Android MacrotellectLink Development Guide

Introduction

This guide will teach you how to use MacrotellectLink SDK to write Android applications that can acquire brainwave data from Macrotellect 's Hardware（BrainLink Pro &BrainLink Lite）. This will enable yourAndroid apps to receive and use brainwave data such as BrainWave and Gravity acquired via Bluetooth, Macrotellect 's Hardware and File source are encapsulated as MacrotellectLink SDK forAndroid supports upgrading Hardware

Function: Receive brainwave data.

Supported Device：

- Bluetooth4.0 BLE
  - BrainLink_Pro
- Bluetooth3.0
  - BrainLink_Lite
  - Mind Link

Supported Android Version： - Android 4.3 +

Your First Project: MacrotellectLinkDemo（Android studio）

1. Copy the SDK to the project's libs folder and add dependencies in build.gradle.

```
         dependencies{
              ..
              implementation files('libs/MacrotellectLink_V1.4.3.jar')
       }
```

2. Add permissions in AndroidMainifest.xml ```

```
 **3.Receive brainwave data from Macrotellect SDK.**
```java

  //check location permissions
  if((checkSelfPermission(Manifest.permission.ACCESS_COARSE_LOCATION)!=PackageManage
         //request permissions
         requestPermissions(new String[{Manifest.permission.ACCESS_COARSE_LOCATION,M
         }
```
```

```
        bluemanage = LinkManager.init(this);
        bluemanage.setDebug(true);//whether to print

         //set brainwave data callback
        bluemanage.setMultiEEGPowerDataListener(new EEGPowerDataListener() {
            @Override
            public void onBrainWavedata(String mac, BrainWavebrainWave) {
                  Log.e(mac, brainWave.toString() ); //receive brainwave data
            }
            @Override
            public void onRawData(String mac, int raw) {
                 //receive raw data
            }
            @Override
            public void onGravity( String mac,  Gravity gravity) {
                //receive gravity data
            }
            @Override
            public void onRR(String mac, ArrayList<Integer> rr, int oxygen) {
                //Receiving data: RR intervals and blood oxygen percentage
            }
        });

        //connection status recall
        bluemanage.setOnConnectListener(new OnConnectListener() {
            @Override
           public void onConnectionLost(BlueConnectDeviceblueConnectDevice) {
                // lost connected Bluetooth
            }
            @Override
            public void onConnectStart(BlueConnectDeviceblueConnectDevice) {
                //trying to connect Bluetooth
            }
            @Override
            public void onConnectting(BlueConnectDeviceblueConnectDevice) {
                //connecting…
             }
            @Override
            public void onConnectFailed(BlueConnectDeviceblueConnectDevice) {
                //Bluetooth disconnected
            }

             @Override
            public void onConnectSuccess(BlueConnectDeviceblueConnectDevice) {
                String mac = blueConnectDevice.getAddress();
                String connectType = blueConnectDevice.isBleConnect ? " 4.0 " : " 3.
Log.e(TAG, "connected device name:" + blueConnectDevice.getName() + " mac: " +
                                mac);
```

```
            }

             @Override
            public void onError(Exception e) {
                Log.e(TAG, "connect Bluetooth error");
                e.printStackTrace();
            }
        });

    bluemanage.setMaxConnectSize(1);//Set the maximum of connections
    bluemanage.setConnectType(LinkManager.ConnectType.ALLDEVICE);//set connect type
    bluemanage.setWhiteList("BrainLink_pro,BrainLink_Lite");//set whitelist
    bluemanage.startScan();//start scan and connect
```

MacrotellectLinkAPI Reference

Brainwave Reference

Brainwave information entity class, a real-time analysis of the chip to the software that obtain detailed brainwave information and power data.

Note：\ Connect to BrainLink_Liteand Mind Link，which only have : signal，att，med，delta，theta，lowAlpha，highAlpha，lowBeta，highBeta, lowGamma, middleGamma, heartRate and temperature.

Basic Brainwave Data

- **Signal** (It represents the signal value of the Macrotellect's hardware. When the signal is 0, it means that the hardware has been put on, and when the signal is 200, it means that hardware is connected to the phone via Bluetooth.)
- **att**(Attention)
- **med**(Relaxation)
- **delta**
- **theta**
- **lowAlpha**,
- **highAlpha**
- **lowBeta**
- **highBeta**
- **lowGamma**
- **middleGamma**
- **ap**(Appreciation)
- **batteryCapacity**(BatteryCapacity)
- **heartRate**
- **temperature**

GravityReference

Gravity information entity class, a real-time analysis of the chip to software that obtain detailed gravity information data.

Note： Connect to BrainLink_Litehas no function of receiving gravity data.

Gravity Data - X value: gravity value in The x axis(Pitching angle) - Yvalue:gravity value in The yaxis(Yaw angle) - Z value: gravity value in The z axis(Roll angle)

EEGPowerDataListener Reference

The class for receiving real-time parse chip data needs to be implemented by the user, which receive brainwave、gravity and raw EEG data. Set the data parsing result callback in the `setEegPowerDataListener (OnConnectListenerOnConnectListener)` of the Linkmanager class.

Method

void onBrainWavedata(String mac,BrainWavebrainWave);
Receive parsed brainwave data. - mac: mac address of Brainwave device - brainWave:brainwave data

void onGravity(String mac,Gravity gravity);
Receive parsed gravity data.

- mac: mac address of Brainwave device
- gravity:gravity data

void onRawData(String mac,int raw);
Receive raw EEG data.

- mac: mac address of Brainwave device
- raw:raw EEG data

void onRR(String mac, ArrayList rr, int oxygen);
Receiving data: RR interval and blood oxygen percentage.

- mac: mac address for BrainLink device
- rr: RR intervals
- oxygen: blood oxygen percentage

OnConnectListenerReference

This class is the connection state monitoring interface between Bluetooth and brainwavedevice needs to be implemented by the user.Set up the connection listener in the Linkmanager'ssetOnConnectListener (OnConnectListener).

Method

oid onConnectStart(BlueConnectDeviceblueConnectDevice);
trying to connect
void onConnectting(BlueConnectDeviceblueConnectDevice);
connecting
void onConnectFailed(BlueConnectDeviceblueConnectDevice);
disconnected
void onConnectSuccess(BlueConnectDeviceblueConnectDevice);
connected
void onConnectionLost(BlueConnectDeviceblueConnectDevice);
Lost connection (disconnected from connected state) **void onError(Exception e);**
connection error

LinkManagerReference

This class handles the interaction between the Macrotellect’s hardware and Bluetooth devices.

Method

public static LinkManagerinit(Context context)
Initialization (singleton)
public void setDebug(booleanisDebug)
whether to print log (no print setting by default)
public void setMaxConnectSize(int count)
Set the maximum number of connections(1 is settedby default)
public void setConnectType(ConnectTypeconnectType);
set the type of connection

- **ConnectType. ONLYCLASSBLUE** Only connect devices by classBluetooth, you need to manually pair them first.
- **ConnectType. ONLYBLEBLUE** Only connect devices by BLE Bluetooth
- **ConnectType. ALLDEVICE** Allow both ways to connect devices

getConnectSize();
receive the number of connected devices
public void setWhiteList(String whiteList)
Set whitelist, only allow to connect whitelist. Please use' , 'to separate names of connected multiple devices.

public void setOnConnectListener(OnConnectListeneronConnectListener)

Set the Bluetooth connection status callback

public void setEegPowerDataListener(EEGPowerDataListenereegPowerDataListener)

Set the brainwave data receiving callback

Change Records

Add blood oxygen percentage