be INSPIRED:
an introductory

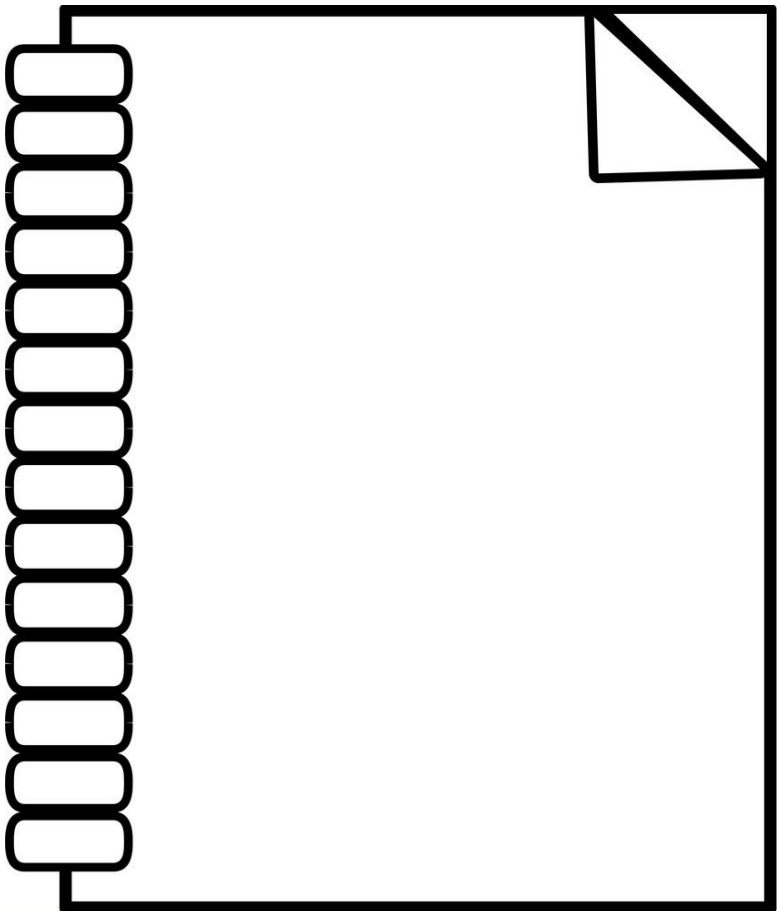
WORKBOOK

for new FIRST Tech Challenge teams

by:
QuadX
6299

table of **contents**

MARKETING **WORKBOOK**



Marketing for FTC Teams

Why would a robotics team need marketing? Members of the marketing team are responsible for presenting a cohesive image of your team to the community. They put a face to your team and allow you to be reached by other teams in FIRST, schools/businesses in your area, and other students interested in STEM and robotics. The marketing team is not only responsible for the image of the team but also oversees the upkeep of the engineering notebook, business plan, and outreach. The tasks of marketing include logo and website design, business/finances, outreach/community engagement, and advertising/social media management.

Marketing can be flashy but doesn't need to be. We are going to help you start your team with the necessary basics in a straight-forward manner that will allow you to be competitive with teams that have been in existence for years. The best teams have a thorough engineering notebook, a complete business plan, and a well thought out outreach program. Together we will learn about FIRST, making a business plan, securing community connections and support, putting in place a sustainability plan, and designing your outreach plan. After these are in place, we can work together to set up a team website and begin working on the flashier aspects of marketing.

Before we begin with learning about marketing and how it applies to FTC, you need to take care of a few small details, such as naming your team, designing a logo, and designing a mission statement. These can be done by a small group but is best when the entire team collaborates to make a name/logo that represents everyone.

Worksheet: What's in a Name?

One of the most important items to decide upon is a name for your team. Everyone will identify your team based on your name. Many teams use their school's mascot as a starting point for coming up with their name. Our program's name is based on our school mascot, the Viper. We are called ViperBots. Each team in the organization takes their team name from a different type of snake. We have Hydra, Copperhead, Venom, etc.

Now how will you name your team? Think about your mascots, things that important to your community, or interesting history that has happened in your region. All of these can be a great starting point when coming up with a name. Make sure that your name is specific enough not to be confused with other teams.

Write some ideas for names below along with the reasons behind using these as possible team names:

Name	Reason behind the name

Worksheet: Designing Your Logo

Your team's logo is what is used to visually recognize your team in competition and on the internet. It is a symbol, drawing, or graphic that represents not just your team but also gives others a look into what your team is about. This is especially important as you don't want your logo to be too similar to others but you don't want it to be so abstract that it is unrecognizable. Your logo also needs to be related in some way to your team name.

How does a team make their logo? They brainstorm and come up with an idea that can be represented by a sketch or a computer drawing. It is not necessary to have a computer to come up with an amazing logo. What is necessary is creativity and a formed idea of what you want to convey to others.

Draw some logo ideas in the boxes below and write a short sentence as to why is should be the logo. Be sure to relate it to your team name!

Logo Idea	Explanation

Logo Idea	Explanation

Worksheet: Choosing Your Mission Statement

Your team's mission statement guides not just what you do with your team but why and how. The mission statement is the single most important part of your business plan (to be discussed later) because it determines what your team is going to do. It is very important that your mission statement is well written, informative and detailed, but not too long. Here are a couple of examples of mission statements:

> *"Our mission is to develop well-rounded life skills for the future through embodying the core values of FIRST while promoting Science, Technology, Engineering, and Mathematics in both our local and global communities, as well as design and implement a robot that effectively and efficiently completes the given tasks of the FIRST Tech Challenge through the cooperation of our team members."*

> *"To design, build, and test a robot that can compete in the First Tech Challenge competition during which our team forms bonds and learns about the STEM field, all while being involved with FIRST and our community."*

When writing a mission statement, you need to have three components: Outreach, Robot, Team. Your first component needs to focus on how your will reach your community. What are your team goals, ideas, or desires for spreading STEM in your region? Write some ideas in the space below:

Your second component focus on the Robot. Although this isn't the most important part of FIRST, it is the most visible. As a team, you will need to come up with a way to state your goals of your robot design and competition strategy in one or two sentences. Give it a try!

The third component is the Team. How will you make sure that every person is involved and functioning as a member of your team? Now state it in a single sentence.

One thing to remember is that your mission statement will be different than any other team. Your team is special and has ideas and circumstances that will be shown in your mission statement. Don't try to emulate another team, be yourselves!

Business Plan

Your business plan explains the foundations of your team. It details your history, your present course, and your future aspirations. It includes demographics, budgets, goals, and outreach. This is done through a mission statement, team organization chart, recruitment plans, and a budget plan. Using these, one can track the progress a team has made over the years, see where the team originated from, and where it is going. The business plan helps the judges understand how your team works and sustains not just itself but also its impact.

There are multiple sections of the business plan. Each section is required for your team to be considered for awards. Here is a list of the necessary sections. We will discuss each in detail and then work to complete them together. Many teams will place a brief summary of FIRST and its beliefs either before the executive summary of in between the executive summary and the goals section.

- **EXECUTIVE SUMMARY**
- **GOALS**
- **CHALLENGES AND MITIGATION**
- **FINANCES**
- **RELATIONSHIPS**
- **OUTREACH**
- **TEAM CONTACT AND LOGOS**

Executive Summary:

The executive summary is possibly the single most important part of your business plan. It contains your mission statement, a list of team members and organization (can be visual or tabular), a statement of your team's vision for the future, an outreach summary, and basic team background information. The executive summary

needs to effectively relay team information and goals to the judges in a concise manner. This summary is only a few pages long but will be read by the judges. In fact, this is often the information that will entice judges to look deeper at your notebook and visit your team to gather additional information.

Here is a sample of an executive summary. We will break it down into its sections so you can begin building your own.

Our Mission

Our mission is to develop well-rounded life skills for the future through embodying the core values of FIRST while promoting Science, Technology, Engineering, and Mathematics in both our local and global communities, as well as design and implement a robot that effectively and efficiently completes the given tasks of the FIRST Tech Challenge through the cooperation of our team members.

Our Team

Rookie Year	2014
Location	Austin, TX
School Affiliation	Vandegrift High School (ViperBots)
Team Demographic	10 high school students: · 3 seniors, 3 juniors, 3 sophomores, 1 freshman · 3 rookies, 7 veterans
Mentors	Our mentors consist of current teachers, FRC participants, and team parents: · 4 FRC members · 3 faculty sponsors
Sponsors	Viper Nation Education Foundation, Leander ISD Educational Excellence Foundation, FIRST in Texas, Leander ISD, Arconic

Worksheet: Mission Statement and Team Information

Your team is going to need to come up with a mission statement and compile the background information of your team. This is pretty easy for your team as you should already have a mission statement completed. Also, because you are a new team you should be able to compile your team background information pretty quickly. The steps for a proper background chart are below.

STEP 1: Your team needs to write down when your team was started/rookie year, the location of your team, and your facilities. This section gives the judges a quick look at how much experience your team may have and gives them a place to begin when asking questions about your community involvement.

STEP 2: Your team needs to write out your team demographic (number of students on the team, their school year, and number of FTC rookies) as well as number of mentors your team has and their affiliation. This information can be used to help your team apply for grants, especially if your team is composed of ethnic minorities or has a majority of females.

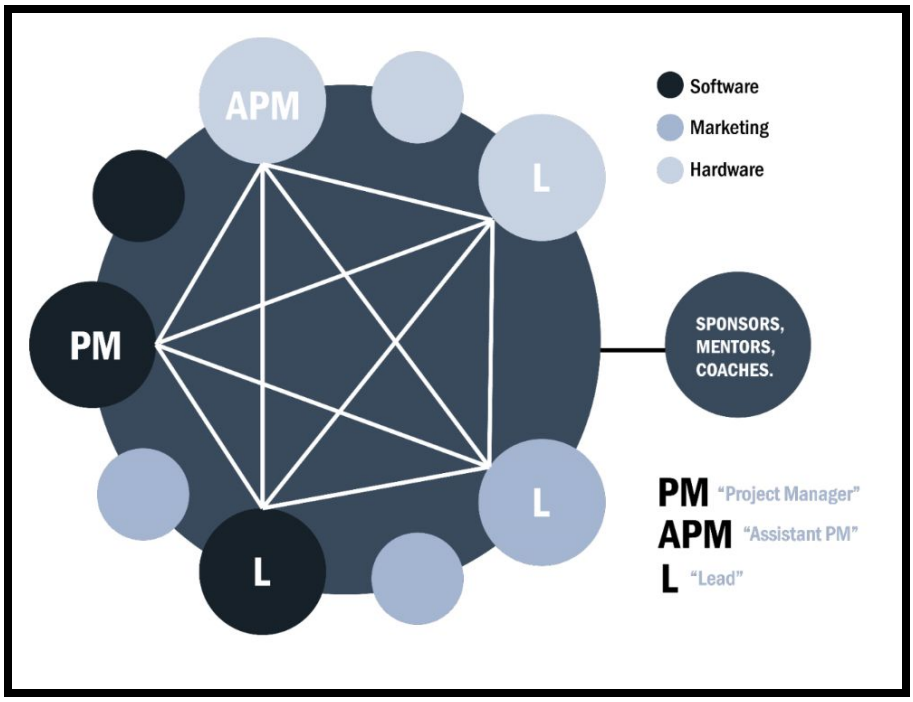
STEP 3. Your team needs to include a list of your sponsors. As part of your contract with sponsors, you need to acknowledge them in your business plan, on your robot, and in all of your team materials. This is done in order to retain sponsorships and to thank them for their support.

Now that your team knows how to write out a team background chart, it's time to practice! Below is a blank chart your team can use to fill out once you have all of the necessary information. Presently you may not have all of these sections, but we will help you to eventually fill them.

Starting Year	
Location	
Facilities	
Team Demographics	
Mentors	
Sponsors	

Worksheet: Team Structure

This is pretty self explanatory. It basically is a chart that shows the organization/leadership positions team members may have, what sub groups members are in, and each person's name. Here is a graphic example, excluding names:



However, most team structure charts are a simple list of team members with their role(s) written next to them.

Name	Job	Group
Becky	Project Manager	Hardware and Marketing
Elizabeth	Assistant Project Manager	Software
Henry	Software Lead	Software

In the space below, write down your team structure, including jobs/roles next to each person's name. If you would like, sketch a structure diagram to be used in your engineering notebook.

COMMUNITY IMPACT: Your community impact is what you have done to spread STEM and FIRST into the area your team lives in. The most important part of your community impact is how you brand it and how it ties together into a cohesive idea. This is especially important in judging as it allows for outreach events to be connected to each other while speaking, leading to a better presentation. In a around a paragraph, you will explain why your team made the impact they did and how. This is done usually later into the season after your team has had the chance to plan and complete outreach. An example is below:

> *"One of the aspects of FIRST we emphasize is community outreach. Because of this, we actively seek a plethora of outreach events in order to impact our community and spread awareness and enthusiasm for FIRST and STEM related careers. So far, we have held presentations at various summer camps in order to excite the youth of our community with science and technology."*

This section is something that grows as your team grows and develops. As a new team, your entry will be pretty simple. However, as you do more outreach over the season, you will need to expand this section.

Goals:

Your team's goals are very important to what your team strives to accomplish. They are broken into different types, such as team goals, subteam goals, season timeline, team schedule, and a weekly schedule. Here are brief descriptions of each type:

SEASON TIMELINE: Your season timeline is the events and outreach events your team has done throughout the season. This timeline will begin now, as this is the start of your season and will evolve as the season continues.

2.1 - Season Timeline

March 2018	• Team selection for 2018-2019 FTC season
May 2018	• Team announcement and rookies introduced to FIRST and FTC
August 2018	• ViperBots Rookie Camp (basic training for rookies)
September 08, 2018	• FTC Rover Ruckus Kickoff – Vandegrift HS
September 29, 2018	• Build Day 1
October 20, 2018	• Build Day 2
November 10, 2018	• AML League Play 1
December 08, 2018	• AML League Play 2
January 12, 2019	• AML League Play 3
January 19, 2019	• Build Day 3
February 09, 2019	• AML League Championships (Vista Ridge HS)
February 16, 2019	• Wildcard Qualifier
March 01, 2019	• Alamo Regional Championships
April 17, 2019	• Houston South World Championships

TEAM GOALS: Your team goals are the goals you have set for your team to complete not just that season but also for the future. Your goals are team chosen and can be as far-fetched or as simple as your team desires.

2.2 - Team Goals

Primary	• To **promote STEM skills** in all of our team members, and within those in our community. We want to do this in order to **improve our team** and our prospects of a **future career within the fields of STEM** . • To **attend the World Championship** in order to gain more time to **develop our skills on a higher level**. We hope that everyone in QuadX gets to experience and play an equal part of every aspect of our decision making process.
Secondary	1. **Maintain productivity** by **meeting deadlines** and keeping the team **organized**. 2. Constant **self-assessment**. At each Taco Tuesday Meeting, **review our weekly progress** and ask ourselves the following questions: a. "What did we accomplish?" b. "What is blocking our progress?" c. "What are we going to do this week?" 3. Effectively **communicate between subsections** and among **people who are not present** at meetings in order to **increase productivity.**

SUBTEAM GOALS: Your subteam goals apply to each subteam for the year and for the future. These goals are specific to each subteam and can vary for winning certain awards to organizing a certain amount of outreach events or finishing the autonomous in a week. This can be displayed as a table of information or as a timeline. Here is an example of Software subteam goals:

Software	1. Promote effective **problem-solving skills** in writing efficient programs. 2. Train our rookies as well as other rookie teams on how to use **GitHub** in order to facilitate an effective and efficient **coding process.** 3. Integrate more **advanced sensors** into the robot in order to have a less significant reliance upon internal calculations and instead **rely on external readings.** In addition, use more advanced **mathematical models** to gather accurate readings from our sensors. 4. Become proficient in programming the new system through **Android Studio** and the use of Java and **overcome limitations** that may arise from programming in Java as opposed to C.

TEAM SCHEDULE: The team schedule discusses a multitude of topics. It displays the number of hours the team works each week, the activities members of the team are involved in outside of robotics, and the subgroup meeting dates per week of competition. This is very important as it shows the judges your team is capable of efficiently budgeting and using time. It also allows judges to see the commitment the team has to FIRST and itself.

Hours	As a team within the larger ViperBots organization, our members are required to **log a minimum of 3 hours a week** to maintain eligibility for competitions; however, QuadX **encourages members to log 8 hours a week** in order to maximize our time and meet our goals. • Texas state law prohibits students from spending more than 8 hours during the school week on a single academic activity, so ViperBots are required to verify their attendance using a **biometric fingerprint scanner** to log in and out of the 'robot room' daily.
Extracurriculars	We pride ourselves on being a diverse team, made up of individuals from many backgrounds and who participate in a variety of activities. Part of the idea of the robotics family encourages all the variety that our team encompasses. On our team we have: • 1 football player • 2 band members • 1 swimmer • 1 person involved with DI (Destination Imagination) • 1 choir member • 1 Cyber Patriots Member • 3 HOSA members • 1 lacrosse member • 2 DECA members In order to help our team run more smoothly, we make sure to **coordinate team meetings with individual subgroups**.

WEEKLY SCHEDULE: The weekly schedule discusses the design meeting, marketing meeting, and software runs, as well as driver practice, that happens in season per week. They talk about why these meetings occur, what typically gets completed during these meetings, and how frequently they occur.

	MON	TUES	WED	THURS	FRI
AM	✘	ALL	SW	SW/MK	HW/PM
PM	ALL	HW	HW/MK	✘	SW/MK

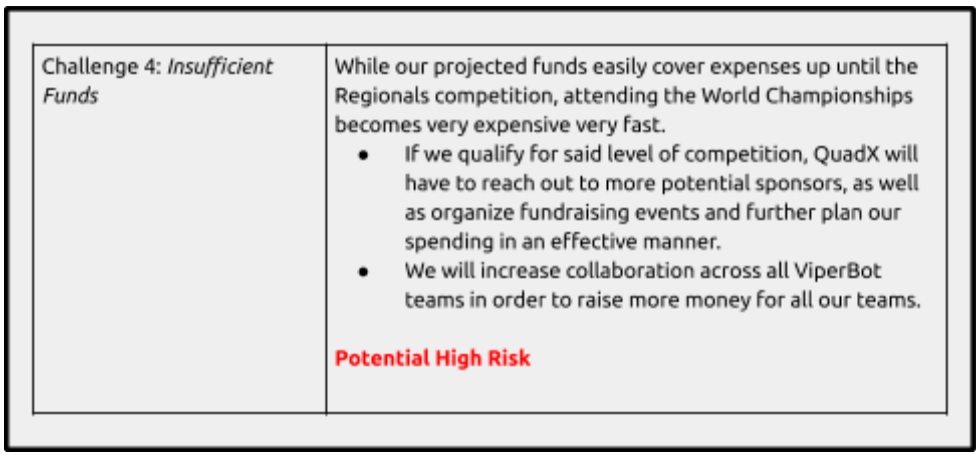
Challenges and Mitigation:

Your team needs to document not just your successes but also your challenges and how you overcame them. This includes anything from running out of snacks to losing a sponsorship. It is very important that documentation of the challenges faced includes why you had that issue and how your team solved it. It shows the judges your team is resilient and capable of facing challenges while continuing to grow as a team.

TRAINING: The first challenge is to find and attend training sessions in the off-season and during the actual competition season. You need to include documentation of the preseason training your team received. This includes any sort of workshop, training session, or camp your team may have attended. Make sure to include who organized the session, who was present , and what you learned. You can also include in school classes that might help with FTC robotics such as programming, marketing/website design, or CAD to just name a few. This shows the judges how your team has gained the skills you will use throughout your season and that you are willing to work with other FTC teams/companies to build connections and set up collaborative relationships.

This is usually done in a 1-2 paragraph statement in your business plan. Please make sure to title it Training or something similar.

CHALLENGES: Your team must explain how your team meets and discusses the challenges you face during the season. These are the major challenges that would prevent your team from being sustainable. Here is an example of a challenge and how to display it in your business plan:



Challenge 4: *Insufficient Funds*	While our projected funds easily cover expenses up until the Regionals competition, attending the World Championships becomes very expensive very fast. • If we qualify for said level of competition, QuadX will have to reach out to more potential sponsors, as well as organize fundraising events and further plan our spending in an effective manner. • We will increase collaboration across all ViperBot teams in order to raise more money for all our teams. **Potential High Risk**

You will need to include challenges that are faced by the team or subgroups (hardware/cad, software, and marketing). Make sure to define the challenge, explain what was decided upon, and then follow up with the solution and the process you took to overcome the challenge.

PROJECT MANAGEMENT: Project management is how your team defines tasks, assigns those tasks, and then makes sure that each task is completed according to the time schedule. For example, your team could use a daily standup meeting, also called a "scrum meeting" as a way to give a quick summary of jobs everyone is doing each day and to discuss any concerns. These meetings are very short (~15 minutes) . They allow everyone to feel connected and to keep people from getting overwhelmed or frustrated. This might not be practical for your team.

Most teams have a weekly program meeting that goes into more detail and allows collaboration between each of the team's sections. In a weekly program meeting each section goes over the week's problems and discuss them all together, helping your team be more aware of what's going on and how you all can fix it together. Also, teams use visual schedules to show timing of subsections of the project so that each group knows their due dates and can adjust their schedule as problems arise. Here is a visual example of a Gantt Chart that shows deadlines and time management.

Including this section in your business plan shows the judges that your team is actively working to stay updated and ahead of the problems you will face.

COMMUNICATIONS: The judges love to see active communication between students/mentors, so it's important that every member of the team is able to access up-to-date information. We used email and messaging apps to help us keep in touch. In this section of your business plan, include how your team maintained

- Google Classroom allows everyone to stay on top of what they are asked to do. The PM and the APM assign each team member "assignments", so that everyone knows what they are supposed to do, and when it is due.
- Slack provides a platform for everyone to speak in, while simultaneously allowing for each subteam to have their own thread to talk specifically about what they are doing.
- Google Hangouts, our main communication app, allows us to plan, bond, communicate various things, and to make decisions that require input from every team member. Google Hangouts helps allow all members of our team to have a voice.
- Google Drive allows us to share and organize our documents and files within one shared folder. This makes it easier for team members to find files specifically for their subteam, while simultaneously allowing all members easy information access to all files.

communications throughout your season and your team's structure for the development of solutions.

This section is generally a couple of paragraphs long and shows your methods for keeping in touch.

Worksheet: Program Management and Communications

Before we get into the details of writing your section of program management for your business plan, it's time to set up a platform that can be used to communicate with each other. Good communication is what makes a team successful. You must come up with a way where people can talk without it being expensive or time-consuming. Our team uses an app called SLACK to communicate. We also use Google Drive as a location to store all our team files. This way, no one has to give over a file, all members will have immediate access and are able to collaborate with one another. Google drive also has access to many free programs including Google Sheets, Google Docs, Google Slides, YouTube, and many more.

In the table below, enter all your team member names, their email addresses, and contact information. You will need this information to set up access to accounts. If each student doesn't have an email, we will help set them up so you can establish a good path to communication.

Name	Email	Phone Number	Address

Now that this is set up, it's time to try and figure out a good communication platform for your team. We will help you set up a team email now that will allow you to access the free Google Drive education software. This will be vital to your group organization and for allowing each student to access the files they need, be it software programs, cad images, or your team's business plan.

Team email address: _____

Team google password:_____

Possible software/APPS to use to communicate:

APP/Program Name	Purpose	How to Access

Plan for project management section of the Business Plan:

We want to meet _____ times per _____.

Each meeting will last _____ minutes.

Each member or team will do or present _____

_____.

We will communicate via _____.

All our programming will be completed using _____

_____.

All our CAD will be done using _____.

Our business plan will be written using _____.

Using the details above, begin to write the Program Management and Communication section of your business plan.

Finances:

Finance is the second most important part of your business plan but the most important item to ensure the continuation of your team. This is the section that the judges must see in order to take your team into consideration for any judged award. This covers your budget plan, how you raise money for your team, and how you team gains sponsorships. This section will help your team fundraise enough money each season to sustain yourselves if you include every bit of information regarding grants, sponsorships, and how to get money for your team. We will provide some information on how to get grants and sponsorships in the following pages.

You will also have a budget plan. This is required to qualify for any judged award at competition. Your budget plan needs to document every purchase, withdraw, deposit (sponsorships), and grant your team receives, giving an accurate display of your team's financial stability.

Expense	Total Cost
HARDWARE- $3,000	
Manufacturing Supplies	
ABS Plastic	70
PLA Plastic	200
Wood panels	50
Robot Supplies	
Motors/Servos	600
Build System	1,200
Parts/Supplies	400
Total Hardware Costs	**$2,520**
SOFTWARE	
Color sensor	75
Adafruit Gyroscope	35
Website Domain	45
Backup phones	80
Total Software Costs	**$235**
MARKETING- $300	
Pit Supplies	39.50
Total Marketing Costs	**$39.50**

Worksheet: Budget Plan

Your team is required to have a budget plan. This is necessary in order to qualify for any award in competition. Anything and everything your team buys throughout your season must be included in your budget. It is wise to have an expenses part of your budget (money used) and a sponsorships part of your budget (money gained). This way it is easier for the judges to track what your team has bought and see if certain events actually happened. It also helps them determine whether your team have the funds and budgeting skills to be sustainable.

In the following paragraphs we will explain step by step how to create your budget plan and maintain it through the years.

STEP 1: Predict your expenses for competition, travel, and food. this should be pretty easy as you just add the registration fees for all your competitions together and include approximate travel costs for your team to get to the competitions. You will also need to include how much you think the food will cost to feed your team along the way. Remember that costs are less if members can share rooms or if you can bring your own food instead of eating at restaurants.

Item	Description	Cost

STEP 2: Create a chart that has two sections. One is for money leaving your team (expenses), and the other is for money entering the team (deposits). This is important for later chart organization.

Deposits	Description	Amount
Expenses	Description	Amount

STEP 3: Add all travel, competition, and robot expenses into the expenses section. Organize it in any way you like, just make sure to include the names of the cash withdrawals and how much they cost.

STEP 4: Apply for grants. This is not as simple as it seems as there are many grants out there for your team to choose from. Your team qualifies for a vast majority of them, which is extremely beneficial for your team. When applying use your team name and number and ensure your team matches whatever requirements the grant is geared towards. Some possible grants your team could receive are minority ethnicity majority team, female majority, rookie team grant, etc. These grants are the lifeblood of some rookie teams so it's very important that your team makes sure to get as many as possible. Any grant you receive will be placed in the deposit section of your workbook. Please include the grant name in the description column and the value in the amount column.

STEP 5: Gain sponsorships. This will be discussed in more detail in the outreach section but your team is going to need to get sponsorships from businesses and universities in your community. These sponsorships will be necessary to fund and support your team financially. You need to contact any businesses and universities that your feel will be likely to want to sponsor your team. Then, your team needs to write up a letter or document on what your team does, what FIRST is and its purpose/mission, and why they should sponsor you. Talk to the people in charge in order to try and streamline the process. If you receive a sponsorships, please list the sponsor in the description section and the award into the amount column. Again, organize how your team likes but include the name of the sponsor and how much they gave your team. Also don't forget to send thank you cards to your sponsors.

STEP 6: Fundraise. This can be done in a variety of ways. You can have a bake sale, make goods for your team to sell, or offer a service to people to support your team. This

will be discussed more in the outreach section later. Add all fundraising expenses to the expenses section including a description and the amount. Also add all fundraising proceeds to the deposits section, just putting the name of the fundraiser in the description column and the cash gained in the amount column.

Relationships:

Your team needs to build relationships not just within the team but also within the community. This is shown within this small section. Your team is going to need to build relationships with the people who run where you are housed/located, nearby businesses/companies, and the people of your community. This is very important as if you don't make sure your team is forming these connections then you are missing out on a major opportunity for outreach and competitive success. The judges need to see evidence of connections formed within and outside of your community.

There are 3 main parts of this section that are pretty straightforward. They are listed below along with pictures of actual business plan examples.

FACILITIES/SCHOOL: This paragraph explains the facilities your team occupies, which is sometimes inside of a school. If so, your team needs to explain how you gained the ability to use school grounds for robotics. If not, your team needs to explain how you acquired the grounds you use for robotics.

COMMUNITY: This paragraph is all about the history of your robotics team/organization and its impact. Describe in detail the reasons why your team was started and how (if applicable) your robotics organization has made an impact on your community.

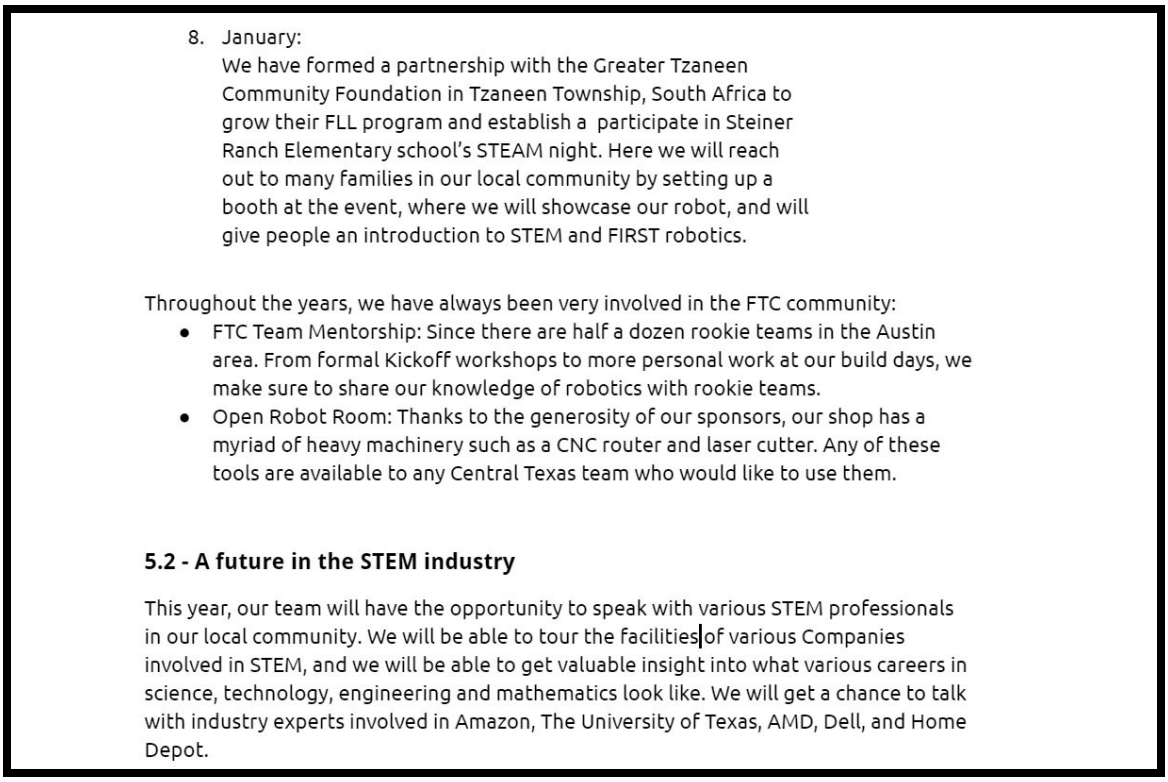
SPONSORS AND MENTORS: This list shows the community mentors, school mentors, and sponsors your team may have. Just separate this list into their respective parts and organize alphabetically.

Outreach:

Community outreach has a vast number of forms and varying ideas behind it. Every team's outreach is slightly different because of what they do at the outreach and why they schedule it. All of this is based upon a team's mission statement and their outreach goals for the year. Your team needs to include a list of your outreach events. These events need to have their impact written next to them along with a brief description of what actually happened. The judges like this format as it gives them the information they are looking for in a clean way.

Your team will need to include a future outreach/future plans paragraph in this section. This is necessary for your team to be competitive in any marketing award at competition.

Below is an example of part of this section. We will break it down into its parts so you can build it with your team.



8. January:
   We have formed a partnership with the Greater Tzaneen Community Foundation in Tzaneen Township, South Africa to grow their FLL program and establish a participate in Steiner Ranch Elementary school's STEAM night. Here we will reach out to many families in our local community by setting up a booth at the event, where we will showcase our robot, and will give people an introduction to STEM and FIRST robotics.

Throughout the years, we have always been very involved in the FTC community:

- FTC Team Mentorship: Since there are half a dozen rookie teams in the Austin area. From formal Kickoff workshops to more personal work at our build days, we make sure to share our knowledge of robotics with rookie teams.
- Open Robot Room: Thanks to the generosity of our sponsors, our shop has a myriad of heavy machinery such as a CNC router and laser cutter. Any of these tools are available to any Central Texas team who would like to use them.

5.2 - A future in the STEM industry

This year, our team will have the opportunity to speak with various STEM professionals in our local community. We will be able to tour the facilities of various Companies involved in STEM, and we will be able to get valuable insight into what various careers in science, technology, engineering and mathematics look like. We will get a chance to talk with industry experts involved in Amazon, The University of Texas, AMD, Dell, and Home Depot.

Worksheet: Making your Outreach Event List

Your team will need to have an outreach list for your business plan that discusses each event in brief detail. This page is full of information. It displays the events your team has completed, their impact, a brief description of each event, and the goals and reason behind doing events in this type of outreach. Let's go through how to make a concise and well written outreach event list together.

STEP 1: Your team needs to take the outreach events you have done and organize them by month. All events done in January need to be grouped together and so on. This makes it easier when going through and writing descriptions about events.

STEP 2: Reach out to where you had the outreach. Try to get in contact with the people who ran the event or organized it in order to figure out the impact of your outreach. If you cannot get in contact with a person then gauge the impact as a team. DO NOT make your outreach seem to have a much bigger impact that it actually did, try to be realistic.

STEP 3: Write the impact and descriptions next to the events. Make sure the keep them in order.

STEP 4: Write in the dates next to the events. This is optional but can look really cool if done correctly.

STEP 5: Write a paragraph at the end that wraps up and summarizes everything your team has done this year. Make it simple and sweet and don't overcomplicate.

Team Contact and Logos:

The final section in your business plan locks in how to contact your team and recognize it online. It is the shortest section. In this section you only include two things: your team contact information and your logo/any logo variations.

LOGOS: This is a couple of variants of your logo and a picture of your team's actual logo

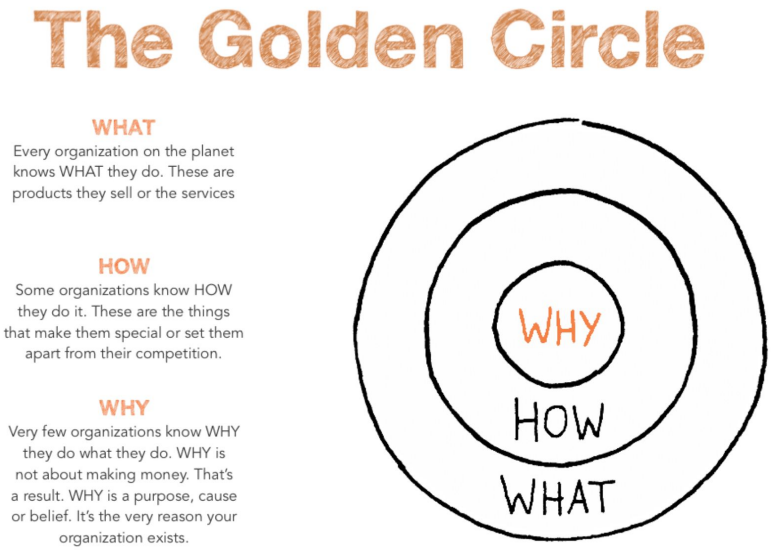
CONTACTS: This is a list of the ways people can reach your team. Some include social media, email, and website URL.

Outreach

Outreach is something incredibly important that the marketing team does year round. It is uncommon for teams to organize outreach events only during their robotics season as it is beneficial for teams to do it during and throughout their offseason as well. Outreach is an umbrella term, meaning it is used as a general term for a diverse variety of events/opportunities. This is kind of hard to understand without experiencing it, but basically outreach is helping the community. Your team's outreach should be chosen with several things in mind:

- what your mission statement is
- how you want to impact your community
- what companies are in your community/area
- what resources you have available
- how many team members you are available

These things are vital to keep in mind as they can help your team determine what you will be able to do. This is just the first step as your team must identify the main reason behind WHY you are doing this type of outreach and WHY you have chosen this way to impact your community.



Worksheet: Designing Outreach

Outreach is how you connect with your community, be it the students and schools or corporations. It is how you give back to the people who have shaped your personality and childhood. This allows you to make a bond with the people and businesses in your community. Your team will have to complete two main types of outreach: corporate and community.

Community Outreach:

The first type of outreach is community outreach. This is any sort of connection you form with your community in order to spread STEM and robotics. Some examples include teaching science experiments to younger kids (see section following this worksheet), having a STEM/science booth or experiment at a communal gathering, or even just volunteering at an event. These are pretty easy to do as they require minimal supplies and are fun to do.

STEP 1: Determine what types of outreach you and your team are interested in completing. In the space below write out some ideas you have for community outreach:

Idea	Possible Location	Time Duration/ Dates of Interest	Needs for Events

STEP 2: Now it's time to set up your outreach. The easiest way to do this is the go talk to the people in charge of the event/facility/class. Getting their permission to actually have the outreach event is the safest way to get outreach. Once you acquire permission, make sure to come up with the supplies you need. Always be sure your budget is able to pay for these items as you do not want your team budget going negative.

STEP 3: Now comes the fun part: planning the event. You need to have the manpower for it. Create a signup sheet for the times needed and assign students different slots. This ensures that part of the team will always be present for the outreach event. If you don't have enough people to help in your organization, don't be afraid to ask others to join you. Collaboration is a great tool that only makes your outreach better. You also need to make sure that everyone has a way to the event. Don't hesitate to ask for help in planning. It takes time and practice to make an event a success. Your outreach partners are a great resource as well as other FTC teams.

Always make sure that the your community knows about the event. The best way to do this is to reach out to local newspapers, radio stations, TV stations, or leaders in the community. We often talk to school officials and let them know that we are having an event. You can also post an announcement on social media. Flyers, posted around the community, are a great tool as well. It's important that you always get permission before posting any posters.

STEP 4: Follow up after the event: Make sure to take lots of pictures of the people at the event. You may need to get a waiver from attendees as some people don't want their picture taken. This is ok and you should never try and force a picture. You will have many opportunities to get pictures and it's always best to have pictures showing happy people having fun. Also, make a feedback form that people can fill out from your event. This will allow you to make changes and improve how you reach your community.

Company or Corporate Outreach:

Company or Corporate outreach is when a team reaches out to companies in their area (preferably STEM companies) and creates a connection with them. This is important because FIRST relies on mentors and coaches that come from the corporate sector. By connecting with companies, you are helping spread FIRST and allowing your team to learn about new fields/topics and make new relationships.

STEP 1: Brainstorm possible companies that can help you and that would be helped by learning about FIRST and your team. These connections are a great way to find sponsors and mentors.

Company Name	Scope of Company	Location	Contact Information
EX. Advanced Micro Devices	Semiconductors	5204 E Ben White Blvd, Austin, TX ·	(512) 602-1000

STEP 2: Reach out to the companies and connect. Call the companies or visit in person, if this is an option. Email is also very effective if you have a contact inside the company. If not, the email may get discarded and you will need to reach out in another manner. Make sure to have a FIRST pamphlet ready, and present it with a smile.

STEP 3: Set up a meeting with the company. There are many ways that you can meet

with people from the company. Here are some examples:

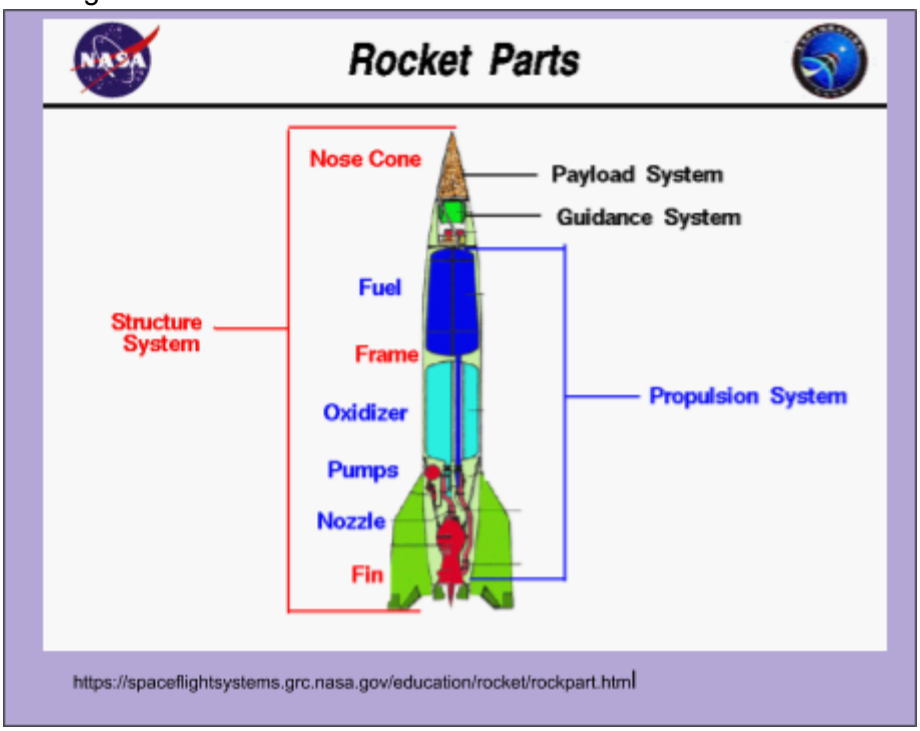
- tour their company and then present your robot and FIRST to employees
- learning about what STEM careers are available by meeting at your venue
- asking a company for sponsorship
- gaining company mentors for your team

STEP 4: Continue your relationship with the company. Invite them to competitions or your other outreach. It's great to let them know that you appreciate their time and it helps strengthen bonds in your FIRST family.

Paper Straw Rockets:

Purpose: To have children investigate how to build a rocket and what is needed to make it fly in a stable manner.

Background: Show the students the model rocket and explain the parts of the rocket and their purpose. You can explain each in as much detail as you want depending on the audience. If you are working with very young children, just draw the rocket on the blackboard and label the body tube (frame), nose cone, and the engines. Also, include the fins after discussing the original drawing.



Frame- the outside of the rocket that holds the rocket together and gives it strength. This is made of many strong materials that are light but can take high pressures and temperatures. Scientists and engineers work hard to make new materials. This is called material engineering. It's an exciting field in technology.

Nose Cone- top of the rocket that allows the rocket to move through the air. The shape is very important so the rocket can fly straight and not get ripped apart when cutting through the air. If the shape isn't right, turbulence (crazy air pockets) can set up and rip the rocket apart.

Engine-the part of the rocket that allows it to lift into the air. These are very strong chemical engines that burn at high temperatures and release huge amounts of energy. You need to have a fuel that burns in these engines to get the rocket to fly. There are many types of fuels including solid rocket fuel, liquid fuels like liquid hydrogen, or even nuclear energy. Scientists

are always trying to create a better fuel and engine.

For more detailed information for older students, refer them to **https://spaceflightsystems.grc.nasa.gov/education/rocket/rockpart.html.**

Now, explain the rockets that will be made today.

Our rockets are very simple. The structure of our rocket will be made from paper. The nose shape will be determined by you. It can be square, round, or pointed. You get to decide. The size and shapes of the fins will be made by you. You can have fins or not. The propulsion system is a straw. You will determine how much energy the rocket has by how hard you blow.

Age Group: This experiment is good for students of all ages. If doing the experiment with very young children, you will need to help with the rolling of the rocket and the cutting of the fins. Also they might need help attaching the fins. We have taught this experiment to children ranging in age from 4 to 15.

Setup: You will need to make a couple of examples of the rockets and make some models of fins so the students will have a template. Depending on the time, you can pre-cut the pieces of paper for the students so they can immediately start making rockets.

Needed Supplies:

Straws (bendable straws work well)
Paper (can use printer, construction, or scrapbook), cut into 5" x 1 ½" strips and 5" x 3" strips
Markers
Tape
Pencils
Scissors

Procedure:

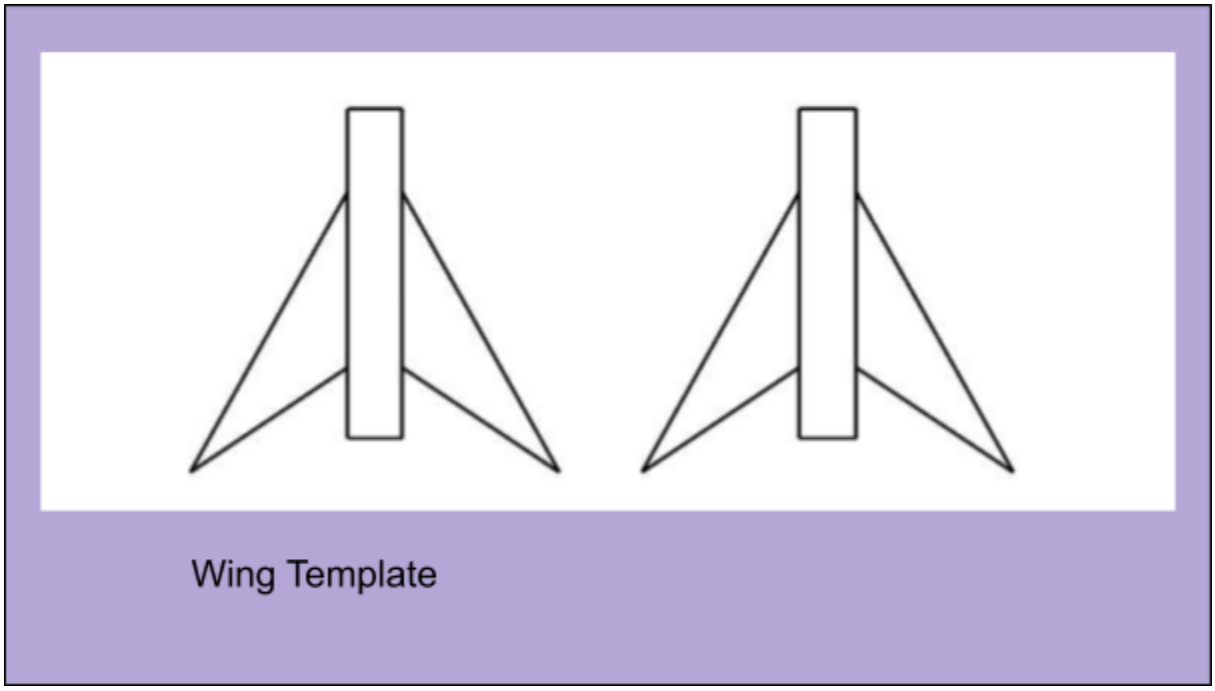
Step 1: Hand out straws to each student.

Step 2: Hand out white paper to each student. Please make sure that each gets at least two different lengths.

Step 3: Allow the students to decorate their rockets with the markers. **Only the 1" wide strip of paper will be visible once the rocket is rolled. Please let them know they only need to decorate that little bit of paper.**

Step 4: Roll the paper around a pencil, beginning with the non-decorated edge of the paper. Tape the edge of the paper down when you are finished rolling. Remove the pencil.

Step 5: Seal the top of the rocket by placing tape over the open top. The students can also use a sharpened pencil to twist the paper around to make the nose cone.



Step 6: Have the students make both rockets (the 5" x 1 ½" and the 5" x 3").

Step 7: Have the students place one rocket on the straw and blow. It will launch into the sky. Repeat for both rockets. Which flew better? Which went higher? Have the students answer the questions.

Step 8 (optional): Make wings using the template, if they choose. It would be great to make one with wings and one without. Trace and cut out. Fold on the dotted lines. Attach the wings with a single strip of tape.

Step 9: Compare with wings and without. Help students to understand why the wings help with flight.

Cost per student:

paper……..$0.05
Scotch tape ……$0.10
Straw ……$0.01

$0.16

Conclusions: Students like to see how the rockets flight changes when they add wings or make a nose cone. This experiment is fun for them because they learn about the parts of a rocket, how to make a rocket, and then what makes a rocket fly better. They also get to take a rocket home with them and are very likely to tell someone about the experiment and maybe even do it with another person. Great way to spread STEM.

References:

NASA/JPL Straw Rockets http://www.jpl.nasa.gov/edu/teach/activity/straw-rocket/

Engineering Notebook

An engineering notebook is just as, if not more, important than a robot. This is because the engineering notebook is required for a team to even be considered for any advancing award. A majority of teams advance to the next level of competition by getting a judging award, therefore we highly recommended that teams create and submit an engineering notebook to maximize their chances of advancement.

It shows the journey and evolution of not just a team's robot and outreach, but the team itself. This makes it not exclusively a marketing project, but instead, a task run by marketing but participated in by the entire team. The engineering notebook is the single most important marketing task a team does each season. The quality of the content of the notebook can mean the difference between not getting an award and obtaining an award and possibly advancing to the next level of competition. This means that it must be of excellent quality and bursting with rich information about your team.

A good engineering notebook is organized well and tells a well-rounded story of how a team has grown throughout the season. Your team's entries need to be filled with relevant information. It is quite important to have pictures, sketches, and plans for and of your robot included in your engineering notebook as these, along with any math and physics used to solve engineering problems, are they are what the judges look for when judging most of the awards. Pictures are particularly are very important, as they provide proof of what you claim happened. Anything and everything that helps tells the story of your team is vital for your EN.

We highly recommend including the following in your notebook:

- Team goals for that period of time
- Attendance and date(s)
- Design iteration sketches and math

- Obstacles you faced and how you overcame them
- Descriptions of changes you made to your hardware and/or software
    - Why you included these changes
    - How did you make these changes
    - The pros and cons of these changes
- Descriptions of outreach events
    - What the event was
    - What impact you had by attending this
    - What you learned from the experience
- Pictures from outreach events
- Pictures of design iterations
- Pictures of software paths and code
- Things you plan to do in the future and why
- Anything else that you feel would help the judges understand what, how and why you did what you did throughout the season

FORMAT:

A team can choose to format and make their engineering notebook in a variety of ways. Some teams use written engineering notebooks, while others use online engineering notebooks. When writing your engineering notebook entries, you should have a previously planned format you follow. We highly suggest your team uses a weekly log format, but you may do whatever you would like. (I.e, Daily, Bi-weekly, etc) Ask your team what type of log schedule they would like and what type of engineering notebook you want to pursue.

LAYOUT:

The layout of your engineering notebook is different for everyone. Some teams prefer to use a layout which highlights what happened that meeting while others focus on the

goals achieved and how efficiency could have been improved. Some layouts are most robot focused while others focus evenly on the subgroups. This is completely your team's choice and must be discussed with your leadership and marketing team.

JUDGING SHEET:

The judging sheet is a single piece of paper that goes on the inside cover of your engineering notebook. This sheet is **VERY IMPORTANT**. It is responsible for helping the judges find where information regarding specific awards can be found. It also introduces your team and explains your achievements and your mission for the season. This is probably one of the most important things related to the engineering notebook.

TABS:

Before submitting their engineering notebook, teams mark what pages they want the judges to read, often colour coding the tabs based on the awards that they want the judges to consider them for. This is known as tabbing.

Teams put a description and key for their tabbing system somewhere in the summary page/judging sheet of the engineering notebook so that judges know exactly what they are looking for when they are observing the team's notebook.

It is important that teams take the tabbing very seriously and make sure to tab the pages of their engineering notebook that they feel are most relevant to the awards they are going for, as what the judges see from the engineering notebook plays a big part in the decision to determine what teams get judges awards.

AWARDS:

The following are the different Judged awards. Write your EN entries with the descriptions of each in mind:

- Inspire Award
  - This is the best overall team
  - They perform well in the robot game, did well in formal and Pit judging, and had an excellent engineering notebook
  - This team is well-rounded, and is a good contender for most if not all of the other judging awards. This is reflected in their Engineering notebook.
  - “high quality, thoughtful, thorough, detailed, and well organized.”
- Think Award
  - This team is recognized for having the best engineering Notebook.
  - Their notebook connects with the judges and allows them to see the growth of the team through the season
  - This team has a solid understanding of the engineering design Process
  - There are many pictures, drawings, and descriptions detailing their entire journey
  - Includes underlying science, mathematics, and game strategies.
- Connect Award - Corporate Relationships
  - Identifies their future goals and the steps they will take to reach those goals. (ex: fundraising goals, sustainability goals, timelines, etc).
- Rockwell Collins Innovate Award - Most Innovative Solution
  - Document their design process and how the team arrived at their solution.

- Design Award - Best Designed Robot
    - Engineering Section with detailed Robot design drawings.
- Motivate Award - Community Relationships/Connections
    - Identifies their future goals and the steps they will take to reach those goals. (ex: fundraising goals, sustainability goals, timelines, etc).
- Control Award - Best Autonomous Period/Code
    - Must include an Engineering Section that documents the control components.

FINAL NOTES:

1. The team may design the front cover of their engineering notebook however they wish, however their team name and number must be on it.
2. Teams divide their engineering notebooks in various ways. How the information is organized varies from team to team. One way that the engineering notebook is divided is by including the following sections:
    a. An engineering section that includes the Robot design processes.
    b. A team section that includes information about the team and outreach activities.
    c. A business, strategic, or sustainability plan that includes information about how the team is structured along with what they plan to do financially. *Including a business plan is required

    Remember that these sections vary in between each team.
3. Teams may submit no more than two notebooks to the judges per competition. This means that teams can submit their engineering notebook along with one other notebook

Below are some examples of engineering notebook pages.

Date: _____ Event Title:_____

Attendance:

Present: _____

_____

Absent:_____

To Do List:

CAD	
Hardware	
Software	
Marketing	
Community Outreach	

11503 HyperFang

Team Meeting 07/7/17

Sunday/ July 7th, 2017 / Chris's House

Attendance:

Kelly, Kendall, Chris, Phil, Daniel, Shawn

Meeting:

We summarized our strategy for the connect award, went over our some of our outreach events, and how we will contact our outreach targets:

- Kelly will be announcing leads on July 10
- We have a google outreach event in January
- Kelly delegated a rookie camp task to several team members:
- Our strategy for the connect award is quantity over quality: Having more small connections is better than just a few strong connections
- Kendall showed us this spreadsheet of the outreach targets that she's emailed

ORGANIZATION	CONTACT	EMAIL 1?	EMAIL 2?	DATE SET
LT Library		done		
Capital food bank		done		
Central Texas food bank	volunteer@centraltexasfoodbank.org	done		
teachers				
dell				

Week of September 25th

Goals

- Get the trollbot running and help software begin auto testing
- Finish the CAD
- Complete the collector prototype

Problems

- The holes in the tubing noodles were too small so we had to enlarge them.
- Both of our prints of the tubing noodle failed.

Accomplishments

- We completed the Tubing Noodle *(See Figure 1)* for manipulator which will hold the snaking surgical tubing and allows us to collect both types minerals out of the crater. (KB)
- We completed the Team Marker Deployer Design *(See Figure 1)*. It consists of a servo that will be attached to the back of the robot, and a lever that the team marker will slide off of. (HR & SC)
- Base CAD Completed
- We made the counterbalance calculations for the counterbalance weight and applied torque *(See Figure 2)*.
- We tested if the manipulator was able collect and deposit the balls and blocks for Tele-op *(See Figure 3)*.
- We completed the Phone Mount *(See Figure 4)*. The design was the base with four sliding pieces on the back of it. (HR & SC)

Meeting notes

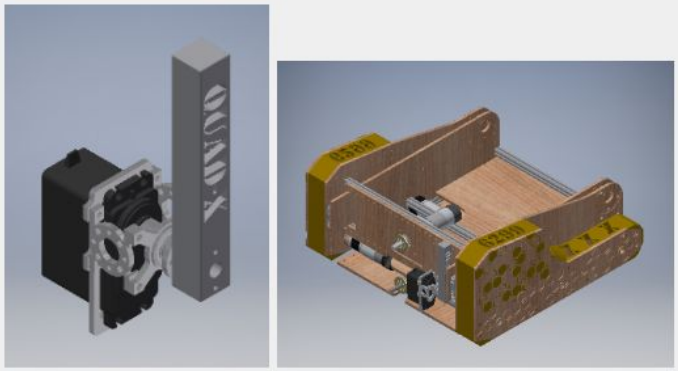


Figure 1:
The Team Marker Deployment Assembly, and its placement on the rear of the base.

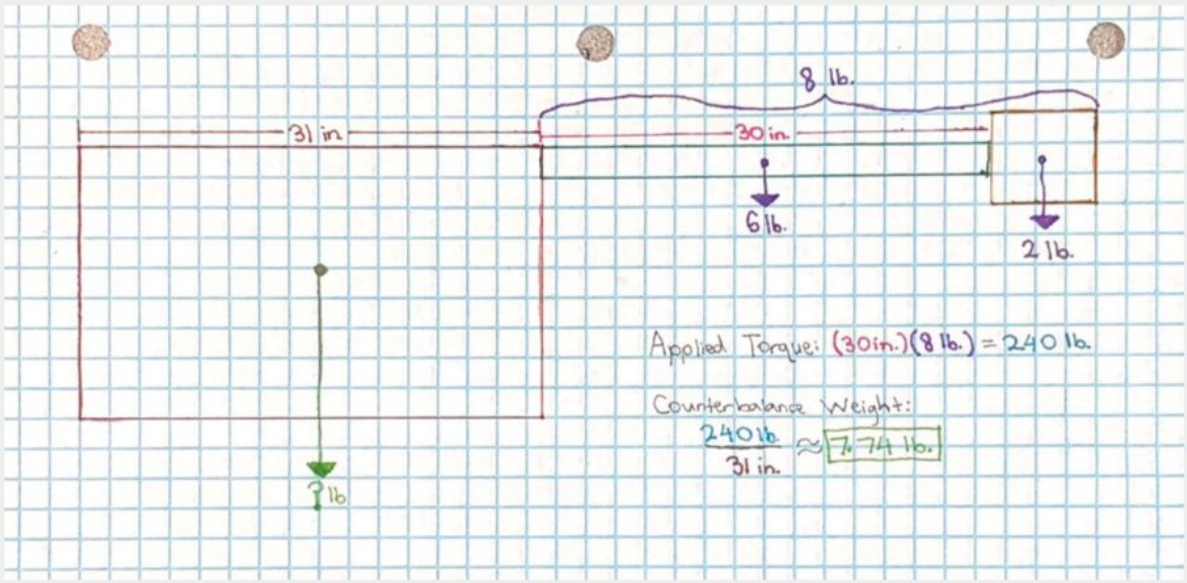


Figure 2: Counterbalance calculations

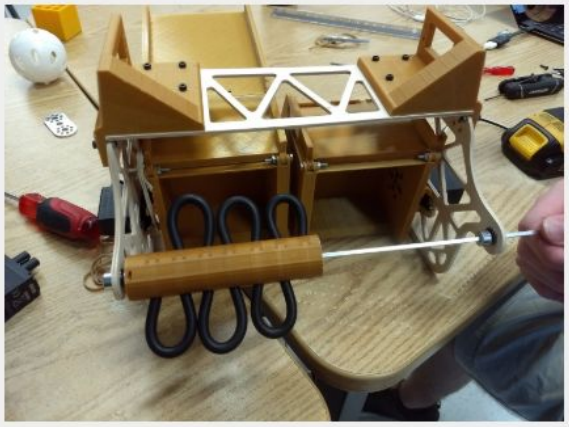


Figure 3:
The Manipulator we tested by doing depositing and collecting

Worksheet: Formatting Your Engineering Notebook

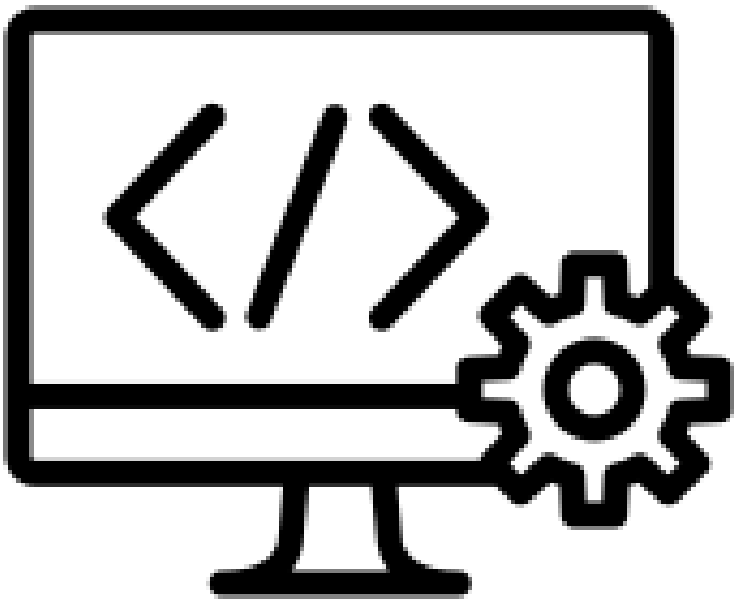
A team can choose to format and make their engineering notebook in a variety of ways. Some notable teams use written engineering notebooks, preferring to write in daily or weekly logs. Others use online engineering notebooks, typing up and printing our their daily or weekly logs. On the topic of logs, a team can choose to have daily logs, biweekly logs, or weekly logs. These mean that a team either writes a meeting down into the engineering notebook every day, twice a week or at the end of the week. The most effective method is probably weekly logs as they allow for progress throughout the week to be condensed into a couple of pages, allowing the evolution of the team and robot to be seen clearly. It is also easier on the team as they don't have to write or type in the engineering notebook everyday or every other day, but instead once a week. Ask your team what type of log schedule they would like and what type of engineering notebook you want to pursue. In the following// pages, we will discuss and guide your team as your create your format and write your first entry.

FORMATTING:

Discuss with your team and determine how often your team will be writing in your engineering notebook. Your team may have a weekly, bi-weekly, or daily entry plan. In the sheet below, draw what your team wants their entries to look like.

YOUR EXAMPLE:

Software Workbook



Software:

Table of Contents:

Software for FTC Teams

Intro to Software

What is Software?

The dictionary definition of software is the programs and other operating information used by a computer. In robotics, the software team is tasked with coding the robot and developing intelligent functions that allow the robot to perform consistently regardless of external factors.

Functions of Software

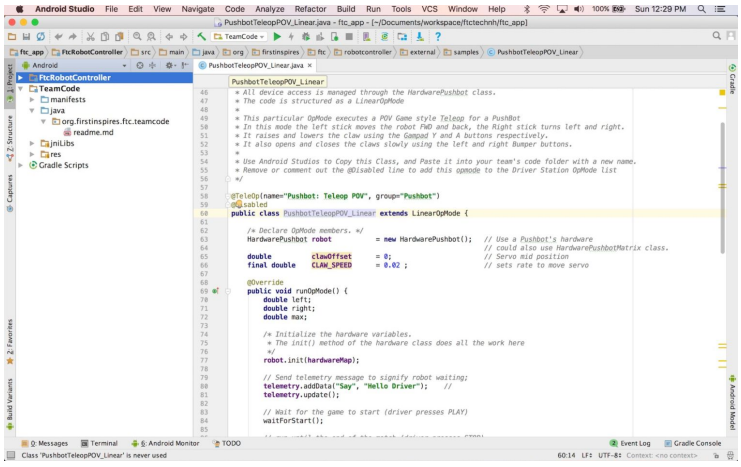
Software allows for you to control your robot, whether it be autonomously or with human drivers. Think of software as being the ‘mover’ of the robot.

FLL software VS FTC software

In FLL, most teams use EV3 and its coding system which involves a combination of blocks with predefined functions such as controlling a drive motor or a sensor. Limited knowledge of syntax is needed to be a successful FLL programmer. However, in FTC, most teams code in the programming language Java. In terms of controlling and coding the robots, the fundamentals are actually quite similar. You must decide the direction and power certain motors must spin, and you must decide how to integrate sensor input to make your robot more efficient.

FLL

FTC



Parts of Software in FIRST - Teleop vs Autonomous

In every FIRST game, there is a 30 second autonomous part of the game in which the robot must move according to a preprogrammed set of instructions. A lot of a software team members' time will be spent coding, testing, and perfecting an autonomous program. During the TeleOp portion of the game, the robot must act according to inputs from two Logitech Controllers which are manipulated by humans on your drive team.

Control System/Phones

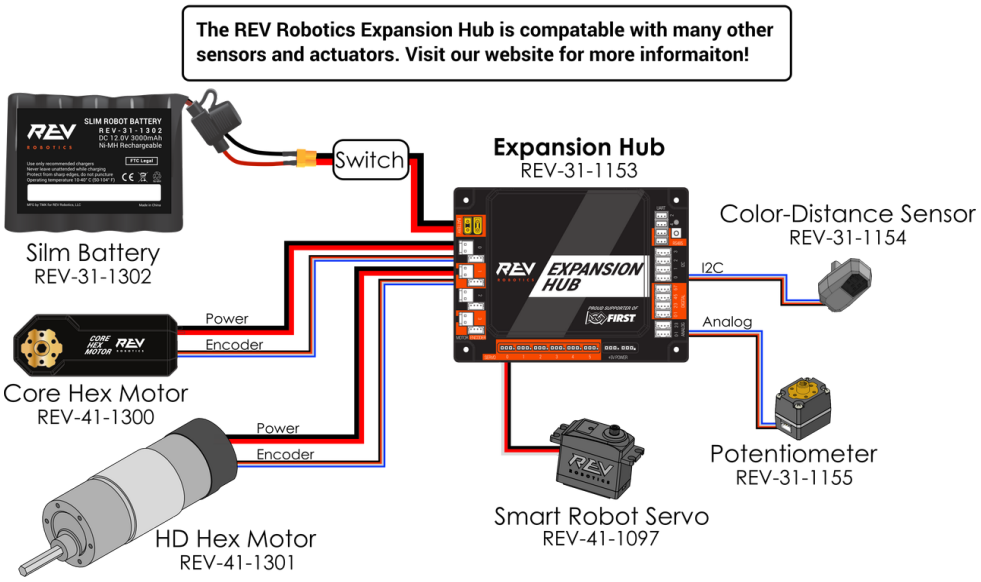
In order to actually control the robot, you will need two phones: 1 Driver Station, and 1 Robot Controller. During a match, the two phones communicate via Wifi-Direct in order to run the code the software team has programmed. During a match, 2 Logitech Controllers will be plugged into the Driver Station with an OTG Cable, allowing for controller input to be sent to the Driver Station. The information is sent to the Robot Controller, which, like its name suggests, is connected to the robot.

Here is a link on how to pair the robot controller and driver station:
https://www.youtube.com/watch?v=psiK2pacsp0

Starting a program on the Driver Station

FTC Robot Controller App

REV Robotics Wiring Reference Sheet



Android Studio

Android Studio will be used to code the robot. Essentially, you will be coding certain portions of a mobile phone app that FTC has given the base code for in Android Studio. Android Studio also allows us to be able to push our changes to the phone to run our new code.

GitHub

GitHub is a version control tool that will allow us to save past versions of code just in case something goes wrong. Additionally, it allows for multiple people to work on different parts of programs simultaneously. If a team member makes changes at home, but can't make it to the meeting, the other software team members will have their changes to test. To make interaction with GitHub easier, we will use GitHub Desktop. There are a couple main functions/commands you need to be aware of:

- Pull: obtains the changes that other team members have done, so that you have to most up to date code
- Push: pushing your code allows for others to see the changes that you have made

Sometimes, when multiple people try and work on the same file, they might overwrite each others' changes. In order to resolve these "merge conflicts" follow the guide in the link below.
https://help.github.com/en/articles/resolving-a-merge-conflict-on-github
[1]

Here is a link to our combined GitHub where we will be putting all of our code:
https://github.com/QuadX6299/SouthAfrica

Intro to Java

What is Java?
Java is a programming language that is used for a variety of different reasons. We will be using Java to code your FTC Robot. To write our code, we will be using Android Studio and deploying our code to an android device.

Variables
Variables hold data and come in many different types.
They are declared as such: Type name = value;
For example an integer with the value of 3 could be declared like this: int example = 3;

Variables can either be declared or left to be declared at runtime:

```
int toBeDetermined;
toBeDetermined = 3;
```

Variables outside of methods have to be declared as private, protected, or public and typically you should declare them as private.

Data Types
There are several types of data in Java.

A variable holds data and must be designated to hold a certain type of data.

- Some primitive data types include **ints (Integers), doubles**
- Some other data types include **Strings (words and sentences) and chars (single characters)**

int / Integer
An int or Integer variable holds only whole numbers, both positive and negative. For example, -1000, -1, 1, and 1000.

To declare an integer variable:

```
int variable = 1;
```

Double
Similarly to an Integer, doubles also hold numbers. However, doubles are special in that they can also hold decimal values as well as whole numbers. For example, -2.5, 1.0, 35.0 are all doubles.

To declare a double variable:

double variable = 0.5;

String
Strings can hold just about any combination of characters you can think of. Individual words, or even a full sentence can be stored in a string variable. Most of the time, in FTC, Strings are used to print telemetry to the driver station. In other words, you can print messages out to the screen that show data that may be important for you to know.

To declare a string variable:

String country = “South Africa!”

*Note: all characters must be contained within quotation mark.

Char (Characters)

- Characters store individual characters or letters

To declare a char variable:

char variable= ‘a’;

*Note: the letter/symbol wanting to be declared needs to be contained within ‘ ‘.

Booleans
Booleans store either a true or false
To declare a boolean:

boolean booling = true;

Operators

The standard math operators work just as you think they will

- Addition: **+**
- Subtraction: -
- Multiplication: *****
- Division: /
- **Modulus**: %
    - Modulus returns the remainder of two numbers, and can be especially helpful if you are trying to determine if a number is odd or even

Try to fill out the following chart determining what the modulus operation will return.

Operation	Result
7 % 2	
10 % 3	
250 % 2	
6 % 5	

Logic

The most basic statement is the **if statement**

- Inside the parentheses, will be your test condition
- If the test evaluates to **true**, the code within the brackets will run

An **if-else statement** can be used to specify what to do if the test condition is false
An **if, else if, else** statement can help you test multiple conditions
There are also logical operators that operate on booleans to create more complex logical statements

&& (and) compares two booleans and if both are true this statement returns true
|| (or) compares two booleans and if one of them is true this statement returns true
! (not) inverts a boolean so !true == false would be true
^ (xor) compares two booleans and if one is true and the other is false it returns true

Try to complete these if and else statements and write what you think will be printed out

boolean isTrue = true;
If (isTrue) {
  System.out.println(“True”);
} else {
  System.out.println(“False”);
}	

boolean true = false;
If (!true) {
  System.out.println(“Hehe”);
} else {
  System.out.println(“Haha”);
}	

boolean trickyone = false;
If (trickyOne) {
  System.out.println(“Maybe”);
} else if (!trickyOne && true) {
  System.out.println(“Maybe so”);
} else {
  System.out.println(“Maybe sooooo”);
}	

Loops
If you want to repeat the same code multiple times you can either write it out multiple times or use a loop. For the most part using a loop is much better since it is faster to write and easier to understand (when you get the hang of it).

There are 2 main kinds of loops we will be going over: the **for** loop and the **while** loop

While loop
A while loop runs while a condition defined in the while loop is true; all code in the bracket will be run during each iteration.
To prevent an infinite loop make sure that your condition will eventually become false. For a counter as seen below this means increasing counter each iteration in order to eventually get the statement to become false.

```
int counter = 0;
while (counter <= 10) {
        System.out.println(counter);
        counter++;
}
```

This will print out 0-10 in the console
Note that counter is incremented inside the loop otherwise this will end up being an infinite loop as counter will never change

For loop
For loops do the exact same thing as while loops just in a lesser format. There are three parts to the for loop.

1. The variable definition (replaces defining counter)
2. The condition
3. The variable modifier that happens at the end of each iteration

```
for (int i = 0; i <= 10; i++) {
        System.out.println(i);
}
```

This will do the same thing as the loop above where **i** will start at zero increase by one each iteration (**i++**) and stop whenever **i <= 10** returns false.

Classes

Though variables hold small pieces of data we often want to organize our code by making objects. In fact java is regarded as an Object Oriented Programming language and everything from ints to Strings are objects/classes.

An Object has 3 main parts

1. Variables
2. Constructors
3. Methods

Objects in robotics are often used to represent parts of the robot. For example, one object can be the drive train which will have the methods for moving around the robot and navigating. Another object can be the manipulator which can move the manipulator around. Logically organizing your code is key in robotics and objects are the best way to do so.

For the first example lets make a Dog object that has a name, can bark, and can be pet.

Making a class does not actually make a dog but rather a generalized template for a dog that we can later use to make an individual dog through the constructor.

```
public class Dog {
	private String name;

	//Constructor
	public Dog(String name) {
		this.name = name;
	}

	private String bark() {
		System.out.println(“Ruf!”);
	}

	public void pet() {
		bark();
	}

	public String getName() {
		return name;
	}
}
```

At the top we create an undefined String name which we later define and access through the constructor and methods.
Right below the name variable we have our constructor which is a method that gets run whenever we create a new Dog.
The rest of the methods define behaviors of the Dog and do various tasks.

To create a new dog from our class and have it bark we must instantiate it as a variable and call the constructor. Anytime you create a new object you must use the **new** keyword and call the object's constructor.

Dog rover = new Dog**("Rover"); **The constructor is only called when making a new object
rover.pet();

Since we are creating a new Dog the variable type is Dog and since our constructor asks for a string we pass in a string to the Dog constructor that will set as its name. After that we take the variable and call the pet method on it.

Methods
Methods tidy up your code by taking your code and putting it into a wrapper that can be given information. This wrapper can then return information as desired. Methods help organize your code and prevent you from writing the same code multiple times.

Ex:

```
public void countTo(int number) {
	for (int i = 0; i <= number; i++) {
		System.out.println(i);
	}
}
```

Let's analyze this method.

What does it do? This method takes in any number and prints all the numbers from 0 to that number.
Ex: if we call countTo(20) the method will print out 0-20

First we define it to be public meaning everyone who can access its parent class can use this method.
	public void countTo(int number)

Next we define the return type and for this method the return type is void which means we will not return anything since we will just print out data to the console.

public **void** countTo(int number)

Then we define the name and parameters of the method. Parameters are what we pass into the function whenever we call it and are always hard typed. This means we need to list the type of data we will give the method.

public void **countTo(int number)**

Finally in the body of the method we make a for loop that counts from 0 to number. Remember number will be defined whenever we call the method but we know that it will be an integer.

```
for (int i = 0; i <= number; i++) {
    System.out.println(i);
}
```

To call a method you must first call its parent class followed by a '.' then the method name.

Ex: Dog.bark();

If you are calling the method in the same class as it was defined in you do not have to put a dot but if it helps you can call **this.**methodName();

Ex: this.bark(); **or** bark(); *If you are calling the method from within the class it was defined in*

Understanding scoping

In java variables can be accessible only in their scope and as such understanding the scope of a variable can help you understand why you are having bugs in your code.

Would this run?

```
If (true) {
    Dog rover = new Dog("rover");
}
rover.pet();
```

No, because rover is defined in the if statement and as such cannot be accessed outside of it.

Variables can only be accessed if they are defined in the same or lesser 'indention' or 'scope' of where you are trying to access them.

Ex:

```
Dog rover = new Dog("rover");
If (true) {
    Dog jill = new Dog("jill");
    rover.pet(); //this works since rover was defined in a greater or equal scope
}
jill.pet(); //this throws an error since jill was defined in a lesser scope
rover.pet(); //this works since rover was defined in the same scope
```

This also applies to methods and instance variables however instead of scope methods are controlled by the public, private, and protected keywords.
Given this class:

```
public class Dog {
	private String name;

	//Constructor
	public Dog(String name) {
		this.name = name;
	}

	private String bark() {
		System.out.println(“Ruf!”);
	}

	public void pet() {
		bark();
	}

	public String getName() {
		return name;
	}
}
```

If we make a dog (Dog rover = new Dog(“rover”);) which methods and instance variables could we not access through the variable (i.e. Dog.method, Dog.variable)?

Since we cannot access private methods and variables the only two methods and variables not accessible are name and bark. If we change those from private to public we could access them.

How to write an Autonomous

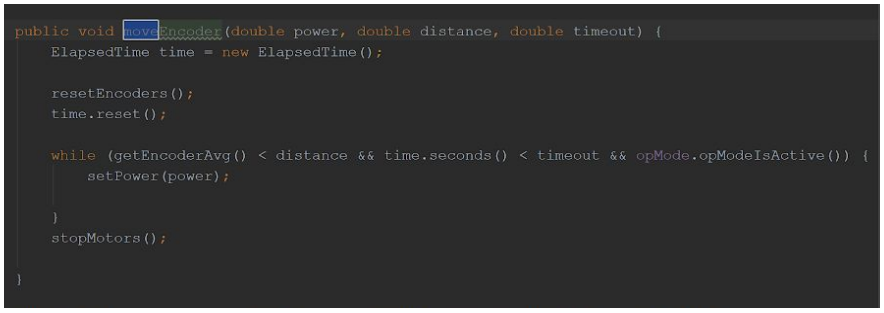
Moving the robot accurately

Wait time

In a wait time auto all you are doing is having the robot move for a given duration of time before giving it another command. While this is fairly simple to code, there are a lot of inconsistencies that may arise due to variances in voltage, slippage of wheels, etc. that basic wait time commands will not account for.
Ex: move(power, seconds) -> turn(power, seconds) -> park()

Encoders

Encoders are used to track the distance the motor rotates and as such can be used to semi-precisely move your motors a certain distance. Using encoders greatly increases your accuracy compared to wait-time autos but they do not achieve complete accuracy.
Ex: move(distance) -> move(distance) -> turn(angle)



```
public void moveEncoder(double power, double distance, double timeout) {
    ElapsedTime time = new ElapsedTime();

    resetEncoders();
    time.reset();

    while (getEncoderAvg() < distance && time.seconds() < timeout && opMode.opModeIsActive()) {
        setPower(power);

    }
    stopMotors();

}
```

Gyro + PID

Gyro + PID is used to make accurate turns that correct themselves for errors. The gyroscope gives you the angle which your robot is at compared to its initial position. So if you start facing 0 and turn 90 degrees the gyro will tell you that you are facing 90 degrees. We can simply say turn right until you hit 90 degrees but the robots inertia will consistently overshoot the targeted angle. To avoid this we use a PID loop (http://blog.opticontrols.com/archives/344) to correct for over shoots and land perfectly at the desired angle.

Writing an autonomous entails taking all of the movement methods and running them in series to complete the various tasks you have to do. For example moving forward to the crater -> turning 70 degrees to the wall -> moving to the wall -> turning to the depot -> moving to the depot -> stopping in the depot.

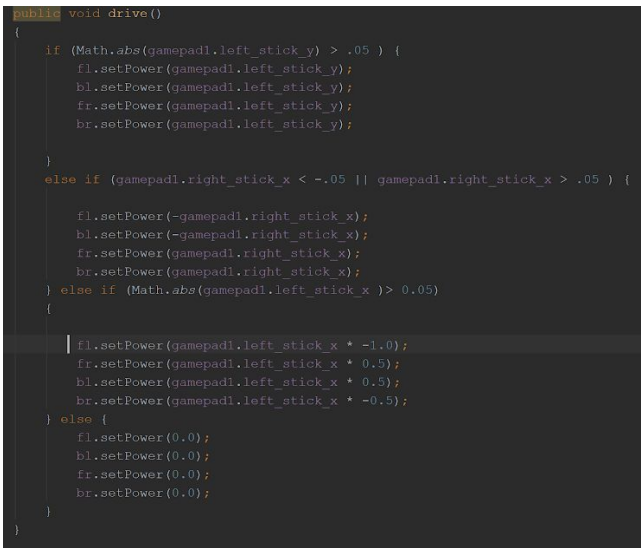
How to write a TeleOp

We have included a sample teleop you can use but essentially a teleop just maps buttons from your controller to movement of the robot. We do this by using the gamepad1 and gamepad2

objects which extend the OpMode class (don't worry about what this means). To map gamepad1 and gamepad2 to the components we must also register the hardware components on our robot in the program by using hardware map.

Here are some examples:

```
public void init() {
    // MOTOR INITIALIZATION
    fl = hardwareMap.dcMotor.get("fl"); //0
    fr = hardwareMap.dcMotor.get("fr"); //2
    bl = hardwareMap.dcMotor.get("bl"); //1
    br = hardwareMap.dcMotor.get("br"); //3

    arm = hardwareMap.dcMotor.get("arm");

    intakeMotor = hardwareMap.dcMotor.get("intakeMotor");
    intakeSlide = hardwareMap.dcMotor.get("intakeSpool");

    //output = hardwareMap.dcMotor.get("output");

    fl.setDirection(DcMotorSimple.Direction.REVERSE);
    fr.setDirection(DcMotorSimple.Direction.FORWARD);
    bl.setDirection(DcMotorSimple.Direction.REVERSE);
    br.setDirection(DcMotorSimple.Direction.FORWARD);

    arm.setZeroPowerBehavior(DcMotor.ZeroPowerBehavior.BRAKE);

    intakeMotor.setDirection(DcMotorSimple.Direction.FORWARD);
    intakeSlide.setDirection(DcMotorSimple.Direction.FORWARD);


    // SERVO INITIALIZATION
    intakeGate = hardwareMap.servo.get("intakeGate");
    knocker = hardwareMap.servo.get("knocker");
```



```
public void drive()
{
    if (Math.abs(gamepad1.left_stick_y) > .05 ) {
        fl.setPower(gamepad1.left_stick_y);
        bl.setPower(gamepad1.left_stick_y);
        fr.setPower(gamepad1.left_stick_y);
        br.setPower(gamepad1.left_stick_y);

    }
    else if (gamepad1.right_stick_x < -.05 || gamepad1.right_stick_x > .05 ) {

        fl.setPower(-gamepad1.right_stick_x);
        bl.setPower(-gamepad1.right_stick_x);
        fr.setPower(gamepad1.right_stick_x);
        br.setPower(gamepad1.right_stick_x);
    } else if (Math.abs(gamepad1.left_stick_x )> 0.05)
    {

        fl.setPower(gamepad1.left_stick_x * -1.0);
        fr.setPower(gamepad1.left_stick_x * 0.5);
        bl.setPower(gamepad1.left_stick_x * 0.5);
        br.setPower(gamepad1.left_stick_x * -0.5);
    } else {
        fl.setPower(0.0);
        bl.setPower(0.0);
        fr.setPower(0.0);
        br.setPower(0.0);
    }
}
```

```
public void intakeSlide() {
    double left_trigger = gamepad1.left_trigger;
    double right_trigger = gamepad1.right_trigger;

    if (left_trigger > 0.05) {
        intakeSlide.setPower(-left_trigger);
    }
    else if (right_trigger > 0.05) {
        intakeSlide.setPower(right_trigger);
    }
    else {
        intakeSlide.setPower(0);
    }
```

Extra Resources

Software is currently one of the most illustrious and easy fields to get into because of the availability of knowledge. With lots of jobs being opened and a vast majority of those jobs having good pay being a software developer is a very good way to make a living and a good path to choose for the future. Since we are here for only 4 days we can only teach you the basics of Java and teach you how to make an elementary autonomous and teleop so I would like to give you some resources that you can use to better advance your robotics careers or just learn software in general.

Coursera

Coursera is an amazing resource to learn new skills with hundreds of internationally acclaimed universities giving you courses that you can take for free. Though paying gives you the added bonus of being able to submit and get your assignment graded just auditing the course gives you all the knowledge you need to do well.

To audit (take it for free) a course such as this one:
https://www.coursera.org/learn/introduction-tensorflow
Click on enroll and after logging in look at the bottom of the popup screen there should be a small blue link that says audit which you can click to get access to the course for free.

Youtube

Though something like coursera will give you an in depth understanding of a topic youtube is a great source for quick tutorials and walkthroughs on certain technologies so I recommend checking youtube if you ever want to explore a topic in software.

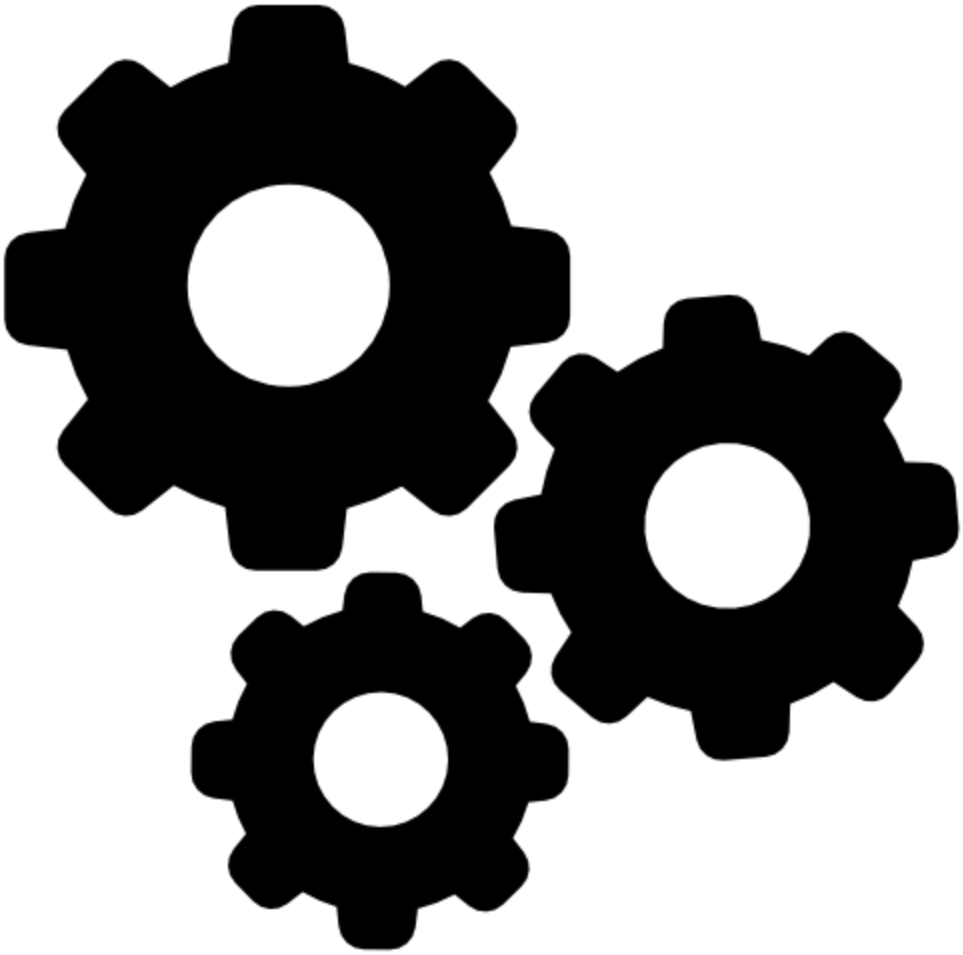
Forums

From the FRC, FTC, and Robotics reddits to ChiefDelphi and the FTC discord there are plenty of FIRST specific resources to learn robotics and interact with the community. In general if you have an issue just google it as there will be plenty of resources to aid you.

StackOverflow

Whenever you get an error in java and just can't fix it head over to stackoverflow and search it up. The massive platform is dedicated to fixing your issues and if you ever find that an issue you have hasn't been explored feel free to make your own post so that people can help you directly.

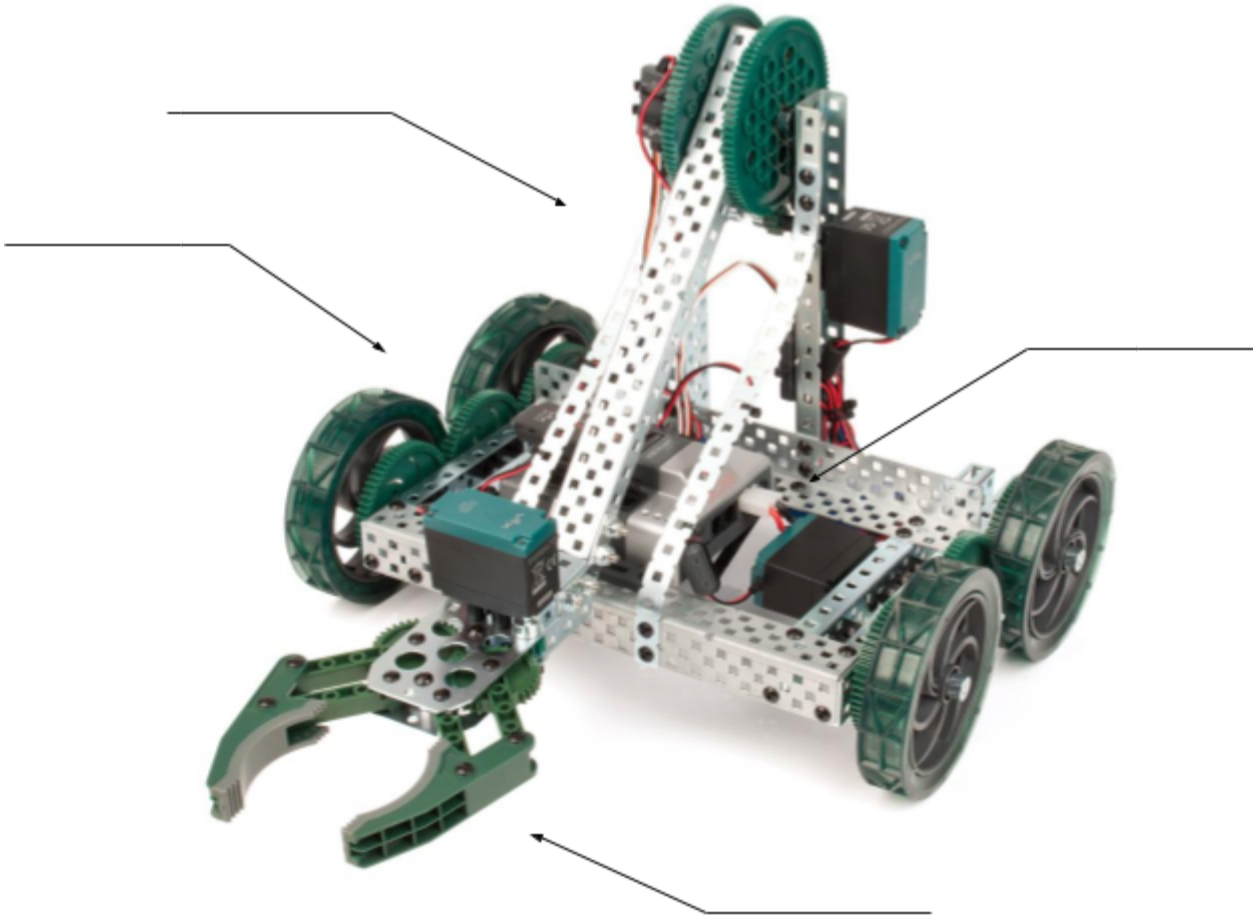
Hardware Workbook



Hardware Basics

The hardware subsection covers all of the physical building of your robot, from the wheels or an arm to wiring and electronics. In order for both your software team and marketing team to do their job, you need to have a really solid and dependable robot.

Basic Anatomy of a Robot



Drivetrain - A drivetrain provides the primary method of moving around the field

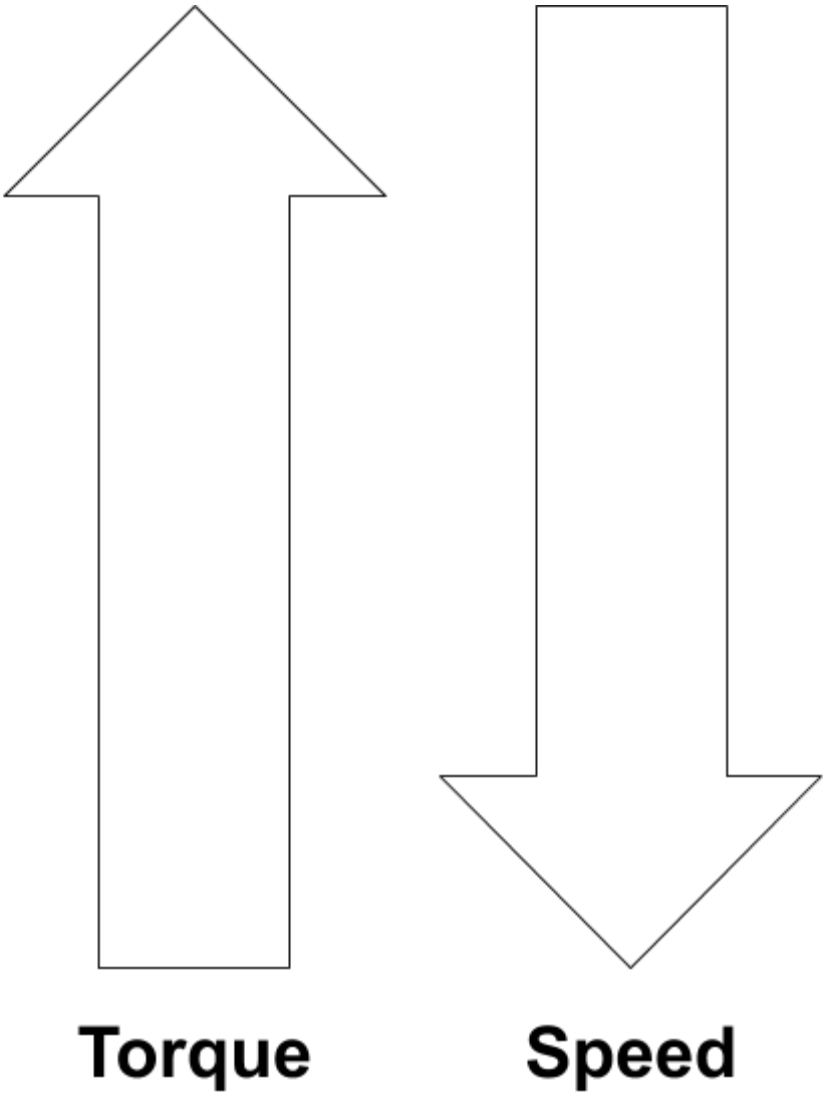
Arm - Allows the manipulator greater mobility

Manipulator - Engaged with field elements

Control System - Receives signals and relays instructions to the rest of the robot

Torque VS Speed

Torque and speed are the two parameters that you always need to think about when designing a mechanism on a robot. Torque is how **strong** something is, while speed is just how **fast** it is. As torque increases speed decreases and as speed increases torque decreases. They are both directly proportional. This is very important because for collecting balls you would want something very fast but it doesn't need lots of torque, whereas when making an arm you need a lot more torque but it can be very slow.



Torque and Speed Practice

Subsystems

- For a **drive train** you would want to use (higher / lower) torque and (higher / lower) speed.

- For an **arm** you would want to use (higher / lower) torque and (higher / lower) speed.

- For an **intake** you would want to use (higher / lower) torque and (higher / lower) speed.

- For a **lift** you would want to use (higher / lower) torque and (higher / lower) speed.

- For a **servo** you would want to use (higher / lower) torque and (higher / lower) speed.

Motors

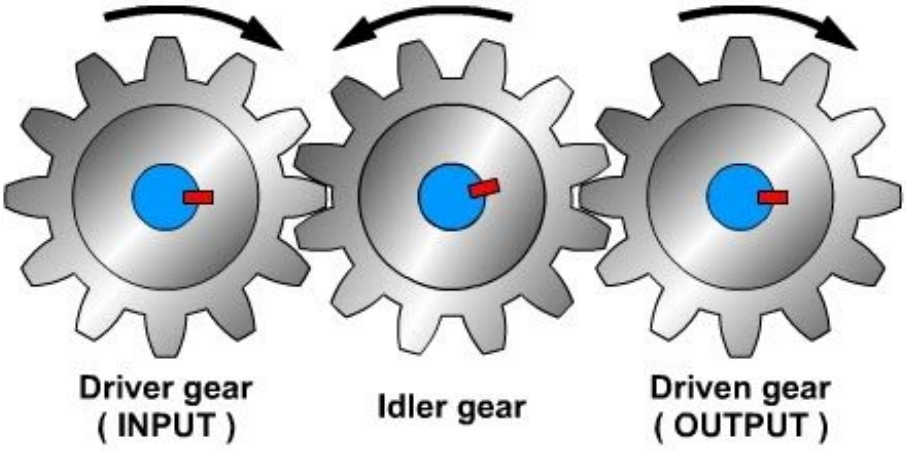
- A **5:1 Motor** has (high / medium / low) torque and (high / medium / low) speed.

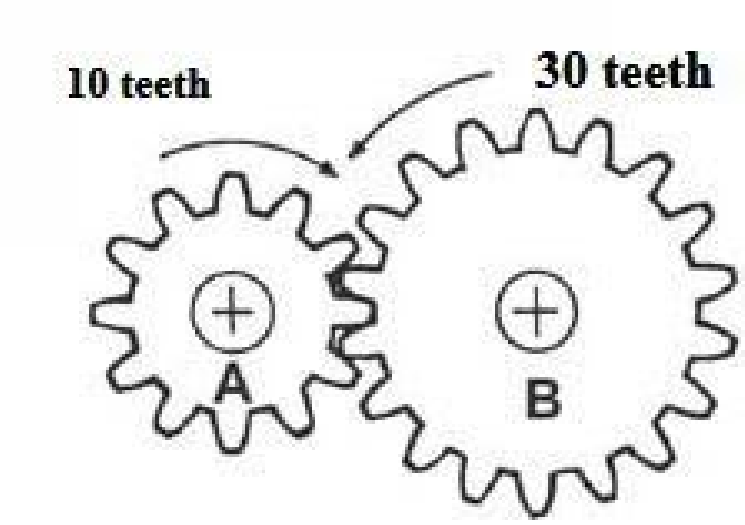
- A **10:1 Motor** has (high / medium / low) torque and (high / medium / low) speed.

- A **25:1 Motor** has (high / medium / low) torque and (high / medium / low) speed.

Gears

Gears are the most traditional way of transferring motion. Gears are circular parts with small teeth around the outside. These mesh together and let you take the power from a motor or servo and apply it to another axle. Something important to remember about gears is how when put together two gears will have opposite directions while spinning. This means sometimes people will use an **Idler Gear** to make the input and output gears spin the same direction.

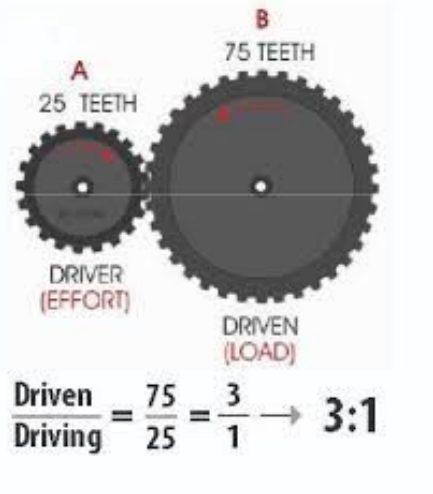




Gear Ratios

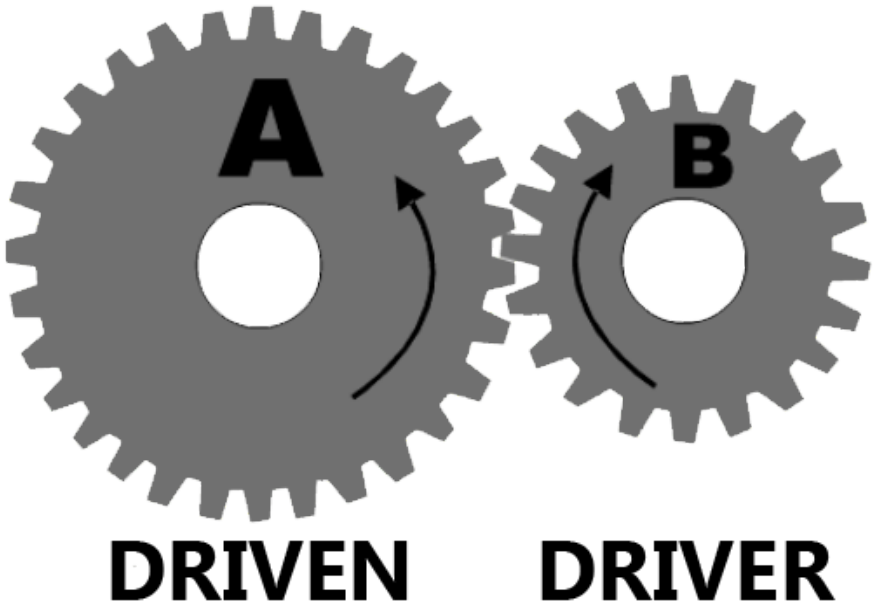
Gear ratios are how you control your torque and speed for mechanisms on your robot. To figure this out you take the teeth of the driven gear and put it over the driving gear. For example, if your motor outputs to a 10 tooth gear then that attaches to a 35 tooth gear the gear ratio would be 3.5:1. This doesn't only apply to gears and is also used to calculate torque when using chain and sprocket. The standard motors that you will be using start with a gear ratio of 26.9:1, 13.7:1, or 3.7:1. The larger the gear ratio, the more torque it has and the slower it is. You will have to choose between these motors when building based on the application. You can add gears or sprockets on these motors to further change the gear ratio. Below is a table with common mechanisms and a recommended gear ratio for each.

Mechanism	**Final Gear Ratio**	**Number of Motors**
Drivetrain	20:1	4
Linear Slides	20:1	1
Linear Slides	10:1	2
Arm	100:1 (Increase if the arm is heavy)	1
Intake	3:1	1



Gears Practice

Gear ratios are written **Driven** gear to **Driving** gear. Always remember to reduce to the lowest common denominator.

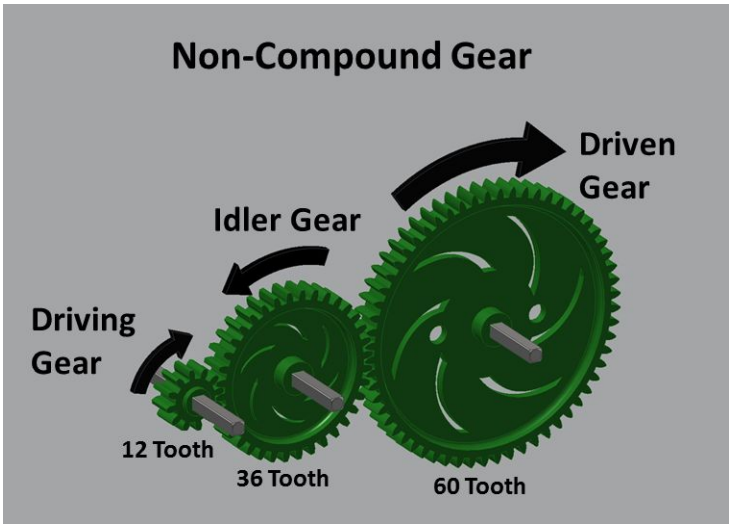


Gear ratios are written Driven gear to Driving gear. Always remember to reduce to the lowest common denominator.

In all of the following examples Gear A is the driving gear and Gear B is the driven gear.

# of teeth	RATIO	HOW MANY TIMES
A= 10; B=20	_____	If A turns one time, B will turn _____.
A= 40; B=10	_____	If A turns one time, B will turn _____.
A= 36; B=60	_____	If A turns one time, B will turn _____.
A= 12; B=60	_____	If A turns one time, B will turn _____.
A= 84; B=36	_____	If A turns one time, B will turn _____.

Multiple Gears in a Train:



In gear-trains in a line the ratio will change from set to set, but ends as a ratio from the first to the last.
Example: A has 10 teeth; B has 50 teeth; C has 40 teeth; the ratio for A:B is 5:1; the ratio for B:C is 4:5. Multiply 5:1 x 4:5 and you have 4:1. *Simpler to ignore all idler gears. C:A - 4:1*

In all of the following examples Gear A is the driving gear, Gear B is the idler, Gear C is the driven gear.

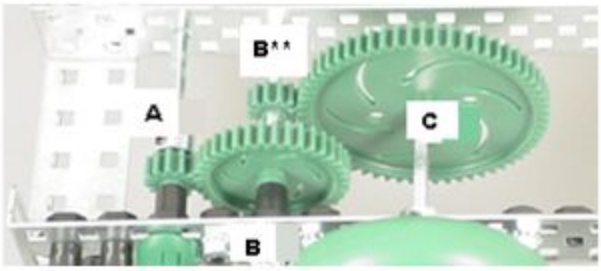
# of teeth	RATIO	HOW MANY TIMES
A= 10; B=20; C=10	_____	If A turns one time, C will turn _____.
A= 10; B=40; C=5	_____	If A turns one time, C will turn _____.
A= 12; B=36; C=60	_____	If C turns one time, A will turn _____.
A= 84; B=12; C=36	_____	If B turns one time, C will turn _____.
A= 36; B=60; C=84	_____	If A turns one time, C will turn _____.

What are the purposes of an idler gear?

In Gear Trains

In Gear-trains with multiple gears the ratio will change from set to set. If there are two sets of gears on a given shaft, *both spin at the same RPM*. The ratio from gear set to gear set is calculated separately then multiplied together.

Example: A has 8 teeth; B has 36 teeth; the ratio for the A:B is 9:2. B** has 8 teeth; C has 60; the ratio for B**:C is 15:2. Multiply 9:2 x 15:2. The net gear ratio is 9135:4



	RATIO	**HOW MANY TIMES**
A= 10; B=20; B**= 5;C=10	_____	If A turns one time, C will turn _____.
A= 10; B=40; B**= 5; C=5	_____	If A turns one time, C will turn _____.
A= 36; B=12; B**= 36; C=3	_____	If C turns one time, A will turn _____.
A= 10; B=60; B**= 5; C=60	_____	If B turns one time, C will turn _____.
A= 12; B=48; B**= 8; C=6	_____	If A turns one time, C will turn _____.

Finding the Speed in a Gear Train

A motor drives a shaft with a 36 tooth gear moving at 120 RPM. The driven gear has the number of teeth listed below.

Driven gear – 36 tooth How fast is the second gear turning?

_____

Driven gear – 60 tooth How fast is the second gear turning?

_____

Driven gear – 84 tooth How fast is the second gear turning?

_____

Driven gear – 12 tooth How fast is the second gear turning?

_____

Finding the Torque in a Gear Train

A motor turns a shaft with a 36 tooth gear that has a torque of 1.5 Nm. That gear is meshed to a gear having the number of teeth listed below.

Driven gear – 36 tooth What is the torque on the second gear? _____

Driven gear – 60 tooth What is the torque on the second gear? _____

Driven gear – 84 tooth What is the torque on the second gear? _____

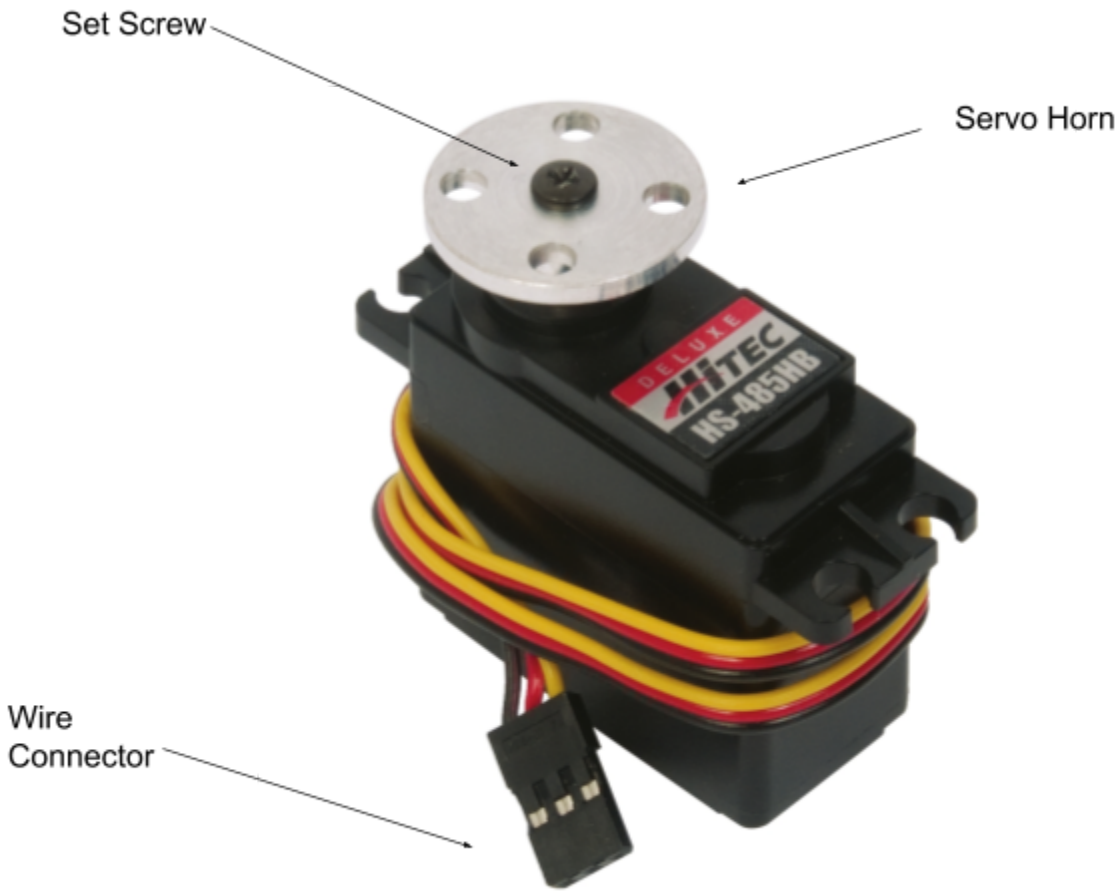
Driven gear – 12 tooth What is the torque on the second gear? _____

Motors

Motors are the primary way that you will be powering things on your robot. In FTC you are limited to 8 motors on your entire robot. Usually, four are dedicated to the wheels and driving the robot, while the other half of them are used for whatever mechanisms you come up with to complete the challenge. Motors are very straightforward and only output by spinning a shaft clockwise or counterclockwise. The motors that you will be using start with a gear ratio of either 26.9:1, 13.7:1, or 3.7:1.

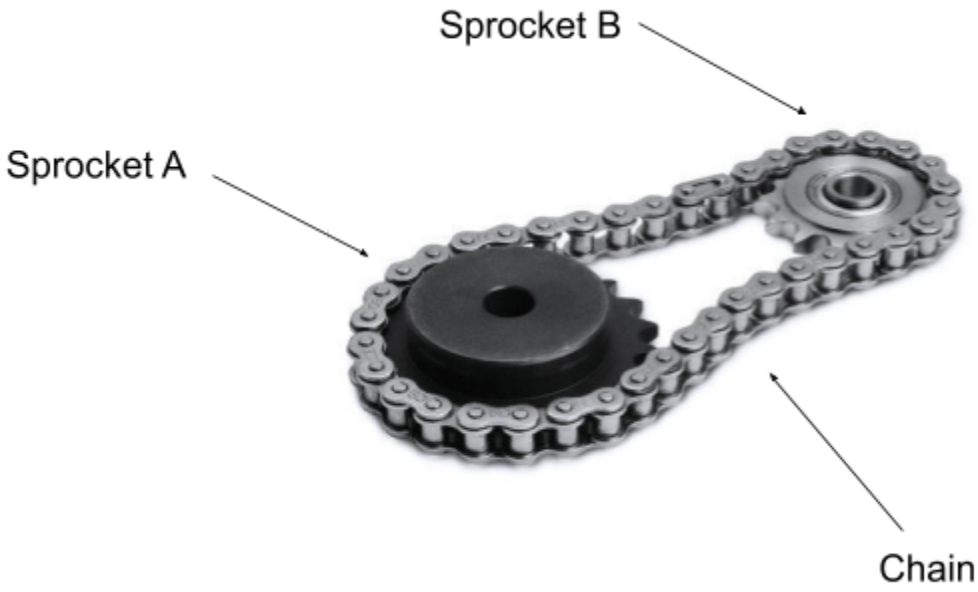
Servos

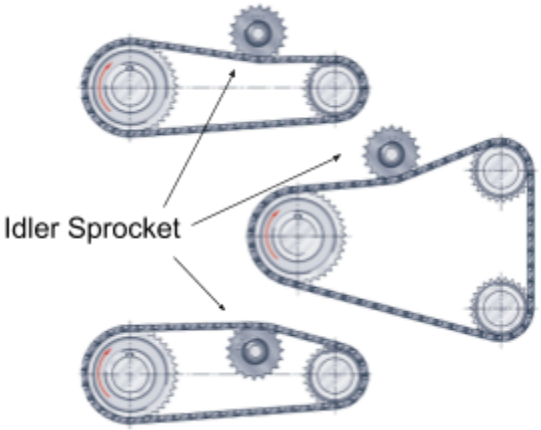
Servos are the secondary way of powering mechanisms on your robot. They are less powerful than motors, but are often more precise. For FTC, you can have up to 12 servos on a single robot for a variety of purposes. There are 180˚, 270˚, and continuous rotation servos available, depending on what task you need to accomplish. 180˚ and 270˚ servos can only turn their respective degree amount in a circle before they will stop.



Sprocket and Chain

Sprockets and chain allow you to transfer rotation power from one place to another. For example, you can turn Sprocket A with a motor, and use chain to connect it to Sprocket B. Sprocket B is now powered, even though it is on a separate axle.

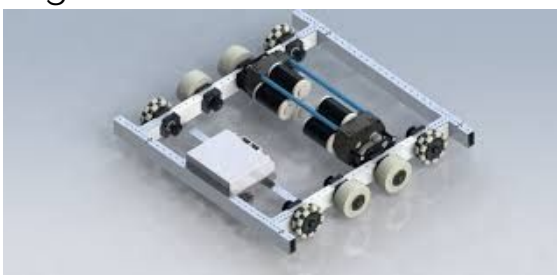
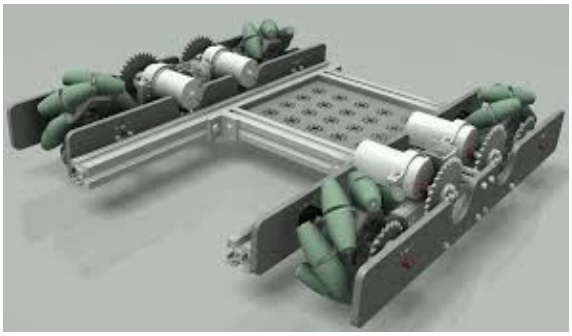
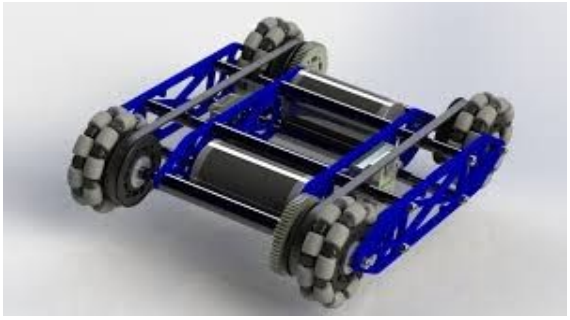


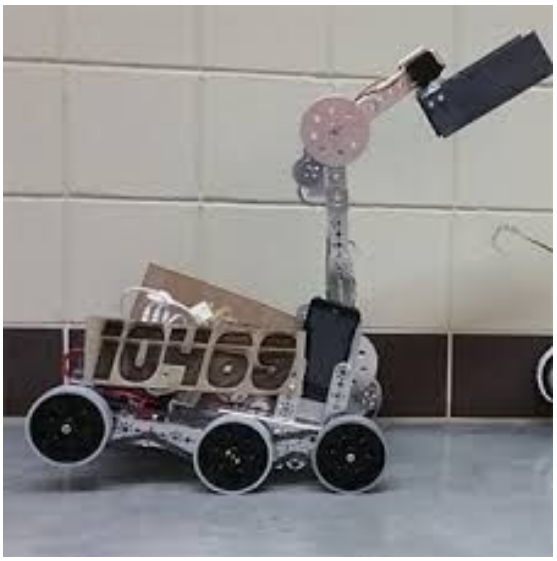
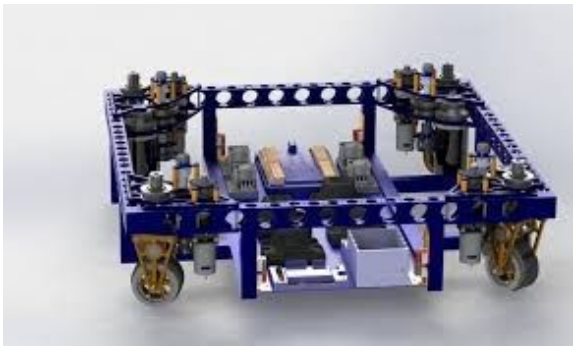


When using chain, it is important to ensure that it is tensioned by adding or removing individual links. This means that there is very little slack, but also that it is not too tight to turn. If a chain is too slack even after removing some links, you may want to add an idler, or roller, than meshes with the chain to reduce slack, but does not take power to another destination. If a chain is too tight, you could add a half link to give yourself a little extra wiggle room.

Drivetrain

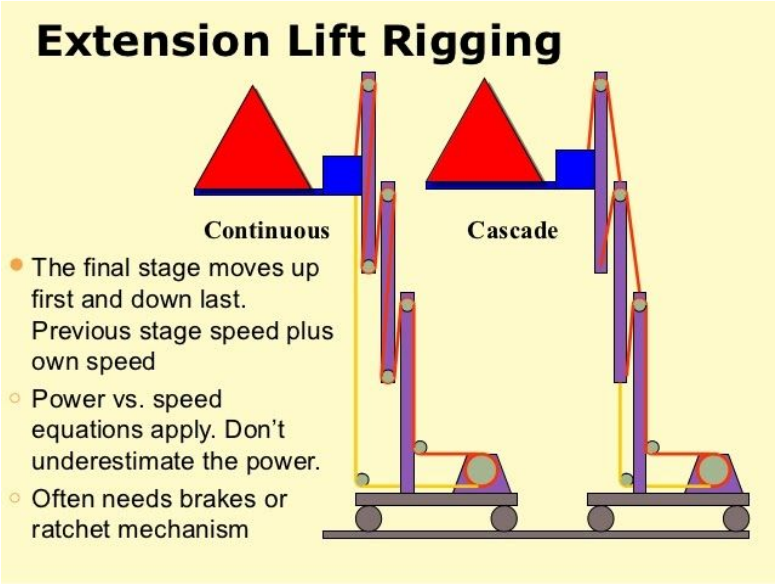
Different types of wheels can be placed in many configurations to achieve unique results. Drivetrains are usually the driving factor in the speed and maneuverability of a robot.

Drivetrain	**Features**	**Notes**
Regular/Omni Mix 	- Mixed wheels give better turning while being harder to push around	
Mechanum 	- Can strafe side to side - More traction than omnis	
Full Omni 	- Turns easily - Little traction when pushed	

Westcoast 	- Center wheel lower - Reduces chance of getting stuck on obstacles	
Holonomic 	- Moves in almost any direction - Easily pushed	
Swerve 	- Moves in almost any direction - Can have complicated programming implications	

Linear Slides

In FTC, linear slide make up the most common source for vertical motion. They can be made from extrusions rigged together, or custom drawer slides. Each set of slides can have as many stages as you need. Each stage adds height and must be rigged to the previous. For rigging, slides can be continuous or cascading. Continuous slides go up one at a time while cascade slide rise simultaneously.



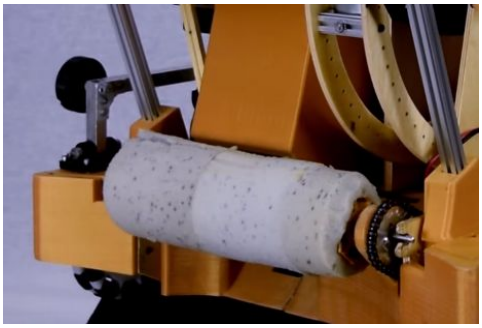
Lifts allow you to transfer elements from the ground, to several feet in the air, usually to score. You can attach your entire intake to your lift and reverse the flow to output elements, or have your intake feed into a separate holding area that the lift takes upwards alone.

Intake

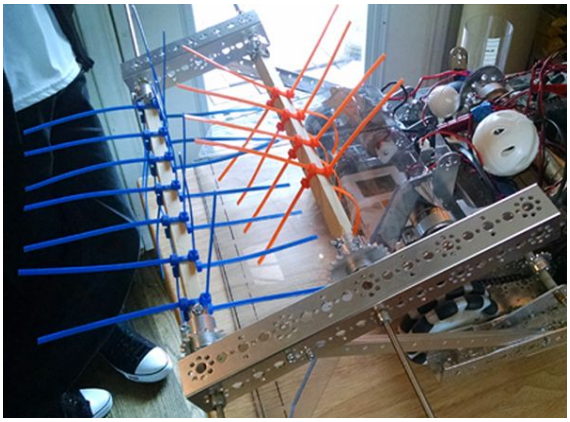
Each year the FTC game features a different type of object to intake and manipulate in order to score. In just the past few years we've seen rings, blocks, and balls ranging from golf ball sized to yoga balls. They are countless ways to collect these items but here are some common ones:

- Surgical Tubing

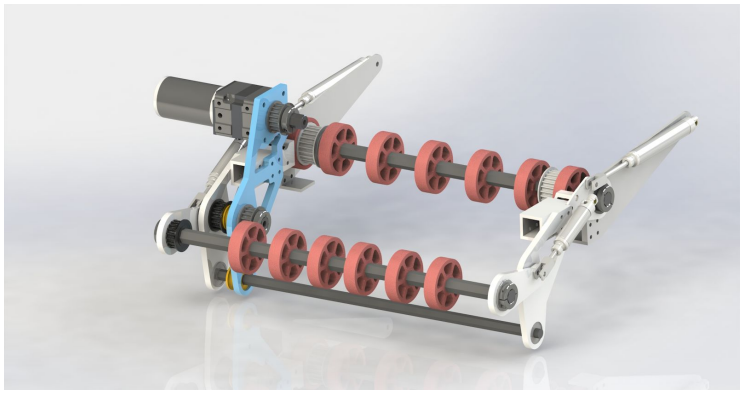
- Foam



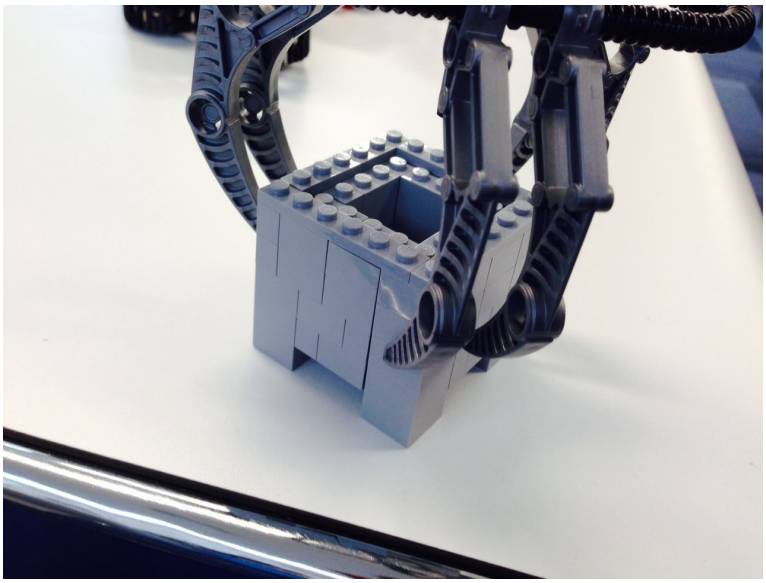
- Zip Ties



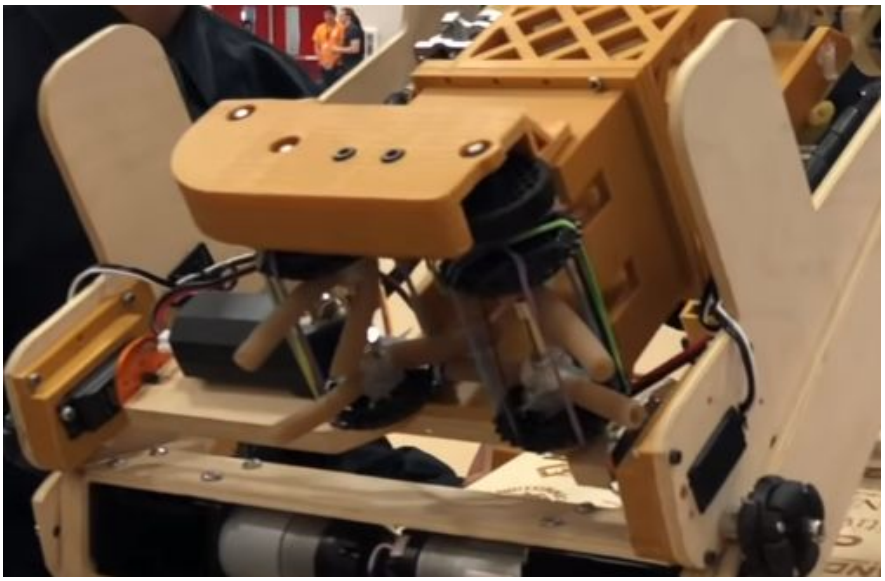
- Rubber rollers



- Claws

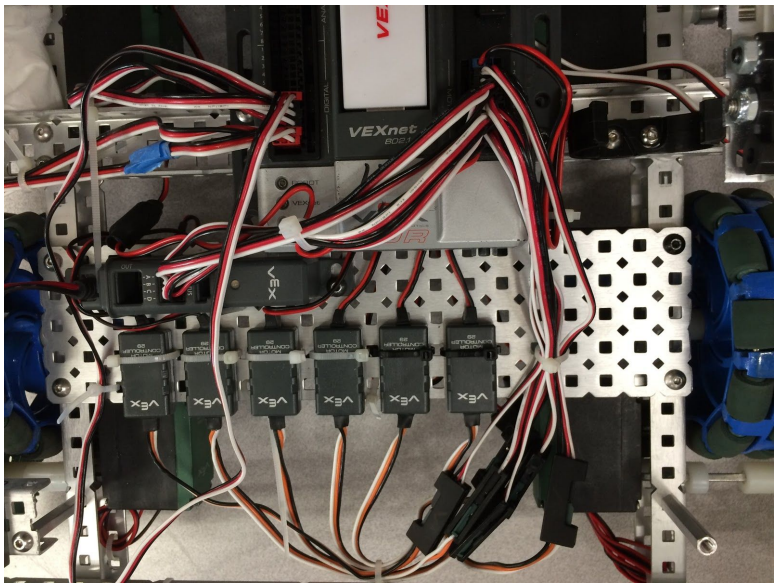


- Rubber bands



Wiring

An absolute necessity in completing any robot is its wiring. All motors, servos, and sensors must be connected back to the control hubs. When wiring, it is important to 1) label all wires (where they came from and where they are going) and 2) keep them neat. Both of these practices make troubleshooting significantly easier when trying to see if something has become unplugged or if you need to test each motor to see if any are faulty. Use zip ties to group wiring going to a common place together, and to the frame of your robot. Any loose wires just create a hazard that could interfere with the function of the robot. Tape and permanent marker can identify what a specific wire is.



The less wiring showing the better!

If possible, place your control hubs near each other to keeping wiring compact and condensed. In addition to labeling the physical wires, make a wiring diagram on paper to reference when you can't be near the robot. This is especially useful for software to have access to when hardware is deep in the middle of a repair or modification.

3D Printing

3D Printing allows for even more freedom in building than traditional materials. Anything you can design on the computer, you can print out. Printing is most useful for creating one of a kind piece that you can't get from a vendor. These are most likely going to be tailored specific to your needs, by you! One of the best things about printing is its fast turnaround time. You can start a print in the evening and return to it the next morning in order to test it out. Make any modifications necessary and start the next version immediately!

Everything seen here in gold is printed!

Printers can be fed a variety of materials such as PLA, ABS, and TPU that vary in strength and flexibility. PLA is a good middle ground, being relatively easy to print with and medium strength. ABS is usually stronger than PLA, but also more brittle. TPU offers a much more flexible route, producing parts that can be squished and then returned to their previous shape.

Once you have designed a part in CAD, you will need a slicer program to convert the file into something that the printer understands. It also allows you to adjust the infill, wall thickness, support structure, and many other factors that can vary the overall strength and durability of a part.

Design Strategy

Before starting to build, you have to have a plan! One way to look at it is to ask “Why?”, “What?”, and *then* “How?”.

Why:

- Breakdown the game and optimize the score
  - Calculate theoretical max scores so you can estimate how much you need to accomplish to be competitive
- Plan your master strategy
  - Which elements do you need to score to gain the most points (not thinking about mechanisms yet!)

What:

- Plan tactics
  - Paths around the field and movement timings
  - Brainstorm multiple for each activity
    - “Throw the ball into the goal” vs “Drop it from above”
- Ways to achieve the score
  - All aspects of the game rather than just the simple act of scoring

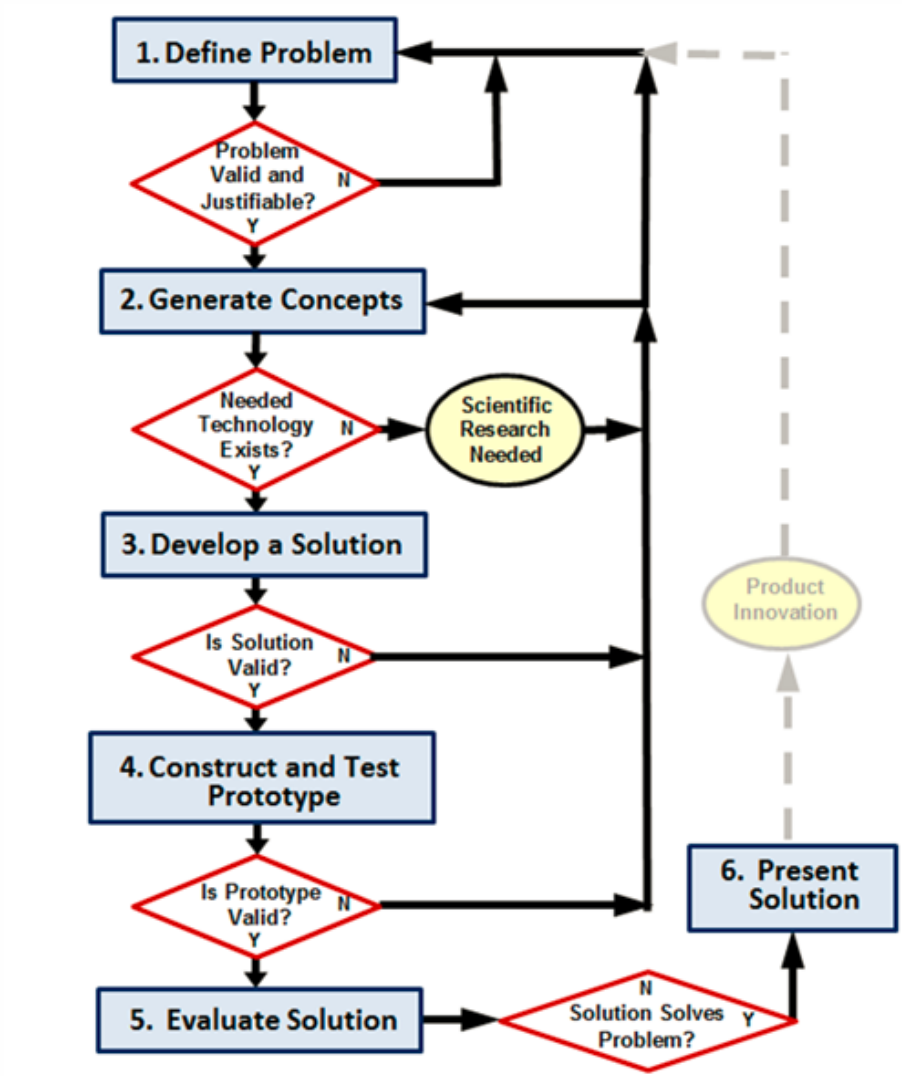
How:

- Think about different mechanisms
  - Create pro and con lists for each idea to systematically decided on the best one
- Consider many ideas
  - Don’t shoot anything down without truly thinking them through - sometimes the craziest ideas can actually be quite effective

Yeah, but How?

- Prototyping your most promising mechanisms
  - Can be working, non-working
  - Real or CAD or detailed sketches
  - Full size or scale models.
  - Can use any materials; Lego, cardboard and duct tape, wood, etc
- Discuss each idea to find the most promising
- Brainstorm how they would fit into the robot
  - Rethink as necessary

When in doubt, the design process flowchart!

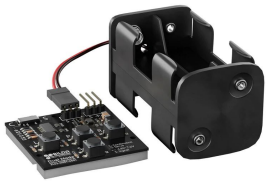


General Hardware Tips

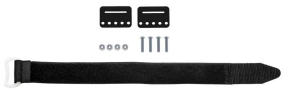
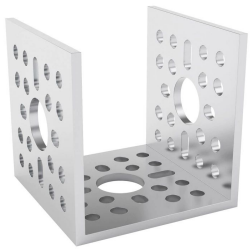
1. Righty tighty, lefty loosey.
    a. Never gets old
2. Measure twice, cut once.
    a. Less wasteful *and* less frustrating
3. Use cardboard, paper, and duct tape, etc to prototype mechanisms.
    a. They are prototypes! They don't have to be perfect by any means. Their purpose is proof of concept.
4. Build for durability.
    a. Practice and competitions will undoubtedly bring a few blows upon your robot. A sturdy robot means fewer repairs necessary.
5. Protect wires.
    a. Protected wires means less chance of disconnect. Plus, it looks really nice.
6. Eliminate as many ways as possible for your robot to get stuck on field elements.
    a. If designed for the worst case scenario, your robot will be working and scoring more of the time!
7. Something can always be improved.
    a. Mechanisms can be optimized and streamline to score faster or work smoother.
8. Keep track of time
    a. While the work is never truly done (see above) make sure software has plenty of time to get their work done, as well as time to driver practice before competition .

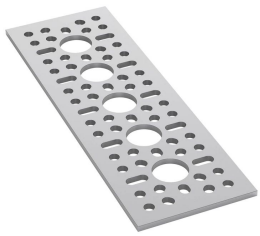
Hardware Components

There are many different parts to an FTC robot. This guide will help you distinguish between the bits and pieces included in your robot kits. Please feel free to record any details you want to remember on these sheets.

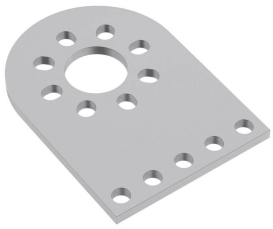
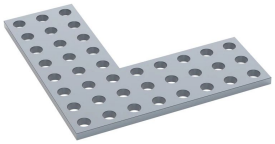
Image of Part	Name, Vendor, Parts Number	Purpose	Notes
	5202 Series Yellow Jacket Planetary Gear Motor (26.9:1 Ratio, 223 RPM, 3.3 - 5V Encoder) GoBILDA.com SKU: 5202-0002-0027	General motor good for large power needs like driving wheels, arms, and lifts.	
	2000 Series Dual Mode Servo (25-1) GoBILDA.com SKU: 2000-0025-0001	General servo good for smaller power needs like flickers or gates.	
	Servo Programmer for 2000 Series Dual Mode Servo GoBILDA.com SKU: 3102-0001-0001	Programmers allowing for servos to be changed from continuous rotation to 180˚ or vice versa.	

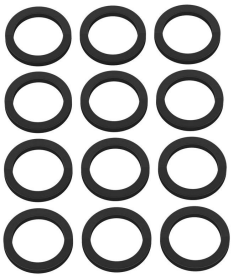
	1801 Series Servo Plate (For Standard Size Servos) GoBILDA.com SKU: 1801-0040-0001	Mounts servos to a robot frame	
	3-Pos TJC8 Servo Extension (MH-FC to FH-MC, 300mm Length) GoBILDA.com SKU: 3802-1718-0300	Connects servos to the control hub when the provided servo wire isn't quite long enough.	
	1900 Series Single Servo Arm (25 Tooth Spline, 32mm Length) GoBILDA.com SKU: 1900-0025-0104	Can be attached directly to a servo to mount further attachments	
	1906 Series Lightweight Servo Hub (25 Tooth Spline, 32mm Diameter) GoBILDA.com SKU: 1906-0025-0032	Can be attached directly to a servo to mount further attachments	
	1910 Series Servo Hub Shaft (25 Tooth Spline, 10mm Shaft Diameter 33mm Length) GoBILDA.com SKU: 1910-0025-1033	Can be attached directly to a servo to mount further attachments	

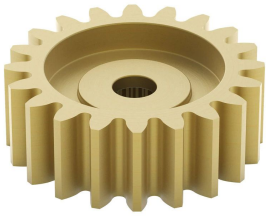
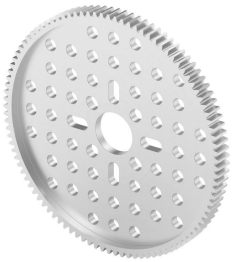
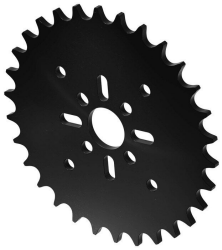
	4001 Series Clamping Servo to Shaft Coupler (25 Tooth Spline to 6mm Round Bore) GoBILDA.com SKU: 4001-0025-0006	Can be attached directly to a servo and shaft to extend a servo's reach	
	Matrix 12V 3000mAh NiMH Battery GoBILDA.com SKU: 14-0014	Power source of the robot	
	XT30 [MH-FC] to Tamiya [MH-FC] Adaptor (60mm Length) GoBILDA.com SKU: 3801-0203-0060	Connects battery to control hub	
	Battery Mount (32-1) GoBILDA.com SKU: 1209-0032-0001	Secures battery to the robot	
	1120 Series U-Channel (1 Hole, 48mm Length) GoBILDA.com SKU: 1120-0001-0048	Basic structural component, available in multiple lengths that can be accessed from three sides	

	1121 Series Low-Side U-Channel (9 Hole, 240mm Length) GoBILDA.com SKU: 1121-0009-0240	Basic structural component, available in multiple lengths that can be accessed from three sides with a single broadside	
	1113 Series L-Channel (1 Hole, 48mm Length) GoBILDA.com SKU: 1113-0001-0048	Basic structural component, available in multiple lengths that can be accessed from two sides	
	1109 Series goRAIL (336mmm Length) GoBILDA.com SKU: 1109-0024-0336	Extrusion that can be cut to any length and often used as lift stages for its rails.	
	4103 Series goTUBE (144mm Length) GoBILDA.com SKU: 4103-0032-0144	Tube with easy access ends, good for making intake rollers	
	1123 Series Pattern Plate (1 x 5 Hole, 48 x 144mm) GoBILDA.com SKU: 1123-0048-0144	Flat plate good for attaching electronics or amour to.	

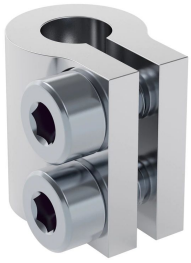
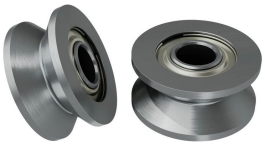
	1116 Series Grid Plate (5 x 9 Hole, 40 x 72mm) GoBILDA.com SKU: 1116-0040-0072	Flat plate good for attaching electronics or amour to.	
	1106 Series Square Beam (9 Hole, 72mm Length) GoBILDA.com SKU: 1106-0009-0072	Beam in multiple lengths for attaching smaller components. Accessible on all sides.	
	1101 Series U-Beam (9 Hole, 72mm Length) GoBILDA.com SKU: 1101-0009-0072	Beam in multiple lengths for attaching smaller components. Accessible on three sides.	
	1103 Series L-Beam (13 Hole, 104mm Length) GoBILDA.com SKU: 1103-0013-0104	Beam in multiple lengths for attaching smaller components. Accessible on two sides.	
	1102 Series Flat Beam (9 Hole, 72mm Length) GoBILDA.com SKU: 1102-0009-0072	Beam in multiple lengths for attaching smaller components. Accessible on one sides.	

	1108 Series Flat Pattern Bracket (1-1) GoBILDA.com SKU: 1108-0001-0001	Good for mounting axles, giving them secondary support from the motor.	
	1205 Series Dual Block Mount (1-2) GoBILDA.com SKU: 1205-0001-0002	Good for mounting structural pieces on a new plane	
	1126 Series Steel Flat Bracket (1-1) GoBILDA.com SKU: 1126-0001-0001	Flat beam good for reinforcing main structure.	
	1137 Series Steel Flat Grid Bracket (1-2) GoBILDA.com SKU: 1137-0001-0002	Flat plate for navigating around corners.	
	1137 Series Steel Flat Grid Bracket (2-1) GoBILDA.com SKU: 1137-0002-0001	Flat plate for mounting offset components	

	1400 Series 1-Side 2-Post Clamping Mount (32mm Bore) GoBILDA.com SKU: 1400-0032-0032	Primary motor mount, or reinforcing support for high stress motors.	
	1504 Series 32mm OD Pattern Spacer (4mm Length) GoBILDA.com SKU: 1504-0032-0040	Spacer that attaches directly to gears or sprockets	
	1501 Series M4 x 0.7mm Standoff (6mm OD, 12mm Length) GoBILDA.com SKU: 1501-0006-0120	Used to reinforce or offset structures. Screws required both ends.	
	1502 Series 4mm ID Spacer (6mm OD, 12mm Length) GoBILDA.com SKU: 1502-0006-0120	Spaces out structure; freespinning	
	1500 Series Plastic Spacer (6mm ID x 8mm OD, 1mm Thickness) GoBILDA.com SKU: 1500-0010-0006	Small spacers to go on axles for specific gear and sprocket placement.	

	2305 Series Brass, MOD 0.8, Servo Gear (25 Tooth Spline, 20 Tooth) GoBILDA.com SKU: 2305-0025-0020	Small gears to change the gear ratio of servos	
	2302 Series Aluminum, MOD 0.8, Hub Mount Gear (14mm Bore, 100 Tooth) GoBILDA.com SKU: 2302-0014-0100	Varying sizes can be used to adjust the output torque and speed of a motor to fit the task. Teeth must mesh together.	
	3311 Series 8mm Pitch Plastic Hub Mount Sprocket (14mm Bore, 30 Tooth) GoBILDA.com SKU: 3311-0014-0030	Varying sizes can be used to adjust the output torque and speed of a motor to fit the task. Must be connected via chain.	
	3309 Series 8mm Pitch Plastic Chain Link GoBILDA.com SKU: 3309-0108-0050	Connects sprockets to transfer power over distance.	
	3601 Series Rhino Wheel (14mm Bore, 96mm Diameter) GoBILDA.com SKU: 3601-0014-0096	Basic drive wheel.	

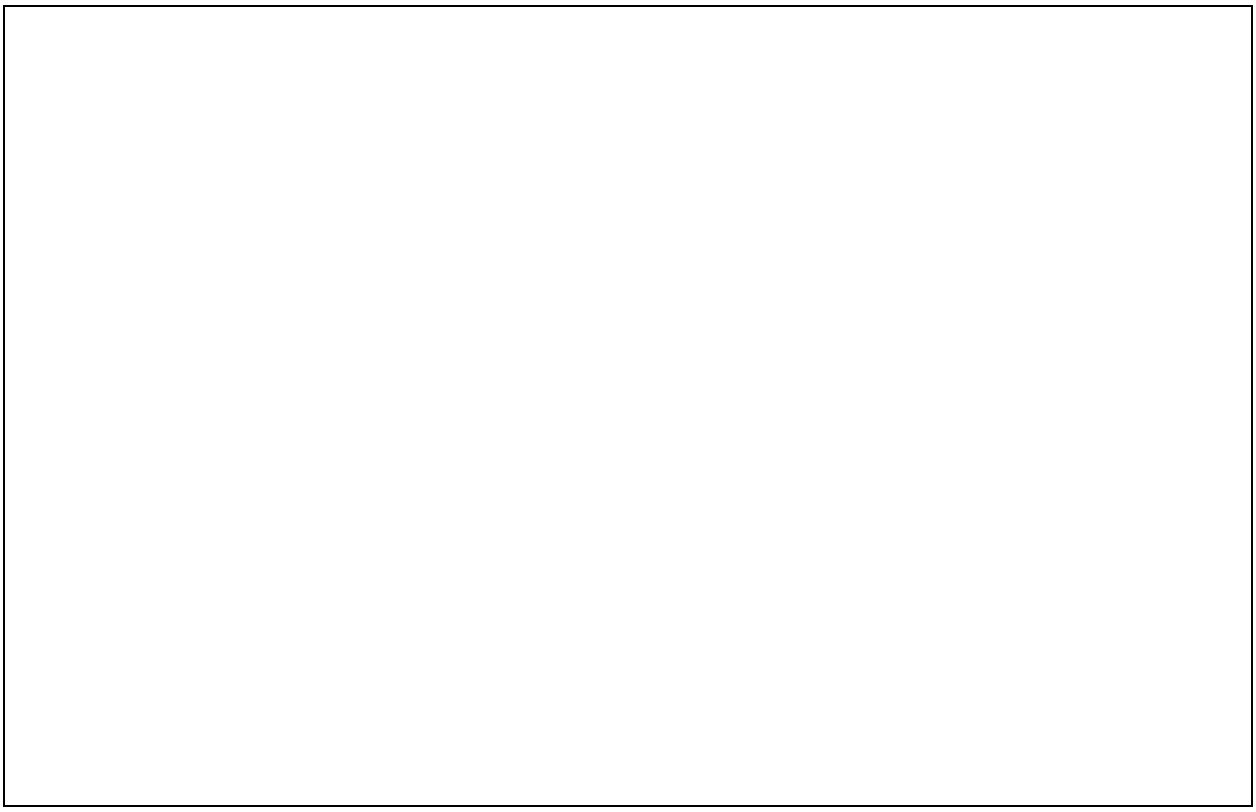
	3604 Series Omni Wheel (14mm Bore, 96mm Diameter) GoBILDA.com SKU: 3604-0014-0096	Drive wheel with rollers to make turning easier.	
	1611 Series Flanged Ball Bearing (6mm ID x 14mm OD, 5mm Thickness) GoBILDA.com SKU: 1611-0514-0006	Mounted in holes on the robot to stabilize and smooth axles movement.	
	1603 Series Face Thru-Hole Pillow Block (10mm Bore) GoBILDA.com SKU: 1603-0032-0010	Mount for bearings.	
	2101 Series Stainless Steel D-Shaft (6mm Diameter, 180mm Length) GoBILDA.com SKU: 2101-0006-0180	Axles to mount gears, sprockets, etc. Used in all rotational motion	
	1308 Series Lightweight Set Screw Hub (6mm D-Bore) GoBILDA.com SKU: 1308-0016-1006	Axle mount with set screw to prevent rotation	

	1301 Series Clamping Hub (6mm D-Bore) GoBILDA.com SKU: 1301-0016-1006	Axle mount with set screw to prevent rotation	
	2906 Series Aluminum Set Screw Collar (6mm Bore, 8mm Length) GoBILDA.com SKU: 2906-0006-0008	Collar for axles with set screw to prevent axles from falling out of place.	
	4000 Series Clamping Shaft Coupler (6mm Round Bore to 6mm Round Bore) GoBILDA.com SKU: 4000-0006-0006	Allows two axles to be secured together, extending reach.	
	1609 Series V-Groove Bearing (4mm ID x 13mm OD, 6mm Thickness) GoBILDA.com SKU: 1609-0613-0004	Grooves allow for string to rest securely within for lifts	
	2800 Series Zinc-Plated Steel Socket Head Screw (M4 x 0.7mm, 6mm Length) GoBILDA.com SKU: 2800-0004-0006	Varying lengths to attach all main structural components. Lefty loosey, righty tighty!	

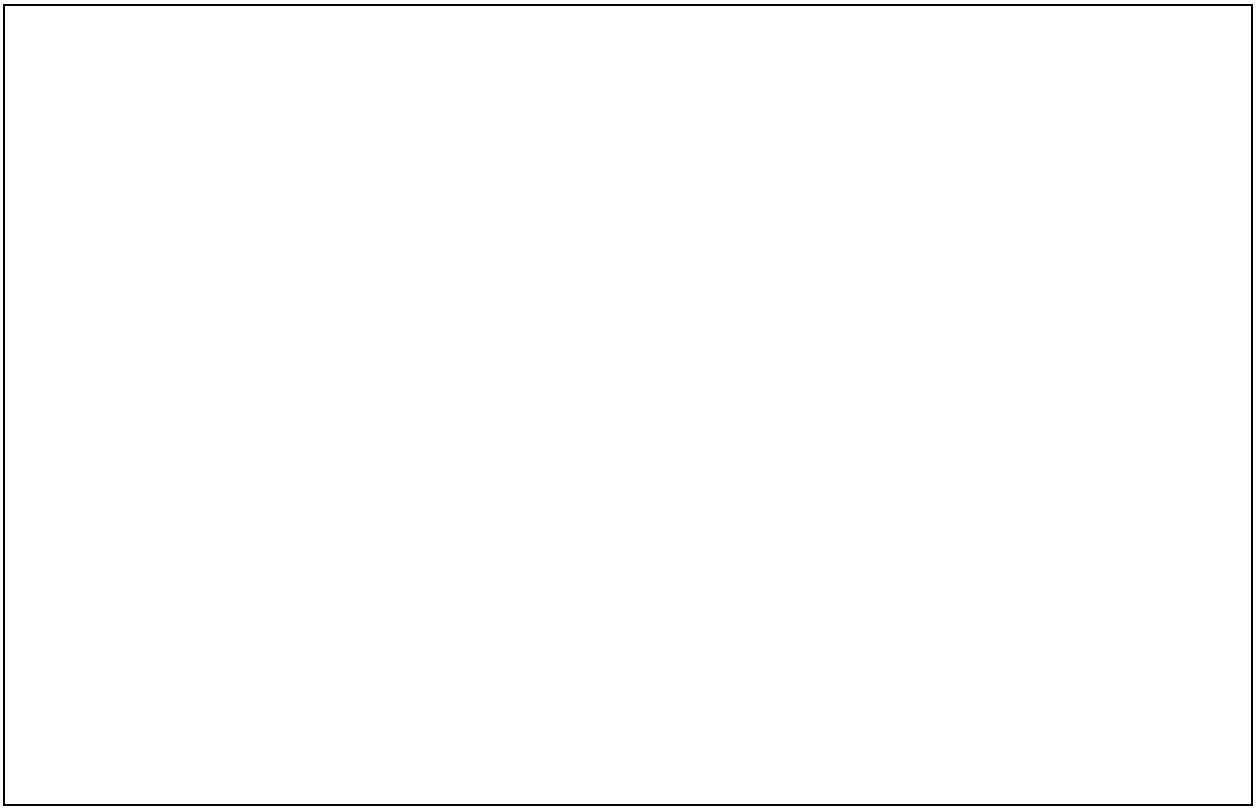
	2812 Series Zinc-Plated Steel Nylon-Insert Locknut (M4 x 0.7mm, 7mm Hex) GoBILDA.com SKU: 2812-0004-0007	Nuts to secure screw once through connecting pieces.	

Brainstorming

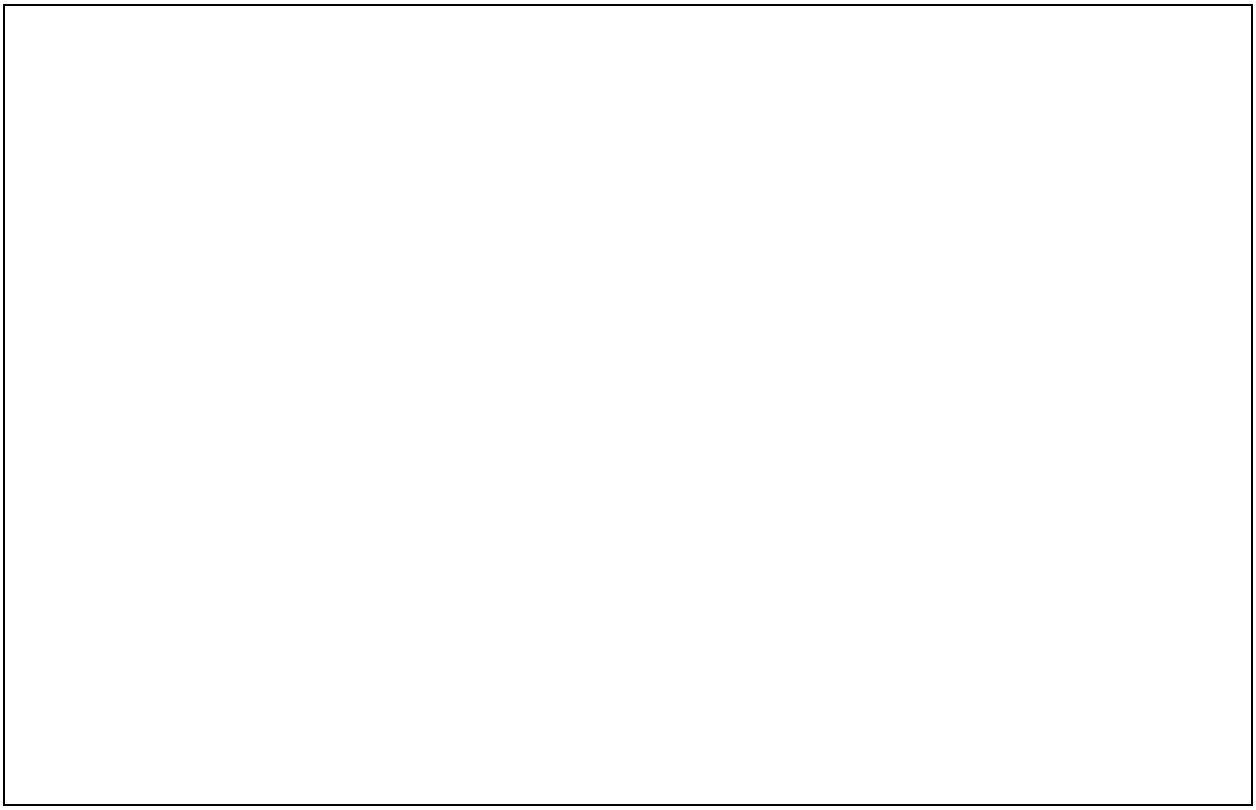
Draw and describe a mechanism for collecting 5cm cubes



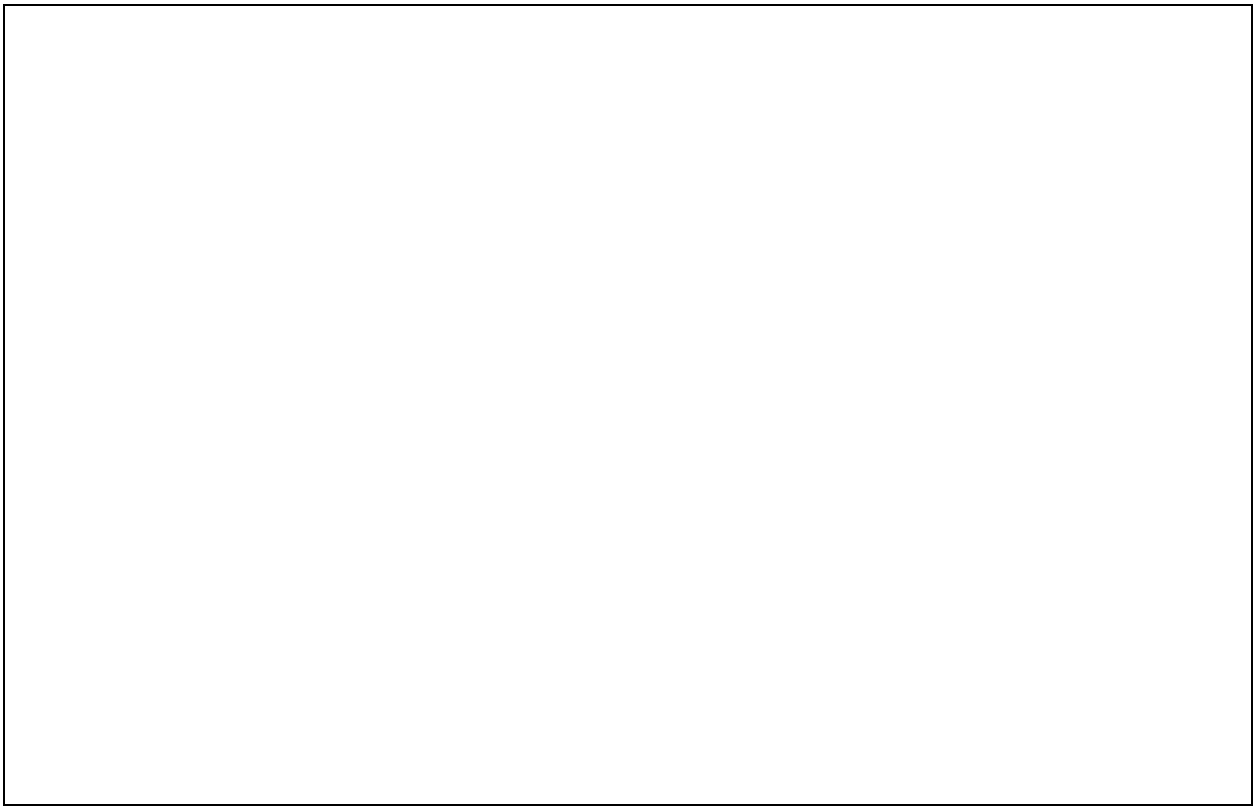
Draw and describe a mechanism for stacking 25cm boxes



Draw and describe a mechanism for shooting 7cm balls



Draw and describe a mechanism for hanging 10 cm rings on pegs



Draw and describe a mechanism for stacking 25cm boxes