

**OidEncoder 欧艾迪®**



# 拉线位移传感器

CANopen 使用说明书 V3.08

深圳欧艾迪科技有限公司  
<https://oidencoder.com>

# 目录

(点击对应目录可跳转)

一、产品特点.....	2
二、应用领域.....	2
三、产品型号说明.....	3
四、型号精度.....	4
五、长度换算方法.....	4
六、拉绳位移传感器规格参数概览.....	5
七、拉绳位移传感器规格参数概览.....	6
八、产品尺寸图.....	7
九、工作原理.....	13
十、拉线位移传感器安装注意事项.....	14
十一、售后服务.....	14
附录一：CANopen 通信协议.....	15
联系我们.....	27



## 一、产品特点

- 拉线位移传感器可以完成几乎任何线性行程测量任务，即使在空间狭小或者环境条件苛刻的情况下，也能轻松胜任，是线性定位和行程位移反馈的理想解决方案。
- 多种测量长度，多种输出方式：Modbus RTU、RS485、CANbus、RS232、CANopen 等数字信号输出或绝对值编码输出，增量式脉冲输出（AB 相或者 ABZ 相脉冲方波），电阻输出（5K 或者 10K），电压输出（0-5V 或者 0-10V），4-20mA 电流输出，均可实现稳定的高精度测量，性价比高。
- 外形紧凑，拉线盒与编码器可以随意搭配，无需任何调节辅助工具，便可轻松安装，模块化设计，方便替换和安装，可节约维护时间，减少维护成本。
- 使用寿命长，运行次数高达 500 万次。
- 精度高，线性精度 $\pm 0.1\%$ ，重复精度 $\pm 0.01\%$ 。
- 采用6061高强度铝合金材料，强度高，能够确保极低的磨损，同时防静电防干扰，能够确保传感器在恶劣环境中安全运行。
- 软性多股不锈钢绳，外层尼龙涂层，耐磨耐腐蚀防锈且柔软。
- 完全可以替代光栅尺。
- 工业级专业防护接口，最高可达防雷级，传感器防护等级可达 IP68 防尘防水，户外可放心使用。

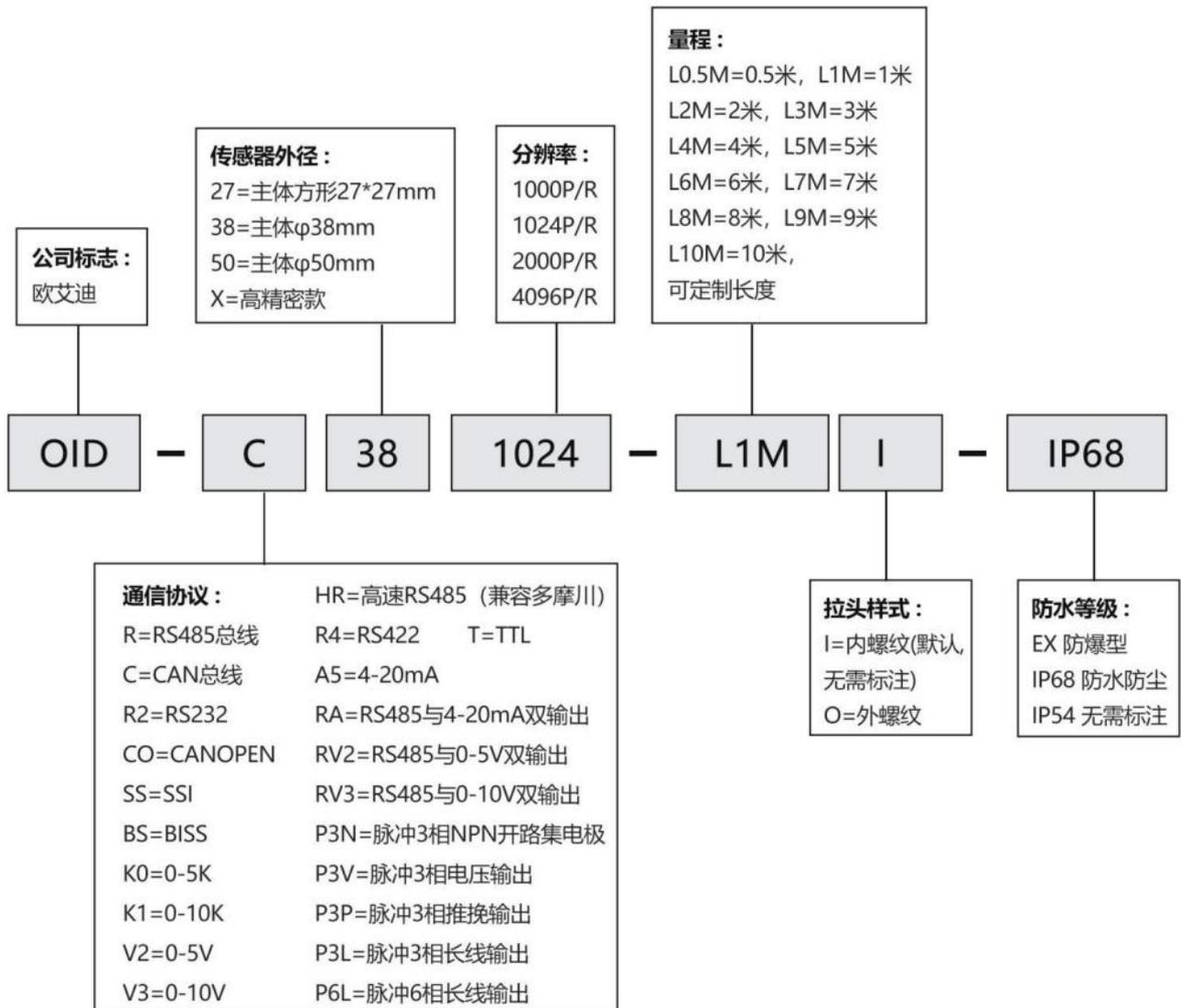


## 二、应用领域

适用于：液压油缸行程检测，闸门开度检测及控制，吊车提升机检测，自动仓储检测，木工机械检测，试验机检测，大包装机械，木工机械，压力机械，仓储位置定位，造纸机械，纺织机械，金属板材机械，印刷机械，建筑机械，水平控制仪，高度机等相关尺寸测量和位置控制，工业机械，自动化控制等。

### 三、产品型号说明

#### 欧艾迪拉绳位移传感器型号定义V3.08



**例:** OID-CO384096-L2M-IP68 表示 CANOPEN 输出，传感器主体外径 38mm，分辨率 10 位（1024P/R），拉绳量程 2M 的拉线位移传感器，IP68 防尘防水。



## 四、型号精度

量程 mm	拉线盒主体尺寸(mm)	内轮毂周长	输出方式	线性精度	分辨率 (P/R)	每个脉冲所代表的量程	防护等级
微型 IP68 防水 500mm	27*27*57	60mm	CANopen	0.10%	10bit(1024P)	0.059mm	IP68
					12bit(4096P)	0.015mm	
500mm	49*49*40	100mm		0.10%	10bit(1024P)	0.098mm	IP54/ IP68
					12bit(4096P)	0.024mm	
1000mm	49*49*40	100mm		0.10%	10bit(1024P)	0.098mm	IP54/ IP68
					12bit(4096P)	0.024mm	
2000mm	63*63*40	150mm		0.10%	10bit(1024P)	0.146mm	IP54/ IP68
					12bit(4096P)	0.037mm	
3000mm	79*79*40	200mm		0.10%	10bit(1024P)	0.195mm	IP54/ IP68
					12bit(4096P)	0.049mm	
4000mm	95*95*40	250mm		0.10%	10bit(1024P)	0.244mm	IP54/ IP68
					12bit(4096P)	0.061mm	
5000mm	95*95*40	250mm		0.10%	10bit(1024P)	0.244mm	IP54/ IP68
					12bit(4096P)	0.061mm	
6000mm	95*95*62	225mm	0.10%	10bit(1024P)	0.22mm	IP54/ IP68	
				12bit(4096P)	0.055mm		
7000mm	95*95*62	225mm	0.10%	10bit(1024P)	0.22mm	IP54/ IP68	
				12bit(4096P)	0.055mm		
8000mm	120*120*111	340mm	0.15%	12bit(4096P)	0.083mm	IP54/ IP68	
9000mm	120*120*111	340mm	0.15%	12bit(4096P)	0.083mm		
10000mm	120*120*111	340mm	0.15%	12bit(4096P)	0.083mm		



## 五、长度换算方法

拉绳位移传感器长度计算可以参考：

1. 先确认拉出拉绳时数据是递增的（方向可自由设置）
2. 举例在拉绳未拉出时设置当前位置值为 1000 (基准，也可另设其他值)

则：公式  $L = (X - 1000) * \text{轮周长} / \text{分辨率}$ （单位 mm）



## 六、拉绳位移传感器规格参数概览

规格参数			
电 压	DC5V~24V 宽电压	波特率	50K~1M (默认 500K);
工作电流	100mA	内核刷新周期	50uS
线性度	0.1%	电气寿命	>100000h
站号地址	1~127(默认 1)	通信协议	请见附录一: CANOPEN 通信
分辨率	1024(10bit)、4096(12bit)		

性能参数			
使用寿命	大于 500 万次	拉绳材质	线径 0.8mm, 软性多股不锈钢绳, 外层尼龙涂层
功率	2W	外壳材质	铝合金, 拉头不锈钢
拉力	2~3N	表面处理	阳极氧化, 防静电干扰
最大拉线速度	1m/s	线缆长度	标配 1-1.2M (可定制)
操作温度	-40~+85°C	防护等级	IP54 常规/ IP68 防水防尘/ Ex 防爆型

环境参数	IP54	IP68 防水防尘	防爆 Ex
工作温度	-40 ~ + 85°C	-40 ~ + 85°C	-40 ~ + 85°C
储存温度	-40 ~ + 85 °C	-40 ~ + 85 °C	-40 ~ + 85 °C
湿度	98 % (无凝露)	98 % (无凝露)	98 % (无凝露)
防护等级	外壳: IP54; 轴/轴承:IP65	整体 IP68	本安防爆
抗震动	10G	10G	10G
抗冲击	50G	50G	50G



## 七、拉绳位移传感器规格参数概览

接线方式 (CANOPEN)	
红线	电源正极 5~24V
黑线	地线 (0V)
黄线	置零 (ZR)
绿线	CANH
白线	CANL



5264 插头

### 7.1 出厂编码器已设置基本参数，可以直接使用（不需要设置，相关设置线悬空并做好绝缘。）

CANOPEN 编码器出厂时的默认配置，波特率默认为 500K；

波特率可配置范围 10Kbps-1Mbps，编码器默认通信地址（站号）为 1。

### 7.2 功能线使用方法

#### 7.2.1 置零功能:

方法一：上电后，黄线接触一下地线(黑线)，编码器置零；

方法二：发送指令置零，详情请参考附件一通信协议部分的置零指令实例。

#### 7.2.2 恢复出厂设置功能:

断电后，把黄线接到黑线上，上电，保持 2 分钟，掉电，去掉黄线悬空，重新上电。

#### 7.2.3 黄线不用时，请悬空并做好绝缘。

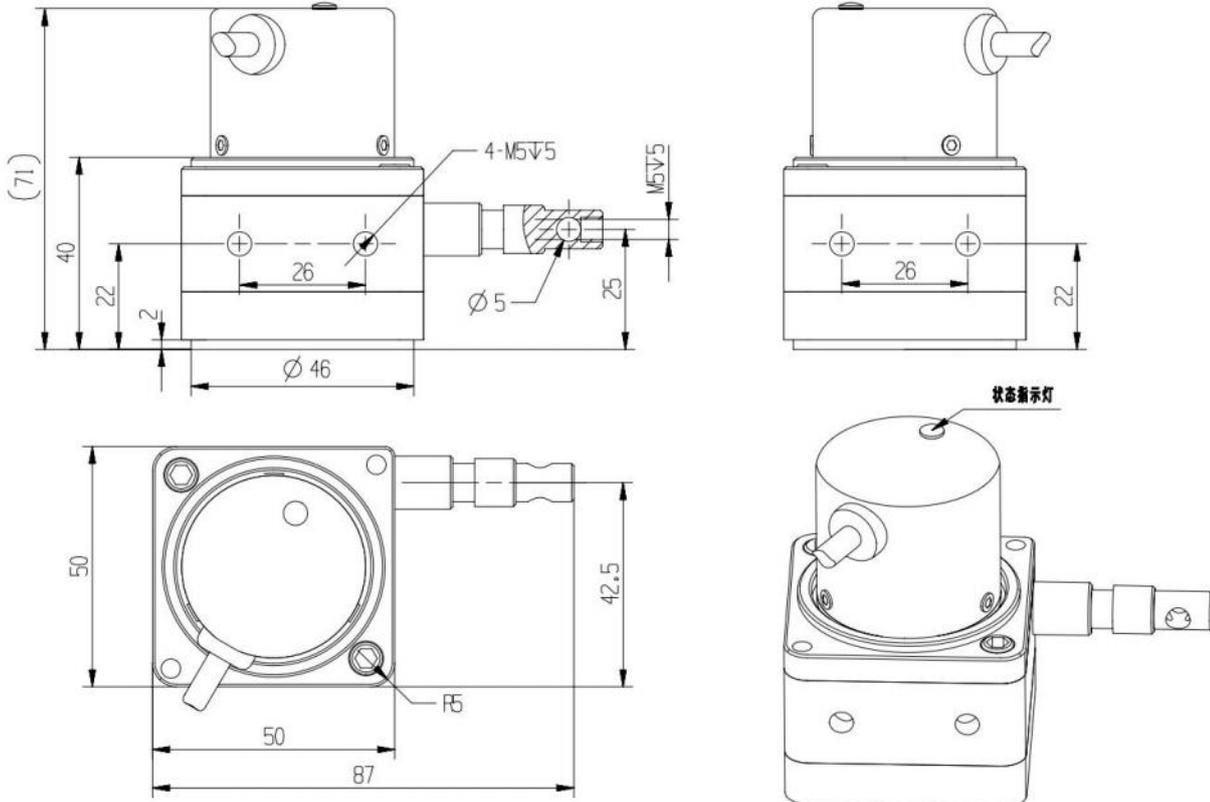
### 7.3 编码器专用电缆，均有金属屏蔽层抗干扰。



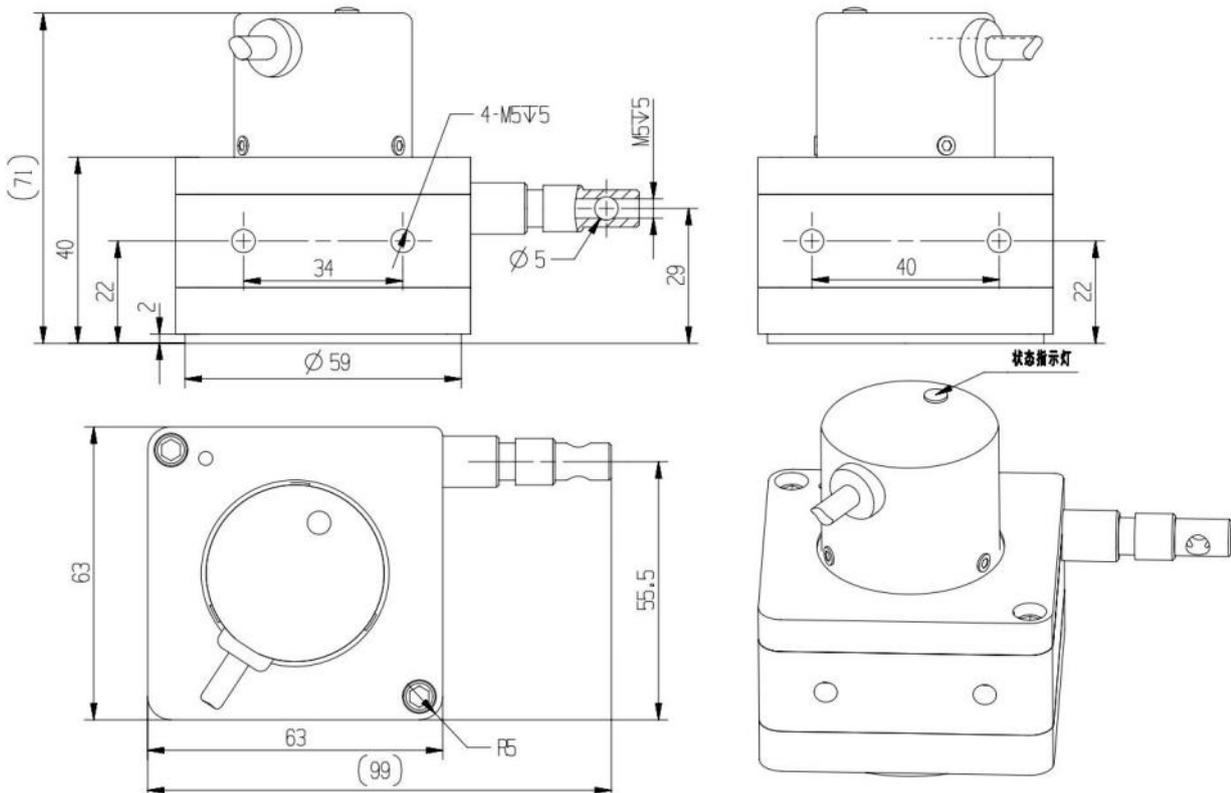
## 八、产品尺寸图

\*2D 图纸及 3D 模型均可从官网下载 (<https://oidencoder.com>)

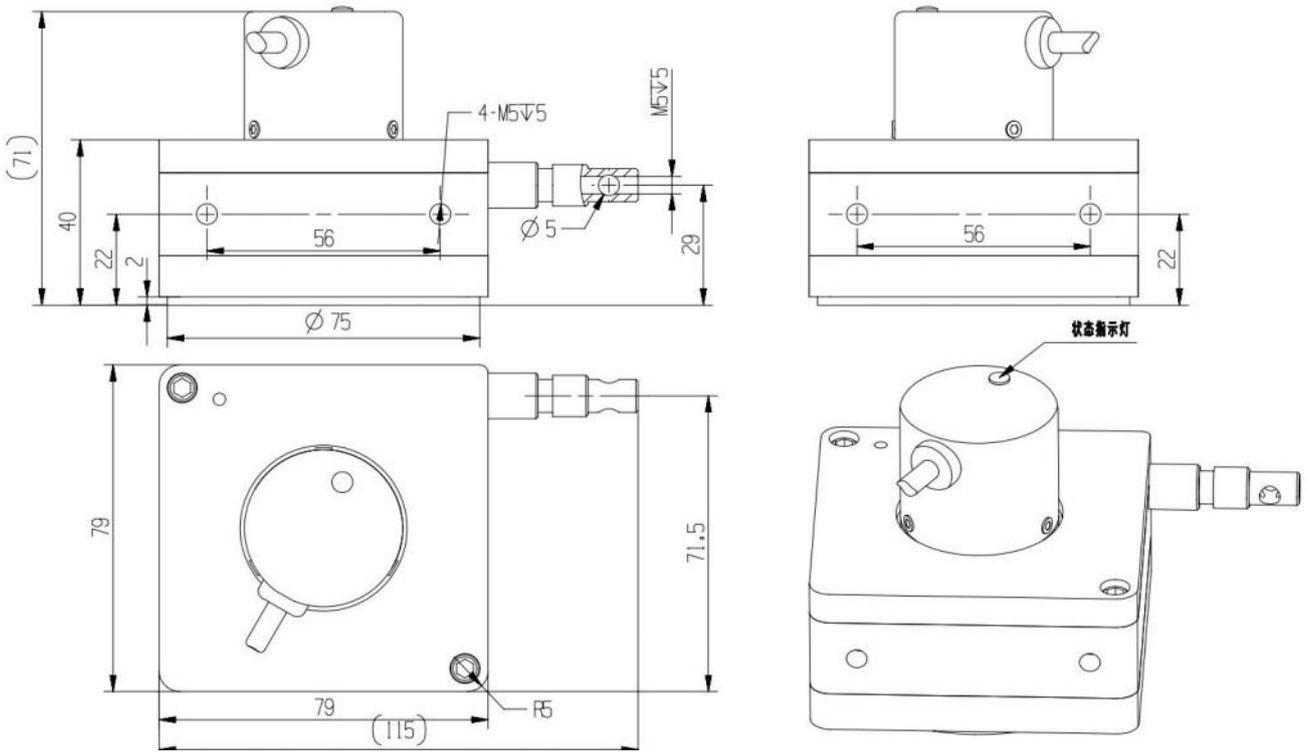
### ■ IP54 OID-38-L0.5&1M 拉线位移传感器 ↓



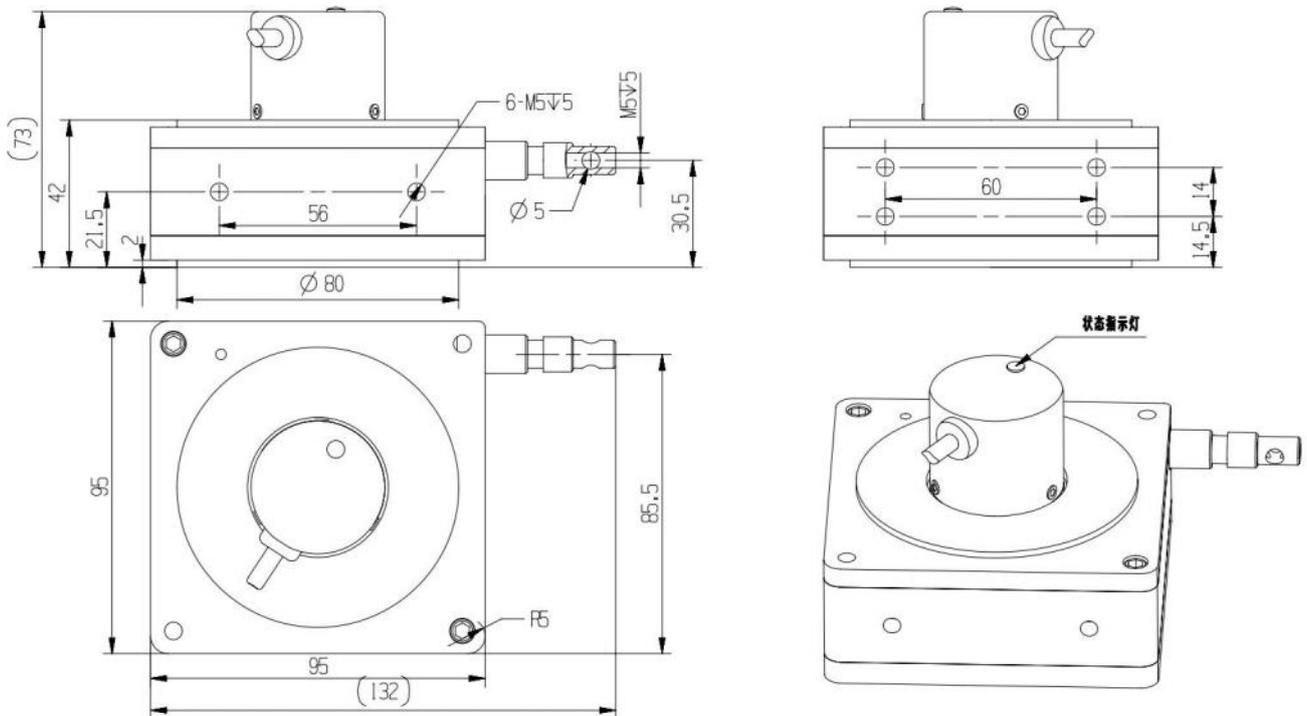
### ■ IP54 OID-38-L2M 拉线位移传感器 ↓



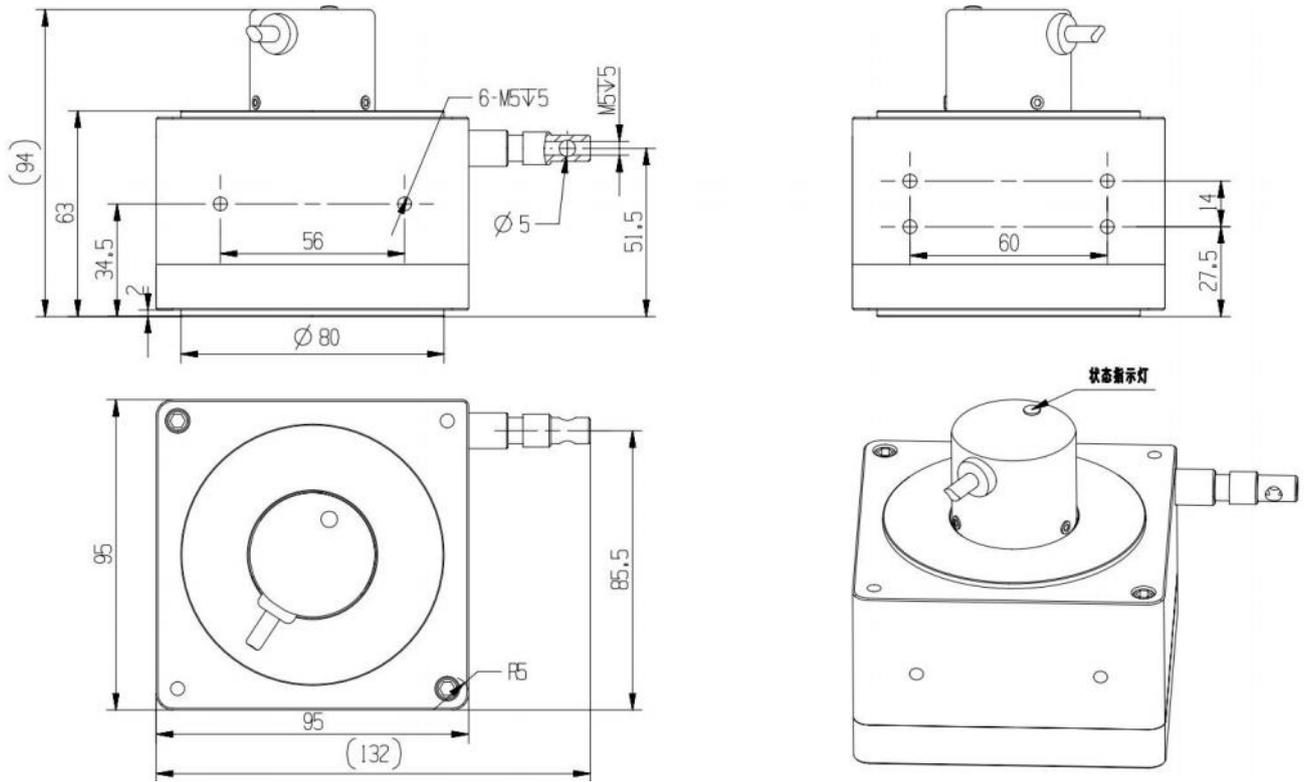
■ IP54 OID-38-L3M 拉线位移传感器 ↓



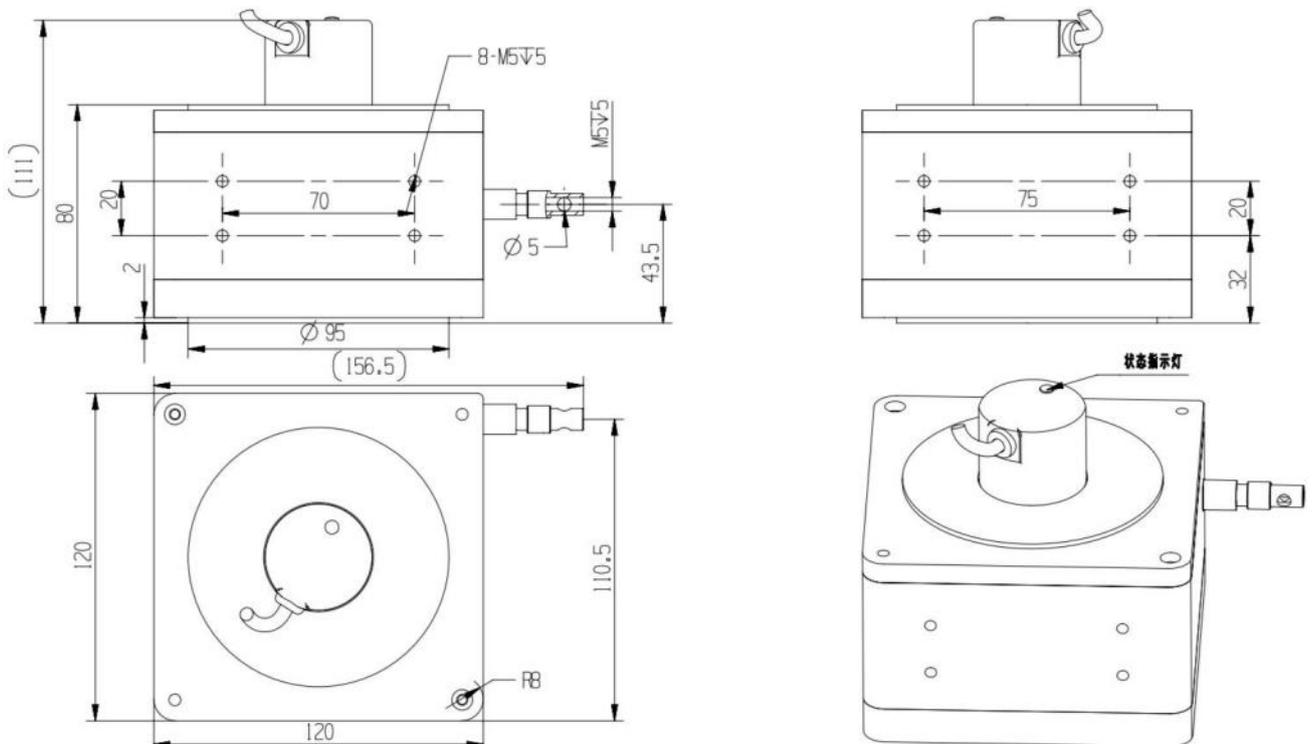
■ IP54 OID-38-L4&L5M 拉线位移传感器 ↓



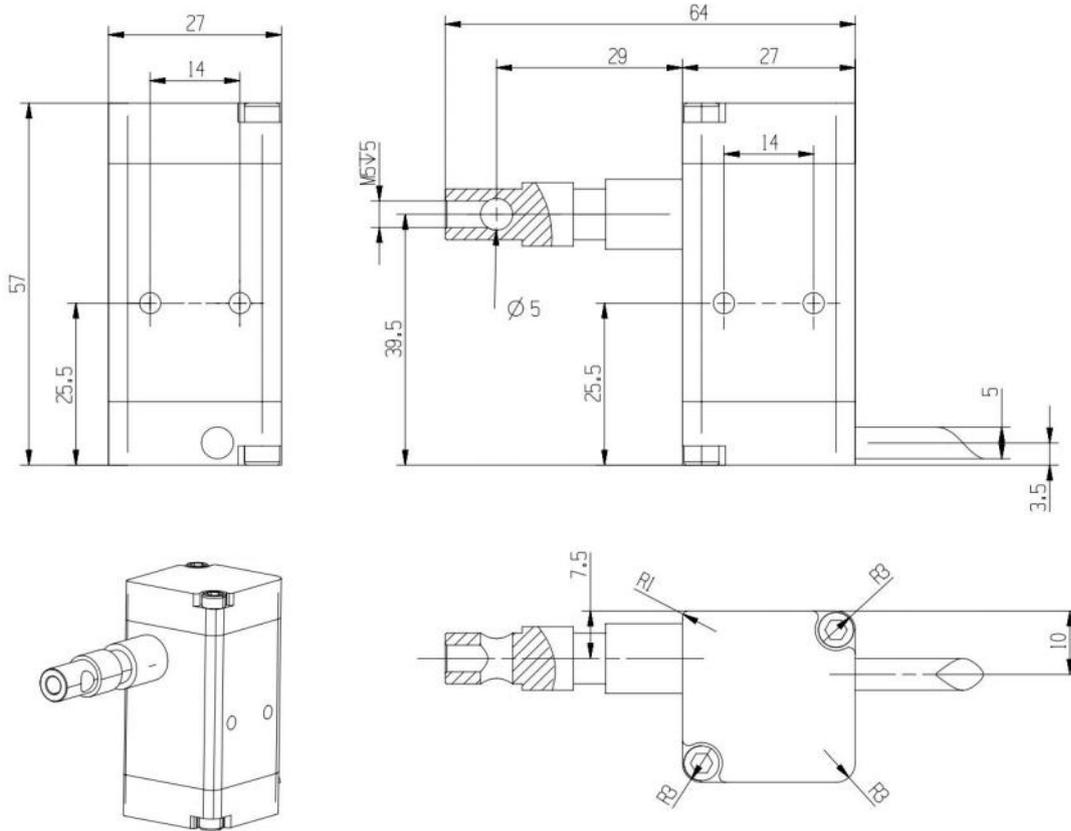
■ IP54 OID-38-L6&L7M 拉线位移传感器 ↓



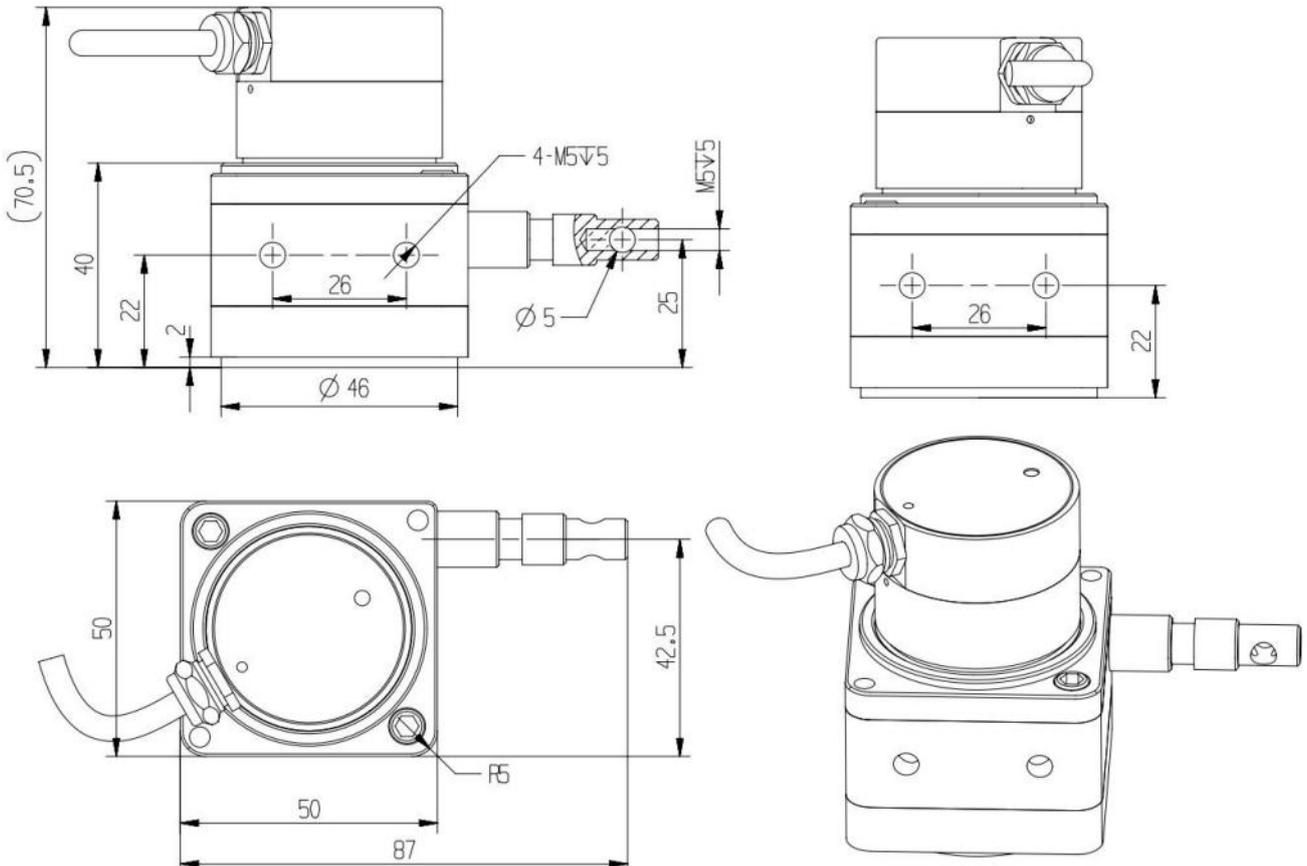
■ IP54 OID-38-10M 拉线位移传感器 ↓



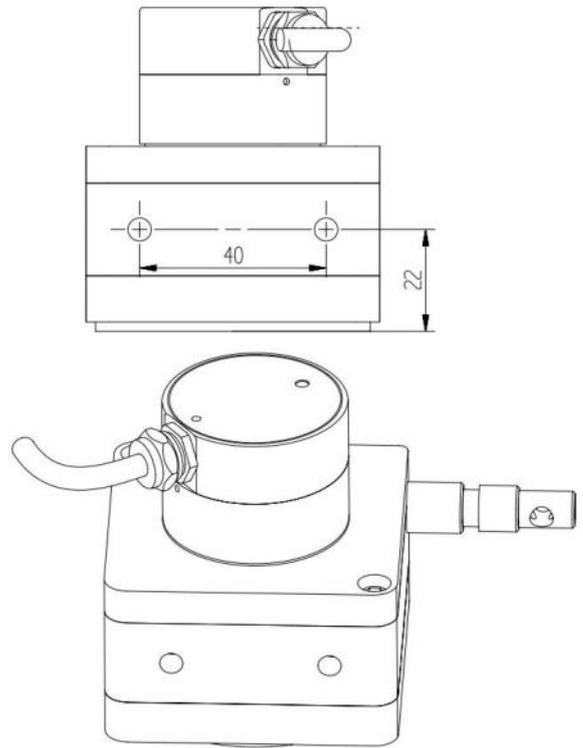
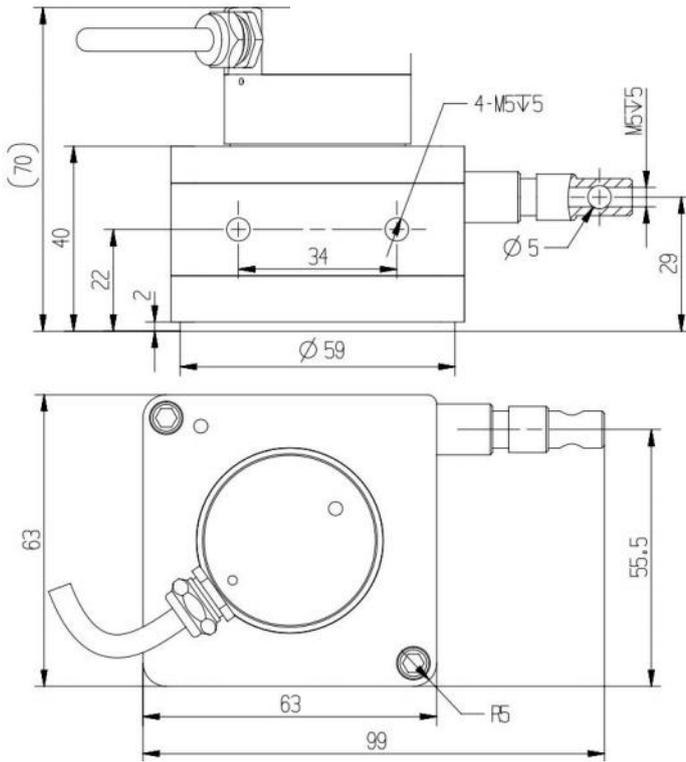
■ IP68 OID-27-L0.5M 拉线位移传感器 ↓



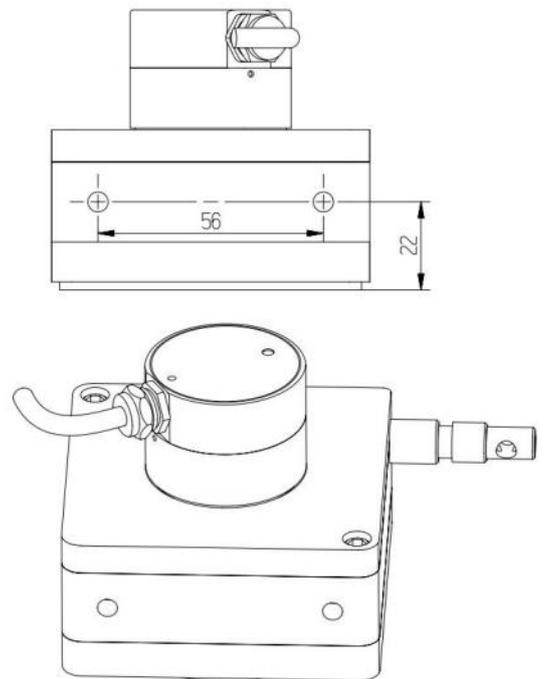
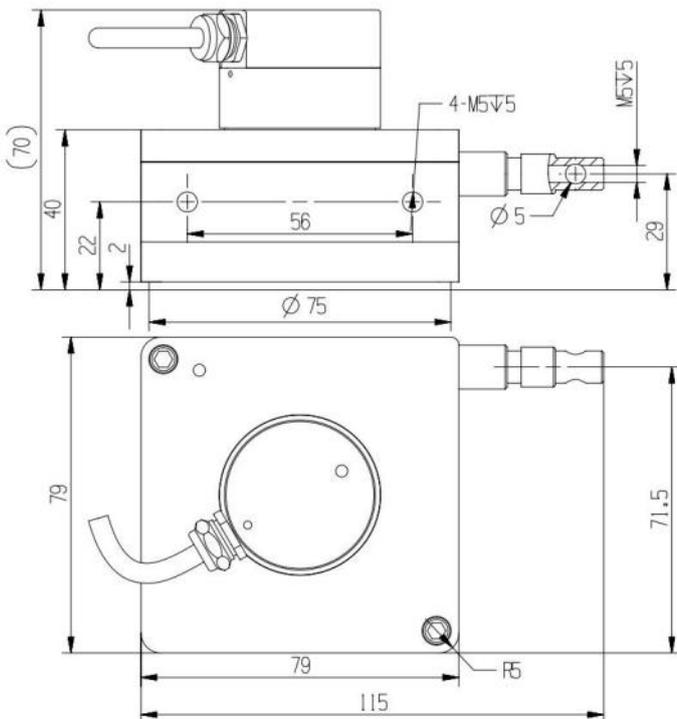
■ IP68 OID-38-L0.5&1M 拉线位移传感器 ↓



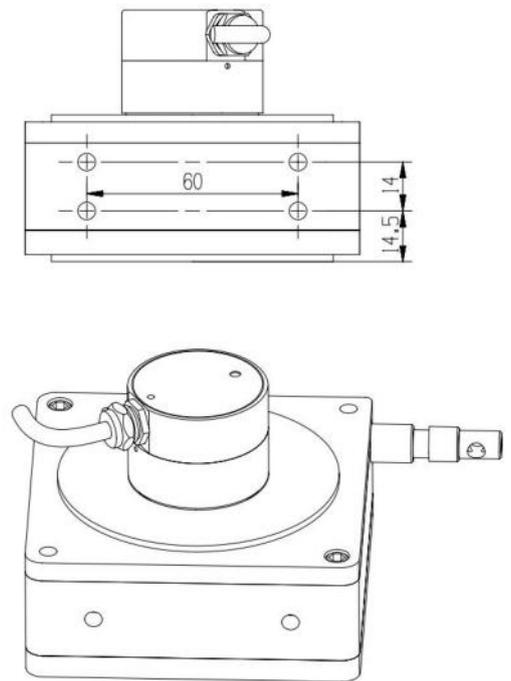
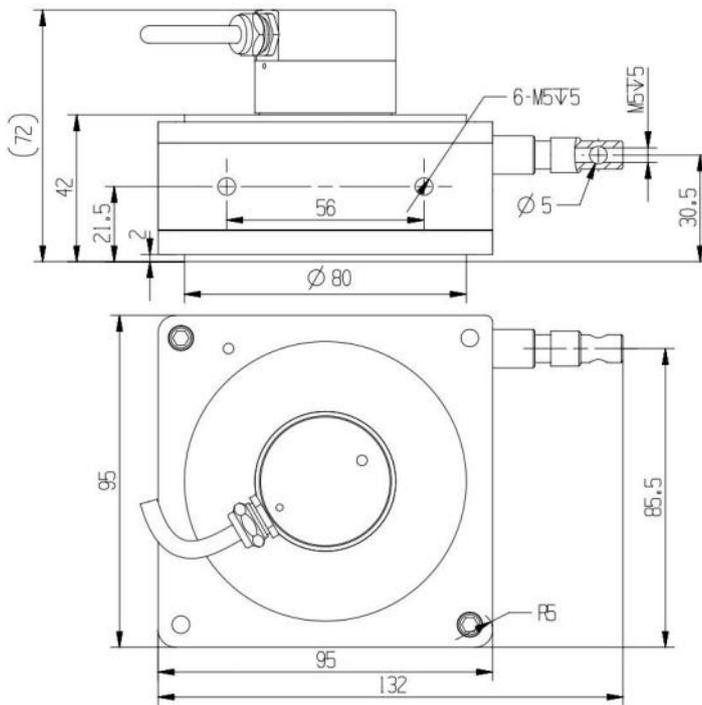
■ IP68 OID-38-L2M 拉线位移传感器 ↓



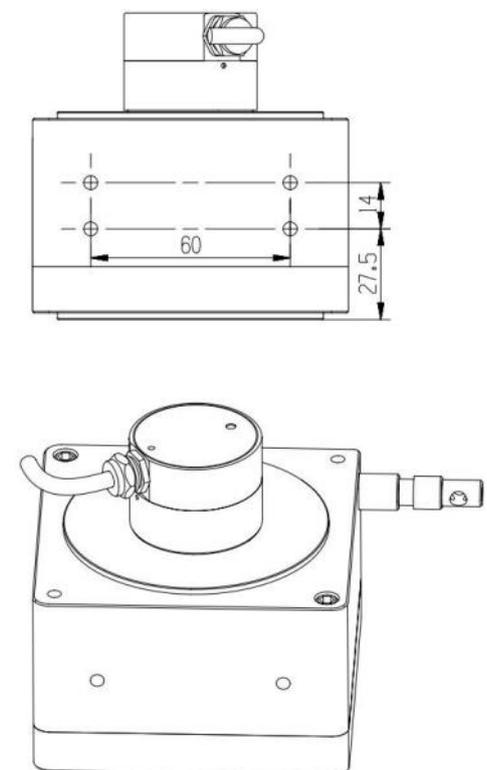
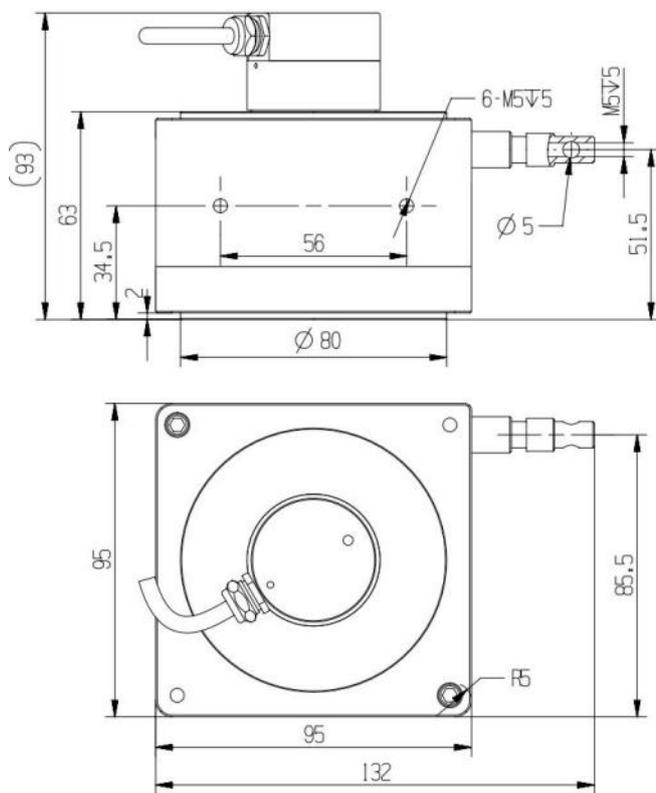
■ IP68 OID-38-L3M 拉线位移传感器 ↓



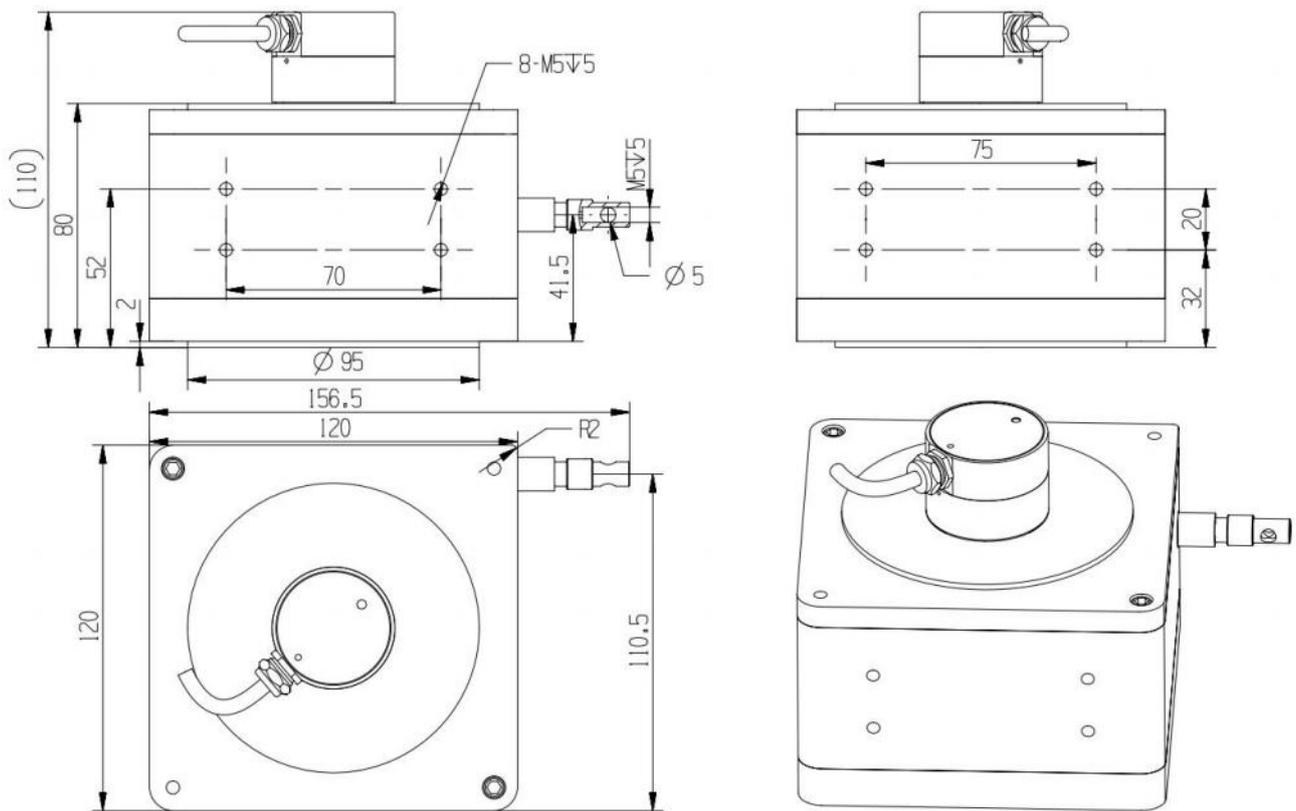
■ IP68 OID-38-L4&L5M 拉线位移传感器 ↓



■ IP68 OID-38-L6&L7M 拉线位移传感器 ↓



■ IP68 OID-38-L8&L9&L10 M 拉线位移传感器 ↓



## 九、工作原理

- 拉绳位移传感器的功能是把机械运动转换成可以计量、记录或传送的电信号（通信信号）。
- 拉绳位移传感器安装在固定位置上，拉绳缚在移动物体上。拉绳直线运动和移动物体运动轴线对准。
- 拉绳位移传感器由可拉伸的不锈钢绳绕在拉线盒主体内的轮毂上，此轮毂与旋转编码器连接在一起，拉动拉绳头即可带动编码器旋转，输出一个与拉绳移动距离成比例的电信号，即测量输出信号，从而可以得出运动物体的位移、方向或速率。



## 十、拉线位移传感器安装注意事项

- 选择拉线位移传感器量程需留有余量，严禁超量程使用，一般在实际行程的基础上选大一规格的行程即可。
- 利用传感器安装支架或者固定螺丝孔，依现场及机器安装空间设施需要，直接安装或另加保护装置。
- 拉绳传感器安装使用时，需注意拉绳与出绳口保持平行状态（容许最大偏差 $\pm 3^\circ$ ），如果运行方向发生改变，应加装滑轮装置，以确保传感器测量精度与使用寿命。
- 拉绳位移传感器未安装固定前，请勿将拉绳拉出并瞬间自行弹回，此举将造成拉绳断裂，严重时可能损坏传感器。
- 非技术人员严禁拆卸，如有需要请在技术人员指导下进行拆卸重装。



## 十一、售后服务

- 本公司生产的产品，正确使用情况下，提供免费保修期 2 年，(非人为破坏)。
- 免保期结束后，我司将继续提供产品终身维修服务，根据产品维修的具体情况，收取相应的材料成本费用。

## 附录一：CANopen 通信协议

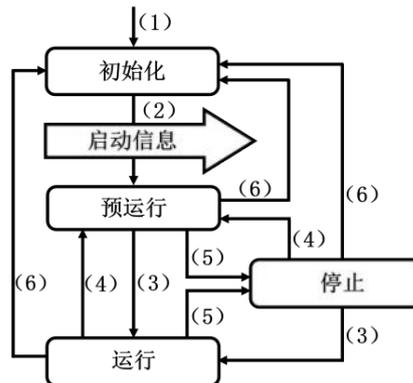
本类编码器遵循“编码器设备行规 Class2”，一般都用作从设备。

### 1.EDS 文件

单圈及多圈 EDS 文件可在我司官网下载（[www.oidencoder.com](http://www.oidencoder.com)），在使用 CANopen 编码器前请在 CANopen 主控制器上安装 EDS 文件。

### 2. 状态机

该 CANopen 设备可以处于不同的工作状态，通过向它发送特定的 NMT 报文，可以在不同的工作状态之间切换。状态图如下所示：



上电 ——> 初始化完成，自动发送启动信息 ——> NMT 报文“启动远程节点” ——> NMT 报文“进行预运行” ——> NMT 报文“关闭远程节点” ——> NMT 报文“复位节点”或“复位通讯”

#### 2.1 初始化

这是上电或硬件复位后，CANopen 设备首次进入的状态。在读取完存储在 EPROM 里的参数信息完成基本的设备初始化后，CANopen 设备（编码器）自动发送启动信息给主控制器进入“预运行”状态。

#### 2.2 预运行

在这种状态下可以通过 SDO 进行通讯。因为 PDO 还不存在，所以不能通过 PDO 进行通讯。通过组态程序可以完成对 PDO 的组态和参数配置。通过发送“启动远程节点”可以直接使编码器进入运行状态。

#### 2.3 运行

在这种状态下，所有的通讯对象都是可用的。从设备根据对象字典中的参数设置可以通过 PDO 发送过程数据。主控制器可以通过 PDO 访问对象字典。

主控制器通过发送“预运行”报文使其进入“预运行”状态。

#### 2.4 停止

这种状态下从设备被强制停止所有通信（除了监控节点）。也不能通过 PDO 和 SDO 进行通信。主控制器通过发送特定的 NMT 报文可以使从编码器直接进入预运行或运行状态。

### 3. 通信对象

共有 4 种通信报文：

- 网络管理 NMT:NMT 主控制器控制 NMT 从设备的 NMT 状态。
- 过程数据对象 PDO：用于传输实时数据。
- 服务数据对象 SDO：用于直接访问 CANopen 设备的对象字典。
- 特殊功能对象：

同步传输（Sync）：提供了基本网络同步机制。使用该服务，主控制器可以发送实时数据。

紧急（Emergency）：每当错误事件发生时，使用该对象。

节点监控（Nodeguard）：用来查看从设备的运行状态。设备状态与通讯对象之间的关系。

	初始化	预运行	运行	停止
NMT		×	×	×
PDO			×	
SDO		×	×	
Sync			×	
Emerg	×	×	×	
Boot-up				
Nodeg		×	×	×

#### 3.1 预定义连接指令

主控制器→编码器（广播）		
通讯对象 COB 类型	功能码(二进制)	COB-ID(十六进制)
NMT	0000	000
SYNC	0001	080
主控制器→编码器（点对点）		
Emergency	0001	081-0FF

PD01(发送)	0011	181-1FF
PD02(发送)	0101	281-2FF
PD03(发送)	0111	381-3FF
SDO(发送)	1011	581-5FF
SDO(接收)	1100	601-67F
节点监控	1110	701-77F

“通讯对象类型“（发送/接收）是站在从设备（编码器）的角度而言的。启动报文使用节点监控对象的 COB-ID。

#### 4.NMT 对象

NMT 结构

COB-ID(11 位)		2 字节 CAN 数据	
功能码	节点 ID	命令	编码器 ID
0000	0	NMT 功能	编码器 ID

如果编码器 ID 为 00h，则 NMT 报文发往接入网络的所有节点。NMT 功能

命令（十六进制）	NMT 函数	节点状态
01	开始远程节点	运行
02	停止远程节点	停止
80	进入预运行	预运行
81	复位节点	预运行
82	复位通讯	预运行

#### 5. 启动（Boot-up）对象

Boot-up 报文结构：

COB-ID(十六进制)	1 字节 CAN 数据
700+节点 ID	00

#### 6. PDO 对象：

PDO（发送）报文有 4 个 CAN 数据字节组成，用来传送编码器的位置值。

PDO 结构

标识		4 字节 CAN 数据			
COD-ID(十六进制)		字节 0	字节 1	字节 2	字节 3
功能码	节点 ID	$2^7_{-2}^0$	$2^{15}_{-2}^8$	$2^{23}_{-2}^{16}$	$2^{31}_{-2}^{24}$
		低字节	...	...	高字节

3 种 PDO 的定义如下:

**PDO 循环模式:** 异步传输。绝对值旋转编码器不经主机查询, 自动将当前过程数据循环发送。周期时间可以编程设定, 值在 1 到 65536 之间, 单位为 ms,(可参见“循环时间: 6200h 对象”)。通过将 PD01 (对象 1800h, 子项 1) 所使用的 COB-ID 的最高位置为“0” (“1”) 就可以使用 (或禁用) 循环模式。

**PD0 同步模式:** 同步传输。同步指的是经过主控制器同步后, 再发送 PDO。同步报文是主控制器发往所有从设备的具有高优先级的 COB, 接收到该同步报文后, 编码器将过程数据传回。每个从设备按照自身的节点 ID 进行数据传输。

**注: 多个传输模式可以共存。**

## 7. SDO 对象:

SDO 报文用来查询或改变编码器的参数, 这些参数都包含在对象字典中。CAN 数据最大 4 个字节, 其他 4 个字节用于命令、索引和子索引等域。控制器发出一个 SDO 报文给编码器时, 编码器要发送响应信息给主控制器 (出错时, 则报警信息)。

SDO 结构

标识符		4 字节 CAN 数据				1 到 4 字节的 CAN 数据			
标识符 (十六进制)		0	1	2	3	4	5	6	7
功能码	节点 ID	命令	索引		子索引	过程数据			
		1 字节	低字节	高字节	1 字节	低字节	...	...	高字节

### 7.1 命令

命令字节的内容以报文的形式通过 CAN 网络传输。有三种报文形式

- 设置: 发送配置参数给设备;
- 请求: 主控制器用来读取设备的数据;
- 报警: 编码器用来发送错误信息给主控制器 (如: 索引不存在、参数无效等)。

命令	COB	COB 类型	数据长度
23h	设置	M→S 请求	4 字节
2Bh	设置	M→S 请求	2 字节
2Fh	设置	M→S 请求	1 字节
60h	设置	S→M 确认	

40h	请求	M→S 请求	0 字节
43h	请求	S→M 应答	4 字节
4Bh	请求	S→M 应答	2 字节
4Fh	请求	S→M 应答	1 字节
41h	请求	S→M 应答,分段传输 SDO	
80h	报警	S→M 应答	4 字节

## 8. 对象字典

每一个对象以如下形式表示：

索引-子索引 对象名称[数据类型，属性]

-索引和子索引使用十六进制标识。

-属性：ro=只读，rw=可读写。

-Unsigned16 数据类型：

过程数据字节	
字节 4	字节 5
低字节	高字节

-Unsigned32 数据类型：

过程数据字节			
字节 4	字节 5	字节 6	字节 7
低字节	...	...	高字节

### 8.1 标准对象 (DS 301)

索引-子索引	对象名称 [数据类型，属性]
1000-00	设备类型 默认值：0001 0196h=单圈编码器，DS 406 [Unsigned32, ro] 0002 0196h=多圈编码器，DS 406
1001-00	错误寄存器 [Unsigned8, ro] 如果该寄存器的某一位设置为“1”，则其对应的错误已经发生。 默认值：00h
1003	预定义的错误域  该对象保存设备已产生过的错误。

-00	发生错误的次数 写入 00H 则清除错误历史记录。	[Unsigned8, rw]																																				
-01	最近一次发生的错误	[Unsigned32, ro]																																				
-02	.																																					
.	.																																					
.	.																																					
-08	更早一些时间发生的错误	[Unsigned32, ro]																																				
1004	Numbe of PDOs Supported (支持的 PDO 数量)																																					
-00	Number of Entries (入口数量)	[Unsigned32, ro]																																				
-01	Number of SyncPDOs (同步 PDO 数量)	[Unsigned32, ro]																																				
-02	Number of AsyncPDOs (异步 PDO 数量)	[Unsigned32, ro]																																				
1005-00	COB-ID 同步报文 默认值: 0000 0080h	[Unsigned32, rw]																																				
1008-00	设备生厂商名 默认值: “OidEncoder”	[String, ro]																																				
1009-00	硬件版本	[String, ro]																																				
100A-00	软件版本	[String, ro]																																				
1010-01	<p>存储参数 [Unsigned32, rw] 这一对象保存所有参数到非易失性存储器。写入的信号为“save”。 控制器→编码器(写入)</p> <table border="1"> <thead> <tr> <th>COB-ID</th> <th>命令</th> <th colspan="2">索引</th> <th>子索引</th> <th colspan="4">数据字节</th> </tr> </thead> <tbody> <tr> <td>600+ID</td> <td>23</td> <td>10</td> <td>10</td> <td>01</td> <td>73</td> <td>61</td> <td>76</td> <td>65</td> </tr> </tbody> </table> <p>编码器→控制器(确认)</p> <table border="1"> <thead> <tr> <th>COB-ID</th> <th>命令</th> <th colspan="2">索引</th> <th>子索引</th> <th colspan="4">数据字节</th> </tr> </thead> <tbody> <tr> <td>580+ID</td> <td>60</td> <td>10</td> <td>10</td> <td>01</td> <td>00</td> <td>00</td> <td>00</td> <td>00</td> </tr> </tbody> </table>	COB-ID	命令	索引		子索引	数据字节				600+ID	23	10	10	01	73	61	76	65	COB-ID	命令	索引		子索引	数据字节				580+ID	60	10	10	01	00	00	00	00	
COB-ID	命令	索引		子索引	数据字节																																	
600+ID	23	10	10	01	73	61	76	65																														
COB-ID	命令	索引		子索引	数据字节																																	
580+ID	60	10	10	01	00	00	00	00																														

<p>1011-01</p>	<p>恢复默认参数 [Unsigned32, rw]            此对象用来恢复所有参数的缺省值。写入的信号为“load”时，设置复位后，恢复的默认值生效。            控制器→编码器(写入)</p> <table border="1" data-bbox="379 409 1428 512"> <thead> <tr> <th>COB-ID</th> <th>命令</th> <th colspan="2">索引</th> <th>子索引</th> <th colspan="4">数据字节</th> </tr> </thead> <tbody> <tr> <td>600+ID</td> <td>23</td> <td>11</td> <td>10</td> <td>01</td> <td>6C</td> <td>6F</td> <td>61</td> <td>64</td> </tr> </tbody> </table> <p>编码器→控制器(确认)</p> <table border="1" data-bbox="379 571 1428 674"> <thead> <tr> <th>COB-ID</th> <th>命令</th> <th colspan="2">索引</th> <th>子索引</th> <th colspan="4">数据字节</th> </tr> </thead> <tbody> <tr> <td>580+ID</td> <td>60</td> <td>11</td> <td>10</td> <td>01</td> <td>00</td> <td>00</td> <td>00</td> <td>00</td> </tr> </tbody> </table>	COB-ID	命令	索引		子索引	数据字节				600+ID	23	11	10	01	6C	6F	61	64	COB-ID	命令	索引		子索引	数据字节				580+ID	60	11	10	01	00	00	00	00
COB-ID	命令	索引		子索引	数据字节																																
600+ID	23	11	10	01	6C	6F	61	64																													
COB-ID	命令	索引		子索引	数据字节																																
580+ID	60	11	10	01	00	00	00	00																													
<p>1014-00</p>	<p>EMCY COB-ID [Unsigned32, rw]            默认值 80h+节点 ID            这一对象定义了 EMCY 写服务的 COB-ID.</p>																																				
<p>1017-00</p>	<p>Producer Heartbeat Time (心跳时间) [Unsigned16, rw]</p>																																				
<p>1018 -01 -02 -03 -04</p>	<p>标识对象            制造商代码 [Unsigned32, ro]            产品代码 [Unsigned32, ro]            修订号 [Unsigned32, ro]            序列号 [Unsigned32, ro]</p>																																				
<p>1800 -00 -01 -02 -03 -05</p>	<p>PD01 通讯参数            入口数量 [Unsigned8, ro]            默认值: 5H            PD01 的 COB-ID [Unsigned32, rw]            0180h+节点 ID            传输类型 [Unsigned8, rw]            默认值: FEH (异步传输)            InhibitTime (禁止时间) [Unsigned16, rw]            单位: 0.1ms            Event Time (发送间隔时间) [Unsigned16, rw]            单位: ms</p>																																				
<p>1801 -00 -01 -02</p>	<p>PD02 通讯参数            入口数量 [Unsigned8, ro]            默认值: 5H            PD02 的 COB-ID [Unsigned32, rw]            0280h+节点 ID            传输类型 [Unsigned8, rw]</p>																																				

		默认值：01H（同步传输） 对于需要 n 个同步信号的 n 值，可以在对象 1801h 的子索引 2 中设定。	
-03	InhibitTime（禁止时间）		[Unsigned16, rw]
	单位：0.1ms		
-05	Event Time（发送间隔时间）		[Unsigned16, rw]
	单位：ms		
1A00	入口数量		[Unsigned8, rw]
	默认值：1		
-00	PD01 映射参量		[Unsigned32, rw]
	默认值：60040020h		
-01	该对象遵循设备行规 DS406 的规定，包含编码器的位置值。		
1A01	-00	入口数量	[Unsigned8, rw]
		默认值：1	
	-01	PD02 映射参量	[Unsigned32, rw]
		默认值：60040020h	

### 8.2 与制造商相关的对象

索引-子索引	对象名称	[数据类型, 属性]			
3000-00	波特率	[Unsigned8, rw]			
	这一对象定义了设备的比特率，如下表所列				
	数据	代表的波特率			
	00h	10Kbit/s			
	01h	20Kbit/s			
	02h	50Kbit/s			
	03h	100Kbit/s			
	04h	125Kbit/s			
	05h	250Kbit/s			
	06h	500Kbit/s(默认)			
	07h	800Kbit/s			
	08h	1000Kbit/s			
	改变波特率的步骤：设置对象 3000h，然后存储参数，最后发送命令“复位节点”（或“复位通信”）。				
	控制器→编码器				
	COB-ID	命令	索引	子索引	数据
600+ID	2F	00	30	00	04 00 00 00
编码器→控制器（确认）					
COB-ID	命令	索引	子索引	数据	
580+ID	60	00	30	00	00 00 00 00
存储参数（参见对象 1010h），若不能存储则重新上电后使用原来的波特率。					

3001-00	节点 ID [Unsigned8, rw] 这一对象定义了设备的节点标识符。 默认值: 01h 改变节点地址的步骤为: 设置对象 3001h, 发送命令存储参数, 最后“复位节点”。 控制器→编码器 (写入)					
	COB-ID	命令	索引		子索引	数据
	600+ID	2F	01	30	00	01 00 00 00
	编码器→控制器 (确认)					
	COB-ID	命令	索引		子索引	数据
	580+ID	60	01	30	00	00 00 00 00
存储参数 (参见对象 1010h), 若不能存储则重新上电后将使用原来的节点 ID.						

### 8.3 设备行规规定的对象 (DS 406)

索引-子索引	对象名称 [数据类型, 属性]																
6000-00	运行参数																
	<table border="1"> <thead> <tr> <th>Bit</th> <th>功能</th> <th>Bit=0</th> <th>Bit=1</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>编码器计数方向</td> <td>顺时针</td> <td>逆时针</td> </tr> <tr> <td>1</td> <td>硬件自检</td> <td>关闭</td> <td>使能</td> </tr> <tr> <td>2</td> <td>缩放</td> <td>关闭</td> <td>使能</td> </tr> </tbody> </table>	Bit	功能	Bit=0	Bit=1	0	编码器计数方向	顺时针	逆时针	1	硬件自检	关闭	使能	2	缩放	关闭	使能
	Bit	功能	Bit=0	Bit=1													
	0	编码器计数方向	顺时针	逆时针													
	1	硬件自检	关闭	使能													
2	缩放	关闭	使能														
默认值: 0000h																	
--编码器计数方向定义了从编码器轴上看去, 旋转轴顺时针或逆时针旋转时, 计数值是增加还是减小。																	
--缩放功能: 如果禁用该功能, 则使用物理分辨率 (参见对象 6501h 和 6502h)。																	
6001-00	每转分辨率 [Unsigned32, rw] 默认值: 2000H。 如果 6000 参数 bit2=1, 可用于改变单圈分辨率, 但应不大于 6501 参数。																
6002-00	总测量范围 [Unsigned32, rw] 默认值: 20000000H。 如果 6000 参数 bit2=1, 可用于改变多圈总分辨率, 但应不大于 6502 参数。																
6003-00	预设值 预设值是设定的位置值, 为防止运行出错, 预设值不得超过“总的硬件分辨率”。																
6004-00	当前位置值 [Unsigned32, ro]																
6200-00	循环时间 [Unsigned16, rw] 默认值: 0064h (100ms)。 循环定时器用在异步通讯中, 用来调整 PD01 (对象 1800-05h) 传输时循环间隔。																
6500-00	操作状态 [Unsigned16, ro]																
6501-00	每转分辨率 [Unsigned32, ro] 该对象用来定义硬件上每转可分辨的步数。要使用其他数值, 参见对象 6001h。																

6502-00	硬件总圈数 该对象定义了硬件上可以测量的最大圈数。要使用其他数值，参见 6001h 和 6002h。	[Unsigned32, ro]
6503-00	报警值	[Unsigned16, ro]
6504-00	报警支持 默认值：1H	[Unsigned16, ro]
6505-00	警告值	[Unsigned16, ro]
6506-00	警告支持 默认值：4H	[Unsigned16, ro]
6507-00	外形和软件版本 默认值：01000100H。	[Unsigned32, ro]
6508-00	运行时间 默认=FFFF FFFFh（不使用） 单位：0.1 小时。	[Unsigned32, ro]
6509-00	偏移量 此对象包含了偏移值，它是根据预置值和位置值计算出来的。	[Integer32, ro]
650B-00	序列号 默认=FFFF FFFFh（不使用）	650B-00

**注：**为了使改动的参数生效，需执行“保存参数”的操作（参见对象 1010h）。如果没执行“保存参数”的操作，则“复位节点”命令，“复位通讯”命令或关闭电源时，参数就会丢失。

## 9. 设置参数

下面是一些参数设置时，主控制器和编码器之间数据交换的例子。“ID”用来表示编码器的地址。数值采用十六进制的记法。

·设置运行、预运行状态

控制器→编

码器

NMT 报文

运行

预运行

COB-ID	命令	节点
000	01	ID
000	80	ID

·设置单圈分辨率（ $2^{12}$ =0000 1000h）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	23	01	60	00	00	10	00	00

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	01	60	00	00	00	00	00

·设置总分辨率（ $2^{24}=0100\ 0000h$ ）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	23	02	60	00	00	00	00	01

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	02	60	00	00	00	00	00

·设置运行参数（计数方向：顺时针，缩放功能：使用，硬件自检：禁用）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	2B	00	60	00	04	00	00	00

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	00	60	00	00	00	00	00

设置预置值（预置值为  $1000=03E8h$ ）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	23	03	60	00	E8	03	00	00

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	03	60	00	00	00	00	00

设置同步计数器（ $n=5=05h$ ）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	2F	01	18	02	05	00	00	00

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	01	18	02	00	00	00	00

使用循环模式

设置循环时间（100ms=64h）

控制器→编码器（设置请求）

COB-ID	命令	索引		子索引	过程数据			
600+ID	2B	00	62	00	64	00	00	00

编码器→控制器（设置确认）

COB-ID	命令	索引		子索引	过程数据			
580+ID	60	00	62	00	00	00	00	00

为了使改动的参数生效，需执行“保存参数”的操作（参见对象 1010h）。如果没执行“保存参数”的操作，则“复活节点”命令，“复位通讯”命令或关闭电源时，参数就会丢失。

### 9.1 警告对象

欲了解警告信息的含义请参考我 [www.can-cia.org](http://www.can-cia.org) 上“CIA 标准草案 301”中的“SD0 异常中断代码”部分。

### 9.2 紧急对象

当设备内部出现错误时会触发紧急对象。

紧急对象结构：

标识	CAN 数据			
COB-ID(hex)	0	1	2	3...7
见对象 1014h	错误代码		错误寄存器	特定代码
	最低位	最高位	1001	00...00

已定义的错误代码：

1000h=节点监控错误

5530h=存储器错误

(关注公众号，获取最新产品资讯和教学视频)



官网



视频号



哔哩哔哩



抖音

## 联系我们

 深圳欧艾迪科技有限公司

 全国服务热线：400-166-0195

左工：15814017675(微信同号)

 邮箱：oid@oidencoder.com

 官网：<https://oidencoder.com>

 地址：深圳市宝安区西乡街道盐田社区银田工业区 B9 栋 3 层