



Faculty of Engineering and Applied Science
Department of Electrical, Computer and Software Engineering

**Design and Analysis of IoT Software
Assignment 2
Project Group 5
Tuesday, November 12, 2019**

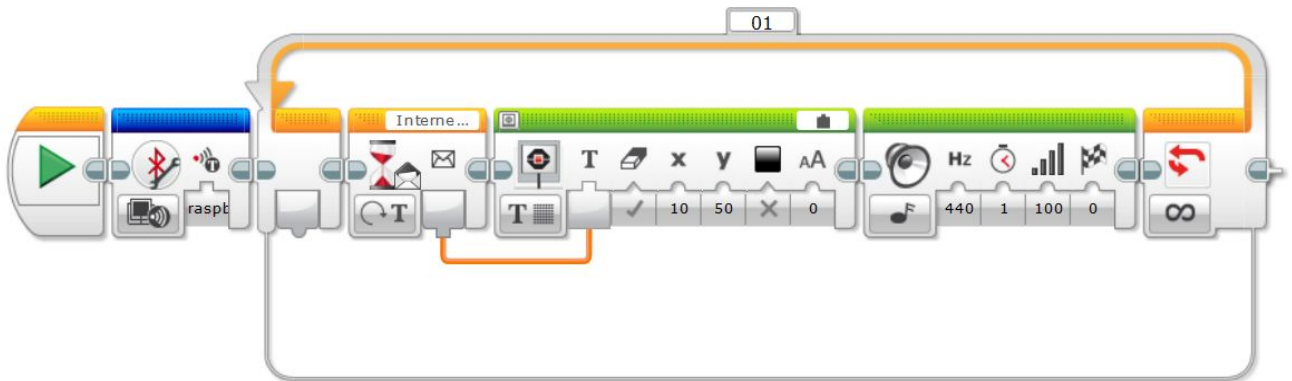
Student Name	Student Number
Sai Pedamallu	100598060
Temiloluwa Femi-gege	100584697

Instructor: Dr. R. Liscano

Task 1: Transmitting messages from the Pi to the EV3 and vice versa

a. Transmit Messages from the Pi to the EV3

The following program waits for a Bluetooth message and prints it on the LCD screen and beeps.

A screenshot of the Thonny IDE running on a Raspberry Pi. The main window shows a Python script named 'SendtoEV3.py'. The script imports 'serial', 'time', and 'EV3BT'. It creates a serial object 'EV3' at '/dev/rfcomm0', encodes a message 'Internet of Things, IoT A2 Message Sent', prints it, writes it to the serial port, sleeps for 1 second, and closes the port. The shell window at the bottom shows the command '%Run SendtoEV3.py' and the resulting hex output: '2e 00 01 00 81 9e 13 49 6e 74 65 72 6e 65 74 20 6f 66 20 54 68 69 6e 67 73 00 14 00 49 6f 54 20 41 32 20 4d 65 73 73 61 67 65 20 53 65 6e 74 00'. The output is displayed in a hex dump format. The top bar of the IDE shows the file path '/home/pi/SendtoEV3.py' and the time '12:11'. The right sidebar shows the 'EV3BT.zip' file.

After the program is run on the EV3, the script is run to communicate with the serial port (**rfcomm0**) and looks for the message “Internet of Things” and sends the message. The above Python script sends a message as hex numbers to the Pi (**49 6f 54 20 41 32 20 4d 65 73 73 61 67 65 20 53 65 6e 74 00**) which converts to “**IoT A2 Message Sent**”. The output can be seen on the EV3 below.



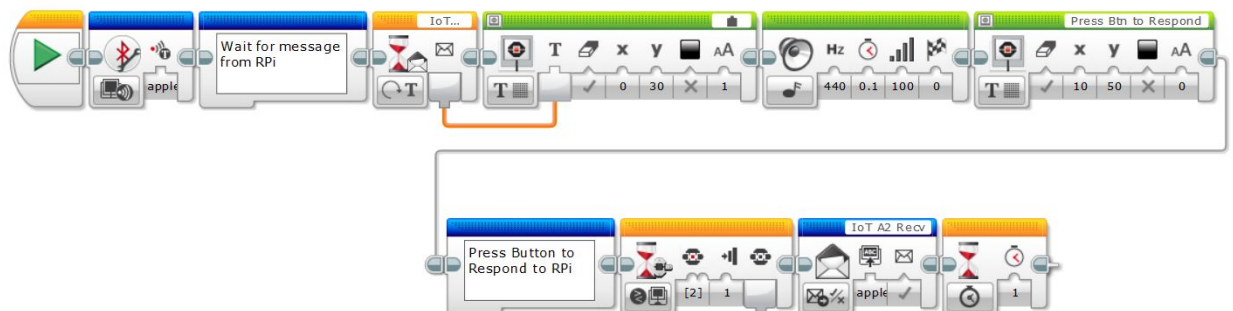
running the program



message sent from Pi to EV3

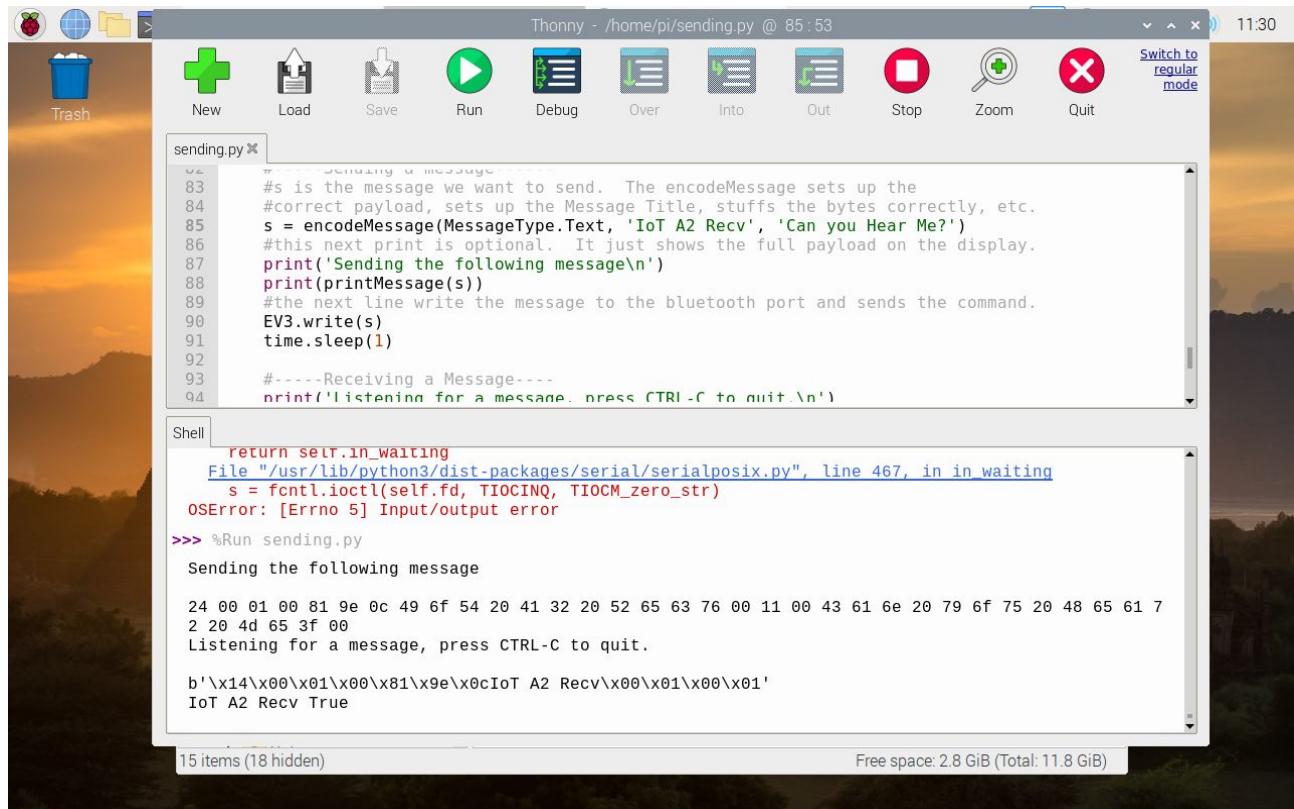
b. Transmit Messages from the EV3 to the Pi

The following program initiates a connection with the Raspberry Pi and once the button is pressed on the EV3, it sends a message back to the Pi as: "IoT A2 Recv True"



Press Btn to Respond is the message that is sent by the Pi to the EV3. **IoT A2 Recv** is the message that is received on the Pi when the user presses the button on the EV3.



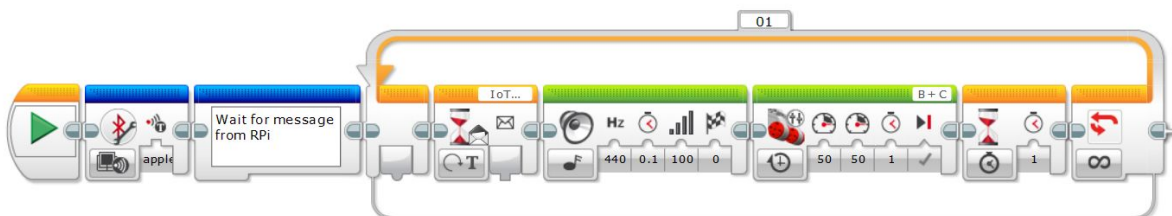


In the above image, the Python script sends a message in hex (**43 61 6e 20 79 6f 75 20 48 65 61 72 29 4d 65 3f 00**) to the EV3 to prompt the user to press a button and then once the button is pressed, the EV3 sends a message to the Pi as “**IoT A2 Recv True**” (last line in output).

Task 2: <https://youtu.be/ISfQY6RALV0>

A demonstration of the following program can be found in the YouTube link above. The EV3 program below is a simple illustration of how it works. Firstly, the EV3 initiates a connection with the Raspberry Pi and then waits for a message (“IoT A2 Recv”). It then runs the blocks to make a sound and then moves the motors for **1 second** with **50** power on the left and right motors.

As you can see in the video (0.28s - 0.30s), when the light intensity is 0.0, the block does not run. It is an infinite loop, so everytime the light intensity on the TI SensorTag is greater than 0.0, a message is sent to the EV3 to run the rest of the blocks to run the motors.



References

[1]M. Shyte, "Raspberry PI and the Lego EV3 connected by bluetooth | Geek Droppings", *Geekdroppings.com*, 2019. [Online]. Available: <http://www.geekdroppings.com/2018/01/21/raspberry-pi-and-the-lego-ev3-connected-by-bluetooth/>. [Accessed: 12- Nov- 2019].

[2]"Sending Bluetooth Messages from Raspberry Pi to LEGO EV3 (stock firmware) | A. Brown Design", *Abrowndesign.com*, 2019. [Online]. Available: <https://www.abrowndesign.com/2018/10/25/sending-bluetooth-messages-from-raspberry-pi-to-lego-ev3-stock-firmware/>. [Accessed: 12- Nov- 2019].