ProtoRIO Control Module
User Guide

ProtoRIO Version Beta1
December 20, 2019

ProtoRIO is a control module specifically designed to help test and prototype components such as those used in First Robotics Competition (FRC) and First Tech Challenge (FTC) robots.

Table of Contents

For support, contact the ProtoRIO support team at 4905protorioteam@gmail.com.

Introduction

The ProtoRIO Control Module is an Arduino® Leonardo-based shield and software that makes robot prototyping, testing, and troubleshooting easy and fast.

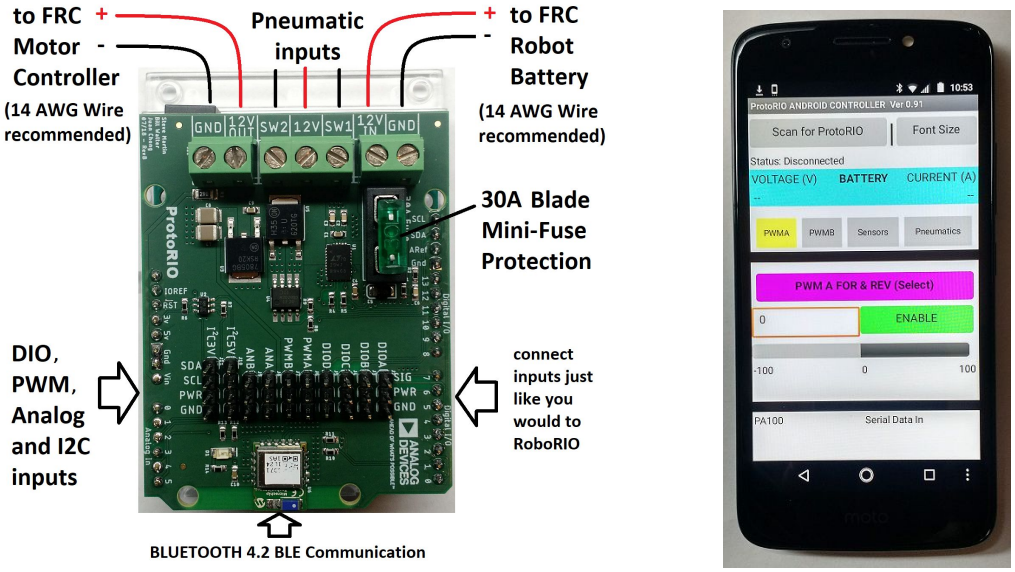
With a simple hardware configuration and Android-based applications that allow you to view and control data, ProtoRIO can communicate with distance sensors, motor controllers, servos, encoders, and pneumatic solenoids. Unlike the RoboRIO used for FRC robots, no additional support modules (PCM, VRM, PDB, etc.) or programming is required with the ProtoRIO. (Apple iOS support is possible in a future release.)

ProtoRIO includes the following:

- ProtoRIO Shield - Includes circuitry for controlling robotic components, supply monitoring/protection, and Bluetooth communication.
- Arduino Leonardo microcontroller, pre-installed with the latest Arduino software - The brains of ProtoRIO.
- ProtoRIO Basic Control application - Controls motors, servos, pneumatic solenoids, and display information from various sensors.
- ProtoRIO Advanced Motor Control application (optional) - Adds limit stops to motor control operations, which is useful for testing linear motion control within operating limits.

ProtoRIO is powered from a standard FRC or FTC robot battery using a fuse to ensure safety. ProtoRIO also includes a high performance supply monitor to observe the real-time effects of loading the battery.

Figure 1: ProtoRIO Hardware and Basic Control Application



Getting Started

Obtain the ProtoRIO hardware and software from your service representative. In addition, you must obtain the following:

- Battery connector to attach ProtoRIO to the robot battery. ProtoRIO supports most standard FRC robot batteries (typically with Anderson SB50 connector) and FTC robot batteries (typically with XT30 connector). Select a connector that is compatible with the robot battery you are using.
- Red and black 14 AWG stranded wire. Do *not* use finely stranded wire (more than 30 strands per wire).
- Android device running at least Android V2.1 with Bluetooth V4.2 support.

Each ProtoRIO has its own unique identification number, so multiple ProtoRIO control modules can be used at the same time. If you are using multiple ProtoRIO control modules, label each module with its identification number. The number is displayed on the ProtoRIO application after Bluetooth connection is made.

Step 1: Install the ProtoRIO Applications

To install the ProtoRIO Basic Control application and the optional Advanced Motor Control application on an Android device:

1. Go to Settings > Security > Unknown sources and enable “Allow installation of apps from unknown sources.”
2. Use QRC codes in Figure 2 and Figure 3 to access the applications and follow the installation prompts.
3. Go to Settings > Security > Unknown sources and disable “Allow installation of apps from unknown sources.”

Figure 2: Basic Control Application QRC

Figure 3: Advanced Motor Control Application QRC

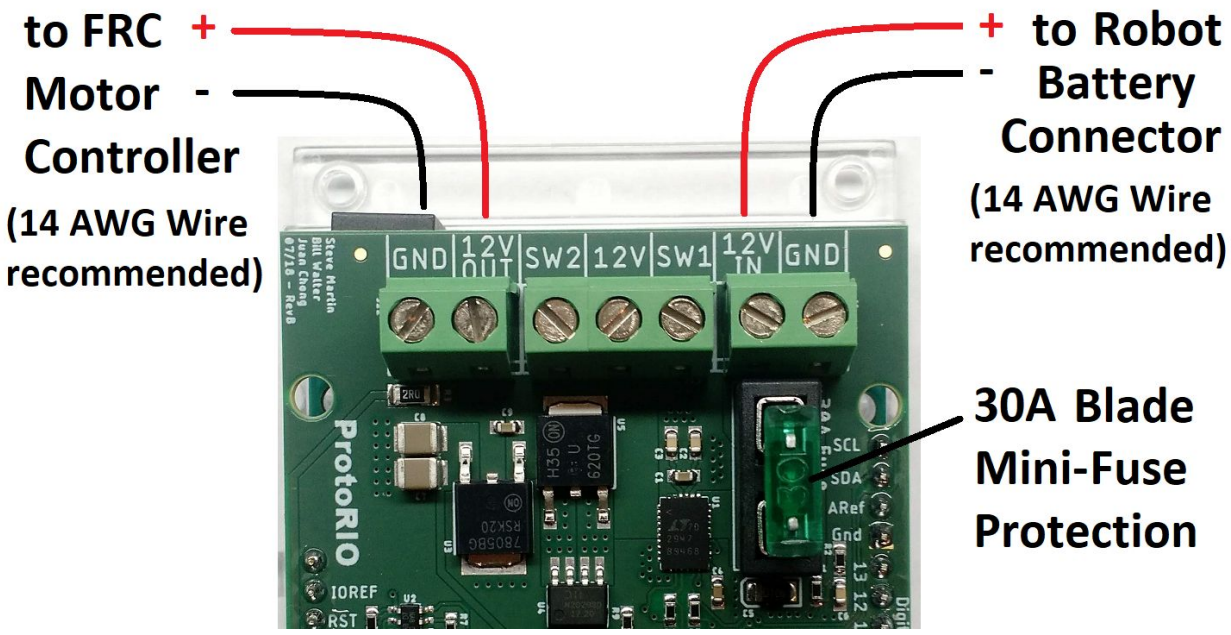
Step 2: Connect ProtoRIO to the Robot Battery

Refer to Figure 4 and follow these steps to connect ProtoRIO to the robot battery:

1. Cut two, 18-inch lengths of 14 AWG stranded wire, one red and one black.
2. Strip enough insulation from one end of each wire to attach to the battery connector.
3. Solder or crimp the wires to the battery connector according to the manufacturer's recommendation, as follows:
   o Red wire to the positive battery terminal side of connector.
   o Black wire to the negative battery terminal side of connector.
4. Strip ¼ inch of insulation from the free ends of the wires.
5. Insert the free end of the red wire into the ProtoRIO terminal labeled `12V IN` and tighten the screw.
   Note: Make sure all wire strands are contained in the terminal block. Exposed wire strands could cause a shorting hazard.
6. Insert the free end of the black wire into the ProtoRIO terminal labeled `GND` and tighten the screw.
7. Attach the connector to the connector on the robot battery.

Note: You must be sure that the red wire on the ProtoRIO connector matches up with the red wire on the robot battery connector. ProtoRIO is not reverse-battery-protected, and it will fail if the battery is not properly connected. ProtoRIO is powered whenever it is properly connected to the battery.

Figure 4: Battery and Motor Controller Power Connections to ProtoRIO



Step 3: Start the Basic Control Application

After installing the ProtoRIO applications, several icons will appear on the Android device, each with a descriptive label.

To start the Basic Control Application, click the icon labeled `ProtoRIO_BTLE`, as shown in Figure 5.

Figure 5: Basic Control Application Icon

Step 4: Connect to ProtoRIO

On the Android device running the Basic Control Application, refer to Figure 6 and follow these steps:

1. Select "Scan for ProtoRIO."
2. When "ProtoRIO_*n*" appears, select "Stop Scanning."
3. Select "ProtoRIO_*n.*"

The connection is successful when you see the following:

- STATUS shows "Connected to - ProtoRIO_*n.*" (See #4 in Figure 6.)
- Battery Voltage appears. (See #5 in Figure 6.)
- Battery Current appears. (See #6 in Figure 6.)

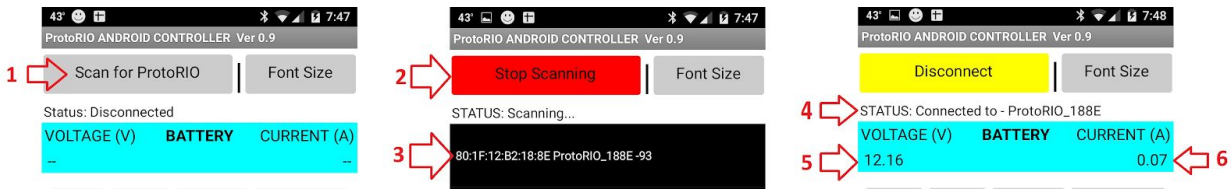


Figure 6: Basic Control Application Connected to ProtoRIO

Changing the Basic Control Application Font Size

The default font size for the ProtoRIO Basic Control Application is 14 points.

To change the font size, select "Font Size" and modify the size. The new font size is saved and automatically set whenever the application is launched.

Disconnecting from ProtoRIO

To disconnect ProtoRIO, on the Android device, select "Disconnect."

ProtoRio Hardware Configurations

ProtoRIO is powered by a standard 12 volt, sealed lead acid battery used with FRC robots or a 12 Volt, NiMH batteries used with FTC robots. A fuse is included on ProtoRIO so it can be safely connected to a robot battery (see Figure 1).

> **Note:** ProtoRIO only requires replacement of the fuse in the event of an over current condition such as a locked motor rotor or short. Replace the fuse with a standard 30AMP Automotive blade mini-fuse.

To control a wide array of robotic components, ProtoRIO includes the following hardware ports, as shown in Figure 1:

- Monitored power output (12V OUT) - Monitors motor controller supply current.
- Two PWM ports (PWMA and PWMB) - Enables PWM control of servos and motor controllers.
- Four digital I/O ports (DIOA thru DIOD) - Provides access to sensors and encoders.
- Two Analog ports (ANA and ANB) - Provides access to potentiometers and analog sensors.
- One I2C Port - Supports both 3.3V and 5V communication.
- Two switched outputs (SW1 and SW2) - Controls pneumatic solenoids.

The PWM, DIO, ANALOG, and I2C ProtoRIO ports are equivalent to the RoboRIO ports and use the same connectors that connect to the RoboRIO.

The “SIG”, “PWR”, and “GND” ProtoRIO pins are equivalent to “S”, “5V” and ground symbol on a RoboRIO, respectively. Thus, it is easy to swap between ProtoRIO and RoboRIO by simply moving the connectors.

Example of Connecting a Motor Controller to ProtoRIO

To connect a PWM motor controller to ProtoRIO, refer to Figure 1 and follow these steps:

1. If the motor controller does not come with input supply wires, cut two, 18-inch pieces of 14 AWG stranded wire (one red wire and one black wire). Strip the wires and connect the wires using appropriate connectors to the supply input of the motor controller. The red wire *must* connect to the positive supply terminal of the motor controller.
2. Strip ¼ inch of insulation from the free ends of both wires.
3. Insert the free end of the red wire into the ProtoRIO terminal labeled `12V OUT` and tighten the screw.

   Note: Make sure all wire strands are contained in the terminal block. Exposed wire strands could cause a shorting hazard.

4. Insert the free end of the black wire into the ProtoRIO terminal labeled `GND` and tighten the screw.
5. Connect the PWM wire of the motor controller to PWMA or PWMB of ProtoRIO.

Note: If the motor controller is to be removed from ProtoRIO, it is recommended that a connector be used to facilitate easy removal and prevent shorting of the motor controller wires when not connected. Anderson Powerpole WP30 connectors are a good choice and widely used in FRC for their reliability.

Example of Connecting a Pneumatic Solenoid to ProtoRIO

To connect a pneumatic solenoid to ProtoRIO, refer to Figure 1 and follow these steps:

1. Strip ¼ inch of insulation from the wires coming from the pneumatic solenoid.
2. Connect the positive or red wires (one wire if single, two wires if dual) from the solenoid to the ProtoRIO terminal labeled `12V` and tighten the screw.
3. Connect one negative or black wire coming from the solenoid to the terminal labeled `SW1` and tighten the screw
4. (Optional for dual control) Connect the second negative or black wire coming from the solenoid to the terminal labeled `SW2` and tighten the screw.

Note: Solenoid control works *only* with 12V solenoids and will not work with 24V solenoids.

Basic Control Operations

To use the ProtoRIO Basic Control Application, start the application, as described in Step 3: Start the Basic Control Application. The following sections describe some common usage scenarios.

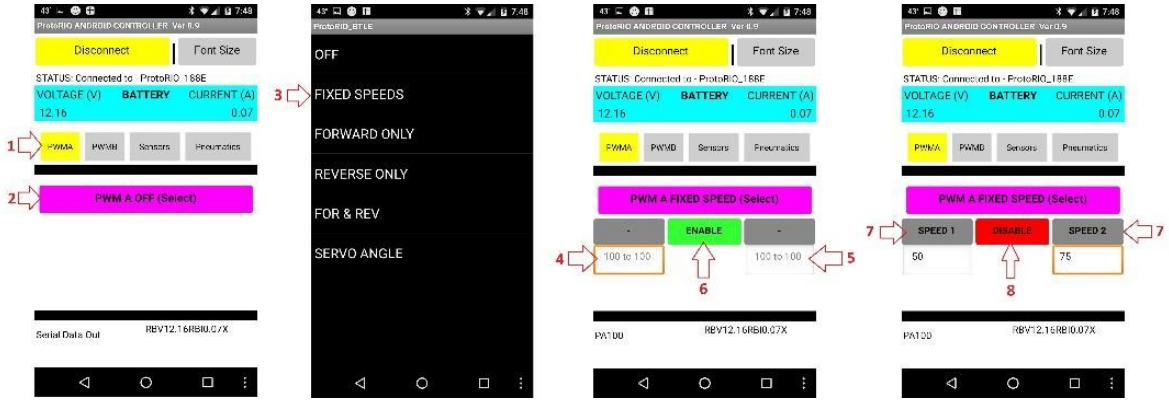
Using ProtoRIO to Control a PWM Device

MOTOR CONTROL: See Example of Connecting a Motor Controller to ProtoRIO for information about connecting a motor controller supply input to ProtoRIO. Connect the PWM motor controller to the desired PWMA or PWMB port. Both ports are equivalent, so selection is arbitrary.

There are four methods for controlling motor controllers connected to a PWM port: FIXED SPEEDS, FORWARD ONLY, REVERSE ONLY, and FOR & REV. As the method names indicate, they allow the motor to be run at two fixed speeds, one direction only, or in both directions.

SERVO CONTROL: The PWM ports can also be used to control a servo. Connect the servo to PWMA or PWMB port and select the SERVO ANGLE method. ProtoRIO can supply up to 1A of current to a servo, which is adequate to run servos approved for use in FTC and FRC.

FIXED SPEEDS Example

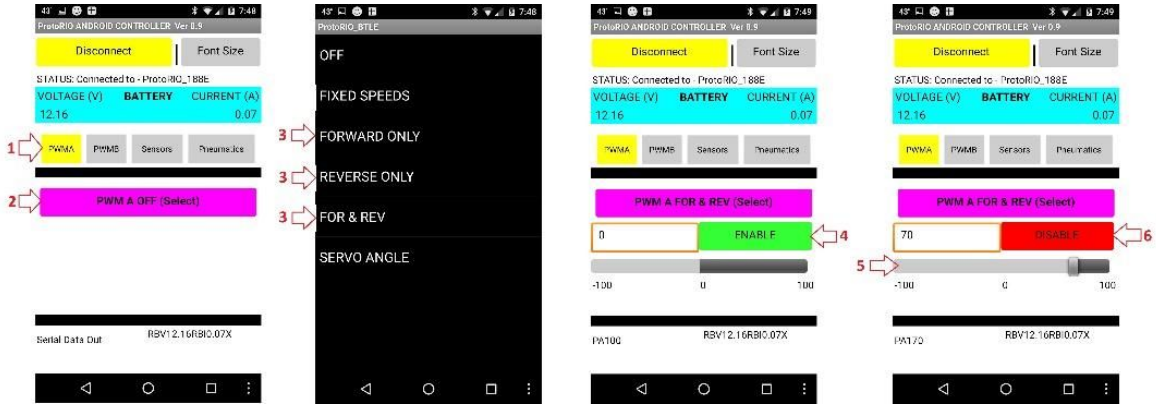


1) Select the PWMA or PWMB tab (PWMA shown).
2) Select "PWM A OFF (Select)" to bring up the selection window.
3) Select "FIXED SPEEDS" method.
4) Enter the desired speed (-100 to 100) in the first box.
5) Enter the desired speed (-100 to 100) in the second box.

   Note: Number represents direction (+/-) and % of full speed.

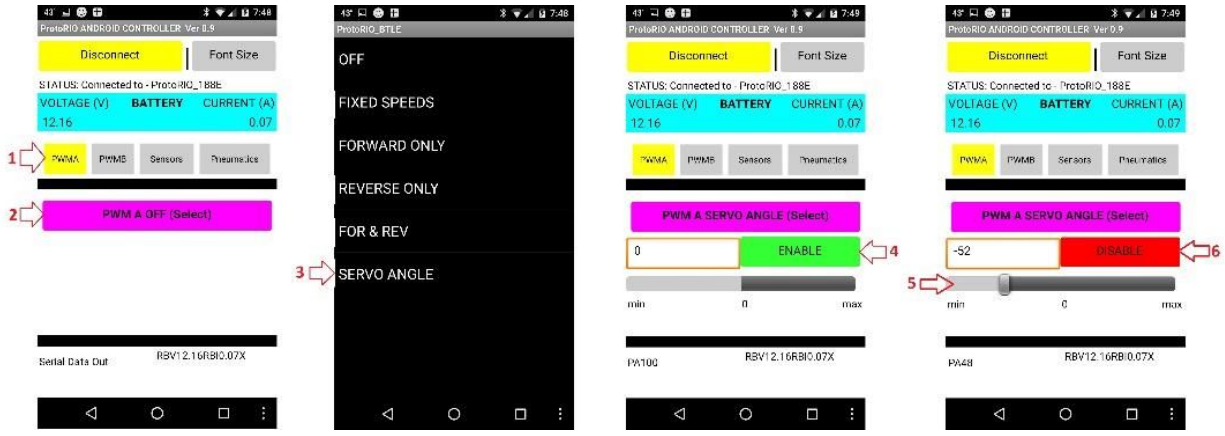
6) Select "ENABLE" to enable motor controller.
7) Select "SPEED 1" or "SPEED 2" to set the speed.
8) Select "DISABLE" to disable the motor controller.

ADJUSTABLE SPEED Example



1) Select the PWMA or PWMB tab (PWMA shown).
2) Select "PWM A OFF (Select)" to bring up the selection window.
3) Select the desired operation "FORWARD ONLY", "REVERSE ONLY" OR "FOR & REV."
4) Select "ENABLE" to enable the motor controller.
5) Use the slider to set the motor speed.
6) Select "DISABLE" to disable the motor controller.

SERVO ANGLE Example



1) Select the PWMA or PWMB tab (PWMA shown).

2) Select "PWM A OFF (Select)" to bring up the selection window.

3) Select "SERVO ANGLE."

4) Select "ENABLE" to enable the servo.

5) Use slider to change the servo angle.

6) Select "DISABLE" to disable the servo.

Reading Sensors Connected to ProtoRIO

1) Select the “Sensors” tab.

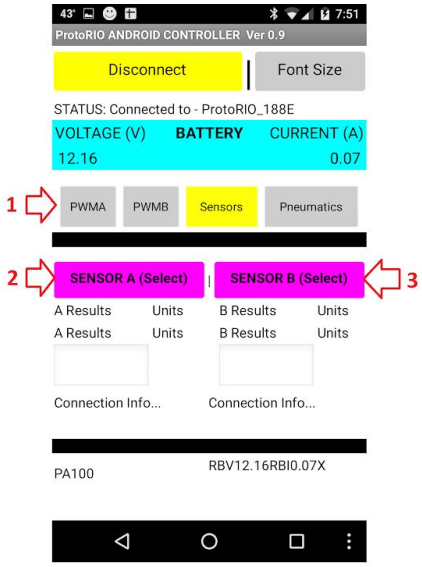
2) Select "SENSOR A (Select)" to bring up the sensor A selection window. Available sensors are:

- NONE - Select to disable Sensor A channel
- ENCODER
- HC-SR04 ultrasonic distance sensor
- MAXBOTIX ultrasonic distance sensors (1cm and 2cm resolutions)
- SHARP IR distance sensors (150cm, 80cm, 30cm and 15cm versions)
- ANALOG READ (any sensor with an analog output such as a potentiometer)

3) Select "SENSOR B (Select)" to bring up sensor B selection window. Available sensors are:

- NONE - Select to disable Sensor B channel
- HC-SR04 ultrasonic distance sensor
- MAXBOTIX ultrasonic distance sensors (1cm and 2cm resolutions)
- SHARP IR distance sensors (150cm, 80cm, 30cm and 15cm versions)
- ANALOG READ (any sensor with an analog output)
- VL53L0X Time-of-Flight distance sensor
- LIDAR Lite distance sensor

Note that some sensors can be used on either channel (HC-SR04, Maxbotix, Sharp IR, Analog Read), but some are only available on the A channel (Encoder), and others are only available on the B channel (VL53L0X TOF and LIDAR Lite).

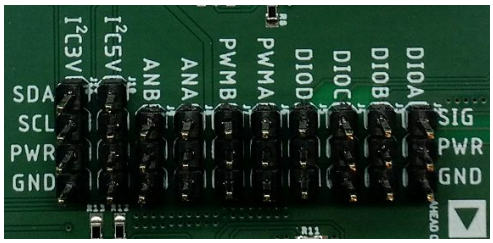


SENSOR A SELECTION

SENSOR B SELECTION

HC-SR04 Example

The “HC-SR04” can be selected from either the Sensor A or Sensor B selection button (HC-SR04 on SENSOR B shown at right).



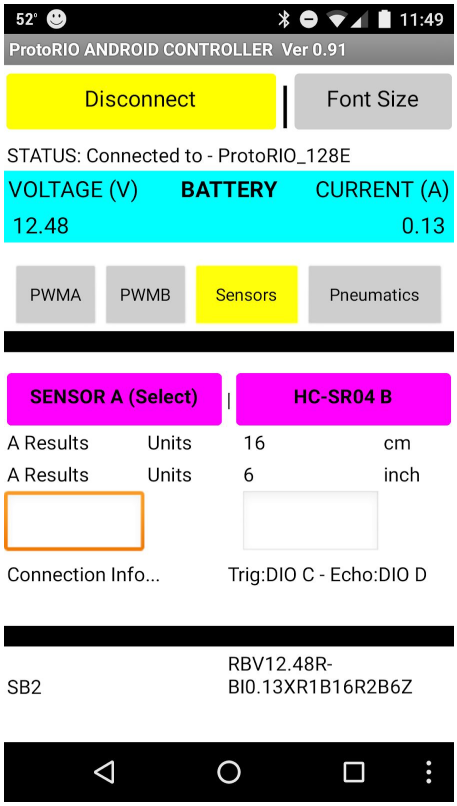
Sensor A Connection:

HC-SR04 Trig to DIOA, HC-SR04 Echo to DIOB

Sensor B Connection:

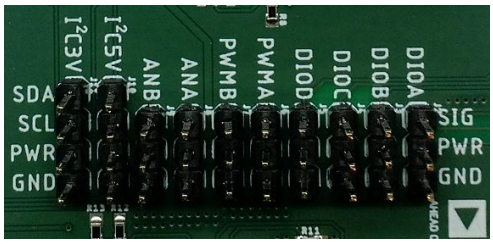
HC-SR04 Trig to DIOC, HC-SR04 Echo to DIOD

After Selecting the HC-SR04 sensor, you will see the distance displayed in “cm” and “inch” below the button. The connection information is shown below the button for convenience.

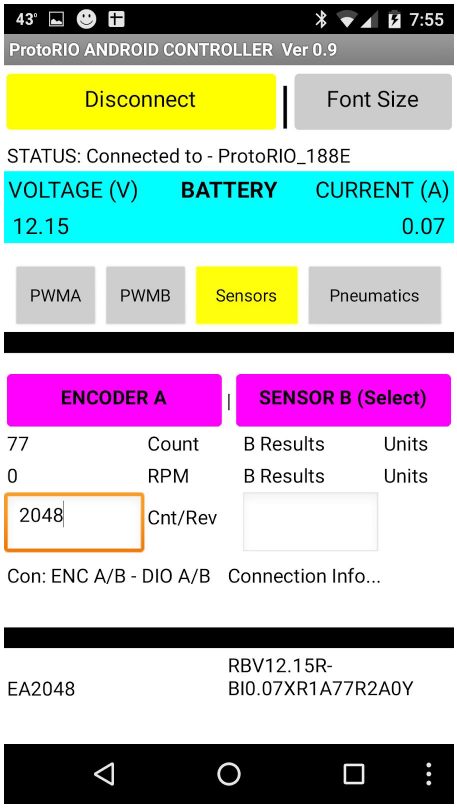


ENCODER Example

The “ENCODER” can be selected only from the Sensor A selection screen. The encoder “A” output is connected to DIOA, and the encoder “B” output is connected to DIOB.

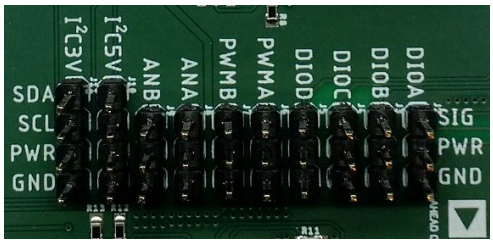


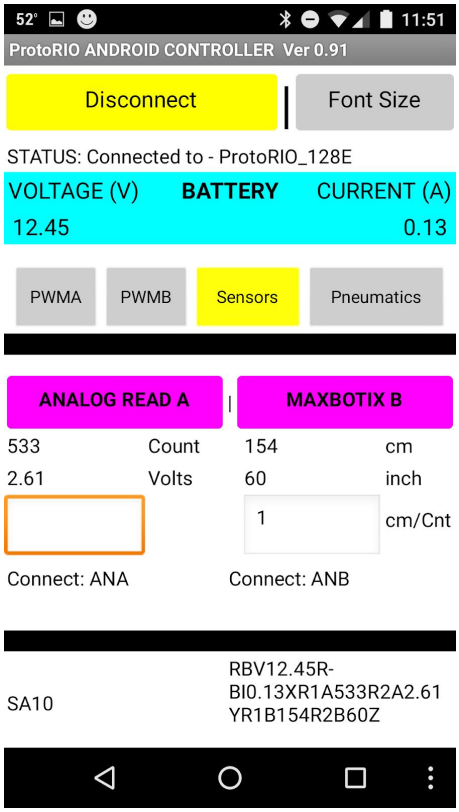
After selecting the ENCODER, you will see the total encoder count and RPM (revolutions per minute) displayed below the button. The correct count per revolution must be entered in the text box to obtain an accurate RPM value (“2048 Cnt/Rev” in the example). The connection information is shown below the button for convenience.



MAXBOTIX, SHARP and ANALOG READ Examples

The “MAXBOTIX” ultrasonic distance sensor, “SHARP IR” distance sensor, and “ANALOG READ” can be selected from either the Sensor A or Sensor B selection screens (ANALOG READ and MAXBOTIX shown at right). Sensors on channel A connect to ANA input, and sensors on channel B connect to ANB input.



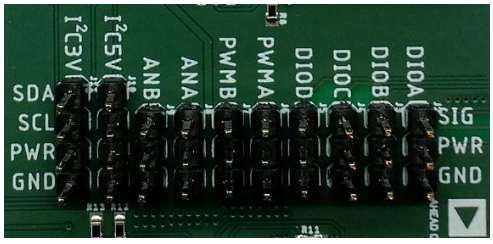


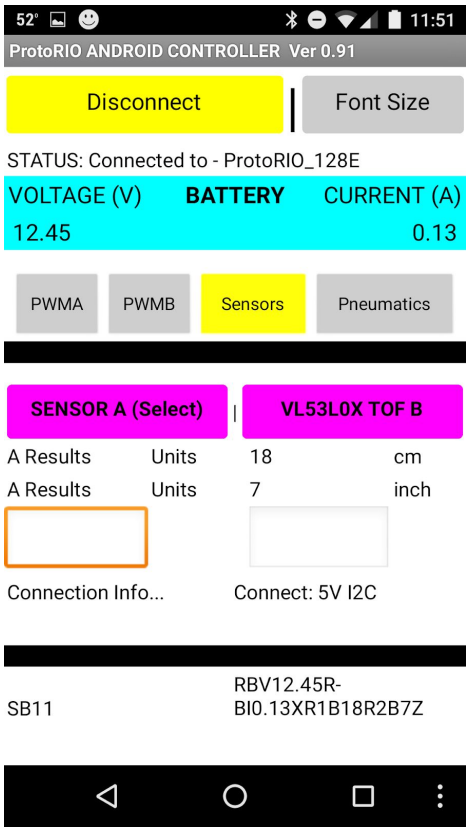
After selecting the MAXBOTIX or SHARP IR sensor, you will see the distance displayed in “cm” and “inch” below the button. The sensor type information is shown in the text box (1cm/Cnt shown).

After selecting ANALOG READ, you will see the digital count (0-1023) and voltage information below the button.

VL53L0X & LIDAR Lite Examples

The “VL53L0X” and “LIDAR Lite” sensors can be selected only from the Sensor B selection page. Both sensors connect to the I2C5V input.





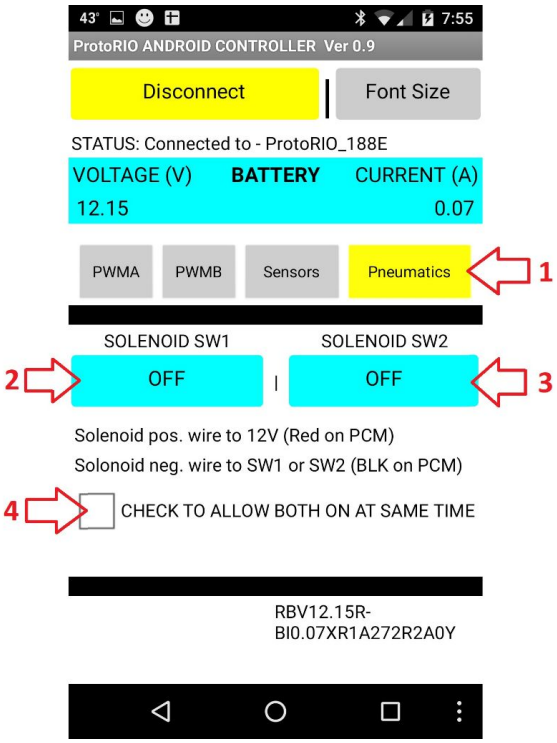
After selecting a sensor, you will see the distance displayed in “cm” and “inch” below the button. The connection information appears below the button for convenience.

VL53L0X on SENSOR B shown at right.

Using ProtoRIO to Control Pneumatics

To using ProtoRIO to control pneumatics, first connect the hardware:

1. Connect the solenoid positive terminal(s) to the center "12V" pin terminal connector on the ProtoRIO.
2. Connect the solenoid negative terminal to either the SW1 or SW2 terminal on the ProtoRIO. SW1 and SW2 can each operate one 12V solenoid.



Then, start the Basic Control Application and follow these steps:

1. Select the "Pneumatics" tab.
2. Select "SOLENOID SW1" to energize the SW1 solenoid.
3. Select "SOLENOID SW2" to energize the SW2 solenoid.
4. Check the box to allow energizing both solenoids at the same time. This box should be left unchecked for pneumatics that use a single solenoid to do both push and pull. This box should only be checked when using two separate single ended solenoids (this will probably be rare).

Advanced Motor Control Operations

After installing the ProtoRIO applications, several icons will appear on the Android device, each with a descriptive label.

To start the Advanced Motor Control Application, click the icon.

The following sections describe some common usage scenarios.

Motor Control Example

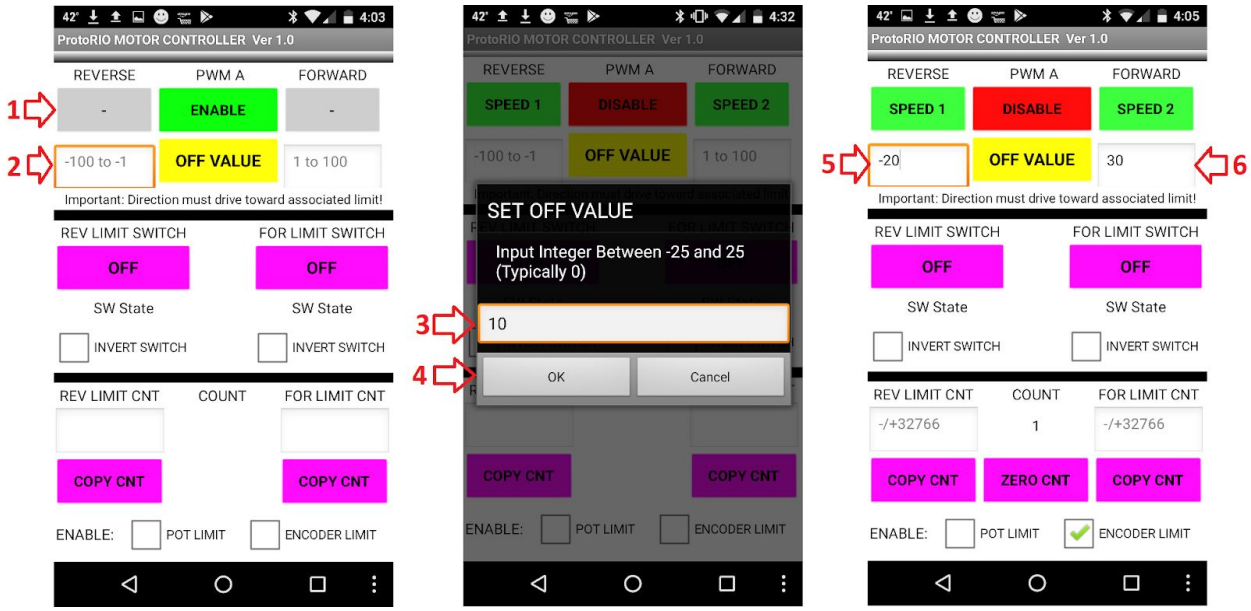
See Example of Connecting a Motor Controller to ProtoRIO for information about connecting a motor controller supply input to ProtoRIO. Connect the PWM motor controller to the PWMA port.

Then, in the Advanced Motor Control Application:

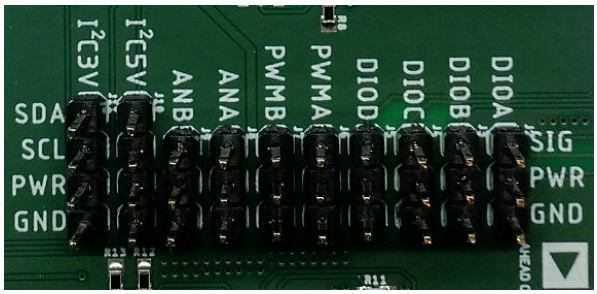
1. Select "ENABLE" to enable motor controller operation.
2. (Optional) Select "OFF VALUE" if motor runs when enabled. This is sometimes necessary for older FRC motor controllers whose off value is not exactly "0".
3. (Optional) Enter the desired off value between -25 to 25 on popup input.
4. (Optional) Select "OK." If motor is still running, go back to step 2 and enter a different value until the motor stops running.
5. Enter the desired "REVERSE" speed between -100 and -1. Reverse is always negative.
6. Enter the desired "FORWARD" speed between 1 and 100. Forward is always positive.

At this point, you should be able to select "SPEED 1" or "SPEED 2" to operate the motor. It is acceptable if the selected direction does not match what is considered forward and reverse, but you can swap the wires to the DC motor if you want the direction to match.

Note: Be careful operating the motor without the limit stops set.



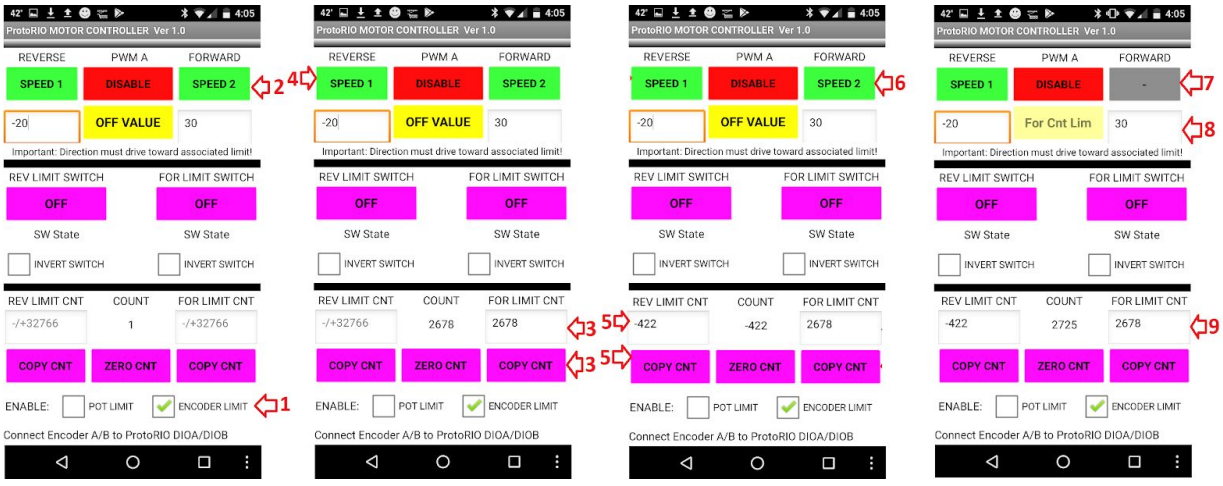
Motor Control with Encoder Limit Stops Example



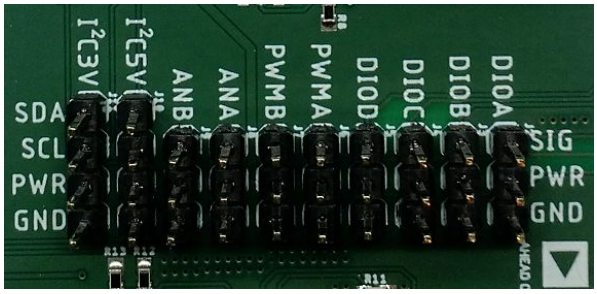
First, connect the hardware. Connect Encoder A & B to DIOA & DIOB on the ProtoRIO, respectively. Note that connection information appears at the bottom of the screen for reference.

Then, in the Advanced Motor Control Application:

1. Enable encoder by selecting the "ENCODER LIMIT" check box.
2. Select the Forward speed button to move the motor to the first stop. Use a slow motor speed to creep up on the first stop. Select the Forward speed button again to stop the motor.
3. Once at the desired stop, either select "COPY CNT" to copy the current count, or input the count manually in the "FOR LIMIT CNT" input box.
4. Select the Reverse speed button to move the motor to the Second stop. Use a slow motor speed again. Select the Reverse speed button again to stop the motor.
5. Once at the desired second stop, either select "COPY CNT" to copy the current count, or input the count manually in the "REV LIMIT CNT" input box.
6. Select the Forward speed button. The motor will automatically stop when it gets to the programmed limit.
7. The Forward speed button will be disabled to prevent over-driving the stop.
8. The "OFF VALUE" button label will change to indicate the limit ("For Cnt Lim" in this case).
9. Because the motor can not be stopped immediately, the actual count will be greater than the stop value. The over-driven amount will depending on many factors (encoder resolution, motor speed, etc) so it is best to experiment with motor speed and limit count to optimize results.



Motor Control with Potentiometer Limit Stops Example



First, connect the hardware. Connect the Potentiometer to the analog ANA input on the ProtoRIO. Note that the connection information appears at the bottom of the screen for reference.

Then, in the Advanced Motor Control Application:

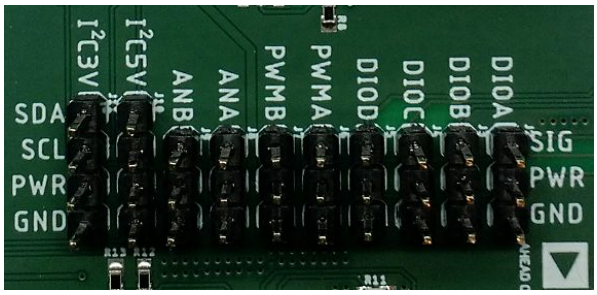
1. Enable Potentiometer by selecting the "POT LIMIT" check box.
2. In this example, the Reverse limit is manually put in the "REV LIMIT CNT" box. Note that this value must be between 0 and 1023, and the potentiometer value must head toward this value when the reverse speed is selected.
3. In this example, the Forward limit is manually put in the "FOR LIMIT CNT" box. This value must be between 0 and 1023, and the potentiometer value must head toward this value when the forward speed is selected.
4. Select the Forward speed button. The motor will automatically stop when it gets to the programmed limit.

The Forward speed button will be disabled to prevent over-driving the stop.

The "OFF VALUE" button label will change to indicate the limit ("For Cnt Lim" in this case).

Because the motor cannot be stopped immediately, the actual "COUNT" (POT value) might be greater than the stop value (as shown). The over-driven amount will depend on many factors, so it is best to experiment with motor speed and limit count to get desired results.

Motor Control with Switch Limit Stops Example



First, connect the hardware. Connect Reverse Limit Switch to DIOC and Forward Limit Switch to DIOD on the ProtoRIO. In this example we are expanding on the potentiometer example to include limit switches. This is more likely to be done with the encoder to set a starting position, but results will be the same. In this example, both limit switches are enabled, but either or both can be disabled.

Then, in the Advanced Motor Control Application:

1. Enable "REV LIMIT SWITCH" by selecting "OFF" button below "REV LIMIT SWITCH". The button label will change to DIO C indicating the limit switch input.
2. Enable "FOR LIMIT SWITCH" by selecting "OFF" button below "FOR LIMIT SWITCH". The button label will change to DIO D indicating the limit switch input.

   The "SW State" label will change to indicate the state of the limit switch when enabled.
3. If either switch state does not match the indicated result, select "INVERT SWITCH" below the switch indicating the wrong state. In the example the "FOR LIMIT SWITCH" was indicating the wrong state, so the "INVERT SWITCH" checkbox is selected.
4. Select the Forward speed button to drive the motor to the forward limit switch.

The forward limit switch will indicate "Tripped!" once tripped. The motor will be disabled once the switch is tripped.

Once the forward limit switch is tripped, the forward speed button will also be disabled to prevent further over-driving of the switch.

The "OFF VALUE" button label will change to indicate the limit ("For Sw Lim" in this case).

Note that the motor is disabled when the first limit is reached. In this example, it is the forward limit switch. If the "FOR LIMIT CNT" is set to 460, in this example, it would have stopped before reaching the forward limit switch and indicated "For Cnt Lim" on the "OFF VALUE" button.

ProtoRIO Maintenance

In some cases, you must perform some maintenance tasks.

Updating Arduino Firmware

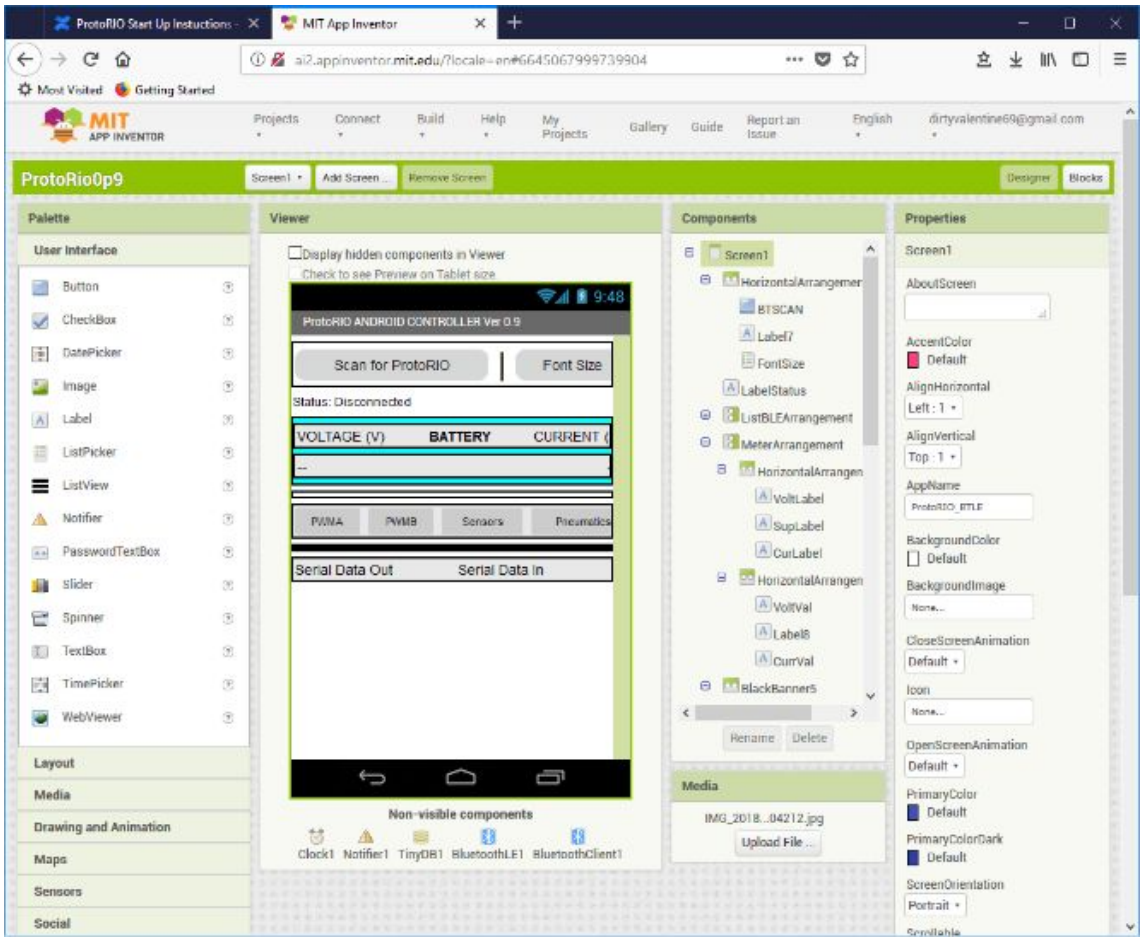
The Arduino firmware is open source and available on GITHUB as described below. ProtoRIO is preinstalled with the latest Arduino software. The steps below are only necessary in the event of an update or if you want to develop and install your own Arduino firmware.

1. Go to https://github.com/juchong/ProtoRIO.
2. Download the "ProtoRIO1p1BTLE" directory under "ProtoRIO-Leonardo-Firmware."
3. Connect the PC's USB port to the ProtoRIO's USB port.
4. Start Arduino IDE.
5. Load the "ProtoRIO1p1BTLE.ino" file.
6. Under "Sketch," select "Manage Libraries" and search for "VL53L0X." Then, select and install library **by Pololu.**
7. Under "Tools," select "Board: Arduino Leonardo."
8. Under "Tools," select the correct COM port of the ProtoRIO (appears as "Arduino Leonardo").
9. Under "Sketch," select "Upload."
10. Once Sketch has uploaded, under "Tools," select "Serial Monitor." This will bring up the serial monitor.
11. At the bottom of the "Serial Monitor," be sure that "Both NL & CR" and "115,200 BAUD" are selected.
12. The Serial Monitor should show "Type 'S' to Setup Bluetooth."
13. Type 'S' and press "Enter."
14. A list of commands should show up on the "Serial Monitor". Enter the following commands:
    a. 'N' then 'ProtoRIO' (sets BT Name).
    b. 'S' (sets the correct BT Service).
    c. 'Q' (Quit).
15. Disconnect the USB port to power down ProtoRIO.

ProtoRIO should now be correctly configured. The ProtoRIO BT module will appear as "ProtoRIO _XXXX" where XXXX will be a number specific to that ProtoRIO."

Using the MIT App Inventor Program (optional)

The ProtoRIO Android-based applications were developed on MIT App Inventor, a web-based Android development platform designed to simplify application development. To encourage further development of ProtoRIO, the MIT App Inventor project files (source code) are open source, so you can develop your own applications or compile your own apk file.



First, download the application source code for the Basic Application:

https://github.com/juchong/ProtoRIO/blob/master/MIT%20App%20Inventor/ProtoRio0p9.aia

Once downloaded, click on the following link to get to the MIT App Inventor Home page:

http://appinventor.mit.edu/explore/#

Select the "Create Apps" button at the top of the WEB page. You will need to create an account if you do not already have one. Once you have an account and are in the main programming page, select "Projects" > "Import project (.aia) from my computer", and import the "ProtoRio9p0.aia" file that you downloaded above. Once this is done you should see a page similar to the one at above.

MIT App Inventor is an easy to use graphical based programming language (like Scratch). If you have not programmed in MIT App inventor I recommend going to the MIT App Inventor Home page above and clicking on "Resources" > "Get Started" to get familiar with the language before diving into the ProtoRIO program.