

自定义模块串口 SDK API 说明 V1.0

Customed Module UART SDK API Documentation V1.0

一、SDK概览

I.SDK Overview

二、详细API说明

II. Detail API Documentation

2.1 powerOn(int)

2.2 powerOff(int)

2.3 ioctl_gpio(int, boolean)

2.4 init(int, int)

2.5 init(int, int, int, int, int)

2.6 openSerial(String, int, int, int, int)

2.7 closeSerial()

2.8 send(byte[])

2.9 receive()

2.10 receiveEx()

2.11 sendAndReceive(byte[], byte[])

三、部分机型串口路径

III. UART Path of Some Models

一、SDK 概览

I. SDK Overview

`com.rscja.deviceapi.Module` 是面向自定义模块的串口操作类，功能涵盖串口启停控制、数据收发，以及相关 GPIO 引脚的上下电管理。

`com.rscja.deviceapi.Module` is a serial port operation class for custom modules, which covers serial port start/stop control, data transmission and reception, as well as power-on/power-off management of related GPIO pins.

1. 简单易用的单例设计

1. Simple and Easy-to-Use Singleton Design

您无需手动创建该类的实例，只需通过 `getInstance()` 方法即可获取全局唯一的管理对象，确保调用过程简单且资源高效。

You do not need to manually create an instance of this class. You can obtain the globally unique management object only through the `getInstance()` method, ensuring that the calling process is simple and resource-efficient.

2. 使用步骤

2.Usage Steps

第一步: 打开串口 `init()`或者`openSerial()`
Step 1: Open the serial port via `init()` or `openSerial()`
第二步: 收发数据 `receiveEx()`、`send()`
Step 2: Send and receive data using `receive()` and `send()`
第三步: 关闭串口 `free()`或者`closeSerial()`
Step 3: Close the serial port via `free()` or `closeSerial()`

3.使用示例

3.Usage Examples

```
public class Test{
    public void main(){
        Module module=Module.getInstance();
        if(module.init(module,baudrate)){
            new Thread(new Runnable() {
                @Override
                public void run() {
                    while (!isStop){
                        byte[] data= module.receiveEx();

                    }
                }
            }).start();
        }
        .....
        module.send(data);
        .....
        module.free();
    }
}
```

二、详细API说明

II.Detailed API description

2.1 powerOn(int)

项目 Item	说明 Description
原型 Prototype	boolean powerOn(int module)
描述 Description	对指定功能模块执行上电操作，仅完成模块上电但不开启串口通信，需配合 <code>powerOff(int module)</code> 方法完成模块下电操作 Perform power-on operation for the specified functional module, which only powers on the module without enabling serial port communication. It must be used in conjunction with the <code>powerOff(int module)</code> method to power off the module.
参数 Parameters	module: 模块编号 3-UHF模块 4-Finger指纹模块 module: Module number, 3-UHF module, 4-Fingerprint module
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.2 powerOff(int module)

项目 Item	说明 Description
原型 Prototype	boolean powerOff(int module)
描述 Description	对指定功能模块执行下电操作，仅完成模块下电但不关闭串口通信，需与 powerOn(int module) 方法配合使用 Perform power-off operation for the specified functional module, which only powers off the module without closing the serial port communication. It must be used in conjunction with the powerOn(int module) method.
参数 Parameters	module: 模块编号 3-UHF模块 4-Finger指纹模块 module: Module number, 3-UHF module, 4-Fingerprint module
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.3 ioctl_gpio(int, boolean)

项目 Item	说明 Description
原型 Prototype	void ioctl_gpio(int gpio, boolean isHigh)
描述 Description	GPIO引脚电平控制方法，用于设置指定GPIO引脚的输出电平状态，该方法仅支持MTK（联发科）平台的设备机型 GPIO pin level control method, used to set the output level state of the specified GPIO pin, this method only supports device models based on the MTK (MediaTek) platform
参数 Parameters	gpio: GPIO引脚编号（需匹配设备硬件定义的有效引脚号） gpio: GPIO pin number (must match the valid pin number defined by the device hardware) isHigh: 引脚电平状态，true:设置为高电平，false:设置为低电平 isHigh: Pin level state, true:set to high level, false:set to low level
返回 Return	无 N/A

2.4 init(int, int)

项目 Item	说明 Description
原型 Prototype	synchronized boolean init(int module, int baudrate)
描述 Description	模块初始化方法（简化版），执行指定功能模块上电操作的同时打开串口通信，通过自定义波特率配置串口参数，使用完成后需调用free()方法释放模块资源 Module initialization method (simplified version), performs power-on operation for the specified functional module and opens serial port communication at the same time, configures serial port parameters through a custom baud rate, and the free() method must be called to release module resources after use
参数 Parameters	module: 模块编号 3-UHF模块 4-Finger指纹模块 module: Module number, 3-UHF module, 4-Fingerprint module baudrate: 自定义串口波特率（需为模块支持的有效波特率值） baudrate: Custom serial port baud rate (must be a valid baud rate value supported by the module)

项目 Item	说明 Description
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.5 init(int, int, int, int, int)

项目 Item	说明 Description
原型 Prototype	synchronized boolean init(int module, int baudrate, int databits, int stopbits, int check)
描述 Description	模块初始化方法（完整版），执行指定功能模块上电操作的同时打开串口通信，支持自定义完整的串口参数（波特率、数据位、停止位、校验位），使用完成后需调用free()方法释放模块资源 Module initialization method (full version), performs power-on operation for the specified functional module and opens serial port communication at the same time, supports customizing complete serial port parameters (baud rate, data bits, stop bits, parity bits), and the free() method must be called to release module resources after use
参数 Parameters	module: 模块编号 3-UHF模块 4-Finger指纹模块 module: Module number, 3-UHF module, 4-Fingerprint module baudrate: 自定义串口波特率（需为模块支持的有效波特率值） baudrate: Custom serial port baud rate (must be a valid baud rate value supported by the module) databits: 串口数据位，仅支持7或8 databits: Serial port data bits, only 7 or 8 are supported stopbits: 串口停止位，仅支持1或2 stopbits: Serial port stop bits, only 1 or 2 are supported check: 串口校验类型，0=无校验，1=奇校验，2=偶校验 check: Serial port parity type, 0 = None, 1 = Odd, 2 = Even
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.6 openSerial(String, int, int, int, int)

项目 Item	说明 Description
原型 Prototype	synchronized boolean openSerial(String path, int baudrate, int databits, int stopbits, int check)
描述 Description	串口单独打开方法，仅完成串口的开启操作且不会对功能模块进行上电，需配合closeSerial()方法关闭串口，适用于模块已上电场景下的串口独立配置 Serial port independent opening method, only completes the serial port opening operation and does not power on the functional module, must be used with the closeSerial() method to close the serial port, suitable for independent serial port configuration when the module is already powered on

项目 Item	说明 Description
参数 Parameters	path: 串口设备路径 (如/dev/ttyS1, 需匹配设备实际串口路径) path: Serial port device path (e.g., /dev/ttyS1, must match the actual serial port path of the device) baudrate: 串口波特率 (需为模块支持的有效波特率值) baudrate: Serial port baud rate (must be a valid baud rate value supported by the module) databits: 串口数据位, 仅支持7或8 databits: Serial port data bits, only 7 or 8 are supported stopbits: 串口停止位, 仅支持1或2 stopbits: Serial port stop bits, only 1 or 2 are supported check: 串口校验类型, 0=无校验, 1=奇校验, 2=偶校验 check: Serial port parity type, 0 = None, 1 = Odd, 2 = Even
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.7 closeSerial()

项目 Item	说明 Description
原型 Prototype	synchronized boolean closeSerial()
描述 Description	串口单独关闭方法, 仅完成串口的关闭操作且不会对功能模块进行下电, 需与openSerial()方法配合使用, 串口使用完成后需及时调用该方法释放串口资源 Serial port independent closing method, only completes the serial port closing operation and does not power off the functional module, must be used with the openSerial() method, and this method must be called in a timely manner to release serial port resources after serial port use
参数 Parameters	无 N/A
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.8 send(byte[])

项目 Item	说明 Description
原型 Prototype	synchronized boolean send(byte[] data)
描述 Description	串口数据发送方法, 向已打开的串口发送指定字节数组数据, 需确保串口已通过openSerial()或init()方法成功开启 Serial port data sending method, sends the specified byte array data to the opened serial port, and it is necessary to ensure that the serial port has been successfully opened through the openSerial() or init() method
参数 Parameters	data: 待发送的字节数组数据 data: Byte array data to be sent
返回 Return	true:成功 false:失败 true: Success, false: Failure

2.9 receive()

项目 Item	说明 Description
原型 Prototype	byte[] receive()
描述 Description	串口数据接收方法（带等待），读取串口缓冲区中的数据；若当前无可用数据，会阻塞等待最多1秒后返回，需配合send()方法或外部设备数据发送使用 Serial port data receiving method (with waiting), reads data in the serial port buffer; if no data is available currently, it will block and wait for a maximum of 1 second before returning, and must be used with the send() method or external device data sending
参数 Parameters	无 N/A
返回 Return	串口接收到的字节数组数据；若无数据/等待超时，返回null Byte array data received by the serial port; returns null if no data/wait timeout

2.10 receiveEx()

项目 Item	说明 Description
原型 Prototype	byte[] receiveEx()
描述 Description	串口数据接收方法（无等待），读取串口缓冲区中的数据；若当前无可用数据，立即返回null，无阻塞等待时间，适用于非阻塞式数据读取场景 Serial port data receiving method (no waiting), reads data in the serial port buffer; if no data is available currently, returns null immediately with no blocking wait time, suitable for non-blocking data reading scenarios
参数 Parameters	无 N/A
返回 Return	串口接收到的字节数组数据；若无数据，立即返回null Byte array data received by the serial port; returns null immediately if no data

2.11 sendAndReceive(byte[], byte[])

项目 Item	说明 Description
原型 Prototype	synchronized int sendAndReceive(byte[] sendData, byte[] outData)
描述 Description	串口数据收发一体化方法，先发送指定数据，再接收响应数据；接收数据时最多阻塞等待3秒，适用于请求-响应式的串口通信场景 Serial port integrated send and receive method, first sends the specified data, then receives response data; blocks and waits for a maximum of 3 seconds when receiving data, suitable for request-response serial port communication scenarios
参数 Parameters	sendData：待发送的字节数组数据 sendData: Byte array data to be sent outData：用于存储接收数据的字节数组缓冲区 outData: Byte array buffer for storing received data

项目 Item	说明 Description
返回 Return	成功：接收到的数据长度（int类型）；失败：-1（表示发送失败） Success: Length of received data (int type); Failure: -1 (indicates sending failure)

三、部分机型串口路径

III. UART Path of Some Models



PDA_72

- UHF: /dev/ttyS1 (A11/A13)



PDA_80

- UHF: /dev/ttyMSM2 (A9)
- UHF: /dev/ttyS0 (A13)
- Fingerprint: /dev/ttyMSM1 (A9)
- Fingerprint: /dev/ttyS0 (A13)



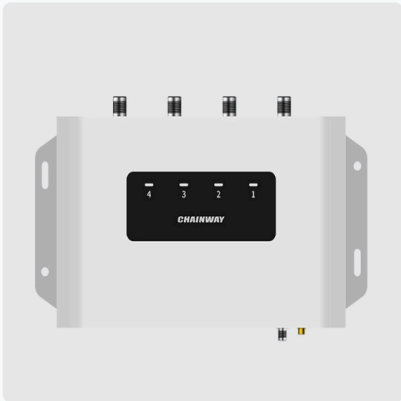
PDA_5

- UHF: /dev/ttyS1 (A11/A13)
- Fingerprint: /dev/ttyS0 (A11/A13)



Reader_4

- Extension: /dev/ttyMSM1 (A9)
- Extension: /dev/ttyHS2 (A13)



Reader_300

- Extension: /dev/ttyS3 (A9)