

---

## **MORE Battery Controller SDK**

---

*Version 0.16.08.15*

---

*2016.08.15*

---

# Table of Contents

|   |   |
|---|---|
| MORE Battery Controller SDK.....              | 1 |
| <i>Version 0.16.08.15</i> .....               | 1 |
| Table of Contents .....                       | 2 |
| 1. Introduction.....                          | 3 |
| 1.1. Overview .....                           | 3 |
| 2. Service.....                               | 4 |
| 2.1. Battery Controller SDK .....             | 4 |
| 2.2. Use SDK Library .....                    | 4 |
| 2.3. Battery Function.....                    | 7 |
| 3. Parameter Definition.....                  | 7 |
| 3.1. OnCallbackResult Function Parameter..... | 7 |
| 敘述電池類型，Ex. Li-ion, Li-Polymer.....            | 8 |
| 3.2. Handler 與 Callback listener 參數對應.....    | 8 |

# **1. Introduction**

## **1.1. Overview**

MORE Battery Controller SDK 主要提供 Smart Mobile Device 更快速與簡潔的開發模組，APP 開發人員只要將 MORE Battery Controller SDK 加入到自己的 APP 專案裡，透過 API 的呼叫即可知道智慧型裝置之電池資訊，目前提供 Android 版本 SDK Library。

## 2. Service

### 2.1. Battery Controller SDK

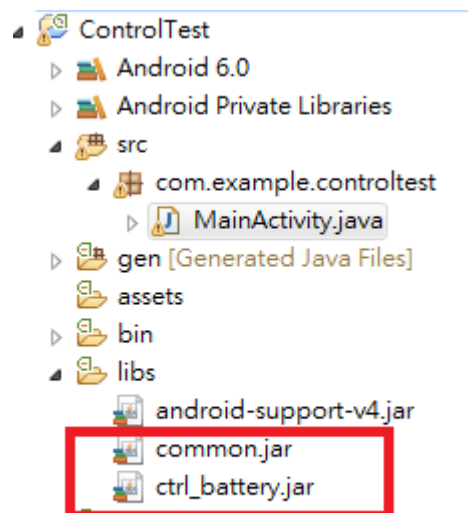
Battery Controller SDK 主要提供給開發者一個能主動通知開發者電量狀況。

### 2.2. Use SDK Library

在使用 Battery Controller SDK 前，需要先透過 Battery Receiver 取得監看電量的權限，開發者可以設定的電量區間內要回報，此模組會透過 Handler 或 Callback Listener 主動通知開發者目前電量。

使用步驟：

1. 先將以下 jar 檔加入到 Android 開發專案的 libs 資料夾



2. 在 AndroidManifest.xml 的 `<application>` 裡新增以下幾行

```
<!-- battery register -->
    <receiver android:name="sdk.ideas.ctrl.battery.BatteryReceiver" >
        <intent-filter>
            <action android:name="android.intent.action.BATTERY_CHANGED" />
        </intent-filter>
    </receiver>
```

### 3. 匯入以下 library

```
import android.os.Bundle;
import android.os.Handler;
import android.os.Message;
import android.util.Log;
import sdk.ideas.common.CtrlType;
import sdk.ideas.common.OnCallbackResult;
import sdk.ideas.ctrl.battery.BatteryHandler;
import java.util.HashMap;
```

### 4. 設定Battery Handler

```
BatteryHandler mBatteryHandler = new BatteryHandler(this);
```

### 5. 本SDK 提供Handler 和Callback Listener方法來去知道程式是否有成功執行，開發者可以以自己喜好去做設定，以下是Handler 方法範例

#### 1) 先宣告Handler

```
private Handler theHandler = new Handler()
{
    @Override
    public void handleMessage(Message msg)
    {
        switch (msg.what)
        {
            case CtrlType.MSG_RESPONSE_BATTERY_HANDLER:
                Log.d("handler response: ", "Result: " + String.valueOf(msg.arg1) + " From: "
                    + String.valueOf(msg.arg2) + " Message: " +
                    ((HashMap<String,String>)msg.obj) );
                break;
            default:
                break;
        }
    }
};
```

2) 設定此 handler

```
mBatteryHandler.setHandler(theHandler);
```

6. Callback Listener 方式

```
mBatteryHandler.setOnCallbackResultListener(new OnCallbackResult()  
{  
    @Override  
    public void onCallbackResult(int result, int what, int from, HashMap<String,  
String> message)  
    {  
        Log.d("Listener response: ", "Result: " + String.valueOf(result) + " What:  
" + String.valueOf(what)  
            + " From: " + String.valueOf(from) + " " + message);  
    }  
});
```

7. 設定是否要看詳細資料(預設是 false)

```
mBatteryHandler.setDetailBatteryInfo(false);
```

8. 設定電量區間內要回報(預設是當上升 5%或下降 5% 會回報)

```
mBatteryHandler.setDiffBatteryLevel(5);
```

9. 開始去監看電池電量，可在 OnResume 時呼叫。

```
mBatteryHandler.startListenAction();
```

10. 停止監看電池電量，可在 onPause, onStop 時呼叫。

```
mBatteryHandler.stopListenAction();
```

## 2.3. Battery Function

| Camera Controller API  |  |
|--|--|
| Function   | Description  |
| BatteryHandler(Context mContext)                                   | 建構元  |
| <b>void</b> setDetailBatteryInfo( <b>boolean</b> isNeedDetailInfo) | 設定是否要電池詳細資訊，輸入 false 則 message 只有目前電量，true 則包含完整資訊，預設是 false，參考 3.1 message 部分 |
| <b>void</b> setDiffBatteryLevel( <b>int</b> diff)                  | 設定電池下降幾%要回報，預設是 5%   |
| <b>void</b> startListenAction()                                    | 開始去監看電池電量  |
| <b>void</b> stopListenAction()                                     | 停止監看電池電量   |
| <b>void</b> setHandler(Handler handler)                            | 設定 Handler，參數對應可參考 3.2   |
| <b>void</b> setOnCallbackResultListener(OnCallbackResult listener) | 設定 callback listener，其參數參考 3.1   |

## 3. Parameter Definition

### 3.1. OnCallbackResult Function Parameter

result :

| Value | Description   |
|-------|---------------|
| 1     | SUCCESS       |
| 0     | UNKNOWN ERROR |

**what :**

| Value | Description                |
|-------|----------------------------|
| 1035  | BATTERY CONTROLLER HANDLER |

**from :**

| Value | Description     |
|-------|-----------------|
| 0     | BATTERY FUCTION |

**message :**

| key           | Description   |
|---------------|---|
| “level”       | 敘述電池剩餘電量  |
| “scale”       | 敘述電池最大值電量，通常為 100   |
| “icon-small”  | 電池圖示 ID   |
| “voltage”     | 敘述電池剩餘電量 mV   |
| “temperature” | 敘述電池攝氏溫度  |
| “technology”  | 敘述電池類型，Ex. Li-ion, Li-Polymer   |
| “plugged”     | 敘述電池如果有插線是哪種充電，AC, USB  |
| “status”      | 電池狀態，有 Unknown, Charging, Discharging, Not Charging, Full               |
| “health”      | 電池健康度，有 Unknown, Good, Overheat, Dead, OverVoltage, Unspecified Failure |
| “present”     | 是否為現在使用的電池  |
| “message”     | 當 result = UNKNOWN ERROR 時會 show 出錯誤訊息                                  |

### 3.2. Handler 與 Callback listener 參數對應

| Handler  | OnCallbackResult |
|----------|------------------|
| msg.what | int what         |
| msg.arg1 | int result       |



|          |                                 |
|----------|---------------------------------|
| msg.arg2 | int from                        |
| msg.obj  | HashMap<String, String> message |