Lab1 Sensor Apps

Task1
TI SensorTag with Android sensor app

TI sensor tag is bluetooth enabled sensor array to quickly prototype applications without need of having to deal with hardware design. Any bluetooth enable devices such as a smartphone can be programmed to retrieve data from the TI Sensor Tag and Make sense of data.

TI sensor tag has six different MEMS sensors are Infrared Temperature, Humidity, Pressure, Accelerometer, Gyroscope, Magnetometer. and also Two general purpose button inputs and is powered by a Single Coin-Cell.

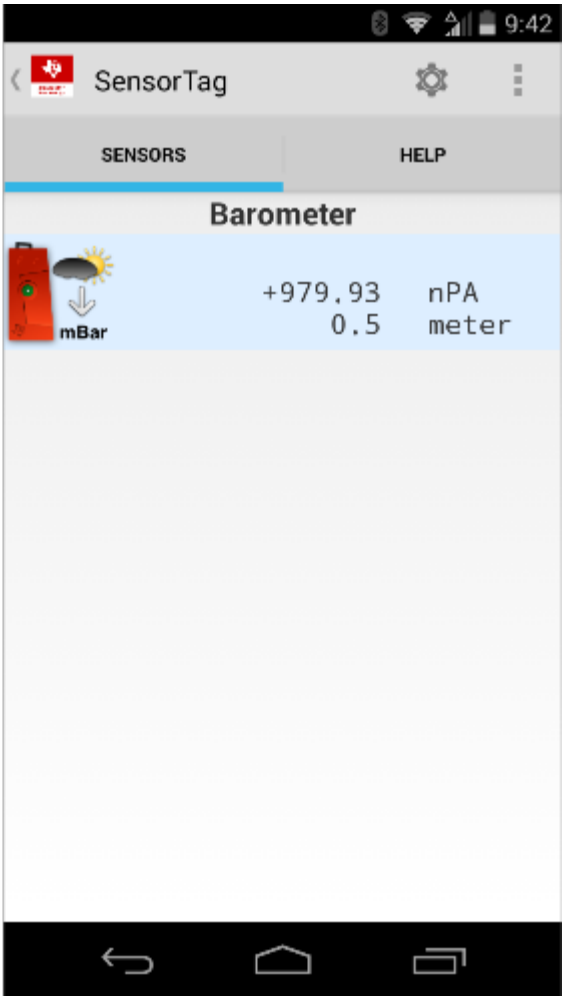
We paired the TI Sensor Tag to an **Android** Smartphone and with the help of the **SensorTag** App we're able to visualize (raw) data.

Android App Links

https://play.google.com/store/apps/details?id=sample.ble.sensortag
(This is not an official application for TI SensorTag)

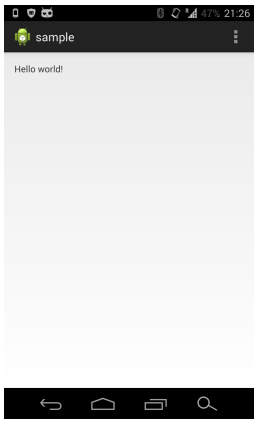
https://play.google.com/store/apps/details?id=com.ti.ble.sensortag
***Official** from TI and better App*

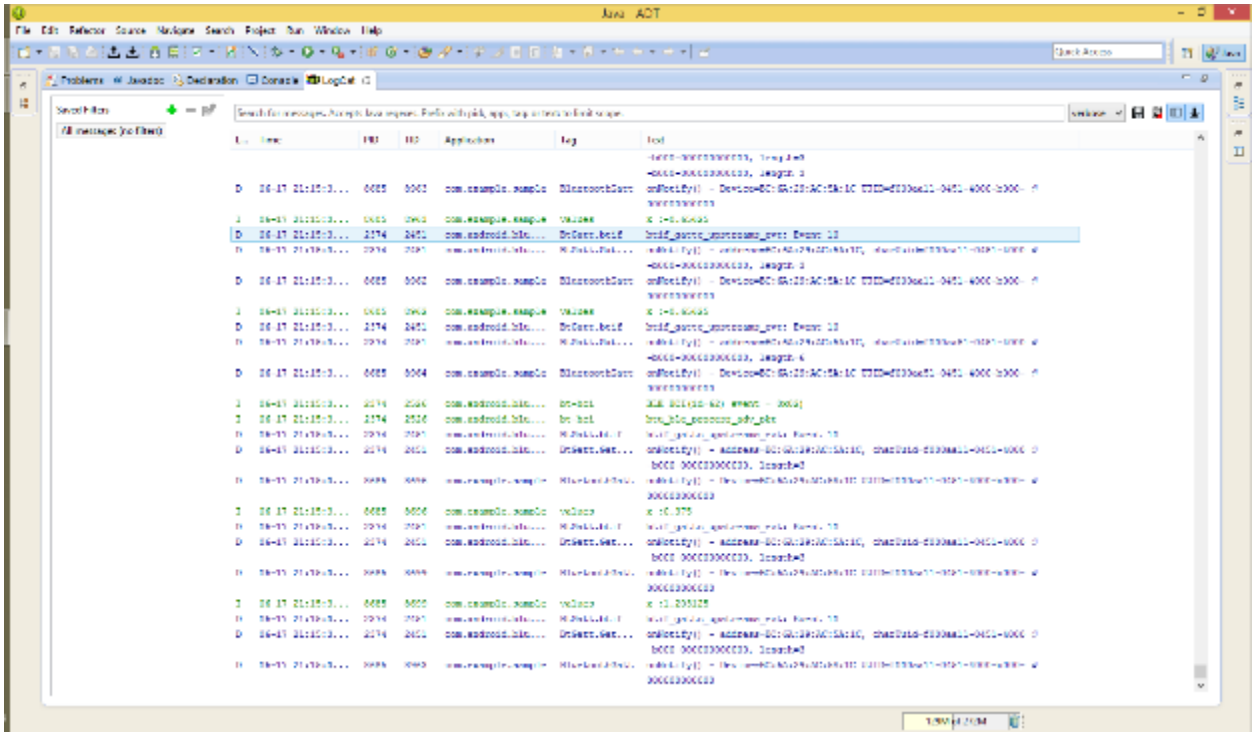
The following pictures show grabbing **Pressure** sensor data from SensorTag by both the Apps.



Sensor Tag App

This app connects the Android phone to SensorTag via Bluetooth and takes the x-axis accelerometer data and logs to PC with **LogCat** with **DDMS** (Dalvik Debug Monitor Server) service.




Java - ADT
File Edit Refactor Source Navigate Search Project Run Window Help
Problems Javadoc Declaration Console LogCat
Saved Filters
All messages (no filters)

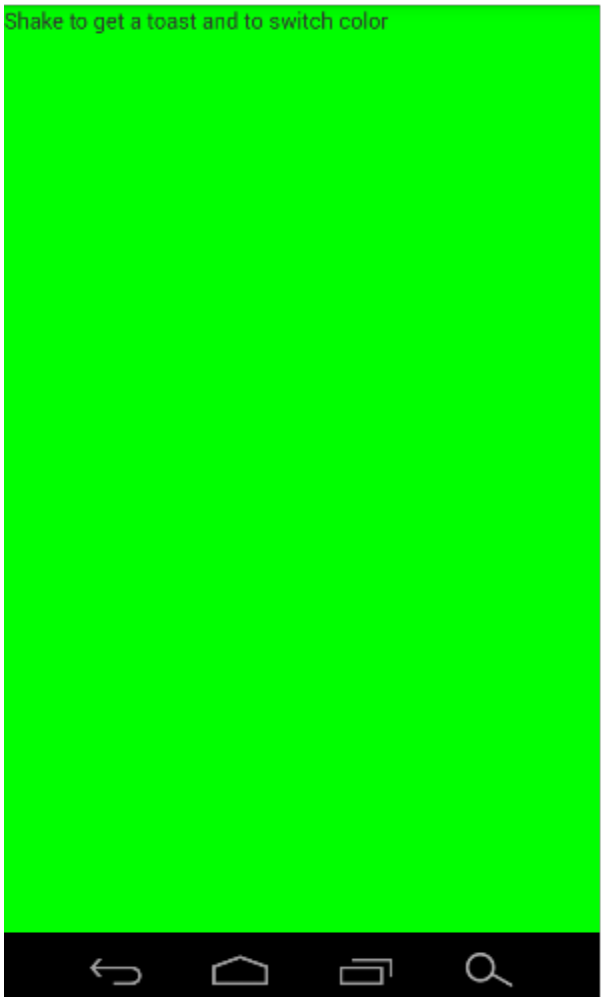
Task2

Mobile sensor with Android sensor app

Todays smartphones themselves are equipped with many sensors to get more contextual information and better serve the user.

For say an Nexus 4 phone has Accelerometer, GPS, Compass, Proximity sensor, Gyroscope, Pressure, Halleffect sensor.

The App changes the screen color from Green to Red once it detects Change in Acceleration.

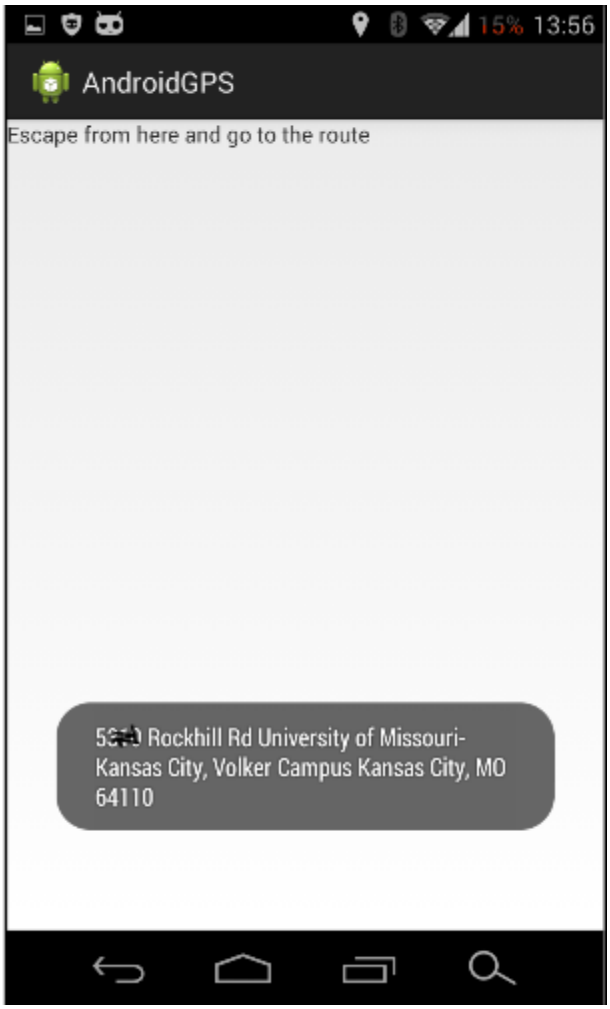


Task3

GPS feature with Android smart phone

This Android App requests the Location Service on Android Phone get Locations **Latitude** and **Longitude** from GPS sensor.

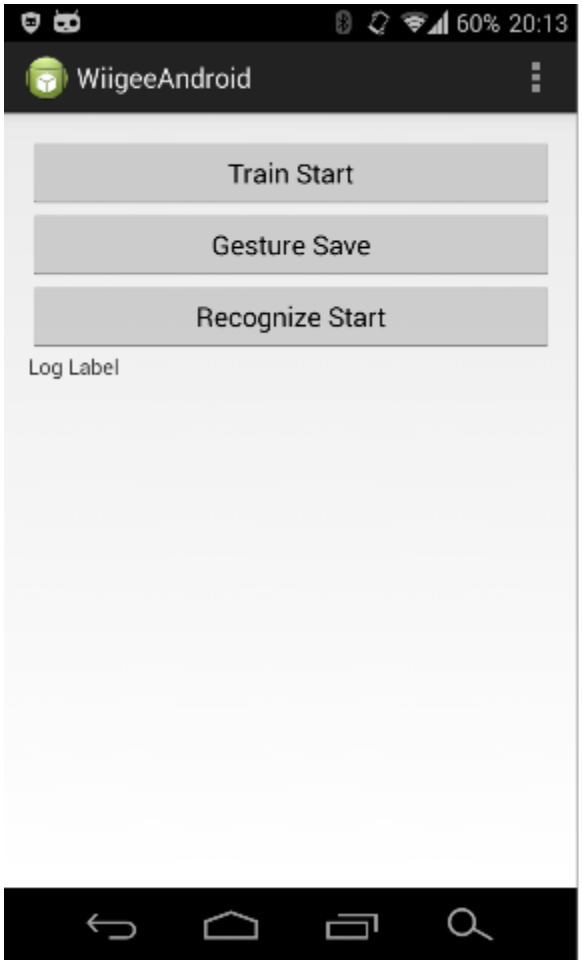
The App then reverse Geo-codes Lat and Long to Address from Google Maps API.

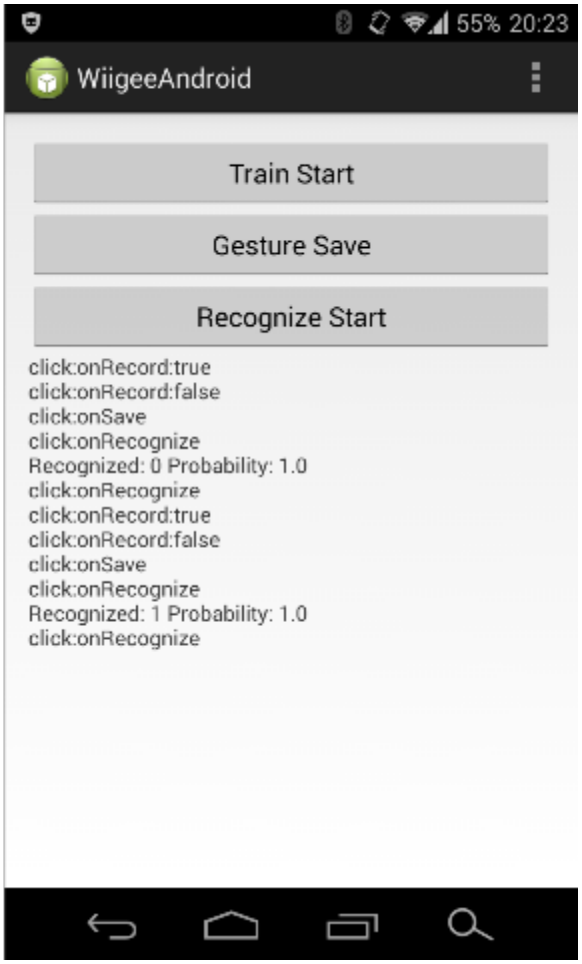


Task4

Wiigee app with Android smartphone

This does record the movement patterns and saves them, and has ability to detect the saved patterns if again occurs along with Probability factor.





Task5

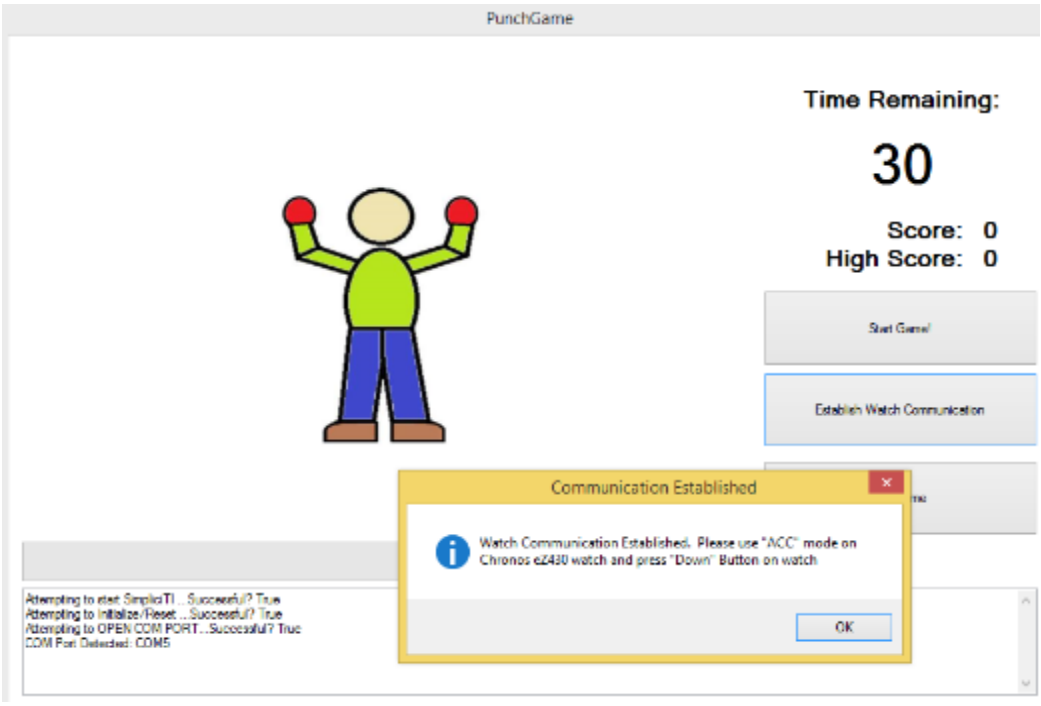
TI Chronos watch with Java app

TI eZ430 is wearable wireless development system, the watch has an Accelerometer, Pressure sensor, temperature and battery Voltage monitor. The display is an 96 segment LCD display.

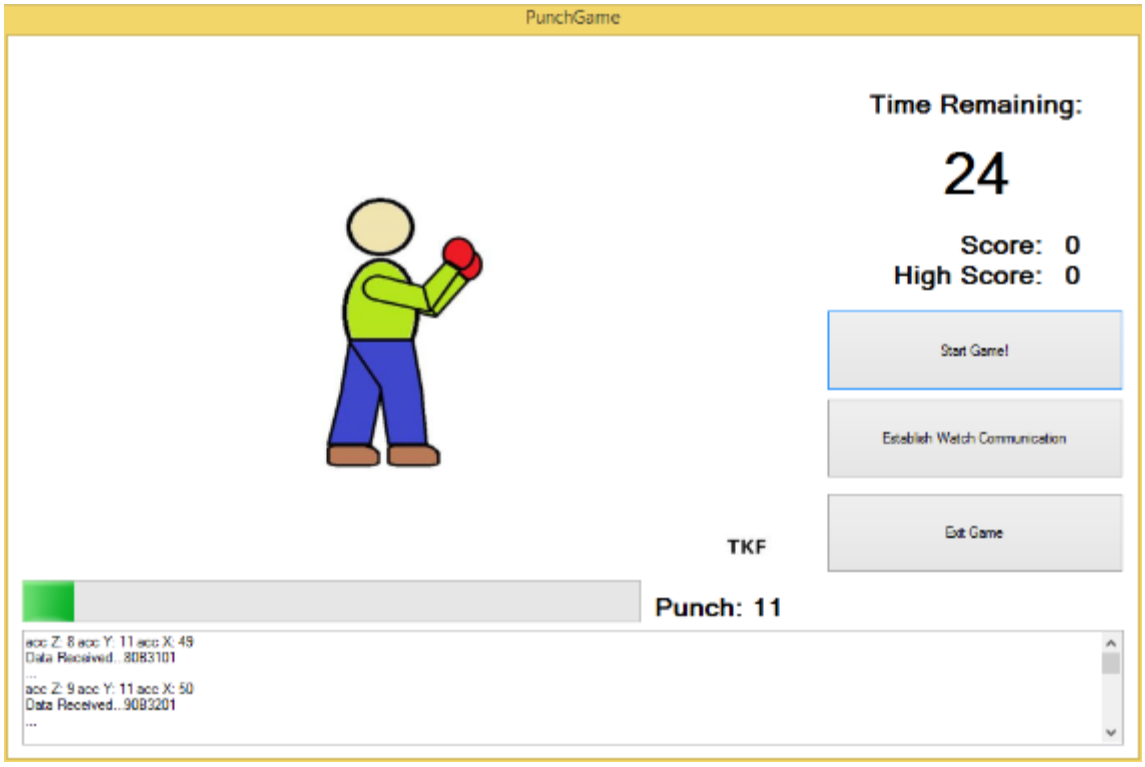
This development system would have been great if it has got Bluetooth wireless connectivity instead of RF and a Dot-Matrix display instead of segment display.

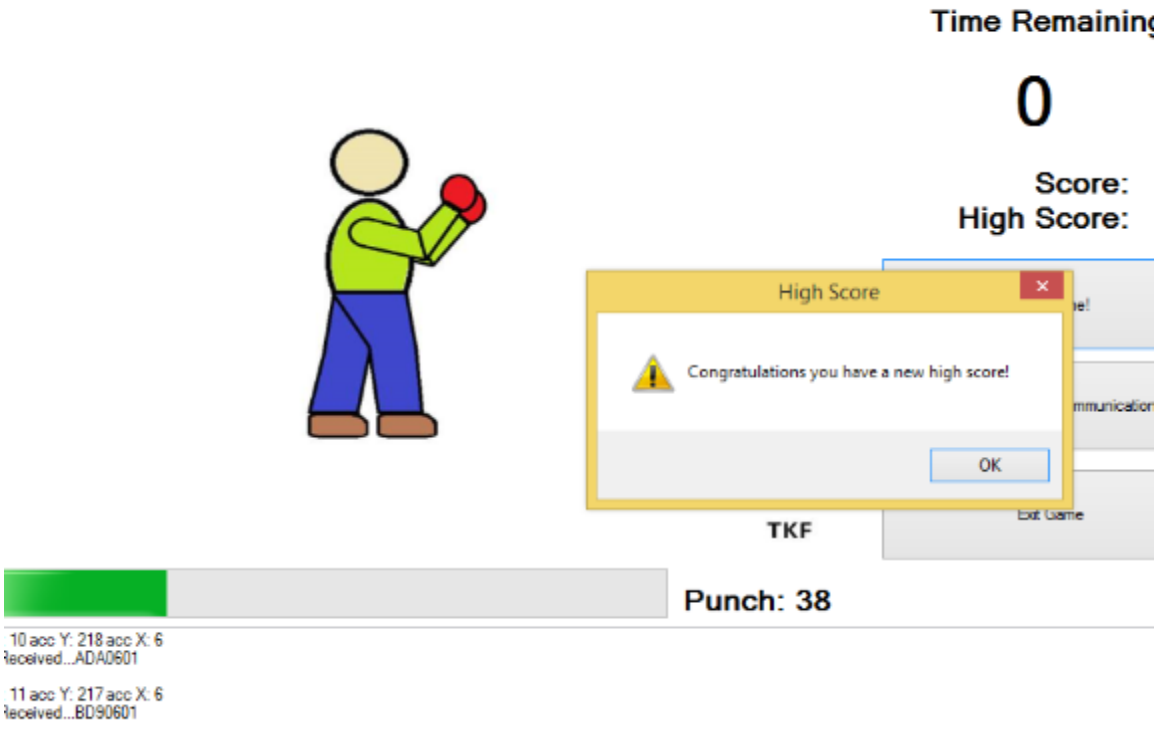
The Dev kit has two components a **Watch** which **transmits** data wirelessly and another a **USB Receiver** which goes into a PC.

The task is to interface TI Chronos watch with Windows Desktop app called **PunchMeter** Where the rate of change in acceleration is tagged as Users force of punch with a score.



Desktop App capturing Accelerometer Data and Scores the Force applied.


PunchGame
Time Remaining:
24
Score: 0
High Score: 0
Start Game!
Establish Watch Communication
Exit Game
TKF
Punch: 11
acc Z: 8 acc Y: 11 acc X: 49
Data Received...8083101
acc Z: 9 acc Y: 11 acc X: 50
Data Received...9083201


Time Remaining
0
Score:
High Score:
High Score
Congratulations you have a new high score!
OK
TKF
Exit Game
Punch: 38
10 acc Y: 218 acc X: 6
Received...ADA0601
11 acc Y: 217 acc X: 6
Received...BD90601