

获取和修改设备开关状态的指令

2018 年 8 月 28 日修订

以下指令使用 TCP socket 调试软件以十六进制模式发送

为方便调试，请关闭自动上报功能：

关闭上报功能，发送指令：00 00 00 00 00 06 01 05 1D 5C 00 00

开启上报功能，发送指令：00 00 00 00 00 06 01 05 1D 5C FF 00

线圈表示输出的通道

1, 修改某一路开或关

发送格式

00	00	00	00	00	06	01	05	02	线圈	开关	00
----	----	----	----	----	----	----	----	----	----	----	----

00 00 00 00 00 06 01 05 02 线圈起始地址 开关标记 (FF 或 00) 结尾 00

例如：

开第一路： 00 00 00 00 00 06 01 05 02 00 FF 00

关第一路： 00 00 00 00 00 06 01 05 02 00 00 00

开第二路： 00 00 00 00 00 06 01 05 02 01 FF 00

关第二路： 00 00 00 00 00 06 01 05 02 01 00 00

红色的部分为输出的地址，以十六进制表示，00 表示第一路，01 表示第二路，02 表示第三路，0A 表示第十一路，0B 表示第十二路，0F 表示第 16 路，10 表示第 17 路，以此类推。

蓝色部分表示开关标记，FF 表示开，00 表示关。

2, 读取某几路状态

与第 5 条指令作用相同，读取的数量可变

发送格式

00	00	00	00	00	06	01	01	02	线圈	00	数量
----	----	----	----	----	----	----	----	----	----	----	----

00 00 00 00 00 06 01 01 02 线圈起始地址 00 读取数量

返回格式：

00 00 00 00 00 [数据总数+3] 01 01 [数据总数] [数据第一字节] [数据第二字节] [数据第三字节].....

例如：

从第一路开始，读取 10 路： 00 00 00 00 00 06 01 01 02 00 00 0A

红色的部分为需要读取的首地址，以十六进制表示，00 表示第一路，01 表示第二路，02 表示第三路，0A 表示第十一路，0B 表示第十二路，0F 表示第 16 路，10 表示第 17 路，以此类推。

蓝色的部分为读取的总共数量，以十六进制表示，00 表示读取 0 路，01 表示读取 1 路，02 表示读取 2 路，0A 表示读取 10 路，0B 表示读取 11 路，0F 表示读取 15 路，10 表示读取 16 路，以此类推。

设备返回应答 00 00 00 00 00 05 01 01 02 ff 03

返回应答红色部分 02 表示读取到 2 个字节数据，紧跟着的 FF 03 表示线圈状态，使用十六

进制表示，读取到的线圈状态可能是多个字节组成，例如 AA BB CC DD ... 每个字节最多表示 8 路，两个字节表示 16 路，3 个字节表示 24 路，以此类推。

3，同时修改 32 路状态

发送格式

03	01	02	00	00	05	00	20	AA	BB	CC	DD
----	----	----	----	----	----	----	----	----	----	----	----

03 01 02 00 00 05 00 20 AA BB CC DD

蓝色部分 AA BB CC DD 一共 4 个字节，使用十六进制表示输出的状态值，每个字节有 8 个比特，每个比特表示一路输出，4 个字节总共表示 32 路输出，从第 1 路开始，到第 32 路。

例如：

32 路全开：03 01 02 00 00 05 00 20 FF FF FF FF

32 路全关：03 01 02 00 00 05 00 20 00 00 00 00

开第 1、2 路，关第 3、4 路：03 01 02 00 00 05 00 20 03 00 00 00

关第 1、2 路，开第 3、4 路：03 01 02 00 00 05 00 20 0C 00 00 00

4，同时取反 32 路状态，翻转 32 路状态

发送格式 24 01 02 00 00 05 00 AA BB CC DD

蓝色部分 AA BB CC DD 一共 4 个字节，使用十六进制表示二进制，每个字节有 8 个比特，每个比特表示一路输出，4 个字节总共表示 32 路输出，从第 1 路开始，到第 32 路。

例如翻转第一路：24 01 02 00 00 05 00 01 00 00 00

例如翻转第二路：24 01 02 00 00 05 00 02 00 00 00

例如翻转第三路：24 01 02 00 00 05 00 04 00 00 00

例如翻转第 1、3、5、6 路：24 01 02 00 00 05 00 35 00 00 00

5，读取 32 路状态

与第 2 条指令作用相同，只是数量固定 32 路

发送格式

01	01	02	00	00	05	00	20	00	00	00	00
----	----	----	----	----	----	----	----	----	----	----	----

01 01 02 00 00 05 00 20 00 00 00 00

返回数据：

01 01 02 01 05 00 00 XX AA BB CC DD

XX 部分忽略

蓝色部分 AA BB CC DD 一共 4 个字节，使用十六进制表示输出的状态值，每个字节有 8 个比特，每个比特表示一路输出，4 个字节总共表示 32 路输出，从第 1 路开始，到第 32 路。

例如：

返回 32 路全开状态：01 01 02 01 05 00 00 10 FF FF FF FF

返回 32 路全关状态：01 01 02 01 05 00 00 10 00 00 00 00