创思特调光模块说明

- **模块供电**

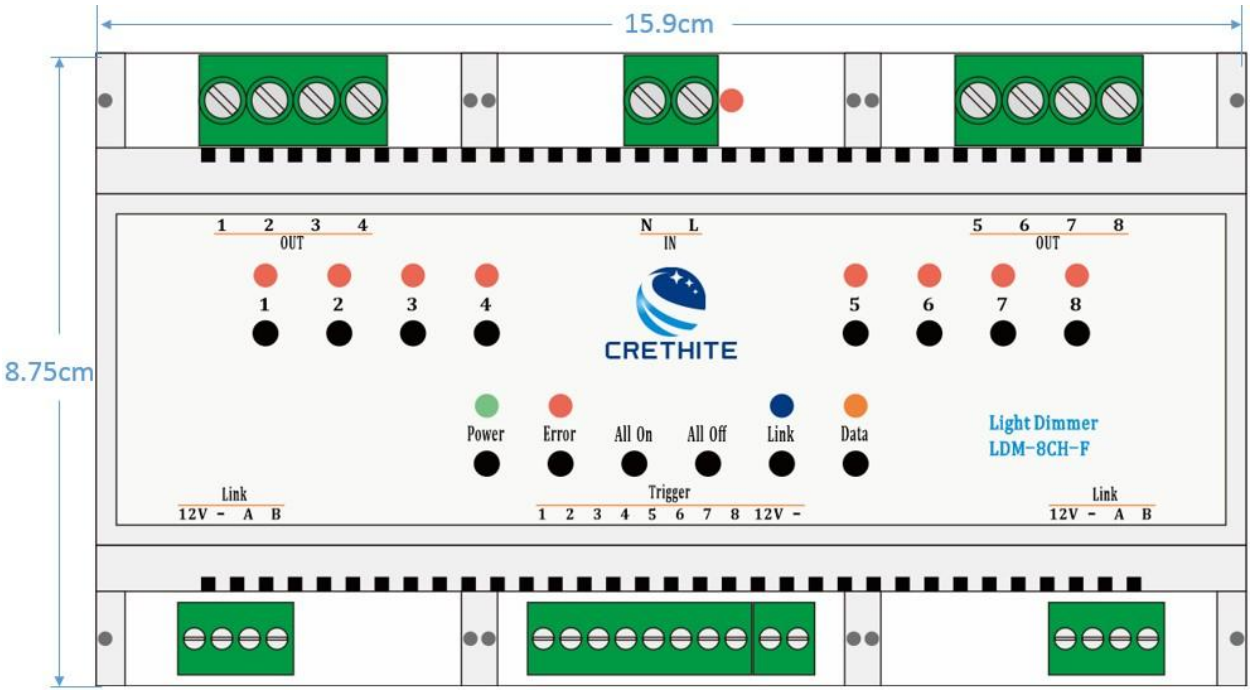
  需选用 12V 1A 以上电源

- **安装方式及尺寸**

  标准导轨式安装

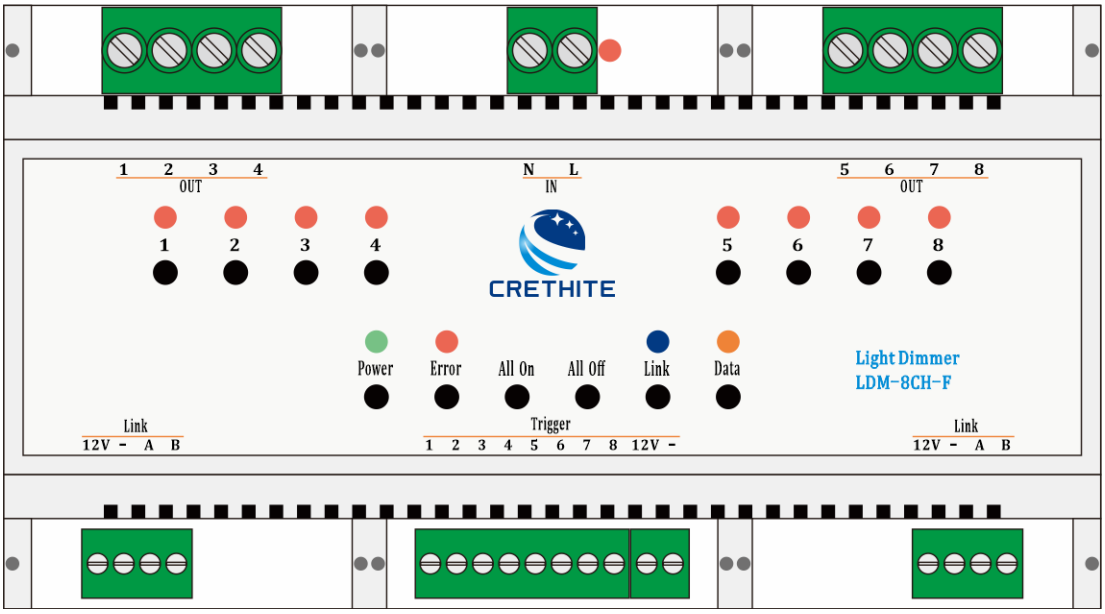
  尺寸 15.9cmx8.75cm

  外壳 PCV 防火材质

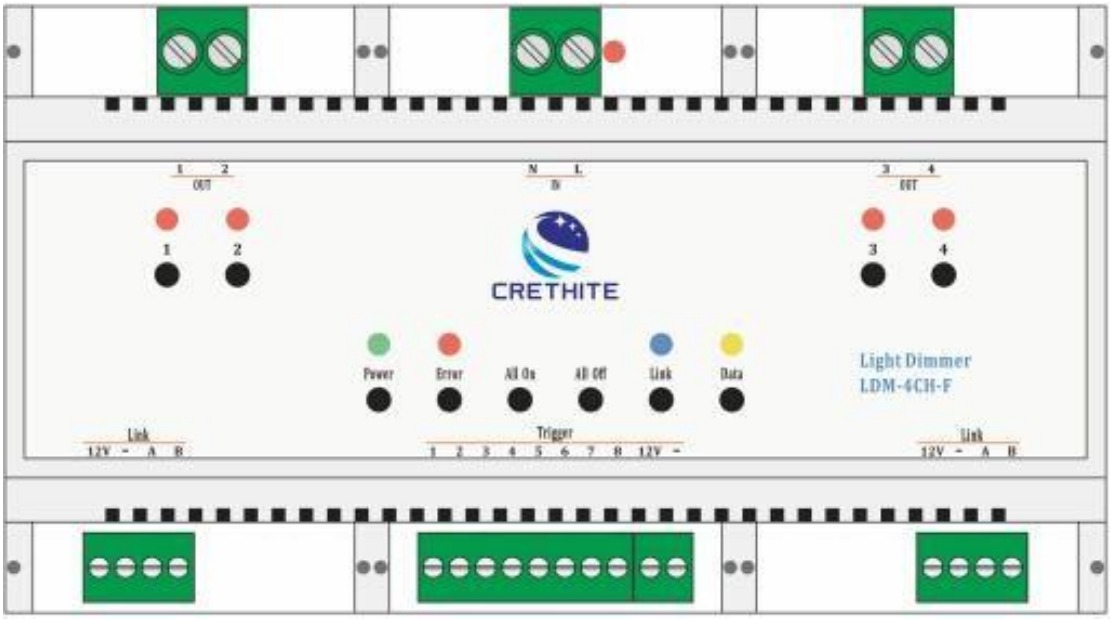


- **型号及功能**

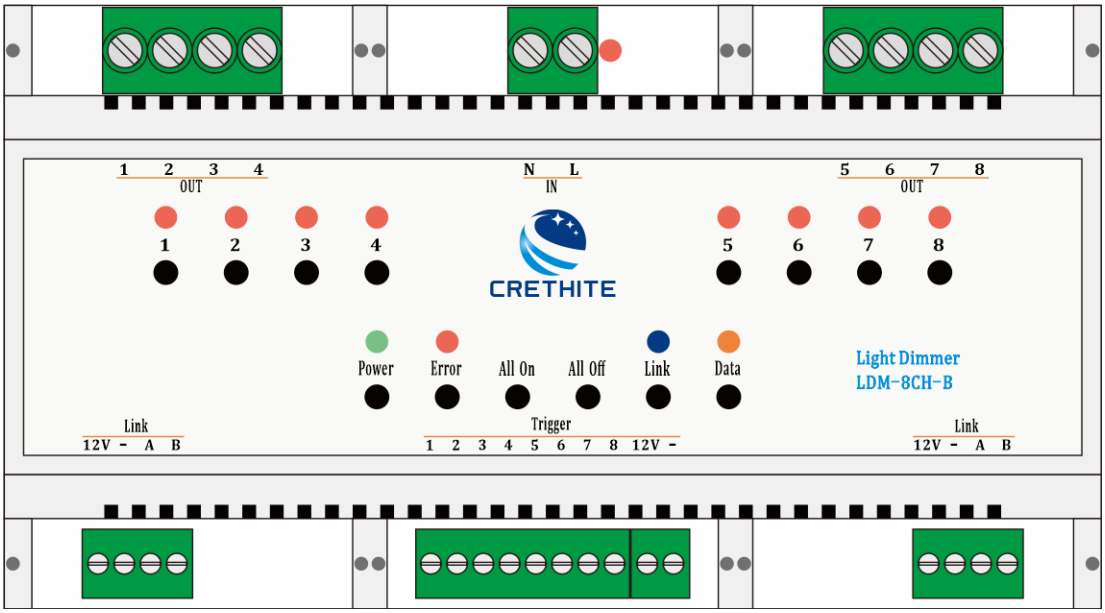
1. LDM-8CH-F 8 路前沿调光模块,单路最大支持 5A,总功率不超过 10A



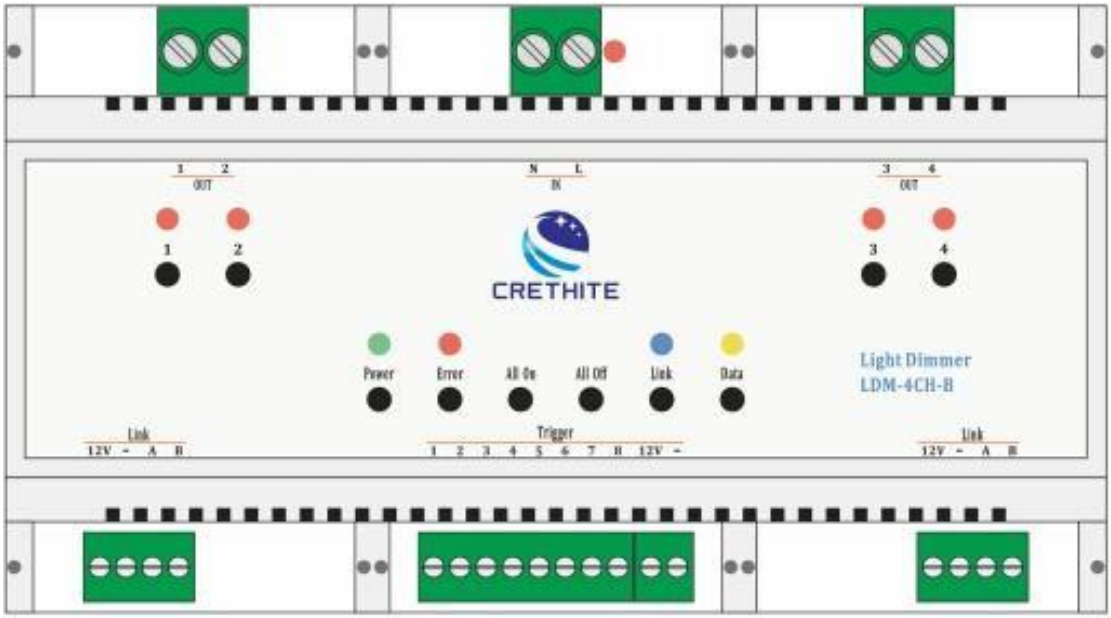
2. LDM-4CH-F 4 路前沿调光模块，单路最大支持 8A,总功率不超过 10A



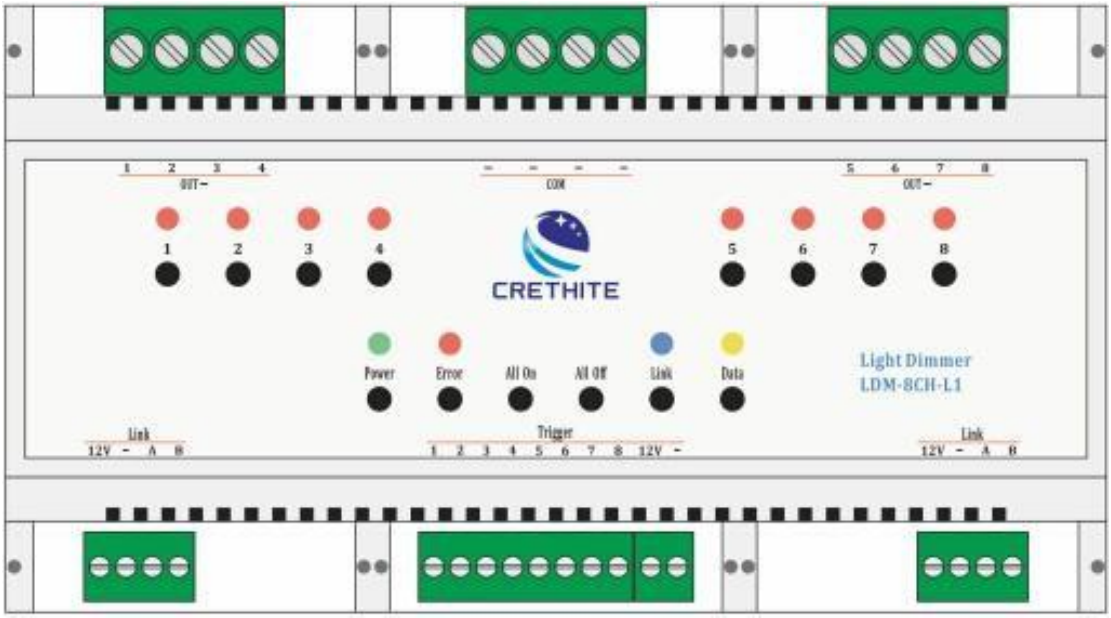
3. LDM-8CH-B 8 路后沿调光模块，单路最大支持 5A,总功率不超过 10A



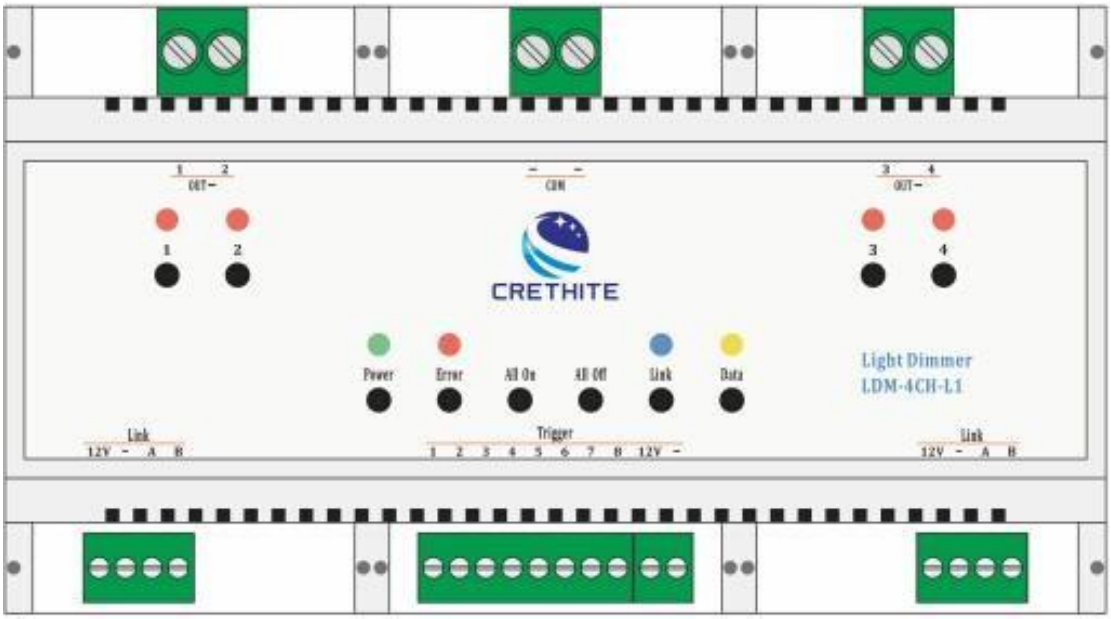
4. LDM-4CH-B 4 路后沿调光模块，单路最大支持 8A,总功率不超过 10A



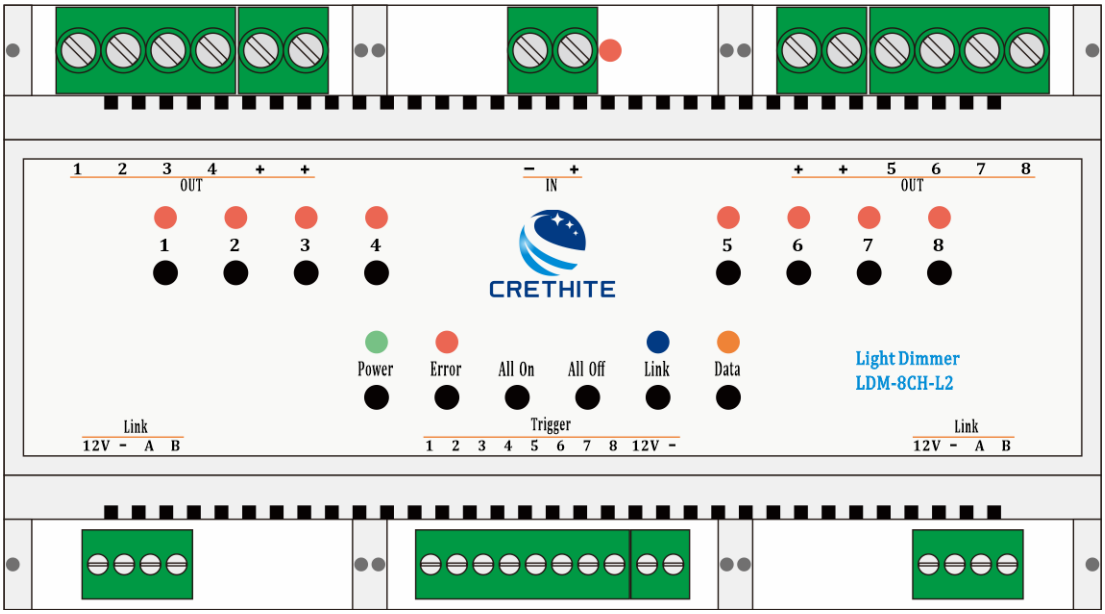
5. LDM-8CH-L1 8 路 LED 调光模块,单路最大支持 5A,总功率不超过 25A



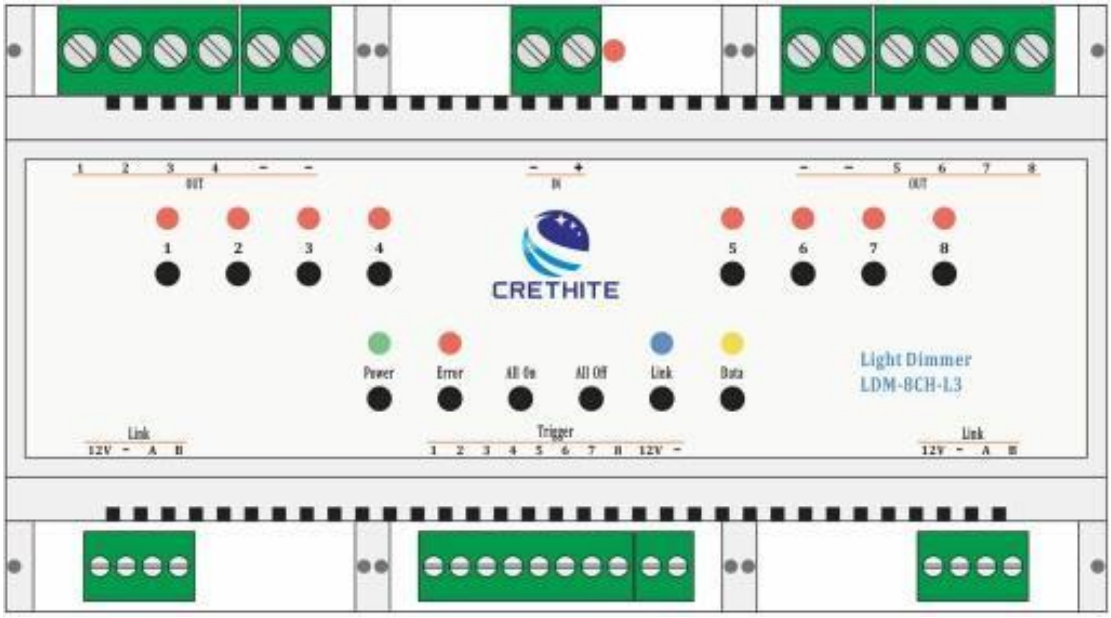
6. LDM-4CH-L1 4 路 LED 调光模块,单路最大支持 15A,总功率不超过 25A



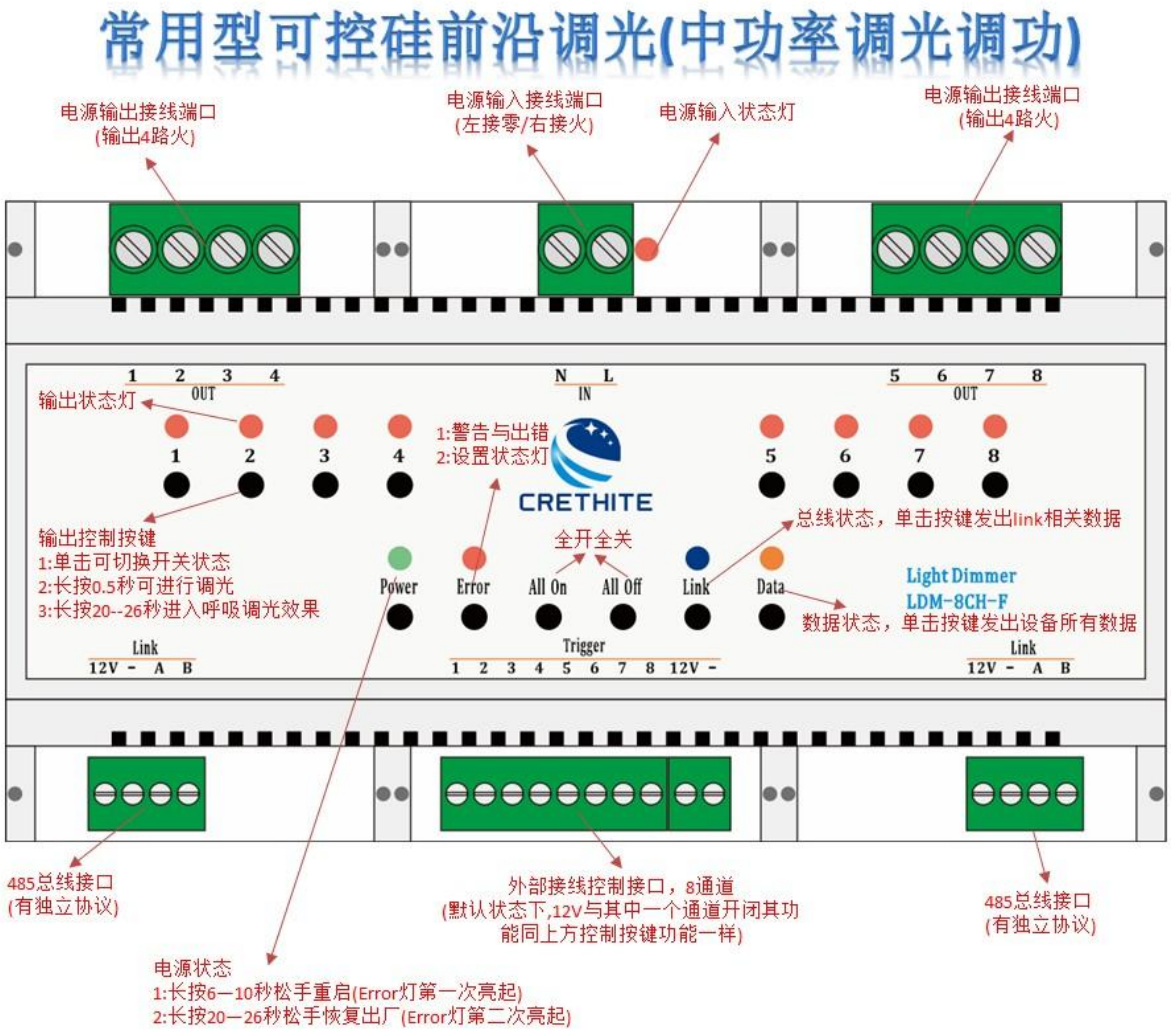
7. LDM-8CH-L2 8 路 LED 调光模块,单路最大支持 5A,总功率不超过 25A



8. LDM-8CH-L3 8 路 LED 调光模块,单路最大支持 5A,总功率不超过 25A

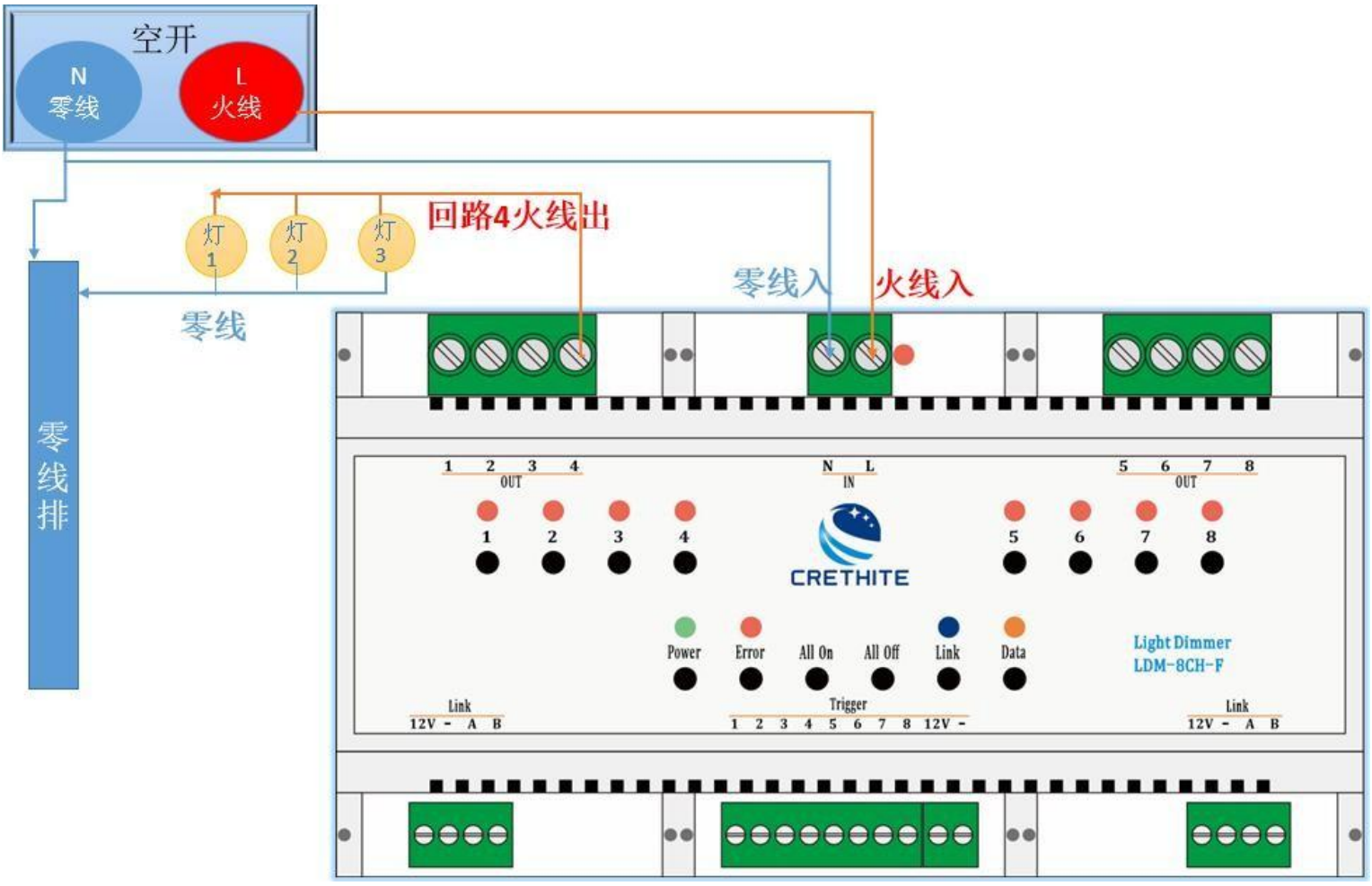


面板功能说明



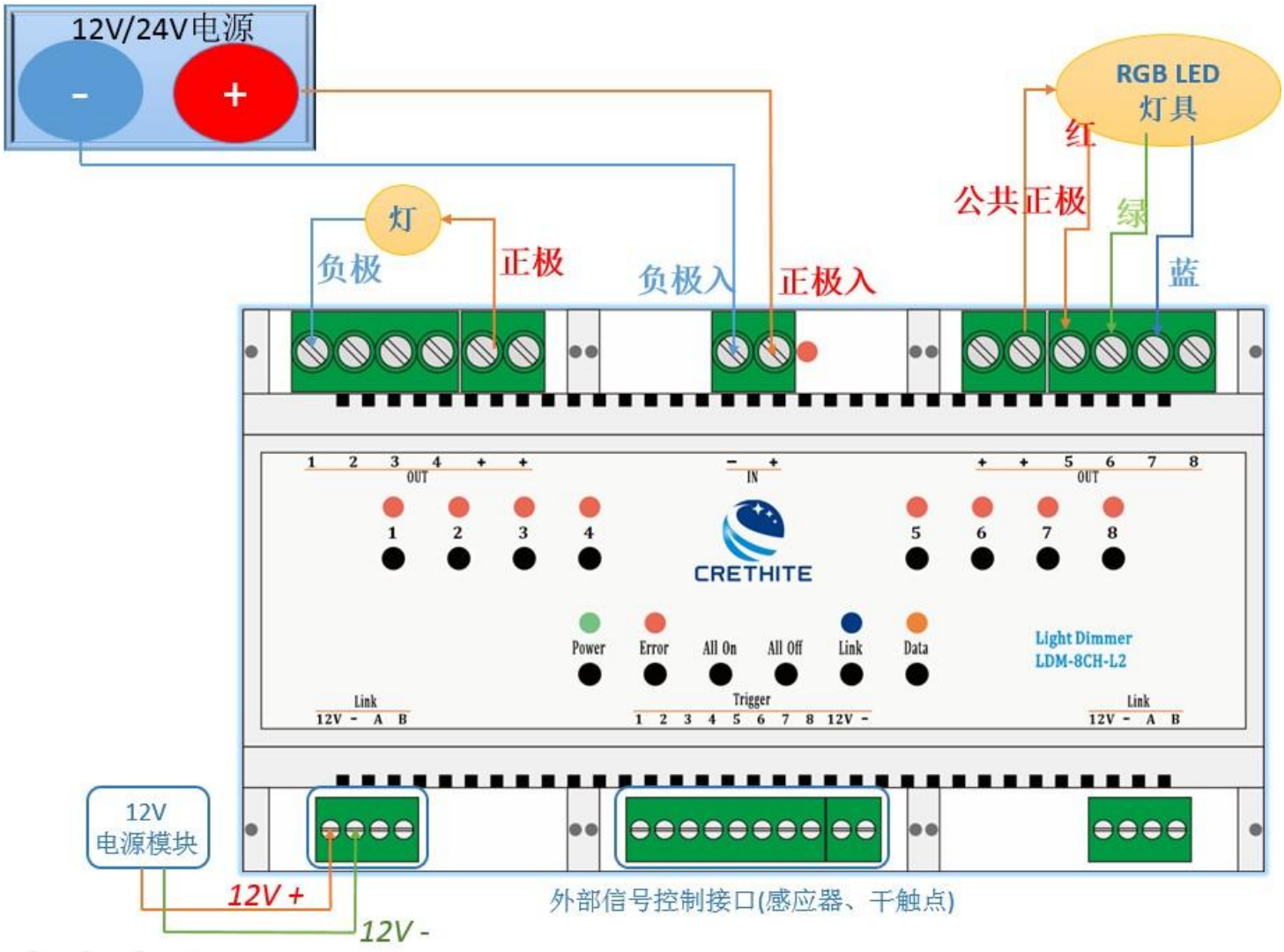
接线示意图

1. LDM-8CH-F 灯光控制
   前沿可控硅调光器



2. LDM-8CH-L2 LED 调光模块接线图

12V/24V 共阳型调光器



控制协议

- **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)

   查询指令 getDev(id,5)

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)

   a) 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. 通道类型设置,将该路通道设置为非调光或者调光类型(适用于不可调设备接入调光模块)

   设置为非调光模式后，该路调光控制失效，命令值大于 0，该通道为最亮状态

   setDev (0,151,CH,isRelay)

   isRelay=0 为调光类型

   isRelay=1 为继电器类型

   例如将设备号为 0 的模块第八路设置为不可调光类型指令为

   setDev (0,151,8,1)

5. **亮度调节指令(不带渐变效果)**

   setDev(id,1,CH,VAL)

   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*

   VAL=亮度值 0-255 0 为灭 255 为最亮

   例如将设备号为 0 的模块第八路调节至 **180** 命令为

   setDev (0,1,**8**,**180**)

   例如将设备号为 0 的模块第八路调节至 **0** 命令为

   setDev (0,1,**8**,**0**)

   查询指令 getDev(id,1,CH)

6. **亮度调节指令(带渐变效果)**

setDev(id,101,CH,VAL,SECOND)

SECOND 以秒为单位，值为 0s-255s

例如将设备号为 0 的模块第八路调节至 180 历时 3 秒命令为

setDev (0,101,8,180, 3)

7. 同时调节多个通道(统一亮度)

setDev (id,7,{ch1,ch2}, VAL)

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，多路用“,” 逗号分开*

VAL=是否开关 开为 1 关为 0

例如调节设备号为 0 的模块第 5,6 路亮度为 150 命令

setDev (0,7,{5,6},150)

例如设备号为 0 的模块第 5,6 路亮度为 0 命令

setDev (0, 7,{5,6},0)

8. 同时调节多个通道(不同亮度)

setDev (id,8,{ch1,ch2},{ VAL1,VAL2})

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，多路用“,” 逗号分开*

VAL=对应通道的亮度值 0-255

例如调节设备号为 0 的模块第 5,6 路亮度方便为 150,255 命令

setDev (0,8,{5,6},{150,255})

例如设备号为 0 的模块第 5,6 路亮度为 50,0 命令

setDev (0,8,{5,6},{50,0})

9. 指定通道亮度加 X(不带渐变效果)

setDev (id,38,CH,X,TPYE)

TPYE=1 时 如亮度超过 255，则会变为最低亮度 0

TPYE=3 时 如亮度超过 255，则会保持 255 例如设备号为 0 的模块第 5 路亮度增加 10 命令

setDev (0,38,5,10,3)

10. 指定通道亮度减 X(不带渐变效果)

setDev (id,38,CH,X,TPYE)

TPYE=0 时 如亮度已经为 0，则会变为最高亮度 255

TPYE=2 时 如亮度已经为 0，则会保持 0 例如设备号为 0 的模块第 5 路亮度减少 10 命令

setDev (0,38,5,10,2)

11. 指定通道亮度加 X(带渐变效果)

setDev (id,138,CH,X,TPYE, SECOND)

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*

X=增加多少亮度

SECOND 以秒为单位，值为 0s-255s

TPYE=1 时 如亮度超过 255，则会变为最低亮度 0

TPYE=3 时 如亮度超过 255，则会保持 255

例如设备号为 0 的模块第 5 路亮度在 1 秒内增加 10 命令

setDev (0,138,5,10,3,1)

12. 指定通道亮度减 X(带渐变效果)

setDev (id,138,CH,X,TPYE, SECOND)

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*

X=增加多少亮度

SECOND 以秒为单位，值为 0s-255s

TPYE=0 时 如亮度已经为 0，则会变为最高亮度 255

TPYE=2 时 如亮度已经为 0，则会保持 0

例如设备号为 0 的模块第 5 路亮度在 1 秒内减少 10 命令

setDev (0,138,5,10,2,1)

13. 置反命令

setDev (id,4,CH)

例如将设备号为 0 的模块第 2 路置为相反状态，如之前为开发码后该路会关闭否则打开

setDev (0,4,2)

14. 延时自动控制,基于组合命令，各组合命令直接延时 0.5 秒执行。

如开 1 秒再关闭，则应该设置为打开 2 次 然后关闭，打开两次时间过去 0.5 秒然后再关闭，则效果为打开 1 秒后关闭。

setDev (0,3,{CH, CH },{255, 0 })

例如开 2 路 0.5 秒后关闭

setDev (0,3,{2,2},{255, 0 })

打开 2 路 1 秒后关闭

setDev (0,3,{2,2,2},{255,255,0 })

打开 3 路 3 秒后关闭

setDev (0,3,{3,3,3,3,3,3,3},{255,255,255,255,255,255,0 }

如不清楚设备号，可打开串口后，连通模块，按下 Link 键，此时设备将发出自身信息。

15. 读取通道状态

getDev(id,1,ch)

例如读取 12 设备第 3 通道状态

getDev (12,1,3)

模块返回 msgPWM(12, 1, 3, 255) 12 号模块第 3 通道为最亮状态

16. 设置模块上电状态

setDev(id,13, state)

state =1 为通电全关(默认)

state =2 为通电全开

state =3 为通电后恢复上次断电状态

state =4 为通电自定义状态(参考指令设置)

例如将 12 号 设备上电模式设置为恢复断电前状态

setDev (12,13,3)

如需将模块上电状态设置为自定义则还需设置用户自定义状态

设置自定义上电状态指令

setDev(id,14,{ch1, ch2 , ch3 , ch4 , ch5 , ch6 , ch7 , ch8 })

ch1-ch8 分别为 8 路的上电状态

例如将 12 号设备上电状态设置为 1, 3, 5, 7 路开, 2, 4, 6, 8 关则指令为

setDev(12,14,{1, 0 , 1 , 0 , 1 , 0 , 1 , 0 })

查询指令 getDev(id,14)

17. 场景存储(将模块当前所有通道状态记录并存储)

setDev(id,23,mode)

mode 为场景编号取值范围 0-100

例如将 12 设备当前通道状态存储为场景 3

setDev(12,23,3)

18. 场景存储(将用户自定义状态存储)

setDev(id,22,mode,{val_1, val_2, val_3, val_4, val_5, val_6, val_7, val_8})

mode 为场景编号取值范围 0-100

val_x 为对应通道的状态

val_x=0 为关

val_x=4-255 均为开

val_x=2 为当前场景不操作该通道

val_x=3 为当前场景下将该通道状态置反

例如将 12 设备场景 2 设置为 通道 1,3 开;2,4 关;5,6,7,8 不被操作

setDev(12,22,2,{255,0,255,0,2,2,2,2})

例如将 12 设备场景 3 设置为 通道 1,3 亮度 150(继电器则为开);2,4 关;5,6,7,8 切换为相反状态

setDev(12,22,3,{150,0,150,0,3,3,3,3})

查询指令 getDev(id,22)

19. 调用场景

setDev(id,24,mode)

例如调用 12 号 设备第 2 场景

setDev (12,24,2)

Trigger 感应接口设置等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