|  |
| --- |
|  |
| **Binatone Android SDK Description** |
| **V1.0** |
|  |
| **Author: jaker** |
| **2018/08/31** |

# Change log

|  |  |  |
| --- | --- | --- |
| **Date** | **Log** | **Author** |
| 2018-08-31 | Create | jaker |
| 2019/05/08 | Adding MTU setup interface | smile |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |
| --- |
| **内部文档 严禁外传** |

# Catalog

[Change log 1](#_Toc536446972)

[Catalog 2](#_Toc536446973)

[Android SDK Intro 6](#_Toc536446974)

[1. Function and Purpose 6](#_Toc536446975)

[Integration 6](#_Toc536446976)

[1 .SDK framework 6](#_Toc536446977)

[2 .Integration 6](#_Toc536446978)

[Eclipse Config 6](#_Toc536446979)

[API 8](#_Toc536446980)

[1.API initialization 8](#_Toc536446981)

[Description 8](#_Toc536446982)

[Parameters 8](#_Toc536446983)

[2. Connnect Device 8](#_Toc536446984)

[Description 8](#_Toc536446985)

[Parameters 8](#_Toc536446986)

[3. Get SDK Version 9](#_Toc536446987)

[Description 9](#_Toc536446988)

[4.Get Battery 9](#_Toc536446989)

[Description 9](#_Toc536446990)

[Parameters 9](#_Toc536446991)

[3. Get Device Information 9](#_Toc536446992)

[Description 10](#_Toc536446993)

[Parameters 10](#_Toc536446994)

[4. Get Device Version 10](#_Toc536446995)

[Description 10](#_Toc536446996)

[Parameters 10](#_Toc536446997)

[5. Get Birthday 10](#_Toc536446998)

[Description 10](#_Toc536446999)

[Parameters 10](#_Toc536447000)

[6. Set Birthday 11](#_Toc536447001)

[Description 11](#_Toc536447002)

[Parameters 11](#_Toc536447003)

[7. Get Alarm 11](#_Toc536447004)

[Description 11](#_Toc536447005)

[Parameters 11](#_Toc536447006)

[8. Set Alarm 11](#_Toc536447007)

[Description 12](#_Toc536447008)

[Parameters 12](#_Toc536447009)

[9. Get Sleep Data (Real-time) 12](#_Toc536447010)

[Description 12](#_Toc536447011)

[Parameters 12](#_Toc536447012)

[10. Stop Getting Data(Real-time) 12](#_Toc536447013)

[Description 12](#_Toc536447014)

[Parameters 12](#_Toc536447015)

[11. Get 24 hours Sleep Report 13](#_Toc536447016)

[Description 13](#_Toc536447017)

[Parameters 13](#_Toc536447018)

[12. Get Sleep Reports 13](#_Toc536447019)

[Description 13](#_Toc536447020)

[Parameters 13](#_Toc536447021)

[13. Firmware Update 1 14](#_Toc536447022)

[Description 14](#_Toc536447023)

[Parameters 14](#_Toc536447024)

[14. Firmware Update 2 14](#_Toc536447025)

[Description 14](#_Toc536447026)

[Parameters 14](#_Toc536447027)

[15. Restore factory settings 14](#_Toc536447028)

[Description 15](#_Toc536447029)

[Parameters 15](#_Toc536447030)

[Object Description 15](#_Toc536447031)

[StatusCode 15](#_Toc536447032)

[Description 15](#_Toc536447033)

[Fields 15](#_Toc536447034)

[IResultCallback<T> 15](#_Toc536447035)

[Description 15](#_Toc536447036)

[Function 16](#_Toc536447037)

[CallbackData<T> 16](#_Toc536447038)

[Description 16](#_Toc536447039)

[Fields 16](#_Toc536447040)

[DeviceInfo 16](#_Toc536447041)

[Description 16](#_Toc536447042)

[Fields 16](#_Toc536447043)

[BatteryBean 17](#_Toc536447044)

[Description 17](#_Toc536447045)

[Fields 17](#_Toc536447046)

[RealTimeData 17](#_Toc536447047)

[Description 17](#_Toc536447048)

[Fields 17](#_Toc536447049)

[HistoryData 17](#_Toc536447050)

[Description 17](#_Toc536447051)

[Fields 18](#_Toc536447052)

[Summary 18](#_Toc536447053)

[Description 18](#_Toc536447054)

[Fields 18](#_Toc536447055)

[Detail 18](#_Toc536447056)

[Description 18](#_Toc536447057)

[Fields 18](#_Toc536447058)

[Analysis 19](#_Toc536447059)

[Description 19](#_Toc536447060)

[Fields 19](#_Toc536447061)

[AlarmConfig 19](#_Toc536447062)

[Description 19](#_Toc536447063)

[Fields 19](#_Toc536447064)

[Birthday 20](#_Toc536447065)

[Description 20](#_Toc536447066)

[Fields 20](#_Toc536447067)

[SleepStatusType 20](#_Toc536447068)

[Description 20](#_Toc536447069)

[Fields 20](#_Toc536447070)

# Android SDK Intro

## Function and Purpose

Sleep\_sensor SDK,launched by Sleepace, is a software development kit for fast Internet APP development on android platform.

The SDK encapsulates the communication process between APP and hardware, and provides functions such as device configuration, device control and data query. Using the SDK, users do not need to care about complex communication protocols and the underlying implementation, only need to focus on the APP interaction and business level.

# Integration

## 1 .SDK framework

|  |  |
| --- | --- |
| **Framework** | **Description** |
| sdkcore.jar | SDK base core |
| binatonesdk.jar | Binatone SDK |
| sdkalgorithm.jar | Algorithm call library |
| libalgorithm.so | Algorithm library |

## 2 .Integration

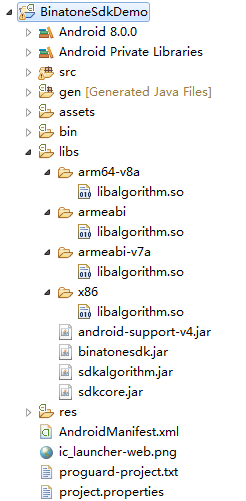
There are many Android development tools, and here we introduce the engineering configuration method of Sleepace SDK with Eclipse.

## Eclipse Config

**Step 1**：

In the project to create a "libs" folder, copy sdkcore.jar, binatonesdk.jar, sdkalgorithm.jar to "libs" folder, copy libalgorithm.so to "libs \ armeabi" folder.

Like this:



**Setp 2:**

Config the “AndroidManifest.xml”

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

<uses-permission android:name=*"android.permission.BLUETOOTH\_ADMIN"*/>

<uses-feature android:name=*"android.hardware.bluetooth\_le"* android:required=*"true"*/>

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

<uses-permission android:name=*"android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"*/>

<uses-permission android:name=*"android.permission.WRITE\_EXTERNAL\_STORAGE"*/>

<uses-permission android:name=*"android.permission.ACCESS\_COARSE\_LOCATION"* />

<uses-permission android:name=*"android.permission.ACCESS\_FINE\_LOCATION"* />

# API

## 1.API initialization

BinatoneHelper.getInstance(Context mContext);

### Description

BinatoneHelper Initialization

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| mContext | Context | Conetxt |

## SET MTU

**public** **void** setMtu(**int** mtu, [IResultCallback](#_IDataCallback<T>)<Integer> cb)

### Description

Used to set the size of Bluetooth device mtu, call before connecting device, refer to Demo

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| mtu | int | MTU size, range: [20, 509] |
| cb | [IResultCallback](#_IDataCallback<T>)<Integer> | Callback interface, when set successfully, callback. getResult returns the MUT size actually supported by the device, which is the real writable data size, excluding one byte of ATT opcode and two bytes of ATT handle. |

## Connnect Device

**public** **void** login(String deviceName, String address, **int** userId, **int** timestamp**, int** timezone**, int** dstOffset**, int** timeout, [IResultCallback](#_IDataCallback<T>)<DeviceInfo> cb)

### Description

Connect Reston and setting userId

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| deviceName | String | deviceName |
| address | String | Bluetooth address |
| userId | int | userId does not belong to Sleepace.  userId belong to partner  **Why need it：**  Reston separates the data according to userId.  It mean user A connects to device, generates and gets the data which only belong to user A. User A can’t get the data of user B |
| timestamp | int | Unix timestamp, Unit(second) |
| timezone | int | time zone, Unit(second) |
| dstOffset | int | Summer time offset, unit seconds |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<DeviceInfo> | Callback function, if success,return [DeviceInfo](#_LoginBean) Obj |

## 3. Get SDK Version

**public** String getSDKVersion()

### Description

get sdk version

## 4.Get Battery

**public** **void** getBattery(**int** timeout, IResultCallback<BatteryBean> cb)

### Description

Get battery

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[BatteryBean](#_BatteryBean)> | Callback function, if success,return [BatteryBean](#_BatteryBean) Object |

## 5.Get Device Information

**public** **void** getDeviceInfo(**int** timeout, IResultCallback<DeviceInfo> cb)

### Description

Get device information

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[DeviceInfo](#_DeviceInfo)> | Callback function, if success,return [DeviceInfo](#_DeviceInfo) Object |

## 6.Get Device Version

**public** **void** getDeviceVersion(**int** timeout, IResultCallback<String> cb)

### Description

Get current version of device

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<String> | Callback function, if success,return the version of device |

## 7.Get Birthday

**public** **void** getBirthday(**int** timeout, IResultCallback<[Birthday](#_Birthday)> cb)

### Description

Get birthday

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[Birthday](#_Birthday)> | Callback function, if success,return the birthday |

## 8.Set Birthday

**public** **void** setBirthday(**int** year, **int** month, **int** day, **int** timeout, IResultCallback<Void> cb)

### Description

set birthday

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| year | int | year |
| month | int | month of year[1-12] |
| day | int | day of month |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<Void> | Callback function |

## 9.Get Alarm

**public** **void** getAlarm(**int** timeout, IResultCallback<[AlarmConfig](#_AlarmConfig_1)> cb)

### Description

Get alarmconfig

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[AlarmConfig](#_AlarmConfig_1)> | Callback function, if success,return the AlarmConfig |

## 10.Set Alarm

**public** **void** setAlarm(**boolean** enable, **int** hour, **int** minute, **int** duration, **int** timeout, IResultCallback<[Void](#_AlarmConfig)> cb)

### Description

set alarm

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| enable | boolean | alarm switch |
| hour | int | hour of day[0-23] |
| minute | int | minute |
| duration | int | Monitoring time, Unit((minute) |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[Void](#_AlarmConfig)> | Callback function |

## 11.Get Sleep Data (Real-time)

**public** **void** startRealTimeData(**int** timeout, IResultCallback<[RealTimeData](#_RealTimeData_1)> cb)

### Description

Get Real-time Data

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<[RealTimeData](#_RealTimeData)> | Callback function |

## 12.Stop Getting Data(Real-time)

**public** **void** stopRealTimeData(**int** timeout, IResultCallback<Void> cb)

### Description

Stop getting real-time data

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<Void> | Callback function |

## 

## 13.Get 24 hours Sleep Report

**public** **void** getLast24HourData(**int** endTime, **int** sex, IResultCallback<HistoryData> cb)

### Description

Get 24 hours of sleep data

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| endTime | int | The time of end point,Unit(Second) |
| sex | int | Gender,1:male 0:female |
| cb | [IResultCallback<HistoryData>](#_HistoryData) | Callback function |

## 14.Get Sleep Reports

**public** **void** historyDownload(**int** startTime, **int** endTime, **int** sex, IResultCallback<List<HistoryData>> cb)

### Description

Get historical sleep data

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| starTime | int | Start time(timestamp), Unit(second) |
| endTime | int | End time(timestamp), Unit(second) |
| sex | int | Gender,1:male 0:female |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<List<[HistoryData](#_HistoryData)>> | Callback function |

## 15.Firmware Update 1

**public** **void** upgradeDevice(**final** **long** crcDes, **final** **long** crcBin, **final** File file, **final** IResultCallback<Integer> cb)

### Description

Firmware Update

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| crcDes | long | Get it from Sleepace |
| crcBin | long | Get it from Sleepace |
| file | File | Firmware object |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<Integer> | Callback function, Return upgrade progress |

## 16.Firmware Update 2

**public** **void** upgradeDevice(**final** **long** crcDes, **final** **long** crcBin, **final** InputStream is, **final** IResultCallback<Integer> cb)

### Description

Firmware Update

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| crcDes | long | Get it from Sleepace |
| crcBin | long | Get it from Sleepace |
| is | InputStream | file inputstream |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<Integer> | Callback function, Return upgrade progress |

## 17.Restore factory settings

**public** **void** restoreFactorySettings(**int** timeout, IDataCallback<Void> cb)

### Description

Restore factory settings

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| timeout | int | Timeout, Unit((Millisecond) |
| cb | [IResultCallback](#_IDataCallback<T>)<Void> | Callback function |

## 18. Analysis of Sleep Data

com.sleepace.sdk.binatone.util.AnalysisUtil

**public** **static** Analysis analysData(Summary summ, Detail detail,

**int** sex)

### Description

### Sleep data analysis

### Parameters

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| summ | Summary | Summary structure of sleep data |
| detail | Detail | Detail structure of sleep data |
| sex | int | 1: Male, 0: Female |

### Return：Analysis

## 19. Stop downloading historical data

**public** **void** abortHistoryDownload()

### Description

### Stop downloading historical data

## 20.Termination of access to the latest 24-hour historical data

**public** **void** abortHistoryDownload()

### Description

Termination of access to the latest 24-hour historical data

# Object Description

## StatusCode

### Description

Status of execution

### Fields

|  |  |
| --- | --- |
| **Field** | **Description** |
| SUCCESS | success |
| FAIL | failed |
| TIMEOUT | timeout |
| DISCONNECT | Bluetooth is disconnected |
| NOT\_ENABLE | Bluetooth is not open |
| PARAMETER\_ERROR | Parameter error |

## IResultCallback<T>

### Description

Callback interface

### Function

**void** onResultCallback(CallbackData<T> cd)

callback function

## CallbackData<T>

### Description

Callback object

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| status | Int | [Status](#_StatusCode) of execution |
| type | int | Interface Type, used to distinguish between operating interface |
| result | T | The result of execution |

## DeviceInfo

### Description

The result of Connnect Device.

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| hardwareVersion | String | Device version |
| deviceId | String | Device id |

## BatteryBean

### Description

The result of getting battery

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| chargingState | int | Charging or not，  0：not  1: charging |
| quantity | int | Percentage of battery, It’s disable when charging |

## RealTimeData

### Description

The result of getting sleep data (Real-time)

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| heartRate | short | Heart rate |
| breathRate | short | Breath rate |
| status | byte | [SleepStatusType](#_SleepStatusType_1) |
| statusValue | int | The value of status. Unit(second) |

## HistoryData

### Description

The result of getting sleep report

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| summary | Summary | History data [Summary](#_Summary) |
| detail | Detail | History data [Detail](#_Detail) |
| analy | Analysis | [Analysis](#_Analysis) |

## Summary

### Description

Summary of sleep report

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| recordCount | int | The length of Collecting.  Unit(minute)  Eg:  300. It means you collect for 300 minutes |
| startTime | int | Start time(timestamp). Unit(second) |

## Detail

### Description

Detail of sleep report

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| breathRate | int[] | Breath rate |
| heartRate | int[] | Heart rate |
| status | int[] | Status |
| statusValue | int[] | The value of status |

## Analysis

### Description

Analysis of sleep report

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| avgBreathRate | int | Average breath rate(n counts per min) |
| avgHeartRate | int | Average heart rate(n counts per min) |
| duration | int | Sleep duration(Unit:min) |
| wake | int | Duration of Awake)(Unit:min) |
| outOfBedDuration | int | Counts of leaving bed |
| discTime | int |  |
| algorithmVer | String | Algorithm version |
| sca\_array | byte[] | Sleep state array  0: device is not connected;  1: not in bed;  2: asleep;  3: awake; |

## AlarmConfig

### Description

Alarm info

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| enable | boolean | Alarm switch |
| hour | int | alarm time hour[0-23] |
| minute | int | alarm time minute[0-59] |
| duration | int | Monitoring time(Unit:min) |

## Birthday

### Description

birthday

### Fields

|  |  |  |
| --- | --- | --- |
| **Field** | **Type** | **Description** |
| year | int | year |
| month | int | month of year[1-12] |
| day | int | day of month |

## SleepStatusType

### Description

Status value of monitoring

### Fields

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Value** | **Description** |
| SLEEP\_OK | byte | 0x00 | normal |
| SLEEP\_INIT | byte | 0x01 | init |
| SLEEP\_B\_STOP | byte | 0x02 | apnea |
| SLEEP\_H\_STOP | byte | 0x03 | Heartbeat pause |
| SLEEP\_BODYMOVE | byte | 0x04 | Body movement |
| SLEEP\_LEAVE | byte | 0x05 | Leaving bed |
| SLEEP\_TURN\_OVER | byte | 0x06 | Turning over |
| SLEEP\_BODYMOVE\_TEMP | byte | 0x07 | Amplitude of body motion |
| SLEEP\_INVALID | byte | -1 | invalid |