深圳市精锐达电子科技有限公司

以下描述，均已C/C++格式组织数据，小端模式，1字节对齐

报文 = 报头 + 正文

报头格式只有一个，正文格式可以很多个，方便扩展。

uint8_t 表示无符号8位整数，uint16_t 表示无符号16位整数

以下16进制表示为0xXX，十进制没有0x前缀

一，报头

```
typedef struct _CmdHead
{
  uint8_t  cmd;
  uint16_t cmd_index;
  uint8_t  cmd_option;
  uint16_t cmd_len;
  uint8_t  data_checksum;
} CmdHead;
```

C/C++格式，小端对齐

参数：cmd 说明，目前支持以下命令

```
#define   CMD_GET_IO_OUT_VALUE          1
#define   CMD_GET_IO_IN_VALUE           2
#define   CMD_SET_IO_OUT_VALUE          3
#define   CMD_GET_IO_NAME               4
#define   CMD_SET_IO_NAME               5
#define   CMD_GET_TIMING_INFO           6
#define   CMD_SET_TIMING_INFO           7
#define   CMD_GET_RTC_VALUE             8
#define   CMD_SET_RTC_VALUE             9
#define   CMD_SET_TIMING_ON_MSK         10
#define   CMD_GET_TIMING_ON_MSK         11
#define   CMD_SET_INPUT_CTL_ON_MSK      12
#define   CMD_GET_INPUT_CTL_ON_MSK      13
#define   CMD_SET_SIG_IO_SOME_BIT       36
#define   CMD_SET_IP_CONFIG             33
#define   CMD_GET_IP_CONFIG             34
#define   CMD_GET_INPUT_CTL_MODE_INDEX  40
#define   CMD_SET_INPUT_CTL_MODE_INDEX  41
```

参数：cmd_index，命令编号，一般从 0 开始递增。

参数：cmd_option，命令选项，表示报文是否正确，或者表示报文状态

参数：cmd_len，正文长度，字节为单位

参数：data_checksum，正文校验和,正文数据累加的结果,由于走的是 TCP 协议，可以忽略此项，以 0 填充

二，正文

1，设置继电器输出状态，命令号：CMD_SET_IO_OUT_VALUE

```
typedef struct _CmdIoValue
{
  uint8_t    io_count;
  uint8_t    io_value[4];
} CmdIoValue;
```

参数：io_count，IO 的数量

参数：io_value[4]，IO 的状态值,1 表示闭合，0 表示打开

例如：发送开和关命令到继电器输出口，

0x03,index_lsb,index_hsb,0x00,0x05,0x00,0x00,0x08,output,0x00,0x00,0x00

Index_lsb,index_hsb 为 命令序号，从 0 开始计数

Output 为 输出开关的位变量

举例:打开第 2 位，其他关闭

0x3,0x25,0x0,0x0,0x5,0x0,0x0,0x8,0x2,0x0,0x0,0x0

通过 TCP 接口发送到下位机就好了，下位机收到命令后会返回一个报文，应答报文和此发送出去的报文相似：（以下列举描述报头和正文的关键参数）

参数：cmd_index，应答和发送的命令编号必须一致，否则错误。

参数：cmd_option，应答选项值低 4 位为 CMD_ACK_OK（此值为 1），表示成功，为 0 表示失败。

参数：io_value[4]，表示下位机当前的 IO 状态值

2， 获取输入端状态：命令号：CMD_GET_IO_IN_VALUE

发送端直接发送报头即可，正文空白，下位机应答一报文，报文格式与 CMD_SET_IO_OUT_VALUE 类似

例如：发送 0x04,0x12,0x34,0x0,0x0,0x0,0x0

3， 获取输出端（继电器）状态，命令号：CMD_GET_IO_OUT_VALUE

此命令与上一条命令类似，只需发送报头即可，正文空白

例如：发送 0x01,0x12,0x35,0x0,0x0,0x0,0x0

4， 触发某一位，命令号：CMD_SET_SIG_IO_SOME_BIT

```
typedef struct _CmdIoIndex
{
  uint8_t    io_msk[4];
} CmdIobitmap;
```

触发模式有5种，触发打开，触发关闭，和触发翻转，还有单触发模式，具体行为根据设置的模式而定。

例如：

0x24,index_lsb,index_hsb,0x0,0x4,0x0,0x0,bit_msk,0x0,0x0,0x0

Index_lsb,index_hsb 为 命令序列号，从0开始计数

bit_msk 为 输出开关位号

举例：触发第4位

0x24,0x6f,0x0,0x0,0x4,0x0,0x0,0x8,0x0,0x0,0x0

应答报文与请求报文相似，应答报文的参数io_msk[4]，表示下位机IO口的当前状态。

5， 设置 IP 地址，命令号：CMD_SET_IP_CONFIG

```
typedef struct _CmdIpConfigData
{
  uint8_t   ipaddr[4];
  uint8_t   netmask[4];
  uint8_t   gateway[4];
  uint8_t   dns[4];
  uint16_t  port;
  uint16_t  webport;
} CmdIpConfigData;
```

应答数据只含报头，根据选项：cmd_option, CMD_ACK_OK(1)表示执行成功, CMD_ACK_KO(0)表示执行失败。

请求报文包含正文，把需要设置的地址参数填入数据结构即可。比如我们常用的内网 IP 号码为 192.168.1.250，则 ipaddr[0] = 192,ipaddr[1] = 168,ipaddr[2]=1,ipaddr[3] = 250

以下类推.

6， 查询 IP 地址，命令号：CMD_GET_IP_CONFIG

只需发送报头即可，无需正文。

返回数据则包含

```
typedef struct _CmdIpConfigData
```

正文数据，含义参照上一命令.