ROS BLUETOOTH TUTORIAL: APPENDIX

A. Setting up RFCOMM channel
(This will work with almost any Smartphone including Apple iPhone)

RFCOMM profile is an emulation of serial connection via Bluetooth. It is the most common way to use Bluetooth for communication between two devices. One way is to establish a permanent serial port which will be seen in your system. You'll be able to work with it like with any other real serial ports in the system.

First, add your user ("ubuntu" by default) to the "dialout" group, so you'll have an access to virtual serial ports created for Bluetooth.

```
duckiebot:~$ sudo useradd -G dialout ubuntu
```

Next you need to register the Serial Port service. Do this after each reset.

```
duckiebot:~$ sudo sdptool add –channel=22 SP
```

You can pick up any channel number you like between 1 and 30, for this tutorial 22 is recommended.

[**IMPORTANT**] This command should give you an output with successful result. If no output was given, do this (this happens with Bluez 5, i.e. on Ubuntu 16.04):

Edit the "`/etc/systemd/system/dbus-org.bluez.service`" file.

```
duckiebot:~$ sudo nano /etc/systemd/system/dbus-org.bluez.service
```

Change this line:

```
ExecStart=/usr/lib/bluetooth/bluetoothd
```

To this:

```
ExecStart=/usr/lib/bluetooth/bluetoothd –-compat
```

Run this command as well:

```
duckiebot:~$ sudo chmod 777 /var/run/sdp
```

Reboot your Duckiebot:

```
duckiebot:~$ sudo reboot now
```

Start Bluetooth server on selected channel.

```
duckiebot:~$ sudo rfcomm listen /dev/rfcomm0 22
```

Now, pick any “Bluetooth Terminal” application you like and try connect to your server form Smartphone. You can download “Bluetooth Terminal” from the Market, there are several apps with this name, pick any, they all do their job pretty well. Don’t stop your server, if you need to enter another Linux command use another terminal here.

https://play.google.com/store/apps/details?id=ptah.apps.bluetoothterminal&hl=en
Recommended “Bluetooth Terminal” app for Android.

Q: How to see data you’ve sent from Smartphone using only Linux commands?

In a new terminal:

```
duckiebot:~$ cat /dev/rfcomm0
```

Now, send anything using Bluetooth Terminal app and see the output in your robot’s terminal.

Q: How to send data to smartphone using only Linux commands?

```
duckiebot:~$ echo “Hi there, dude!” > /dev/rfcomm0
```

See the output on your Bluetooth Terminal app.

B. Using bluetooth_bridge to communicate with Bluetooth in ROS.

This part will tell you how to use "bluetooth_bridge" created for this demo separately, in case you want to use it in your final project.

Another option is to use Python directly to communicate with Bluetooth. For this exercise, download and install "bluetooth_bridge" ROS node. You can use this code as an example of Bluetooth communication in Python. This node will set up Bluetooth server on your robot, wait for incoming connections, and then send messages between ROS and Bluetooth device.

```
duckiebot:~$ cd your_duckietown_working_folder
duckiebot:~$ git clone https://github.com/YuryXW/bluetooth_bridge.git
duckiebot:~$ cd ~/duckietown/catkin_ws
duckiebot:~$ catkin_make
```

This node has two important topics:

"/bluetooth/send", type "std_msgs.String" - send a "String" message to this topic to send it via bluetooth.

"bluetooth/received", type "std_msgs.String" - any message received via Bluetooth will be published through this topic.

[IMPORTANT]: do not forget to set up the Serial Port service on your Duckiebot if you've rebooted your robot or did not set it before (sudo rfcomm listen /dev/rfcomm0 22).

Launch bluetooth_bridge:

```
duckiebot:~$ roslaunch bluetooth_bridge bluetooth_bridge.launch
rfcomm_channel:=22
```

Now check if messages are being published in /bluetooth/received topic:

```
duckiebot:~$ rostopic echo /bluetooth/received
```

Send something via Bluetooth Terminal from your smartphone.

Next, send a string to /bluetooth/send topic and see it appear in your Bluetooth Terminal:

```
duckiebot:~$ rostopic pub /bluetooth/send std_msgs/String "Hello" --once
```

This is it, you've set up the bridge between ROS and your Bluetooth device (Smartphone, in this case).
