrive.setMode(mode);
    }
}

//-----
// Telemetry Configuration
//-----

void composeTelemetry() {

    telemetry.addData("ball color red","ball red color ");
    telemetry.update();
    //At the beginning of each telemetry update, grab a bunch of data
    //from the IMU that we will then display in separate lines.
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
telemetry.addAction(new Runnable() { @Override public void run()
{
    // Acquiring the angles is relatively expensive; we don't want
    // to do that in each of the three items that need that info, as that's
    // three times the necessary expense.

    robot.angles = robot imu.getAngularOrientation(AxesReference.INTRINSIC, AxesOrder.ZYX, AngleUnit.DEGREES);
}

});
```

```
telemetry.addLine()
.addData("status", new Func<String>() {
    @Override public String value() {
        return robot imu.getSystemStatus().toShortString();
    }
})
.addData("calib", new Func<String>() {
    @Override public String value() {
        return robot imu.getCalibrationStatus().toString();
    }
});
```

```
telemetry.addLine()
.addData("heading", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot.angles.angleUnit, robot.angles.firstAngle);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
        }
    })
.addData("roll", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot.angles.angleUnit, robot.angles.secondAngle);
    }
})
.addData("pitch", new Func<String>() {
    @Override public String value() {
        return formatAngle(robot.angles.angleUnit, robot.angles.thirdAngle);
    }
}

});

//-----
// Formatting
//-----
```

```
String formatAngle(AngleUnit angleUnit, double angle) {
    return formatDegrees(AngleUnit.DEGREES.fromUnit(angleUnit, angle));
}

String formatDegrees(double degrees){
    return String.format(Locale.getDefault(), "%.1f", AngleUnit.DEGREES.normalize(degrees));
}

private void initVufor()
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
/*
 * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
 */
VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

parameters.vuforiaLicenseKey = VUFORIA_KEY;
parameters.cameraDirection = BACK;

// Instantiate the Vuforia engine
vuforia = ClassFactory.getInstance().createVuforia(parameters);

}

/**
 * Initialize the Tensor Flow Object Detection engine.
 */
private void initTfod() {

    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());

    TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters
(tfodMonitorViewId);
    tfod = ClassFactory.getInstance().createTFObjectDetector(tfodParameters, vuforia);
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LA-
BEL_SILVER_MINERAL);
}
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Following is the Autonomous Program used for Meet 3:

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;
import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.tfod.Recognition;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

@Autonomous(name = "FF: AutoPosition", group = "Auto")
//@Disabled
public class AutoPosition extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
    TurnDef2 turn = new TurnDef2();

    LiftDef2 lift = new LiftDef2();
    DriveDistanceDef distance1 = new DriveDistanceDef();
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
boolean phone_locked;
boolean phone_in;
boolean phone_out;
int phonecyclecount;

private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";

/*
 * IMPORTANT: You need to obtain your own license key to use Vuforia. The string below with which
 * 'parameters.vuforiaLicenseKey' is initialized is for illustration only, and will not function.
 * A Vuforia 'Development' license key, can be obtained free of charge from the Vuforia developer
 * web site at https://developer.vuforia.com/license-manager.
 *
 * Vuforia license keys are always 380 characters long, and look as if they contain mostly
 * random data. As an example, here is a example of a fragment of a valid key:
 * ...
 * Once you've obtained a license key, copy the string from the Vuforia web site
 * and paste it in to your code onthe next line, between the double quotes.
 */

// Variables to be used for later

private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTz2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/  
MZZ7Zvp35rYD6lavfkgCMZ0DVAXhBv4J5LlrGlVXYfhhS1NklTGPdqVRW2aBmKLwctHzaztzycau3g//  
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/  
rEsoZMaPnFkYGcE-  
Zuz1DStPn6ZriIRE+FhMistaO3ntLvZdi3WBTbr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/  
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here
```

```
/**  
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
 * localization engine.  
 */  
private VuforiaLocalizer vuforia;  
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFOBJECTDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();  
  
private ElapsedTime runtime = new ElapsedTime();  
  
@Override public void runOpMode() throws InterruptedException {  
    String print_val;  
    print_val = "Test";  
    String target_print_val;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
target_print_val = "Target";
String turn_print_val;
turn_print_val = "Turn";
String hanging_print_val;
hanging_print_val = "Not Hanging";
/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);
// lift.init(hardwareMap, this);

distance1.init(hardwareMap, this);
drive.init(hardwareMap, this);
turn.init(hardwareMap, this);
// touch.init(hardwareMap, this);

// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Autonomous Position");

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.rightdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
robot.rightdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Send telemetry message to indicate successful Encoder reset
telemetry.addData("1", "Starting drive position at Left %7d - Right %7d Lefttwo %7d - Righttwo %7d",
    robot.leftdrive.getCurrentPosition(),
    robot.rightdrive.getCurrentPosition(),
    robot.lefttwo.getCurrentPosition(),
    robot.righttwo.getCurrentPosition());

robot.position_option = 0;

/* while (robot.position_side == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.position_side = 1;
    }else if (gamepad2.y) {
        robot.position_side = 2;
    }else {
        telemetry.addData("5","Select D2 - x for Red Side cnt - %d", robot.count);
        telemetry.addData("6","Select D2 - y for Blue Side");
        telemetry.update();
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
    robot.wait(1);
}
robot.wait(1);/*
robot.count = 0;
while (robot.position_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.position_option = 3;
        print_val = "Crater";
    } else if (gamepad2.b) {
        robot.position_option = 1;
        print_val = "Depot";
    } else {
        telemetry.addData("5", "Select D2 - a for position 1 crater cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - b for position 2 depot");
        telemetry.update();
        robot.wait(1);
    }
}
// turning right or left
/* robot.wait(1);
robot.count = 0;
while (robot.turn_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.turn_option = 1;
        turn_print_val = "Right";
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
 } else if (gamepad2.y) {
    robot.turn_option = 2;
    turn_print_val = "Left";
} else {
    telemetry.addData("5", "Select D2 - x for turning right cnt - %d", robot.count);
    telemetry.addData("6", "Select D2 - y for turning left");
    telemetry.update();
    robot.wait(1);
}
}/*
// going for the target mineral or not
/*  robot.wait(1);
robot.count = 0;
while (robot.find_target == -1 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.find_target = 1;
        target_print_val = "Find Target";
    } else if (gamepad2.b) {
        robot.find_target = 0;
        target_print_val = "Do Not Find Target";
    } else {
        telemetry.addData("5", "Select D2 - a for find mineral target cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - b for do not find mineral target");
        telemetry.update();
        robot.wait(1);
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

}*/

```
robot.find_target = 1;
target_print_val = "Find Target";
// robot hanging or not
/*  robot.wait(1);
robot.count = 0;
while (robot.hanging == -1 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.hanging = 1;
        hanging_print_val = "Hanging";
    } else if (gamepad2.y) {
        robot.hanging = 0;
        hanging_print_val = "Not Hanging";
    } else {
        telemetry.addData("5", "Select D2 - x for Hanging cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - y for NOT Hanging");
        telemetry.update();
        robot.wait(1);
    }
} */
telemetry.addData("3", "before setting up tfod");

initVuforia();

if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
telemetry.addData("3","setting up tfod");
telemetry.update();
initTfod();
telemetry.addData("3","back from setting up tfod");
telemetry.update();
robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}

// Send telemetry message to indicate successful Encoder reset
if (robot.position_side == 1) {
    telemetry.addData("1", "All setup at Position Red Side %s - %s - %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
} else {
    telemetry.addData("1", "All setup at Position Blue Side %s - %s- %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
}
telemetry.addData("1", "Position %d - %d - %d",
    robot.turn_option + robot.position_option,robot.turn_option, robot.position_option);
telemetry.addData("2","Waiting to start");
telemetry.update();

// wait for the start button to be pressed.
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
waitForStart();

int location = 0;

// init phone
robot.phone_position = .08;
robot.phone.setPosition(robot.phone_position);

robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

if (robot.hanging == 1) {
    // get down from the lander
    // lift.liftmove(1, .25);
}

robot.path_option = robot.turn_option + robot.position_option;
if (robot.find_target == 1) {
    // put phone servo in start position
    telemetry.addData("6", " moving phone");
    robot.init(hardwareMap, 2);
    //get the target value
    if (tfod != null) {
        tfod.activate();
    }
    phone_locked = true;
    findMineral();
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
telemetry.addData("7", "Location is %d", robot.gold_position);

telemetry.addData("6", "Option going to move the gold mineral");
if (robot.gold_position == 1){
    //turn to the left and move forward
    //turn left
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 2, 2.5);
    if (robot.position_option == 1 ){
        //on the depot side
        telemetry.addData("6", "Option going forward 48in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 37, 2.5);
        //turn right
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 1, 2.5);
        //forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.0);

    } else {
        //on the crater side
        telemetry.addData("6", "Option going forward 31in");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
    }
} else if (robot.gold_position == 2) {
    //going straight
    if (robot.position_option == 1 ){
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
// on the depot side
telemetry.addData("6", "Option going forward 48in");
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 52, 4.5);
} else {
    // on the crater side
    telemetry.addData("6", "Option going forward 28in");
    //forward towards the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 39, 5.5);
}
} else if (robot.gold_position == 3) {
    // turning right and then going straight
    //turn right
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 1, 3.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 35in ");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 35, 2.5);
        //turn left
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 2, 2.5);
        //forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.5);
    } else {
        // on the crater side
        telemetry.addData("6", "Option going forward 31in ");
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
//forward towards the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
}

}

telemetry.update();

robot.marker_power = .7;
robot.marker.setPosition(robot.marker_power);
robot.wait(2);
robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

} else if (robot.path_option == 2){
    telemetry.addData("6", "Depot Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
    //forward
    drive.encoder2Drive(robot.DRIVE_SPEED1, 32, 5.5);
    //turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);
    //forward
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
    //turn left
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 2, 3.5);
//forward
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
// dump off the marker

// backwards towards the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);

} else if (robot.path_option == 3) {
    telemetry.addData("6", " Depot Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15, 1.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
    //forward toward the depot
    drive.encoder2Drive(robot.DRIVE_SPEED, 12, 3.5);
    // dump off the marker
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
//forward into the crater
drive.encoder2Drive(robot.DRIVE_SPEED, 5, 3.5);
// turn right
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,1, 3.5);
//forward toward the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 96, 10.5);

} else if (robot.path_option == 4) {
    telemetry.addData("6", " Crater Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
    //forward going past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 29, 5.5);
    // turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15,2, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 26, 5.5);

}else if (robot.path_option == 5) {
    telemetry.addData("6", " Crater Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    // turn left
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
//forward and go to the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 60, 8.5);
//turn right
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
//dump off the marker

//backwards toward the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);
} else {
//nearest to the audience
//the distance is 23 15/16 but allow for role from stone to parking zone
telemetry.addData("6", " No Valid option picked");
telemetry.update();

}

if (tfod != null) {
tfod.shutdown();
}
robot.wait(20);
telemetry.addData("Path", "Autonomous Complete");
telemetry.update();

idle(); //Always call idle()
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

}

// find the position of the gold mineral

```
private void findMineral() {
    double mineral_location;
    double mineral_location_silver1;
    double mineral_location_silver2;
    phone_in = true;
    phone_locked = false;
    boolean just_switched = false;
    robot.phone_position = .06;
    String label = "outward";
    robot.phone.setPosition(robot.phone_position);

    telemetry.addData("2", "in findMineral");
    if (tfod != null) {
        // getUpdatedRecognitions() will return null if no new information is available since
        // the last time that call was made.
        int count = 0;
        int found = 0;

        mineral_location = -1;
        mineral_location_silver1 = -1;
        mineral_location_silver2 = -1;
        int goldMineralX = -1;
        int silverMineral1X = -1;
        int silverMineral2X = -1;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

// loop through to find the three targets

```
while ((goldMineralX == -1 && silverMineral1X == -1 && silverMineral2X == -1) || (!isStopRequested()
() && opModelsActive())) {

    List<Recognition> updatedRecognitions = tfod.getUpdatedRecognitions();

    // if data - check it
    if (updatedRecognitions != null) {
        // telemetry.addData("# Object Detected", updatedRecognitions.size());
        count++;
        // telemetry.addData("7", "track is %d", count);
        int item_count = 0;
        //phone_locked = false;
        if (updatedRecognitions.size() > 0) {

            for (Recognition recognition : updatedRecognitions) {
                label = recognition.getLabel();
                item_count++;
                telemetry.addData("7", "checking - %s found %d", label, found);
                telemetry.update();
                if (recognition.getLabel().equals(LABEL_GOLD_MINERAL) && goldMineralX == -1) {
                    found++;
                    telemetry.addData("7", "found gold");
                    telemetry.update();
                    goldMineralX = (int) recognition.getLeft();
                    mineral_location = robot.phone_position;
                }
            }
        }
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
robot.wait(1);
}
if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral1X == -1) {
    found++;
    telemetry.addData("7", "found silver");
    telemetry.update();
    silverMineral1X = (int) recognition.getLeft();
    mineral_location_silver1 = robot.phone_position;
    robot.wait(1);
} else if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral2X == -1
&&
mineral_location_silver1 != robot.phone_position) {
    found++;
    telemetry.addData("7", "found silver");
    telemetry.update();
    silverMineral2X = (int) recognition.getLeft();
    mineral_location_silver2 = robot.phone_position;
    robot.wait(1);
}
// telemetry.addData("4", "count %d - internal count %d", updatedRecognitions.size(),
item_count);
// telemetry.update();
if (item_count == updatedRecognitions.size()) break;
}
}
}
telemetry.addData("er", "should move phone %b %b %b", phone_in, phone_out, phone_locked);
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
telemetry.update();
// move the phone box to find the target
if (phone_in && !phone_locked) {
    //move phone (bottom is 1)
    if (robot.phone_position > .41) {
        phone_in = false;
        phone_out = true;
    } else {
        robot.phone_position += .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
} else if (phone_out && !phone_locked) {
    // phone (top position is 0)
    if (robot.phone_position < .19) {
        phone_in = true;
        phone_out = false;
    } else {
        robot.phone_position -= .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
}
if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 25000) {
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
    phone_locked = false;
    phonecyclecount = 0;
}

}

telemetry.addData("8", "1Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1,
mineral_location_silver2);
// telemetry.update();
//robot.wait(5);
idle(); // Always call idle()
// if all items found then break out
if (found == 3) break;
}//end of while

if (mineral_location != -1) {
    if (mineral_location == .18) {
        telemetry.addData("Gold Mineral Position", "Left");
        robot.gold_position = 1;
    } else if (mineral_location == .42) {
        telemetry.addData("Gold Mineral Position", "Right");
        robot.gold_position = 3;
    } else if (mineral_location == .3){
        telemetry.addData("Gold Mineral Position", "Center");
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
    robot.gold_position = 2;
}
}

// telemetry.addData("8", "2Phone in %b Phone out %b pos %.3f", phone_in, phone_out, robot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d", found,goldMineralX,silverMineral1X,silverMineral2X);
telemetry.addData("8", "mineral location %f %f %f",mineral_location, mineral_location_silver1, mineral_location_silver2);
telemetry.update();
}

// robot.wait(20);
}

public void waitForTick(long periodMs) {

    long remaining = periodMs - (long)period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
        try {
            Thread.sleep(remaining);
        } catch (InterruptedException e) {
            Thread.currentThread().interrupt();
        }
    }
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
// Reset the cycle clock for the next pass.  
    period.reset();  
}  
  
private void initVufor() {  
    /*  
     * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.  
     */  
    VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();  
  
    parameters.vuforiaLicenseKey = VUFORIA_KEY;  
    parameters.cameraDirection = VuforiaLocalizer.CameraDirection.BACK;  
  
    // Instantiate the Vuforia engine  
    vuforia = ClassFactory.getInstance().createVuforia(parameters);  
  
}  
/**  
 * Initialize the Tensor Flow Object Detection engine.  
 */  
private void initTfod() {  
  
    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(  
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());  
  
    TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
(tfodMonitorViewId);
    tfod = ClassFactory.getInstance().createTFOBJECTDetector(tfodParameters, vuforia);
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LA-
BEL_SILVER_MINERAL);
}

}
```

Following is the Robot Hardware Definition File:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.hardware.bosch.BNO055IMU;
import com.qualcomm.hardware.bosch.JustLoggingAccelerationIntegrator;
import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cRangeSensor;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.CRServo;
import com.qualcomm.robotcore.hardware.ColorSensor;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.hardware.DigitalChannel;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.hardware.I2cAddr;
import com.qualcomm.robotcore.hardware.Servo;
import com.qualcomm.robotcore.hardware.TouchSensor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.Func;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;

import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotorSimple.Direction.REVERSE;
import static java.lang.Thread.currentThread;
import static java.lang.Thread.sleep;

/**
 * This is NOT an opmode.
 * <p>
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
 * <p>
 * This hardware class assumes the following device names have been configured on the robot:
 * Note: All names are lower case and some have single spaces between words.
 */
public class PrinceCharlesBaDazzle {

    // setup for calculation of the how far to move
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
static final double P_TURN_COEFF      = 0.1; // Larger is more responsive, but also less stable
static final double HEADING_THRESHOLD = 1;   // As tight as we can make it with an integer
gyro

static final double COUNTS_PER_MOTOR_REV = 1680; // eg: AndyMark Motor Encoder
static final double DRIVE_GEAR_REDUCTION = 1; // This is < 1.0 if geared UP
static final double WHEEL_DIAMETER_INCHES = 4.0; // For figuring circumference
static final double GEAR_DIAMETER_INCHES = 1.0; // For figuring circumference

static final double COUNTS_PER_INCH     = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
(WHEEL_DIAMETER_INCHES * 3.1415);
static final double LIFT_COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
(GEAR_DIAMETER_INCHES * 3.1415);

static final double DRIVE_SPEED       = 0.5;
static final double DRIVE_SPEED1      = 0.9;
static final double STONE_DRIVE_SPEED = 0.2;
static final double TURN_SPEED        = 0.3 ;
static final double TOUCH_SPEED       = 0.1;
static final double LIFT_SPEED         = 0.2;
static final double STOP_SPEED        = 0;

int floor_color_value;
int position_option = 0;
int position_side = 0;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
int turn_option = 0;
int path_option = 0;
int find_target = -1;
int hanging = -1;
int gold_position = 0;
int count = 0;

// Driver motors
DcMotor rightdrive;
DcMotor leftdrive;
DcMotor righttwo;
DcMotor lefttwo;
float right = 0;
float left = 0;

// Distance control

ModernRoboticsI2cRangeSensor distance;

//motors to lift and lower
DcMotor right_liftdrive;
DcMotor left_liftdrive;
float lift_power = 0;

// move the collector in/out up/down
DcMotor extenderdrive;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
// pick up the minerals
DcMotor collectordrive;
float extender_power = 0;
float collector_power = 0;

// open position
double right_position = 0;
double left_position = .5;

// Side phone
Servo phone;
double phone_position = 0.26;
CRServo left_wrench;
CRServo right_wrench;
double wrench_power = 0;

Servo lock;
double lock_power = 0;
Servo marker;
double marker_power = 0;
Servo right_arm;
double right_arm_power = 0.5;
Servo left_arm;
double left_arm_power = 0.5;

DigitalChannel lower_stop;
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
DigitalChannel upper_stop;
DigitalChannel extender_lower;
DigitalChannel extender_upper;
DigitalChannel wrench_lower;
DigitalChannel wrench_upper;
// Our sensors, motors, and other devices go here, along with other long term state
BNO055IMU imu;

// State used for updating telemetry
Orientation angles;
Acceleration gravity;

/* local OpMode members. */
HardwareMap hardwareMap = null;
private ElapsedTime period = new ElapsedTime();

// Private Members
private LinearOpMode myOpMode;

/* Constructor */
public PrinceCharlesBaDazzle() {

}

/* Initialize standard Hardware interfaces */
public void init(HardwareMap ahwMap, int option) {
    // Save reference to Hardware map
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
hardwareMap = ahwMap;

/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */
// if option is one define hardware... otherwise put it start position..
if (option == 1) {
    // start of drive train definitions
    rightdrive = hardwareMap.dcMotor.get("right_drive");
    leftdrive = hardwareMap.dcMotor.get("left_drive");
    leftdrive.setDirection(REVERSE);
    righttwo = hardwareMap.dcMotor.get("righttwo");
    lefttwo = hardwareMap.dcMotor.get("lefttwo");
    lefttwo.setDirection(REVERSE);
    // end of drive train definitions

    right_liftdrive = hardwareMap.dcMotor.get("right_lift");
    right_liftdrive.setDirection(REVERSE);
    left_liftdrive = hardwareMap.dcMotor.get("left_lift");
    right_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);
    left_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);

    collectordrive = hardwareMap.dcMotor.get("collector");
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
extenderdrive = hardwareMap.dcMotor.get("extender");
extenderdrive.setDirection(REVERSE);
// stop the lifter when down flat
lower_stop = hardwareMap.get(DigitalChannel.class, "lower_stop");
upper_stop = hardwareMap.get(DigitalChannel.class, "upper_stop");
// Center piece
wrench_lower = hardwareMap.get(DigitalChannel.class, "wrench_lower");
wrench_upper = hardwareMap.get(DigitalChannel.class, "wrench_upper");
// outer piece
extender_lower = hardwareMap.get(DigitalChannel.class, "extender_lower");
extender_upper = hardwareMap.get(DigitalChannel.class, "extender_upper");
// set the digital channel to input.
lower_stop.setMode(DigitalChannel.Mode.INPUT);
upper_stop.setMode(DigitalChannel.Mode.INPUT);
wrench_lower.setMode(DigitalChannel.Mode.INPUT);
wrench_upper.setMode(DigitalChannel.Mode.INPUT);
extender_lower.setMode(DigitalChannel.Mode.INPUT);
extender_upper.setMode(DigitalChannel.Mode.INPUT);

phone = hardwareMap.servo.get("phone");
left_wrench = hardwareMap.get(CRServo.class, "left_wrench");
right_wrench = hardwareMap.get(CRServo.class, "right_wrench");

lock = hardwareMap.servo.get("lock");
marker = hardwareMap.servo.get("marker");
right_arm = hardwareMap.servo.get("right_arm");
left_arm = hardwareMap.servo.get("left_arm");
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
// set up the guidance hardware

// Set up the parameters with which we will use our IMU. Note that integration
// algorithm here just reports accelerations to the logcat log; it doesn't actually
// provide positional information.
BNO055IMU.Parameters parameters = new BNO055IMU.Parameters();
parameters.angleUnit      = BNO055IMU.AngleUnit.DEGREES;
parameters.accelUnit       = BNO055IMU.AccelUnit.METERS_PERSEC_PERSEC;
parameters.calibrationDataFile = "BNO055IMUCalibration.json"; // see the calibration sample
opmode
parameters.loggingEnabled   = true;
parameters.loggingTag        = "IMU";
parameters.accelerationIntegrationAlgorithm = new JustLoggingAccelerationIntegrator();

// Retrieve and initialize the IMU. We expect the IMU to be attached to an I2C port
// on a Core Device Interface Module, configured to be a sensor of type "AdaFruit IMU",
// and named "imu".
imu = hardwareMap.get(BNO055IMU.class, "imu");
imu.initialize(parameters);

} else {

    // set the phones
    phone_position = .26;
    phone.setPosition(phone_position);

}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
}

public void wait(int sec) {

    for (int i = 0; i < 2 * sec; i++) {
        try {
            sleep(500);
        } catch (InterruptedException e) {
            currentThread().interrupt();
            break;
        }
    }
}

/**
 *
 * * waitForTick implements a periodic delay. However, this acts like a metronome with a regular
 * periodic tick. This is used to compensate for varying processing times for each cycle.
 * The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
 *
 * @param periodMs Length of wait cycle in mSec.
 */
public void waitForTick(long periodMs) {

    long remaining = periodMs - (long) period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
}
```

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

```
if (remaining > 0) {  
    try {  
        sleep(remaining);  
    } catch (InterruptedException e) {  
        currentThread().interrupt();  
    }  
}  
  
// Reset the cycle clock for the next pass.  
period.reset();  
}  
}
```

Our morning started out early as our coach was asked to bring some flags. We made the flags after eating breakfast and before our 1hr travel to Sanford.



Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

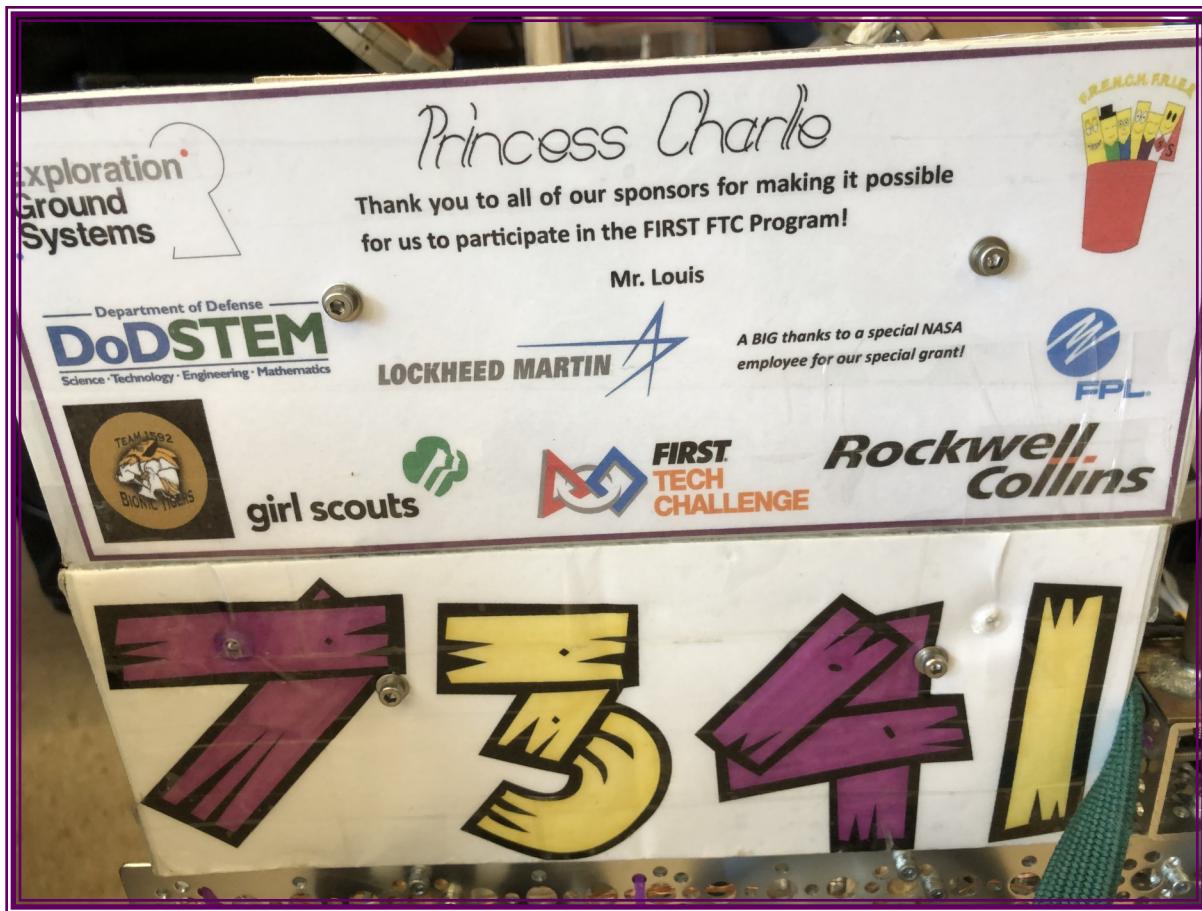
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

We finally finished our Sponsor banner for our robot.... Great group of sponsor to have:



Our day went fairly well. We ended up not getting to play our first match because the robot phone and the robot hardware was not talking to each other. We had to power down and then back up and things worked much better for the rest of the day.

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

On of our matches our alliance team ran into us during the Autonomous part and our marker holder was damaged. Quick thinking team member came up with the idea that we should tape the side back on, so the tape was obtain and the bandage was applied.



Shelby was in an artistic mood Friday night during the sleepover.....

Do we now have a Minion in our mists.....

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

We are currently ranked 11 in the win-lose ranking and ranked 5 in the RP points, so we feel that we had a very lucky meet!
Thank you to all our alliance teams.

Team Number	Team Name	QP	RP	Ranked Matches	Matches
4227	Metal Morphosis	20	1129	10	17
7592	Roarbots	20	1122	10	17
6323	The Pink Team	20	1087	10	17
7477	Super 7	20	1036	10	17
12090	STEM-Punk	20	966	10	17
14534	Robo Pegz	20	710	10	17
15401	CyberNoles	20	462	10	17
9013	Merritt Island Matrix	18	693	10	17
8945	Touch-Down!	17	802	10	17
4717	Mechromancers	16	728	10	17
7341	F.R.E.N.C.H. F.R.I.E.S.	12	985	10	17
14976	PAC RATS	12	949	10	17
14856	Wandering Wildcats	11	976	10	17
12245	Cobalt	10	863	10	11
14538	VCS Panthers Robotics - Crimson Blaze	10	705	10	12
14989	Florida Prep Academy	9	688	10	11
9671	Tomahawk Technauts	8	838	10	11
14215	Plaid Piranhas	8	760	10	11
14765	Mechatronic Mustangs	8	685	10	11
11499	Plaid Pelicans	8	545	10	11
8392	Lift-Off!	7	777	10	17
15179	LMP Griffins	4	418	6	6
14673	Odyssey One	4	335	5	5
15100	SL-Sea Lions	2	421	6	6
8617	Techno Tigers	2	295	6	6

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Following is the match score of the third meet. We are proud that we played 6 matches .

Match Num-	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penal-	Blue Auto	Blue TeleOp	Blue Penal-
1	4717 14538	7592 8392	107	166	40	67	0	60	106	0
2	7477 15401	14765 8617	130	15	55	75	0	0	15	0
3	9013 14976	14534 6323	152	310	65	87	0	120	190	0
4	4227 9671	15100 14856	166	45	30	136	0	20	25	0
5	15179 8945	12245 12090	74	194	30	44	0	80	114	0
6	14538 14534	7341 15401	137	23	70	67	0	0	23	0
7	4717 7477	12090 14856	242	109	100	142	0	55	54	0
8	7592 9013	9671 12245	249	50	125	124	0	0	10	40
9	7341 6323	15100 8945	236	101	105	131	0	30	71	0

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
10	14765 4227	14976 15179	87	77	45	42	0	15	62	0
11	12245 8392	8617 7341	73	63	25	48	0	40	23	0
12	7477 15100	14538 9013	151	146	45	106	0	65	81	0
13	14856 15179	15401 6323	20	193	0	20	0	65	128	0
14	12090 7592	4227 14534	302	220	150	152	0	120	100	0
15	8617 14976	9671 4717	67	66	0	67	0	0	56	10
16	15100 14765	8392 8945	44	75	0	44	0	50	25	0
17	7341 9013	15401 4227	109	110	60	49	0	80	30	0
18	14856 8617	14538 15179	50	59	10	40	0	25	34	0

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Match Number	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penalty	Blue Auto	Blue TeleOp	Blue Penalty
19	14534 12245	4717 14765	121	85	30	91	0	30	55	0
20	8945 7477	14976 7592	252	205	100	152	0	80	125	0
21	6323 9671	8392 12090	228	181	95	133	0	75	106	0
22	9013 15100	15179 4717	95	103	40	55	0	15	88	0
23	8945 14538	14534 9671	113	165	30	83	0	45	110	10
24	6323 12245	7477 4227	210	290	65	145	0	120	170	0
25	12090 15401	8392 14976	197	65	120	77	0	40	25	0
26	7592 7341	14765 14856	129	62	30	99	0	25	37	0
27	8617 4717	4227 8945	52	149	0	52	0	60	89	0

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 11-12, 2019

Process:

Match Num-	Red	Blue	Red Score	Blue Score	Red Auto	Red TeleOp	Red Penal-	Blue Auto	Blue TeleOp	Blue Penal-
28	6323 12090	14538 14765	245	80	135	110	0	40	40	0
29	14856 8392	9013 14534	29	168	0	29	0	75	93	0
30	15401 8617	7592 15100	59	166	30	29	0	80	86	0
31	15179 9671	7341 7477	102	153	15	87	0	40	113	0
32	12245 15100	14976 14538	77	98	10	67	0	45	53	0



We made it safe to the third meet. The team had put our all the goodie bags and the table was ready for the day's event.

Signature : Jessica Anderson

Date: Jan. 12, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

Date: January 13, 2019

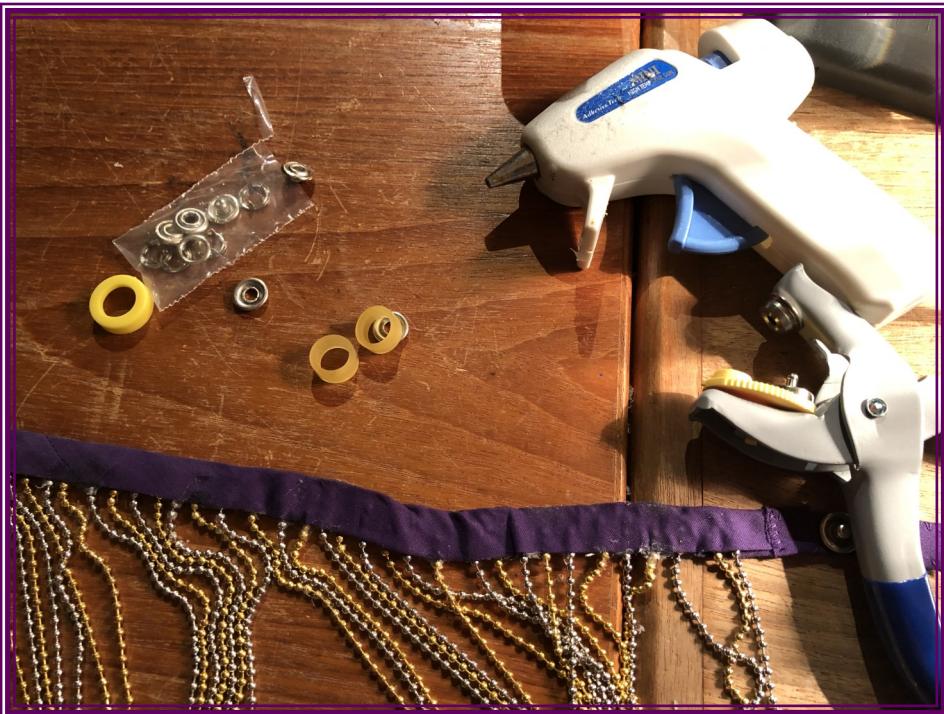
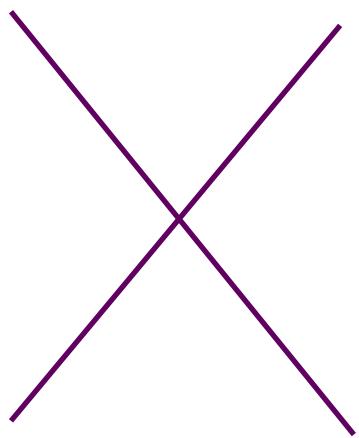
Purpose of the Activity:

- | | |
|--|---|
| Creating a mineral location indicator | Getting ready for a few outreach events |
| Updating the software to have a special mode where all wheels are controlled by just one stick | Update to the hooking mechanism |

Process:

When we first went out we found some rick-rack to use as a mineral location indicator for the prototype. Later that week our coach found some small chain and made us a second mineral locator with the help of our Sister FIRST LEGO League team C.A.K.E. B.A.T.T.E.R.S. These girls will be joining our team next year.

The girls learned how to use different types of rules, wire cutter pliers and how hot glue the chain to the fabric. We thank our future members for their help.



to

signature : Shelby Greer

Date: Jan. 13, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

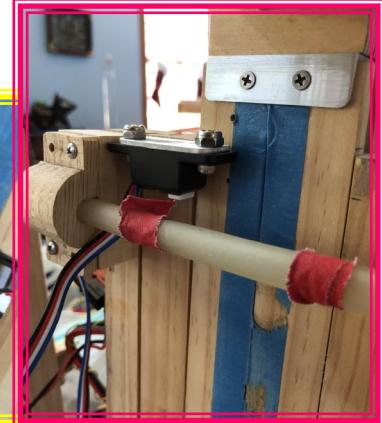
Engineering Activity Continued

Date: January 13, 2019

Process:



We performed a dry run test and determined that the new hook would work to hang at the end of the match if the could drive up quick enough to complete the task. We also changed out our magnetic sensor for two touch sensors so that we do not over extend the center portion of our lift. The magnetic sensors were not reliable enough.



Signature : Shelby Greer

Date: Jan. 13, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity

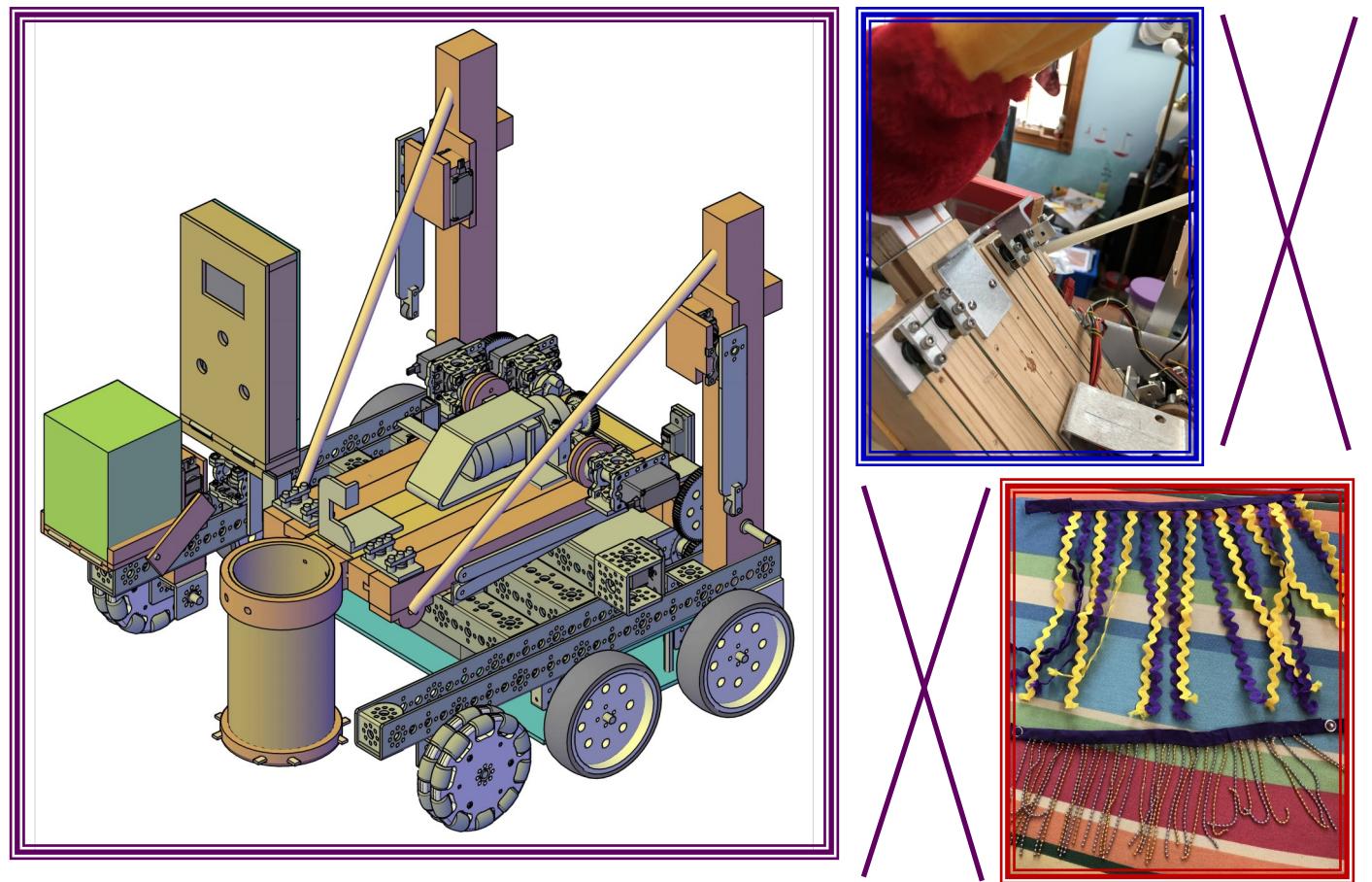
Date: January 27, 2019

Purpose of the Activity:

- | | |
|--|--|
| Add the finishing touches to the Engineering Note-
book | Complete the Control Award Document
Promote Video completed |
| Prepare for the League Championship | New capes |

Process:

We have changed just a few things for our championship meet. The latching mechanism is simpler and we created two different type of mineral guidance items we will see which works best.



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

Our coach was able to post our Promote video and we thank the C.A.K.E. B.A.T.T.E.R.S. team in there support in making the video. We also verified the usage of the mineral locator will be helpful in guiding the end effector to the correct spot.



We also have a surprise element of our team costume, Thanks to our coach for finishing the capes for the Championship. We are grateful to our coach for helping us get the cape completed and that they include a pocket to hide our items.



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

Control Award

Autonomous Objectives—We have one Autonomous program which will let you score on either the crater or depot side of the lander. At the beginning of the autonomous program the coach is asked what side of the lander you are using. Then based on that information the robot know which logic to use. The robot will scan the area when the phone is move to three different locations to locate the gold mineral. If you are on the depot side of the lander and once the gold mineral is located the scanning stops and the robot moves toward the gold mineral pushing it into the depot. Once in the depot the robot dumps the marker into the depot. If the gold mineral is either on the left or right side the robot will move towards the crater, stopping at the crater. If you are on the crater side of the lander and once the gold mineral is located the scanning stops and the robot moves toward the gold mineral pushing it towards the crater. We prefer to start on the depot side for a score of 42 points.

Sensors - The encoders on the motor/servo are used to calculate the distance traveled for either the movement of the phone or the movement of the robot.

Key Algorithms The use of the encoders on both the drive motor and servo help the robot know the distance travel. Using the servo's position feedback give the phone has found the gold mineral target we know the position of the gold mineral and what path is needed to be taked inorder to get the gold mineral into the depot.

```
if (mineral_location == .18) {  
    telemetry.addData("Gold Mineral Position", "Left");  
} else if (mineral_location == .42) {  
    telemetry.addData("Gold Mineral Position", "Right");  
} else if (mineral_location == .3){  
    telemetry.addData("Gold Mineral Position", "Center");  
}
```

Also given that we know the position we can then easily move toward the depot. The following calcu-

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

lation is used to calculate the movement of the motor to travel a given distance. You will use the following information in the calculation:

1. Diameter of the wheel— 4in * pie
2. Output counts per revolution of Output Shaft (cpr)— 1680 Pulses
3. Drive gear reduction— 1, because there are no gears being used

$$\text{COUNTS_PER_INCH} = (\text{COUNTS_PER_MOTOR_REV} * \text{DRIVE_GEAR_REDUCTION}) / (\text{WHEEL_DIAMETER_INCHES} * 3.1415);$$

This value is used with the distance needed to traveled and sent to the motor's encoder for the robot to travel to the correct position.

Driver Controlled Enhancements We use the touch sensor to insure that we do not extend the motors to damage the motor and or gears of lifting elements of the robot. We have additional logic if one of the A button on the drivers control the robot will drive (forwards and backwards) using the right control stick providing power to all wheel at the same time.

Engineering Notebook References Judges also use the Teams Engineering Notebook to evaluate details of the Control elements. To help guide this effort, Teams should provide pointers to where in the Engineering Notebook control related information is located. Some things to consider including as pointers are: Team goals for control activities, strategies for autonomous mode, Robot performance with and without added sensors, requirements for successful autonomous operation, performance improvements using algorithms and sensors, and testing results.

Autonomous Program Diagrams The diagram on the next page depicts the path of the robot for each position of the gold mineral. The paths are the same for each the blue or red side of the lander. The difference is weather you are on the crater or depot side of the lander. Each path is set so that the silver mineral are not disturbed.

Signature : Jessica Anderson

Date: Jan. 27, 2019

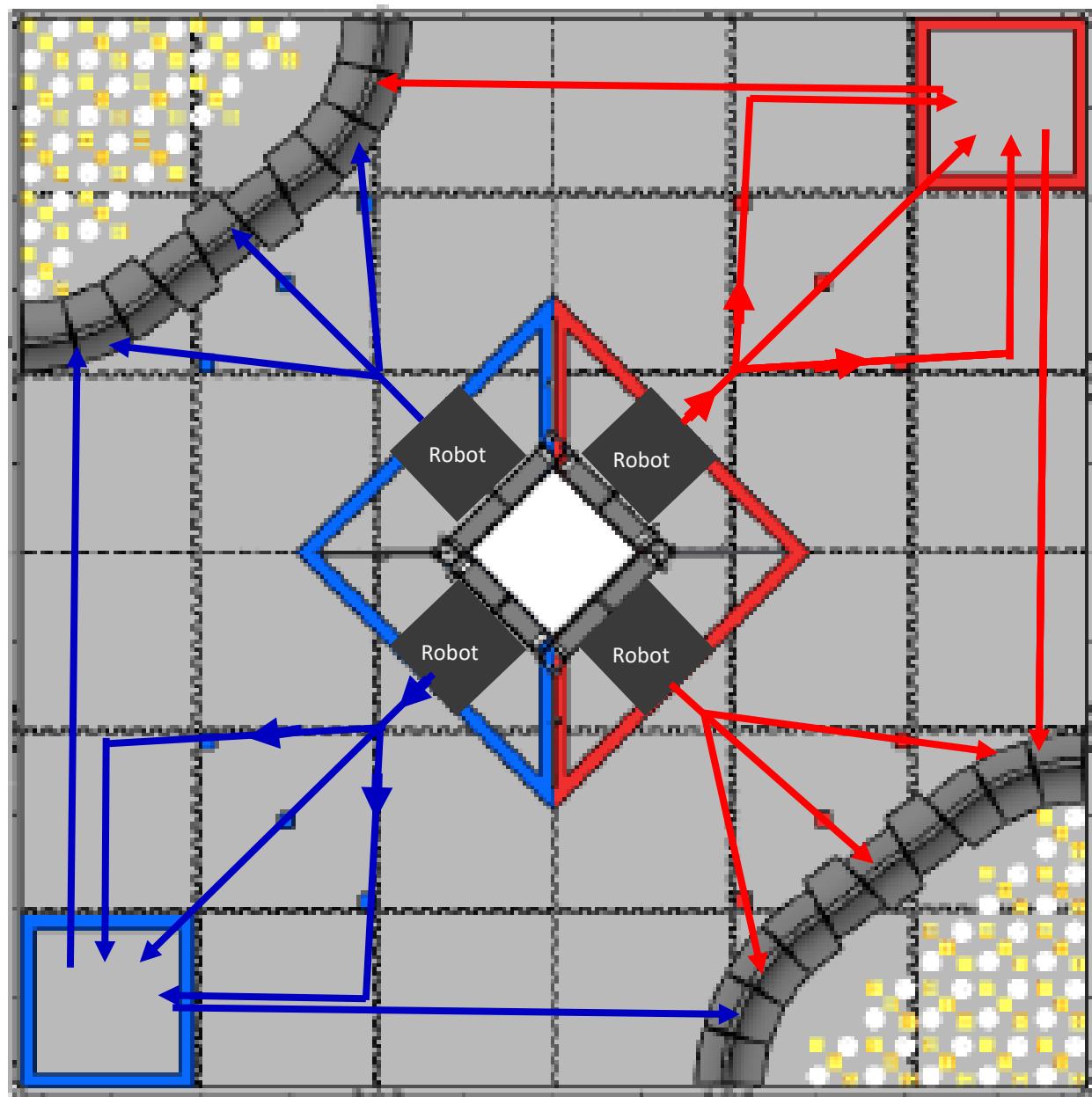
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

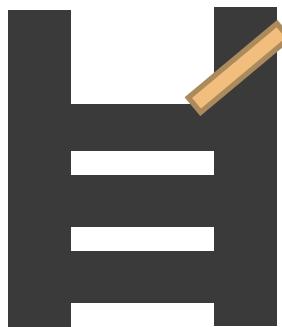
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: _____

Process:

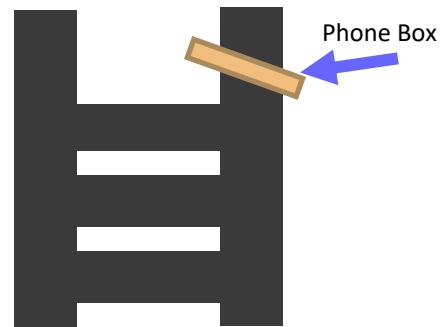
Additional Summary Information (optional) The method used to determine which position the gold mineral is the position of the phone as it can see just one mineral at a time if place properly.



Position 1

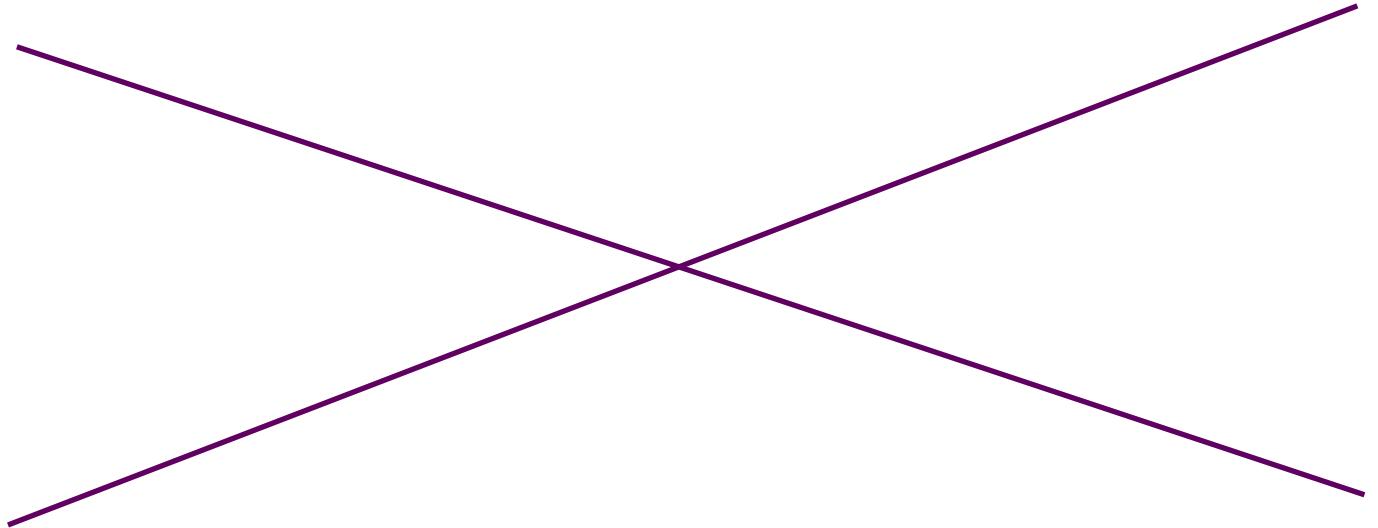


Position 2



Position 3

You can see the autonomous program 4 pages past this page.



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

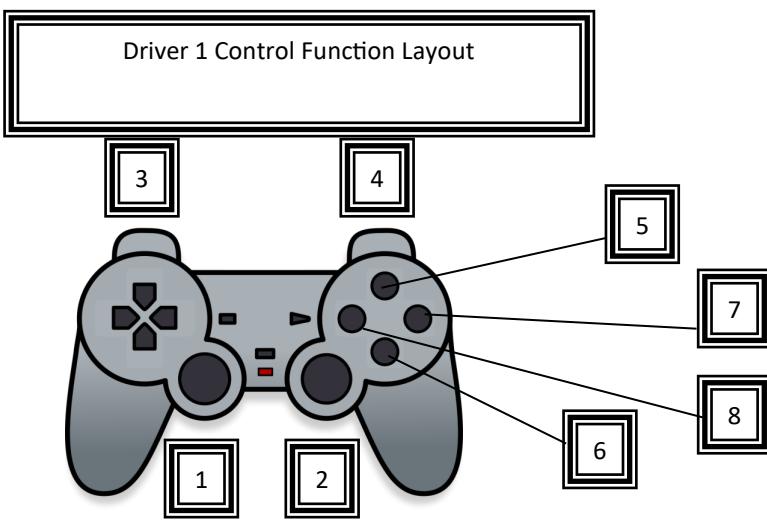
F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

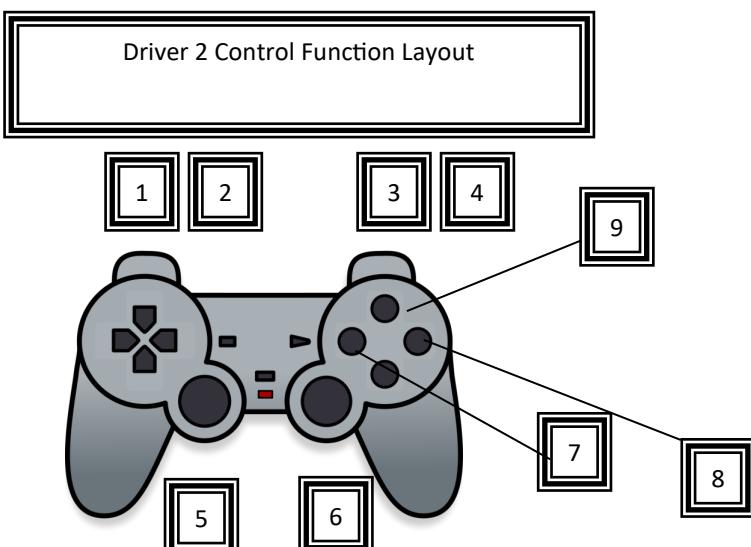
Process:

Described below is the functional layout for the robot controller buttons.



Driver 1 Control Function Definition

1. Left Wheels control
2. Right Wheels control
3. Move the phone left
4. Move the phone right
5. Normal Driving
6. End Game Driving
7. Arms Up
8. Arms down



Driver 2 Control Function Definition

1. Lock the second layer of the slide
2. Unlock the second layer of the slide
3. Raise the second layer of the slide
4. Lower the second layer of the slide
5. Lift and lower grabber
6. Extend and retract grabber
7. Make Grabber go backwards
8. Make Grabber pick up items
9. Stop Grabber

Signature : Jessica Anderson

Date: Jan. 27, 2019

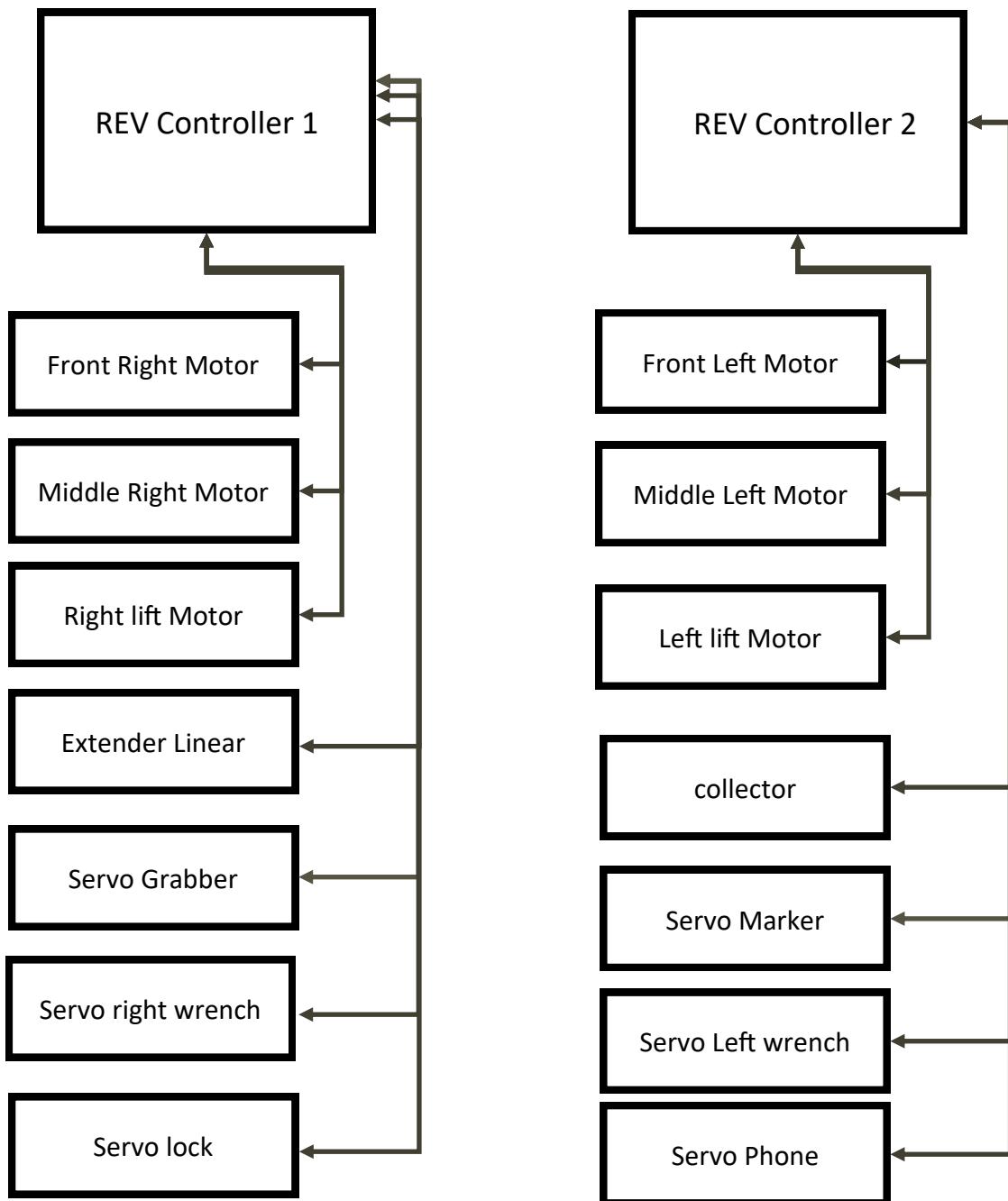
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:



Signature : Jessica Anderson

Date: Jan. 27, 2019

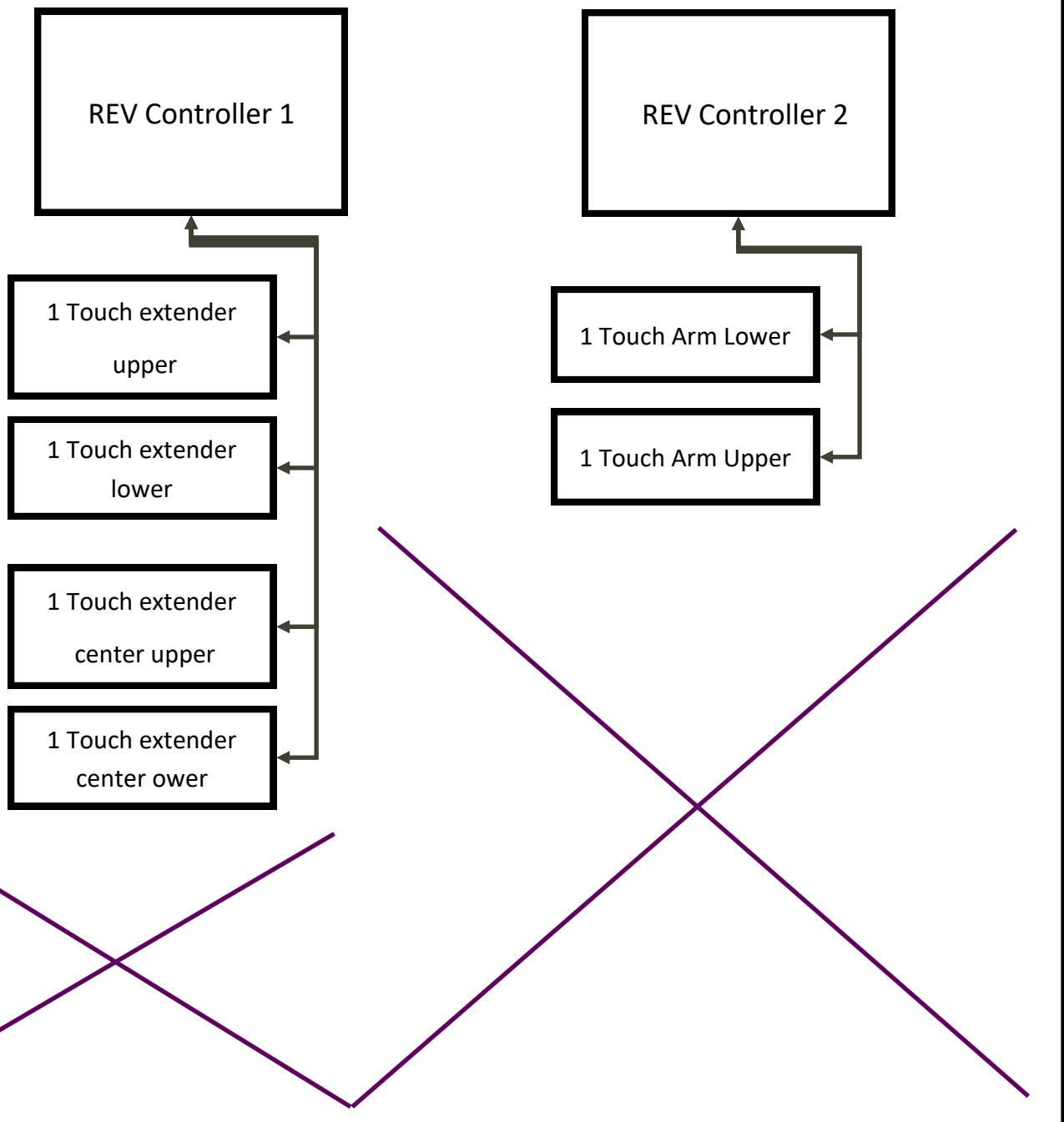
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

We have updated our Autonomous Program to travel backwards if the gold mineral is on the left

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

*The function of this program is to run autonomously to put a ball into the vortex
push the big ball off the base and then and go on the base.*

This will work on only the blue side.

*Neither the name of Qualcomm Technologies Inc nor the names of its contributors
may be used to endorse or promote products derived from this software without
specific prior written permission.*

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS
LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
import com.qualcomm.robotcore.eventloop.opmode.Autonomous;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.DcMotor;
import com.qualcomm.robotcore.util.ElapsedTime;
import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.tfod.Recognition;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;

{@Autonomous(name = "FF: AutoPosition", group = "Auto")
//@Disabled
public class AutoPosition extends LinearOpMode {
    private ElapsedTime period = new ElapsedTime();

    PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle(); // Use a Princess's Charlie hardware

    // Define your functions
    DriveDef2 drive = new DriveDef2();
    TurnDef2 turn = new TurnDef2();

    LiftDef2 lift = new LiftDef2();
    DriveDistanceDef distance1 = new DriveDistanceDef();
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
boolean phone_locked;
boolean phone_in;
boolean phone_out;
int phonecyclecount;
int goto_crater = 0;

private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";

/*
 * IMPORTANT: You need to obtain your own license key to use Vuforia. The string below with which
 * 'parameters.vuforiaLicenseKey' is initialized is for illustration only, and will not function.
 * A Vuforia 'Development' license key, can be obtained free of charge from the Vuforia developer
 * web site at https://developer.vuforia.com/license-manager.
 *
 * Vuforia license keys are always 380 characters long, and look as if they contain mostly
 * random data. As an example, here is a example of a fragment of a valid key:
 * ... ylgIzTqZ4mWjk9wd3cZO9T1axEqzuhxoGlfOOI2dRzKS4T0hQ8kT ...
 * Once you've obtained a license key, copy the string from the Vuforia web site
 * and paste it in to your code onthe next line, between the double quotes.
 */

// Variables to be used for later

private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTi2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1O-
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

Num6rS90mBDJwrv1CJM~~c~~5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/
M~~z~~7Zvp35rYD6lavfkgCMZ0DV~~A~~xHBv4J5LlrGIVXYfhhS1NkITGPDqVRW2aBmKLwctHZaztzycau3g//
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnFkYGcE-
Zuz1DStPn6ZriRE+FhMistaO3ntLvZdi3WB~~T~~br8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here

```
/**  
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia  
 * localization engine.  
 */  
private VuforiaLocalizer vuforia;  
/**  
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object  
 * Detection engine.  
 */  
private TFOBJECTDetector tfod;  
  
// Leave argument list empty if you want to disable the camera monitor view.  
TFOBJECTDetector.Parameters tfodParameters = new TFOBJECTDetector.Parameters();  
  
private ElapsedTime runtime = new ElapsedTime();  
  
@Override public void runOpMode() throws InterruptedException {  
    String print_val;  
    print_val = "Test";  
    String target_print_val;  
    target_print_val = "Target";
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
String turn_print_val;
turn_print_val = "Turn";
String hanging_print_val;
hanging_print_val = "Not Hanging";
/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);
// lift.init(hardwareMap, this);

distance1.init(hardwareMap, this);
drive.init(hardwareMap, this);
turn.init(hardwareMap, this);
// touch.init(hardwareMap, this);

// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Autonomous Position");

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

robot.leftdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.leftdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righthdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righthdrive.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.lefttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
robot.lefttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);
robot.righttwo.setMode(DcMotor.RunMode.RUN_USING_ENCODER);

// Send telemetry message to indicate successful Encoder reset
telemetry.addData("1", "Starting drive position at Left %7d - Right %7d Lefttwo %7d - Righttwo %7d",
    robot.leftdrive.getCurrentPosition(),
    robot.rightdrive.getCurrentPosition(),
    robot.lefttwo.getCurrentPosition(),
    robot.righttwo.getCurrentPosition());

robot.position_option = 0;

/*
  while (robot.position_side == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
      robot.position_side = 1;
    }else if (gamepad2.y) {
      robot.position_side = 2;
    }else {
      telemetry.addData("5","Select D2 - x for Red Side cnt - %d", robot.count);
      telemetry.addData("6","Select D2 - y for Blue Side");
      telemetry.update();
    }
    robot.wait(1);
  }
  robot.wait(1);*/
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
robot.count = 0;
while (robot.position_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.a) {
        robot.position_option = 3;
        print_val = "Crater";
    } else if (gamepad2.b) {
        robot.position_option = 1;
        print_val = "Depot";
    } else {
        telemetry.addData("5", "Select D2 - a for position 1 crater cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - b for position 2 depot");
        telemetry.update();
        robot.wait(1);
    }
}
// turning right or left
/* robot.wait(1);
robot.count = 0;
while (robot.turn_option == 0 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.turn_option = 1;
        turn_print_val = "Right";
    } else if (gamepad2.y) {
        robot.turn_option = 2;
        turn_print_val = "Left";
    } else {
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
telemetry.addData("5", "Select D2 - x for turning right cnt - %d", robot.count);
telemetry.addData("6", "Select D2 - y for turning left");
telemetry.update();
robot.wait(1);

}

}*/



robot.find_target = 1;
target_print_val = "Find Target";
// robot hanging or not
/*  robot.wait(1);
robot.count = 0;
while (robot.hanging == -1 && (!isStopRequested() || opModelsActive())) {
    robot.count++;
    if (gamepad2.x) {
        robot.hanging = 1;
        hanging_print_val = "Hanging";
    } else if (gamepad2.y) {
        robot.hanging = 0;
        hanging_print_val = "Not Hanging";
    } else {
        telemetry.addData("5", "Select D2 - x for Hanging cnt - %d", robot.count);
        telemetry.addData("6", "Select D2 - y for NOT Hanging");
        telemetry.update();
        robot.wait(1);
    }
} */
telemetry.addData("3","before setting up tfod");
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
initVufor();

if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {
    telemetry.addData("3", "setting up tfod");
    telemetry.update();
    initTfod();
    telemetry.addData("3", "back from setting up tfod");
    telemetry.update();
    robot.wait(5);

} else {
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");
}

// Send telemetry message to indicate successful Encoder reset
if (robot.position_side == 1) {
    telemetry.addData("1", "All setup at Position Red Side %s - %s - %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
} else {
    telemetry.addData("1", "All setup at Position Blue Side %s - %s - %s - %s",
        print_val, turn_print_val, target_print_val, hanging_print_val);
}
telemetry.addData("1", "Position %d - %d - %d",
    robot.turn_option + robot.position_option, robot.turn_option, robot.position_option);
telemetry.addData("2", "Waiting to start");
telemetry.update();
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// wait for the start button to be pressed.  
  
waitForStart();  
  
int location = 0;  
  
// init phone  
robot.phone_position = .08;  
robot.phone.setPosition(robot.phone_position);  
  
robot.marker_power = .4;  
robot.marker.setPosition(robot.marker_power);  
  
if (robot.hanging == 1) {  
    // get down from the lander  
    // lift.liftmove(1, .25);  
}  
  
robot.path_option = robot.turn_option + robot.position_option;  
if (robot.find_target == 1) {  
    // put phone servo in start position  
    telemetry.addData("6", " moving phone");  
    robot.init(hardwareMap, 2);  
    //get the target value  
    if (tfod != null) {  
        tfod.activate();  
    }  
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
phone_locked = true;
findMineral();
telemetry.addData("7", "Location is %d", robot.gold_position);

telemetry.addData("6", "Option going to move the gold mineral");
if (robot.gold_position == 1){
    // turn to the left and move forward
    // turn left
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 2, 2.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option Depot going forward Gold on the left");
        // forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 37, 2.5);
        // turn right
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 1, 2.5);
        // forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.0);
        goto_crater = 1;
    } else {
        // on the crater side
        telemetry.addData("6", "Option going forward 31in to the crater");
        // forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
    }
} else if (robot.gold_position == 2) {
    // going straight
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
if (robot.position_option == 1 ){
    // on the depot side
    telemetry.addData("6", "Option going forward 52in into the depot");
    //forward towards the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 52, 4.5);
} else {
    // on the crater side
    telemetry.addData("6", "Option going forward 39in to the crater");
    //forward towards the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 39, 5.5);
}
} else if (robot.gold_position == 3) {
    // turning right and then going straight
    // turn right
    drive.encoder2Drive(robot.DRIVE_SPEED1, 4, 2.5);
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 5, 1, 3.5);
    if (robot.position_option == 1 ){
        // on the depot side
        telemetry.addData("6", "Option going forward 35in into the depot ");
        //forward towards the wall
        drive.encoder2Drive(robot.DRIVE_SPEED1, 35, 2.5);
        // turn left
        turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 11, 2, 2.5);
        //forward towards the depot
        drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 3.5);
    } else {
        // on the crater side
        telemetry.addData("6", "Option going forward 31in to the crater");
    }
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
//forward towards the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 31, 5.5);
}

}

telemetry.update();

robot.marker_power = .7;
robot.marker.setPosition(robot.marker_power);
robot.wait(2);
robot.marker_power = .4;
robot.marker.setPosition(robot.marker_power);

if (goto_crater == 1) drive.encoder2Drive(-robot.DRIVE_SPEED1, 7*12, 5.5);

} else if (robot.path_option == 2){
    telemetry.addData("6", " Depot Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15,1.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
    //forward
    drive.encoder2Drive(robot.DRIVE_SPEED1, 32, 5.5);
    //turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7,2, 3.5);
    //forward
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// turn left
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 2, 3.5);
// forward
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
// dump off the marker

// backwards towards the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);

} else if (robot.path_option == 3) {
    telemetry.addData("6", " Depot Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15, 1.5);
    //turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 24, 5.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7, 1, 3.5);
    //forward toward the depot
    drive.encoder2Drive(robot.DRIVE_SPEED, 12, 3.5);
    //dump off the marker
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
//forward into the crater
drive.encoder2Drive(robot.DRIVE_SPEED, 5, 3.5);
//turn right
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
//forward toward the crater
drive.encoder2Drive(robot.DRIVE_SPEED1, 96, 10.5);

} else if (robot.path_option == 4) {
    telemetry.addData("6", " Crater Side turning Right");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15, 1.5);
    //turn right
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 1, 3.5);
    //forward going past the lander
    drive.encoder2Drive(robot.DRIVE_SPEED1, 29, 5.5);
    //turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
    //forward toward the wall
    drive.encoder2Drive(robot.DRIVE_SPEED1, 26, 5.5);

}else if (robot.path_option == 5) {
    telemetry.addData("6", " Crater Side turning Left");
    telemetry.update();
    //forward and get past the lander legs
    drive.encoder2Drive(robot.DRIVE_SPEED, 15, 1.5);
    //turn left
    turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 15, 2, 3.5);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
//forward and go to the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 60, 8.5);
//turn right
turn.encoder2DriveTurn2(robot.DRIVE_SPEED, 7.1, 3.5);
//forward towards the wall
drive.encoder2Drive(robot.DRIVE_SPEED1, 36, 5.5);
//dump off the marker

//backwards toward the crater
drive.encoder2Drive(-robot.DRIVE_SPEED1, -84, 10.5);
} else {
    //nearest to the audience
    //the distance is 23 15/16 but allow for role from stone to parking zone
    telemetry.addData("6", " No Valid option picked");
    telemetry.update();

}

if (tfod != null) {
    tfod.shutdown();
}
robot.wait(20);
telemetry.addData("Path", "Autonomous Complete");
telemetry.update();

idle(); //Always call idle()
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
//find the position of the gold mineral
private void findMineral() {
    double mineral_location;
    double mineral_location_silver1;
    double mineral_location_silver2;
    phone_in = true;
    phone_locked = false;
    boolean just_switched = false;
    robot.phone_position = .06;
    String label = "outward";
    robot.phone.setPosition(robot.phone_position);

    telemetry.addData("2", "in findMineral");
    if (tfod != null) {
        // getUpdatedRecognitions() will return null if no new information is available since
        // the last time that call was made.
        int count = 0;
        int found = 0;

        mineral_location = -1;
        mineral_location_silver1 = -1;
        mineral_location_silver2 = -1;
        int goldMineralX = -1;
        int silverMineral1X = -1;
        int silverMineral2X = -1;
        // loop through to find the three targets

        while ((goldMineralX == -1 && silverMineral1X == -1 && silverMineral2X == -1) || (!isStopRequested())
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
&& opModelsActive())) {  
  
    List<Recognition> updatedRecognitions = tfod.getUpdatedRecognitions();  
  
    // if data - check it  
    if (updatedRecognitions != null) {  
        // telemetry.addData("# Object Detected", updatedRecognitions.size());  
        count++;  
        // telemetry.addData("7", "track is %d", count);  
        int item_count = 0;  
        //phone_locked = false;  
        if (updatedRecognitions.size() > 0) {  
  
            for (Recognition recognition : updatedRecognitions) {  
                label = recognition.getLabel();  
                item_count++;  
                telemetry.addData("7", "checking - %s found %d", label , found);  
                telemetry.update();  
                if (recognition.getLabel().equals(LABEL_GOLD_MINERAL) && goldMineralX == -1) {  
                    found++;  
                    telemetry.addData("7", "found gold");  
                    telemetry.update();  
                    goldMineralX = (int) recognition.getLeft();  
                    mineral_location = robot.phone_position;  
                    robot.wait(1);  
                }  
                if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral1X == -1) {  
                    found++;  
                }  
            }  
        }  
    }  
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
telemetry.addData("7", "found silver");
telemetry.update();
silverMineral1X = (int) recognition.getLeft();
mineral_location_silver1 = robot.phone_position;
robot.wait(1);
} else if (recognition.getLabel().equals(LABEL_SILVER_MINERAL) && silverMineral2X == -1
&&
    mineral_location_silver1 != robot.phone_position) {
found++;
telemetry.addData("7", "found silver");
telemetry.update();
silverMineral2X = (int) recognition.getLeft();
mineral_location_silver2 = robot.phone_position;
robot.wait(1);
}
// telemetry.addData("4", "count %d - internal count %d", updatedRecognitions.size(),
item_count);
// telemetry.update();
if (item_count == updatedRecognitions.size() || mineral_location != 0) break;
}
}
}
telemetry.addData("er", "should move phone %b %b %b", phone_in, phone_out, phone_locked);
telemetry.update();
// move the phone box to find the target
if (phone_in && !phone_locked) {
//move phone (bottom is 1)
if (robot.phone_position > .41) {
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
phone_in = false;
phone_out = true;
} else {
    robot.phone_position += .12;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;

}
} else if (phone_out && !phone_locked) {
    // phone (top position is 0)
    if (robot.phone_position < .19) {
        phone_in = true;
        phone_out = false;
    } else {
        robot.phone_position -= .12;
        robot.phone.setPosition(robot.phone_position);
        phone_locked = true;
    }
}
if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 25000) {
        phone_locked = false;
        phonecyclecount = 0;
    }
}

telemetry.addData("8", "1Phone in %b Phone out %b pos %.3f", phone_in, phone_out, ro-
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
bot.phone_position);
    // telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
    telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1,
mineral_location_silver2);
    // telemetry.update();
    //robot.wait(5);
    idle(); // Always call idle()
    // if all items found then break out
    if (found == 3) break;
} //end of while

if (mineral_location != -1) {
    if (mineral_location == .18) {
        telemetry.addData("Gold Mineral Position", "Left");
        robot.gold_position = 1;
    } else if (mineral_location == .42) {
        telemetry.addData("Gold Mineral Position", "Right");
        robot.gold_position = 3;
    } else if (mineral_location == .3){
        telemetry.addData("Gold Mineral Position", "Center");
        robot.gold_position = 2;
    }
}
// telemetry.addData("8", "2Phone in %b Phone out %b pos %.3f", phone_in, phone_out, ro-
bot.phone_position);
// telemetry.addData("8", "telemetry %d %d %d %d
",found,goldMineralX,silverMineral1X,silverMineral2X);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
telemetry.addData("8", "mineral location %f %f %f ",mineral_location, mineral_location_silver1,
mineral_location_silver2);
telemetry.update();
}

// robot.wait(20);
}

public void waitForTick(long periodMs) {

long remaining = periodMs - (long)period.currentTimeMillis();

// sleep for the remaining portion of the regular cycle period.
if (remaining > 0) {
try {
Thread.sleep(remaining);
} catch (InterruptedException e) {
Thread.currentThread().interrupt();
}
}

// Reset the cycle clock for the next pass.
period.reset();
}

private void initVuforia() {
/*
 * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
*/}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
/*
VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

parameters.vuforiaLicenseKey = VUFORIA_KEY;
parameters.cameraDirection = VuforiaLocalizer.CameraDirection.BACK;

// Instantiate the Vuforia engine
vuforia = ClassFactory.getInstance().createVuforia(parameters);

}

/**
 * Initialize the Tensor Flow Object Detection engine.
 */
private void initTfod() {

    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());

    TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters
(tfodMonitorViewId);
    tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LABEL_SILVER_MINERAL);

}
}
```

We have also added new logic for an option to lock the tracking of the drive to be controlled by one of the joy sticks.

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

/ Copyright (c) 2015 Qualcomm Technologies Inc*

All rights reserved.

The function of this program is to run teleop.

Neither the name of Qualcomm Technologies Inc nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

*NO EXPRESS OR IMPLIED LICENSES TO ANY PARTY'S PATENT RIGHTS ARE GRANTED BY THIS LICENSE. THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE. */*

package org.firstinspires.ftc.Team7341;

```
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.eventloop.opmode.TeleOp;
import com.qualcomm.robotcore.hardware.DcMotor;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.ClassFactory;
import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer;
import org.firstinspires.ftc.robotcore.external.navigation.VuforiaTrackables;
import org.firstinspires.ftc.robotcore.external.tfod.TFObjectDetector;

import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotor.RunMode;
import static com.qualcomm.robotcore.util.Range.clip;
import static java.lang.Math.abs;
import static java.lang.String.format;
import static org.firstinspires.ftc.robotcore.external.navigation.VuforiaLocalizer.CameraDirection.BACK;

@TeleOp(name = "FF: ChickenNoodleFrenchFry", group = "FF")
//@Disabled
public class ChickenNoodleFrenchFry extends LinearOpMode {
    /* Declare OpMode members. */
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
private ElapsedTime runtime = new ElapsedTime();

PrinceCharlesBaDazzle robot = new PrinceCharlesBaDazzle();
private static final String VUFORIA_KEY = "AW/Sw13////
AAAAGVySmTiZ2EZAiMSFgHDTn7GDLYxyMC7ZEHNyvwpbJlmrEGBajczWU1Oi-
Num6rS90mBDJwrv1CJMc5Gk4rfMrqupHJIHQanX8hrPOwutOu5C918/
MZz7Zvp35rYD6lavfkgCMZ0DVAXhBv4J5LlrGlVXYfhhS1NkITGPDqVRW2aBmKLwctHzaztzycau3g//
QQ2EE0yCkj3K+rf5al3O64VWweNlaM9cptXyUaAP6/
rEsoZMaPnPfkyGcE-
Zuz1DStPn6ZriIRE+FhMistaO3ntLvZdi3WBTr8IE/9PXx2TIVmeEd7EZSawWCi+TcNfj8kNluN/
FOMjjlrFtBH+Uj/vVQZkJDx8QqH2EEed+AM+WKq"; // Insert your own key here

/**
 * {@link #vuforia} is the variable we will use to store our instance of the Vuforia
 * localization engine.
 */
private VuforiaLocalizer vuforia;
/**
 * {@link #tfod} is the variable we will use to store our instance of the Tensor Flow Object
 * Detection engine.
 */
private TFObjectDetector tfod;

// Leave argument list empty if you want to disable the camera monitor view.
TFObjectDetector.Parameters tfodParameters = new TFObjectDetector.Parameters();
private static final String TFOD_MODEL_ASSET = "RoverRuckus.tflite";
private static final String LABEL_GOLD_MINERAL = "Gold Mineral";
private static final String LABEL_SILVER_MINERAL = "Silver Mineral";
private boolean targetVisible = false;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// private boolean targetVisible = false;Valid choices are: BACK or FRONT

private static final VuforiaLocalizer.CameraDirection CAMERA_CHOICE = BACK;
// Variables to be used for later
private VuforiaLocalizer vuforiaLocalizer;
private VuforiaLocalizer.Parameters parameters;
private VuforiaTrackables visionTargets;
@Override
public void runOpMode() throws InterruptedException {

    // data for moving the phone
    boolean phone_locked;
    // set to be at the bottom
    int extender_position1 = 0;
    float extender_power1 = 0;
    int last_extender_position = -999;
    int extender_position = 2;
    int extender_direction = 0;
    int attop = 0;
    int atbottom = 1;
    int location = 0;
    int end_game = 0;
    int phonecyclecount;
    phone_locked = false;
    phonecyclecount = 0;

    // set phone data
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// State used for updating telemetry
// Send telemetry message to signify robot waiting;
telemetry.addData("Status", "Setting Up ChickenNoodleFrenchFry"); //
telemetry.update();

/*
 * Use the hardwareMap to get the dc motors and servos by name.
 * Note that the names of the devices must match the names used
 * when you configured your robot and created the configuration file.
 */

/*
 * Initialize the drive system variables.
 * The init() method of the hardware class does all the work here
 */
robot.init(hardwareMap, 1);

String startDate;

startDate = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss").format(new Date());

// end of the front-end of the robots definitions

telemetry.addData("Text", "Waiting to start ChickenNoodleFrenchFry");
telemetry.update();

// this is to initialize the camera for target viewing
// initVuforia();    telemetry.update();
/*
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
if (ClassFactory.getInstance().canCreateTFOBJECTDetector()) {  
    telemetry.addData("3", "setting up tfod");  
    telemetry.update();  
    initTfod();  
    telemetry.addData("3", "back from setting up tfod");  
    telemetry.update();  
    robot.wait(5);  
  
} else {  
    telemetry.addData("Sorry!", "This device is not compatible with TFOD");  
}  
*/  
  
waitForStart();  
  
// put servos in start position  
  
robot.init(hardwareMap, 2);  
robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);  
//move phone out of the way  
robot.phone_position = .84;  
robot.phone.setPosition(robot.phone_position);  
  
robot.lock_power = .5;  
robot.lock.setPosition(robot.lock_power);  
// Set up our telemetry dashboard  
composeTelemetry();
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// setting up timer
runtime.reset();

// while the op mode is active, loop and read the RGB data.
// Note we use opModelsActive() as our loop condition because it is an interruptible method.

while (opModelsActive()) {

    /*
    * Gamepad 1 controls the motors via the left/right stick
    */
    // set end flag
    if (gamepad1.a) {
        end_game = 1;
    }
    if (gamepad1.y) {
        end_game = 0;
    }

    // this is for the motor control function
    // forward is negative power value
    // backwards is positive power value
    int drive_mode = 2;

    if (gamepad1.right_stick_y != 0 || gamepad1.left_stick_y != 0 ||
        gamepad1.right_stick_x != 0 || gamepad1.left_stick_x != 0) {
        if (gamepad1.right_stick_y <= 0) {
            robot.right = ((-gamepad1.right_stick_y - (abs(gamepad1.right_stick_x))) / (float) .5);
        } else {
            robot.right = ((-gamepad1.right_stick_y + abs(gamepad1.right_stick_x))) / (float) .5;
        }
        if (gamepad1.left_stick_y <= 0) {
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
robot.left = ((-gamepad1.left_stick_y - (abs(gamepad1.left_stick_x))) / (float) .5);
} else {
    robot.left = ((-gamepad1.left_stick_y + abs(gamepad1.left_stick_x)) / (float) .5);
}

// clip the right/left values so that the values never exceed +/- 1
robot.right = clip(robot.right, -1, 1);
robot.left = clip(robot.left, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
robot.right = (float) scaleInput(robot.right);
robot.left = (float) scaleInput(robot.left);

if (end_game == 1) {
    robot.left = robot.right;
}
// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
} else {
    robot.right = 0;
    robot.left = 0;
// clip the right/left values so that the values never exceed +/- 1
robot.right = clip(robot.right, -1, 1);
robot.left = clip(robot.left, -1, 1);

// scale the joystick value to make it easier to control
// the robot more precisely at slower speeds.
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
robot.right = (float) scaleInput(robot.right);
robot.left = (float) scaleInput(robot.left);

// write the values to the motors
setDrivePower(robot.right, robot.left, drive_mode);
}

// Gamepad 2

// Lift (Raise and Lower) the end effector
// You should not go much past 90degrees
// up is negative value on the controller
// down is positive value on the controller
// lower_stop is to make sure the motor stop when touching the floor
// lift_top is to stop when just past being vertical
// if the digital channel returns true it's HIGH and the button is unpressed.
if ((gamepad2.left_stick_y < 0 && robot.upper_stop.getState() == true ) ||
    (gamepad2.left_stick_y > 0 && robot.lower_stop.getState() == true )) {
    // set lift speed to a constant value use the negative
    // negate the power value to get the motors going in the direction desired

    if (gamepad2.left_stick_y < -.75) gamepad2.left_stick_y =(float) -.75;
    if (gamepad2.left_stick_y > .95) gamepad2.left_stick_y =(float) .95;

    robot.lift_power = -gamepad2.left_stick_y;

    // clip the right/left values so that the values never exceed +/- 1
    robot.lift_power = clip(robot.lift_power, -1, 1);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
// scale the joystick value to make it easier to control  
// the robot more precisely at slower speeds.  
  
robot.lift_power = (float) scaleInput(robot.lift_power);  
  
robot.right_liftdrive.setPower(robot.lift_power);  
robot.left_liftdrive.setPower(robot.lift_power);  
  
} else {  
  
    // stop the lift motor  
  
    robot.lift_power = 0;  
    robot.lift_power = (float) scaleInput(robot.lift_power);  
    robot.right_liftdrive.setPower(robot.lift_power);  
    robot.left_liftdrive.setPower(robot.lift_power);  
}  
  
  
// raise the mineral grabber up/down  
// up is negative power value  
// going extender position = 2 we are at the bottom  
extender_position1 = robot.extenderdrive.getCurrentPosition();  
if (extender_position1 > last_extender_position) extender_direction = 2;  
if (extender_position1 < last_extender_position) extender_direction = 1;  
  
if ((gamepad2.right_stick_y < 0 )
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
|| (gamepad2.right_stick_y > 0)){  
  
last_extender_position = extender_position1;  
  
// speed to a constant value  
  
robot.extender_power = gamepad2.right_stick_y;  
// clip the right/left values so that the values never exceed +/- 1  
extender_power1 = clip(robot.extender_power, -1, 1);  
robot.extender_power = extender_power1;  
// scale the joystick value to make it easier to control  
// the robot more precisely at slower speeds.  
if (!robot.extender_lower.getState() && extender_direction == 1 && atbottom == 0) {  
    attop = 1;  
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);  
    location = robot.extenderdrive.getCurrentPosition();  
}  
if (!robot.extender_lower.getState() && extender_direction == 2 && attop == 0){  
    atbottom = 1;  
    robot.extenderdrive.setMode(DcMotor.RunMode.STOP_AND_RESET_ENCODER);  
    robot.extenderdrive.setMode(RunMode.RUN_WITHOUT_ENCODER);  
    location = robot.extenderdrive.getCurrentPosition();  
}  
if (robot.extender_lower.getState()){  
    atbottom = 0;  
    attop = 0;  
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
if (!robot.extender_lower.getState() && extender_direction == 1 && robot.extender_power < 0 /* && attop == 1 */ ) robot.extender_power = 0;
    if (!robot.extender_lower.getState() && extender_direction == 2 && robot.extender_power > 0 /* && atbottom == 1 */ ) robot.extender_power = 0;
        // over ride because you are at the top or bottom and you need to move
        if (!robot.extender_lower.getState() && atbottom == 1 && extender_power1 < 0) robot.extender_power = extender_power1;
        if (!robot.extender_lower.getState() && attop == 1 && extender_power1 > 0) robot.extender_power = extender_power1;

        robot.extender_power = (float) scaleInput(robot.extender_power);
        robot.extenderdrive.setPower(robot.extender_power);

    } else {

        // stop the extension motor

        robot.extender_power = 0;
        robot.extender_power = (float) scaleInput(robot.extender_power);
        robot.extenderdrive.setPower(robot.extender_power);

    }

    // wrench up/down
    if (gamepad2.right_bumper && robot.wrench_upper.getState() == true ) {
        //going up
        //make sure you are unlocked
    }
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
//unlocking
robot.lock_power = .5;
robot.lock.setPosition(robot.lock_power);

robot.wrench_power = .75;
robot.left_wrench.setPower(robot.wrench_power);
robot.right_wrench.setPower(-robot.wrench_power);
} else if (gamepad2.right_trigger != 0 && robot.wrench_lower.getState() == true) {
    //going down
    robot.wrench_power = .75;
    robot.left_wrench.setPower(-robot.wrench_power);
    robot.right_wrench.setPower(robot.wrench_power);
} else {
    //stop wrench
    robot.wrench_power = 0;
    robot.left_wrench.setPower(robot.wrench_power);
    robot.right_wrench.setPower(robot.wrench_power);
}

//set the lock so we can lift the robot
if (gamepad2.left_bumper) {
    //locking
    robot.lock_power = .16;
    robot.lock.setPosition(robot.lock_power);
} else if (gamepad2.left_trigger != 0) {
    //unlocking
    robot.lock_power = .5;
    robot.lock.setPosition(robot.lock_power);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
}

// pick the minerals up
if (gamepad2.b) {
    robot.collector_power = 1;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
if (gamepad2.x) {
    robot.collector_power = 0;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
// spit the minerals out
if (gamepad2.y) {
    robot.collector_power = -1;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
if (gamepad2.a) {
    robot.collector_power = 0;
    robot.collector_power = (float) scaleInput(robot.collector_power);
    robot.collectordrive.setPower(robot.collector_power);
}
// MOVE THE ARMS
if (gamepad1.x) {
    robot.right_arm_power = .5;
    robot.right_arm.setPosition(robot.right_arm_power);
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
robot.left_arm_power = .5;
robot.left_arm.setPosition(robot.left_arm_power);
}

if (gamepad1.b) {
    robot.right_arm_power = 1;
    robot.right_arm.setPosition(robot.right_arm_power);
    robot.left_arm_power = 0;
    robot.left_arm.setPosition(robot.left_arm_power);
}

// right bumper move out the phone
// left bumper move in the phone

if (gamepad1.right_bumper && robot.phone.getPosition() >= 0 && !phone_locked) {
    //move phone (bottom is 1)
    robot.phone_position += .1;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;

} else if (gamepad1.left_bumper && robot.phone.getPosition() <= 1 && !phone_locked) {
    // phone (top position is 0)
    robot.phone_position -= .1;
    robot.phone.setPosition(robot.phone_position);
    phone_locked = true;
}

if (phone_locked) {
    phonecyclecount++;
    if (phonecyclecount == 100){
        phone_locked = false;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
    phonecyclecount = 0;
}
}

telemetry.addData("0", "FF ChickenNoodleFrenchFry - %2.5f S Elapsed", runtime.seconds());
// get the value of the light

telemetry.addData("D1", format("Rx: %.2f Lx: %.2f Ry: %.2f Ly: %.2f", gamepad1.right_stick_x,
gamepad1.left_stick_x, gamepad1.right_stick_y, gamepad1.left_stick_y));
telemetry.addData("D2", format("Rx: %.2f Lx: %.2f ", gamepad2.right_stick_x,
gamepad2.left_stick_x));
telemetry.addData("1", format("R - %.2f L - %.2f - Lift - %.2f - Extender - %.2f", robot.right, ro-
bot.left, robot.lift_power,robot.extender_power));

// telemetry.addData("3", format("Floor Color %2d Blue %2d Red %2d", robot.floor_color.alpha(),
robot.floor_color.blue(), robot.floor_color.red()));
telemetry.addData("4", format("Lower %s upper %s", robot.lower_stop.getState(), ro-
bot.upper_stop.getState()));
telemetry.addData("Extender state is ", robot.extender_lower.getState());
telemetry.addData("5", format("phone position %.2f - %d",robot.phone.getPosition
(),robot.extenderdrive.getCurrentPosition()));
telemetry.addData("6",format("extnder direction %d",extender_direction));
telemetry.addData("9",format("atbottom %d atop %d",atbottom, atop));
telemetry.addData("9",format("location %d end_game %d",location, end_game));
telemetry.update();
idle(); // Always call idle() at the bottom of your while(opModelsActive()) loop
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
}

/*
 * This method scales the joystick input so for low joystick values, the
 * scaled value is less than linear. This is to make it easier to drive
 * the robot more precisely at slower speeds.
 */
double scaleInput(double dVal) {
    double[] scaleArray = {0.0, 0.05, 0.09, 0.10, 0.12, 0.15, 0.18, 0.24,
        0.30, 0.36, 0.43, 0.50, 0.60, 0.72, 0.85, 1.00, 1.00};
;

    // get the corresponding index for the scaleInput array.
    int index = (int) (dVal * 16.0);
    if (index < 0) {
        index = -index;
    } else if (index > 16) {
        index = 16;
    }

    double dScale = 0.0;
    if (dVal < 0) {
        dScale = -scaleArray[index];
    } else {
        dScale = scaleArray[index];
    }

    return dScale;
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
public void setDrivePower(float rightPower, float leftPower, int power_mode) {
    // telemetry.addData("2", format("PM - %d", power_mode));
    // telemetry.update();
    if (power_mode == 2) {
        // set front and back motors full power
        robot.righthdrive.setPower(rightPower);
        robot.leftdrive.setPower(leftPower);
        robot.righttwo.setPower(rightPower);
        robot.lefttwo.setPower(leftPower);
    } else if (power_mode == 1) {
        // set front motors
        robot.righthdrive.setPower(0);
        robot.leftdrive.setPower(0);
        robot.righttwo.setPower(0);
        robot.lefttwo.setPower(0);
    } else {
        // set front motors
        robot.righthdrive.setPower(0);
        robot.leftdrive.setPower(0);
        robot.righttwo.setPower(0);
        robot.lefttwo.setPower(0);
    }
}

public void setDriverMode(RunMode mode) {
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
if (robot.leftdrive.getMode() != mode) {  
    robot.leftdrive.setMode(mode);  
}  
if (robot.righthdrive.getMode() != mode) {  
    robot.righthdrive.setMode(mode);  
}  
  
}  
//-----  
// Telemetry Configuration  
//-----  
  
void composeTelemetry() {  
  
    telemetry.addData("ball color red", "ball red color");  
    telemetry.update();  
    // At the beginning of each telemetry update, grab a bunch of data  
    // from the IMU that we will then display in separate lines.  
  
    telemetry.addAction(new Runnable() { @Override public void run()  
    {  
        // Acquiring the angles is relatively expensive; we don't want  
        // to do that in each of the three items that need that info, as that's  
        // three times the necessary expense.  
        robot.angles = robot imu.getAngularOrientation(AxesReference.INTRINSIC, AxesOrder.ZYX, AngleUnit.DEGREES);  
    }  
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

});

```
telemetry.addLine()
    .addData("status", new Func<String>() {
        @Override public String value() {
            return robot imu.getSystemStatus().toShortString();
        }
    })
    .addData("calib", new Func<String>() {
        @Override public String value() {
            return robot imu.getCalibrationStatus().toString();
        }
    });
}

telemetry.addLine()
    .addData("heading", new Func<String>() {
        @Override public String value() {
            return formatAngle(robot angles.angleUnit, robot angles.firstAngle);
        }
    })
    .addData("roll", new Func<String>() {
        @Override public String value() {
            return formatAngle(robot angles.angleUnit, robot angles.secondAngle);
        }
    })
    .addData("pitch", new Func<String>() {
        @Override public String value() {
            return formatAngle(robot angles.angleUnit, robot angles.thirdAngle);
        }
    });
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
}

});

//-----
// Formatting
//-----



String formatAngle(AngleUnit angleUnit, double angle) {
    return formatDegrees(AngleUnit.DEGREES.fromUnit(angleUnit, angle));
}

String formatDegrees(double degrees){
    return String.format(Locale.getDefault(), "%.1f", AngleUnit.DEGREES.normalize(degrees));
}

private void initVuforia() {
    /*
     * Configure Vuforia by creating a Parameter object, and passing it to the Vuforia engine.
     */
    VuforiaLocalizer.Parameters parameters = new VuforiaLocalizer.Parameters();

    parameters.vuforiaLicenseKey = VUFORIA_KEY;
    parameters.cameraDirection = BACK;

    // Instantiate the Vuforia engine
    vuforia = ClassFactory.getInstance().createVuforia(parameters);
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
}

/**
 * Initialize the Tensor Flow Object Detection engine.
 */

private void initTfod() {

    int tfodMonitorViewId = hardwareMap.appContext.getResources().getIdentifier(
        "tfodMonitorViewId", "id", hardwareMap.appContext.getPackageName());

    TFOBJECTDETECTOR.Parameters tfodParameters = new TFOBJECTDETECTOR.Parameters
    (tfodMonitorViewId);

    tfod = ClassFactory.getInstance().createTFOBJECTDETECTOR(tfodParameters, vuforia);
    tfod.loadModelFromAsset(TFOD_MODEL_ASSET, LABEL_GOLD_MINERAL, LABEL_SILVER_MINERAL);

}

}
```

Following is the hardware definition of Princess Charlie:

```
package org.firstinspires.ftc.Team7341;

import com.qualcomm.hardware.bosch.BNO055IMU;
import com.qualcomm.hardware.bosch.JustLoggingAccelerationIntegrator;
import com.qualcomm.hardware.modernrobotics.ModernRoboticsI2cRangeSensor;
import com.qualcomm.robotcore.eventloop.opmode.LinearOpMode;
import com.qualcomm.robotcore.hardware.CRServo;
import com.qualcomm.robotcore.hardware.ColorSensor;
import com.qualcomm.robotcore.hardware.DcMotor;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
import com.qualcomm.robotcore.hardware.DigitalChannel;
import com.qualcomm.robotcore.hardware.HardwareMap;
import com.qualcomm.robotcore.hardware.I2cAddr;
import com.qualcomm.robotcore.hardware.Servo;
import com.qualcomm.robotcore.hardware.TouchSensor;
import com.qualcomm.robotcore.util.ElapsedTime;

import org.firstinspires.ftc.robotcore.external.Func;
import org.firstinspires.ftc.robotcore.external.navigation.Acceleration;
import org.firstinspires.ftc.robotcore.external.navigation.AngleUnit;
import org.firstinspires.ftc.robotcore.external.navigation.AxesOrder;
import org.firstinspires.ftc.robotcore.external.navigation.AxesReference;
import org.firstinspires.ftc.robotcore.external.navigation.Orientation;

import java.util.Locale;

import static com.qualcomm.robotcore.hardware.DcMotorSimple.Direction.REVERSE;
import static java.lang.Thread.currentThread;
import static java.lang.Thread.sleep;

/**
 * This is NOT an opmode.
 * <p>
 * This class can be used to define all the specific hardware for a single robot.
 * In this case that robot is PrinceCharles.
 * See AutoBlue and others classes starting with "FF" for usage examples.
 * <p>
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

* This hardware class assumes the following device names have been configured on the robot:
* Note: All names are lower case and some have single spaces between words.
*/

```
public class PrinceCharlesBaDazzle {

    // setup for calculation of the how far to move

    static final double P_TURN_COEFF      = 0.1;    // Larger is more responsive, but also less stable
    static final double HEADING_THRESHOLD = 1;       // As tight as we can make it with an integer gyro

    static final double COUNTS_PER_MOTOR_REV  = 1680; // eg: AndyMark Motor Encoder
    static final double DRIVE_GEAR_REDUCTION = 1;     // This is < 1.0 if geared UP
    static final double WHEEL_DIAMETER_INCHES = 4.0;   // For figuring circumference
    static final double GEAR_DIAMETER_INCHES = 1.0;   // For figuring circumference

    static final double COUNTS_PER_INCH      = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
        (WHEEL_DIAMETER_INCHES * 3.1415);
    static final double LIFT_COUNTS_PER_INCH = (COUNTS_PER_MOTOR_REV *
DRIVE_GEAR_REDUCTION) /
        (GEAR_DIAMETER_INCHES * 3.1415);

    static final double DRIVE_SPEED        = 0.5;
    static final double DRIVE_SPEED1       = 0.9;
    static final double STONE_DRIVE_SPEED  = 0.2;
    static final double TURN_SPEED         = 0.3 ;
    static final double TOUCH_SPEED        = 0.1;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
static final double LIFT_SPEED = 0.2;
static final double STOP_SPEED = 0;
```

```
int floor_color_value;
int position_option = 0;
int position_side = 0;
int turn_option = 0;
int path_option = 0;
int find_target = -1;
int hanging = -1;
int gold_position = 0;
int count = 0;
// Driver motors
DcMotor rightdrive;
DcMotor leftdrive;
DcMotor righttwo;
DcMotor lefttwo;
float right = 0;
float left = 0;
```

// Distance control

```
ModernRoboticsI2cRangeSensor distance;
```

//motors to lift and lower

```
DcMotor right_liftdrive;
DcMotor left_liftdrive;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
float lift_power = 0;

// move the collector in/out up/down
DcMotor extenderdrive;
// pick up the minerals
DcMotor collectordrive;
float extender_power = 0;
float collector_power = 0;

// open position
double right_position = 0;
double left_position = .5;

// Side phone
Servo phone;
double phone_position = 0.26;
CRServo left_wrench;
CRServo right_wrench;
double wrench_power = 0;

Servo lock;
double lock_power = 0;
Servo marker;
double marker_power = 0;
Servo right_arm;
double right_arm_power = 0.5;
Servo left_arm;
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
double left_arm_power = 0.5;
```

```
DigitalChannel lower_stop;
DigitalChannel upper_stop;
DigitalChannel extender_lower;
DigitalChannel extender_upper;
DigitalChannel wrench_lower;
DigitalChannel wrench_upper;
```

```
// Our sensors, motors, and other devices go here, along with other long term state
BNO055IMU imu;
```

```
// State used for updating telemetry
Orientation angles;
Acceleration gravity;
```

```
/* local OpMode members. */
HardwareMap hardwareMap = null;
private ElapsedTime period = new ElapsedTime();
```

```
// Private Members
private LinearOpMode myOpMode;
```

```
/* Constructor */
public PrinceCharlesBaDazzle() {

}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
/* Initialize standard Hardware interfaces */
public void init(HardwareMap ahwMap, int option) {
    // Save reference to Hardware map
    hardwareMap = ahwMap;

    /*
     * Use the hardwareMap to get the dc motors and servos by name.
     * Note that the names of the devices must match the names used
     * when you configured your robot and created the configuration file.
     */
    // if option is one define hardware... otherwise put it start position..
    if (option == 1) {
        // start of drive train definitions
        rightdrive = hardwareMap.dcMotor.get("right_drive");
        leftdrive = hardwareMap.dcMotor.get("left_drive");
        leftdrive.setDirection(REVERSE);
        righttwo = hardwareMap.dcMotor.get("righttwo");
        lefttwo = hardwareMap.dcMotor.get("lefttwo");
        lefttwo.setDirection(REVERSE);
        // end of drive train definitions

        right_liftdrive = hardwareMap.dcMotor.get("right_lift");
        right_liftdrive.setDirection(REVERSE);
        left_liftdrive = hardwareMap.dcMotor.get("left_lift");
        right_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);
        left_liftdrive.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);
    }
}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
collectordrive = hardwareMap.dcMotor.get("collector");

extenderdrive = hardwareMap.dcMotor.get("extender");
extenderdrive.setDirection(REVERSE);
// stop the lifter when down flat
lower_stop = hardwareMap.get(DigitalChannel.class, "lower_stop");
upper_stop = hardwareMap.get(DigitalChannel.class, "upper_stop");
// Center piece
wrench_lower = hardwareMap.get(DigitalChannel.class, "wrench_lower");
wrench_upper = hardwareMap.get(DigitalChannel.class, "wrench_upper");
// outer piece
extender_lower = hardwareMap.get(DigitalChannel.class, "extender_lower");
extender_upper = hardwareMap.get(DigitalChannel.class, "extender_upper");
// set the digital channel to input.
lower_stop.setMode(DigitalChannel.Mode.INPUT);
upper_stop.setMode(DigitalChannel.Mode.INPUT);
wrench_lower.setMode(DigitalChannel.Mode.INPUT);
wrench_upper.setMode(DigitalChannel.Mode.INPUT);
extender_lower.setMode(DigitalChannel.Mode.INPUT);
extender_upper.setMode(DigitalChannel.Mode.INPUT);

phone = hardwareMap.servo.get("phone");
left_wrench = hardwareMap.get(CRServo.class, "left_wrench");
right_wrench = hardwareMap.get(CRServo.class, "right_wrench");

lock = hardwareMap.servo.get("lock");
marker = hardwareMap.servo.get("marker");
right_arm = hardwareMap.servo.get("right_arm");
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
left_arm = hardwareMap.servo.get("left_arm");
// set up the guidance hardware

// Set up the parameters with which we will use our IMU. Note that integration
// algorithm here just reports accelerations to the logcat log; it doesn't actually
// provide positional information.
BNO055IMU.Parameters parameters = new BNO055IMU.Parameters();
parameters.angleUnit      = BNO055IMU.AngleUnit.DEGREES;
parameters.accelUnit       = BNO055IMU.AccelUnit.METERS_PERSEC_PERSEC;
parameters.calibrationDataFile = "BNO055IMUCalibration.json"; // see the calibration sample op-
mode
parameters.loggingEnabled   = true;
parameters.loggingTag        = "IMU";
parameters.accelerationIntegrationAlgorithm = new JustLoggingAccelerationIntegrator();

// Retrieve and initialize the IMU. We expect the IMU to be attached to an I2C port
// on a Core Device Interface Module, configured to be a sensor of type "AdaFruit IMU",
// and named "imu".
imu = hardwareMap.get(BNO055IMU.class, "imu");
imu.initialize(parameters);

} else {

    // set the phones
    phone_position = .26;
    phone.setPosition(phone_position);

}
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

}

```
public void wait(int sec) {

    for (int i = 0; i < 2 * sec; i++) {
        try {
            sleep(500);
        } catch (InterruptedException e) {
            currentThread().interrupt();
            break;
        }
    }

    /**
     *
     * * waitForTick implements a periodic delay. However, this acts like a metronome with a regular
     * periodic tick. This is used to compensate for varying processing times for each cycle.
     * The function looks at the elapsed cycle time, and sleeps for the remaining time interval.
     *
     * @param periodMs Length of wait cycle in mSec.
     */
}

public void waitForTick(long periodMs) {

    long remaining = periodMs - (long) period.milliseconds();

    // sleep for the remaining portion of the regular cycle period.
    if (remaining > 0) {
```

Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Engineering Activity Continued

Date: January 27, 2019

Process:

```
try {
    sleep(remaining);
} catch (InterruptedException e) {
    currentThread().interrupt();
}
}

// Reset the cycle clock for the next pass.
period.reset();
}
```

We look forward to add more to the document after the meet on this coming Saturday

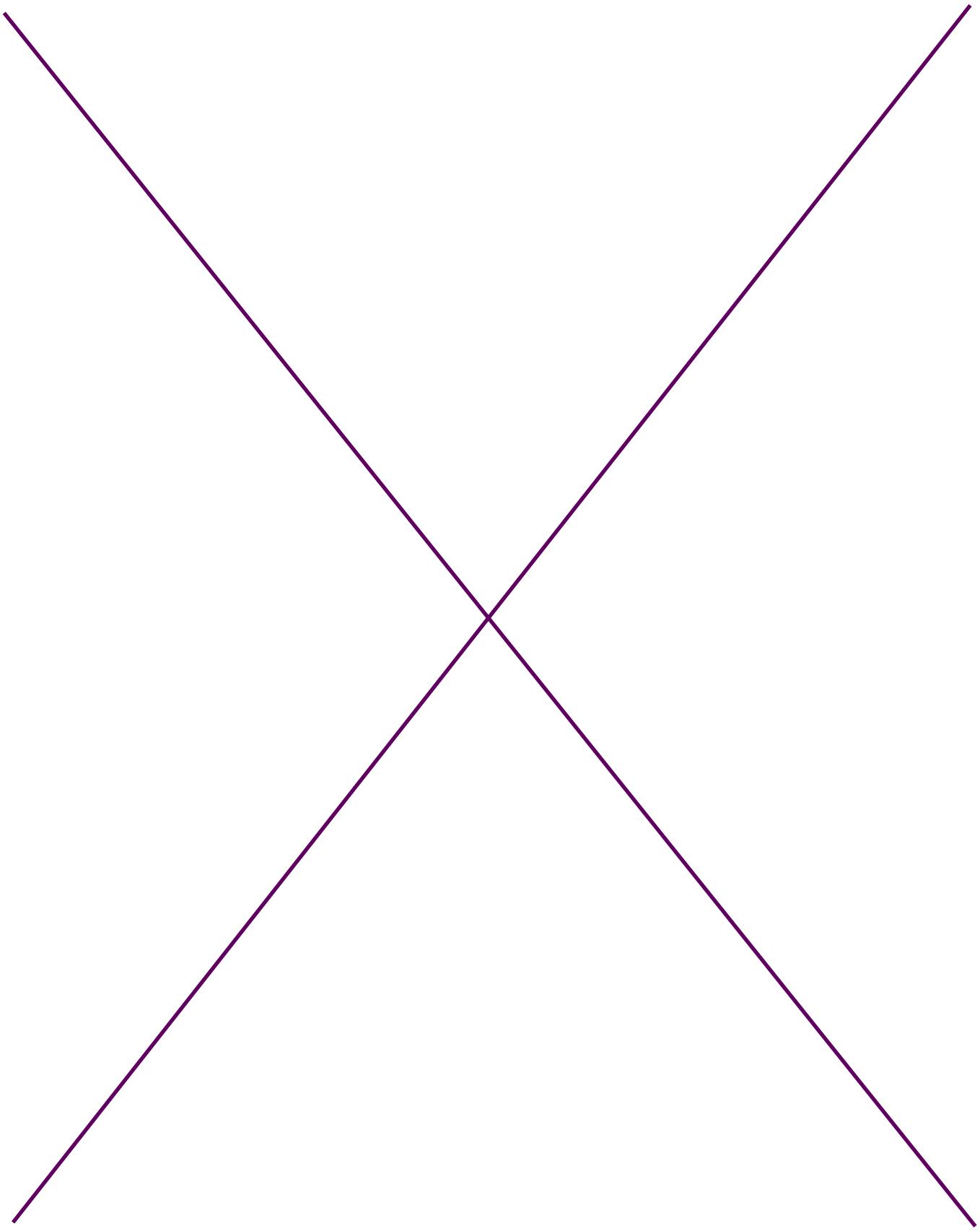
Signature : Jessica Anderson

Date: Jan. 27, 2019

**BUSINESS
MANAGEMENT**

Signature : Jessica Anderson

Date: Jan. 27, 2019



Signature : Jessica Anderson

Date: Jan. 27, 2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan

Introduction

FIRST

FIRST (For Inspiration and Recognition of Science and Technology) was founded in 1989 to inspire young people's interest and participation in science and technology. Based in Manchester, NH, **FIRST** is a not-for-profit public charity that designs accessible, innovative programs that motivate young people to pursue education and career opportunities in science, technology, engineering, art, and math (STEAM) while building self-confidence, knowledge, and life skills.

This organization supports the building of these skills from Kindergarten through High School. The **FIRST® LEGO® League Jr.** (FLL Jr) supports grades K-4 and through exploratory research, hands-on construction, teamwork, imagination, and fun with LEGO elements, FIRST® LEGO® League Jr. challenges teams of up to six kids to explore a real-world scientific concept, then build a motorized model and develop a Show Me Poster to illustrate their journey of discovery. Throughout the season, adult coaches provide guidance and inspiration.

The FIRST® LEGO® League (FLL) supports grades 4-8 and through apply science, engineering, and math concepts, plus a big dose of imagination, to develop solutions to real-world challenges with adult Coaches to guide a team of up to 10 members. They also design, build, and program LEGO MINDSTORMS®-based robots to perform autonomous "missions" on a challenge field. Along the way, they develop critical thinking, team-building, and presentation skills.

The FIRST® Tech Challenge (FTC) supports grades 7-12 and follows a sports model where teams of up to 10 members design, build, program, and operate robots of their own design to play a floor game in an alliance format. They are guided by adult Coaches and Mentors where the students develop STEAM skills and practice engineering principles, while realizing the value of hard work, innovation, and sharing ideas. Participants have access to tens of millions of dollars in college scholarships.

Team 4323, B.E.E.s

We are a Girl Scout team of 6 students who attend different schools in Brevard County and we are based in Titusville, Florida. We participate in the FIRST® LEGO® League Jr. program. Our team started 3 years ago and is ever changing as the girls' progress through the FIRST® program.

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

With the skills that we have learned as a team our team members are ready to conquer any challenge they meet out in the world.

Team 7407, C.A.K.E. B.A.T.T.E.R.S.

We are a Girl Scout team of 4 students who attend different schools in Brevard County and we are based in Titusville, Florida. We participate in the FIRST® LEGO® League Jr. program. Our team started 9 years ago (the team name changed several times) and is ever changing as the girls' progress through the FIRST® program.

With the skills that we have learned as a team our team members are ready to conquer any challenge they meet out in the world.

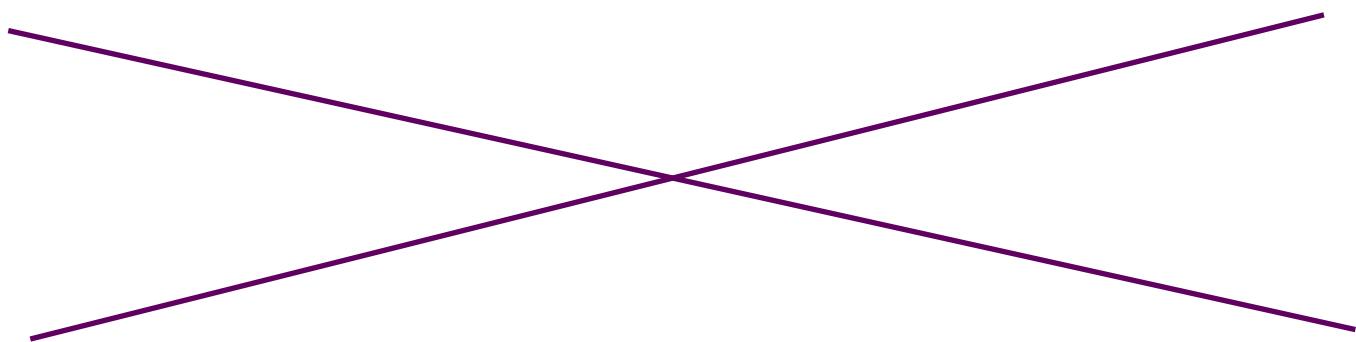
Team 7341, F.R.E.N.C.H. F.R.I.E.S.

We are a Girl Scout team of 2 High School students who attend different schools in Brevard County and we are based in Titusville, Florida. We have participated in the FIRST® Tech Challenge program for the last 6 years, and is ever changing as the girls' progress through school and graduate.

With the skills that we have learned as a team our graduating seniors are ready to conquer any challenge they meet out in the world.

Mission Statement (All Levels)

To build a world class model, robots, engineering notebooks, and spread the word about robotics and our new skills in our schools and community to promote *FIRST* and STEAM learning as we grow in our own knowledge.



Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Goals

Major Project Milestones

FIRST LEGO League Jr

- August 1, 2018 FLL Jr Challenge Release
- Early September Receive the FLL Jr Inspiration Kit
- September 2018 Build the Inspiration Model and determine a project
- October 2018 Complete the Model and Show Me Poster
- November 2018 Complete our Project
- November – December 2018 Share our Model, Show Me Poster and what we have learned
- January - March 2019 Attend one or two Expos
- April 19-20, 2019 World Championship, Houston, Texas

FIRST LEGO League

- August 28, 2018 Complete our Guess of the FLL Challenge
- August 29, 2018 FLL Challenge Release
- September 2018 Complete programming ½ of the missions and determine a project
- January 2019 Attend a qualifier tournament in Florida

FIRST Tech Challenge

- September 8, 2018 FTC Game Release
- Attending Eastern-Central Florida FTC 2018-19 Season Kick-off
- November 2018 Space Coast League Qualifier in Rockledge Florida
- December 2018 Co-hosted Space Coast League Qualifier in Titusville Florida
- January 2019 Space Coast League Qualifier in Sanford Florida
- February 2019 Space Coast League Championship in Oviedo Florida
- February 2019 Florida State Championship
- April 16-20, 2019 World Championship in Houston Texas

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Team Goals

FIRST LEGO League Jr

We meet at the beginning of each competition season, to set goals for our team. The goals we have set for the 2018-2019 season are as follows:

- Design and build our model based on the challenge requirements:
 - ⇒ Learn about real-world issues that are based on this year's challenge
 - ⇒ Developing a comprehensive LCAD model to share
 - ⇒ Find a solution to one of the real-world issues and share our information
 - Apply what we have learned about Core Values and Gracious Professionalism to our team activities and our everyday life.
 - Reach out to professionals to gain insight on how to improve our model, project and to grow in our own knowledge.

FIRST LEGO League

We meet at the beginning of each competition season, to set goals for our team. The goals we have set for the 2018-2019 season are as follows:

- Design and build our robot mission on the challenge requirements:
 - ⇒ Learn about real-world issues that are based on this year's challenge
 - ⇒ Developing a comprehensive LCAD model to share
 - ⇒ Find a solution to one of the real-world issues and share our information
- Apply what we have learned about Core Values and Gracious Professionalism to our team activities and our everyday life.
- Reach out to professionals to gain insight on how to improve our project and to grow in our own knowledge.

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

FIRST Tech Challenge

We meet at the beginning of each competition season, to set goals for our team. The goals we have set for the 2018-2019 season are as follows:

- Design and fabricate a fast, nimble and robust robot that is competitive at the world level by:
 - ⇒ Developing a comprehensive AutoCAD model
 - ⇒ Designing a more efficient attachment to complete the task
 - ⇒ Design the robot in modular parts
- Grow FTC and STEAM in our schools and community by
 - ⇒ Mentoring teams in our community, both FLL Jr, FLL and other FTC teams
 - ⇒ Sharing more with other teams using social media posting our findings
 - ⇒ Hosting a FIRST Tech Challenge Meet
- Reach out to professionals to gain insight on how to improve our robot and to grow in our own knowledge.

Projected Hours and Team Commitment for all three teams

Last year, it was determined that each team member averaged between 600 and 650 hours on robotics related activities including, but not limited to, robot design and build, outreach activities, training, and competitions. We talk about all three teams because the lower aged teams feed up to the upper aged teams.

This year, we project the total number of hours spent on robotic related activities to decrease due to college prep, family time, budget concerns, and extracurricular activities. The team has committed to the following:

- 3-4 hours/week to design and build competition robot; hours spent on robot design will increase the closer we get to a tournament or expo.
- Participation in Expos, qualifiers, League Championship and State Tournaments
- Mentoring other FIRST LEGO League Jr, FIRST LEGO League or FIRST Tech Challenge teams
- Demonstrations

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

We believe with this level of commitment, we can be a very competitive teams and can help promote robotics and STEAM in our community.

Sustainability

2018 – 2019 Budget

FIRST LEGO League Jr 2018 – 2019 Budget

FIRST Lego League Jr	Income:	Expenses:
Sponsorships	\$7,383	
FLL Jr Registration		\$103
Fundraising	\$500	
FLL Jr KSC FIRST Season Launch Event		\$240
FLL Jr Expo (3)		\$100
FLLL Jr World Championship*		\$300
Lodging (6 Rooms x 4 nights @\$110/room)*		\$2,640
Airfare (5 team + 1 coach + 5 parent ~\$300)*		\$3,300
LEGO Parts		\$350
WeDo Motors - 2 @\$20.95/each		
WeDo Battery pack – 2 @\$62.95/each		
Boost Creative Toolbox - \$159.95		
Display Board – Show Me Poster		\$50
Team Table/Giveaways (4 Expos)		\$200
Outreach		\$200
Team Shirts and Costumes		\$200
Miscellaneous		\$200
Total:	\$7,883	\$7,883
* Must qualify in order to compete at these tournament.		

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

FIRST LEGO League 2018 – 2019 Budget

FIRST LEGO League	Income:	Expenses:
Sponsorship	\$1,438	
FLL Registration		\$311.95
Fundraising	\$500	
FLL KSC FIRST Season Launch Event		\$240
League Fees - FLL Qualifiers (2)		\$150
FLL Regional Tournament*		\$75
FLL State Tournament*		\$100
FLL Parts		\$186
Motors - 2 @ \$26.95		
Battery Packs - 1 @ \$68.50		
Touch Sensor - 1 @ \$21.95		
Color Sensor - 1 @ \$41.95		
Challenge Set		\$75
Display Board		\$50
Pit/Giveaways		\$200
Outreach		\$200
Team Shirts and Costume Update		\$200
Miscellaneous		\$200
Total:	\$1,938	\$1,938
* Must qualify in order to compete at these tournaments.		

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

FIRST Tech Challenge 2017 – 2018 Budget

FIRST Tech Challenge	Income:	Expenses:
Sponsorship	\$7,489	
Special Grant	\$2,595	
FTC Registration		\$275
Fundraising	\$500	
League Fees		\$240
FLL KSC FIRST Season Launch Event		\$240
FTC Qualifiers/Championship (3)		\$160
FTC State Tournament		\$500
State Lodging (4 Rooms x 3 nights @\$110/room) *		\$1,320
World Lodging (4 Rooms x 4 nights @\$110/room) *		\$1,760
Airfare (4 team + 1 coach + 1 chaperone ~\$200)*		\$1,1200
FTC Parts		\$1,369
Motors - 4 @ \$28/each		
Servos – 6 @\$25/each		
REV Robotics Expansion Hub \$150.00		
Hardware Parts - \$97.00		
Sprocket/Chain Pack - \$70		
Battery Packs - 2 @ \$50/each		
Replacement Tiles \$32		
Power Pole Set - \$75		
Wheels 4 @ 9.95/each		
Omni Wheels 2 @ \$25/each		
Motor mounts 4 @ \$15.95		
Tools – Vise Grip/Chain Breaker, \$40		
Shipping - \$100 (estimate)		
Rover Ruckus Field		\$520
Engineering Notebook		\$150
Pit/Giveaways		\$250
Outreach		\$400
Team Shirts and Costume update		\$200
Miscellaneous		\$500
Total:	\$10,584	\$10,584
* Must qualify in order to compete at these tournaments.		

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Recruiting Team Members (All Teams)

We recruit new team members using the following guidelines:

- Must be a or want to join Girl Scouts
- Must be interested in learning new things
- Must be willing to work hard
- Must want to have fun
- No previous robotics experience necessary

Training (All Teams)

We are actively pursuing ways to work with professional mentors to learn new skills and to improve our knowledge of engineering design, CAD, programming, and other STEAM areas. Our goals for this year are:

- Learn AutoCAD or LCAD
- Learn more about JAVA Programming, WeDo² Programming and MINDSTORMS® Programming
- Learn more about videography
- Learn more about tools in the woodshop to build parts and project pieces

Risks and Opportunities

Our risks and opportunities are focused on the sustainability of the team from an execution and monetary perspective. The risks and opportunities in competition are fluid and can only be minimized and maximized, respectively, with solid preparation.

Risks

- Risk 1 – Conflict with other Activities: All the members of the team participate in other after-school activities including ROTC, Band, Church, Girl Scouts and Sports. Conflicts with these activities have already been identified, specifically, an important performance in Band by two of the members. To date, no major Church, ROTC, Girl Scouts or Sports conflicts have been identified. Risk 1 is open and carries a “medium” level.

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

- Risk 2 – Budget Shortfall: Due to the current economic environment, fundraising is more challenging than in past years. Also, an additional level of competition, the Super Regional, has been added to the tournament schedule. To mitigate this risk the team has put more emphasis on planning both expenses and fundraising. The detailed budget in section 3.1 was developed to ensure all expenses are necessary and is constantly reviewed. For example, if the team qualifies for advancement in an early tournament, competing in additional qualifying tournaments at that level may be dropped to conserve funds. Risk 2 is open and carries a “high” level.

Opportunities

- Opportunity 1 – Expand Corporate Sponsor Requests: In the past, requests for support were made mainly to Engineering companies and organizations. Leveraging the past work in STEAM, it may be possible to get other organizations interested in supporting the team. Candidates identified for the expanded sponsorship are: government grants (from the Legislators) and local businesses.
- Opportunity 2 – Expand Fundraising and Family Contributions: To be competitive at the World Tournament level requires a major commitment of time from all team members. This year the team building time can be spent on fundraising doing activities like car or dog washes. Any shortfall of funding the budget will need to be made up by Family Contributions.

Outreach and Recognition

2018 - 2019 Outreach Activities

The mission of *FIRST* is to show students of every age that science, technology, and problem-solving are fun and rewarding. We have become passionate about this mission as well and have focused our outreach activities to spread this vision. We have learned first-hand that while helping others is fun and rewarding. Some of our community outreach activities will include:

- Girl Scout Homecoming Events
- Girl Scout STEAM Event
- Girl Scouts Annual Meeting
- Community Events
- Lockheed Martin – Young Minds at Work (YMAW) event

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Recognition

FIRST LEGO League Jr – B.E.E.s

With our success during the last 3 seasons, we have been honored to receive the following recognitions:

- Simply Awesome Machine Award 2017-2018
- Amazing Movement Award and Artistic Eye Award 2017-2018
- Solid As A Rock Model Design Award 2017-2018
- Invited to World Champion 2017-2018
- World Champion Simply Awesome Machine Award – 2016 - 2017
- Amazing Movement Award – 2016 - 2017
- Master Programmer Award – 2016 – 2017
- Featured in the Citrus Peel volume 8
- World Champion Outstanding Teamwork Award – 2015 - 2016

FIRST LEGO League C.A.K.E. B.A.T.T.E.R.S.

With our success during the last 9 seasons, we have been honored to receive the following recognitions:

- Teamwork—2017-2018
- Raising Star – 2015-2016
- Project Design – 2014 - 2015
- Project Award – 2013-2014
- Team Spirit Award – 2012-2013
- Technical Design – 2011 - 2012

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

- Robot Programming and Design – 2010 – 2011

FIRST Tech Challenge F.R.E.N.C.H. F.R.I.E.S.

With our success during the last 5 seasons, we have been honored to receive the following recognitions:

- Control Award - 2017-2018
- 2nd Place Inspire Award - 2017-2018
- 3rd Place Think Award - 2017-2018
- Connect Award - 2016-2017
- 2nd Place Inspire Award - 2016-2017
- 3rd Place Think Award – 2016-2017
- Think Award at a Qualifying Tournament
- Think Award – 2015-2016
- 2nd Place Inspire Award - 2015-2016
- 3rd Place Think Award - 2015-2016
- Think Award at a Qualifying Tournament
- Recognized in the Citrus Peel volume 4
- Design Award – 2013-2014
- 2nd Place Think Award - 2013-2014
- 3rd Place Inspire Award - 2013-2014
- Finalist for the Think Award at a Qualifying Tournament

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Resources

Brevard Girl Scout Robotics

In 2010, the FIRST LEGO League team was started, then the FIRST Tech Challenge team was started after enough of the FIRST LEGO League team reached 9th grade. Three years ago, we added the FIRST LEGO League Jr team. These teams were created to:

- Serve the Girl Scouts by offering fun, technology-based enrichment activities and outreach with a focus on programs that are affordable and sustainable.
- To provide a platform allowing the girls to immerse in the engineering process

If you are interested in sponsoring our teams, we will display your Company's logo on our robot, display boards and on our website as well as in our pit area. Please provide a digital copy of your logo along with any sponsorship to:

Girl Scout Pathways
Attn: Caroline Achee
7905 Windover Way
Titusville, Florida 32780

For more information on FIRST and the B.E.E.s, C.A.K.E. B.A.T.T.E.R.S., and F.R.E.N.C.H. F.R.I.E.S., please visit the following links:

FIRST Links

- FIRST Website: <https://www.firstinspires.org>
- FIRST Tech Challenge Website: <https://www.firstinspires.org/robotics/ftc>
- FIRST LEGO League Website: <https://www.firstinspires.org/robotics/fll>
- FIRST LEGO League Jr Website: <https://www.firstinspires.org/robotics/flljr>
- FTC in Florida: <http://ftc.flfirst.org/>

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

- FLL in Florida: <https://firstfl.org/firstinfl/fll/> and <http://www.centralfloridarobotics.org/>
- FLL Jr in Florida: <https://firstfl.org/firstinfl/fll-jr/> and <http://www.centralfloridarobotics.org/>

B.E.E.s Links

- Team Website: <http://girlscouteverywhere.org/bees/>

Social Media

- Facebook: <https://www.facebook.com/groups/414996782209218/>
- Twitter: <https://twitter.com/fflJr4323BEEs>
- Instagram: <http://instagram.com/frenchfries>

C.A.K.E. B.A.T.T.E.R.S. Links

- Team Website: <http://girlscouteverywhere.org/cakebatters/>
- You Tube Channel: <http://www.youtube.com/user/cakebatters>

Social Media

- Facebook: <https://www.facebook.com/groups/1479542412290348/>
- Twitter: <https://twitter.com/FLL7407>
- Instagram: <http://instagram.com/cakebatter7407/>

F.R.E.N.C.H. F.R.I.E.S. Links

- Team Website: <http://girlscouteverywhere.org/frenchfries/>

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Social Media

- Facebook: <https://www.facebook.com/groups/754597614555375/>
- Twitter: https://twitter.com/FTC_FRENCHFRIES
- Instagram: <http://instagram.com/frenchfries>

Team Mission Statement

To inspire ourselves and others in learning more about the STEAM (Science, Technology, Engineering, Arts and Math) through activities like FIRST.

FIRST Description

The mission of FIRST is to inspire young people to be science and technology leaders, by engaging them in exciting mentor-based programs that build science, engineering and technology skills, that inspire innovation, and that foster well-rounded life capabilities including self-confidence, communication, and leadership.

Program Summary

The FIRST Tech Challenge (FTC) is more than a robotics program. We are a community focused on building a better world for tomorrow by engaging students in Science, Technology, Engineering and Math (STEM).

FTC is about teaching students the value of hard work, innovation and creativity. It goes beyond competition by teaching teenagers the importance of working together, sharing ideas and treating each other with respect and dignity.

Team Origin, Description and History

The F.R.E.N.C.H. F.R.I.E.S. (Friendly, Reliable, Energetic, Nerdy, Classy, Heros, proFessional, gRacious, engI-neering, tEamwork, FIRST) began in 2013 when the Minty Matrix team graduated from FLL where our last

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

project was featured in the “Inventors” magazine (November/December 2013). We are a Girl Scout team, and this is our fifth year. Currently we have 5 members from grades 7-12 who attend 3 different schools, where one is going into the veterinarian field, one in the teaching field and the others are undecided. Two of our members have graduated up from a FLL team.

In 2013 we attended the Florida State Championship as a rookie team and missed going to Super Regionals by one slot. Our robot was unique in that we had a wooden scissor lift.

In 2014 we missed attending the Florida State Championship by one slot. The team received/recognized for the following:

Tesla Robotics League Championship PTC Design Award (Robot Design)

Third place for the Inspire Award

A finalist for the Think Award.

In 2015 was the first year of the Space Coast League. The team received the following:

1st and 2nd League Qualifier Tournament Think Award

Space Coast League Championship Think Award

In 2016 was the second year of the Space Coast League. The team received the following:

3rd League Qualifier Tournament Think Award

Space Coast League Championship Connect Award

Space Coast League Championship Second Place Inspire Aware

Space Coast League Championship Third Place Think Award

In 2017 was the third year of the Space Coast League. The team received the following:

In 2018 was the fourth year of the Space Coast League. The team received the following:

Space Coast League Championship Control Award

Space Coast League Championship Second Place Inspire Award

Space Coast League Championship Third Place Think Award

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Team Organizational Structure

The Team is managed by our Girl Scout Leader and a parent of a former team member. In addition, there are two technical mentors who help the team on engineering principals and construction. Team members take on the responsibility of managing the Team, deciding on roles and completing project tasks. Everyone shares in the responsibility of fundraising. We make sure that we review and practice the following:

- Design, build, and program robots
- Apply real-world math and science concepts
- Develop problem-solving, collaboration, and team-building skills
- Build and become strong leaders
- Understand and practice Gracious Professionalism™
- Cooperate and compete in alliances at tournaments
- Compete for awards on and off the field
- Participate in a tiered competition that culminates at the FIRST Championship
- Qualify for millions of dollars in college scholarships
- Have fun!

Team Relationships

- 2018 Sponsors: Girl Scouts, Lockheed Martin, NASA GSDO, DoD, FPL, Rockwell Collins and our Mentor
- Girl Scouts provided a computer, robot parts, field mats and field perimeter
- Special Grant from a Former NASA Employee
- Lockheed Martin provided a competitive edge grant allowing for the purchase of robot parts, team activities, and tournament fees (\$1000)
- NASA GSDO provided a competitive edge grant allowing for the purchase of robot parts and funds for league registration (\$2028)
- NASA provided funds for FIRST Registration (\$275)
- DoD provided funds for FIRST Equipment (\$350)
- FPL provided funds (\$500)
- Rockwell Collins provided funds (\$500)
- Mentor supplies the wood and aluminum parts for the robot
- Host a FIRST Tech Challenge Meet

5.8.4 Team Impact and Goals

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

5.8.4.1 Team Use of Resources

Team F.R.E.N.C.H. F.R.I.E.S. has taken and will continue to take their robot to local and out of county events providing information about the FIRST Programs and our Team. We encourage children to drive the robot at the events so that it would start to get them thinking about “I would like to do that!”.

5.8.4.2 Team Future Plans

In 2018-2019, we plan on attending as many outreach events as possible sharing our knowledge of the FIRST Programs and encouraging more girls to participate in a FIRST program where they will learn a lot of different skills that they will use the rest of their lives. We show how the following are accomplished through sharing our Engineering Notebooks, earned awards, and life experiences:

- Design, build, and program robots
- Apply real-world math and science concepts
- Develop problem-solving, collaboration, and team-building skills
- Build and become strong leaders
- Understand and practice Gracious Professionalism™
- Cooperate and compete in alliances at tournaments
- Compete for awards on and off the field
- Participate in a tiered competition that culminates at the FIRST Championship
- Qualify for millions of dollars in college scholarships
- Have fun!

5.8.5 Sustainability

5.8.5.1 Team Action/Implementation Plan

The Team has identified the following actions for growth and sustainability

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Strategy	Actions	Responsibility	Planned
Prepare the Team to transition to JAVA	Teach the girls the difference between JAVA	Team Mentor	June 2019
Plan and implement STEAM Activity day	Get girls interested in STEAM careers	Team Members	April 2019
Work with the B.E.E.s and C.A.K.E.	Keep the girl interest in robots so they	Team Mem-	Continual

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

2016-2017 Budget		
Team 7341 - F.R.E.N.C.H. F.R.I.E.S.	Estimate	
Category	Expenses	Estimated Expenses Notes
FTC 2017-2018 Participation	\$775.00	
National Registration Fee	\$275.00	Payment to FIRST (Paid by another grant)
New Game	\$400.00	2017-2018 Challenge specific game elements
League Dues	\$100.00	League Dues and competition fees
Team Recognition (Marketing)	\$995.00	
Team Shirts	\$50.00	Purchase additional shirts for new members
Embroidering on shirts	\$100.00	Add team logo on shirts and create a banner
Banner		
Banner for sponsors	\$45.00	Create a banner for tournament
Competition “Give-A-Ways”	\$200.00	Pins, stickers or candy...
Display Board and supplies	\$100.00	Informational display board
Tablet for Display Board	\$200.00	Tablet to play informational videos
Outreach projects	\$300.00	Purchase items for our outreach activities
Robot	\$650.00	
Raw Materials	\$200.00	Raw Material to make robot parts
New Parts	\$450.00	New Motors/sensors/wheels/wires
Florida State Tournament	\$1,800.00	Florida - February 2018
Registration Fee	\$500.00	Florida FTC State Tournament
Hotel	\$800.00	4 rooms x 1 night (200.00)
Transportation	\$50.00	Two vehicles - gas
Food	\$450.00	10 people x \$45 (2 dinners and 1 lunch)
Total Budget - Reaching State	\$4,425.00	

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

2015-12016 Budget		
Category	Estimate	
	Income	Estimated Income Notes
Grants		
DoD	\$385.00	Purchase of FIRST Equipment
NASA GSDO	\$2028.00	Competitive Edge Grant
Lockheed Martin	\$1000.00	Competitive Edge Grant
Total	\$3313.00	

Team Fundraising Opportunities

Fundraiser Idea	Projected Income	Category	Notes
Panda Express	\$200.00	Fundraiser	One-time fundraiser
Village Inn	\$200.00	Fundraiser	One-time fundraiser
STEAM Event	\$200.00	Fundraiser	Set up to be an annual event for the Girl Scout in the area

Team Risk and Opportunity Analysis

We have outlined the concerns that might impact our current goals and strategies.

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Following is our Current Spending for all three teams:

Date	Income	Expenses	Team	Purchase
1/23/2019		\$25.65	7341	PURCHASE AUTHORIZED ON 01/23 HOBBYLOBBY 2540 S. WASHIN TITUSVILLE FL P00469024037322449
1/23/2019		\$17.00	7341	PURCHASE AUTHORIZED ON 01/22 REVROBOTICS 184-425-5226 TX S309022774647422
1/22/2019		\$24.91	7341	PURCHASE AUTHORIZED ON 01/19 SUBWAY 0311 TITUSVILLE FL S309019585447492
1/16/2019		\$75.00	7407	CHECK # 1069 Tournament Fee
1/14/2019		\$5.15	7341	PURCHASE AUTHORIZED ON 01/13 WM SUPERC Wal-Mart Sup TITUSVILLE FL P00000000182635046
1/11/2019		\$7.68	7341	PURCHASE AUTHORIZED ON 01/11 Wal-Mart Super Center TITUSVILLE FL P00000000672463222
1/11/2019		\$78.86	7341	PURCHASE AUTHORIZED ON 01/11 WAL-MART #0649 TI-TUSVILLE FL P00000000983252316
1/10/2019		-\$50.23	7407	PURCHASE AUTHORIZED ON 01/08 RIVER CITY BAR & G TI-TUSVILLE FL S589008654630781
1/9/2019		\$36.71	7341	PURCHASE AUTHORIZED ON 01/07 VILLAGE-INN-REST # TI-TUSVILLE FL S309007634651137
1/7/2019		\$58.69	7407	PURCHASE AUTHORIZED ON 01/04 WENDY'S #473 TI-TUSVILLE FL S469004622167322
1/4/2019		\$72.88	7407	PURCHASE AUTHORIZED ON 01/04 WALGREENS STORE 4600 S WA TITUSVILLE FL P00309004730590067
1/4/2019		\$52.22	7407	PURCHASE AUTHORIZED ON 01/02 MR SUBMARINE AND S TITUSVILLE FL S469002633150601
1/3/2019	\$1,050.00			MOBILE DEPOSIT : REF NUMBER :407030145594 Meet Money
1/3/2019	\$500.00			MOBILE DEPOSIT : REF NUMBER :807030146742 Grant Rockwell Collins
1/2/2019		\$17.24	7407	PURCHASE AUTHORIZED ON 01/01 Wal-Mart Super Center TITUSVILLE FL P00000000930251384
12/19/2018		\$275.00	7341	CHECK # 1068
12/19/2018		\$69.17	7341	PURCHASE AUTHORIZED ON 12/19 WM SUPERC Wal-Mart Sup TITUSVILLE FL P00000000779023414
12/17/2018	\$542.87		error	CARD FINAL CREDIT 11217187404
12/14/2018	\$1.00		error	CARD PROVISIONAL CREDIT 11210181145

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Date	Income	Expenses	Team	Purchase
10/29/2018		\$50.28	7341	PURCHASE AUTHORIZED ON 10/28 SP * BIGTOPSHIRTSH HTTPSMCS22.MY CO S308301622402631 CARD 3053
10/26/2018		\$111.11	7341	PURCHASE AUTHORIZED ON 10/25 REVROBOTICS 184-425- 5226 TX S588299017283806 CARD 3053
10/24/2018		\$50.42	7341	PURCHASE AUTHORIZED ON 10/22 RYANS PIZZA & PUB CO- COA FL S468295802640198 CARD 3053
10/24/2018		\$76.40	7341	PURCHASE AUTHORIZED ON 10/20 ONLINEMETALS.COM 206- -285-8603 WA S308293459495409 CARD 3053
10/22/2018		\$99.99	7341	PURCHASE AUTHORIZED ON 10/21 MICROSOFT *STORE MSBILL.INFO WA S388294626682974 CARD 3053
10/22/2018		\$55.00	7341	PURCHASE AUTHORIZED ON 10/21 ACE HDWE TITUSVILLE FL P00388294570003118 CARD 3053
10/22/2018	\$156.50		7407	MOBILE DEPOSIT : REF NUMBER :010210995678 Return of League Fees
10/15/2018		\$62.00	7341	PURCHASE AUTHORIZED ON 10/14 REVROBOTICS 184-425- 5226 TX S388287751253245 CARD 3053
10/15/2018		\$130.99	7341	PURCHASE AUTHORIZED ON 10/10 PITSCO INC SCO.COM KS S468283397565651 CARD 3053
10/1/2018		\$10.56	7341	PURCHASE AUTHORIZED ON 09/29 WEST MELBOURNE ACE MELBOURNE FL S388272541161034 CARD 3053
9/28/2018		\$14.00	All	MONTHLY SERVICE FEE
9/27/2018		\$275.00	7341	CHECK # 1067 League Fee
9/27/2018		\$146.33	7341	PURCHASE AUTHORIZED ON 09/26 SERVOCITY 620-221-0123 KS S388269661770995 CARD 3053
9/18/2018		\$150.00	7407	PURCHASE AUTHORIZED ON 09/17 WPY*Central Florid 855- 4693729 CA S308260453799052 CARD 3053 League Fees
9/7/2018		\$90.34	All	PURCHASE AUTHORIZED ON 09/07 WAL-MART #0649 TI- TUSVILLE FL P00000000473057770 CARD 3053
9/7/2018		\$48.91	7341	PURCHASE AUTHORIZED ON 09/04 PITSCO INC SCO.COM KS S308247411743416 CARD 3053
9/6/2018		\$95.93	7341	PURCHASE AUTHORIZED ON 09/05 ULINE *SHIP SUPPL 800- 295-5510 WI S468248455500555 CARD 3053
9/4/2018		\$44.32	4323	PURCHASE AUTHORIZED ON 09/02 WAL-MART #0649 TI- TUSVILLE FL P00000000934200992 CARD 3053
9/4/2018	\$311.00		7407	MOBILE DEPOSIT : REF NUMBER :713010881135 NASA Grant Money
8/31/2018		\$14.00	All	MONTHLY SERVICE FEE
8/29/2018	\$1,969.00			MOBILE DEPOSIT : REF NUMBER :107290057529 NASA Grant
8/29/2018	\$965.00			MOBILE DEPOSIT : REF NUMBER :107290057378 NASA Grant
8/27/2018		\$62.29	7341	PURCHASE AUTHORIZED ON 08/26 FLASHINGBLINKYLIGH 888-755-9449 CA S388238454575224 CARD 3053

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Date	Income	Expenses	Team	Purchase
8/27/2018		\$63.60	7341	PURCHASE AUTHORIZED ON 08/26 SP * BIGTOPSHIRTSH HTTPSMCS22.MY CO S468238450641029 CARD 3053
8/13/2018		\$148.75	All-KSC	PURCHASE AUTHORIZED ON 08/10 EB FIRST IN FLORID 8014137200 CA S588222544627250 CARD 3053
8/6/2018	\$297.50		error KSC	PURCHASE RETURN AUTHORIZED ON 08/03 EB FIRST IN FLORID 8014137200 CA S628216545605370 CARD 3053
8/6/2018	\$29.75		error KSC	PURCHASE RETURN AUTHORIZED ON 08/03 EB FIRST IN FLORID 8014137200 CA S628216545605369 CARD 3053
8/3/2018	\$311.95			MOBILE DEPOSIT : REF NUMBER :908030402657 Girl Scout Registration Fee
7/31/2018		\$14.00	All	MONTHLY SERVICE FEE
7/26/2018		\$327.25	error KSC	PURCHASE AUTHORIZED ON 07/25 EB FIRST IN FLORID 801- 413-7200 CA S308206708264376 CARD 3053
7/26/2018		\$233.65	7341	PURCHASE AUTHORIZED ON 07/23 PITSCO INC SCO.COM KS S588204395521688 CARD 3053
7/23/2018		\$35.04	7341	PURCHASE AUTHORIZED ON 07/22 Wal-Mart Super Center TITUSVILLE FL P0000000671292429 CARD 3053
7/16/2018	\$2,595.00			MOBILE DEPOSIT : REF NUMBER :609160019264 Special NASA Grant
7/13/2018		\$311.95	7407	PURCHASE AUTHORIZED ON 07/12 US FIRST 603-6663906 NH S468193685373867 CARD 3053
7/12/2018		\$641.47	7341	PURCHASE AUTHORIZED ON 07/09 ANDY MARK INC 765-868- 4779 IN S588190747950651 CARD 3053
Totals	\$10,072.76	\$5,754.49		

Signature : Jessica Anderson

Date: 1/27/2019

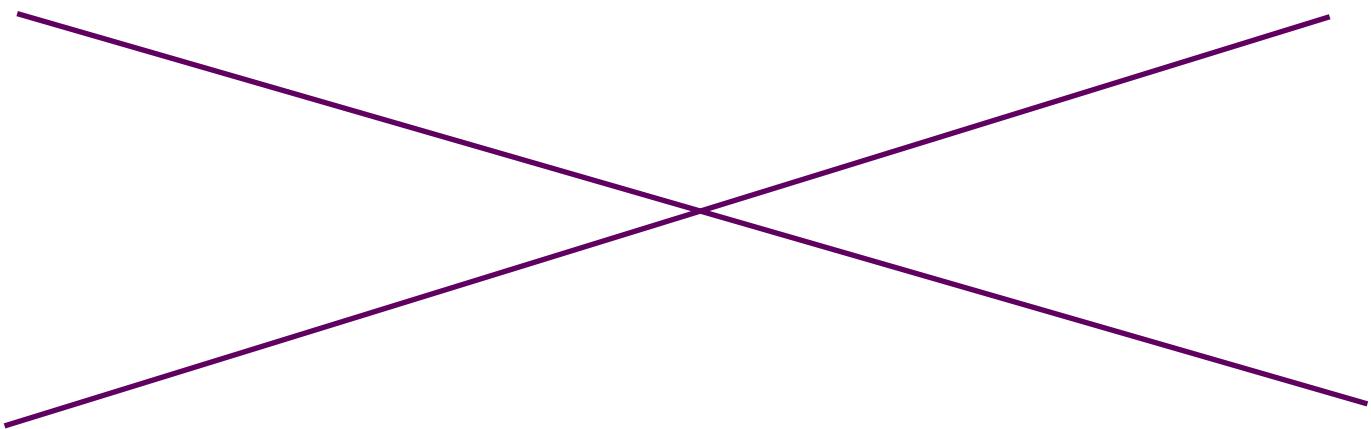
Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Following is the “Cost To Rebuild” the Robot if there is an accident where we could not recover:

Robot Bill of Material				
	Description	Cost	Quantity	Total
	Electronics			
1	REV Robotics Expansion Hub	\$175.00	2	\$350.00
2	USB Retention Mount and Cable	\$10.00	1	\$10.00
3	Gamepad controller	\$16.00	2	\$32.00
4	Touch Sensors	\$6.00	6	\$36.00
5	Cables 30cm	\$4.00	2	\$8.00
6	Cables 50cm	\$5.00	4	\$20.00
7	Cables 100cm	\$10.00	3	\$30.00
8	Conversion kits	\$25.00	8	\$200.00
9	Power Switch	\$6.00	1	\$6.00



Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Robot Bill of Material “cont’d.”				
	Description	Cost	Quantity	Total
	Hardware			
10	Battery	\$50.00	3	\$150.00
11	MAX Motor Shaft Hubs	\$5.95	4	\$23.80
12	MAX Axle Hubs	\$6.95	1	\$6.95
13	MAC Axle Set Collar	\$3.95	3	\$11.85
14	416 mm Channel	\$17.95	2	\$35.90
15	288 mm Channel (Package of 2)	\$16.95	5	\$84.75
16	160 mm Channel (Package of 2)	\$14.95	2	\$29.90
17	Inside C Connector (Package of 2)	\$4.95	10	\$49.50
18	Inside Corner Bracket (Package of 2)	\$4.95	3	\$14.85
19	MAX Flat Bracket	\$5.95	3	\$17.85
20	64 mm Flat (Package of 2)	\$4.95	3	\$14.85
21	Bronze Bushing (12)	\$13.95	1	\$13.95
22	40-Tooth Gear (Package of 2)	\$22.95	2	\$45.90
23	80-Tooth Gear	\$17.95	2	\$35.90
24	1/8" Nylon Axle Spacer	\$1.95	1	\$1.95
25	3/8" Nylon Axle Spacer	\$0.95	1	\$0.95
26	4" Tire/Wheel	\$9.95	4	\$39.80
27	4" Omni Wheel	\$19.99	2	\$39.98
28	Button Head Cap Screw (50)	\$9.75	1	\$9.75
29	Socket Head Cap Screws (25)	\$3.25	3	\$9.75
30	Kep Nuts (100)	\$3.25	2	\$6.50
31	Self-locking Nuts (100)	\$5.97	2	\$11.94
32	Axles	\$15.95	3	\$47.85

Signature : Jessica Anderson

Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

Our Business Plan “Continued”

Major companies used are: Robot Bill of Material "Cont'd."				
	Description	Cost	Quantity	Total
⇒ Re Robotics	33 Motor Mount	\$15.95	7	\$111.65
⇒ Andymark				
⇒ Servo City				
⇒ Online Metals	Motors			
34	Nerverest 40	\$26.00	2	\$52.00
35	Nerverest 60	\$26.00	4	\$104.00
36	TorchNATO	\$29.95	1	\$29.95
37	Encoder Cables	\$5.00	5	\$25.00
38	Polycarbonate Sheet	\$12.00	3	\$36.00
	Servo and Parts			
39	Standard Hub Shaft ServoBlock™ (24T Spline)	\$26.99	4	\$107.96
40	Continuous Servo	\$24.50	2	\$49.00
41	Standard Servo	\$17.59	2	\$35.18
42	X-rail - 12"	\$4.95	1	\$4.95
43	8mm Lead Screw - 12"	\$9.99	1	\$9.99
44	Lead Screw nut	\$7.99	1	\$7.99
45	8mm Bore Bottom Tapped	\$6.49	1	\$6.49
46	Clamping Shaft	\$4.99	1	\$4.99
47	1.50" Aluminum Channel	\$2.99	1	\$2.99
48	1/4" Bore, Face Thru-Hole Pillow Block	\$5.99	1	\$5.99
49	1/4" Stainless Steel Precision Shaft - 2"	\$0.89	1	\$0.89
50	Green Cable	\$2.99	2	\$5.98

Signature : Jessica Anderson

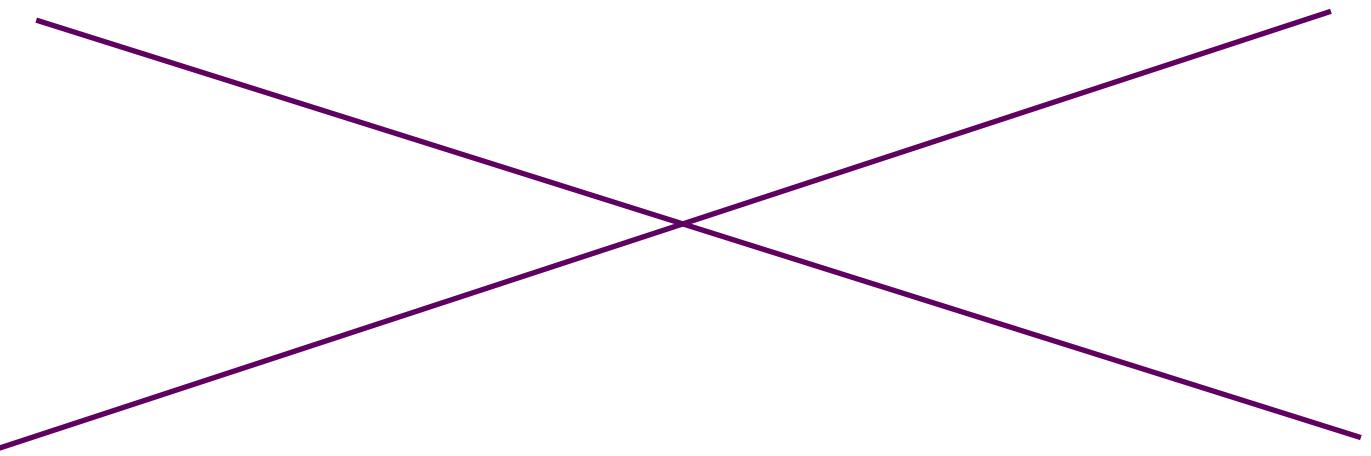
Date: 1/27/2019

Team 7341

F.R.E.N.C.H. F.R.I.E.S.

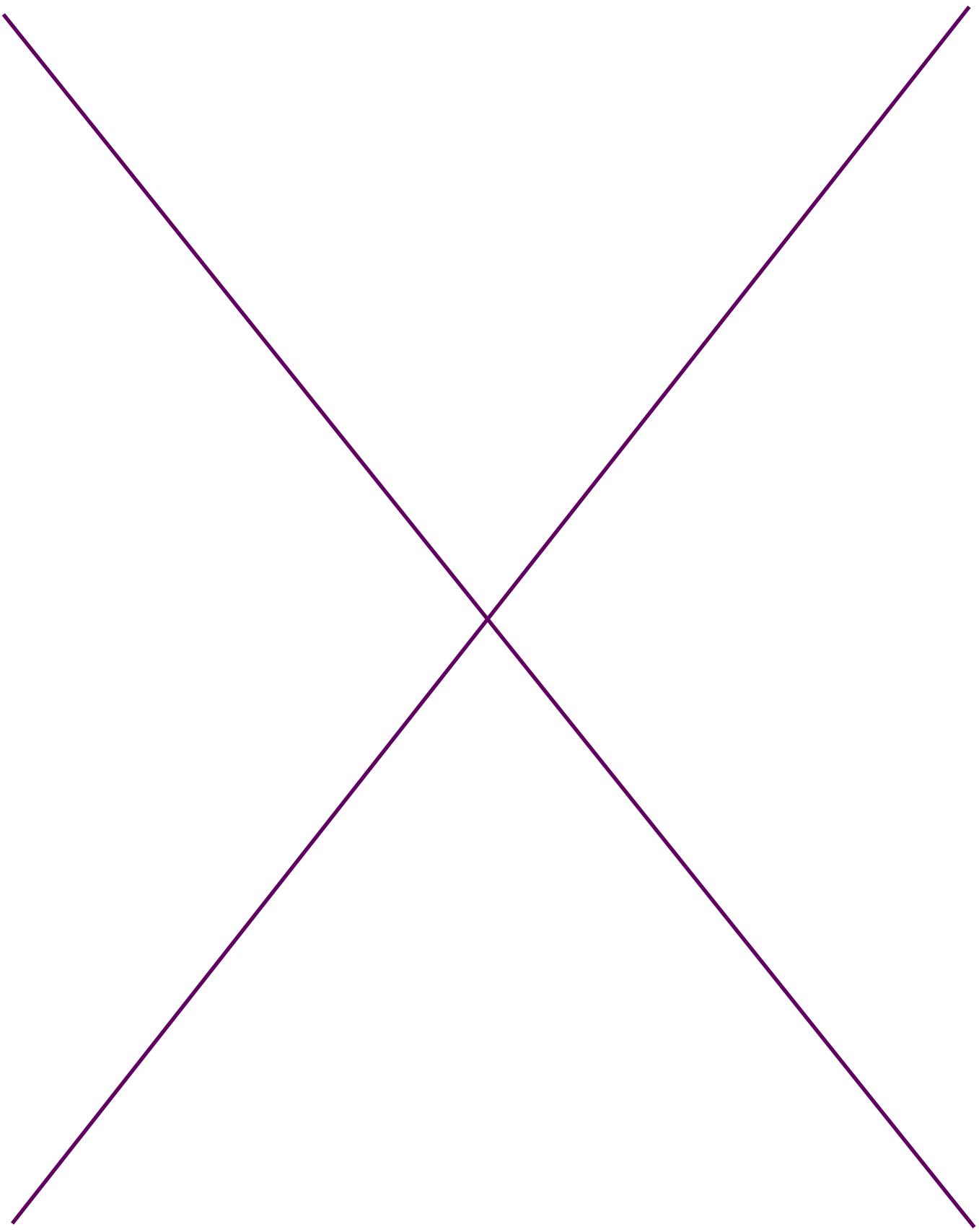
Our Business Plan “Continued”

Robot Bill of Material "Cont'd."				
	Description	Cost	Quantity	Total
	Special Supplies			\$0.00
51	Wood screws	\$8.00	1	\$8.00
52	Phone	\$229.99	2	\$459.98
53	Phone Cover	\$8.00	2	\$16.00
54	Raw Material for scissor bracket	\$15.00	1	\$15.00
55	Raw Material for the Marker tray	\$10.00	1	\$10.00
56	Raw Material for the End effector	\$10.00	1	\$10.00
57	Raw Material for the lift	\$20.00	1	\$20.00
58	Raw Material for the lift pulleys	\$40.00	1	\$40.00
59	Raw Material for 2 phone boxes	\$30.00	2	\$60.00
60	Raw Material for marker	\$10.00	2	\$20.00
61	Raw Material for converter plates	\$20.00	2	\$40.00
62	Raw Material for Robot phone boxes	\$30.00	2	\$60.00
	Total			\$2,756.40



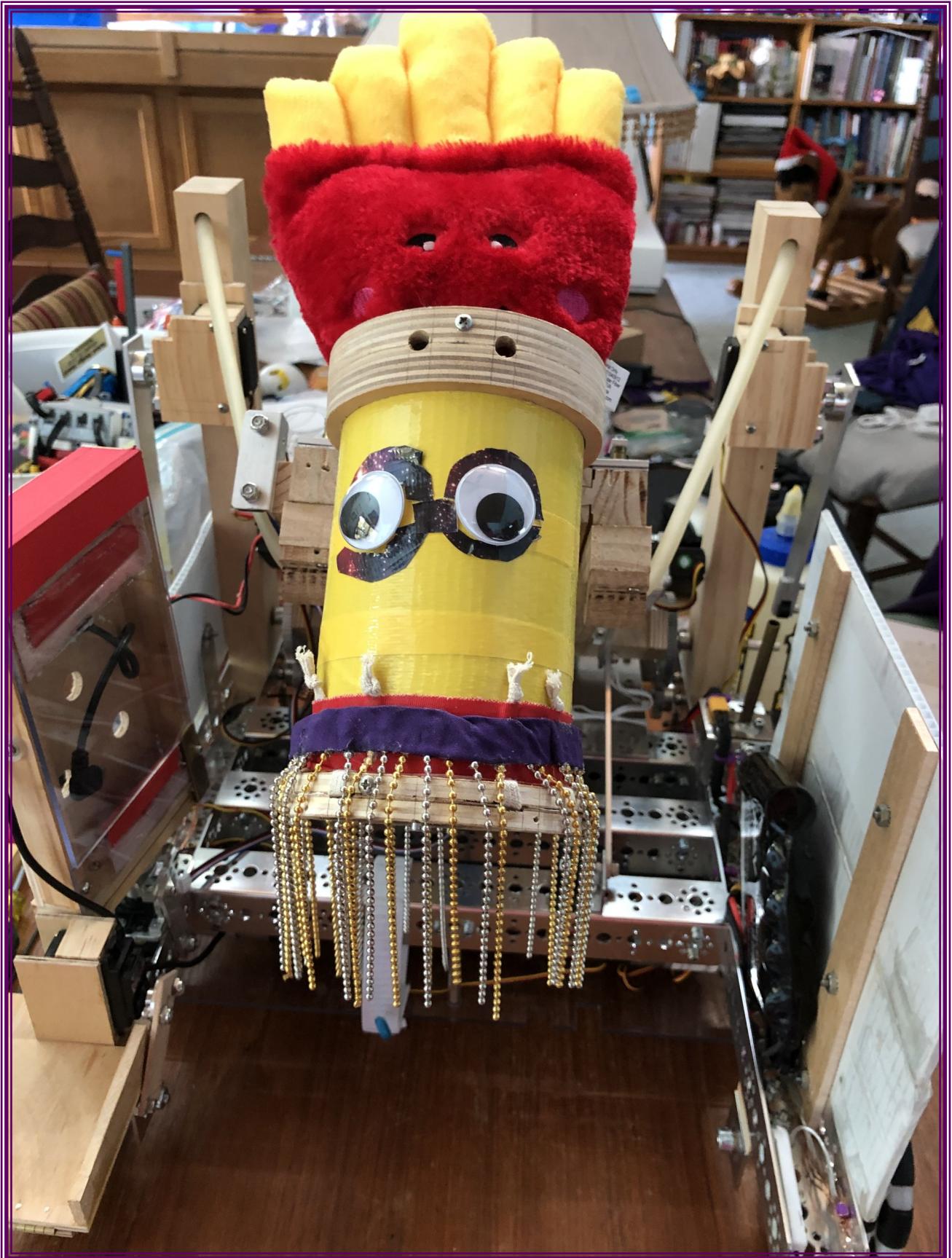
Signature : Jessica Anderson

Date: 1/27/2019



Signature : Jessica Anderson

Date: 1/27/2019



Signature : Jessica Anderson

Date: 1/27/2019