

# Foot-strike Analyzer and Trainer

Zachary Hand

Faculty Advisor: Dr. Barry Grossman, Dept of Electrical Engineering, Florida Institute of Technology

*A fitness accessory for runners*

## Goal

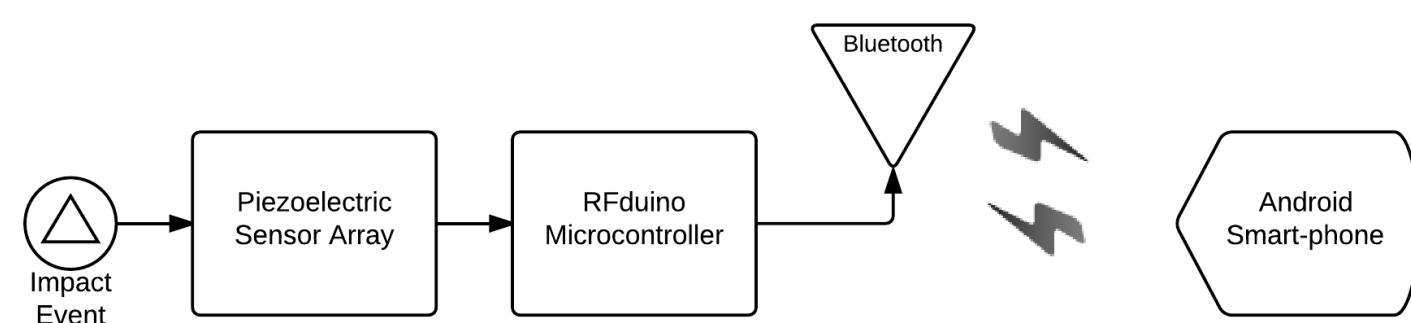
The goal of this project is to implement a portable system that is capable of monitoring how a runner's foot interacts with the ground. It provides real-time feedback allowing the user to modify their running form to fit their fitness goals.

## Project Overview

The project consists of three subsystems:

- Piezoelectric Sensor Array
- Data acquisition/processing/TX
- RX and User Interface (Smartphone Application)

The sensor array is embedded in a running shoe, and the conditioned output connects to a Bluetooth equipped microcontroller. When the shoe makes contact with the ground, the impact is detected and localized, and this information is transmitted via Bluetooth to an Android smartphone.



## Sensor Array

- Consists of two piezoelectric sensing elements
- Discriminates between heel-strikes and forefoot-strikes



## Data Acquisition

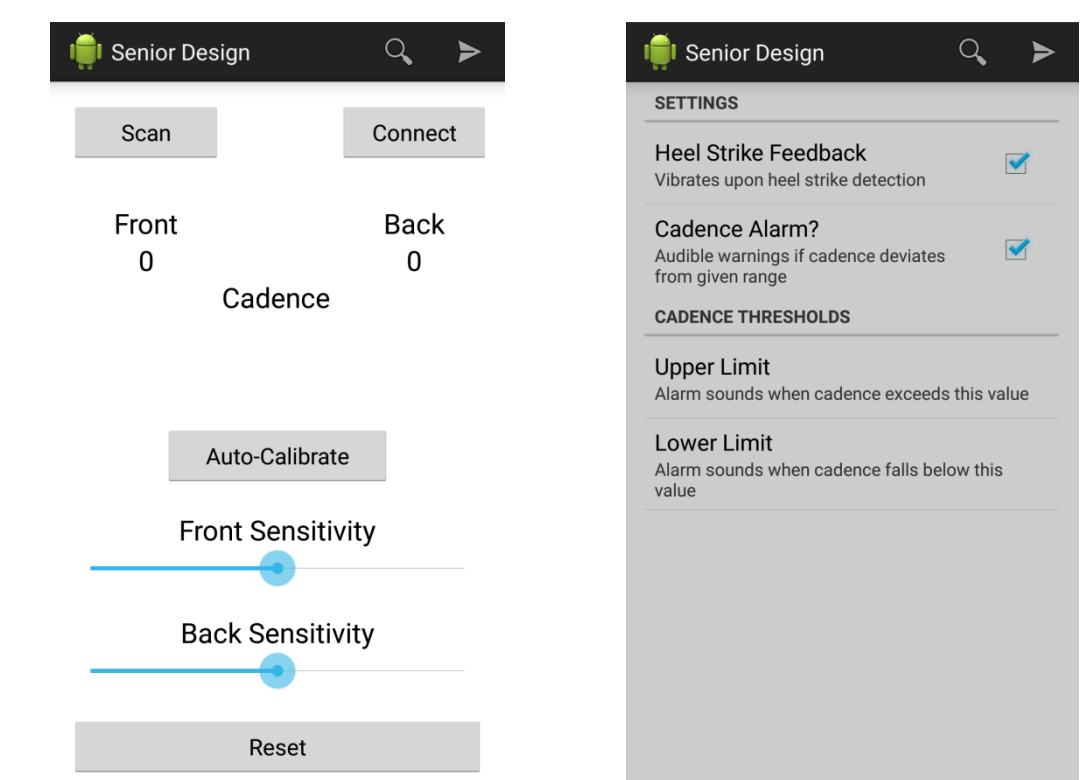
- RFduino microcontroller transmits sensor data and receives commands via onboard Bluetooth LE
- Auto calibration code sample:

```
//Auto Calibrate
if (cal == 1)
{
  delay(200);
  //Reads analog input port
  sensorReading = analogRead(Sensor1);
  //Initializes values
  threshHi1 = sensorReading;
  threshLo1 = sensorReading;
  for ( int i = 0; i < 200; i++)
  //Takes 200 samples from Sensor1
  {
    sensorReading = analogRead(Sensor1);
    //Finds high value
    if (sensorReading > threshHi1)
    threshHi1 = sensorReading;
    //Finds low value
    else if (sensorReading < threshLo1)
    threshLo1 = sensorReading;
    delay(10); //Wait 10ms between readings
  }
  //Padding added to reduce false
  positives
  threshHi1 = threshHi1 + padding;
  threshLo1 = threshLo1 - padding;
}
```

## Android Application

- Displays and records foot-strike and cadence (foot-strikes per minute) data
- Wireless Control:
  - Start/Stop data collection
  - Auto or Manual sensor calibrate
- Optional user feedback:
  - Cadence: Audible alarm if value deviates from user defined range
  - Heel-strikes: Vibrational feedback
- Sample of cadence alarm code:

```
if (cadence_enable)//Option selected
{
  if (cadence_upper != 0 && cadenceAvg > cadence_upper && cadenceViolation > 4) {
    //Checks if an upper limit has been entered, the limit has been exceeded
    //and if out of limit for at least 5 events
    cadenceViolation = 0; //Reset counter
    //Load sound file
    mp = MediaPlayer.create(this.getBaseContext(), R.raw.soundhi);
    mp.setOnCompletionListener(new MediaPlayer.OnCompletionListener()
    { //When sound file is done playing, do this:
      @Override
      public void onCompletion(MediaPlayer mp)
      { //Self destruct" after playing sound
        mp.reset();
        mp.release();
        mp = null; } });
  }
  //Play Sound
  mp.start();
}
```



**NORTHROP GRUMMAN**



Engineering & Science  
Student Design Showcase  
at Florida Institute of Technology

