App Inventor + IoT: Wheeled robot control with LinkIt 7697(BLE)

90 mins

(including Arduino IDE setup)

Level: advanced

This tutorial will help you get started with App Inventor + IoT and control a two-wheeled robot of LinkIt 7697 (Arduino compatible) with buttons on your app!

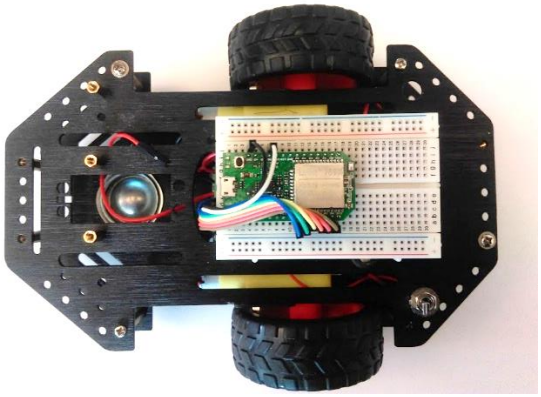
- **source .ino** / **source .aia**

Note: This is an advanced project, we assume you have intermediate understanding of App Inventor, Arduino and Bluetooth. Please check the MIT App Inventor IoT website for help getting started.

Hardware

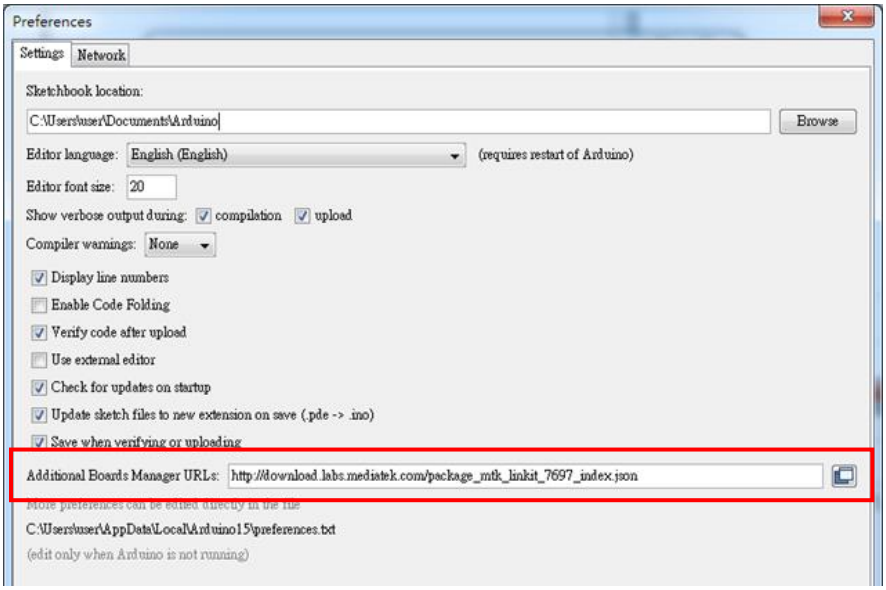
LinkIt 7697 is an Arduino compatible dev board with Wi-Fi / BLE. You can use it like just like any other Arduino, and interface with App Inventor through its Bluetooth commutation.

In this project, we are going to drive a two-wheeled robot of LinkIt 7697. By four buttons on your app, you can move the robot forwards, backwards, left, and right, as well as stopping it. LinkIt 7697 and our sample robot is like below:

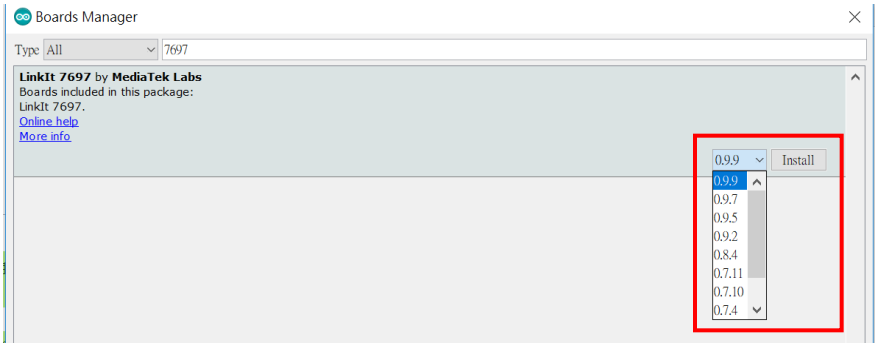


Arduino IDE Setup

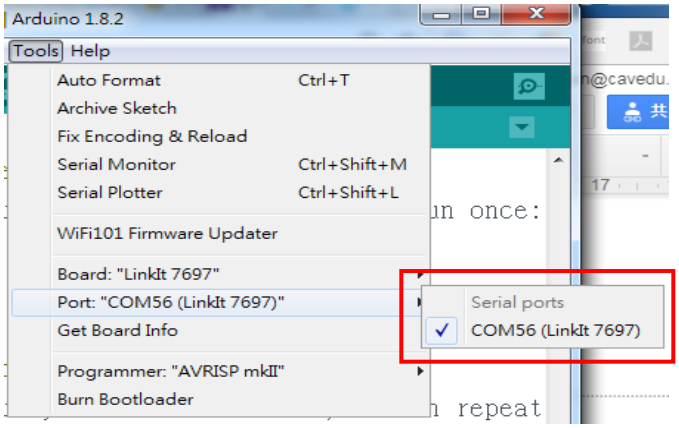
1. First go to the Arduino IDE 1.8.x version. Download the .zip file, unzip and click arduino.exe to open the IDE.
2. From **File** 👉 **Preference** menu, enter the link below to Additional Boards Manager URLs field:
    - http://download.labs.mediatek.com/package_mtk_linkit_7697_index.json



3. Open **Tools/ Board/ Board Manager**, then search "**7697**" and install the latest version of 7697 SDK.

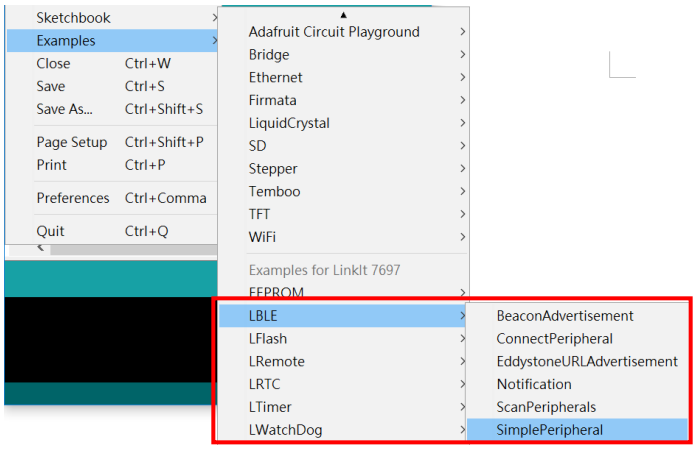


4. Download and install the **CP2102N driver** (Windows / MAC/OSX) , then check the COM port in your Device manager. Look for "**Silicon Labs CP210 USB to UART Bridge(COMXX)**", this is the COM port number of your LinkIt 7697.
5. For MAC users, it should be something like "**/dev/tty.usbserialXXX**…" and forWindows users, please check the picture below:

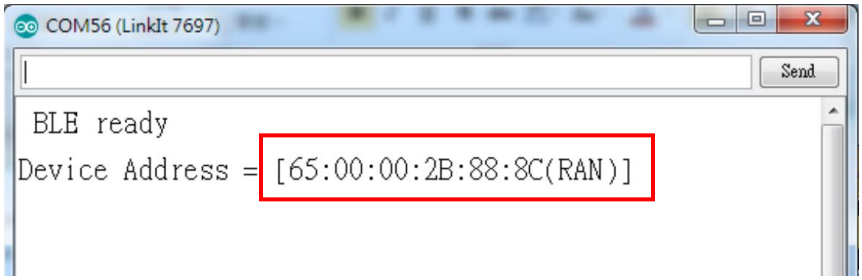


Get the BLE address of LinkIt 7697

1. For safety reasons, not every board has its Bluetooth address marked on the board (Arduino 101 is an exception). In Arduino IDE, first set the board to "**LinkIt 7697**" then open the example "**SimplePeripheral**" from File/Examples/LBLE menu.



2. Compile and upload to your LinkIt 7697 then open Arduino IDE's Serial Monitor. You should see an image like below. The [**XX:XX:XX:XX:XX:XX**] 12-digit string is the Bluetooth address of your LinkIt 7697, we have to modify the **addr** variable value of your AI2 project. Later we will use the same .ino file to receive command from App Inventor.



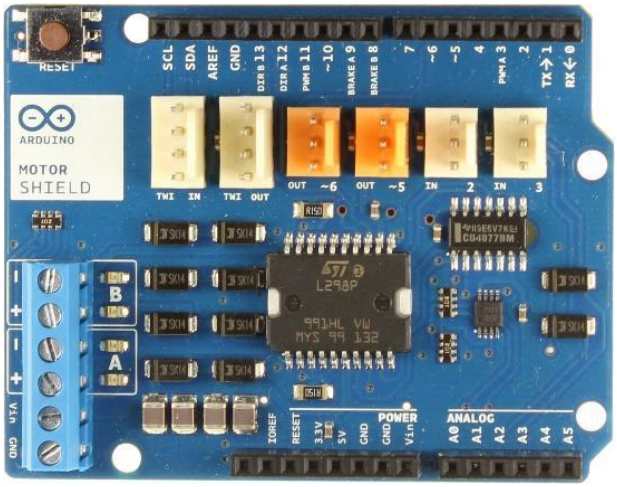
Hardware Assembly

Hardware of this project is a bit complicated, we separated it into two parts: motor and chassis. Here are the components we need to build this project:

- LinkIt 7697 dev board, 1
- breadboard, 1
- wires, several
- DC-motor extension shield, 1
- robot chassis (with 2 DC-motor and wheels)

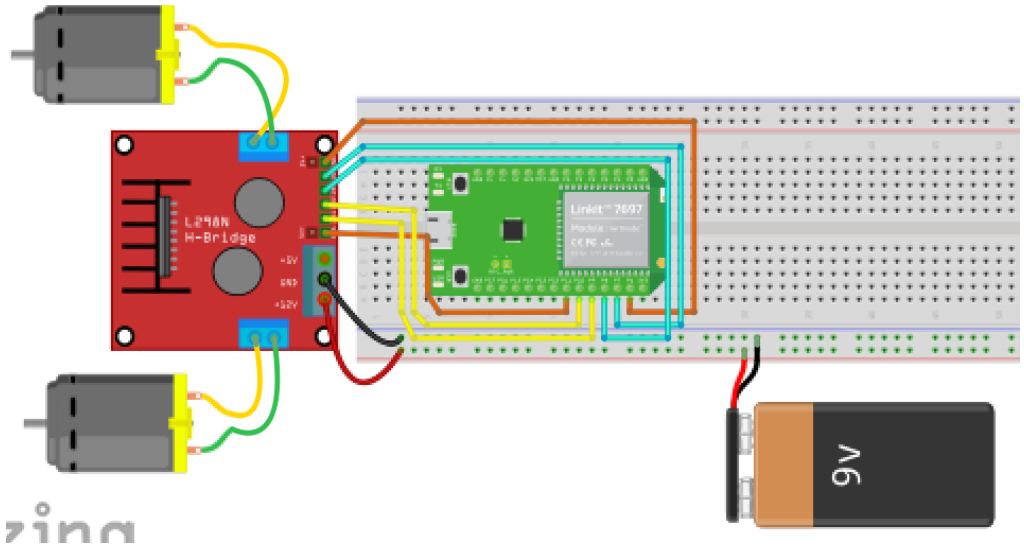
1. Motor

We recommend that you should use motor extension shield (like **Arduino motor shield**) rather than single H-bridge DC-motor control chip. Since motor shield has an integrated chip, you can save a lot of time with wiring. Additionally other hand, shield also has certain protection for your board to be damaged by surge. Any model of Arduino-compatible DC-motor shield should work fine with the LinkIt 7697.



Arduino motor shield

Your circuit should look like the picture below, and please check the table for more details.



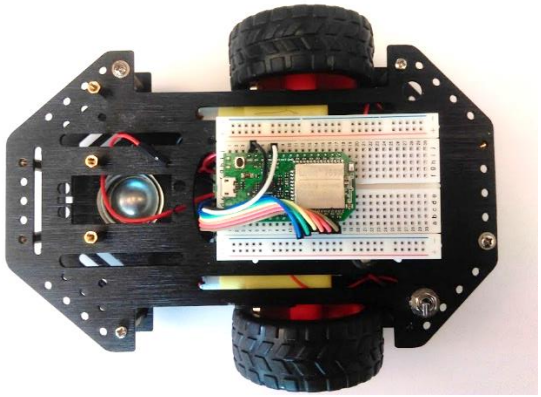
DC Motor shield	LinkIt 7697	9V or 12V battery
ENA	D11	
Motor#1+(IN1)	D10	
Motor#1-(IN2)	D9	
Motor#2+(IN3)	D8	
Motor#2-(IN4)	D7	
ENB	D6	
+12V	-	+
GND	GND	-

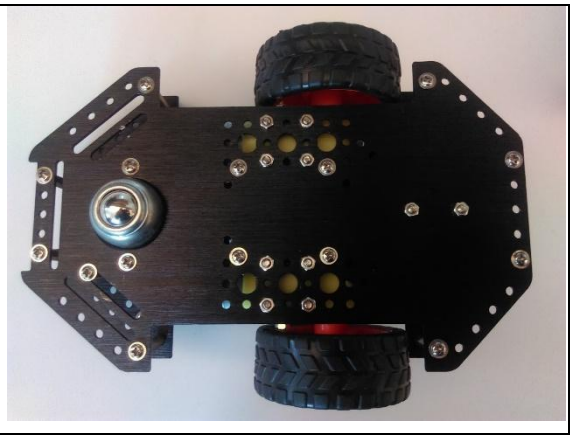
2. Robot chassis

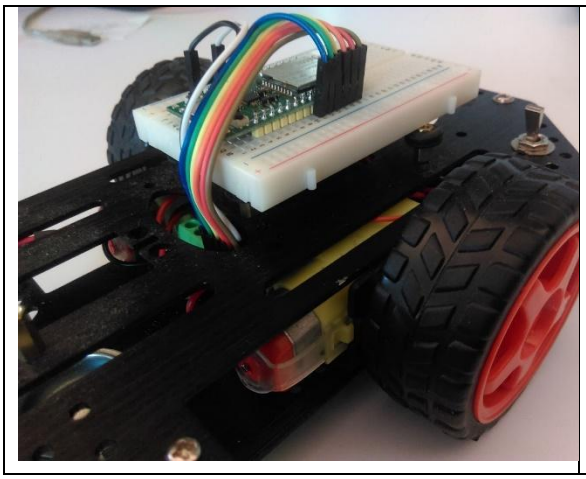
For the robot chassis, you can pick any kind of two-wheel chassis (two motors at each side with opposite directions). Here we list several of them:

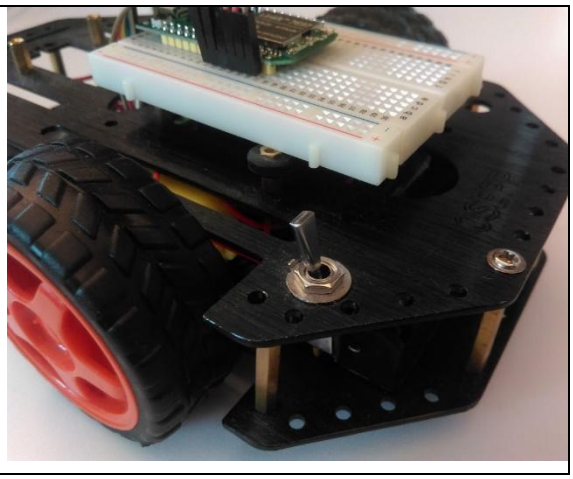
- https://www.adafruit.com/product/3216
- https://www.dfrobot.com/product-1144.html
- https://www.seeedstudio.com/AlphaBot%2C-Basic-robot-building-kit-for-Arduino-p-2843.html

Here are some photos of our sample robot:









App Inventor

We are going to make the robot move and stop by using buttons on your app. The main idea is to send out a number (integer) when you press a button, and the robot will do a corresponding action. Now let's take a look of the relationship between numbers and actions:

Number sent by App Inventor	Robot action
4	turn left
3	turn right
2	go forward
1	go backward
0	Stop

Note: the robot's behavior depends on which platform you choose. Factors like chassis length, wheel, and motor torque are all going to affect how your robot move. Try to play with different kinds of robot chassis and adjust the code, you will discover more!

Now log in to your **MIT App Inventor account** and create a new project.

Designer

1. The most used components in this project are buttons (to trigger actions) and labels (to show relevant messages).
2. We need to import the BLE extension from this URL:
    - http://iot.appinventor.mit.edu/assets/com.bbc.micro:bit.profile.aix
    - add the BLE extension by dragging it into Viewer.
3. Add a **TableArrangement** component, set its width to **Fill parent**, height to **200** pixels, Row to **2** and Column to **3**.
1. Add four **buttons** into previous **TableArrangement** component. Give each a width of 33% and a height of 100 pixels. And modify the Text to "⬆", "⬇", "⬅" and"➡", representing different moving direction of robot.

After some adjusting, your designer should look similar to this. It doesn't have to be exactly the same. Feel free to modify the component's background color, position and text size.

Blocks

Let's take a look of our blocks step by step:

1. **Variable for Bluetooth address**

    Please replace the addr variable value with what you get from Arduino's Serial Monitor, this is the Bluetooth address of LinkIt

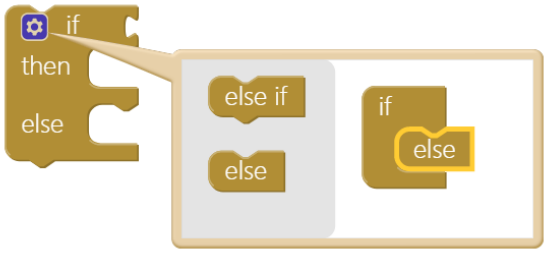
7697.

2. **Initialize and connect**

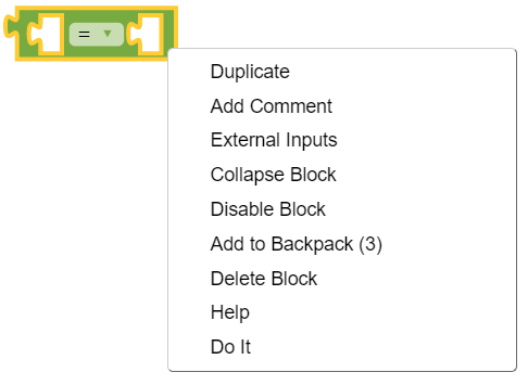
3. In **Screen1.Initialize** event, we ask **BluetoothLE** component to scan for BLE devices nearby (**BluetoothLE1.StartScanning**).

In **Button_connect.Click** event, we are going to connect or disconnect from Bluetooth device depending on the button text.

First, add an **if** condition, then click its blue gear to add an **else**.

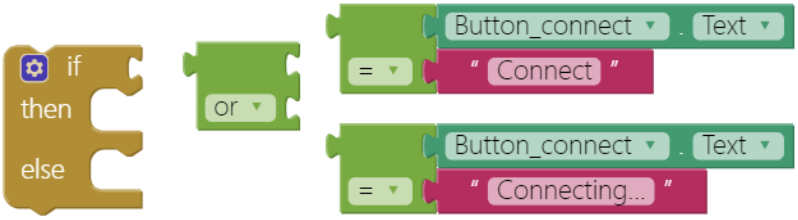


Add an **or** command from logic block, then right-click it and select "**External Inputs**". This will make it into more rectangular shape with input on the right hand side.

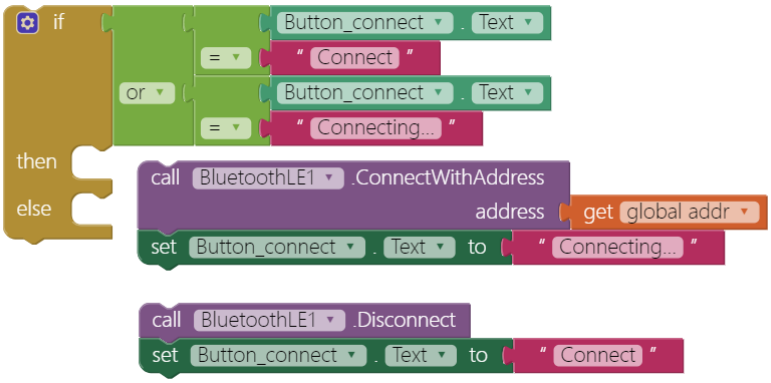


Now we want to check whether the **Button_connect.Text** status is "**Connect**" OR "**Connecting...**", this is how App

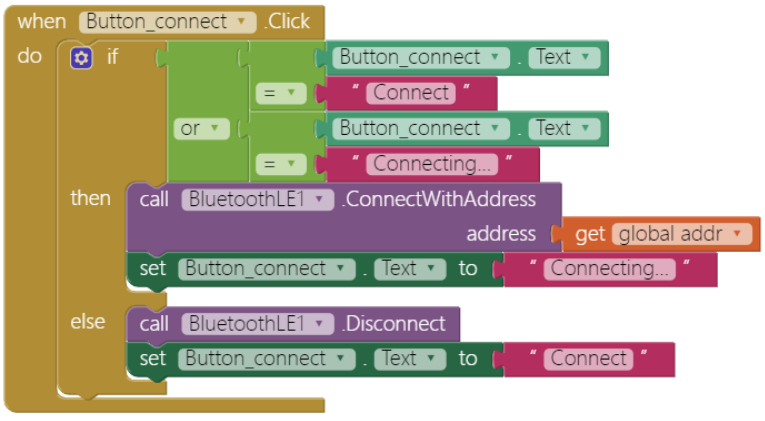
Inventor decide to connect or disconnect Bluetooth connection with LinkIt 7697. please combine these blocks.



Good! When the **Button_connect**'s text is "**Connect**" OR "**Connecting...**", we will connect to the specified Bluetooth device (**BluetoothLE1.ConnectwithAddress**), which is our LinkIt 7697. If not, then disconnect (**BluetoothLE1.Disconnect**) and show message on Button_connect.

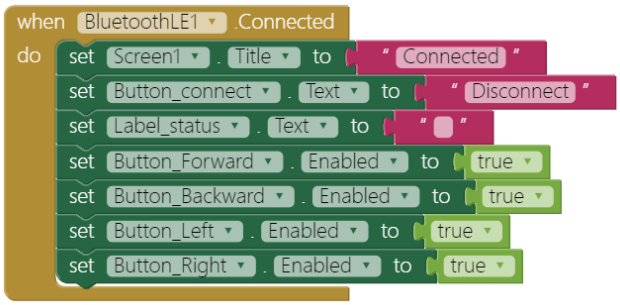


Put everything into **Button_connect.Click** event, and finish like this:



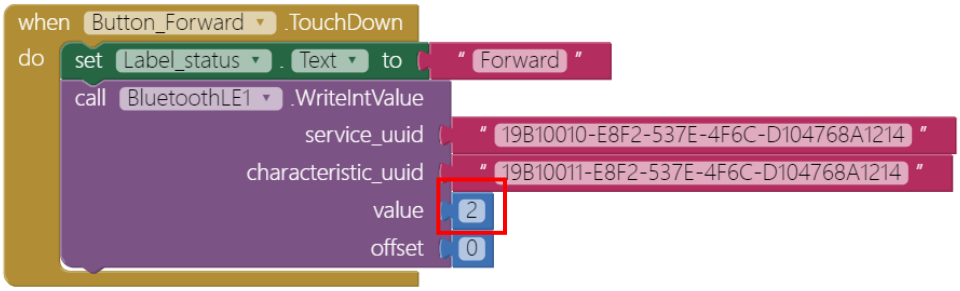
4. **BLE Connected**

   When connected successfully (**BluetoothLE.Connected** event), we show relevant messages on several components and enabled **Button_open** and **Button_close** to be clicked.



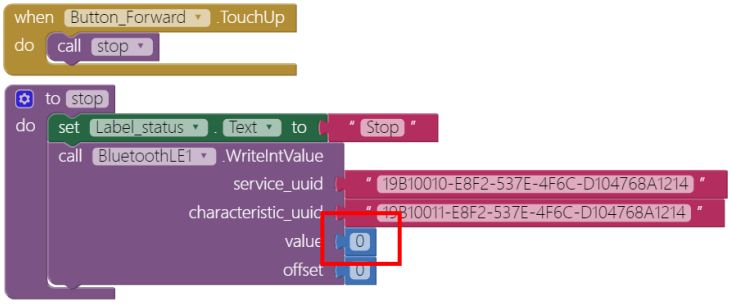
5. **Button touchdown to move the robot**

   When **Button_open** is clicked, we show the relevant message and send out a number **2** by the **BluetoothLE.WriteIntValue** method. Notice that the **service_uuid** and **characteristic_uuid** must be identical with what in Arduino sketch. LinkIt 7697 will put its digital pin #7 to HIGH voltage level, turning on the LED.



6. **Button touchup to stop the robot**

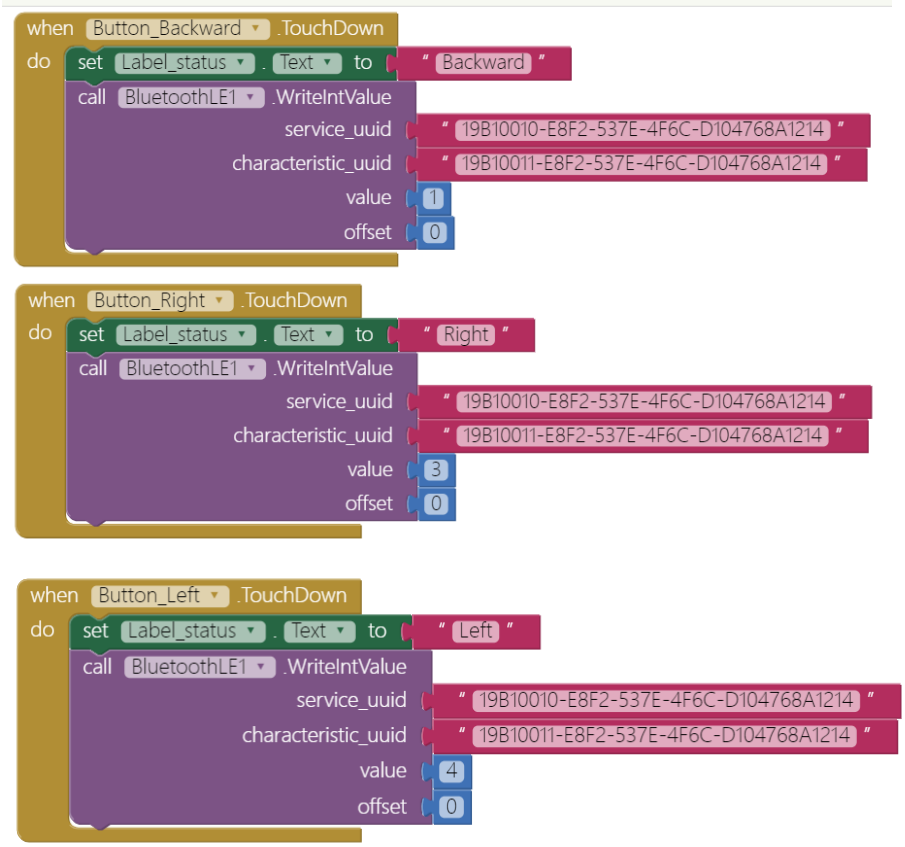
   We are doing almost exactly the same here except sending out number **0** instead of 1. LinkIt 7697 will.



7. **Other three buttons**

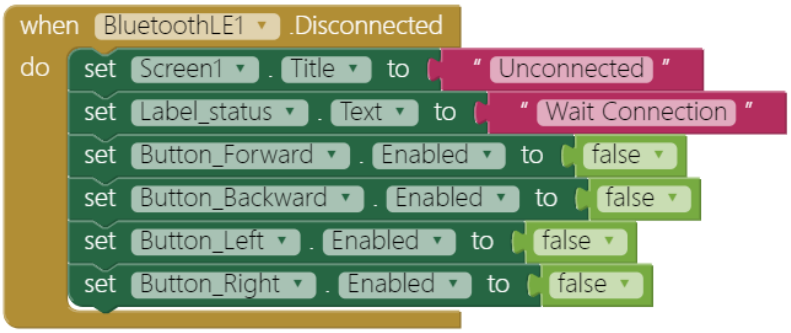
When click other three buttons, we use the same command but send out different numbers:

- Backward: 1
- Right: 3
- Left: 4



8. **Disconnect**

The Bluetooth connection will end when you click the **Button_connect** or pressed the **USR** button (D6) of LinkIt 7697. This will reset the app to its initial state to wait for next connect request.



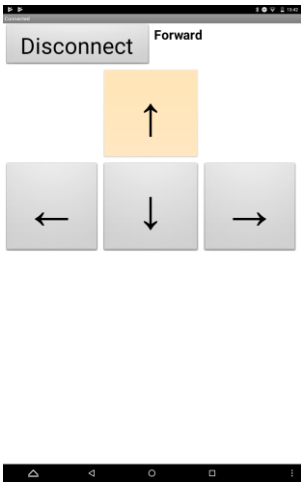
Arduino code

Please download from [here] and upload to your LinkIt 7697.

Press the "**Upload**" right-arrow button, this will compile and upload the Arduino sketch in Arduino IDE to your LinkIt 7697. Please make sure you see the "**done uploading**" message in the console.

Tips

Make sure your LinkIt 7697 is running correctly as a BLE peripheral. Open your app and click **Connect** button. Click each button to drive your robot moving around.



Brainstorming

1. Add a Speechrecognizer component and use voice command to

drive your robot like "**go forward**", "**go backward**", "**turn left**", "**turn right**" and "**stop**". (reference: **LinkIt 7697 LED control tutorial**)

2. Add two more buttons to trigger another LED on LinkIt 7697 (hint: more cases in Arduino sketch).
3. Try to find another robot chassis (tank is good!) and modify your robot.