

Trigger 接口使用说明

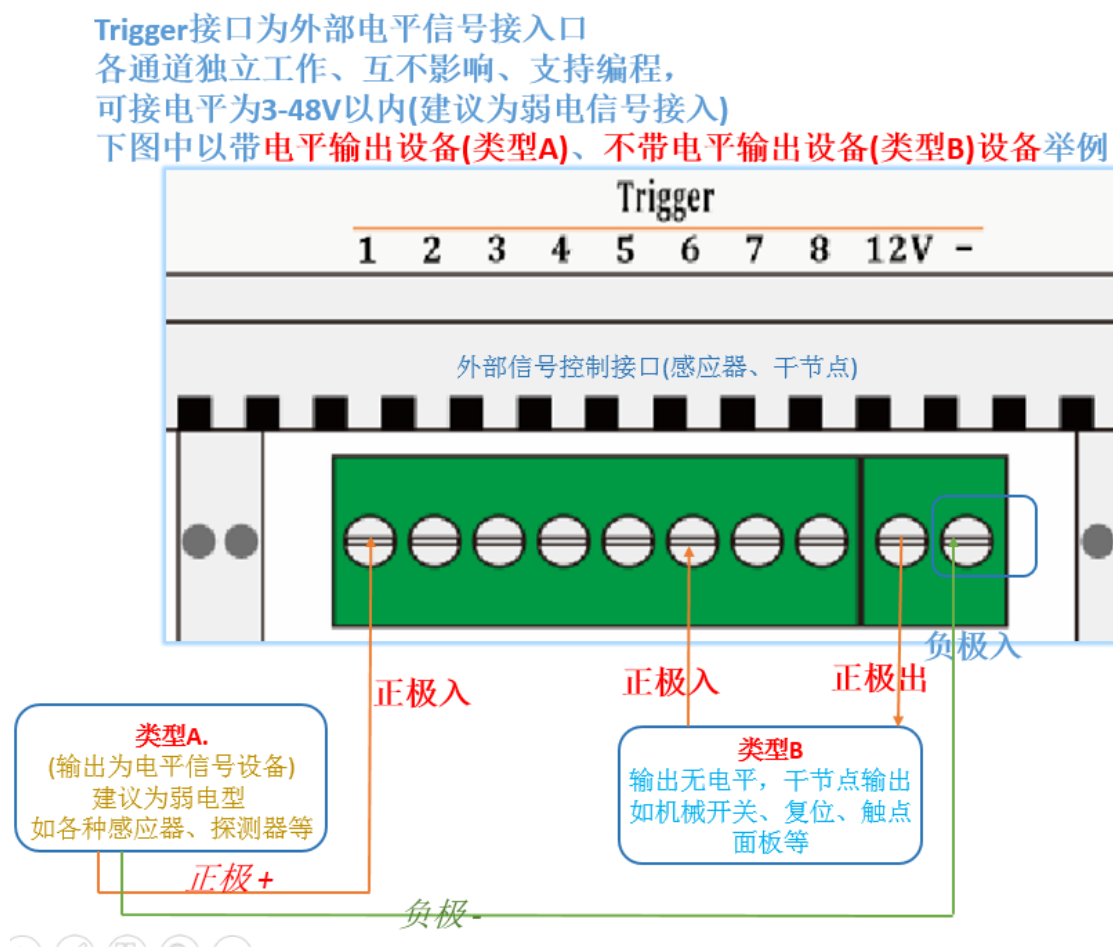
- 接口功能

创思特各模块上的 Trigger 接口功能一致，可检测电平信号为 3-48V，为公地型输入
可接机械开关、触点面板、感应器、等干节点或者电平信号，系统将信号经过处理后，不仅能轻松实现对自身模块功能的调用，还可通过总线发出自定义指令，实现对总线上任意模块的控制，如总线上加入串口服务器等设备，可对 Trigger 接口进行编程实现对串口服务器上设备的直接控制。

- 支持的触发事件

1. 仅按下(接通)
2. 仅松开(断开)
3. 按下(接通)及松开(断开)
4. 长按 3 秒及松开
5. 每按住 1.5 秒及松开

- 接线示例



- 功能示例

复位/机械开关接入 Trigger 简单设置后可实现对灯光控制或者使指令发送至接入总线的第三方中控或受控设备，实现控制目的。

感应器、探测器、消防信号等接入 Trigger 接口可根据需要进行设置，如有消防信号触发，关闭电器设备，打开应急灯。如感应器触发后自动将单路/多路灯光调节至合适亮度。

设置 Trigger 接口协议

- **串口通信**

波特率 19200,数据位 8,停止位 1,无校验

1. 485 类型模块接线方式

A 为 + ----- 接控制器 A(+)
B 为 - ----- 接控制器 B(-)

2. 232 类型模块接线方式

TX 为设备往外发送针脚 ----- 接控制器(PC) RX
RX 为设备接收外部命令针脚 ----- 接控制器(PC) TX
- 为公共地 ----- 接控制器(PC) GND

- **控制指令**

指令均为 ASCII 码，不区分大小写，且无回车换行。

设备具有超强处理能力，1 秒冷启动。支持连续收码及快速处理,控制无需延时发码

1. 启用或者禁用缓存(如总线上模块较多，建议启用缓存机制，可防止数据冲突)

setSYS(id,5,isOn)

设置设备号为 12 的模块启用缓存机制(设备号为 0 将设置总线上所有模块)

setSYS(12,5,1)

设置设备号为 12 的模块禁用缓存机制(设备号为 0 将设置总线上所有模块)

setSYS(12,5,0)

2. 设置地址指令(如用 0 代替现有地址，需先将该模块连接到其他模块的总线断开，以免误将其他模块地址改变)

setSYS(old_id,1,new_id)

如将设备号为 1 的模块设备号改为 2.

setSYS(1,1,2)

如总线上模块设备号改为 2.(旧设备用 0 代替，但防止误操作请其他模块从总线上断开)

setSYS(0,1,2)

查询设备地址(查看该台模块地址,模块需已禁用缓存机制)

getSYS(0,1)

模块返回信息如下 **msgSYS(0,1,12)** 代表该模块地址为 12

3. 根据模块序列号设置地址 (模块需已禁用缓存机制,按下模块 Link 按钮,模块会将自身信息从总线接口发出)

setSYS(0,7,id,"00122F3CDDE3EB")

如模块发出 SN:"00122F3CDDE3EB " 代表该模块序列号为"00122F3CDDE3EB "

将该模块地址设置为 6 指令为

setSYS(0,7,6,"00122F3CDDE3EB")

4. 启用 Trigger 接口反馈系统代码功能(该协议适用于串口服务器)

set_RS(id,30,isOn)

如设置设备号为 12 的模块启用自定义代码功能

set_RS(12,30,1)

如设置设备号为 12 的模块禁用自定义代码功能

`set_RS(12,30,0)`

查询命令 `get_RS(id,30)`

5. 启用 Trigger 接口动作模式 (该协议适用于串口服务器)

`set_RS(id,25,CH,clickType)`

`id`=设备号 默认为0 `id=0` 时将控制所有总线上模块

`CH`=需要控制的通道 `CH=0` 时为全部通道控制

`clickType` =1 代表仅执行松开(断开)代码

`clickType` =2 代表执行按下、松开两种代码

`clickType` =3 代表仅执行按下(接通)代码

`clickType` =4 代表禁用按下事件，启用长按事件，按下 3 秒后，发出按下代码，松开时执行对应指令

`clickType` =5 代表禁用按下事件，启用长按事件，每按住 1.5 秒重复发出按下代码，松开时执行对应指令

`clickType` =6 代表禁用自定义代码，此时功能与面板上 8 个按键一致

如设置 12 号设备 Trigger 2 口只执行按下代码

`set_RS(12,25,2,3)`

查询命令 `get_RS(id,25,ch)`

6. 设置 Trigger 接口触发后需调用的模式

`set_RS(id,27,CH,clickType,{ModeArr})`

`id`=设备号 默认为0 `id=0` 时将控制所有总线上模块

`CH`=需要控制的通道 `CH=0` 时为全部通道控制

`clickType` =0 代表设置按下与松开模式

`clickType` =1 代表仅设置按下模式

`clickType` =2 代表仅设置松开模式

`ModeArr` 为需调用的模式组(1-8 个，多个用“,”逗号隔开)

如设置 12 号设备 Trigger 2 口设置按下调用模式 8

`set_RS(12,27,2,1,{8})`

如设置 12 号设备 Trigger 2 口设置按下调用模式 1,3,5,7

`set_RS(12,27,2,1,{1,3,5,7})`

查询命令 `get_RS(id,27,ch,clickType)`

7. 设置模式存档 ASICC 格式代码(临时指定波特率及校验位)

脱离中控或者是做应急

`set_RS(id,22,ModelID,CH, baudID, parity, "DATAS")`

`id`=设备号 默认为0

`ModelID`=需要存档的模式编号

`CH`=需要控制的通道

`baudID` =为波特率类型 0 代表默认, 1 为

2400, 2 为 4800, 3 为 9600, 4 为 19200, 5 为

38400, 6 为 57600, 7 为 115200

`parity`=校验位 0:无校验 1:奇校验 2:偶校验

`DATAS` 为需要存储的数据

例如 1 号地址模块存储场景 2, 数据为从第 7 路发送数据 Power on(波特率 19200 无校验)

`set_RS(1,22,2,7,4,0,"Power on")`

查询命令 `get_RS(id,22,modelID)`

8. 设置模式存档 16 进制格式代码(临时指定波特率及校验位)

`set_RS(id,23,ModelID,CH, baudID, parity, "DATAS")`

`id`=设备号 默认为0

ModelID=需要存档的模式编号

CH=需要控制的通道

baudID =为波特率类型 0 代表默认, 1 为 2400, 2 为 4800, 3 为 9600, 4 为 19200, 5 为 38400, 6 为 57600, 7 为 115200

parity=校验位 0:无校验 1:奇校验 2:偶校验

DATAS 为需要存储的数据

例如 1 号地址模块存储场景 2, 内容为第 7 路发送数据 A9172E0000003F9A (波特率 38400 偶校验)

set_RS(1,23,2,7,5,2,"A9172E0000003F9A")

查询命令 get_RS(id,23,modelID)

9. 通过指令调用已存储的模式

set_RS(id,24,ModelID)

id=设备号 默认为0

ModelID=需要调用模式的编号

例如 1 号地址模块调用场景 2, 将已存档的命令发出;

set_RS(1,24,2)

查询命令 get_RS(id,24,modelID)

10. 通过指令调用 Trigger 接口事件

get_RS(id,29,CH, clickType)

id=设备号 默认为0

clickType =1 代表需调用按下事件对应模式

clickType =2 代表需调用松开事件对应模式

例如 1 号地址模块调用 Trigger 2 口按下事件, 将已存档的命令发出;

get_RS(1,29,2, 1)

更多详细说明请上官方网站查阅 www.crethite.com

淘宝企业店铺(经销商折扣请联系客服)

<https://shop155265173.taobao.com/?spm=2013.1.0.0.x1GetC>

关注公众号
可直接购买

