Lab 5 Report
Setting Up Bluetooth For Tank Control

Group Members: Laila Abudahi, Erin Long, Vishesh Sood

Hardware Setup

Table 1 below shows how we wired the Beaglebone pins using the H Bridge and the BlueSMiRF Bluetooth module, along with the sensors and motors.

GPIO Pin	Usage
P8.26	BIN2 (H Bridge)
P8.44	BIN1 (H Bridge)
P8.46	PWMA (H Bridge)
P8.65	PWMB (H Bridge)
P8.66	STDBY (H Bridge)
P8.67	AIN1 (H Bridge)
P8.68	AIN2 (H Bridge)
P8.13	PWMA (H Bridge)
P8.19	PWMB (H Bridge)
ADC0	Front Sensor
ADC1	Left Sensor
ADC2	Right Sensor
ADC3	Back Sensor
UART1_TXD (24)	Bluetooth Receiver
UART1_RXD (26)	Bluetooth Transmitter

Table 1: A table to show our pin mapping for the Tank

We did not use any additional hardware components to support the tank.

Software Setup

There was a lot of software involved in this lab. Firstly, we custom built an Android application using Android Studio and Java. We then built and compiled the program onto an Android phone to control the tank. The app took us quite a few hours, as we had to understand how Bluetooth transmission works on Android. The user is able to use the application to control the tank wirelessly using the controls on the application.

The most important code we wrote in this lab was the serial interrupt handler for reading values from the Bluetooth (written in /dev/ttyO1) without using Minicom. Being the first group to solve this issue, a lot of groups used our code to move on with the lab, and it is pretty impressive. The code sets several flags for the termios struct and also the signal handler flags. The code can be seen in bluetooth_tank.c.