

## CH9140Library 库接口函数说明

版本: 1A

<http://wch.cn>

### 概述

CH9140Library是安卓系统下的BLE串口接口库, 用于为CH914X低功耗蓝牙转串口的芯片模组, 提供BLE串口设备的自动搜索连接、打开关闭、串口参数设置、MODEM操作、串口读写等操作API函数。

支持Android 5.1及以上系统, 蓝牙控制器4.0及以上版本。

### 1、权限声明

需要在 AndroidManifest.xml 清单文件中声明以下权限, 部分权限需要动态申请:

```
<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" />
```

### 2、getInstance

```
public static CH9140BluetoothManager getInstance()
```

用于创建全局唯一实例

返回	返回全局唯一实例
----	----------

### 3、init

```
public void init(@NonNull
    android.app.Application application)
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

初始化上下文环境, 需要在自定义 Application 中使用

返回	返回全局唯一实例
参数	application - 全局上下文
抛出	cn.wch.ch9140lib.exception.CH9140LibException

### 4、startEnumDevices

```
public void startEnumDevices(@NonNull
    cn.wch.ch9140lib.callback.EnumResult enumResult)
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

开始枚举附近设备

参数	enumResult - 枚举结果回调
抛出	cn.wch.ch9140lib.exception.CH9140LibException

### 5、stopEnumDevices

```
public void stopEnumDevices()
```

throws cn.wch.ch9140lib.exception.CH9140LibException

停止枚举设备

参数	scanCallback - 扫描回调
抛出	throws cn.wch.ch9140lib.exception.CH9140LibException

## 6、openDevice

```
public void openDevice(java.lang.String mac,  
                        long timeout,  
cn.wch.ch9140lib.callback.ConnectStatus connectStatus)  
                        throws cn.wch.ch9140lib.exception.CH9140LibException
```

根据 MAC 地址打开设备

参数	mac - MAC 地址 timeout - 连接超时时间, 单位为 ms connectStatus - 连接状态回调
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 7、closeDevice

```
public void closeDevice(java.lang.String mac,  
                        boolean force)  
                        throws cn.wch.ch9140lib.exception.CH9140LibException
```

主动关闭设备

参数	force - 是否强制断开: true 强制断开释放资源, 可能不会有连接状态 (ConnectStatus) 断开的回调; false 正常断开, 一般会产生断开回调
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 8、registerSerialModemNotify

```
public void registerSerialModemNotify(cn.wch.ch9140lib.callback.ModemStatus  
modemNotify)  
注册上报串口以及 Modem 状态的通知
```

参数	modemNotify - 通知回调, 为 null 可以取消回调
----	-----------------------------------

## 9、setMTU

```
public void setMTU(int mtu,  
cn.wch.ch9140lib.callback.CH9140MTUCallback ch9140MTUCallback)  
                        throws cn.wch.ch9140lib.exception.CH9140LibException
```

设置 MTU

参数	mtu - 希望设置的 MTU 大小 ch9140MTUCallback - 设置 MTU 结果回调
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 10、getMTU

```
public int getMTU()  
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

获取当前 MTU

返回	返回当前 MTU
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 11、setSerialBaud

```
public boolean setSerialBaud(int baudRate,  
                             int dataBit,  
                             int stopBit,  
                             int parity)  
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

设置波特率以及其他参数

返回	操作成功或者失败
参数	baudRate - 波特率例如 9600, 115200... dataBit - 数据位 5-8 stopBit - 停止位 1-2 parity - 校验位 0: 无校验; 1: 奇校验; 2: 偶校验; 3: 标志位; 4: 空白位
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 12、setSerialModem

```
public boolean setSerialModem(boolean flow,  
                              int DTR,  
                              int RTS)  
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

设置硬件流控开关以及 Modem 状态

返回	操作成功或者失败
参数	flow - 硬件自动流控开关 true 开; false 关 DTR - 1 有效, 低电平; 0 无效, 高电平 RTS - 1 有效, 低电平; 0 无效, 高电平
抛出	cn.wch.ch9140lib.exception.CH9140LibException

## 13、isDeviceOpened

```
public boolean isDeviceOpened(java.lang.String mac)  
    throws cn.wch.ch9140lib.exception.CH9140LibException
```

检查蓝牙设备是否正在被连接

返回	true 被连接; false 未被连接;
参数	mac - 蓝牙 MAC 地址

#### 14、write

```
public int write(@NonNull  
byte[] data,  
int length)
```

通过蓝牙写数据（这是一个耗时操作，避免在主线程执行）

返回	该值为负，则发送出错；不为负，为发送成功的数据长度
参数	data - 数据数组 length - 写数组的长度，从 data[0] 开始

#### 15、stopWrite

```
public void stopWrite()  
取消发送
```

#### 16、registerRSSINotify

```
public void registerRSSINotify(cn.wch.ch9140lib.callback.CH9140RSSICallback  
ch9140RSSICallback)
```

注册 RSSI 信号强度的通知, 只有先主动调用 readRSSI() 函数，回调才会返回结果

参数	ch9140RSSICallback - RSSI 状态回调, 为 null 可以取消回调
----	---

#### 17、readRSSI

```
public boolean readRSSI()  
用于连接设备后，主动读取 RSSI 值。需配合 registerRSSINotify() 函数使用
```

返回	true 读取成功;false 读取失败
----	----------------------

#### 18、isSupportedFirmware

```
public boolean isSupportedFirmware()  
throws cn.wch.ch9140lib.exception.CH9140LibException
```

用于连接成功后调用，区分新旧版本固件，当前不支持旧版本固件

返回	true 支持的新版本固件;false 不支持的旧版本固件
----	-------------------------------