



A76xx Series Open SDK_UART 开发_应用指导

芯讯通无线科技(上海)有限公司
上海市长宁区临虹路289号3号楼芯讯通总部大楼
电话: 86-21-31575100
技术支持邮箱: support@simcom.com
官网: www.simcom.com

名称:	A76xx Series Open SDK_UART开发_应用指导
版本:	V1.00
类别:	应用文档
状态:	已发布

版权声明

本手册包含芯讯通无线科技（上海）有限公司（简称：芯讯通）的技术信息。除非经芯讯通书面许可，任何单位和个人不得擅自摘抄、复制本手册内容的部分或全部，并不得以任何形式传播，违反者将被追究法律责任。对技术信息涉及的专利、实用新型或者外观设计等知识产权，芯讯通保留一切权利。芯讯通有权在不通知的情况下随时更新本手册的具体内容。

本手册版权属于芯讯通，任何人未经我公司书面同意进行复制、引用或者修改本手册都将承担法律责任。

芯讯通无线科技(上海)有限公司

上海市长宁区临虹路289号3号楼芯讯通总部大楼

电话：86-21-31575100

邮箱：simcom@simcom.com

官网：www.simcom.com

了解更多资料，请点击以下链接：

<http://cn.simcom.com/download/list-230-cn.html>

技术支持，请点击以下链接：

<http://cn.simcom.com/ask/index-cn.html> 或发送邮件至 support@simcom.com

版权所有 © 芯讯通无线科技(上海)有限公司 2023，保留一切权利。

Version History

Version	Date	Owner	What is new
V1.00	2022-11-20		第一版

About this Document

本文档适用于 A1803S open 系列、A1603 open 系列、A1606 open 系列。

SIMCom
Confidential

目录

版权声明.....2

Version History.....3

About this Document.....4

目录.....5

1UART 说明.....6

 1.1 引脚功能复用.....6

2UART API 接口：函数.....8

 2.1 sAPI_UartWrite.....8

 2.2 sAPI_UartRead.....8

 2.3 sAPI_UartWriteString.....8

 2.4 sAPI_UartSetConfig.....9

 2.5 sAPI_UartGetConfig.....9

 2.6 sAPI_UartPrintf.....9

 2.7 sAPI_SendATCMDWaitResp.....9

 2.8 sAPI_UartRxStatus.....10

 2.9 sAPI_UartControl.....10

 2.10sAPI_UartRegisterCallback.....10

 2.11sAPI_UartRegisterCallbackEX.....11

 2.12sAPI_UartRs485DePinAssign.....11

3UART 的配置.....12

1UART 说明

A76xx Series Open 系列 ASR1603S、ASR1803S 和 ASR1606S(带蓝牙模块)支持 3 路串口，分别为 UART1、UART2、UART3，ASR1606S 不带蓝牙的模块支持 4 路串口。UART1 默认配置是主串口，默认显示 Uldemo 界面，可配置为 AT 口，默认波特率 115200；UART2 默认配置是 debug 口，显示开机 log，默认波特率 115200；UART3 作为普通串口，带 gps 的模块，该串口与 gps 相连接，未开放给客户使用，不带 gps 的模块该串口可由用户自主控制，默认波特率 115200。

本文以 A7600C 系列模块为例，相关硬件设计等信息仅供参考；具体硬件设计请参考硬件设计文档。

1.1 引脚功能复用

A7600C 的引脚对应如下：（其他型号的相关引脚设置请参考硬件设计文档）

UART 1 复用关系				
模组引脚名	模组引脚号	默认功能	复用功能 1	复用功能 2
UART1_TXD	71	UART_TXD		
UART1_RXD	68	UART_RXD		

UART 2 复用关系				
模组引脚名	模组引脚号	默认功能	复用功能 1	复用功能 2
DBG_TXD	42	DBG_TXD		
DBG_RXD	106	DBG_RXD		

UART 3 复用关系				
模组引脚名	模组引脚号	默认功能	复用功能 1	复用功能 2

UART2_TXD	50	UART2_TXD	GPIO54	
UART2_RXD	52	UART2_RXD	GPIO53	

SIMCom
Confidential

2UART API 接口：函数

模块支持如下几个串口 API 接口：

串口接口：需要包含文件头：simcom_uart.h

2.1 sAPI_UartWrite

接口：	SC_Uart_Return_Code sAPI_UartWrite(SC_Uart_Port_Number port, UINT8 *data, UINT32 length);
输入：	port: 串口的端口号（SC_UART / SC_UART2 / SC_UART3） data: 需要写的数据地址指针 length: 数据最大长度
输出：	无
返回值：	0:成功 -1:失败
NOTE:	这个接口：用于发送数据到相应的串口上。

2.2 sAPI_UartRead

接口：	int sAPI_UartRead(SC_Uart_Port_Number port, unsigned char *data, int len);
输入：	port: 串口的端口号（SC_UART / SC_UART2 / SC_UART3） data: 串口读取的数据返回的数据地址指针 len: 读取数据的最大长度
输出：	无
返回值：	实际读取到的数据长度
NOTE:	该接口：用于串口读取数据。

2.3 sAPI_UartWriteString

接口：	SC_Uart_Return_Code sAPI_UartWriteString(SC_Uart_Port_Number port, UINT8 *data);
输入：	port: 串口的端口号（SC_UART / SC_UART2 / SC_UART3） data: 需要写的数据地址指针
输出：	无
返回值：	0:成功 -1:失败

NOTE:

这个接口：用于直接发送字符串数据到相应的串口上。

2.4 sAPI_UartSetConfig

接口:	SC_Uart_Return_Code sAPI_UartSetConfig(SC_Uart_Port_Number port, const SCuartConfiguration *config);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) config: 指向 UART 配置块的指针, 此参数属于 SCuartConfiguration 结构类型
输出:	无
返回值:	0:成功 -1:失败
NOTE:	这个接口：用于配置串口的数据帧格式。

2.5 sAPI_UartGetConfig

接口:	SC_Uart_Return_Code sAPI_UartGetConfig(SC_Uart_Port_Number port, SCuartConfiguration *config);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) config: 指向 UART 配置块的指针, 此参数属于 SCuartConfiguration 结构类型
输出:	无
返回值:	0:成功 -1:失败
NOTE:	该接口：用于获取串口的数据帧格式。

2.6 sAPI_UartPrintf

接口:	int sAPI_UartPrintf(const char *fmt, ...);
输入:	需要输出：的格式字符串。
输出:	无
返回值:	字符串的实际长度。
NOTE:	该接口：用于在 debug 口打印字符串。

2.7 sAPI_SendATCMDWaitResp

接口:	int sAPI_SendATCMDWaitResp(int sATPInd, char *in_str, int timeout, char *ok_fmt, int ok_flag, char *err_fmt, char *out_str, int outLen);
输入:	sATPInd: 通道索引, 建议通道 10 inStr: AT 命令的字符串 timeOut: 超时时间

	okFmt: 正确执行时指令的返回格式
	okFlag: 标志
	errFmt: 错误执行时指令的返回格式
	outStr: 返回 AT 命令的字符串
	outLen: 字符串的长度
输出:	无
返回值:	0:成功 否则:失败
NOTE:	该接口: 用于发送/接收内部 AT 命令。

2.8 sAPI_UartRxStatus

接口:	SC_Uart_Rx_Status sAPI_UartRxStatus(SC_Uart_Port_Number port, int timeout);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) timeout: 一个阈值, 如果 Rx 超过阈值并且没有接收到数据, 则 Rx 空闲, 否则 Rx 繁忙。单位为毫秒。
输出:	无
返回值:	0:成功 -1:失败
NOTE:	该接口: 用于获取串口 Rx 的状态。如果 Rx 超过指定时间且未接收到数据, 则 Rx 处于空闲状态, 否则 Rx 处于忙碌状态。

2.9 sAPI_UartControl

接口:	SC_Uart_Return_Code sAPI_UartControl(SC_Uart_Port_Number port, SC_Uart_Control ctl);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) ctl: 控制的类型, 打开或关闭
输出:	无
返回值:	0:成功 -1:失败
NOTE:	该接口: 用于控制串口的打开与关闭。默认情况串口是打开状态, 关闭后会将串口配置为 GPIO 功能并拉低电平。

2.10 sAPI_UartRegisterCallback

接口:	SC_Uart_Return_Code sAPI_UartRegisterCallback(SC_Uart_Port_Number port, SC_Uart_Callback cb);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) cb: 绑定串口的回调函数。
输出:	无

返回值:	0:成功 -1:失败
NOTE:	该接口：用于绑定串口高级中断的回调函数。

2.11 sAPI_UartRegisterCallbackEX

接口:	SC_Uart_Return_Code sAPI_UartRegisterCallbackEX(SC_Uart_Port_Number port, SC_Uart_CallbackEX cb, void *reserve);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) cb: 绑定串口的回调函数 reserve: 保留参数
输出:	无
返回值:	0:成功 -1:失败
NOTE:	该接口：用于为 UART 高级中断绑定回调函数。

2.12 sAPI_UartRs485DePinAssign

接口:	GPIOReturnCode sAPI_UartRs485DePinAssign(SC_Uart_Port_Number port, Module_GPIONumbers GpinNum);
输入:	port: 串口的端口号 (SC_UART / SC_UART2 / SC_UART3) GpinNum: 绑定 Rs485 dePin 的 GPIO
输出:	无
返回值:	0:成功 -1:失败
NOTE:	该接口：用于将 Rs485 dePin 分配给 UART。绑定的 gpio 口在选定的串口发送数据前 500 微秒拉高电平，然后在指定串口发送完成后拉低 gpio 电平。

3UART 的配置

串口的配置在 CP 侧已经是做了初始化配置的,如果客户在 APP 侧想配置串口的参数,请参考 `cus_uart.c`。

SIMCom
Confidential