# Sockets 编程接口

## 简介

本文档介绍了套接字线程的功能、接口以及应用案例。

## 功能

Sockets模块可以实现TCP或UDP服务器的功能，接收来自客户机的信息或向客户机发送信息。

在使用本模块收发数据之前，需先配置套接字类型与协议，服务器IP地址与端口，以及接收与发送数据buff的长度，这些配置由SOCK\_INIT结构体确定。

本模块暂时开放的接口很少，缺乏灵活性，仍需商讨。

本模块具体的使用方式见下面的应用案例。

## 接口

详细介绍创建套接字服务器所需的参数列表以及函数。

### 参数结构体

* **sockets\_handle** : 套接字句柄，指向套接字参数列表结构体
* **SOCK\_OBJ** ：套接字参数列表结构体

参数成员列表如下：

* **SOCK\_PARAM params** ：

创建套接字所需参数，包括服务器IP地址与端口。

* **SOCK\_BUFF\_OBJ sock\_buff**：

套接字接收与发送数据buff，以及接收buff与发送buff的长度。

* **char exit\_ctrl** ：

线程结束控制标志，线程已结束为true，否则为false

* **char sock\_connect\_ctrl**：

创建本地套接字控制，true表示创建，false表示不

* **char sock\_recv\_ctrl** ：

数据接收或发送控制，true表示接收或者发送数据，false表示不

* **char sock\_recv\_snd** ：

发送与接收数据选择；

SOCK\_RECV\_ONLY：此线程仅接收数据

SOCK\_SEND\_ONLY：此线程仅发送数据

SOCK\_RECV\_SEND：此线程既发送数据，又接收数据

* **char sock\_snd\_flag** ：

数据解析是否完成标志，true标志解析完成，并将应答消息传递给套接字线程的发送buff,此时可以启动发送；false表示未解析完成，套接字线程等待，此标志只在SOCK\_SEND\_ONLY、SOCK\_RECV\_SEND有效。

* **SOCK\_OPT opt[SOCK\_OPT\_NUM]**：

套接字选项属性列表；由本模块外部传入，在创建套接字句柄时初始化。

* **int CurrOptNum** ：

套接字选项数目，由模块外部传入，在创建套接字句柄时配置。

* **SOCK\_PARAM** :

创建套接字所需参数列表：

* **int addr\_family ：**

地址类型，其值可为SOCK\_AF\_IPv4、SOCK\_AF\_IPv6等

* **int sock\_type ：**

套接字类型，其值可为SOCK\_UDP\_DGRAM、SOCK\_TCP\_STREAM等

* **int protocol ：**

协议，其值可为SOCK\_PROTO\_TCP、SOCK\_PROTO\_UDP等

* **int AddrIP ：**

二进制IP地址

* **int port ：**

端口号

* **int RecvFlags ：**

接收标志，用于改变接收的默认行为,一般为0

* **int SndFlags ：**

发送标志，用于改变发送的默认行为,一般为0

* **SOCK\_BUFF\_OBJ ：**

套接字接收与发送数据buff

* **void \*RecvBuff ：**

接收数据存放的空间，创建套接字句柄时分配

* **int RecvBuffLen ：**

接收数据存放空间的大小，由模块外部传入

* **int RecvDataLen ：**

从客户端接收到的当前数据的大小，

显然**RecvDataLen不能大于RecvBuffLen。**

* **void \*SndBuff ：**

发送数据存放的空间，创建套接字句柄时分配

* **int SndBuffLen ：**

发送数据存放空间的大小，由模块外部传入

* **int SndDataLen ：**

此次发送给客户端的数据大小，

显然**SndDataLen不能大于SndBuffLen。**

* SOCK\_INIT ：

用于外部初始化，此结构体内有需由外部配置的套接字线程参数。

* **SOCK\_PARAM params ：**

套接字的类型、协议、IP地址以及端口号等需由外部配置。具体配置值见下。

* **int OptList[SOCK\_OPT\_NUM]：**

套接字选项列表，包含本套接字线程需设置的套接字选项

* **int CurrOptNum ：**

需设置的套接字选项数目

* **int RecvBuffLen ：**

由外部配置接收数据buff的最大长度

* **int SndBuffLen ：**

由外部配置发送数据buff的最大长度

* **char sock\_recv\_snd ：**

决定此次套接字线程是仅接收数据、仅发送数据还是收发数据均有。

### 接口函数

* 函数原型：**sockets\_handle sock\_server\_create(SOCK\_INIT \*SockInit)**

函数功能：创建套接字句柄，套接字线程内部所需的buff以及参数均在此函数内配置。

输入参数：**SockInit ：**

指向初始化结构体，由模块外配置的套接字线程参数值均在此列

返回结果：返回指向套接字参数列表结构体的句柄

* 函数原型：**pthread\_t sock\_server\_start(sockets\_handle h,int priority)**

函数功能：启动套接字线程

输入参数：**h** : 套接字句柄；**priority ：**优先级

返回结果：返回线程ID

* 函数原型：**void sock\_server\_delete(sockets\_handle h)**

函数功能：关闭套接字线程，释放创建时分配的内存空间

输入参数：**h** : 套接字句柄

返回结果：无

* 函数原型：**void sock\_snd\_flag\_set(sockets\_handle h,char state)**

函数功能：配置数据解析完成标志

输入参数：**h** :套接字句柄；**state**：标志状态，其值为**true/false**

返回结果：无

* 函数原型：**char sock\_snd\_flag\_get(sockets\_handle h)**

函数功能：查询数据解析是否完成的标志

输入参数：**h** : 套接字句柄

返回结果：返回数据解析完成状态，**true:**解析完成；**false：**未完成

## 参数配置表

* Sockets 地址类型（**addr\_family**），映射表如下：

|  |  |
| --- | --- |
| Suma地址类型 | Sockets 地址类型 |
| **SOCK\_AF\_IPv4** | AF\_INET |
| **SOCK\_AF\_IPv6** | AF\_INET6 |
| **SOCK\_AF\_UNIX** | AF\_UNIX |
| **SOCK\_AF\_UNSPEC** | AF\_UNSPEC |

* Sockets 类型（**sock\_type**），映射表如下

|  |  |
| --- | --- |
| Suma套接字类型 | Sockets 类型 |
| **SOCK\_UDP\_DGRAM** | SOCK\_DGRAM |
| **SOCK\_TCP\_STREAM** | SOCK\_STREAM |
| **SOCK\_IP\_API** | SOCK\_RAW |

* Sockets 协议（**protocol**），映射表如下：

|  |  |
| --- | --- |
| Suma 协议 | Sockets协议 |
| **SOCK\_PROTO\_TCP** | IPPROTO\_TCP |
| **SOCK\_PROTO\_UDP** | IPPROTO\_UDP |
| **SOCK\_PROTO\_IPv4** | IPPROTO\_IP |
| **SOCK\_PROTO\_IPv6** | IPPROTO\_IPv6 |
| **SOCK\_PROTO\_UNSPEC** | 0 |

* Sockets 选项（**OptList列表内容**），映射表如下：

|  |  |  |
| --- | --- | --- |
| Suma 套接字选项 | Sockets 选项 | 类型及含义 |
| **SOCK\_OPT\_LISTEN** | SO\_ACCEPTCONN | int,是否能被监听，仅供查询 |
| **SOCK\_OPT\_BROADCAST** | SO\_BROADCAST | int,广播数据包 |
| **SOCK\_OPT\_DEBUG** | SO\_DEBUG | int,启用网络驱动调试功能 |
| **SOCK\_OPT\_DONTROUTE** | SO\_DONTROUTE | int,绕过通常路由 |
| **SOCK\_OPT\_ERROR** | SO\_ERROR | int,返回挂起的套接字错误并清除 |
| **SOCK\_OPT\_KEEPALIVE** | SO\_KEEPALIVE | int,启用周期性keep-alive消息 |
| **SOCK\_OPT\_LINGER** | SO\_LINGER | struct linger,未发消息并且套接字关闭时延迟时间 |
| **SOCK\_OPT\_OOBINLINE** | SO\_OOBINLINE | int,将带外数据放到普通数据中 |
| **SOCK\_OPT\_RCVBUF** | SO\_RCVBUF | int,以字节为单位的接收缓冲区大小 |
| **SOCK\_OPT\_RCVLOWAT** | SO\_RCVLOWAT | int,接收调用中返回的以字节为单位的最小数据量 |
| **SOCK\_OPT\_RCVTIMEO** | SO\_RCVTIMEO | struct timeval,接收调用的超时值 |
| **SOCK\_OPT\_REUSEADDR** | SO\_REUSEADDR | int,重用bind的地址 |
| **SOCK\_OPT\_SNDBUF** | SO\_SNDBUF | int,以字节为单位的发送缓冲区大小 |
| **SOCK\_OPT\_SNDLOWAT** | SO\_SNDLOWAT | int,发送调用中以字节为单位的发送的最小数据量 |
| **SOCK\_OPT\_SNDTIMEO** | SO\_SNDTIMEO | struct timeval，发送调用的超时值 |
| **SOCK\_OPT\_TYPE** | SO\_TYPE | int,标识套接字类型，仅供查询 |

## 应用案例

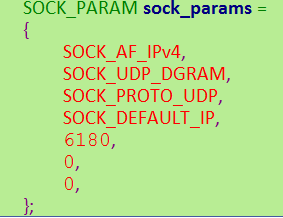
UDP控制协议案例：

此案例是基于UDP套接字的服务器，接收来自上位机发送的UDP控制协议信息，解析后按UDP控制协议标准返回相应的应答信息，但此案例未能实现参数设置与查询，因而返回的应答消息均为错误帧格式。UDP控制协议的内容见《9550-串口控制协议》。

上位机发送消息格式参考《串口控制定义及举例-9550A》。

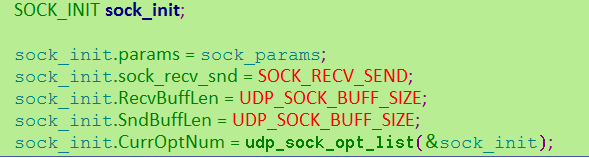
1. 套接字参数列表配置值，如下：

可以看到此处说明创建的套接字属于UDP数据包套接字，并且配置了服务器IP地址以及套接字端口。



1. 配置套接字线程参数，如下：

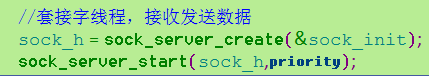
此结构体会传入到创建套接字句柄的函数内，用来配置套接字线程参数。



下图是套接字列表注册：



1. 创建套接字句柄，启动套接字线程，收发数据



1. 创建udp控制协议句柄，启动udp解析线程，并将解析结果保存到套接字线程的发送buff。

