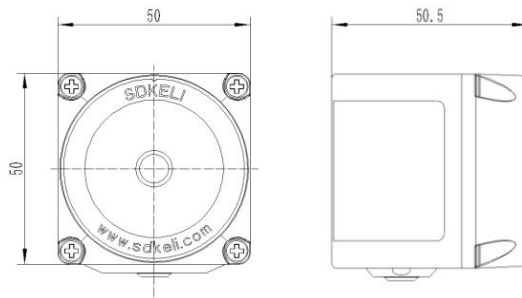


LD 单点激光雷达-测量型
使用说明书
(2026 年 1 月)

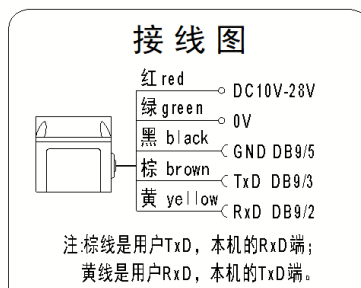


济宁科力光电产业有限责任公司

2、外形尺寸 (mm)

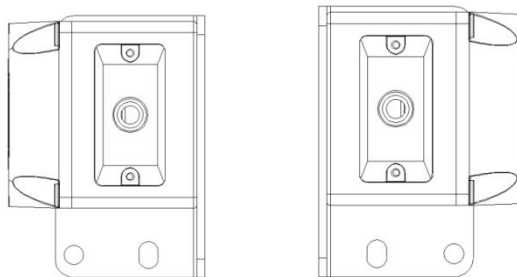


3、接线图



注意
雷达通过转接头与电脑连接时, 必须与转接头共地 (即雷达黑色线需连接到转接头的GND端), 否则, 会造成数据传输失败。

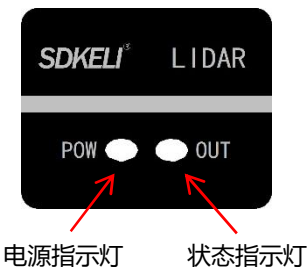
4、安装方式



(左安装)

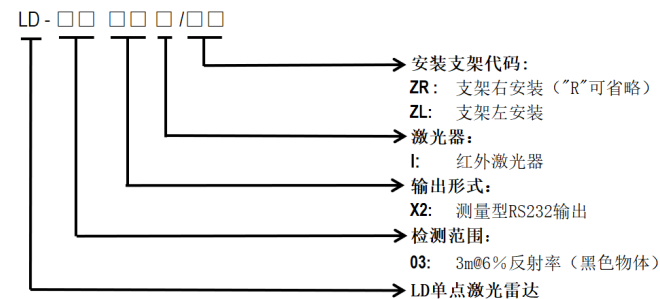
(右安装)

5、外观信息和指示标识



电源指示灯 状态指示灯

6、编号规则



7、通讯协议

7.1、数据输出格式

LD 单点激光雷达通过 RS232 接口周期性输出测量距离数据, 每次输出 3 个字节, 编号 A、B 和 C, 其中前两个字节 A、B 最高位为 0, 最后一个字节 C 最高位为 1, 有效数据位剩余 21 位。在剩余 21 位中, A6、A5 和 A4 为校验位, A3、A 和 A1 恒为 0 (预留扩展功能), 其余 A0、B6...B0、C6...C0 表示为距离, 范围 0~32767mm。

1、技术参数

光学特性	
激光光源	波长 905nm, 一类激光产品
检测范围	3 米@6%反射率黑色物体 10 米@90%反射率白色物体
激光出口光斑大小	5mm
重复精度	8mm@500Hz
距离误差	±20mm
温漂误差	0.25mm/°C
电气/机械参数	
工作电压	DC10V~DC28V
上电启动时间	典型值 3s
功耗	小于 3W
外形尺寸	50mm×50mm×50.5mm
电缆长度	2m
输出形式	RS232 (115200, 8, N, 1)

7.2、 校验算法

```
const unsigned char check_bit_table[256] =
{
    0, 1, 1, 2, 1, 2, 2, 3, 1, 2, 2, 3, 2, 3, 3, 4,
    1, 2, 2, 3, 2, 3, 3, 4, 2, 3, 3, 4, 3, 4, 4, 5,
    1, 2, 2, 3, 2, 3, 3, 4, 2, 3, 3, 4, 3, 4, 4, 5,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    1, 2, 2, 3, 2, 3, 3, 4, 2, 3, 3, 4, 3, 4, 4, 5,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    3, 4, 4, 5, 4, 5, 5, 6, 4, 5, 5, 6, 5, 6, 6, 7,
    1, 2, 2, 3, 2, 3, 3, 4, 2, 3, 3, 4, 3, 4, 4, 5,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    3, 4, 4, 5, 4, 5, 5, 6, 4, 5, 5, 6, 5, 6, 6, 7,
    2, 3, 3, 4, 3, 4, 4, 5, 3, 4, 4, 5, 4, 5, 5, 6,
    3, 4, 4, 5, 4, 5, 5, 6, 4, 5, 5, 6, 5, 6, 6, 7,
    3, 4, 4, 5, 4, 5, 5, 6, 4, 5, 5, 6, 5, 6, 6, 7,
    4, 5, 5, 6, 5, 6, 6, 7, 5, 6, 6, 7, 6, 7, 7, 8,
};

unsigned char GetCrcPackage3Byte(unsigned char *buf)
{
    return (check_bit_table[buf[1]] + check_bit_table[buf[2]]) &
    0x07;
}
```

7.3、 数据解析

```
/*
buf 为一个数据包指针，依次存放 A、B 和 C 三个字节，
返回距离值，校验不对返回-1
*/
int GetLD_Distance(unsigned char *buf)
{
    int distance;
    unsigned char cal_crc = GetCrcPackage3Byte(buf);
    unsigned char rcv_crc = (buf[0]>>4)&0x07;
    if(cal_crc!= rcv_crc)
    {
        return -1;
    }
    /* 计算距离， A0、 B6...B0、 C6...C0 */
    distance =
    ((buf[0]&0x1)<<14)+(buf[1]&0x7F)<<7)+((buf[2]&0x7F));
    return distance;
}
```