

# FTC Training Document

---

## Blocks Programming Training Manual

Document Version 1.0

*FIRST Tech Challenge*

11/3/2016

This document contains training material to introduce students and mentors how to write programs for the FIRST Tech Challenge Robot Controller platform using the blocks programming tool that is included with the FTC software.



Revision History		
Revision	Date	Description
1.0	11/3/2016	Initial Release

## Contents

1	Introduction	4
1.1	The <i>FIRST</i> Tech Challenge Control System	4
1.2	What's an Op Mode?	5
1.3	The Blocks Programming Tool	5
2	Getting Started	6
2.1	System Requirements	6
2.2	Naming Your Smartphones	6
2.3	Installing the <i>FIRST</i> Tech Challenge Apps	10
2.4	Forgetting the External Wireless Network	12
2.5	Placing the Phones into Airplane Mode with Wi-Fi On	13
3	Using the FTC Apps	15
3.1	Pairing the Driver Station to the Robot Controller	15
3.2	Configuring the Robot Controller	18
3.2.1	Connecting the Robot Hardware to the Robot Controller Smartphone	18
3.2.2	Associating the Motor Controller with the Robot Controller App	20
3.2.3	Creating a Simple Configuration File	20
3.3	Writing Your First Op Mode	24
3.3.1	Installing a Javascript-enabled Browser	24
3.3.2	Putting the Robot Controller into Blocks Programming Mode	25
3.3.3	Manually Connecting to the Blocks Programming Mode Wi-Fi Network	29
3.3.4	Access the Blocks Programming Mode Server with your Browser	32
3.3.5	Creating a New Op Mode	32
3.3.6	Inserting Telemetry Statements	42
3.3.7	Saving Your Op Mode	45
3.3.8	Exiting Programming Mode	46
3.3.9	Connecting a Gamepad to Your Driver Station	46
3.3.10	Selecting and Running Your Op Mode	48
4	Controlling a Servo Motor with an Op Mode	51
4.1	What is a Servo Motor?	51

4.2	Adding a Servo Controller and Servo to the Test System	52
4.3	Configuring the Servo Controller	54
4.4	Switching the Robot Controller into Programming Mode	56
4.5	Mapping Joystick Position to Target Servo Position	58
4.6	Modifying Your Op Mode	60
4.7	Adding Telemetry Blocks to Display Position Information	66
4.8	Save Your Edited Op Mode!	67
5	Using Sensors	67
5.1	What is a Sensor?	67
5.2	Connecting Your Sensors	67
5.3	Modifying the Configuration File	69
6	Sample Op Modes	79
6.1	Basic Tank Drive	79
6.2	Basic Line Following	80
7	Additional Tips and Information	81
7.1	Troubleshooting Tips	81
7.1.1	Cannot See the Blocks Programming Mode Wireless Network	81
7.1.2	“Save project failed. Error code 0.”	81
7.1.3	Op Mode Blocks Are Missing...	82
7.1.4	Driver Station Appears Unresponsive	83
7.1.5	Warning: problem communicating...	84

# 1 Introduction

The *FIRST* Tech Challenge (FTC) uses an Android-based control system for its competition robots. Students who participate in the *FIRST* Tech Challenge can use the *blocks programming* tool to customize the behavior of their competition robots. This document provides an overview of the blocks programming tool and includes instructions on how to create, edit and deploy op modes using this development tool.

## 1.1 The *FIRST* Tech Challenge Control System

The *FIRST* Tech Challenge uses Android phones to control its robots. During a competition, each team has two Android phones.



Figure 3 - The new FTC platform is a point-to-point solution.

One phone is mounted onto the robot and is called the *Robot Controller*. The Robot Controller acts as the “brains” of the robot. It does all of the thinking for the robot and tells the robot what to do.

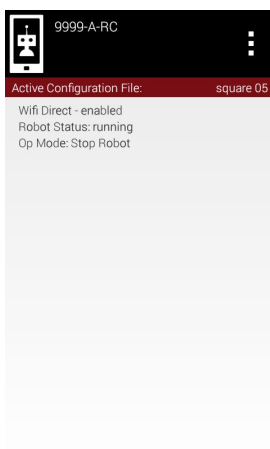


Figure 1 - Robot Controller main screen.

A second phone sits with the team drivers. This second phone is known as the *Driver Station*. The Driver Station is sort of like a remote control that you might use to control your television. The Driver

Station allows a team to communicate remotely (using a secure, wireless connection) to the Robot Controller and to issue commands to the Robot Controller.

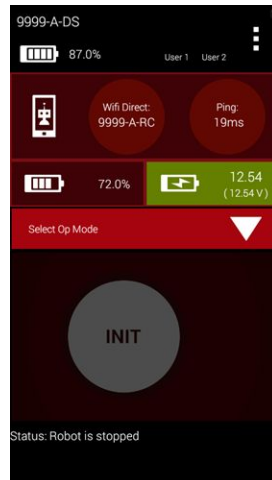


Figure 2 - Driver Station main screen.

## 1.2 What's an Op Mode?

During a typical *FIRST* Tech Challenge match, a team's robot has to perform a variety of tasks in an effort to score points. For example, a team might want their robot to follow a white line on the competition floor and then score a game element into a goal autonomously during a match. Teams write "op modes" (which stand for "operational modes") to specify the behavior for their robot.

Teams who are participating in the *FIRST* Tech Challenge have a variety of programming tools that they can use to create their own op modes. This document explains how to use the blocks programming tool to write an op mode for an FTC robot.

## 1.3 The Blocks Programming Tool

The *blocks programming* tool is a user friendly programming tool that is served up by the Robot Controller phone. A user can create custom op modes for their robot using this tool and then save these op modes directly onto the Robot Controller.

Users drag and drop jigsaw-shaped programming blocks onto a design "canvas" and arrange these blocks to create the program logic for their op mode.

The blocks programming tool is powered by Google's *Blockly* software and was developed with support from Google.

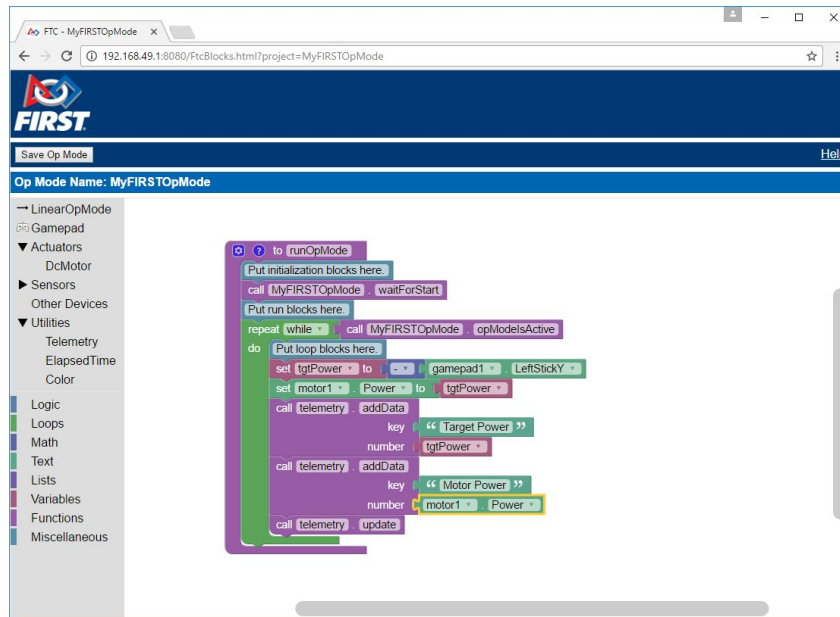


Figure 3 - Users arrange jigsaw-shaped programming blocks to create the logic for their op modes.

## 2 Getting Started

### 2.1 System Requirements

In order to do the exercises that are contained in this document, you will need to make sure you have the following components:

1. A pair of Android smartphones (version Marshmallow or higher... refer to the annual *FIRST* Tech Challenge game manual, part 1, for details on approved phones).
2. A wireless Internet connection for the smartphones (so you can download apps from the Google Play Store).
3. A Google account (which are available for free from Google).
4. A laptop (Windows, Mac or Chromebook) that has a wireless network adapter.
5. FTC-compatible robot hardware. Note that you do not need a fully assembled robot, but you will need a motor controller, a servo controller, a sensor I/O module, and the matching cables to do the exercises in this document.
6. A Logitech F310 gamepad with Micro USB OTG adapter (to connect to the Driver Station phone).

### 2.2 Naming Your Smartphones

The first thing that you need to do before you start programming your system is to rename your Android smartphones. The official *FIRST* Tech Challenge rules (<RS01> in the official Robot Rules, part 1) state that teams must name their Android phones using their FTC Team number followed by a dash (hyphen) and the letters “RC” if the phone is to be used as a Robot Controller or the letters “DS” if the phone is to be used as a Driver Station. A team can insert an additional dash and a letter (“A”, “B”, “C”, etc.) if the team has more than one set of Android phones.

If, for example, a team has a team number of 9999 and the team has multiple sets of phones, the team might decide to name one phone “9999-C-RC” for the Robot Controller and the other phone “9999-C-DS” for the Driver Station.<sup>1</sup>

You can rename your phone through the Android Settings of the phone:

1. Click on the “Settings” icon on your Android phone to display the Settings screen.

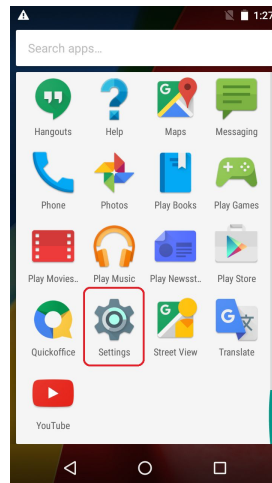


Figure 4 - Click on Settings icon to launch Settings screen.

2. On the Settings screen, click on the “Wi-Fi” item to launch the Wi-Fi Screen.

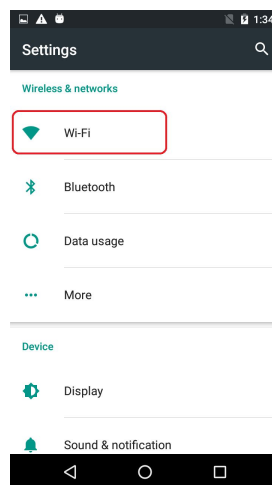


Figure 5 - Click on "Wi-Fi" to launch Wi-Fi Screen.

3. On the Wi-Fi screen, touch the three dots in the upper right hand of the screen to display a pop-up menu. Select “Advanced” from the pop up menu.

---

<sup>1</sup> Note that the screens on your phone might differ slightly from the screens shown in this document.



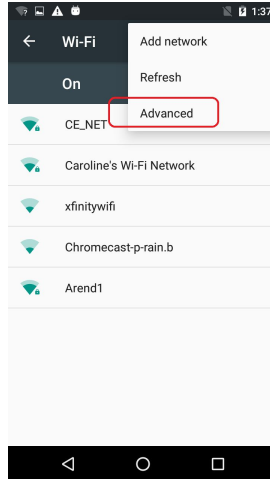


Figure 6 - Click on "Advanced" to display the Advanced Wi-Fi screen.

4. On the Advanced Wi-Fi screen, select "Wi-Fi Direct" to display the Wi-Fi Direct Menu.

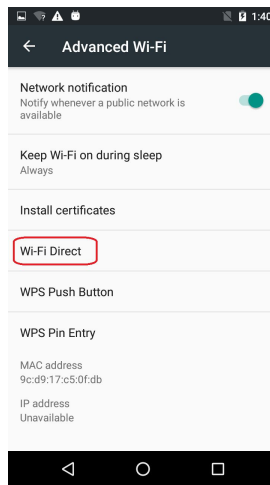


Figure 7 - Click on "Wi-Fi Direct" to display the Wi-Fi Direct Screen.

5. On the Wi-Fi Direct screen, click on the three dots in the upper right hand corner to display a pop up menu. Select "Configure device" from pop-up menu.

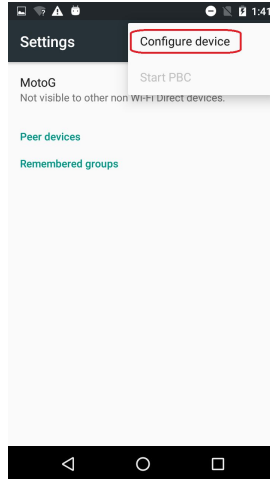


Figure 8 - Select "Configure device" from the pop up menu.

6. In the Configure device screen, use the touch keypad to type in the new name of your device. You can also set the Wi-Fi Direct Inactivity timeout to "Never disconnect" and then hit the "SAVE" button to save your changes.

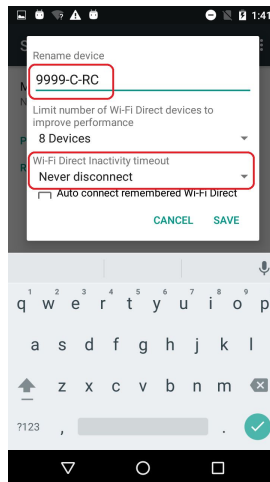


Figure 9 - Use keypad to type in new name, set timeout to "Never disconnect" then push "SAVE" to save changes.

7. After you have renamed your phone, power cycle (i.e., restart) your phone.

You will want to rename both your Robot Controller and Driver Station phones so that they comply with the *FIRST* Tech Challenge naming convention. For instance, if your team number is "9999", then your Robot Controller Android device would be named "9999-RC" and your Driver station Android device would be named "9999-DS" (assuming that you only have one set of phones).

After you have renamed your phones, you might want to consider labeling the exterior of the phones (using a label maker or some masking tape and a marker) with their respective names so that you can easily distinguish between each phone even when the phones are powered off.

## 2.3 Installing the *FIRST* Tech Challenge Apps

Before you can begin to program your system, you first need to install the FTC Robot Controller and FTC Driver Station apps onto your phones. The FTC apps are available to download for free from the Google Play store. You will need to have your Android phones connected to a WiFi network that has Internet access before you can access the Google Play store. You will also need a Google account to be able to download the apps from the Google Play store.

1. Go to the Settings screen of Android and launch the Wi-Fi screen.
2. Find the name of your wireless network (“CE\_NET” in the example below) and touch on the wireless network to connect to it.

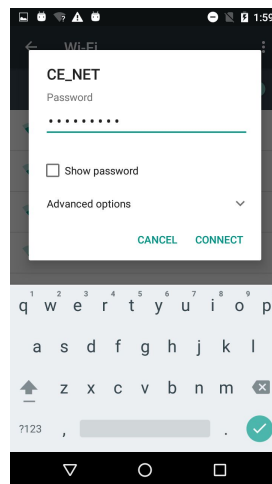


Figure 10 - Find your network in the list of available networks, tap on it and provide the password to connect.

3. Find the Google Play icon on your smartphone and click it to launch the Google Play app.

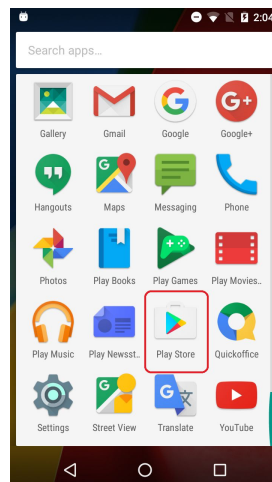


Figure 11 - Launch the Google Play Store app.

4. If you haven't signed in to your Google account yet, follow the onscreen instructions to login to your Google account.

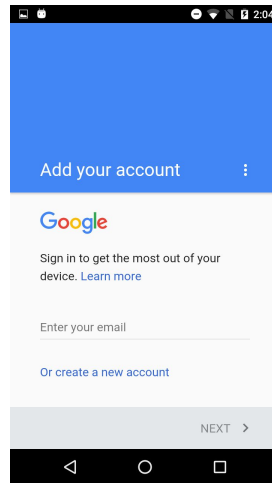


Figure 12 - Follow onscreen instructions to login to your Google account.

5. In the search window of the Google Play app, type in the words "FTC Robot Controller" to find the Robot Controller or "FTC Driver Station" to find the appropriate FTC app for your phone.

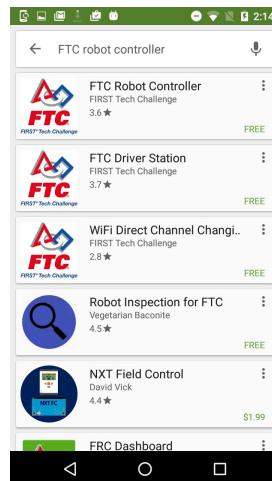
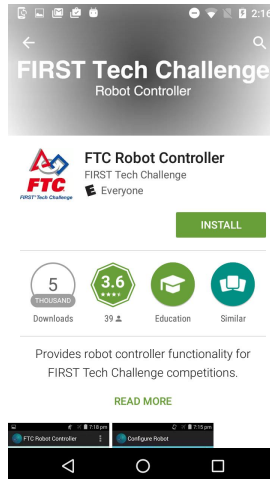


Figure 13 - Type in "FTC Robot Controller" or "FTC Driver Station" to find the appropriate app for your phone.

6. Tap on the app in the Google Play listing to bring up the installation screen. Follow the onscreen instructions to install the appropriate app for your phone.



**Important note:** When you install the FTC apps, only install one FTC app (FTC Robot Controller or FTC Driver Station) per phone. You should avoid installing both apps onto the same phone. Doing so can cause Wi-Fi connection problems. You should only install the FTC Robot Controller app onto the phone that will be the Robot Controller and the FTC Driver Station app onto the phone that will be the Driver Station.

## 2.4 Forgetting the External Wireless Network

After you have successfully downloaded and installed the appropriate FTC app onto your phone, you should disconnect from the wireless network that you used to access Google Play. In general, your FTC phones should not be connected to any external network. You typically only want to use the wireless adapters on the phones to connect the phones to each other (Driver Station to Robot Controller).

You should use the Wi-Fi Settings screen of the Android phone to forget the external wireless network:

1. Go to the Settings screen of the Android phone and click on “Wi-Fi” to launch the Wi-Fi screen.
2. On the Wi-Fi screen, find the name of your connected network and tap on the network to bring up a pop up box with information about your network.

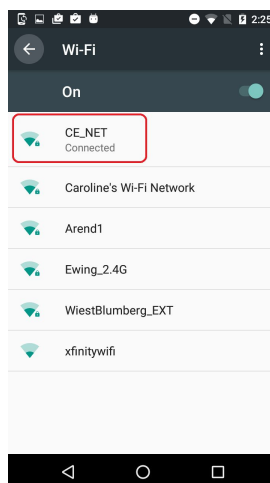


Figure 14 - Tap on the listing for the connected network to display a pop up box with info about the network.

3. On the pop up box, click on “FORGET” to forget the wireless network.

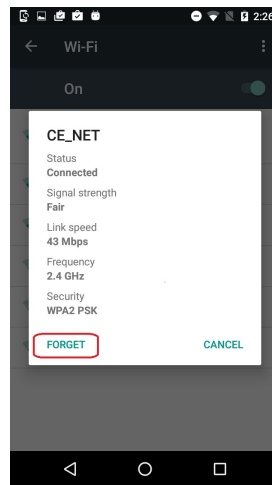


Figure 15 - Click on "FORGET" to forget the network.

## 2.5 Placing the Phones into Airplane Mode with Wi-Fi On

For the *FIRST* Tech Challenge competitions, it is important that you place your Robot Controller and Driver Station phones into Airplane mode but keep their Wi-Fi radios turned on. This is important because you do not want any of the cellular telephone functions to be enabled during a match. The cellular telephone functions could disrupt the communication between the Robot Controller and Driver Station and could interrupt the function of the robot during a match.

For Motorola phones (the process is similar on other Android phones) if you would like to switch your phone to Airplane mode, from the main Android screen, use your finger to slide from the top of the screen down towards the bottom of the screen to display the quick configuration screen.

Look for the “Airplane mode” icon (which is shaped like an airplane) and if the icon is not activated, touch the icon to activate it. When the icon is activated it will be brighter than when the icon is disabled. When the Airplane mode icon is activated, the cellular telephone functions of the phone are disabled.

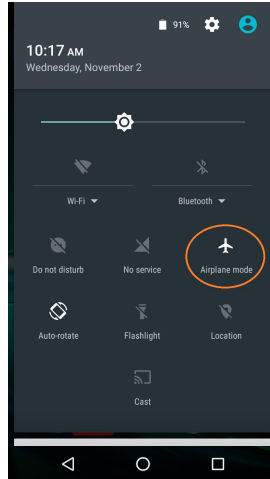


Figure 16 - Touch the Airplane mode icon to put phone in Airplane mode.

Placing the phone into airplane mode will turn off the Wi-Fi radio. If the Wi-Fi icon has a diagonal line through it (see Figure 16 above), then the Wi-Fi radio is disabled. You will need to touch the “Wi-Fi” icon on the quick configuration screen to turn the Wi-Fi radio back on.

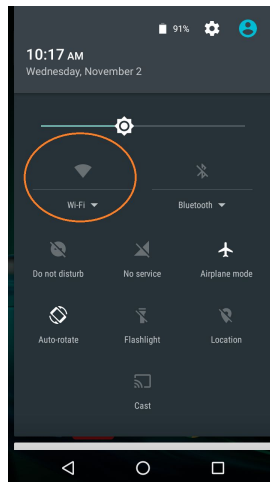


Figure 17 – After touching the Wi-Fi icon, the Wi-Fi radio should turn on and the diagonal line should disappear.

## 3 Using the FTC Apps

### 3.1 Pairing the Driver Station to the Robot Controller

Once you have successfully installed the FTC apps onto your Android phones, you will want to establish a secure wireless connection between the two devices. The process to connect the two phones is known as “pairing.”

To pair your phones, first launch the Robot Controller app:

1. Find the FTC Robot Controller icon.

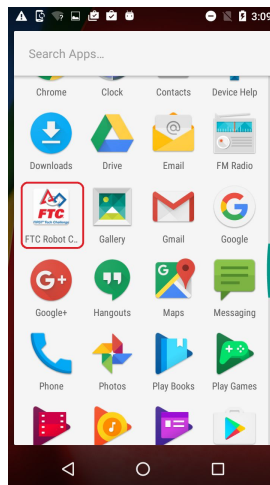


Figure 18 - Find and tap on the FTC Robot Controller icon.

2. Tap on the icon to launch the FTC Robot Controller. The main screen for the Robot Controller app should appear.

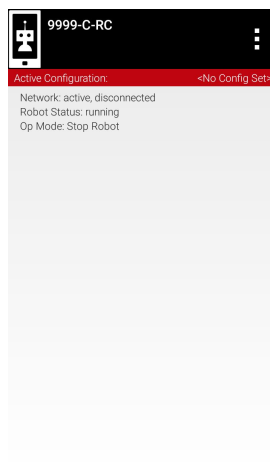
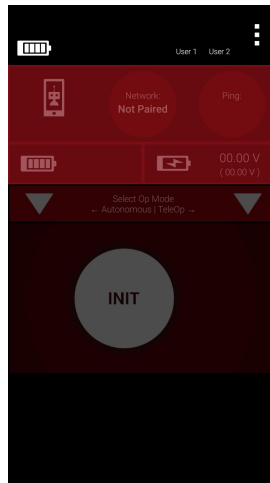


Figure 19 - The main Robot Controller screen should appear.



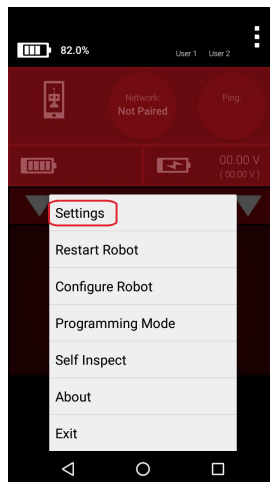
Similarly, on the Driver Station phone, find the FTC Driver Station icon and tap on it to launch the main FTC Driver Station screen.



**Figure 20 - User should find and touch the FTC Driver Station icon to launch the app.**

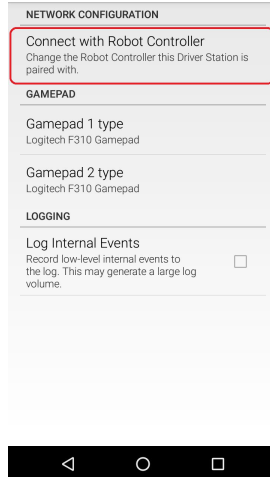
Once you have launched the Driver Station app, you can initiate a connection request to the Robot Controller from the Driver Station:

1. Touch the three dots in the upper right hand corner of the FTC Driver Station main screen. This should display a pop up menu. Select “Settings” from the pop up menu.



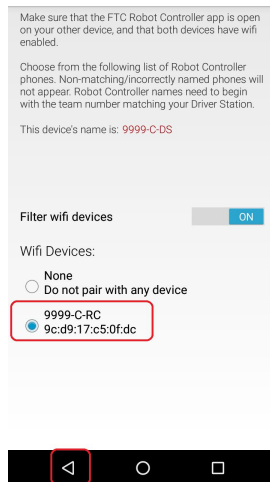
**Figure 21 - Select "Settings" from the pop up menu on the Driver Station.**

2. From the Settings screen, look for and select “Connect with Robot Controller” to launch the Connect with Robot Controller screen.



**Figure 22 - Select "Connect with Robot Controller" to launch the Connect with Robot Controller screen.**

3. Find the name of your Robot Controller from the list and select it. After you have made your selection, use the back arrow key to return to the Settings screen. Then press the back arrow key one more time to return to the main Driver Station screen.



**Figure 23 - Select your Robot Controller from the list and use the back arrow to return to the main screen.**

4. When the Driver Station returns to its main screen, the first time you attempt to connect to the Robot Controller a prompt should appear on the Robot Controller screen. Click on the "ACCEPT" button to accept the connection request from the Driver Station.

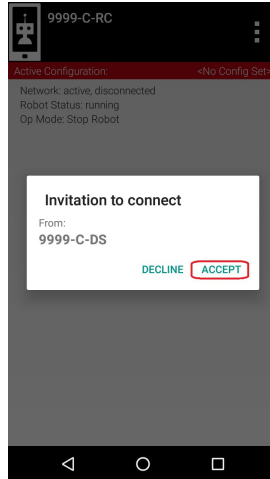


Figure 24 - Press "ACCEPT" on the Robot Controller to accept the connection request from the Driver Station.

5. If the Driver Station successfully establishes a connection to the Robot Controller, the screens of both devices will change to reflect the new connection status.

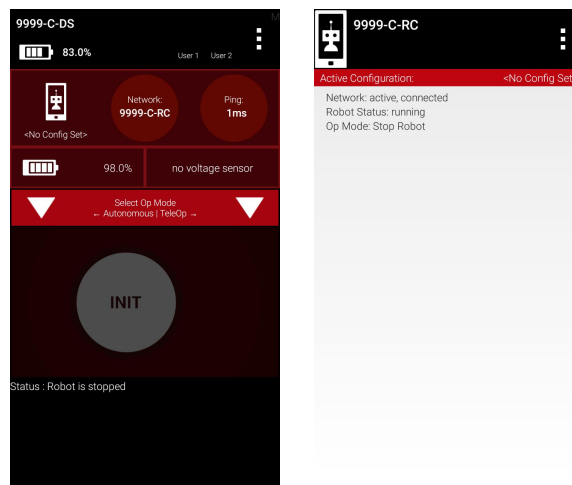


Figure 25 - The Driver Station and Robot Controller screens should indicate that they are connected to each other.

## 3.2 Configuring the Robot Controller

### 3.2.1 Connecting the Robot Hardware to the Robot Controller Smartphone

Now that Driver Station has been paired to the Robot Controller, you are ready to physically connect the Robot Controller to a device that is called a *motor controller*. As its name implies, the motor controller is used to control DC motors. The Robot Controller is able to send commands to the motor controller and the motor controller uses the commands to drive the motors by providing power to its *motor ports*.

#### 3.2.1.1 Required Components

In this example, you will need the following components:

1. FTC-approved 12V battery.

2. Modern Robotics Power Distribution Module.
3. Modern Robotics Motor Controller.
4. FTC-approved 12V DC motor and 12V motor power cable.
5. USB cables and Micro USB OTG adapter.
6. 12V power cable (with red and black Anderson Powerpoles on both ends).
7. Robot Controller and Driver Station smartphones.

### 3.2.1.2 Connecting Your Devices

Connect your devices as shown in Figure 26 (see below).

1. Use the Micro USB OTG adapter and a USB cable to connect the phone to the input USB port on the PDM.
2. Use a USB cable to connect the PDM to the motor controller. Note the port number that you connect to on the PDM is not important. The phone should be able to find the Motor Controller on any of the USB ports of the PDM.
3. Use the 12V power cable to connect the 12V input power port on the motor controller to a 12V output port on the PDM. The 12V power cables have the red and black Anderson Powerpole connectors on both ends.
4. Use the 12V motor power cable to connect the DC motor to port #1 (labeled “MOTOR 1”) on the motor controller.
5. Make sure the on/off switch on the PDM is turned OFF. Connect the 12V battery to the 12V battery cable on the PDM.
6. If everything is properly connected, you can turn on the Power Supply.

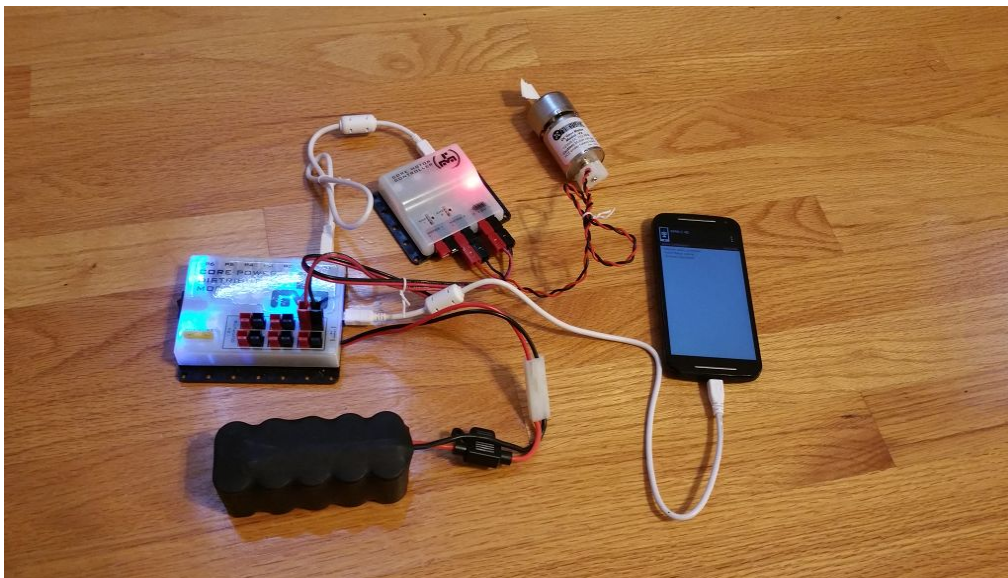


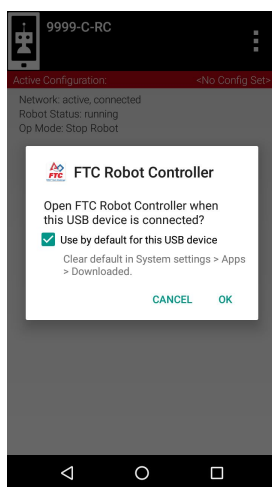
Figure 26 - Connect your devices as shown in this figure.

**IMPORTANT NOTE:** If you need to turn off your robot hardware, the manufacturer recommends that you keep the Power Distribution and other Core Control modules (such as the motor controller) turned off for a full 35 second before turning the system back on.

**IMPORTANT NOTE:** In Figure 26, the DC motor is not mounted to a stable structure. When setting up your test hardware, you should consider mounting the DC motor to a stable structure (for example, a small metal frame built from the kit of parts that includes a proper DC motor mount). A stable mount will prevent the motor from moving uncontrollably when a user is driving it.

### 3.2.2 Associating the Motor Controller with the Robot Controller App

The first time you connect the Robot Controller phone to the Power Distribution Module (or the FIRST time you power on the PDM while it's connected to the phone) you should see a prompt from the Android operating system asking you if it is OK to associate the newly discovered USB-connected hardware (i.e., the motor controller in our example) with the FTC Robot Controller app.



**Figure 27 - Always check the "Use by default" option and click "OK" to associate the device with the Robot Controller app.**

Whenever you are prompted by your phone with this message, you should always select the “Use by default for this USB device” option and hit the “OK” button to associate the USB device with the FTC Robot Controller app. If you do not do this, then the Robot Controller app might not reliably connect to this USB device whenever the phone is connected to the Power Module’s USB hub.

### 3.2.3 Creating a Simple Configuration File

In order to be able to program and control our robot hardware, we have to create a configuration file for the Robot Controller, so that the op modes that we create later on will know what hardware is connected to the phone.

To create a configuration, on the Robot Controller or Driver Station, touch the three dots in the upper right hand corner to display the pop up menu. Select from the pop up menu the “Configure Robot” option to display the Configuration screen. Note that while the USB device association must be performed on the Robot Controller phone, the configuration can be done through the user interface of

the Robot Controller or the Driver Station. Note that the configuration file will physically reside on the Robot Controller once it has been created and saved.

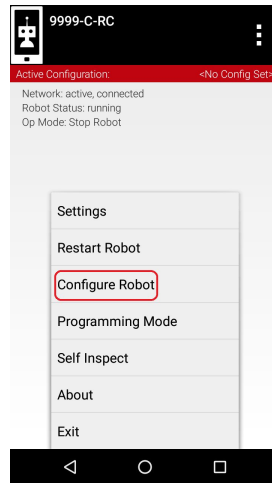


Figure 28 - Select the Configure Robot option from the pop up menu.

If your Robot Controller does not have any existing configuration files, the screen will display a message indicating that you need to create a file before proceeding. Hit the “New” button to create a new configuration file for your Robot Controller.

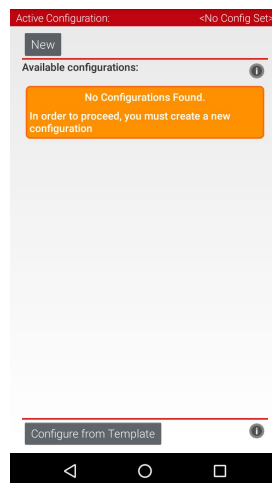
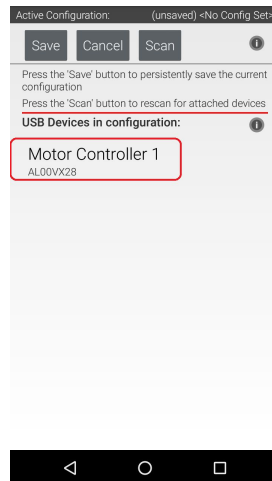


Figure 29 -Press the “New” button to create a new configuration file for your Robot Controller.

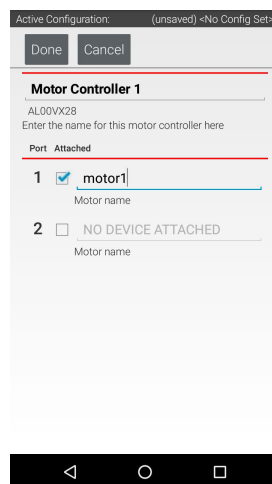
When the new configuration screen appears, the Robot Controller app will do a scan of the USB bus to see what devices are connected to the phone. It will display the devices that it found in a list underneath the words “USB Devices in configuration.” Note that you have a device that you believe is connected to the phone, but this device is not listed on the screen, press the “Scan” button a few times to see if the phone can detect the device on a re-scan of the USB bus.



**Figure 30 - Devices that were detected during the USB scan will be listed on the Configuration screen.**

In our example we only have one device (the Modern Robotics Motor controller) connected to our phone through the PDM’s USB hub. This device should be listed as “Motor Controller 1” with the device’s serial number also displayed. Touch the screen to display the configuration information for this motor controller.

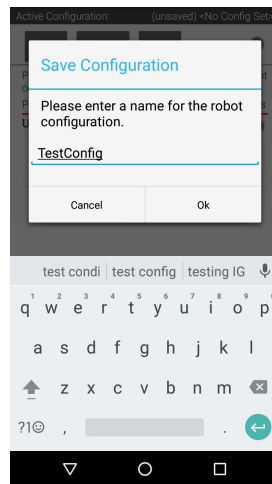
The Robot Controller app is able to scan the USB bus and find the Modern Robotics motor controller on the bus. However, the Robot Controller app is not able to scan the Modern Robotics motor controller to determine what devices are connected to it. The user must manually configure this information for each motor controller that is detected.



**Figure 31 - The user must manually configure the motors that are connected to the motor controller.**

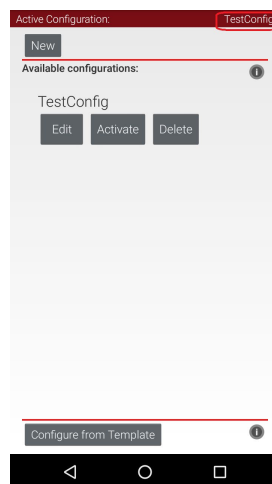
In our example, since you only have a single motor connected to the first port on the motor controller, you only have to configure motor port #1 on the Motor Controller 1 configuration screen. Check the box next to port #1 and type in the name “motor1” next to port #1. This name “motor1” is the name that you will refer to whenever you want to access this motor in your op mode.

Click on the “Done” button to leave the Motor Controller 1 configuration screen. On the main Configuration screen, click the “Save” button and then type in a file name for this configuration and hit “OK” to save the information with this filename.



**Figure 32 - Click on the Save button in the main Configuration screen, and then provide a filename and hit "OK" to save.**

Once you have saved the information, this configuration file becomes the active configuration for your Robot Controller. The name of the active configuration file should appear in the upper right hand corner of the Configuration screen.



**Figure 33 - The filename for the active configuration should be displayed in the upper right hand corner of the screen.**

Press the Android back arrow button to return to the main screen. The active configuration filename should still be visible on the main Robot Controller screen.



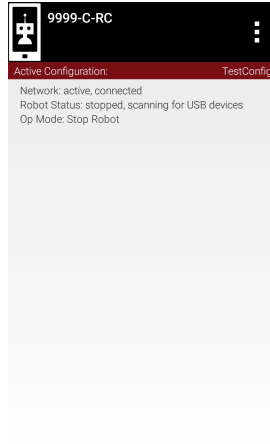


Figure 34 - The active configuration filename should also be visible in the main Robot Controller screen.

### 3.3 Writing Your First Op Mode

You are just about ready to begin writing your first op mode. Before you start, there is one additional step that needs to be performed. You will need to put your Robot Controller into blocks programming mode, and then you will need to connect your laptop to the programming mode server through a secure wireless connection.



Figure 35 - The laptop communicates with the programming server through a wireless connection.

Once the wireless connection between the laptop and the Robot Controller has been established, you can use your web browser to create, edit, or delete op modes and then save these changes directly onto the Robot Controller.

#### 3.3.1 Installing a Javascript-enabled Browser

In order to be able to program using the blocks programming mode server, your laptop will need a Javascript-enabled browser. FIRST recommends the use of Google Chrome as the Javascript-enabled browser for the blocks programming mode. The blocks programming mode functions have been thoroughly tested using Google Chrome.

Google Chrome is available for free from the Google website. Visit the following URL and follow the onscreen instructions to download and install Chrome onto your computer:

<https://www.google.com/chrome/browser/desktop/index.html>

Note that your computer might prompt you with a security warning during the installation process. If you are prompted with this warning, click on the “Run” button to continue with the installation.

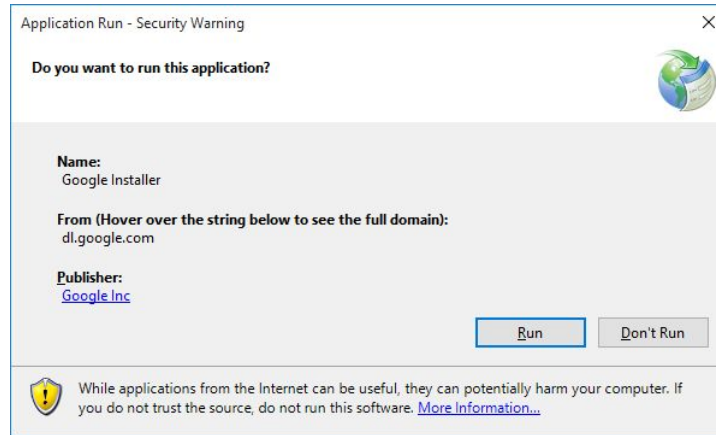


Figure 36 – Your computer might prompt you with one or more security warnings during the installation process.

### 3.3.2 Putting the Robot Controller into Blocks Programming Mode

On the Robot Controller or the Driver Station, touch the three dots in the upper right hand corner of the screen and select “Programming Mode” from the pop up menu.

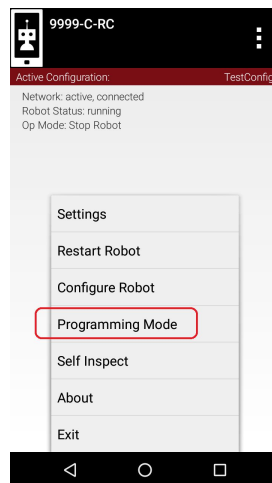
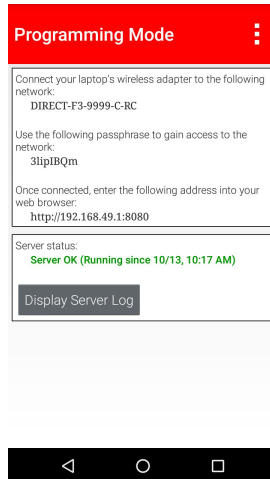


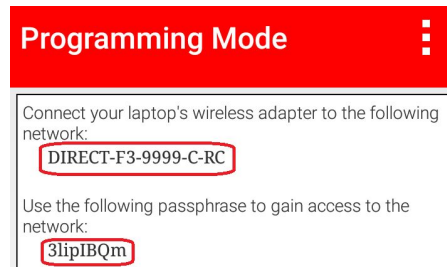
Figure 37 - Select Programming Mode from the pop up menu.

The Robot Controller should now be in programming mode. The Programming Mode screen displays important information that you can use to connect your laptop to the blocks programming mode server.



**Figure 38 - The Programming Mode screen contains the info you need to connect to the server.**

Towards the top of the screen, the name of the blocks programming mode wireless network is displayed (see Figure 39 below). In this example, the name of the Wi-Fi network is “DIRECT-F3-9999-C-RC” and the secure passphrase is “3lipIBQm”.



**Figure 39 - Network name and passphrase for Programming Mode wireless connection.**

You want to configure your laptop computer to connect to the blocks programming mode network. For Windows 10 users, in the lower right hand corner if you click on the Wi-Fi symbol (see Figure 40 below) you should see a list of available Wi-Fi Networks in your vicinity.

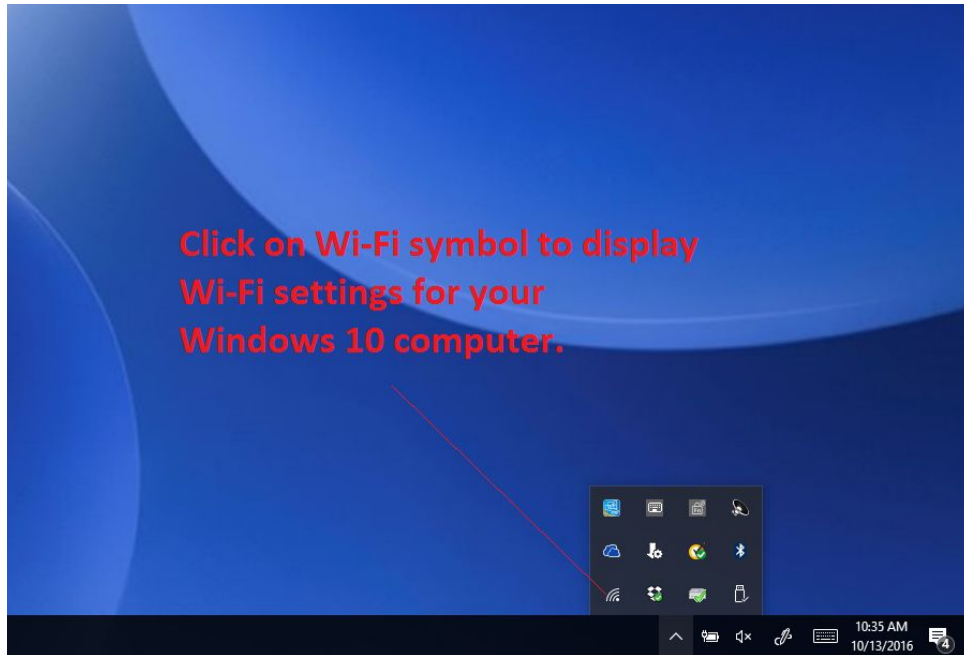


Figure 40 - Windows 10 users can click on the Wi-Fi symbol to display a list of available Wi-Fi networks.

Look for the wireless network that matches the name displayed on the Robot Controller. In this example, the name of the wireless network is “DIRECT-F3-9999-C-RC” and the network is visible in the list displayed on the Windows 10 computer.



Figure 41 - Find and click on the blocks programming mode network in the list of available networks.

Once you have found the target network in the list, click on it to connect to the network.

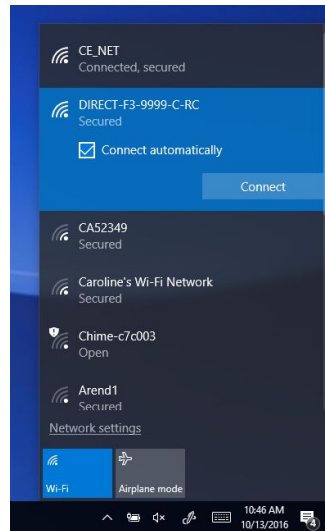


Figure 42 - Connect to the Programming Mode network.

When prompted, provide the network passphrase (in this example “3lipIBQm”) and press “Next” to continue. Note that the passphrase is *case sensitive*. Make sure that your spelling and capitalization matches the original spelling and capitalization shown on the Programming Mode screen.

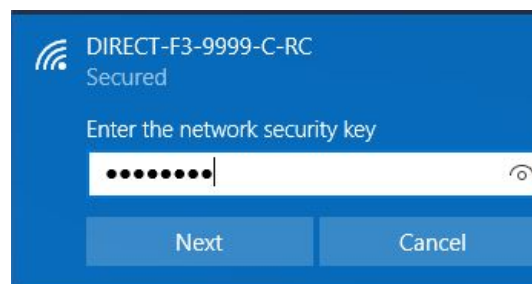


Figure 43 - Provide network passphrase (use the passphrase shown on the Robot Controller screen).

Once you have successfully connected your Windows 10 laptop to the Programming Mode wireless network, the status should be displayed in the wireless settings for your machine. Note that when you are connected to the blocks programming mode server, *your laptop will not have access to the Internet*. It only has direct access to the programming mode server.

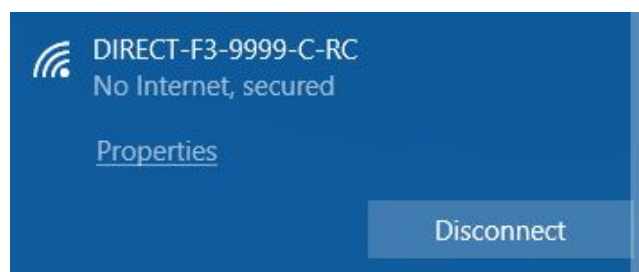


Figure 44 – If you connect successfully, the status info will be displayed in the network listing.

### 3.3.3 Manually Connecting to the Blocks Programming Mode Wi-Fi Network

Section 3.3.2 describes how to search for the blocks programming mode Wi-Fi network from a list of available networks and then connect to it with a Windows laptop. For some Windows devices, the laptop might not display your blocks programming mode Wi-Fi network in its list of available networks. We believe that this problem occurs with some Windows 10 machines (and also possibly with some Windows 8 machines).

If you are having problems seeing your Hosted Programming Wi-Fi network in your list of available networks, make sure that your Driver Station is paired and connected to your Robot Controller (see section 3.1 of this document). Also make sure that your Robot Controller is in Programming Mode (see 3.3.2 of this document). Also, make sure that your Windows 10 device has its most current updates installed from Microsoft.

If you have verified that the Driver Station is paired and connected to the Robot Controller and that the Robot Controller is in Programming Mode, and if you have verified that your Windows 10 updates are current, then you might have to manually connect your Windows 10 computer to the blocks programming mode Wi-Fi network.

You can manually connect to this network as if the network were a *hidden* network (i.e., a network that does not broadcast its presence to other Wi-Fi devices). In the lower right hand corner of the Windows 10 desktop, click on the network icon in the system tray (see Figure 40 on page ) to display a list of available Wi-Fi networks. If you still do not see your blocks programming mode network listed, then scroll to the bottom of the list and look for the item “Hidden Network”.



Figure 45 – Scroll towards the bottom of the list and look for the item labeled “Hidden Network”.

Click on the “Hidden Network” listing to start the connection process. The listing should display a “Connect” button. Make sure the option “Connect automatically” is checked and then click on the “Connect” button to continue with the process.

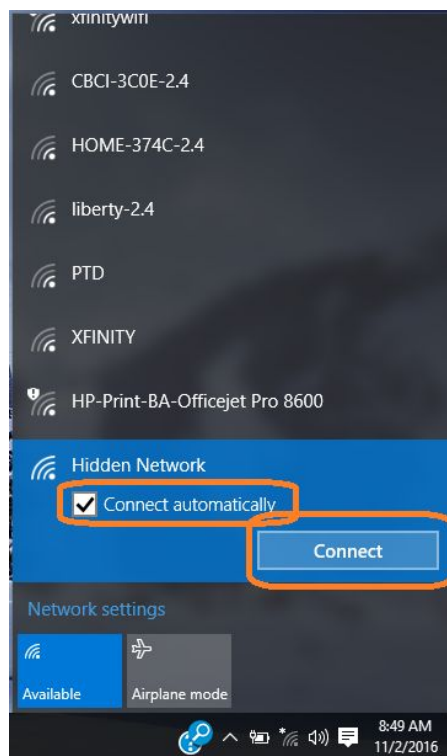


Figure 46 – Check “Connect automatically” and click “Connect”.

The computer should prompt you for the name or *SSID* of your blocks programming mode Wi-Fi network. You should type in the network name that is displayed in the Programming Mode window of the Android device (see Figure 39 on page ). Note, the SSID or network name is *case sensitive*. Make sure the capitalization of the name that you enter matches the capitalization of the name displayed in the Programming Mode Window.

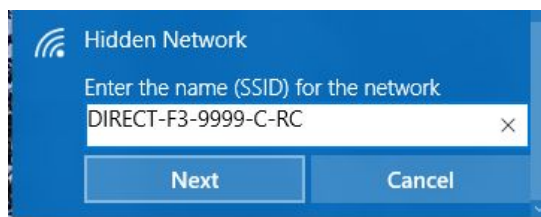


Figure 47 - Enter in the name of the blocks programming mode Wi-Fi network.

The computer should then prompt you for the passphrase to access this Wi-Fi network. You should type in the network passphrase that is displayed in the Programming Mode window of the Android device (see Figure 39 on page ). Note that the passphrase is *case sensitive*. Make sure that your spelling and capitalization matches the original spelling and capitalization shown on the Programming Mode screen.

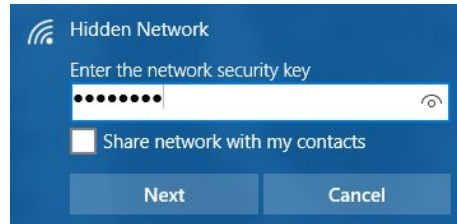


Figure 48 – Enter the passphrase for the blocks programming mode network.

Your computer will prompt you on whether or not you want to make your PC discoverable by other devices on this network. Click “Yes” to continue.

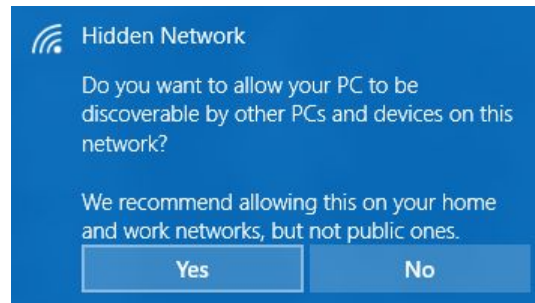


Figure 49 - Click on "Yes" to allow other devices to be able to discover your computer on this network.

The computer will attempt to connect to your network. Note that it could take several minutes before it connects.

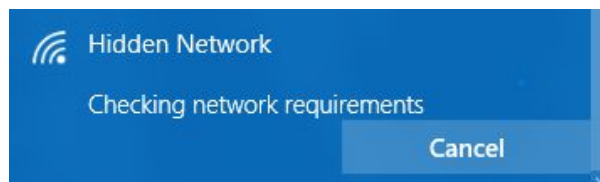


Figure 50 – The computer will check the network requirements and attempt to connect to the network.

If you were able to successfully connect to the network, it will eventually appear in the list of networks on your computer. Note that when your computer is connected to the blocks programming mode Wi-Fi network, *it will not have access to the Internet.*



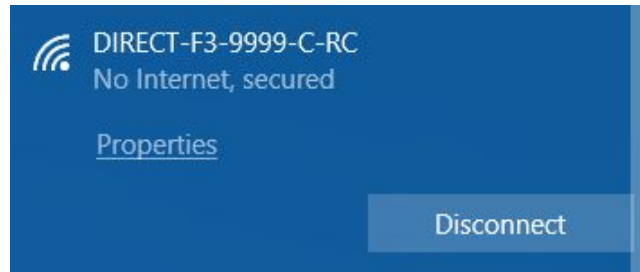


Figure 51 – If you connect successfully, the blocks programming mode Wi-Fi network will be displayed on your computer.

### 3.3.4 Access the Blocks Programming Mode Server with your Browser

If you successfully established a wireless connection to the blocks programming mode wireless network, you can use a Javascript enabled browser to connect to the programming mode server. We recommend using Google Chrome as your browser. Refer to the Google Chrome webpage (see <https://www.google.com/chrome/>) for details on installing this software.

Launch your Javascript enabled browser. Use the web address that is displayed on the Programming Mode screen. In our example, the web address is “192.168.49.1:8080” (see Figure 52).

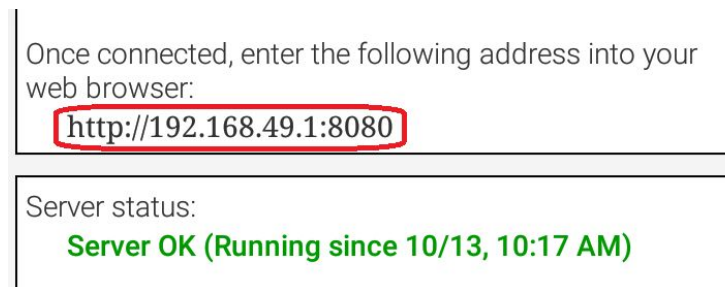


Figure 52 - The web address of the Programming Mode server is displayed on the Programming Mode screen.

If you are able to connect to the programming mode server successfully, the main project window should be displayed.

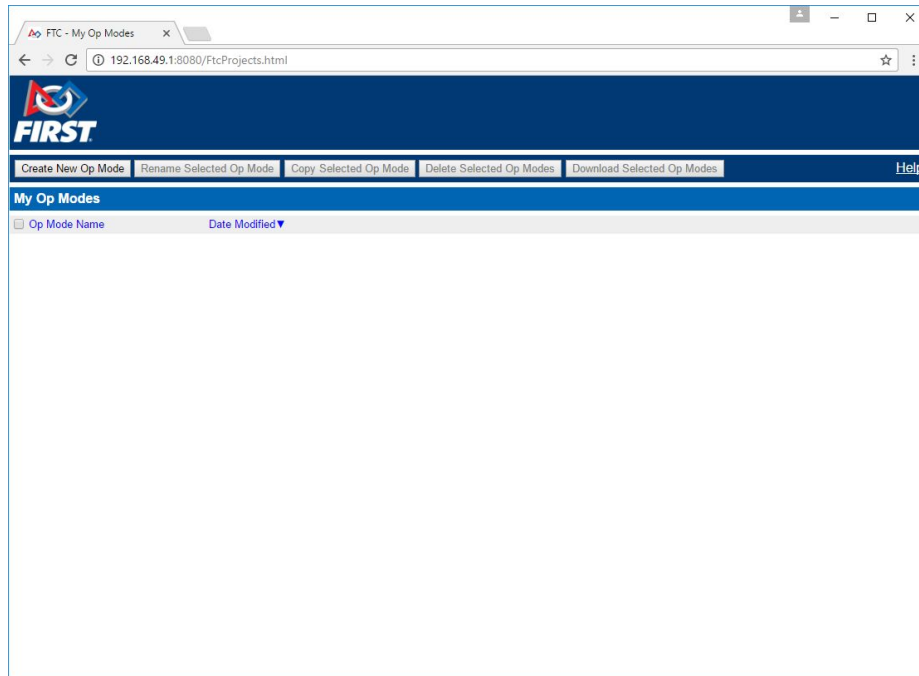


Figure 53 - The main project window.

### 3.3.5 Creating a New Op Mode

To create our first op mode, press the “Create New Op Mode” button which should be visible towards the upper left hand corner of the browser window. When prompted, specify a name for the op mode and hit “OK” to continue. Note that the name that you type in can only contain alpha-numeric characters.

**Create New Op Mode**

Op Mode name:

Figure 54 - Specify a name for the op mode.

The browser window should switch to the design mode for your newly created op mode. In design mode, the left hand side displays a categorized list of programming blocks that you can use in your op mode. The right hand side of the design mode window is the virtual “canvas” that you can drag your programming blocks onto to create your op mode logic.

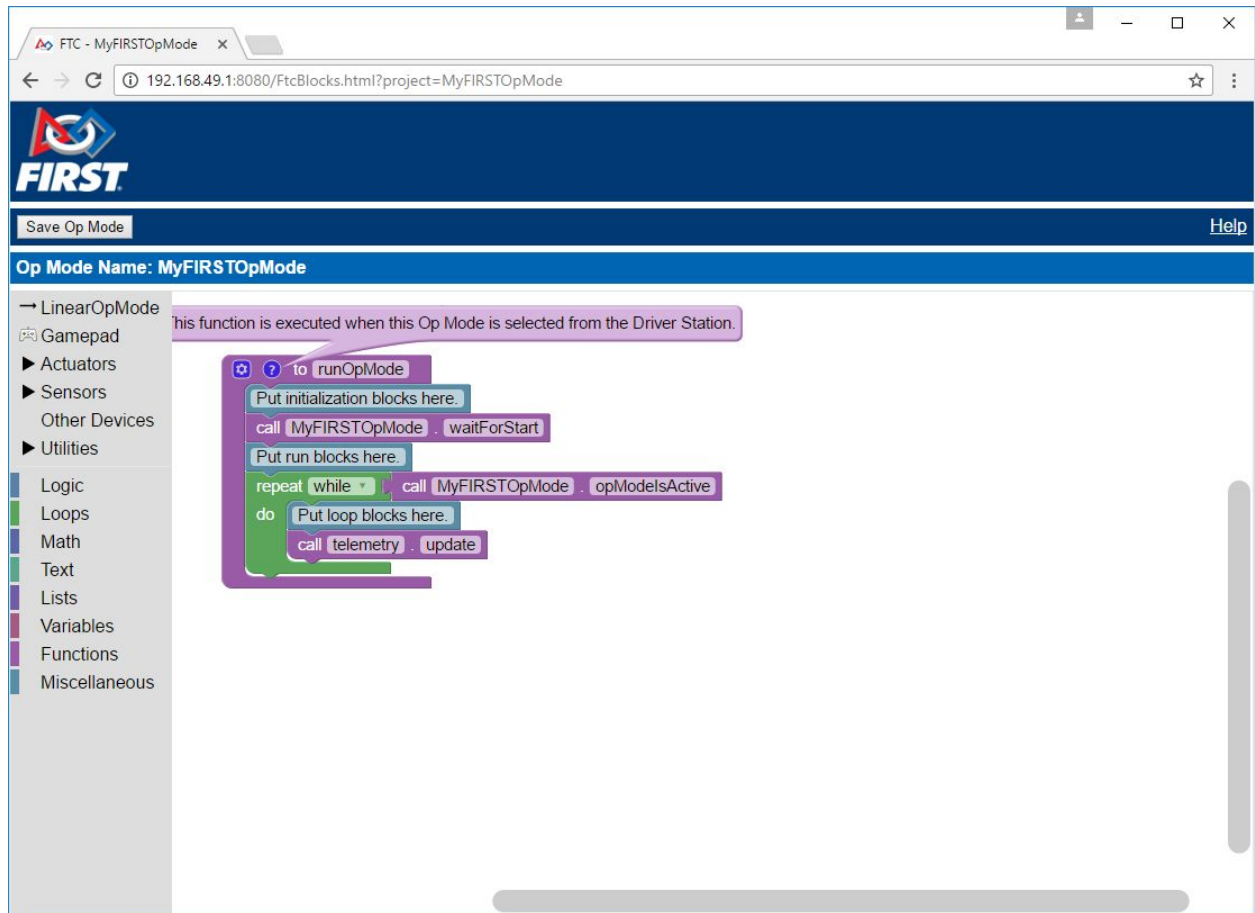


Figure 55 – The left side of the browser window has a categorized list of available programming blocks.

When you create a new op mode, there should already be a set of programming blocks that are placed on the design canvas for your op mode. These blocks are automatically included with each new op mode that you create. They create the basic structure for your op mode.

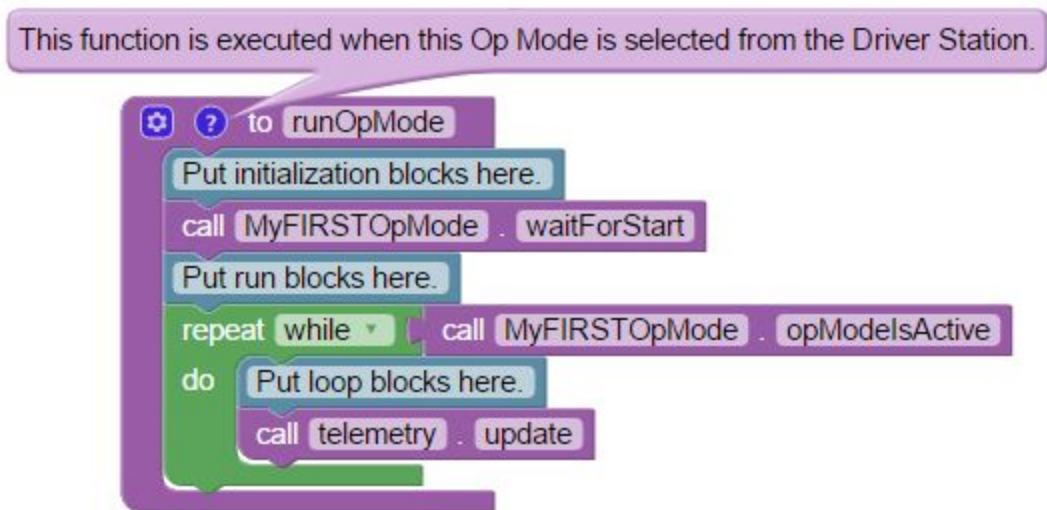


Figure 56 – Each new op mode includes a set of programming blocks that form the basic structure of the op mode.

In Figure 56 the main body of the op mode is defined by the outer purple bracket that has the words “to runOpMode” at the top. As the help tip indicates, this function is executed when this op mode (“MyFIRSTOpMode” in this example) is selected from the Driver Station.

It can be helpful to think of an op mode or a computer program as a list of tasks for the Robot Controller to perform. The Robot Controller will process this list of tasks sequentially. Users can also use control loops (such as a *while* loop) to have the Robot Controller repeat (or iterate) certain tasks within an op mode.



Figure 57 - It can be helpful to think of an op mode as a task list for the Robot Controller.<sup>2</sup>

<sup>2</sup> Clip art image was downloaded from ClipArtBest.com on 10/13/16.

If you think about an op mode as a list of instructions for the robot, this particular set of instructions will be executed by the robot whenever a team member selects the op mode called “MyFIRSTOpMode” from the list of available op modes for this Robot Controller.

Let’s hide the help text by clicking on the blue button with the question mark (“?”) on it. Let’s take a look at the flow of this basic op mode. The blue colored block with the words “Put initialization blocks here” is a comment. Comments are placed in an op mode for the benefit of the human user. The robot will ignore any comments in an op mode.

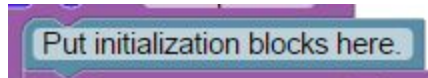


Figure 58 – Comments are blue-colored and are ignored by the robot.

Any programming blocks that are placed after the “Put initialization blocks here” comment (and before the “call MyFIRSTOpMode.waitForStart” block) will be executed when the op mode is first selected by a user at the Driver Station.

When the Robot Controller reaches the block labeled “call MyFIRSTOpMode.waitForStart” it will stop and wait until it receives a Start command from the Driver Station. A Start command will not be sent until the user pushes the Start button on the Driver Station. Any code after the “call MyFIRSTOpMode.waitForStart” block will get executed after the Start button has been pressed.

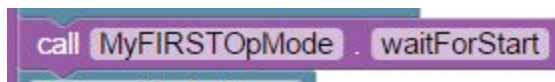


Figure 59 – The Robot Controller will stop and wait for a Start command when it reaches this programming block.

Any blocks that are placed after the “Put run blocks here” comment and before the green block labeled “repeat while call MyFirstOpMode.opModelsActive” will be executed sequentially by the Robot Controller after the Start button has been pressed.

The green block labeled “repeat while call MyFirstOpMode.opModelsActive” is an iterative or looping control structure.



Figure 60 - The green programming block is a while loop control block.

This green control block will perform the steps listed under the “do” portion of the block as long as the condition “call MyFIRSTOpMode.opModelsActive” is true. What this means is that the statements included in the “do” portion of the block will repeatedly be executed as long as the op mode

“MyFIRSTOpMode” is running. Once the user presses the Stop button, the “call MyFIRSTOpMode.opModelsActive” clause is no longer true and the repeat while loop will stop repeating itself.

Let’s modify our op mode to add some blocks that will allow us to control a DC motor with a gamepad. On the left hand side of the screen click on the category called “Variables” to display the list of block commands that are used to create and modify variables within your op mode. Since you have not yet created any variables, the only action that appears is the “Create variable...” command. Click on this item to create a new variable that will represent the target motor power for our op mode.

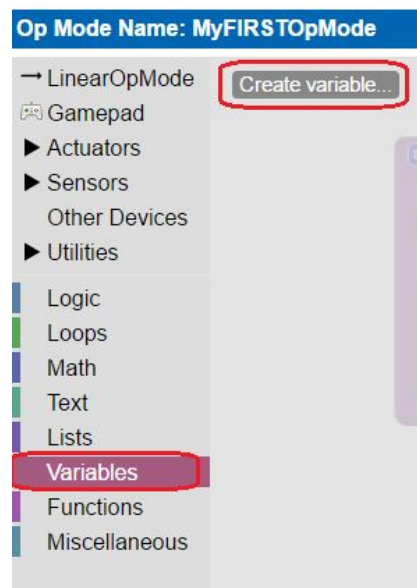


Figure 61 - Click on Variables->Create variable... to create a new variable.

When prompted type in a name (“tgtPower”) for your new variable.

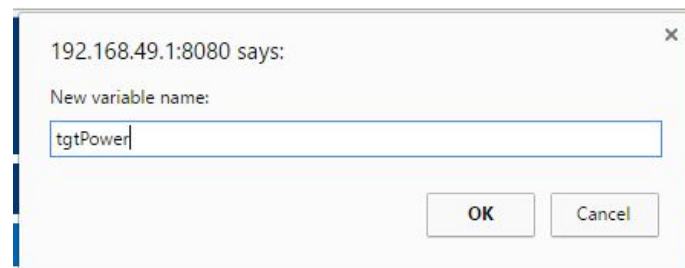


Figure 62 – Specify a name for the new variable

Once you have created your new variable, some additional programming blocks should appear under the “Variables” block category.

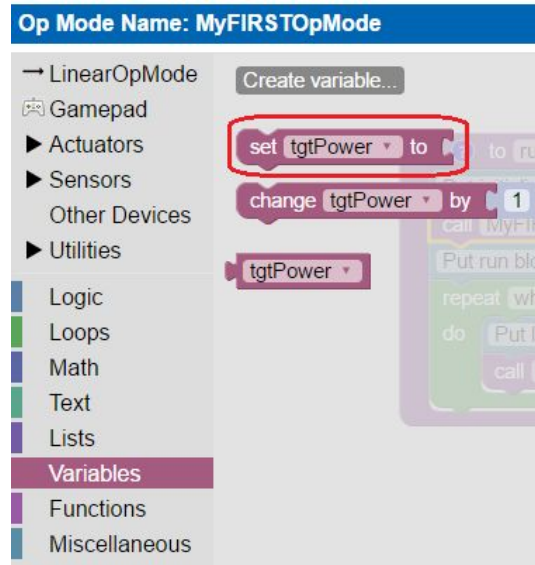


Figure 63 - Click on the "set tgtPower to" block.

Click on the “set tgtPower to” programming block and then use the mouse to drag the block to the spot just after the “Put loop blocks here” comment block. The “set tgtPower to” block should snap right into position.

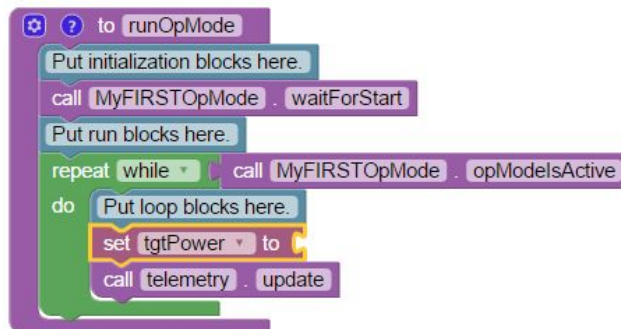


Figure 64 – Drag the “set tgtPower to” block to the spot just after the “Put loops block here” comment.

For this example op mode, we want to take the input from the Logitech gamepad (which will be connected to the driver station) and use the position of the left joystick to control the speed of the motor. Click on the “Gamepad” category of the programming blocks and select the “gamepad1.LeftStickY” block from the list of available blocks.

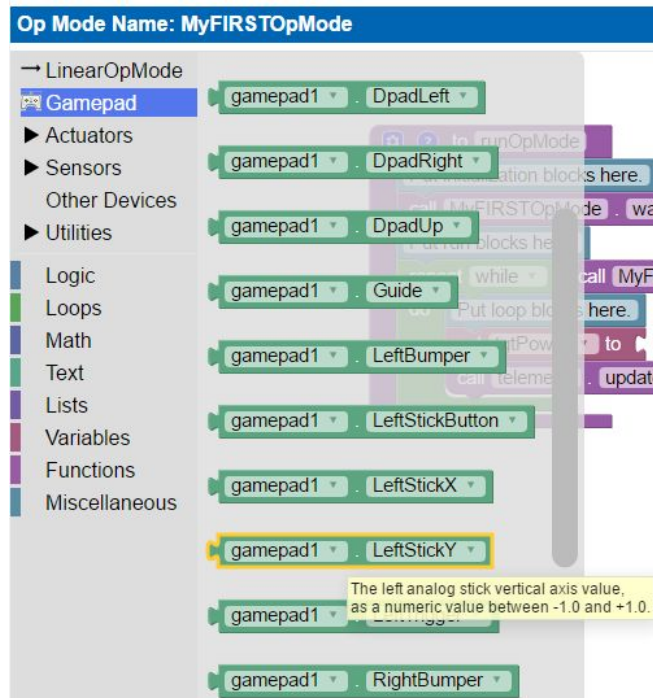


Figure 65 - Select Gamepad->gamepad1.LeftStickY.

Drag the “gamepad1.LeftStickY” block so it snaps in place onto the right side of the “set tgtPower to” block.

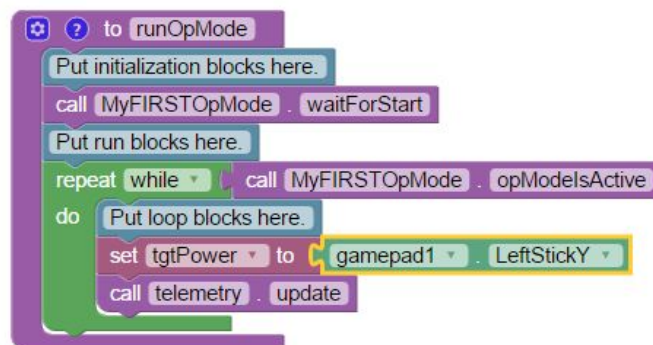


Figure 66 - Drag the "gamepad1.LeftStickY" block to the right side of the “set tgtPower to” block.

This set of blocks will continually loop and read the value of the gamepad #1’s left joystick (the y position) and set the variable tgtPower to the Y value of the left joystick. Note that for the F310 gamepads, the Y value of the joysticks range from -1 when the joystick is in its topmost position to +1 when the joystick is in its bottommost position (see Figure 67 below). This means that for the blocks shown in Figure 66 (see above) if the left joystick is pushed to the top, the variable tgtPower will have a value of -1. Since we want tgtPower to represent the motor power, let’s modify our op mode so that tgtPower will be +1 whenever the left joystick is pushed to its topmost position.





Figure 67 - The Y value of the joysticks range from -1 in the topmost position to +1 in the bottommost position.

Click on the “Math” category for the programming blocks and select the negative symbol (“-”).

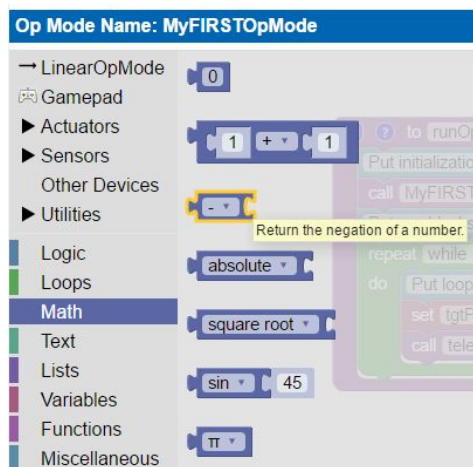


Figure 68 – Select “Math” and click on the negative symbol (“-”).

Drag the negative symbol (also known as a “negation operator”) to the left of the “gamepad1.LeftStickY” block. It should click in place after the “set tgtPower to” block and before the “gamepad1.LeftStickY” block.

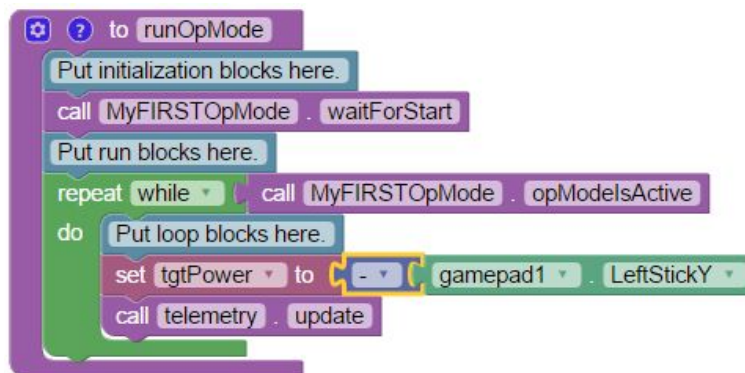


Figure 69 - Negate the left joystick position.

In Figure 69 (see above) the variable `tgtPower` will be set to +1 if the left joystick is in its topmost position and will be set to -1 if the left joystick is in its bottommost position.

Click on the “Actuators” category of blocks. Since our Robot Controller’s configuration file only has a single DC motor controller configured, the only actuator subcategory that is listed is the “DcMotor” subcategory.



Figure 70 - Only the DcMotor subcategory is available for our Op Mode since we only have a motor controller configured.

Select the “set motor1.Power to 1” programming block.

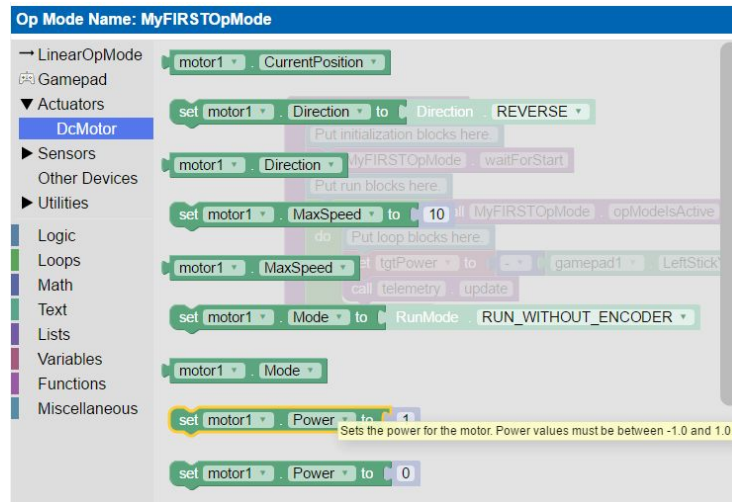


Figure 71 - Select the "set motor1.Power to 1" programming block.

Drag and place the “set motor1.Power to 1” block so that it snaps in place right below the “set tgtPower to” block.

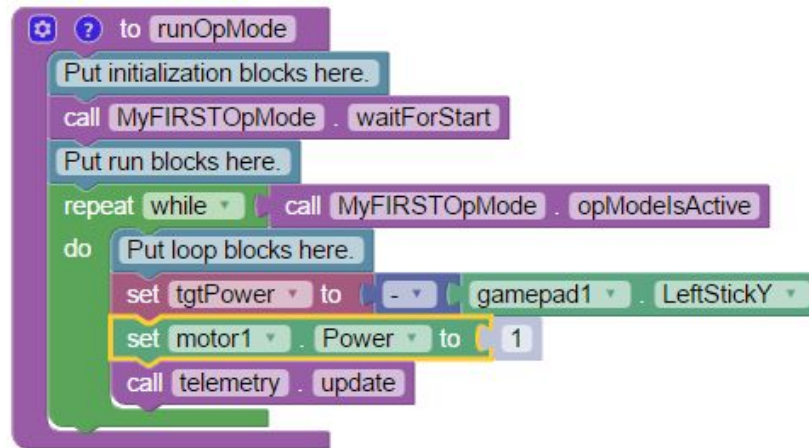


Figure 72 – Drag the “set motor1.power to” block so it snaps in place below the “set tgtPower to” block.

In the blocks show in Figure 72 the “set motor1.Power to” block has a default value of 1. This means that this block will set the motor’s power to the value of 1 (motor power ranges from -1 to +1) unless we tell the block what non-default value to use.

Click on the “Variables” block category and select the “tgtPower” block.

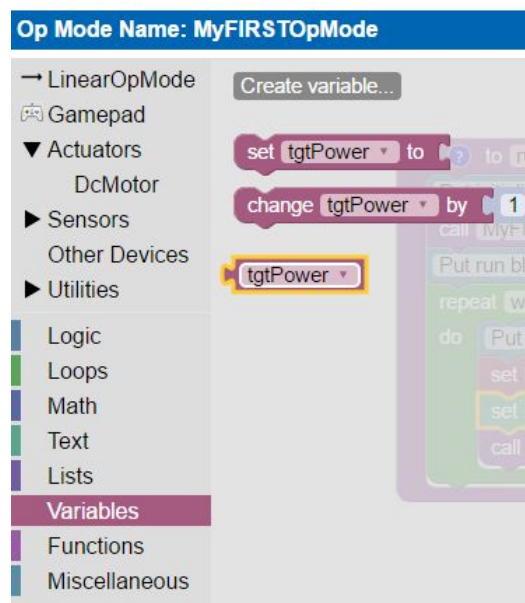


Figure 73 – Click on the “Variables” category and then select the “tgtPower” block.

Drag the “tgtPower” block so it snaps in place just to the right of the “set motor1.Power to” block. The “tgtPower” block should automatically replace the default value of “1” block. Your op mode should look like the one shown in Figure 74 (see below).

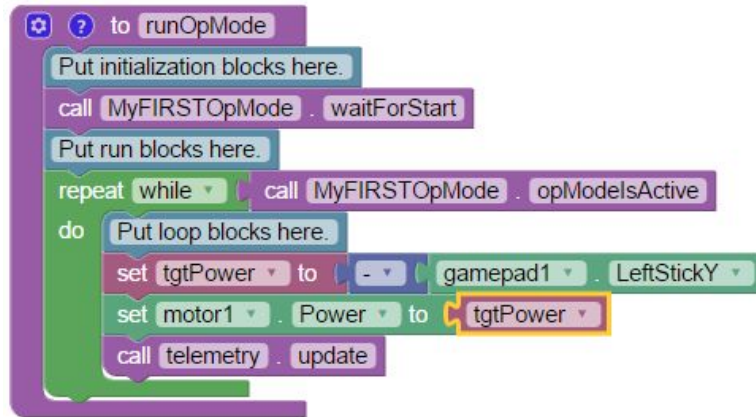


Figure 74 - Drag "tgtPower" so it clicks in place to the right of the "set tgtPower to" block.

### 3.3.6 Inserting Telemetry Statements

Your op mode is just about ready to run. However, before we continue, let's add a couple of *telemetry statements* that will send information from the Robot Controller to the Driver Station for display on the Driver Station user interface. This *telemetry* mechanism is a useful way to display status information from the robot on the Driver Station. You can use this mechanism to display sensor data, motor status, gamepad state, etc. from the Robot Controller to the Driver Station.

Click on the "Utilities" category on the left hand side of the browser window. Select the "Telemetry" subcategory and select the "call telemetry.addData(key, number)" block.

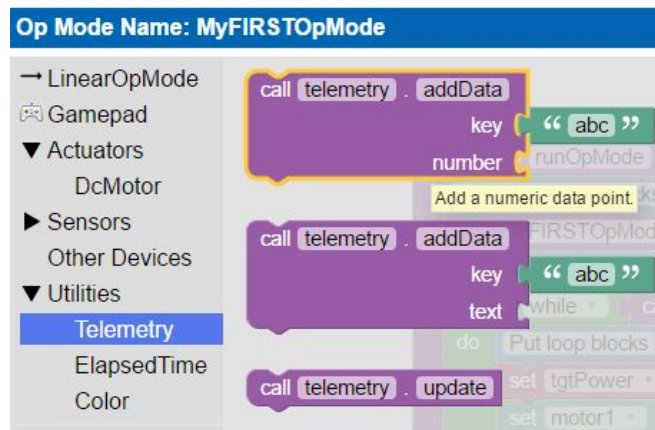


Figure 75 - Select the "call telemetry.addData(key, number)" programming block.

Drag the "call telemetry.addData(key, number)" block and place it below the "set motor1.Power to" block. Click on the green text block "abc" and highlight the text and change it to read "Target Power". Your op mode should look like the blocks shown below.

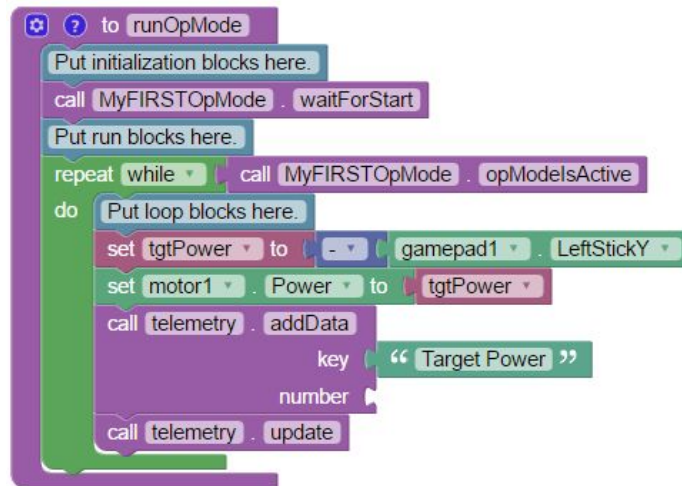


Figure 76 - Set the key parameter to "Target Power".

Click on the “Variables” block category and select the “tgtPower” block. Drag the block so it clicks into place next to the “number” parameter on the telemetry programming block.

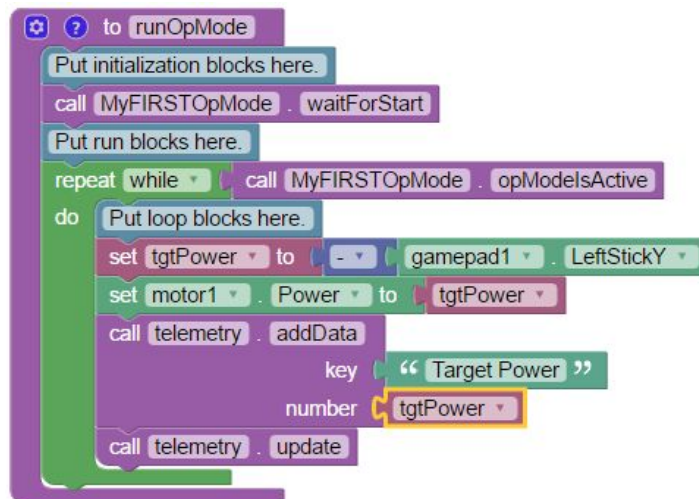


Figure 77 – Drag a “tgtPower” variable block and drop it next to the “number” parameter.

For the telemetry block shown in Figure 77, the Robot Controller will send the value of the variable tgtPower to the Driver Station with a key or label of “Target Power”. The key will be displayed to the left of the value on the Driver Station.

Repeat this process and name the new key “Motor Power”. Your blocks should look like the following image.



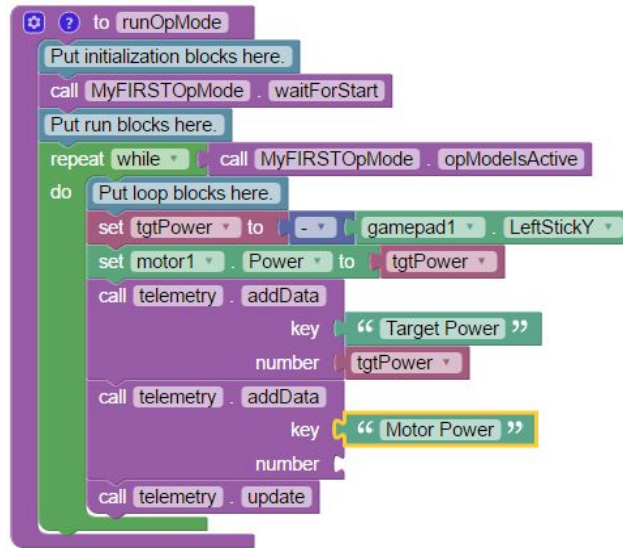


Figure 78 – Create another telemetry block (for a number) and set the key to “Motor Power”.

For this second telemetry block, we want to send over the numerical value of the motor power. Click on the “Actuators”->“DcMotor” category and look for the green programming block labeled “motor1.Power”.

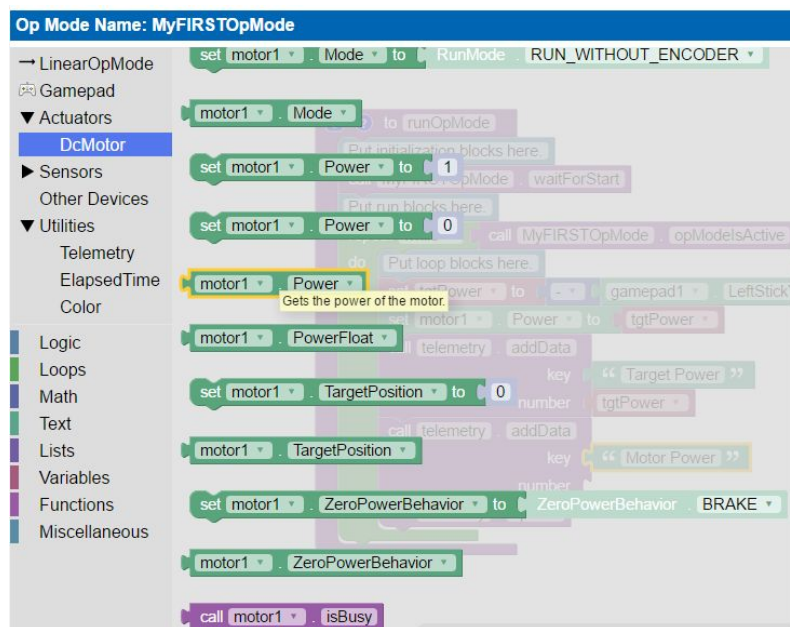


Figure 79 – Find and select “Actuators”->“DcMotor”->“motor1.Power”.

Drag the “motor1.Power” block to the “number” parameter of the second telemetry block. Your op mode should look like the following image.

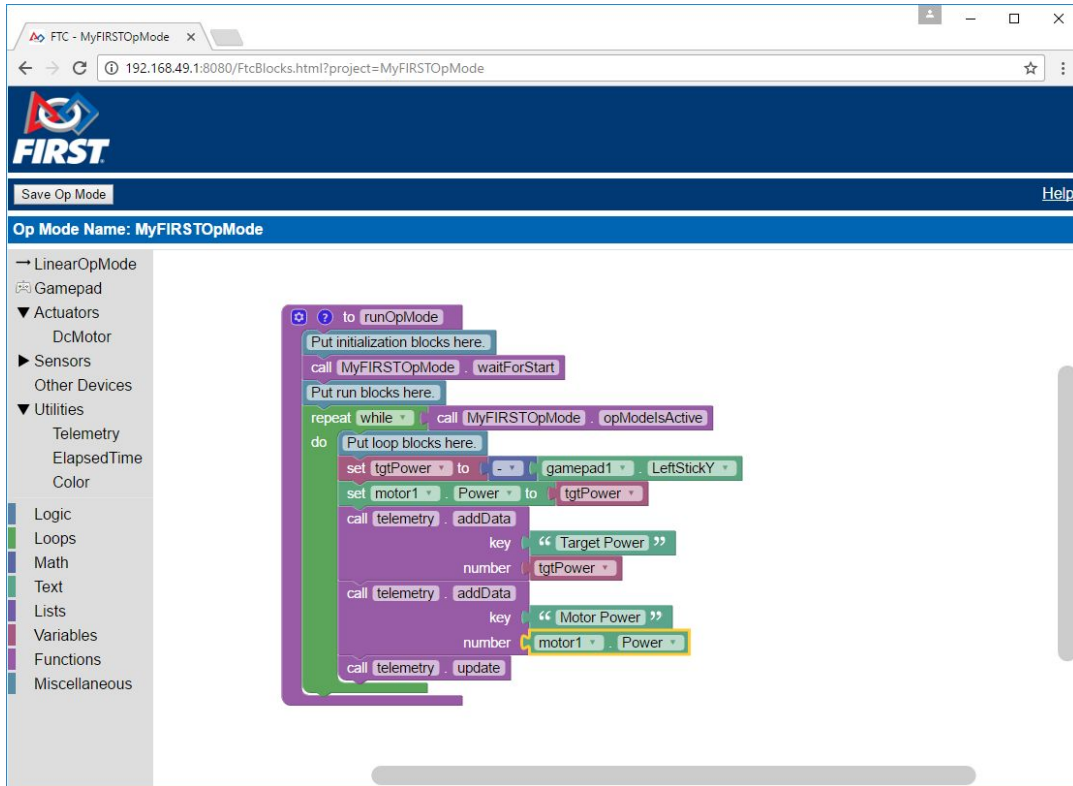


Figure 80 – Drag the “motor1.Power” block to the “number” parameter of the second telemetry block.

### 3.3.7 Saving Your Op Mode

Once you have written your op mode, press the “Save Op Mode” button to save the op mode to the Robot Controller’s programming mode server. If your save was successful, you should see “Save completed successfully” in green letters next to the button.

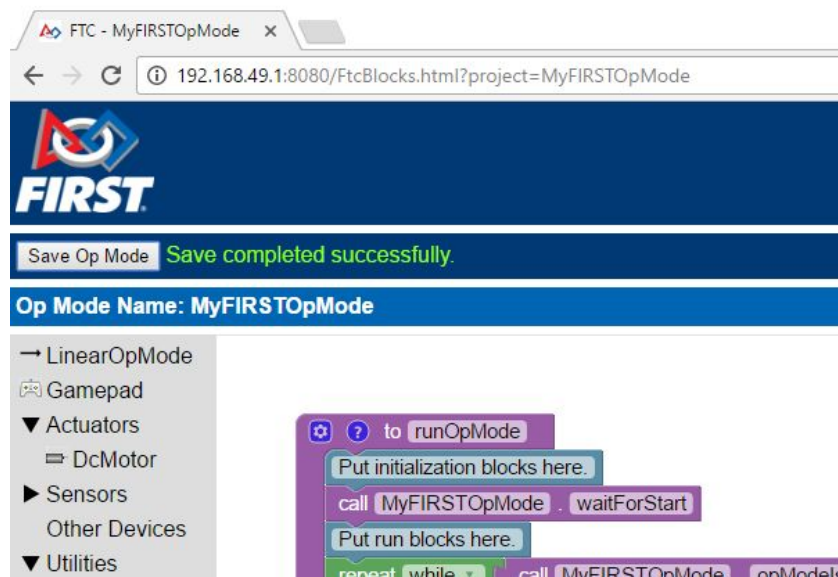


Figure 81 – Press the “Save Op Mode” button in the upper left hand corner to save your work.

If you are having problems saving your op mode, make sure your Robot Controller is in the Programming Mode and make sure your laptop is connected to the Programming Mode wireless network before you push the save button.

### 3.3.8 Exiting Programming Mode

After you have successfully saved your Op Mode, you can exit Programming Mode and return to normal operation of the Robot Controller. On the device that displays the Programming Mode main screen (with the Network and Passphrase information on the screen, see figure below), press the Android back arrow to exit Programming Mode. You need to exit Programming Mode before you can run your op mode.

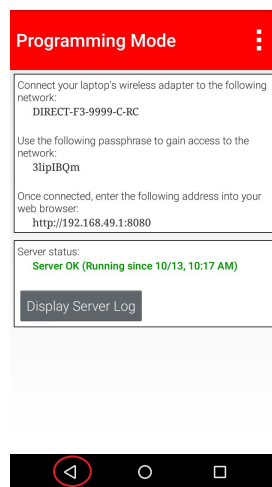


Figure 82 - Press the Android back arrow button to exit Programming Mode.

### 3.3.9 Connecting a Gamepad to Your Driver Station

For this example, your op mode takes input from a gamepad and uses this input to control a DC motor. In order to run your op mode, you will need to connect a Logitech F310 gamepad to the Driver Station. Before you connect your gamepad to the phone, verify that the switch on the bottom of the gamepad is set to the “X” position.





Figure 83 - Make sure the switch is in the "X" position.

The Logitech F310 gamepad needs a Micro USB OTG adapter cable to allow the gamepad to connect to the micro OTG USB port on the Android phone.



Figure 84 - Micro USB OTG adapter cable.

Connect the gamepad to the Driver Station as shown in the figure below.



Figure 85 - Connect gamepad to Driver Station.

The FTC Driver Station app is designed to support up to two gamepads connected to the phone. This means that an FTC team can have up to two people controlling their robot simultaneously. When you connect a gamepad to the phone, you need to tell the FTC Driver Station app which driver (#1 or #2) that this gamepad should represent.

You can select which driver a gamepad will represent by pushing the **Start** key on the gamepad while simultaneously pressing the **A** button if you want to be driver #1, or the **B** button if you want to be driver #2.



**Figure 86 - Press Start + A to be driver 1 or Start + B to be driver 2.**

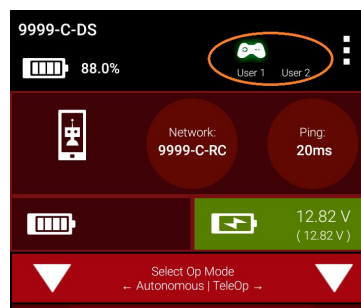
When you first connect your gamepads to the Android device, you must push a button combination to designate which driver your gamepad will represent.

Once you have designate which driver your gamepad will represent the gamepad status information will appear below the appropriate area near the top of the screen.

For example, suppose the gamepad that represents driver #1 is active, then the gamepad icon for gamepad 1, near the upper right hand corner of the screen, will be highlighted green whenever you push a button or move a joystick on that gamepad.

If the gamepad that represents driver #2 is active, then the gamepad icon for gamepad 2, near the upper right hand corner of the screen, will be highlighted green.

If both gamepads are being used simultaneously, then both gamepad icons will be highlighted.



**Figure 87 – Gamepad activity is displayed in the upper right hand corner of the screen.**

For your example op mode, you will want to have the gamepad designated as User 1. Press the Start and A buttons to designate your gamepad as driver #1.

### **3.3.10 Selecting and Running Your Op Mode**

If your gamepad is connected properly and your phones are connected to each other, your driver station phone should look similar to the image below.

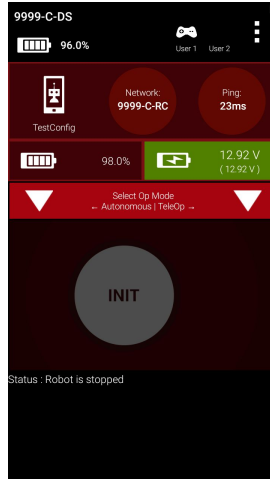


Figure 88 – System is connected and ready to go.

Touch the “TeleOp” selection control on the right hand side of the Driver Station screen to display a list of available TeleOp op modes. The word “TeleOp” is short for “Tele-Operated” and it implies a driver controlled op mode (i.e., an op mode that gets input from a human driver).

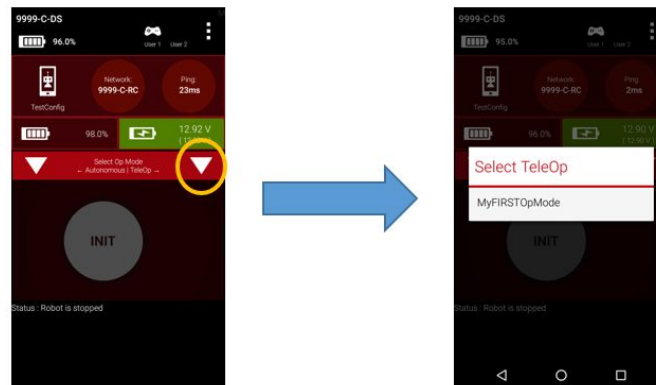
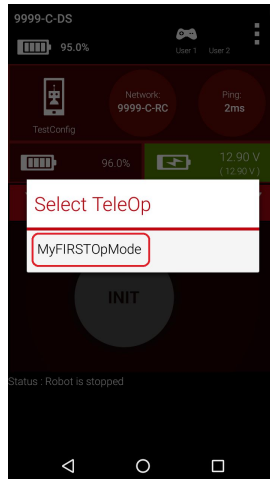


Figure 89 – Touch the selection control to display a list of available op modes.

The Driver Station should list a single op mode “MyFIRSTOpMode” in the “Select TeleOp” list.

Figure 90 – The Driver Station should display “MyFIRSTOpMode” in its list of available op modes.

Select “MyFIRSTOpMode” to load your op mode on the Robot Controller. Remember, even though you are using the Driver Station to select the op mode, the actual op mode instructions will be executed on the Robot Controller phone.



**Figure 91 - Select "MyFIRSTOpMode" to launch your op mode onto the Robot Controller.**

Press the "INIT" button to initialize your op mode.



**Figure 92 - Press the INIT button to initialize the op mode.**

At this point, if you want to abort your op mode run, you could push the smaller Stop button (designated by the square-shaped symbol, see Figure 93). To start the op mode run, push the Start button (designated by the triangular shaped symbol).

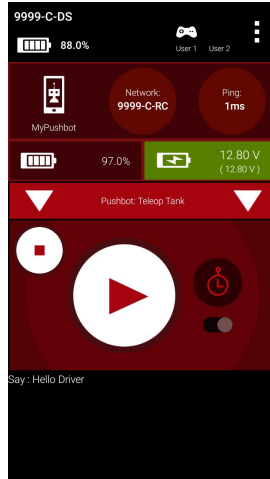


Figure 93 – Press the Start button (designated with the triangular shape) to start the op mode.

If everything is working properly, the left joystick of the gamepad should control the operation of the DC motor. As you manipulate the joystick, the target power and the motor power should be displayed in the lower left hand corner of the screen.

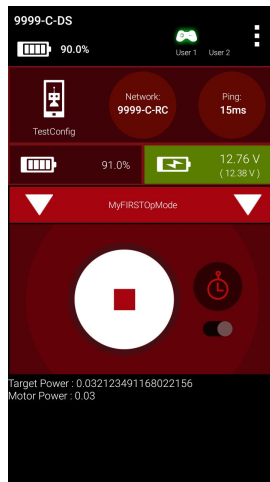


Figure 94 - The left joystick should control the motor. Telemetry data should appear in lower left hand corner of screen.

To stop your op mode, press the Stop button (designated by the square) on the Driver Station.

Congratulations! You’ve just written your first op mode!

## 4 Controlling a Servo Motor with an Op Mode

### 4.1 What is a Servo Motor?

In this section you will learn how to write an op mode to control a *servo* motor. A servo motor is a special type of motor. A servo motor is designed for precise motion. A typical servo motor has a limited range of motion. In Figure 95 a “standard scale” 180-degree servo is shown. This type of servo is

popular with hobbyists and with *FIRST* Tech Challenge teams. This servo motor can rotate its shaft through a range of 180 degrees. Using an electronic module known as a *servo controller* you can write an op mode that will move a servo motor to a specific position. Once the motor reaches this target position, it will hold the position, even if external forces are applied to the shaft of the servo.



Figure 95 – An example of a servo motor that is commonly used on *FIRST* Tech Challenge robots.<sup>3</sup>

Servo motors are useful when you want to do precise movements (for example, sweep an area with a sensor to look for a target) or if you do not need a high torque motor to move something (for example, to move the control surfaces on a remote controlled airplane).

## 4.2 Adding a Servo Controller and Servo to the Test System

For this example, you will need to connect a Modern Robotics servo controller to your test hardware. Add a servo controller and a servo to your test setup so that it looks similar to the setup depicted in Figure 96 (see below).

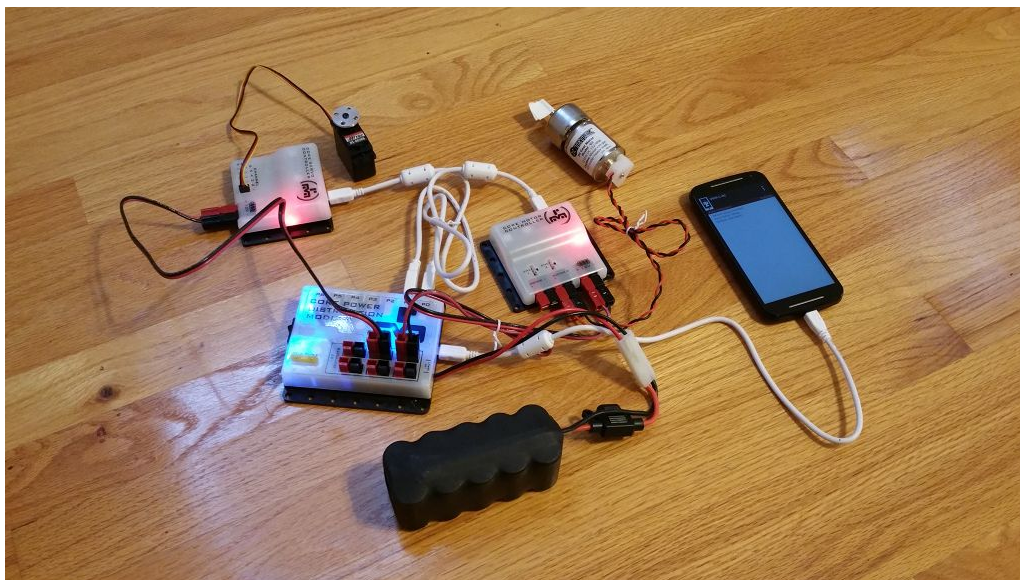


Figure 96 - Add a servo controller to the power distribution module and connect a servo to port #1 of the controller.

---

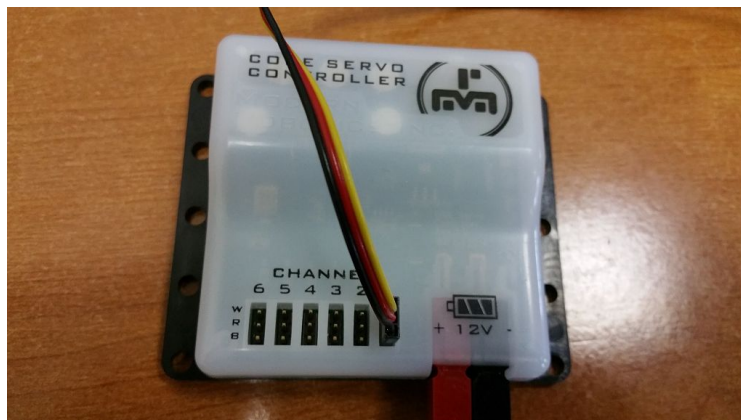
<sup>3</sup> Image taken from the Pitsco website on 10/17/2016 (see [https://c10645061.ssl.cf2.rackcdn.com/product/icongo/icg\\_39197\\_180degserve.jpg](https://c10645061.ssl.cf2.rackcdn.com/product/icongo/icg_39197_180degserve.jpg)).

You will need to provide a 12V DC connection from the PDM to the servo controller (see Figure 96). You will also need to provide a USB connection from the PDM to the servo controller (also see Figure 96).

Note that in Figure 96 the servo is plugged into channel #1 (the rightmost port) of the Modern Robotics servo controller.

Also note that when you connect your servo to the channel on the servo controller, you must make sure that the ground wire is connected to the bottom-most pin of the servo channel port. If you look closely on the servo controller, you will see the letters “W”, “R” and “B” printed on the left side of the case. The manufacturer, Modern Robotics, uses the “B” to denote the ground pin. For most servos, the ground wire is usually colored black or brown.

In Figure 97 (see below), the servo has a cable with a yellow, red and black wire. The black wire on the servo cable is the ground wire for this servo. The connector in Figure 97 is aligned so that the black wire is connected to the “B” pin of the servo channel port.



**Figure 97 - Make sure to line up the yellow, red and black wires of the servo with the matching letter (“**

Also note that when you first connect your Robot Controller to the Power Distribution Module with the servo controller added, the Android operating system might prompt you on whether or not you want to associate the newly detected hardware to the Robot Controller app. If this happens, you should select the “Use by default” option and click on “OK” to associate the new hardware with the app.

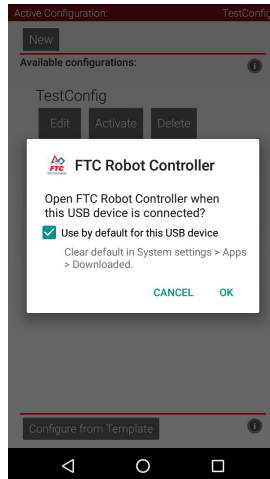


Figure 98 – If prompted to associate the newly added hardware to the app, select “Use by default” and click “OK”.

### 4.3 Configuring the Servo Controller

Once the servo controller has been connected to the Power Distribution Module, you will need to modify your configuration file to include the newly added devices. With your Robot Controller phone connected to the input USB port of the Power Distribution Module and the power switch on the PDM turned on, touch the three vertical dots in the right hand corner of the Robot Controller screen to display the pop-up menu. Select “Configure Robot” from the pop-up menu to display the main Configure Robot screen.

Locate the configuration file from the previous example (“TestConfig”) and push the “Edit” button to edit the existing file.

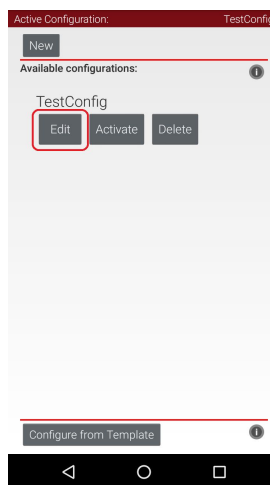
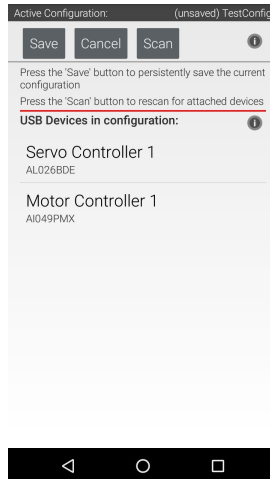


Figure 99 - Press the "Edit" button to edit the existing configuration file.

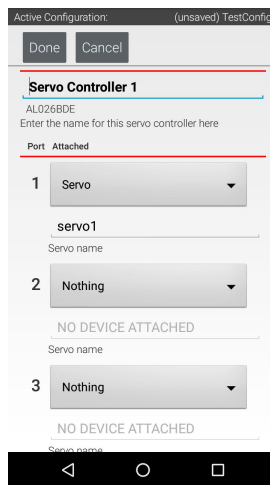
Click on the “Scan” button to force the Robot Controller to scan the USB. It should detect and display the newly added servo controller to the list of devices.





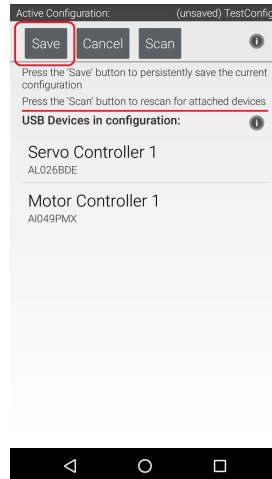
**Figure 100 – The newly added servo controller should be detected and displayed after the USB scan.**

Click on the newly added device (which should have a default name of “Servo Controller 1”) to configure the ports on the device. Modify the configuration information to indicate that a Servo with the name of “servo1” is connected to port #1 of the servo controller. Press “Done” to exit the servo controller configuration screen.



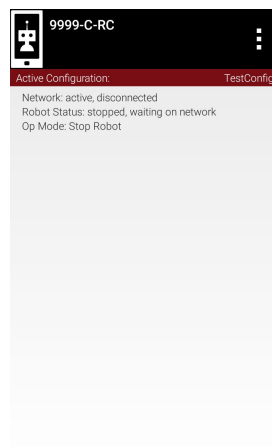
**Figure 101 – Add the servo information to port #1 of the servo controller then hit “Done” to continue.**

Press “Save” and follow the onscreen prompts to save the configuration file with the updated information.



**Figure 102 - Press "Save" and follow onscreen prompts to save the updated info.**

Once you have saved the updated configuration information, use the Android back arrow button to return to the main Robot Controller screen.



**Figure 103 – Use the back arrow key to return to the main screen of the Robot Controller.**

#### 4.4 Switching the Robot Controller into Programming Mode

From the main Robot Controller screen, touch the three vertical dots in the upper right hand corner to display the pop-up menu. Select “Programming Mode” to put the Robot Controller back into Programming Mode.

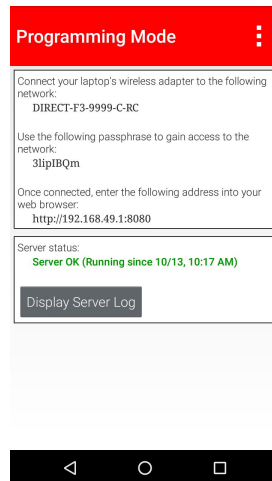


Figure 104 - Put the Robot Controller back into Programming Mode.

On your laptop computer, check to see if it has reconnected the blocks programming mode Wi-Fi network provided by your Robot Controller phone. If it has not, reconnect it (refer to section 3.3.2 on page of this manual). Once the Wi-Fi connection has been re-established, return to your Javascript browser where you were editing your previous op mode. Click on the *FIRST* logo in the upper left hand corner of the screen to return to the main projects menu for the Programming Mode server.

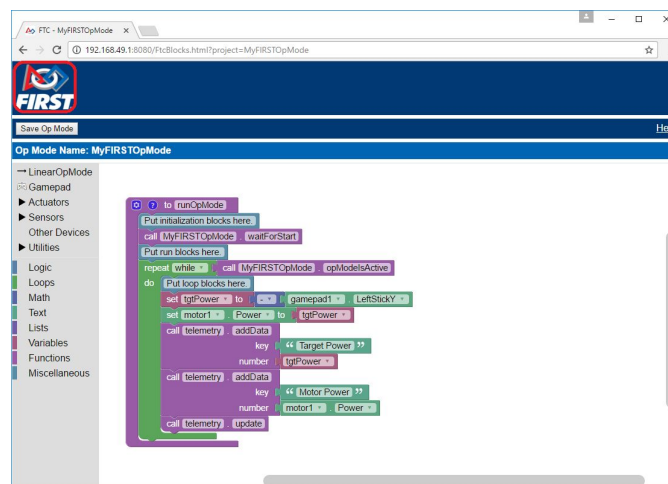


Figure 105 – Click on the *FIRST* logo to return to the main project screen for the Programming Mode server.

On the main project screen, you should see your op mode (named “MyFIRSTOpMode”) listed as an available project to edit. Click on the “MyFIRSTOpMode” listing to re-open the project. When your Javascript enabled browser reopens this project, it will be updated to include the new servo controller information in the design block palette.

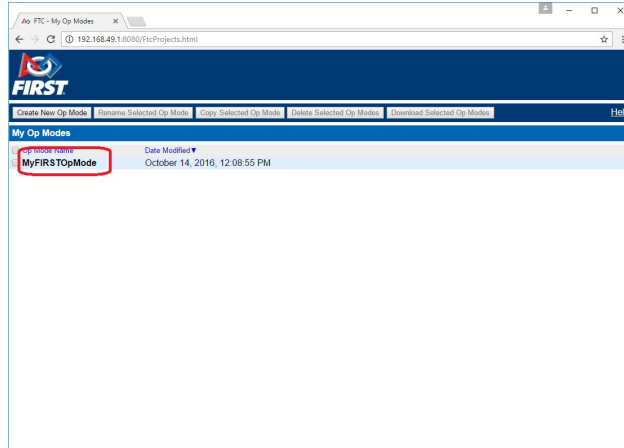


Figure 106 - Click on the "MyFIRSTOpMode" listing to re-open the project and to import the new servo configuration info.

After you have re-opened the project, since the Robot Controller is now configured with the new configuration file that includes the servo controller and servo information, the design block palette on the left hand side should include two additional block subcategories *Servo* and *ServoController*. These two new subcategories should be listed under the *Actuators* category.

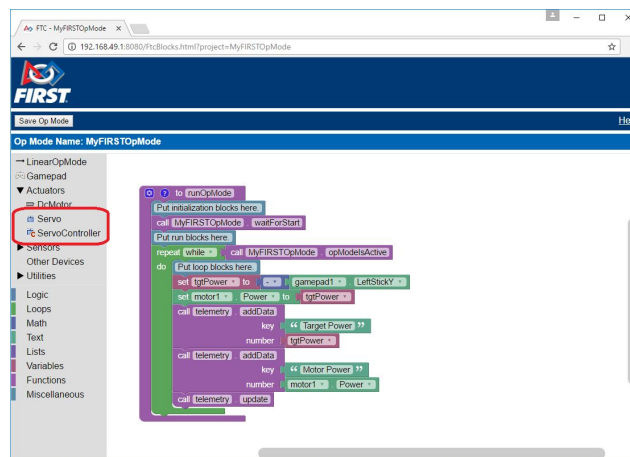


Figure 107 – After closing and then reopening the project, two new block categories should appear, *Servo* and *ServoController*.

## 4.5 Mapping Joystick Position to Target Servo Position

For this example, let's use the right joystick to control the motion of the servo. With a typical servo, you can specify a target position for the servo. The servo will turn its motor shaft to move to the target position, and then maintain that position, even if moderate forces are applied to try and disturb its position.

For the blocks programming mode server, you can specify a target position that ranges from 0 to 1 for a servo. A target position of 0 corresponds to zero degrees of rotation and a target position of 1 corresponds to 180 degrees of rotation for a typical servo motor.



Figure 108 - A typical servo motor can rotate to and hold a position from 0 to 180 degrees.

Similar to the left joystick, the Y values of the right joystick ranges from -1 when the joystick is in its topmost position to +1 when the joystick is in its lowest position. For this example, you want the bottommost position to represent 0 degrees (or a target position of 0) and the topmost position to represent 180 degrees (or a target position of 1).

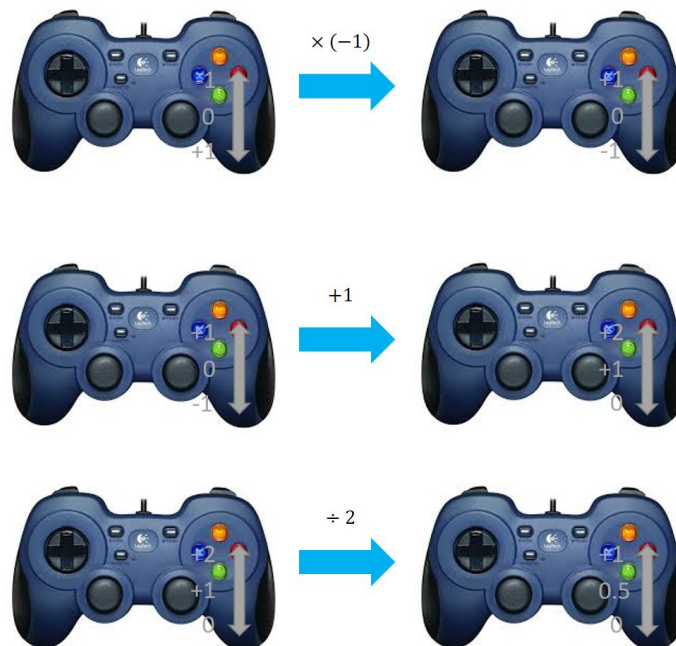


Figure 109 – The Y position of the joystick should be negated, offset by 1, then scaled by 2.

To convert the Y position of the right joystick, you will need to multiply the Y position by negative one. Then you will need to offset the value by plus one and then scale the output down by dividing by two. If you do these operations, then the bottommost position of the right joystick will correspond to a target position of 0 (or 0 degrees) on the servo and the topmost position will correspond to a target position of 1 (or 180 degrees) on the servo.

## 4.6 Modifying Your Op Mode

For this example, let's modify your existing op mode to enable control of the servo. Start by creating a new variable to keep track of the target servo position. From the blocks design palette on the left hand side click on the "Variables" category and select "Create variable..." to create a new variable called "tgtPosition".

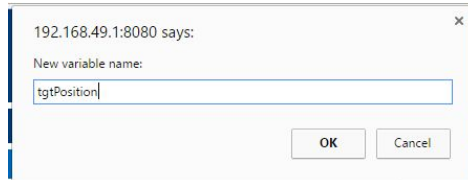


Figure 110 - Create a new variable called "tgtPosition".

Click on the "Variables"->"set tgtPosition to" block and drag it to the spot immediately below the "set motor1.Power to" block in the while loop of your op mode.

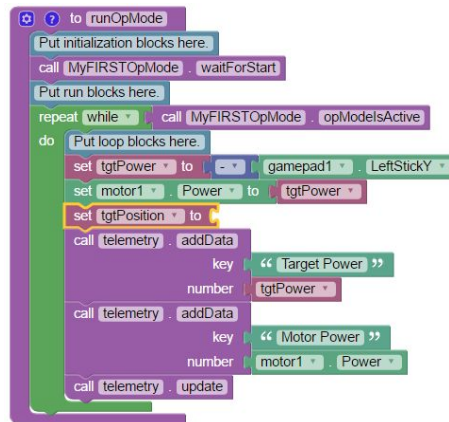


Figure 111 - Insert "set tgtPosition to" block immediately after the "set motor1.Power to" block.

Use the design blocks that are contained in the "Math" and "Gamepad" categories so that the variable "tgtPosition" is set to the negative value of the right joystick.

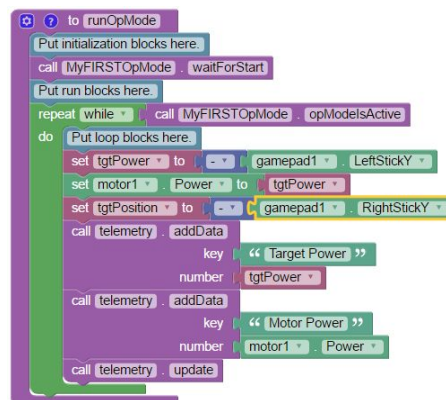


Figure 112 - Set tgtPosition to the negative value of the gamepad1.RightStickY.

Select another “set tgtPosition to” block and drag it to the position immediately below the first “set tgtPosition to” block.

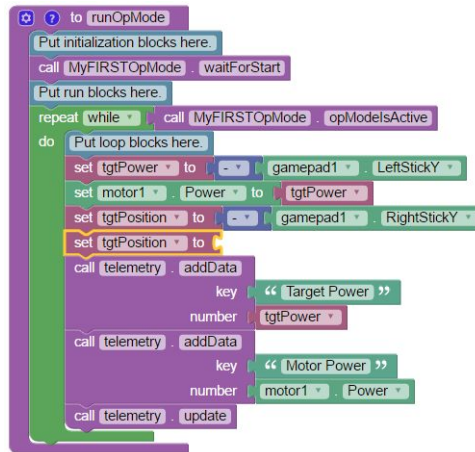


Figure 113 – Drag a second “set tgtPosition to” block below the first “set tgtPosition to” block.

In the “Math” category of the design palette, select the *addition* block.

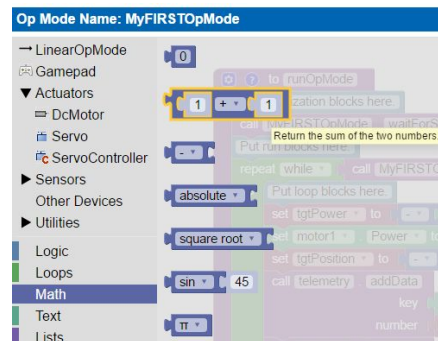


Figure 114 – Select an addition block.

Drag the addition block so it clicks into place on the right hand side of the second “set tgtPosition to” block.



Figure 115 – Drag the addition block so it clicks into place on the right side of the second “set tgtPosition to” block.

From the “Variables” category, select a “tgtPosition” block from the “Variables” category.

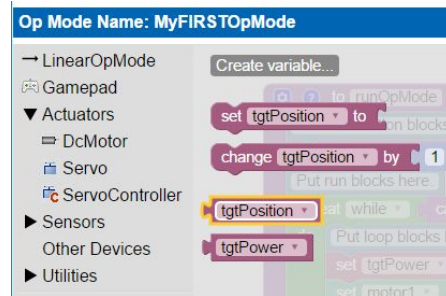


Figure 116 – Select a “tgtPosition” block and drag it to the addition block.

Drag the “tgtPosition” block so that it clicks into position of the left *argument* of the addition block. Your op mode should look like the one depicted in Figure 117 below. This operation will increment the value of tgtPosition by one.

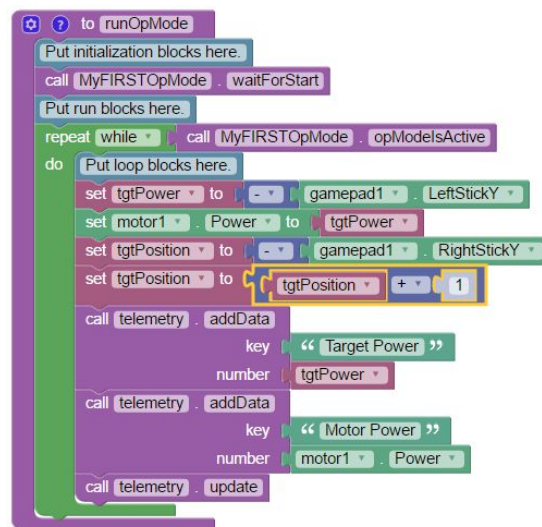


Figure 117 – Add a “tgtPosition” block to the left argument of the addition block. This will increment tgtPosition by 1.

Click on the “Variables” block and drag a third “set tgtPosition to” block into your op mode. Click on the “Math” block again and drag an addition block onto the “set tgtPosition to” block.



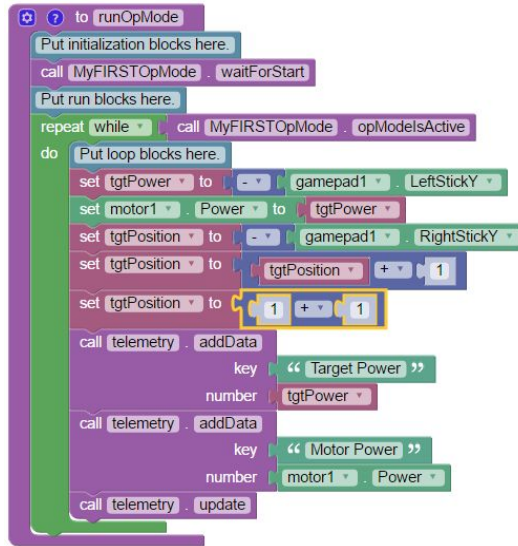


Figure 118 – Drag a third “set tgtPosition to” block to your op mode and drag an addition block to the right hand side.

In the middle of the addition block, there should be a little triangular shaped arrow to the right of the plus symbol. Click on this symbol to display a list of math operations that can be performed in the block (see Figure 119 below). Select the division symbol (“÷”) from the drop down list of operators.

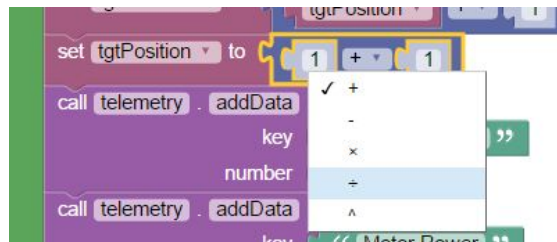


Figure 119 - Select the division operator from the drop down list.

The binary arithmetic operation should change from addition to division and your op mode should look like the image depicted in Figure 120 below.

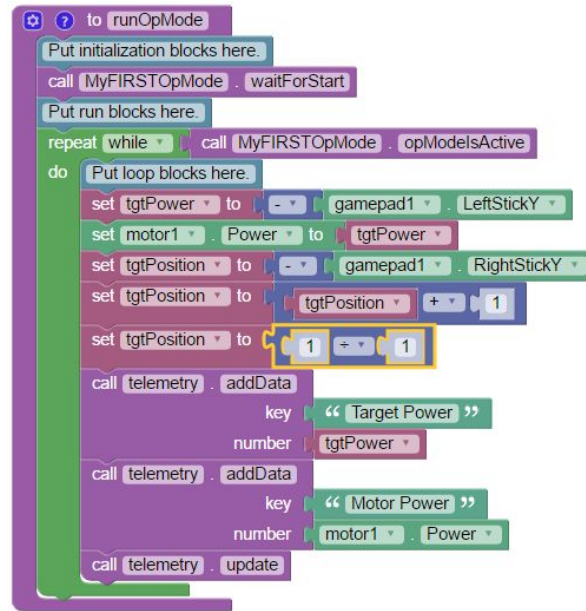


Figure 120 - The binary arithmetic operation for the third "set tgtPosition to" block should now be division.

Drag a "tgtPosition" block to the left argument of the division block.



Figure 121 – Drag a "tgtPosition" block to the left argument of the division operation block.

Click on the number "1" in the right argument of the division operation block. Highlight the number "1" and type in the number "2" to overwrite the default value.



Figure 122 - Change the right hand argument to the number "2" for the division block.

Your op mode should look like the image shown below.

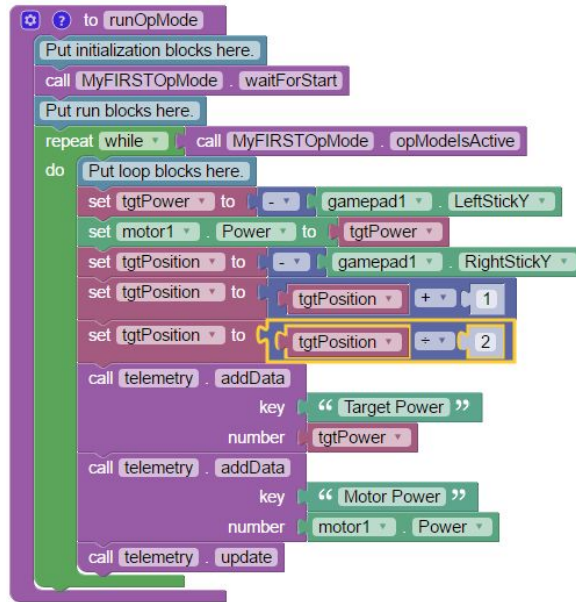


Figure 123 – RightStickY is negated, offset by one, then scaled by 2.

Now you need to set the servo position to the value of the variable “tgtPosition”. Click on the “Actuators”->”Servo”-> “set servo1.Position to” block from the design palette.

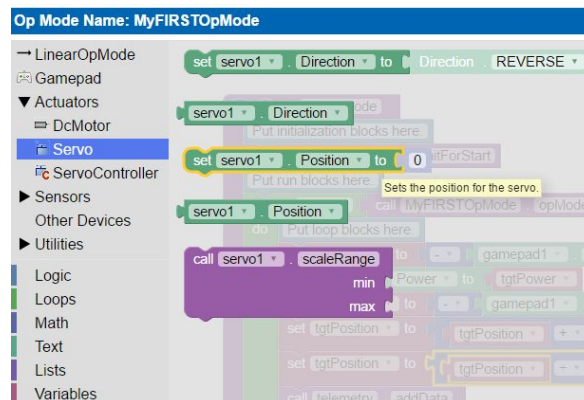


Figure 124 - Select the "set servo1.Position to" block from the design palette.

Drag the “set servo1.Position to” block to the spot immediately below the third “set tgtPosition to” block.

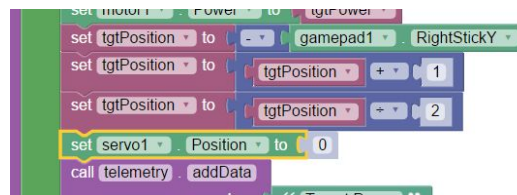


Figure 125 – Drag the “set servo1.Position to” block to the spot immediately below the third “set tgtPosition to” block.

Drag a “tgtPosition” block from the “Variables” category on the palette and click it in place on the right hand side of the “set servo1.Position to” block. Your op mode should look like the op mode depicted in the image below.

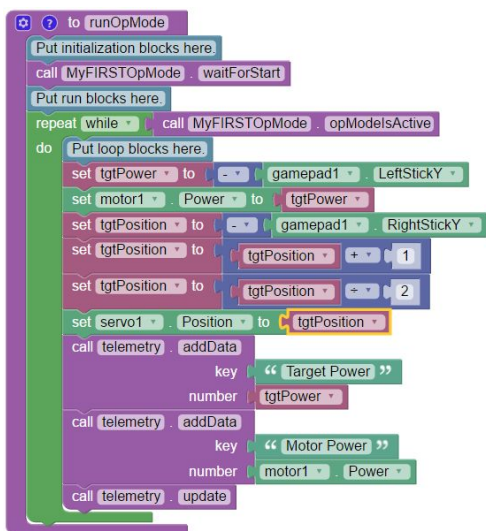


Figure 126 – Drag a “tgtPosition” block to the right side of the “set servo1.Position to” block.

## 4.7 Adding Telemetry Blocks to Display Position Information

Modify your op mode to include two additional telemetry blocks to display the position information on the Driver Station. Your op mode should look like the one shown in the figure below.

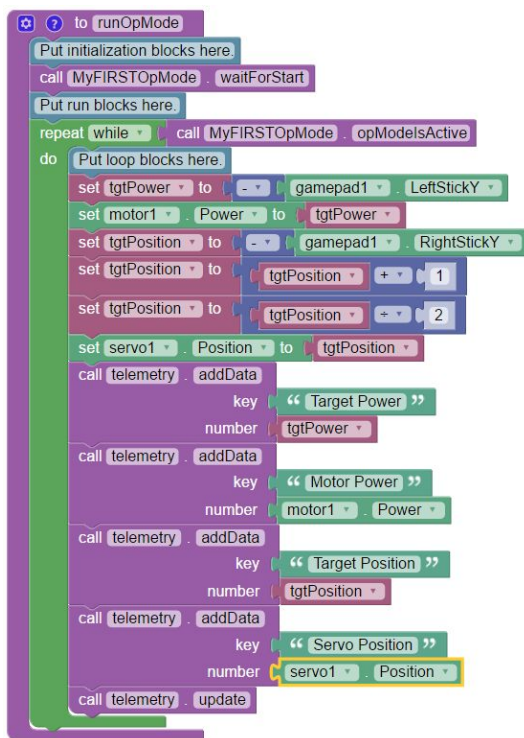


Figure 127 - Add telemetry blocks for target position and servo position.

## 4.8 Save Your Edited Op Mode!

Before you can run your newly edited op mode, you must make sure to save it successfully onto the programming mode server. Press the “Save Op Mode” button in the upper left hand corner of your browser window. If the op mode was successfully saved, the words “Save completed successfully” should appear in green letters to the right of the button.

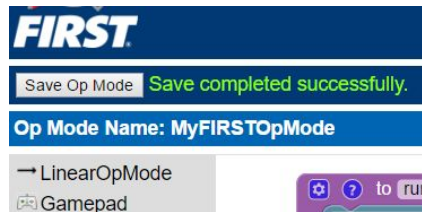


Figure 128 - Save the newly edited changes to the programming mode server.

Once you have saved your updated op mode, use the back arrow on the Android phone (see Figure 82 on page ) to exit programming mode and return to normal operation for your Robot Controller.

Once you have exited programming mode, select and test your updated op mode (see section 3.3.10 on page for details on selecting and running your op mode). The right joystick should now cause the servo to rotate.

## 5 Using Sensors

### 5.1 What is a Sensor?

A *sensor* is a device that lets the Robot Controller get information about its environment. The *FIRST* Tech Challenge control system supports a module, known as the *Device Interface Module*, that can be used to read data from a variety of commercially available sensors. For this example, we will connect a *touch sensor* and an *optical distance sensor* to the Device Interface Module and display the output from the devices onto the Driver Station screen.

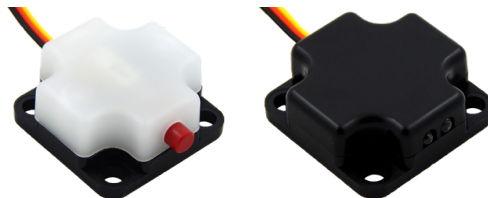


Figure 129 – The Modern Robotics touch and optical distance sensors.

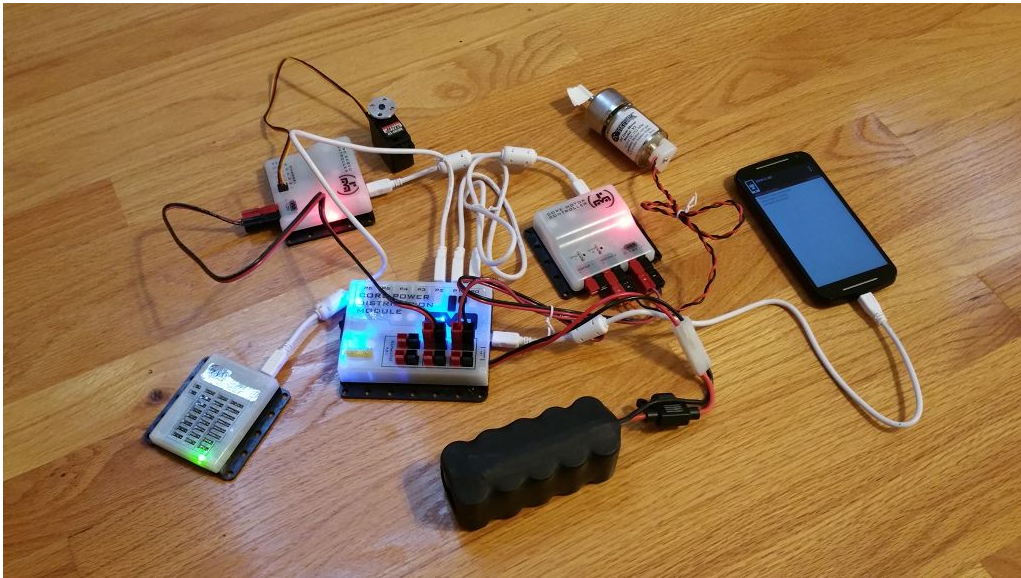
### 5.2 Connecting Your Sensors

The Device Interface Module needs to be connected to the Power Distribution Module’s USB hub. However, unlike the motor and servo controllers, the Device Interface Module does not need an



additional 12V power connection. Instead, the Device Interface Module draws its power from the Power Distribution Module's USB hub.

Using a USB cable, connect your Device Interface Module to the Power Distribution module so that your test setup looks like the setup depicted in Figure 130 (see below). Note that the sensors have not yet been connected to the Device Interface Module in Figure 130 . Also note that the switch on the PDM is turned on.



**Figure 130 - Connect the Device Interface Module to the Power Distribution Module's USB hub.**<sup>4</sup>

Don't forget, when you connect your phone to the input USB port of the Power Distribution Module with the Device Interface Module connected to the Power Distribution Module, the Android phone might prompt you to associate the newly detected hardware with the FTC Robot Controller app. If this happens, make sure you enable the "Use by default" option and click on the "OK" button to continue.

---

<sup>4</sup> Note the sensors have not yet been plugged into the Device Interface Module in this figure.

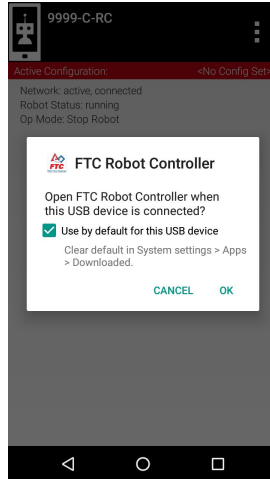


Figure 131 - If prompted by the Android device, enable the “Use by default” option and click OK to continue.

Once your Device Interface Module is connected to the USB hub, plug in the two sensors into the appropriate ports on the module. The touch sensor is digital sensor and has a three pin connector. Plug the sensor into the digital port that is labeled “D0” so that *black wire* (which represents ground for that sensor) is aligned on the *right hand side* of the port (see Figure 132).

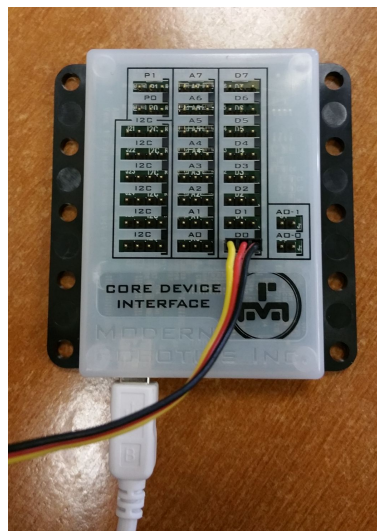


Figure 132 – Connect touch sensor to digital port “D0” with the black wire aligned on the right hand side.

Similarly, connect the optical distance sensor, which is an analog sensor, to analog port “A0” so that the black wire is aligned on the right side of the port (see Figure 133).

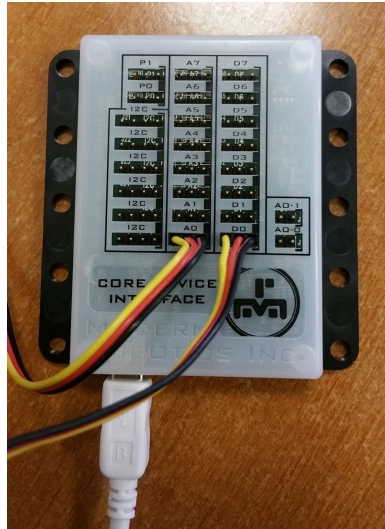


Figure 133 – Connect optical distance sensor to analog port labeled “A0” with the black wire aligned on the right hand side.

### 5.3 Modifying the Configuration File

After you have successfully connected the Device Interface Module and the sensors, you need to update the configuration file for your Robot Controller to include these new devices. From the Robot Controller screen, touch on the three dots of the main screen and select “Configure Robot” to launch the main Configuration screen. Look for your existing configuration file (which should have a name of “TestConfig”) and press the “Edit” button to edit. Verify that the power is turned on for your Power Distribution Module. Press the “Scan” button to force the Robot Controller to scan for devices connected on the serial bus. If your Device Interface Module is properly connected, you should see it listed as an available device in the configuration. Touch the Device Interface Module listing (its default name should be “Device Interface Module 1” on the screen to edit its list of connected devices.

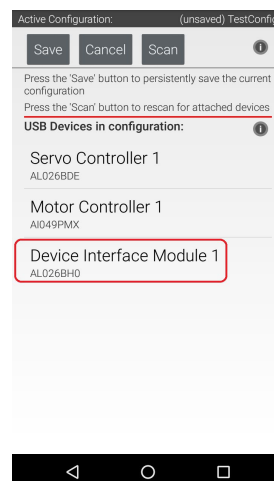


Figure 134 – Touch the Device Interface Module listing to edit its list of connected devices.



Touch the listing for the Device Interface Module to edit the list of connected devices. The optical distance sensor is an analog device that is connected to the “A0” port of the module. Touch the “Analog Input Devices” to configure the analog ports on the module.



Figure 135 – Touch “Analog Input Devices” to configure the analog sensor that is connected to this module.

The optical distance sensor is on the port labeled “A0” on the module. Configure analog port #0 so that it has a device type of “Optical Distance Sensor” and type “sensor\_ods” for the device name. Click “Done” to continue.

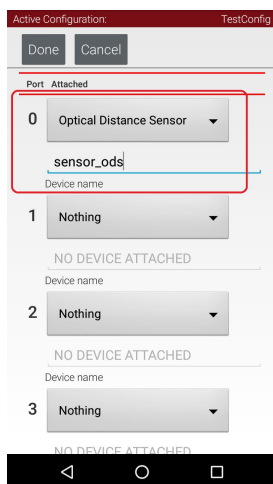


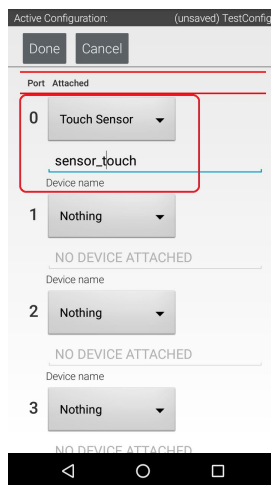
Figure 136 – Set the type of port 0 to “Optical Distance Sensor” and use “sensor\_ods” for the device name.

The touch sensor is a digital device that is connected to port “D0” of the module. Click on the words “Digital Devices” to edit the digital port.



**Figure 137 - Click on Digital Devices to edit the digital sensor information.**

Configure digital port #0 so that its device type is “Touch Sensor” and type “sensor\_touch” for its device name. Hit “Done” to continue.



**Figure 138 – Set digital port #0 to be a Touch Sensor and use “sensor\_touch” for its name.**

Save the updated information to the configuration file, and use the Android back arrow to return to the main Robot Controller screen.

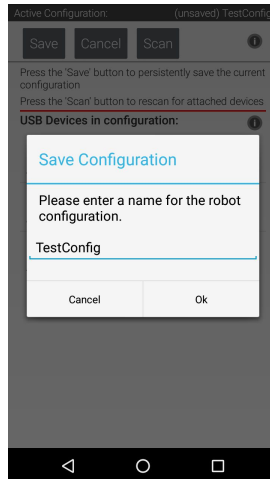


Figure 139 – Save the updated configuration information, then use the back arrow to return to the main screen.

Place the Robot Controller back into Programming Mode.

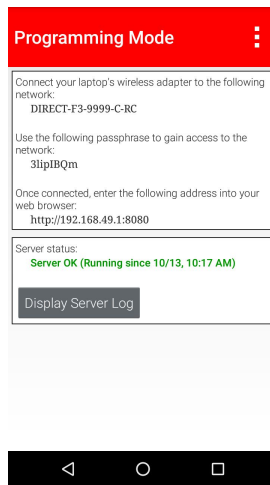


Figure 140 - Put the Robot Controller back into Programming Mode.

From your Javascript enabled browser on your laptop, close, and then re-open your project using the procedure outlined section 4.4 of this document (see page ). When you re-open your op mode project, there should be two additional devices listed under the “Sensors” category in the blocks palette.

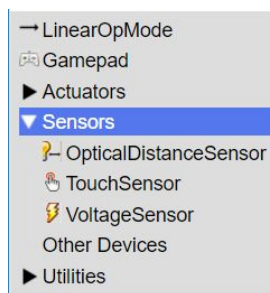


Figure 141 - "OpticalDistanceSensor" and "TouchSensor" programming blocks should now be available.

Let's use the telemetry function of the control system to display information from the optical distance sensor. The optical distance sensor has an LED and a light detector installed in a plastic housing. The light that is emitted by the sensor's LED will be reflected off of a nearby surface. The sensor uses the light detector to measure how much of the illuminated light is reflected off of the target surface. This measured light information can be used to estimate distance to a target of known reflectivity. It can also be used to detect when the sensor is over a light surface (such as a piece of white tape) or when the sensor is over a dark grey floor tile) and can be used in line-following applications.

Use a "call telemetry.addData" block and "sensor\_ods.LightDetected" block to send the measured light value back to the Driver Station. Place the "telemetry.addData" block above the "call telemetry.update" block so that your op mode looks like the following image.

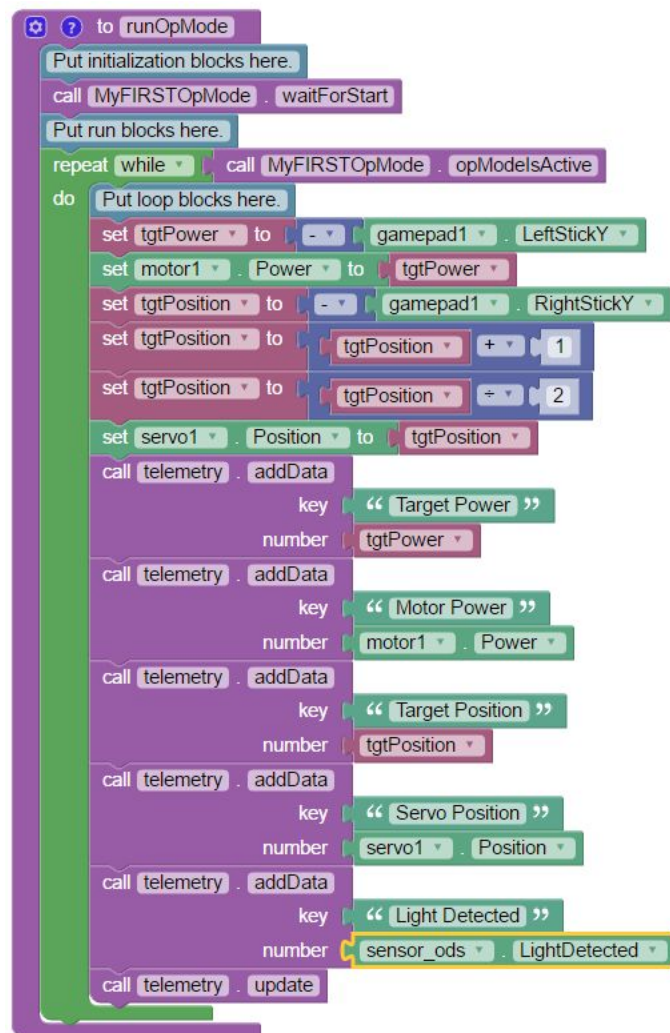


Figure 142 – Insert a "telemetry.addData" block to send the value of the light detected by the sensor to the Driver Station.

For the touch sensor, it will be helpful to use an if-do-else programming block to send information about the state of the sensor. On the design palette, click on the “Logic” category and then select the “if-do-else” block.

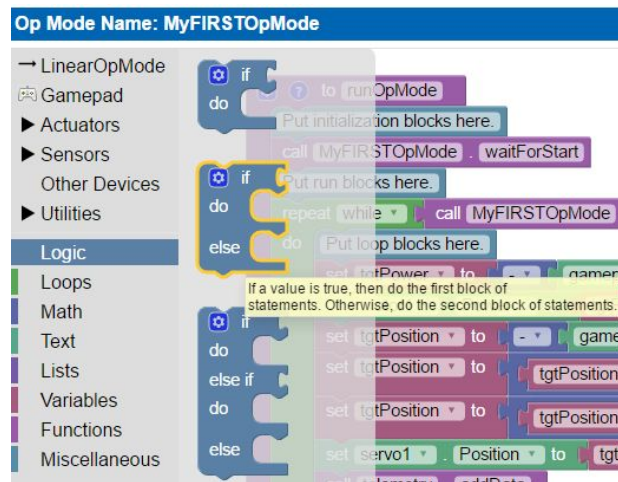


Figure 143 - Select "Logic"->"if-do-else" from the design palette.

Drag the “if-do-else” block to the position just before the “call telemetry.update”

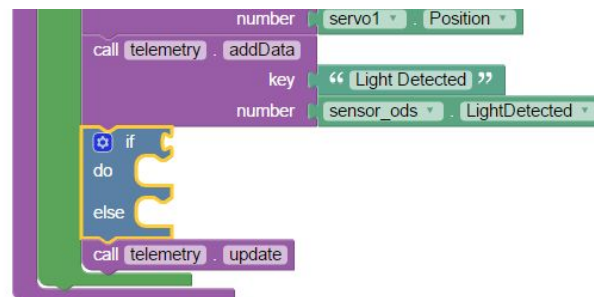


Figure 144 – Insert the “if-do-else” block immediately above the “call telemetry.update” block.

Select “TouchSensor”->”sensor\_touch.IsPressed” from the design palette.

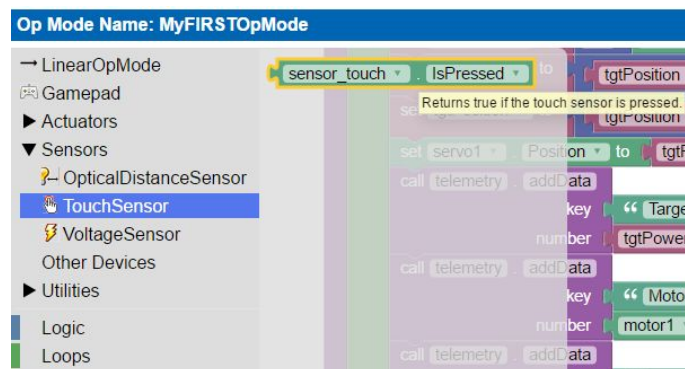


Figure 145 - Select "TouchSensor"->"sensor\_touch.isPressed" from the design palette.

Drag the “sensor\_touch.IsPressed” block so it becomes the conditional expression that is evaluated by the “if-do-else” block.

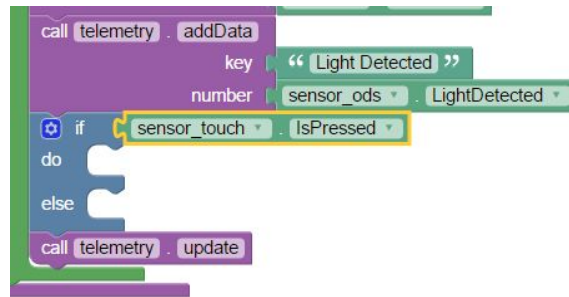


Figure 146 – Drag the “sensor\_touch.IsPressed” block so it clicks to the right of the word “if”.

If the button on the touch sensor is pressed, you want to send a telemetry message indicating this state. Click on the “Utilities” category and find and select the “Telemetry”->“call telemetry.addData” (the one that is used to send text values) block (see image below).

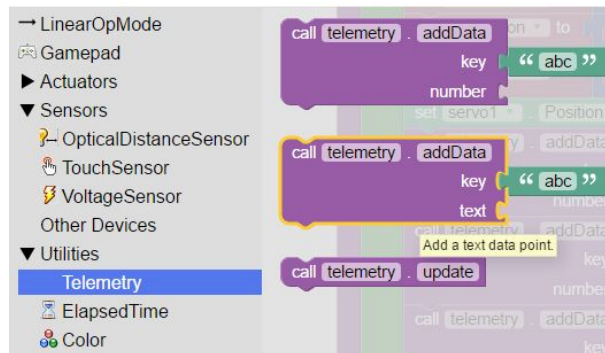


Figure 147 - Select the text addData block (and not the number addData block).

Drop the “call telemetry.addData” block to the “do” clause of the “if-do-else” block.

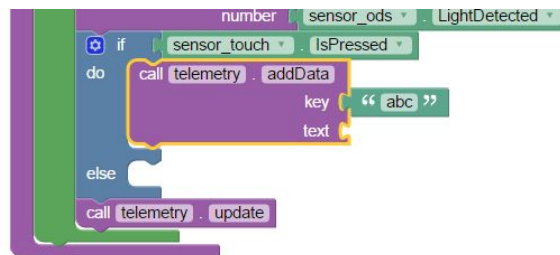


Figure 148 – Drag the “telemetry.addData” block to the “do” clause.

Highlight the key text (“abc”) and replace the text with the words “Button State” (see Figure 149).

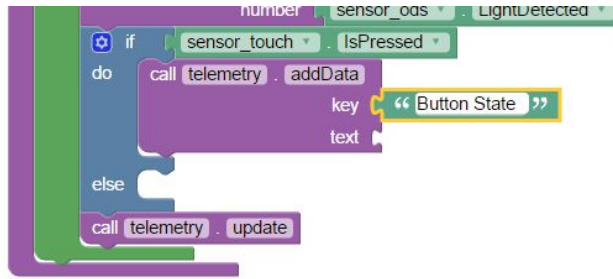


Figure 149 – Change the key text to read “Button State”.

Select a new text string block from the “Text” category of the design palette.

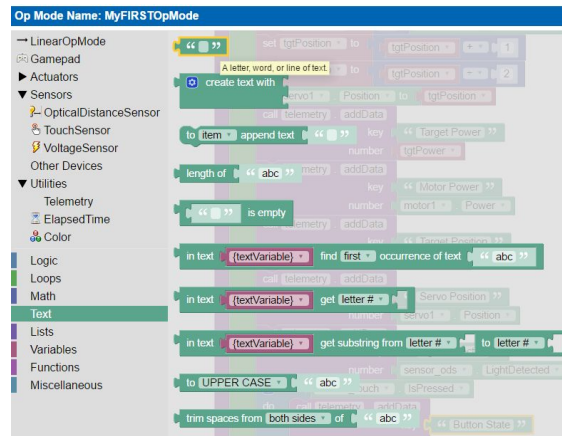


Figure 150 – Select a blank text block from the “Text” category of the design palette.

Drag the blank text block to the “text” parameter of the “call telemetry.addData” block and type in the words “PRESSED” in the block.

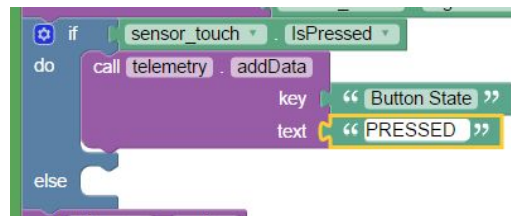


Figure 151 – Set the text parameter of the block to “PRESSED”.

Repeat the process for the “else” clause of the “if-do-else” block and set the text field to be “UNPRESSED”. Your op mode should look like the following image.

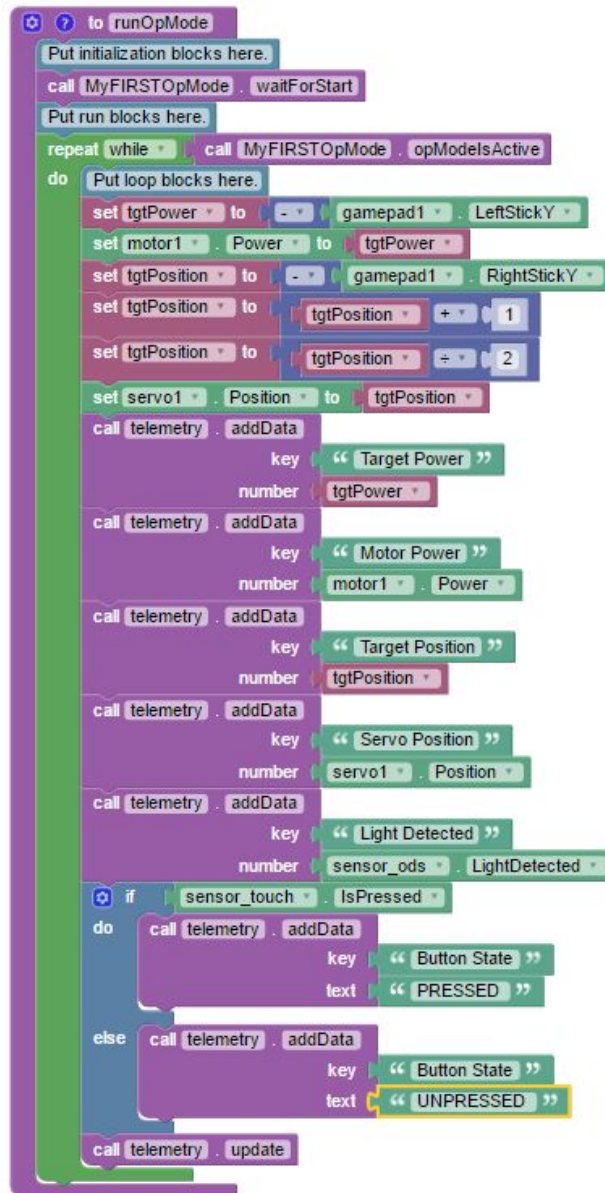


Figure 152 - Your updated op mode should look like this image.

Don't forget to save your op mode!!!

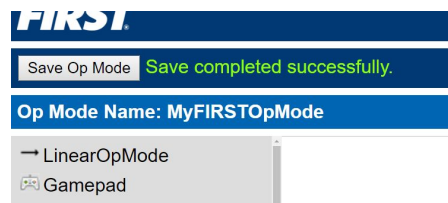


Figure 153 – Save your updated op mode.



After you have saved your op mode, exit Programming Mode, and then select and run the “MyFIRSTOpMode” from the list of available op modes. While the op mode is running, the detected ri of the touch sensor should be displayed at the bottom of the Driver Station screen. If you push the button on the touch sensor, or hold a reflective object (like your hand) in front of the optical distance sensor, you should see the values change on the Driver Station.

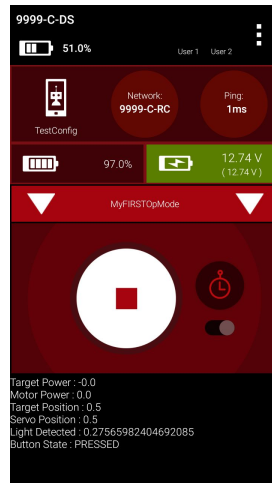


Figure 154 – The amount of light detected and the state of the touch sensor button should be displayed.

## 6 Sample Op Modes

### 6.1 Basic Tank Drive

The following op mode assumes that you have two DC motors called “left\_drive” and “right\_drive” and that the “right\_drive” motor should be reserved so that a positive power setting for that motor will move the robot forward.

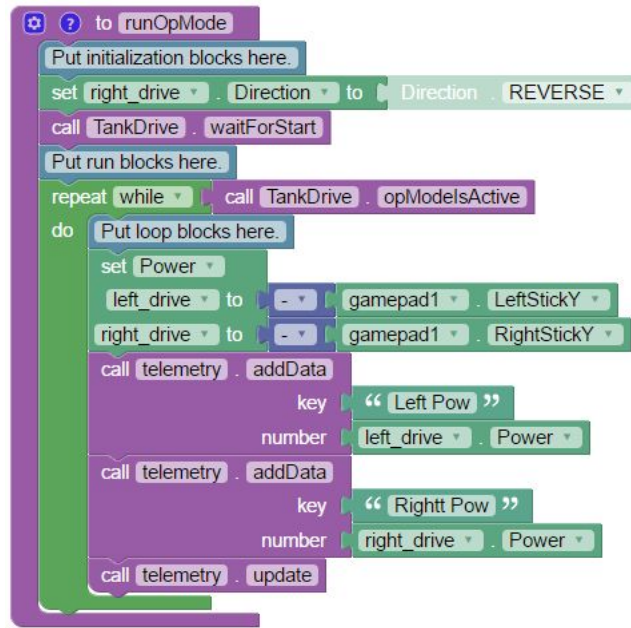


Figure 155 - TankDrive example op mode.

This op mode uses a tank drive scheme. The left joystick (y direction) controls the left motor and the right joystick (y direction) controls the right motor. To move the robot forward, the user pushes both joysticks forward. To turn to the left, the user pushes the right joystick forward and pulls the left stick backwards.

## 6.2 Basic Line Following

This example op mode assumes the following configuration:

- Left motor is called "left\_drive"
- Right motor is called "right\_drive" and should be reversed
- An optical distance sensor is included with the name "sensor\_ods"

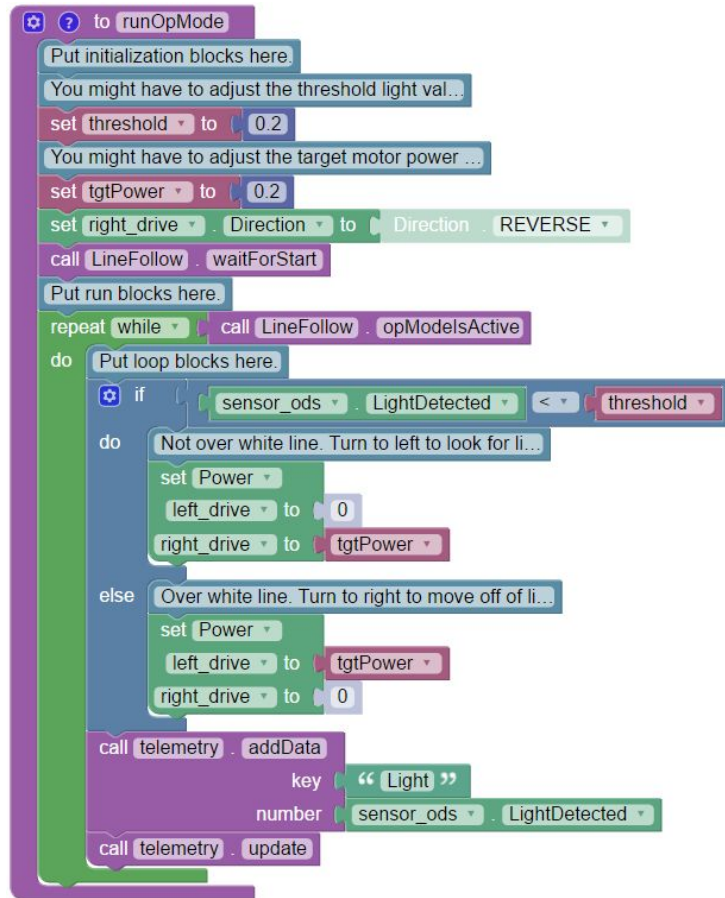


Figure 156 - Line-following op mode.

The logic for the line-following op is as follows:

1. Initialize the threshold and target power variables (note you might have to adjust each of these values to accommodate your test environment and to accommodate your robot design and motor performance).
2. Use the “set right\_drive.Direction to” block to reverse the direction of the right motor.
3. Wait for the Start command from the Driver Station.
4. While the op mode is still running, do the following.
  - a. If the light detected is below the threshold value, assume the sensor is over the dark surface and turn to the left.
  - b. Otherwise, if the light is greater or equal to the threshold value, assume the sensor is over the white line and turn to the right.
  - c. Add telemetry information on the amount of light that was detected.
  - d. Update (send) the telemetry information to the Driver Station.

## 7 Additional Tips and Information

### 7.1 Troubleshooting Tips

In this section we provide some basic tips on troubleshooting problems that you might encounter when using the blocks programming mode server to write op modes for your Robot Controller.

#### 7.1.1 Cannot See the Blocks Programming Mode Wireless Network

If you are trying to connect to the blocks programming mode wireless network so that you can create/edit an op mode for your Robot Controller, but you cannot see this wireless network listed as an available network for your laptop to connect to, verify the following items:

1. Make sure the Driver Station is successfully paired to the Robot Controller (see section 3.1 on page for details). Often times the Robot Controller's Wi-Fi Direct network will time out if it is not connected to the Driver Station.
2. Make sure the Robot Controller has been switched successfully to programming mode (see section 3.3.2 on page ).
3. Power cycle (turn off and then turn back on) your Robot Controller phone and then relaunch the FTC Robot Controller app. Reconnect the Driver Station to the Robot Controller, and then turn off the wireless adapter on your laptop for a few seconds, and then turn it back on (to force a rescan of the available Wi-Fi networks).

#### 7.1.2 "Save project failed. Error code 0."

If you attempt to save the op mode that you are currently editing, but you receive an error message indicating that the "Save project failed. Error code 0." you might have not be connected to the blocks programming mode sever:

1. Make sure the Robot Controller is in programming mode (see section 3.3.2 on page ).
2. Make sure that your laptop is connected to the blocks programming mode Wi-Fi network (see section 3.3.3 on page ).
3. If you have verified the first two items, press the "Save Op Mode" button again to re-attempt the save operation.

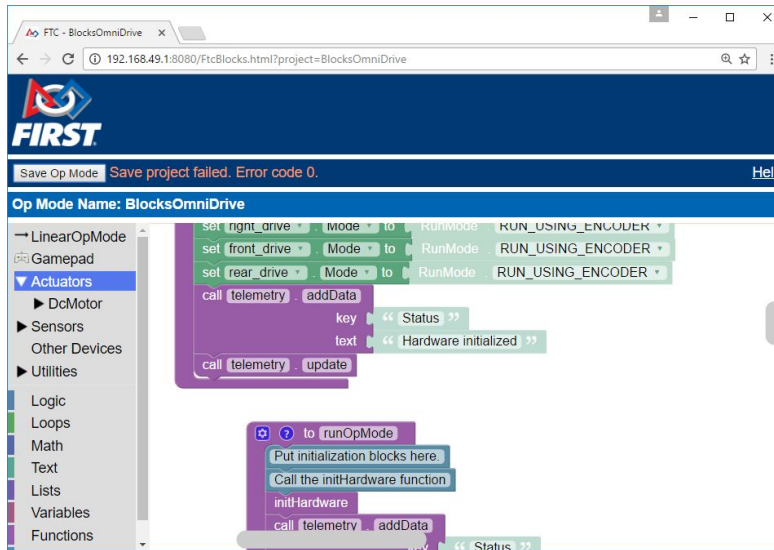


Figure 157 – If the save attempt fails, you might not be connected to the blocks programming mode network and/or server.

### 7.1.3 Op Mode Blocks Are Missing...

If you have opened an existing op mode to edit it in your Javascript-enabled browser, but the programming blocks are missing, check the following:

1. Did you remember to save the op mode the last time you edited and then exited the op mode? If you did not save the op mode after the last editing session, you might have lost some of your changes.
2. Are the blocks *collapsed* and/or in an area of the design “canvas” (or design pane) that is outside your current browser window? If so, you can use the *expand* and *cleanup* functions of the blocks programming tool to expand all of the blocks on your screen and to organize them in an easy-to-view (and easy-to-find) manner.

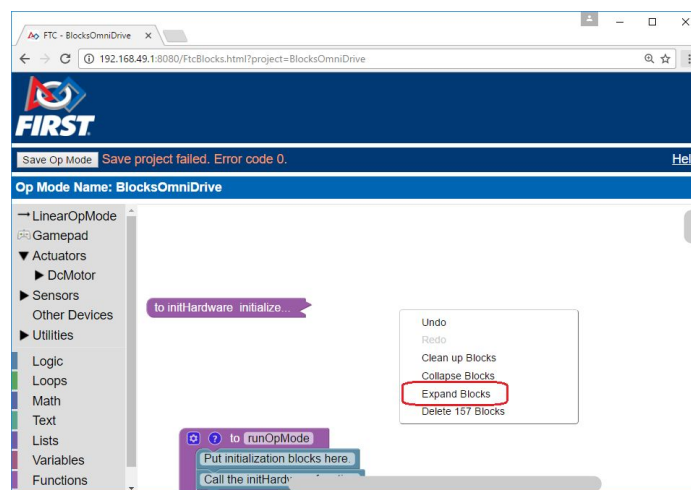


Figure 158 – You can right mouse click on “canvas” and select Expand Blocks to expand all of the blocks in your op mode.

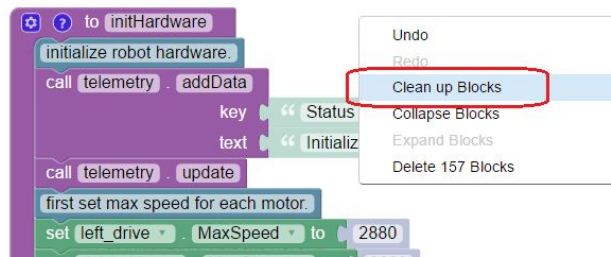


Figure 159 – Right mouse click on the canvas/design pane and select Clean up Blocks to organize all your blocks.

3. Are your programming blocks missing and you only see a solitary gray rectangular block on your screen? If this is the case, then you should check to see if the active configuration file for the Robot Controller is the same configuration file that you originally used to create the op mode. There is a bug in early versions of the blocks programming software that prevents the blocks server from properly rendering the programming blocks if the active configuration of the Robot Controller does not match the original configuration file used to create the op mode. More specifically, if some of the hardware devices (such as the DC motors or servos) from the original file are missing from the current configuration file, the blocks mode server will not properly display the programming blocks in the design pane.

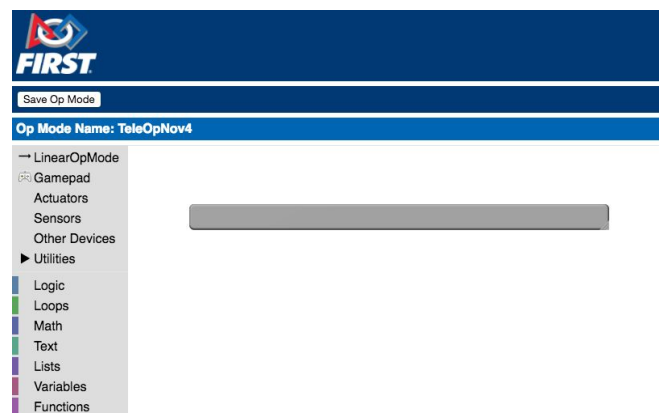


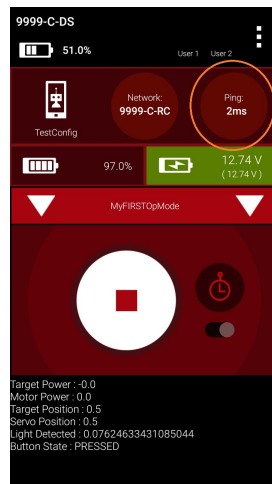
Figure 160 – Are the programming blocks missing and only a solitary gray rectangular block is visible?

#### 7.1.4 Driver Station Appears Unresponsive

If you are ready to run an op mode, but the Driver Station is unresponsive and you cannot initialize or start your selected op mode, check the following items:

1. Verify that the Driver Station is properly paired to the Robot Controller (see section 3.1 on page ).
2. Make sure that the Robot Controller is not in Programming Mode.
3. Check the *ping* times on the Driver Station main screen. The ping time is the average time it takes for the Driver Station to send a message to the Robot Controller and for the Robot Controller to acknowledge that it received the message. If the ping time is low (< 20 msec) the

wireless connection between the Driver Station and Robot Controller is good. If the ping time is consistently high (> 50 msec) there could be some wireless interference in your venue that is causing the problems between the Driver Station and the Robot Controller.



### 7.1.5 Warning: problem communicating...

If you are trying to run an op mode and you notice error messages similar to the ones displayed below, it could be that your wired connection between the phone and the electronic modules is bad.

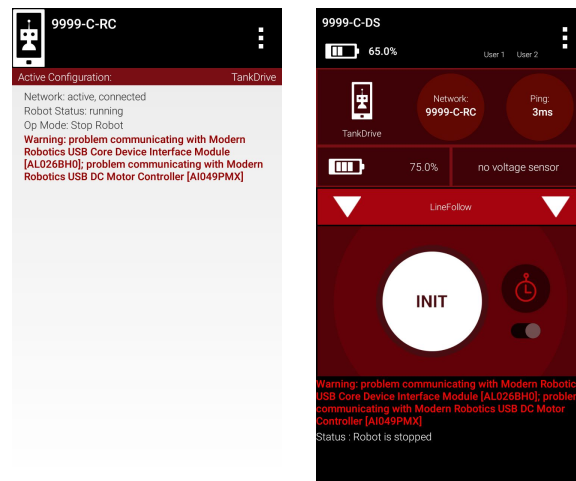


Figure 161 - A "problem communicating with..." message often indicates a bad connection between the phone and modules.

If you notice this error message, here are some things you can try:

1. Verify that the USB cables connecting the phone to the Power Distribution Module (PDM) and the PDM to the other modules (motor controller, servo controller, and device interface) are secure and well connected.
2. Verify that the 12V power cables connecting the modules that need them (motor and servo controllers) are properly secured and connected from the PDM to the other modules.

3. Try to do a “Restart Robot” from the pop up menu (touch the three vertical dots in the upper right hand screen of the Robot Controller or Driver Station apps).
4. If that does not work, shut down the main power switch on the Power Distribution Module, wait for 35 seconds, then power the PDM back on.