Cornell University
School of Electrical and Computer Engineering

Let Music Ease Your Bike Ride!

Siyu Chen (sc2564@cornell.edu) Advisor: Prof. Bruce R. Land

Introduction

In places like Ithaca where there are a lot of steer slopes, riding a bike can be really challenging. People always get exhausted after a long ride or when riding up a steer slope, and would want some inspiring and encouraging music to help them regenerate and get back on track. Basically, the main purpose of my design is to let music sooth the pain from exhausting bike rides. The major function of my design is making the Android smart phone switching to different music playlists adapting to the bike speed change.

The design system consists of two parts: the hardware embedded on the bike which handles the speed measurement, and the customized Android app which can switch to different playlists based on the change of speed pattern. The hardware communicates with the Android app via Bluetooth, whose wireless feather maximizes the separation and independency of the hardware and the smart phone.

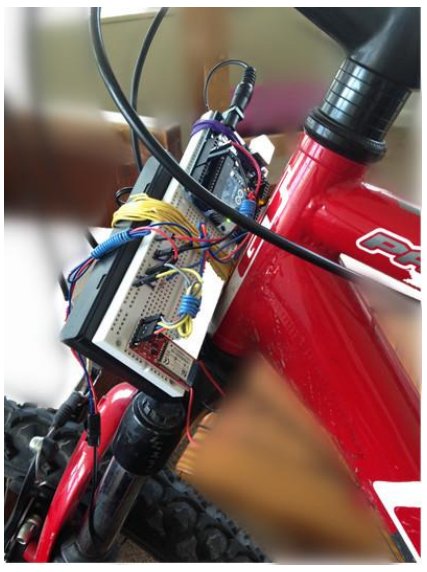
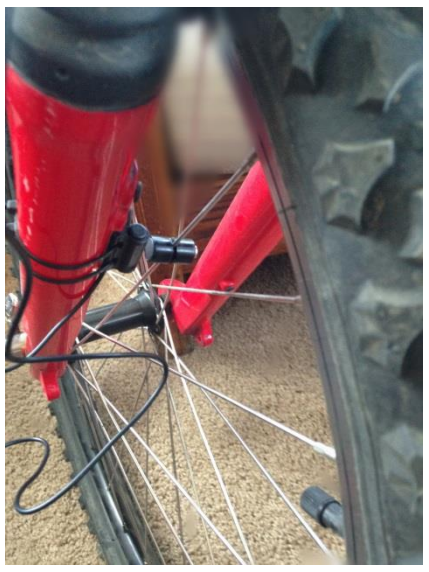



Figure 1 – Hardware Mounted on the Bike

Hardware Design

The bike embedded hardware is responsible for counting the time interval between two consecutive revolutions. It is also designed to drive the Bluetooth module which transfers the raw bytes of that interval to the Android smart phone.

For the bike embedded hardware design, Arduino UNO board was chosen as the MCU. A pair of magnet & magnetic sensor is used for signaling each individual revolution.

As shown in Figure 1, the magnet is mounted on the spoke of the wheel, and the magnetic sensor is fixed on the hub. For each revolution, the sensor meets the magnet for one time, generating a voltage falling edge. By feeding it back to external interrupt of MCU, the time interval between two consecutive revolutions can be calculated.

The Bluetooth chip being used is RN-42, which has a communication radius of 10 meters, being perfect for short range, battery powered applications. RN-42 uses only 26 μA in sleep mode while still being discoverable and connectable.

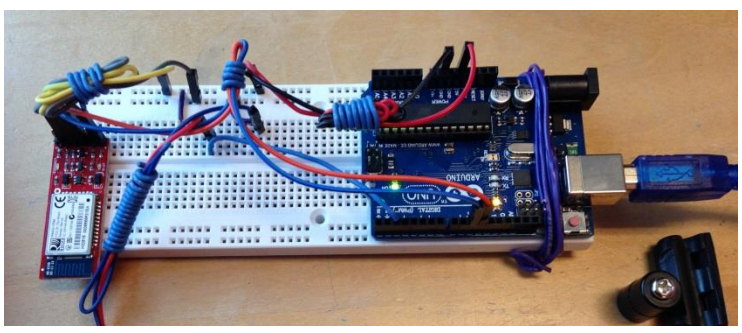
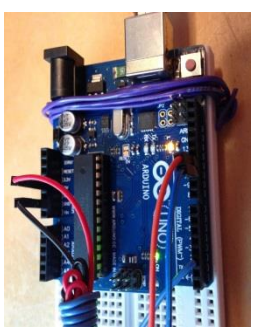


Figure 2 – Hardware Circuits

Magnetic Sensor

Figure 3 – Bluetooth Module



Arduino Board

Android Software Design

Hysteresis dynamic average speed control is adopted in the design. It is called hysteresis control since the switching behavior is just like a hysteresis loop: slow functional playlist is played when real time speed is lower than previous dynamic average speed deducting some threshold (one of the sensitivity indicator, which can be set by user), and only after real time speed exceeding previous dynamic average speed plus the aforementioned threshold would the music player change back to play normal functional playlists. When real time speed exceeding some preset value, fast functional playlist will be played mainly as a warning of safety.

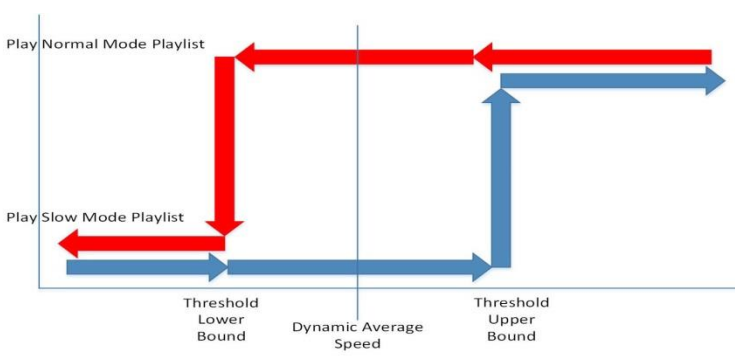


Figure 4 – Hysteresis Average Speed Control

Figure 5 – Main UI

Control Parameter