FIRST GAME CHANGERS℠

POWERED BY STAR WARS FORCE FOR CHANGE

PRESENTED BY Qualcomm

FIRST® GAME CHANGERS℠ powered by *Star Wars: Force for Change*

2020-2021 *FIRST*® Tech Challenge

Blocks Programming

Autonomous Mode

FIRST TECH CHALLENGE

FIRSTINSPIRES.ORG/ROBOTICS/FTC

Sponsor Thank You

Thank you to our generous sponsors for your continued support of the *FIRST*® Tech Challenge!

***FIRST*® Tech Challenge**
Season Presenting Sponsor

Qualcomm

***FIRST*® Tech Challenge**
Official Program Sponsor

Collins Aerospace

***FIRST*® Tech Challenge**
Official 3D Augmented Reality/
Virtual Reality Sponsor

ptc

Revision History		
Revision	**Date**	**Description**
1	07/21/2020	Initial Release

Contents

Introduction

What is FIRST® Tech Challenge?

FIRST® Tech Challenge is a student-centered program that focuses on giving students a unique and stimulating experience. Each year, teams engage in a new game where they design, build, test, and program autonomous and driver operated robots that must perform a series of tasks. To learn more about *FIRST*® Tech Challenge and other *FIRST*® Programs, visit www.firstinspires.org.

FIRST Core Values

We express the *FIRST*® philosophies of *Gracious Professionalism*® and *Coopertition*® through our Core Values:

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

Gracious Professionalism®

FIRST® uses this term to describe our programs' intent.

Gracious Professionalism® is a way of doing things that encourages high-quality work, emphasizes the value of others, and respects individuals and the community.

Watch Dr. Woodie Flowers explain *Gracious Professionalism* in this short video.

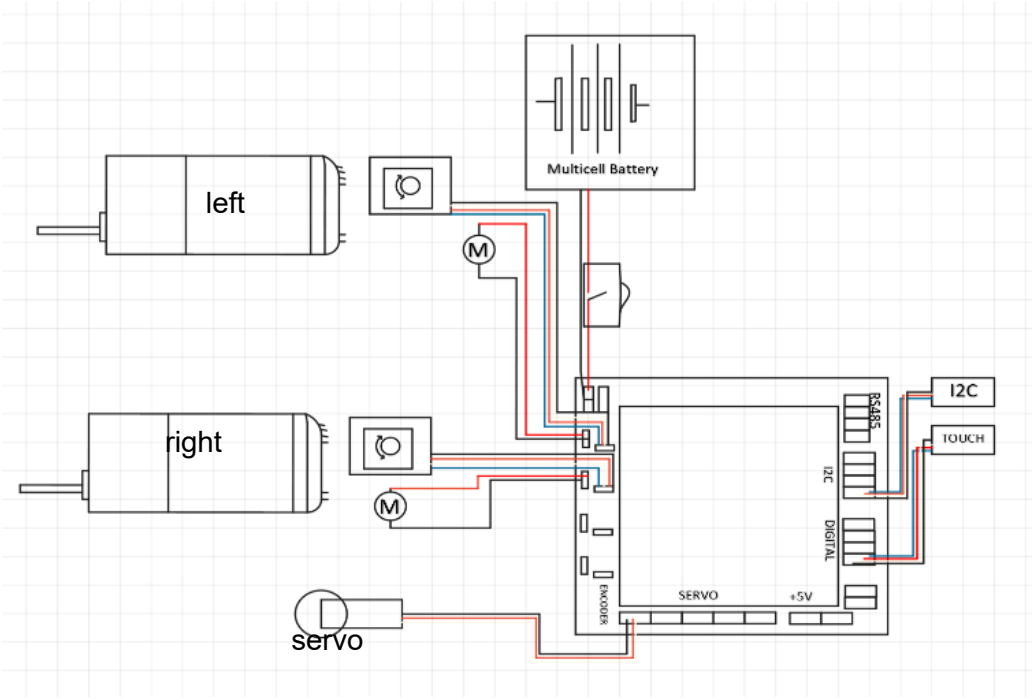
Introduction: Basic Guide for Running Tank Drive and Basic Autonomous

In this guide you will perform the following activities:

- Set up your hardware names.
- Set up configuration guidelines.
- Perform Robot Configuration for multiple motors.
- Use a Tank Drive Op Mode to begin driving.
- Run the Op Mode on the robot.
- Modify the Op Mode.
- Create an Autonomous program without encoders.
- Create an Autonomous program with encoders.

Set up your hardware names

Create an electrical diagram with the names of the hardware. Using an electrical diagram with consistent naming will help you have programs that transfer and are easier to troubleshoot.



Create naming guidelines such as:

- use all lowercase.
- use names that relate to the function of the hardware.
- use names that avoid abbreviations and that could be misinterpreted.

Set up your configuration guidelines

The first step in programming the robot is to make sure you have a configuration file that is current. Configuration is the process of setting up the control hub or expansion hub so that it knows which port has a hardware connection. These names are then transferred as variables into the programming environment. It is important that your program and your configuration file use the same names for the hardware.

There are approaches to ensure that your program matches your configuration file:

1. Open a template program or other program you have downloaded. Change the configuration file to the variables used in the program.

or

2. Change the program to your configuration file.

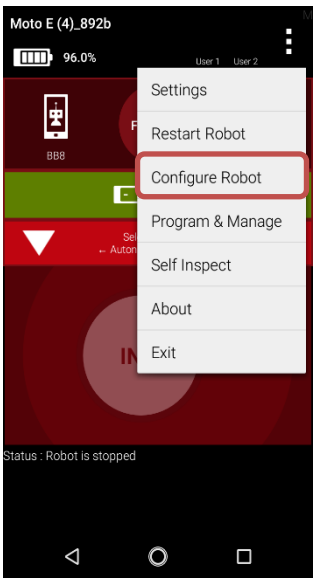
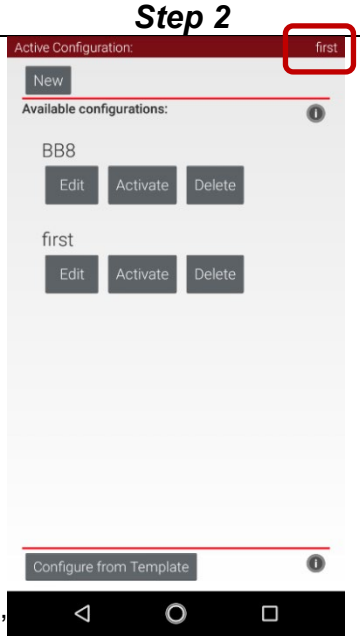
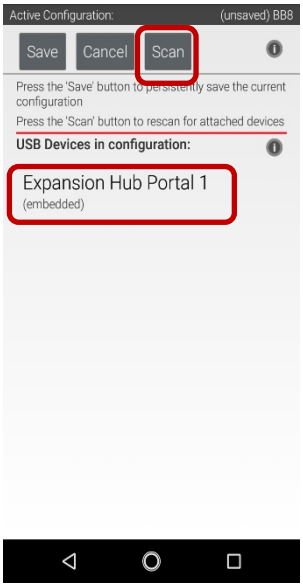
Consistent naming will allow for easier troubleshooting and cross program usage. Students should develop a process for how they create programs that match the configuration of their robot.

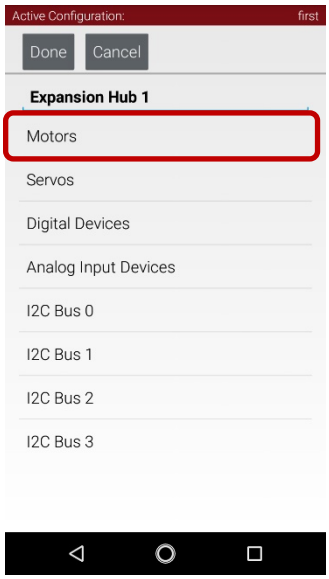
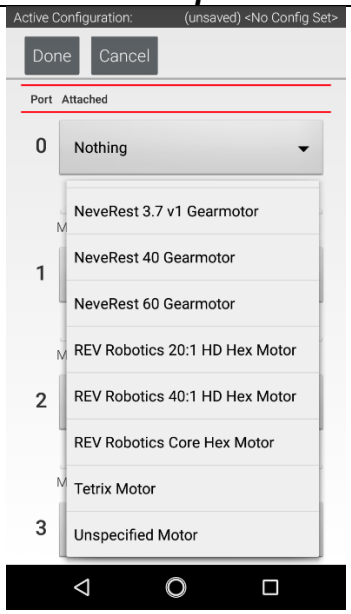
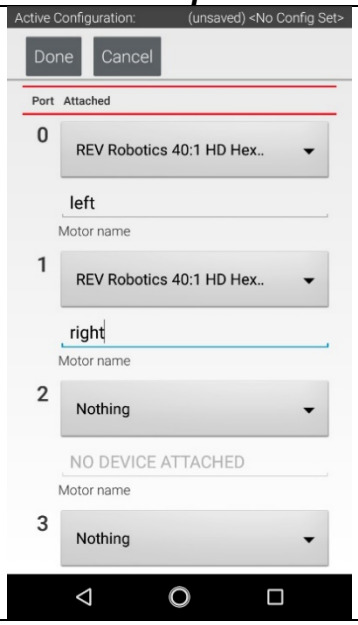
- For example:
  - Add motor for manipulator
  - Add motor to configuration using agreed on naming guidelines.
  - Use name variables that are created in the wiring diagram in all programs.

The names in the configuration file should not change from one configuration to another. For example, in one configuration the left motor is named left_motor but inanother it is left. This will create issues in programs created where the program will no longer match the configuration file.

In this tutorial, we will set up our own configuration file and change templates to fit our configuration file. We will first set up a configuration file for two drive motors on a robot and use the tank drive sample OpMode to drive the robot. The last step will be to set up a basic autonomous program to drive a specific distance and stop.

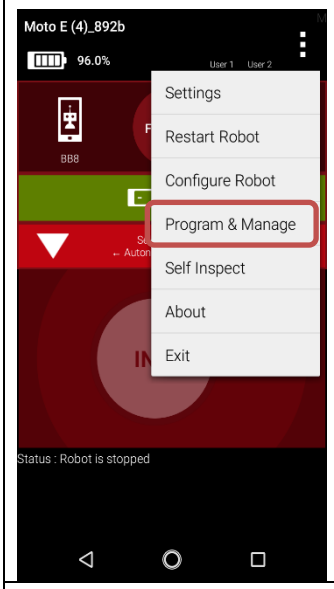
Robot Configuration Steps

Step 1	Step 2	Step 3
		
From the driver station selectthe three dots at the top-right corner. When the menu opens select Configure Robot.	The screen shows available configuration files. You can click new to create a new file or edit the one listed. If this is your first setup, you will not have a config file. **Note**: the name of active file is at the top right.	If the Control Hub is not shown in the USB devices in configuration, select the scan button . If it is shown, select the Expansion Hub.

Step 4	Step 5	Step 6
		

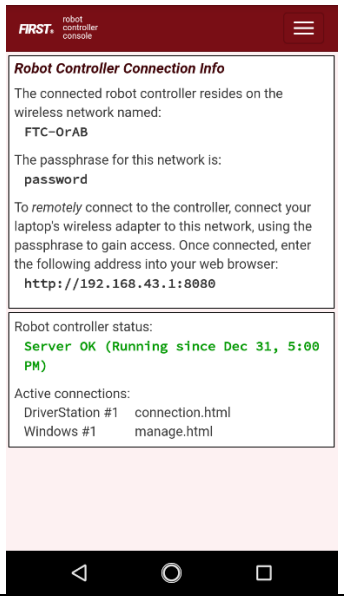
Next select motors from the menu. You will notice that this is a list of the different types of ports that are located on the hub.	Choose the motor type you will be using from the drop-down menu. Motor type allows you to use encoders in other steps correctly.	Type in the name of the motors. These are named “left” and “right” to match the electrical diagram. Click done. This will take you back to the screen in step 2.

Step 1



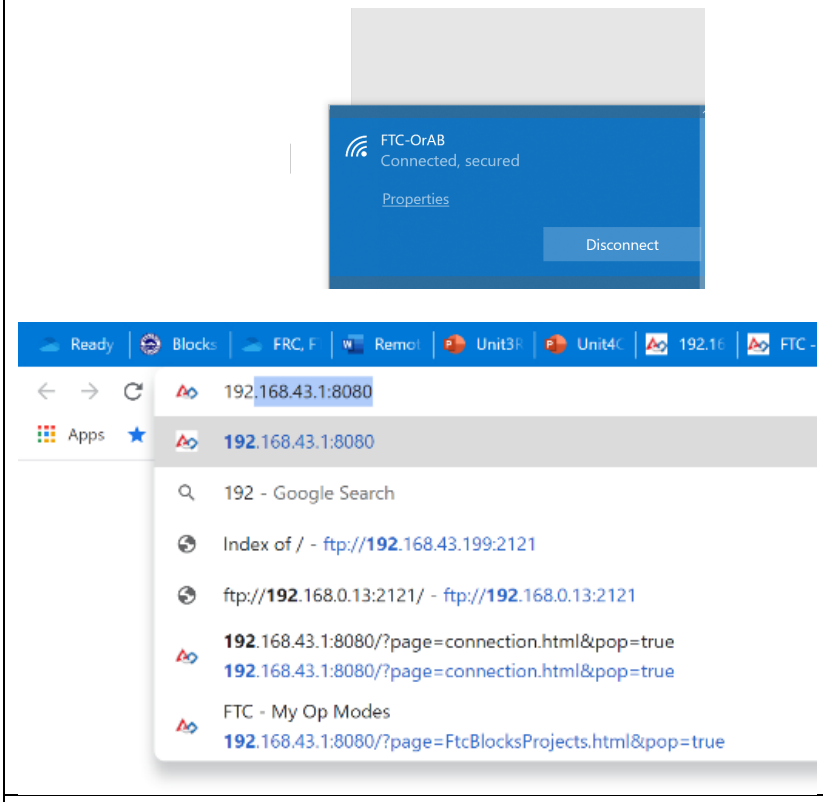
From the driver's station, select the three dots at the top-right corner then select Program & Manage.

Step 2



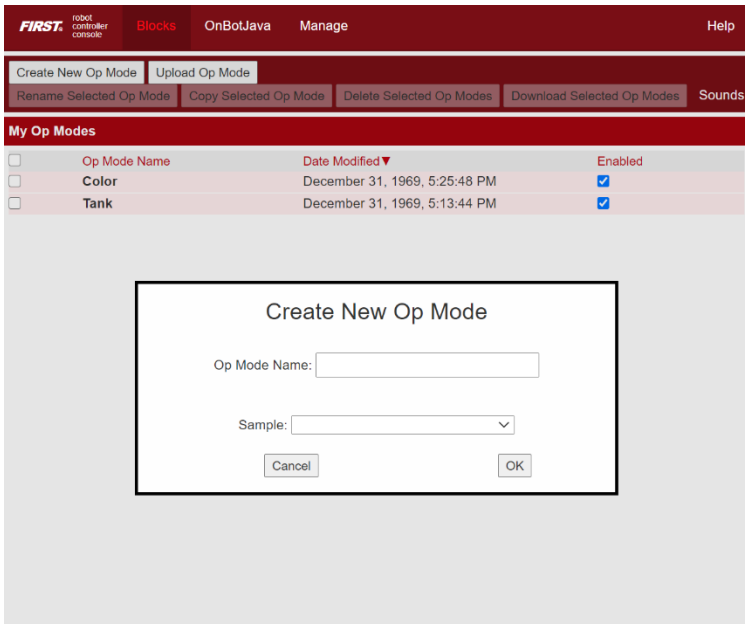
Use the Program & Manage Screen to identify the wireless network name and password. Not the IP address you will use in the next step.

Step 3



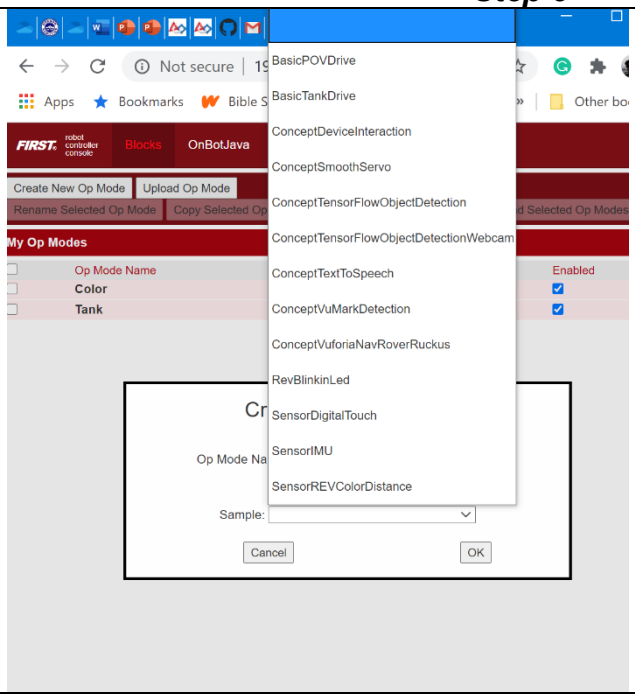
On your computer, access the control hub wireless network using your wireless option. Open an internet browser and type in the IP address listed in the Program & Manage screen. This can be opened with anything with a web browser.

Step 4



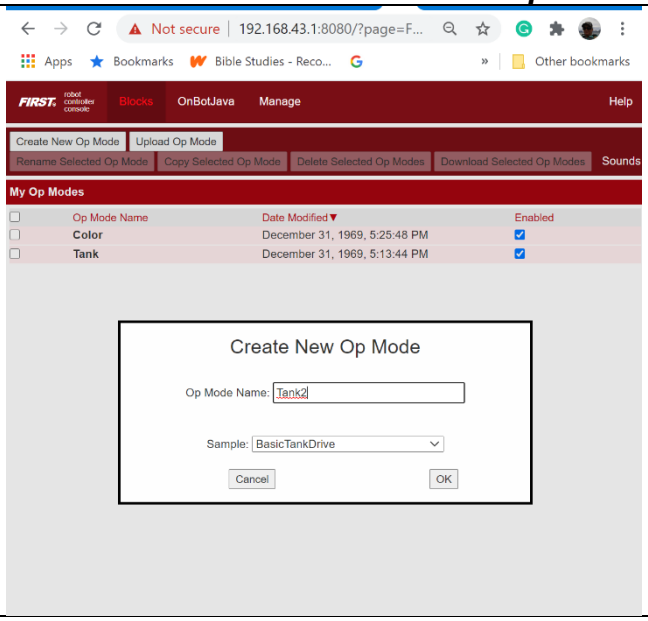
Select Blocks from the top menu. Then choose Create New Op Mode.

Step 5



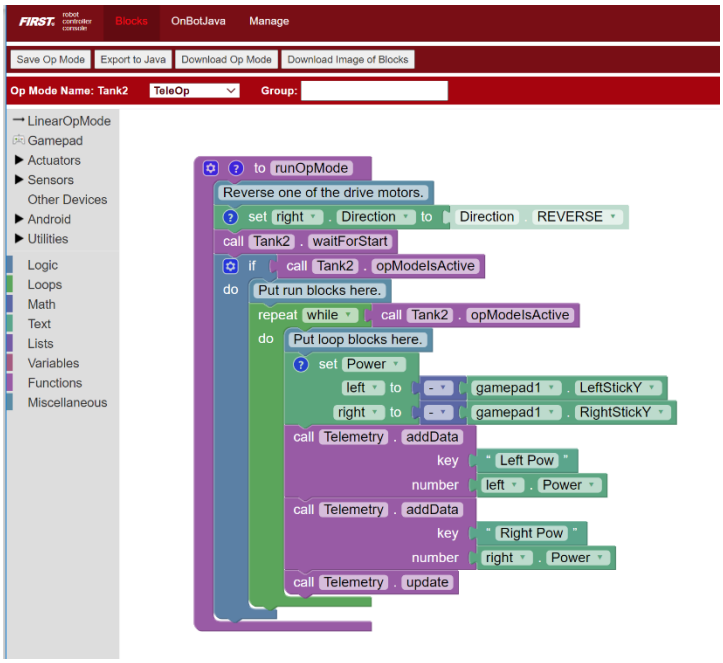
Select “Basic Tank Drive” from the Sample drop-down menu.

Step 6



Create a name for the program and click the OK button.

Step 7



Explore the program and consider how it is receiving input from the joystick and where it is sending the data.
Note: that the right motor power is inversed in the program. This is because the motor is mounted in the opposite direction so the power must be reversed so that it runs in the same direction as the left motor in the program.
Set Power of the left motor to - LeftStickY.
Set Power of the right motor to - RightStickY.

Save the Op Mode once you have explored the program.

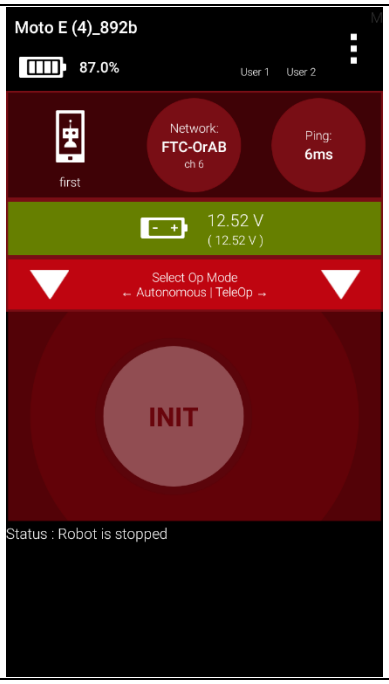
Step 8

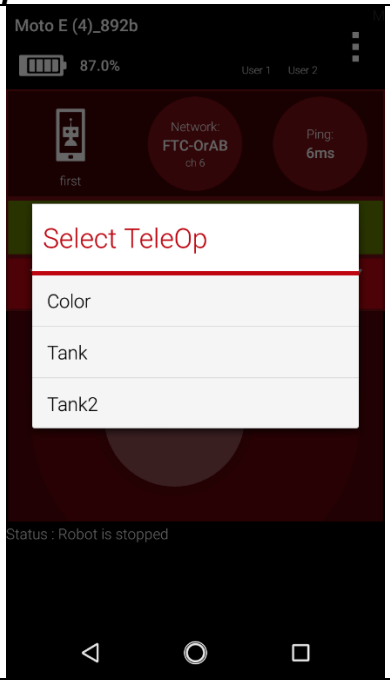
Make sure that your driver's station phone is connected to the gamepad.

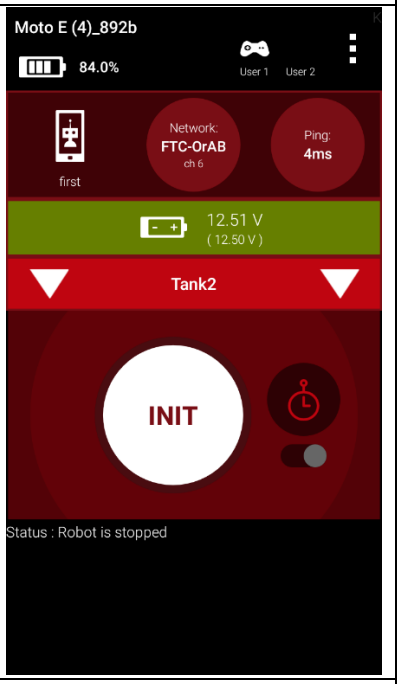
Open the FTC Driver Station App

Click Start A on the gamepad. This signals the app to recognize the gamepad. You should see an icon light up with User 1 when it is connected.

Run the Op Mode on the Robot



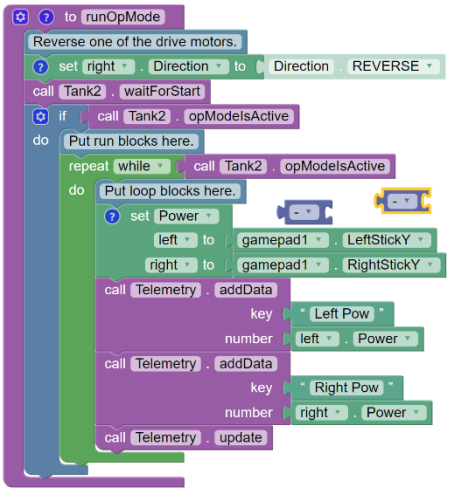




Steps to run the Op Mode on the robot.

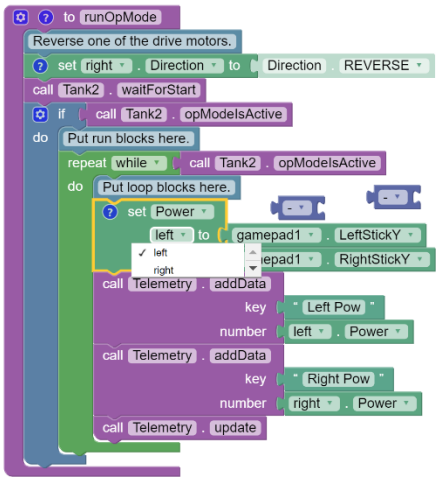
1. Select the right downward facing triangle for TeleOp mode on the driver station.
2. Select the Op Mode that you just created and saved.
3. Check to make sure that the gamepad is recognized, which is indicated by User 1 joystick becomes white.
4. Select the INIT button to initialize.
5. Select the center triangle to Play
6. Drive, test, and record how inputs are affecting robot actions.
7. Select the center square to stop the program.
8. Make changes, as necessary, to the program. Save and repeat the processes above until the robot actions are achieved.

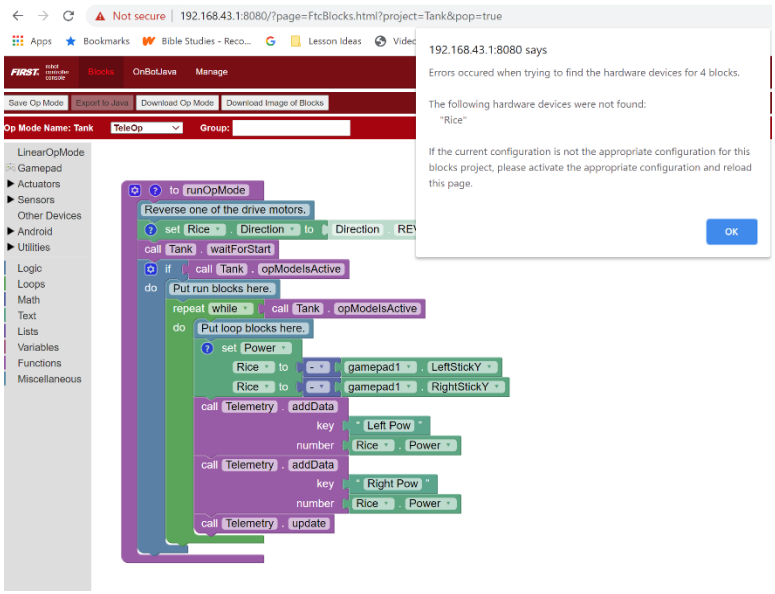
Step 9 Modifying the Program



You may find that the template does not specifically fit the way you would like to control your robot.

You can make changes to fit your needs, such as removing the negative values on the joysticks to invert the values and how the data is processed.

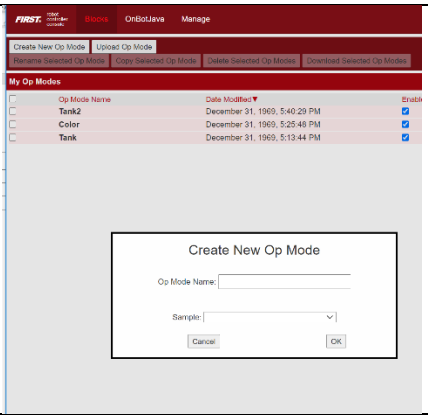
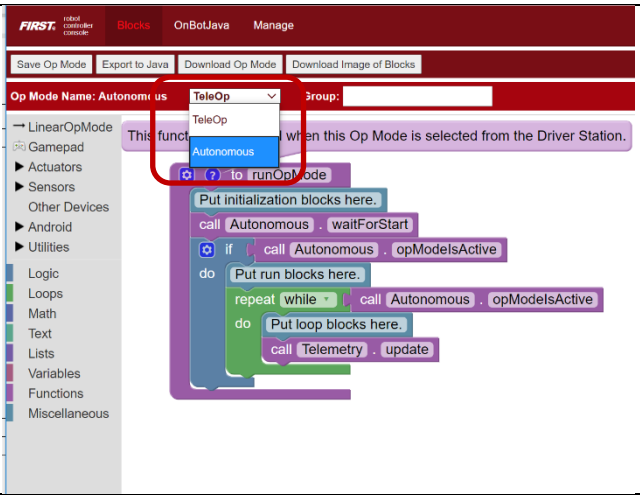
<table>
<tr>
<td> 
</td>
<td>You can also change the motors that are being controlled by the LeftSticky or Rightstick Y to fit your desired driving style.

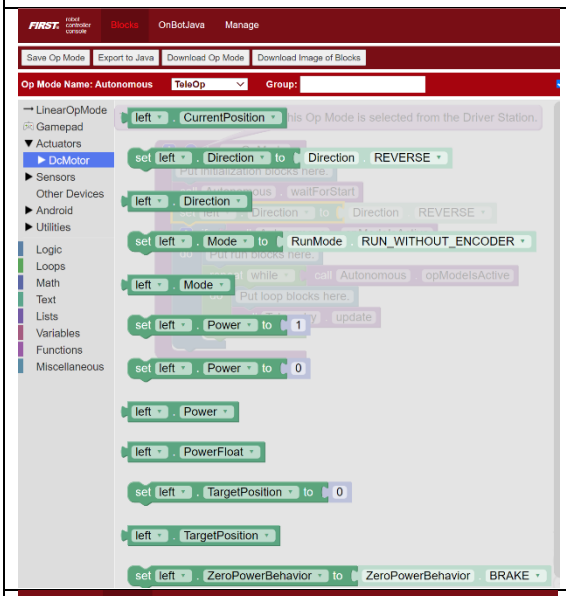
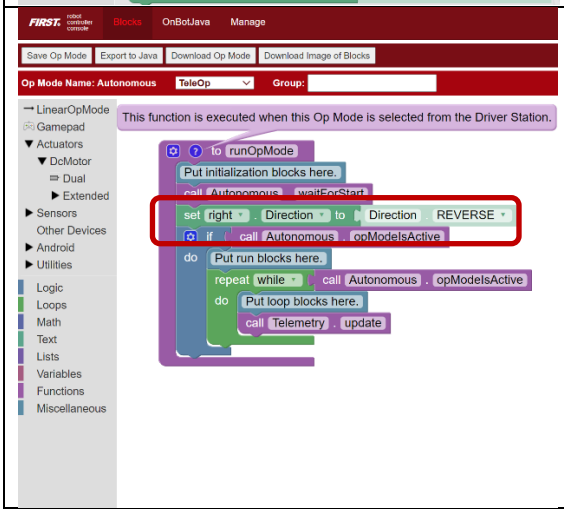
Save the OpMode and follow the steps above for running the Op Mode.</td>
</tr>
<tr>
<td> 
</td>
<td>Note: that if you use programs that have been saved with a different configuration file, then you will get an error.

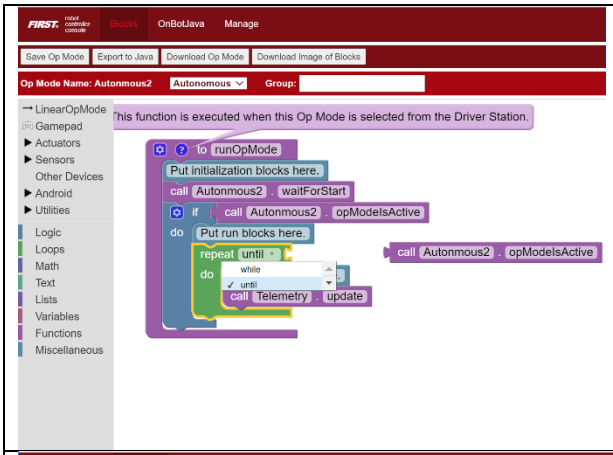
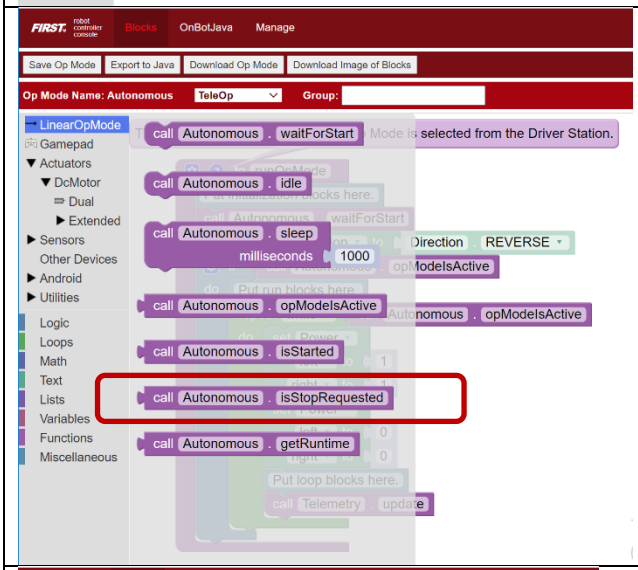
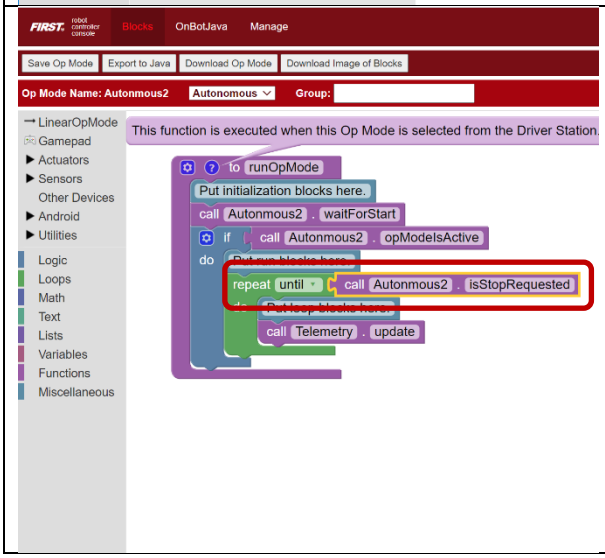
You can fix the error by choosing the correct motor from above that matches the configuration file.

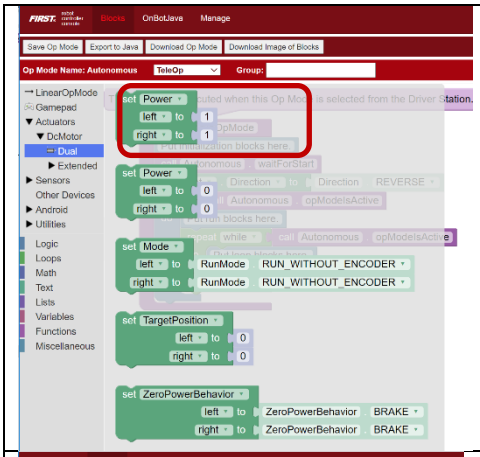
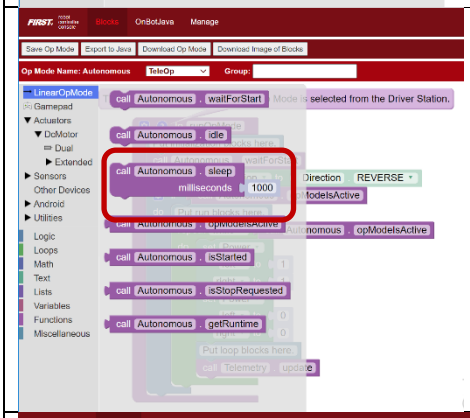
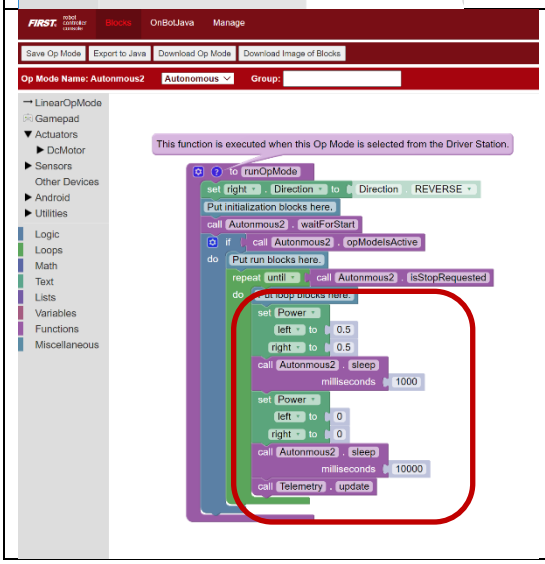
You either have to use a single configuration file or change the program to meet the current configuration.</td>
</tr>
</table>

Creating A Basic Autonomous Mode

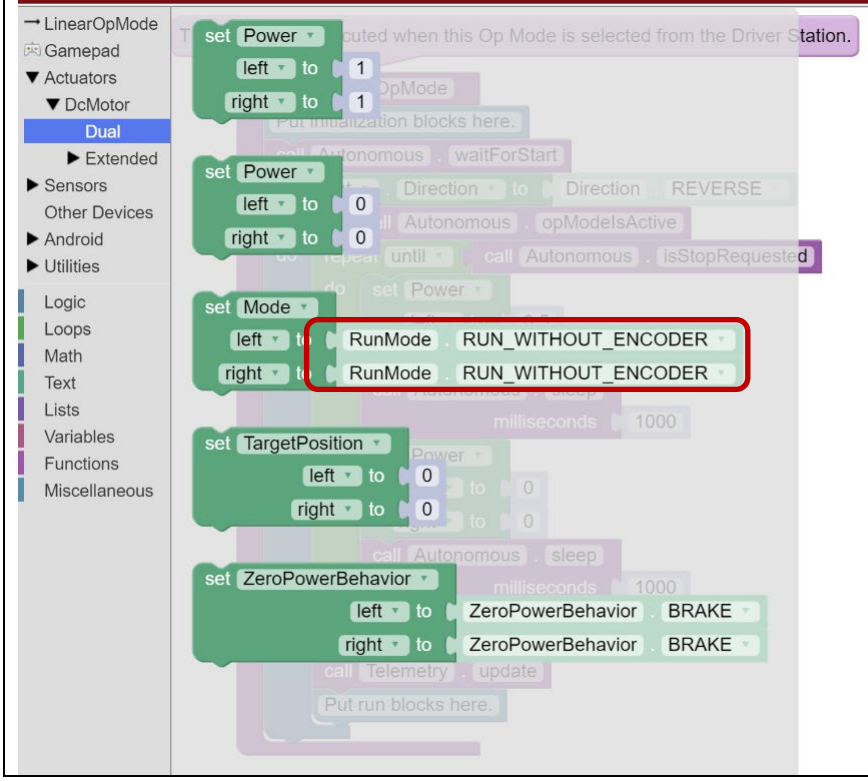
	Configure your robot
	Create a New Op Mode and name it “Autonomous Mode” Select OK.
	Learn about encoders
	Select Autonomous from the drop-down menu. NOTE: Selecting the Autonomous option will require the program to be 30 seconds or less.

Write the PSEUDOCODE 1. *Repeat until the stop button is pressed.* 2. *Set left motor power to 1.* 3. *Set right motor power to 1.* 4. *Wait 1 second.* 5. *Set left motor power to 0.* 6. *Set right motor power to 0.*	Write out the pseudocode or list of actions for the autonomous mode. For example: Drive forward 1 foot and stop.
	Explore the DC motor tools in the Blocks programming environment. Refer to the Tank Drive OpMode you created. You will need to start the program by setting the direction of the right motor to reverse. Select and drag the "set left. Direction to Direction Reverse" block.
	Drag the block into the loop, before the blue "if…do" statement. Then change the drop-down motor from "left" to "right".

	Notice that the green “repeat”loop can be adjusted from “while” to “until.”
	Go to the Linear OpMode menu and select:“callAutonomous.isStopRequested”
	Attach the “callAutonomous.isStopRequested” to the “repeat until” loop. This will ensure that the loop runs until the Stop button is pushed. This will prevent the robot from damaging itself by repeating the loop continually.

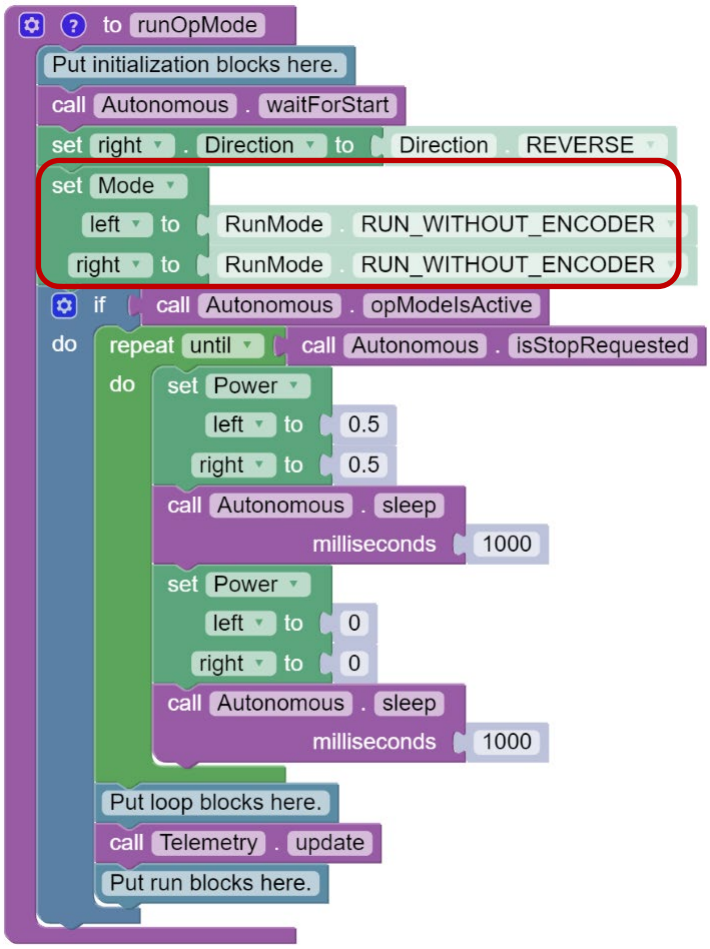
	Next, set your motor power by choosing: Actuators, DCMotor, and then Dual. Then select “SetPower” block with the variable of 1. Change the power level to 0.5 for each motor.
	Now if you refer to your pseudocode the next step is to Add a “wait” block. Select the LinearOpMode from the left menu. Find and select the “callAutonomous.sleep Milliseconds-1000” block.
	Use the previous steps to add “set the motor power to 0” from the DC Motor menu and an additional “callAutonomous.sleep Milliseconds-1000” block from the LinearOpMode. Place the blocks according to the image.

Using Encoders

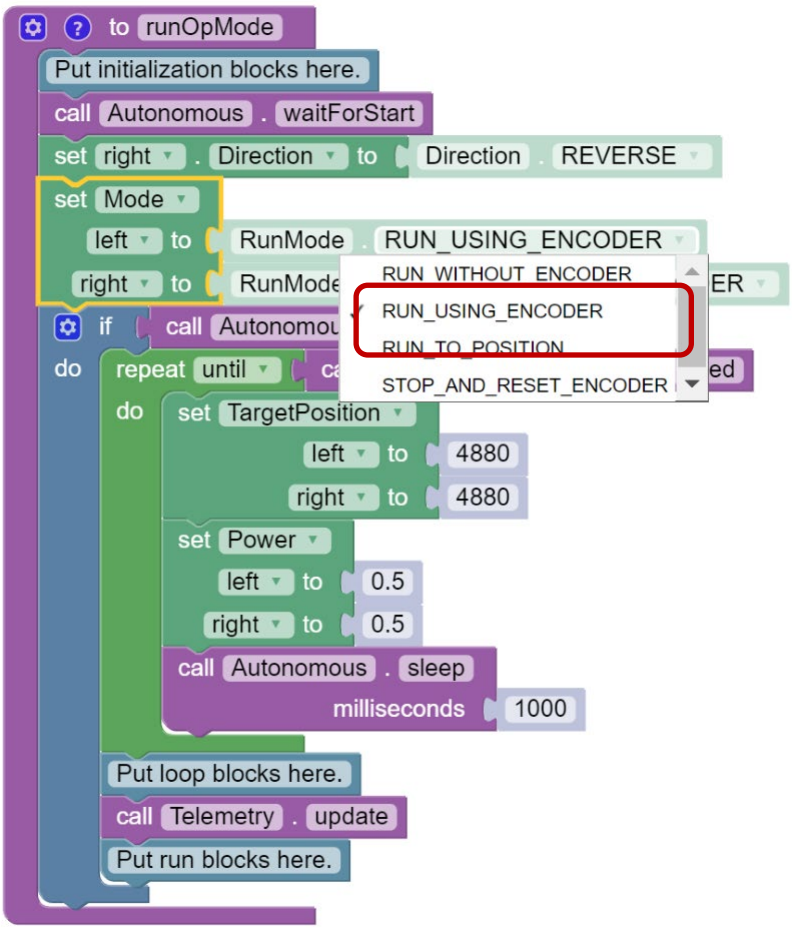


Encoders are a sensor built into the motor that measures the rotation of a motor shaft with pulses. Using encoders can allow your robot to be more reliable and consistent.

In Blocks, they are found in the drop-down menu for motor controls under Actuators, DC Motor.
Select the "Set Mode
left to RunMode
RUN_WITHOUT _ENCODER
right to RunMode
RUN_WITHOUT_ENCODER"



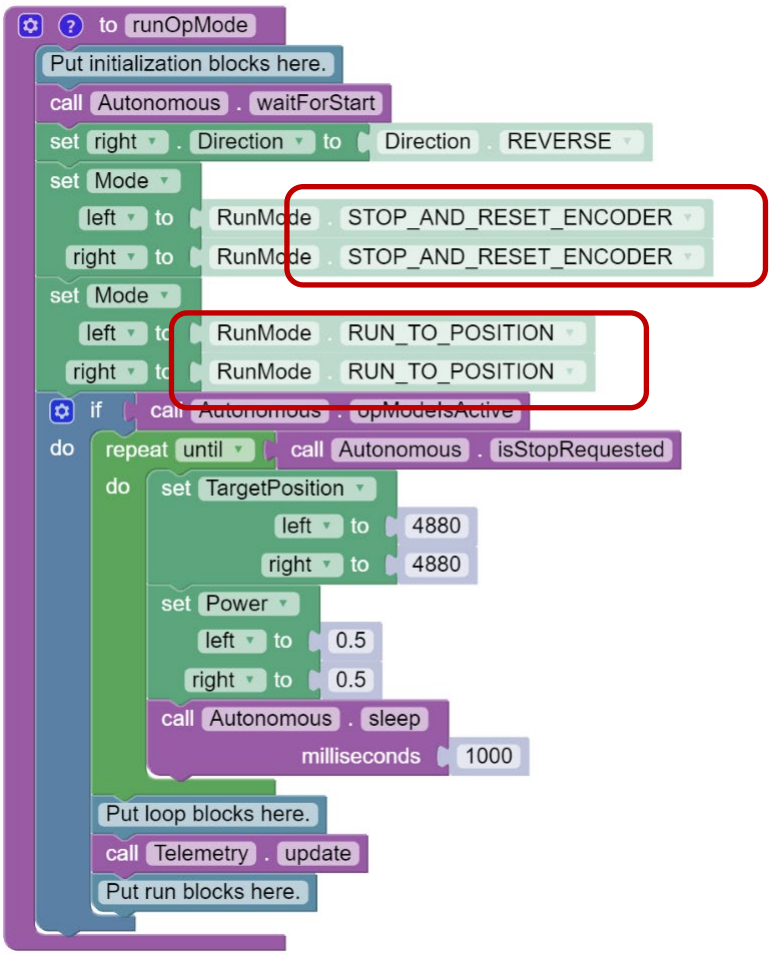
Drag the block under the "Setright.Direction to Direction.REVERSE"



Then choose the option to "RUN_USING_ENCODER" from the drop-down menu.

Choose it for both left and right motors in the block.

Running to position



You can further use encoders to run to a specific position. This will allow you to specifically adjust.

To do this you will need first need to change the first mode for both motors to "STOP_AND_RESET_ENCODER."

This will reset the encoder to 0. Then duplicate the block and change both motors to RUN_TO_POSITION

Then go to the DC Motors, Dual drop- down menu and select the "setTargetPosition" block.

In this example, it will move to 4880 at 50 percent power. Then it will sleep 1000 milliseconds.

If you want the program to stop here for the remainder of the autonomous, change the sleep to 300000 which is around 30 seconds.

Save the Op Mode, the run on the Driver Station.

Appendix A – Resources

Game Forum Q&A

https://ftcforum.firstinspires.org/
Anyone may view questions and answers within the *FIRST*® Tech Challenge game Q&A forum without a password. To submit a new question, you must have a unique Q&A system username and password for your team.

FIRST Tech Challenge Game Manuals

Part 1 and 2 - https://www.firstinspires.org/resource-library/ftc/game-and-season-info

FIRST Headquarters Pre-Event Support

Phone: 603-666-3906
Mon – Fri
8:30am – 5:00pm
Email: Firsttechchallenge@firstinspires.org

FIRST Websites

FIRST homepage – www.firstinspires.org

FIRST Tech Challenge Page – For everything *FIRST* Tech Challenge.

FIRST Tech Challenge Event Schedule – Find *FIRST* Tech Challenge events in your area.

FIRST Tech Challenge Social Media

FIRST Tech Challenge Twitter Feed - If you are on Twitter, follow the *FIRST* Tech Challenge Twitter feed for news updates.

FIRST Tech Challenge Facebook page - If you are on Facebook, follow the *FIRST* Tech Challenge page for news updates.

FIRST Tech Challenge YouTube Channel – Contains training videos, game animations, news clips, and more.

FIRST Tech Challenge Blog – Weekly articles for the *FIRST* Tech Challenge community, including outstanding volunteer recognition!

FIRST Tech Challenge Team Email Blasts – contain the most recent *FIRST* Tech Challenge news for teams.

Feedback

We strive to create support materials that are the best they can be. If you have feedback about this manual, please email firsttechchallenge@firstinspires.org. Thank you!