

蓝牙温湿度记录仪

开发文档

For Android SDK

状态	<input type="checkbox"/> 草稿 <input type="checkbox"/> 评审 <input checked="" type="checkbox"/> 发布 <input type="checkbox"/> 修订		
版本号	3.0.15		
作者	Forrest	时间	2019.07.14
审核	Lisa	时间	

前 言

蓝牙温度记录仪，是指 BT04、BT04B、BT05、BT05B、TempU06+ 系列设备。该文档是介绍在 Android 平台使用 Java 语言进行 APP 开发。

前 言.....	1
一、 使用 Android Studio 新建工程.....	2
二、 初始化 SDK.....	3
三、 搜索设备.....	3
1. 扫描设备广播.....	3
2. 指定设备 ID.....	4
3. 指定类型设备列表.....	5
四、 提取设备数据报告.....	5
1. 连接设备.....	5
2. 初始化 DataManager 对象.....	6
3. 开启通知.....	6
4. 解锁.....	6
5. 获取设备状态.....	7
6. 设置提取时间段.....	7
7. 获取报警设置.....	7
8. 获取 Mark 设置.....	7
9. 获取提取数据信息.....	7
10. 开启接收.....	8
11. 关闭接收.....	8
12. 生成报告.....	8
13. 其它功能函数介绍.....	8
五、 设备参数修改.....	8
1. 连接设备.....	8
2. 初始化 ConfigManager 对象.....	9
3. 开启通知.....	10
4. 解锁.....	10
5. 读取、修改参数函数介绍.....	10
六、 注意事项.....	10

一、使用 Android Studio 新建工程

将 com.tzone.btlogger.jar 包放入 libs 文件夹下,然后在当前工程下的 build.gradle 文件配置项中的 dependencies 新增内容, 如下 compile files('libs/com.tzone.btlogger.jar')

Android Manifest 文件说明:

```
<uses-permission android:name="android.permission.BLUETOOTH" />
```

```

<uses-permission android:name="android.permission.BLUETOOTH_ADMIN" />

<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />

<uses-feature android:name="android.hardware.bluetooth_le" android:required="true" />

```

二、初始化 SDK

在 APP 启动界面初始化，如 MainActivity 界面

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    .....

    BleManager.getInstance().init(getApplication());
    BleManager.getInstance()
        .setReConnectCount(3, 5000)
        .setConnectOverTime(20000)
        .setOperateTimeout(5000);
    .....
}

```

参考示例程序：com/tzone/btloggerexample/MainActivity.java

三、搜索设备

1. 扫描设备广播

示例如下：

```

BleScanRuleConfig scanRuleConfig = new BleScanRuleConfig.Builder()

    .setScanTimeOut(1000 * 60).build();

BleManager.getInstance().initScanRule(scanRuleConfig);

BleManager.getInstance().scan(new BleScanCallback() {

    @Override

    public void onScanFinished(List<BleDevice> scanResultList) {

```

```

        Log.i(TAG, "onScanFinished => " + scanResultList.size());
    }

    @Override

    public void onScanStarted(boolean success) {

        Log.i(TAG, "onScanStarted => " + success);
    }

    @Override

    public void onScanning(BleDevice bleDevice) {

        if (bleDevice == null)

            return;

        Log.i(TAG, "onScanning => " + bleDevice.getMac() + " " +
bleDevice.getName());

        Scan device = new Scan();

        device.fromBroadcast(bleDevice) ;

        //查询条件

    }

});

```

参考示例程序： `com/tzone/btloggerexample/MainActivity.java`

2. 指定设备 ID

例如搜索指定设备 ID = 012345678，如下：

```

String deviceId = "012345678";
if(device.getID().equals(deviceId)){
    .....
}

```

3. 指定类型设备列表

例如搜索指定设备类型为 TempU06+, 如下:

```
if(device.getDeviceType() == DeviceType.TempU06){  
    .....  
}
```

说明: 1、SDK 中设备类型 TempU06 即为 TempU06+
2、数值=-1000, 代表为空。

四、提取设备数据报告

1. 连接设备

```
BleManager.getInstance().connect(_Device.getMac(), new BleGattCallback() {  
    @Override  
    public void onStartConnect() {  
        Log.i(TAG, "onStartConnect: OK");  
    }  
  
    @Override  
    public void onConnectFail(BleDevice bleDevice, BleException exception) {  
        Log.i(TAG, "onConnectFail: bleDevice => " + (bleDevice != null ?  
bleDevice.getMac() : " null"));  
        BleManager.getInstance().destroy();  
    }  
  
    @Override  
    public void onConnectSuccess(BleDevice bleDevice, BluetoothGatt gatt, int  
status) {  
        Log.i(TAG, "onConnectSuccess: bleDevice => " + (bleDevice != null ?  
bleDevice.getMac() : " null"));  
        _BleDevice = bleDevice;  
        InitDataManager();  
    }  
  
    @Override  
    public void onDisConnected(boolean isActiveDisConnected, BleDevice device,  
BluetoothGatt gatt, int status) {  
        Log.i(TAG, "onDisConnected: bleDevice => " + (device != null ?  
device.getMac() : " null"));  
        _BleDevice = null;  
    }  
});
```

2. 初始化 DataManager 对象

```
public void InitDataManager() {  
    if (_BleDevice == null)  
        return;  
    if (_DataManager == null) {  
        if (_Device.getDeviceType() == DeviceType.TempU06)  
            _DataManager = new com.tzone.bt.u06.DataManager();  
        else if (_Device.getDeviceType() == DeviceType.BT04)  
            _DataManager = new com.tzone.bt.bt04.DataManager();  
        else if (_Device.getDeviceType() == DeviceType.BT04B)  
            _DataManager = new com.tzone.bt.bt04b.DataManager();  
        else if (_Device.getDeviceType() == DeviceType.BT05)  
            _DataManager = new com.tzone.bt.bt05.DataManager();  
        else if (_Device.getDeviceType() == DeviceType.BT05B)  
            _DataManager = new com.tzone.bt.bt05b.DataManager();  
        else {  
            return;  
        }  
        _DataManager.InitSetting(_BleDevice, dataCallback);  
    }  
  
    _Report = null;  
    if (_Device.getDeviceType() == DeviceType.TempU06) {  
        _DataManager.Notify();  
    }else {  
        _DataManager.Unlock(Password);  
    }  
  
}
```

3. 开启通知

```
_DataManager.Notify();
```

说明：BT04、BT04B、BT05、BT05B，不需要开启通知。

4. 解锁

```
_DataManager.Unlock(Password);
```

说明：BT04、BT04B、BT05、BT05B，必须解锁才能操作。TempU06+ 可以根据密码级别，看需求是否需要解锁。详细请查看设备用户手册文档中介绍。

5. 获取设备状态

```
_DataManager.GetLogStatus();
```

//设备状态类型

```
enum DeviceRecordType {  
    Initialize, //初始化状态  
    Stop, //停止(设备按键停止)  
    Stop_USB, //Usb 停止  
    Stop_StorageFull, //存储满停止  
    Stop_App, //APP 停止  
    Recording, //记录中  
    Delay, //延时  
}
```

6. 设置提取时间段

```
long beginTime = 0;  
long endTime = 0;  
_DataManager.SetConfig(beginTime, endTime);
```

说明：为 0 代表，提取设备全部数据。

7. 获取报警设置

```
_DataManager.GetAlarm();
```

8. 获取 Mark 设置

```
_DataManager.GetMark();
```

9. 获取提取数据信息

```
_DataManager.RequestDataInfo();
```

```
String[] info = new String[13];  
info[0] // Data Count  
info[1] // Begin Time  
info[2] // Time Span  
info[3] // Record Status  
info[4] // Delay Time  
info[5] // Repeat Start  
info[6] // Temperature Unit  
info[7] // Stop Button  
info[8] // Start Mode  
info[9] // Start Time
```

```
info[10] // Description
info[11] // NOTE
info[12] // MKT
```

10. 开启接收

```
_DataManager.Receive(true);
```

11. 关闭接收

```
_DataManager.Receive(false);
```

12. 生成报告

```
_Report.Generate();
```

13. 其它功能函数介绍

	功能	函数名	BT04	BT04B	BT05	BT05B	TempU06+	
1	开启记录	SetStart	V	V	V	V	V	
2	停止记录	SetStop	V	V	V	V	V	
3	设置 Mark	SetMark	X	X	X	X	V	BT04/BT04B/BT05/B
4	设置飞行模式	SetFlighMode	X	X	X	X	V	T05B 不支持

参考示例程序: com/tzone/btloggerexample/DeviceActivity.java

五、设备参数修改

1. 连接设备

```
BleManager.getInstance().connect(_Device.getMac(), new BleGattCallback() {
    @Override
    public void onStartConnect() {
        Log.i(TAG, "onStartConnect: OK");
    }

    @Override
    public void onConnectFail(BleDevice bleDevice, BleException exception) {
        Log.i(TAG, "onConnectFail: bleDevice => " + (bleDevice != null ?
bleDevice.getMac() : " null"));
        BleManager.getInstance().destroy();
    }

    @Override
```



```

        public void onConnectSuccess(BleDevice bleDevice, BluetoothGatt gatt, int
status) {
            Log.i(TAG, "onConnectSuccess: bleDevice => " + (bleDevice != null ?
bleDevice.getMac() : " null"));
            _BleDevice = bleDevice;
            InitConfigManager();
        }

        @Override
        public void onDisConnected(boolean isActiveDisConnected, BleDevice device,
BluetoothGatt gatt, int status) {
            Log.i(TAG, "onDisConnected: bleDevice => " + (device != null ?
device.getMac() : " null"));
            _BleDevice = null;
        }
    });

```

2. 初始化 ConfigManager 对象

```

public void InitConfigManager() {
    if (_BleDevice == null)
        return;
    ConfigManagerBase _ConfigManager = null;
    if (_ConfigManager == null) {
        if (_Device.getDeviceType() == DeviceType.TempU06)
            _ConfigManager = new com.tzone.bt.u06.ConfigManager();
        else if (_Device.getDeviceType() == DeviceType.BT04)
            _ConfigManager = new com.tzone.bt.bt04.ConfigManager();
        else if (_Device.getDeviceType() == DeviceType.BT04B)
            _ConfigManager = new com.tzone.bt.bt04b.ConfigManager();
        else if (_Device.getDeviceType() == DeviceType.BT05)
            _ConfigManager = new com.tzone.bt.bt05.ConfigManager();
        else if (_Device.getDeviceType() == DeviceType.BT05B)
            _ConfigManager = new com.tzone.bt.bt05b.ConfigManager();
        else {
            return;
        }
        _ConfigManager.InitSetting(_BleDevice, configCallback);
    }

    if (_Device.getDeviceType() == DeviceType.TempU06) {
        _ConfigManager.Notify();
    } else {
        _ConfigManager.Unlock>Password);
    }
}

```

}

}

3. 开启通知

_DataManager.Notify();

说明：BT04、BT04B、BT05、BT05B，不需要开启通知。

4. 解锁

_DataManager.Unlock(Password);

说明：BT04、BT04B、BT05、BT05B，必须解锁才能操作。TempU06+ 在低级加密、高级加密，必须解锁才能操作。

5. 读取、修改参数函数介绍

	功能	函数名	BT04	BT04B	BT05	BT05B	TempU06+	
1	得到设备信息	GetDeviceInfo	√	√	√	√	√	
2	设置密码	SetPassword	√	√	√	√	√	
3	设置设备名	SetDeviceName	√	√	√	√	√	
4	设置设备时间	SetDateTime	√	√	√	√	√	
5	得到设备时间	GetDateTime	√	√	√	√	√	
6	设置广播参数	SetBroadcastSetting	√	√	√	√	×	TempU06+ 不支持
7	得到广播参数	GetBroadcastSetting	√	√	√	√	×	
8	设置记录参数	SetLogSetting	√	√	√	√	√	
9	得到记录参数	GetLogSetting	√	√	√	√	√	
10	设置 PDF 参数	SetPDFSetting	√	√	√	√	√	BT04/BT04B/BT05/B T05B 只能设置描述、备注
11	得到 PDF 参数	GetPDFSetting	√	√	√	√	√	
12	设置报警参数	SetAlarm	√	√	√	√	√	
13	得到报警参数	GetAlarm	√	√	√	√	√	

说明：1、温度、湿度、电量、电压 = -1000，代表为空；

2、时间 = 0，代表时间错误。

3、不同型号设备功能不相同，如果设备没有该项参数接口返回为空，详细请对照协议文档、操作的手册。

六、注意事项

1、只有在 Android 4.3 版本以上才可以使用 BLE 功能；在 Android 6.0 以上使用扫描方法

必须获取位置权限；某些型号手机还可能需要打开定位功能。

2、建议在两次操作之间间隔一段时间。比如连接成功之后，间隔 **100ms** 再进行 **notify** 或 **wirte**，具体时间数据可以在不同手机上尝试选择一个最短的有效时间。