

# TYH31 系列 2 轴霍尔操纵杆

版本: V1.0 2015-11-27

## TYH31A 系列(嵌入式安装)



TYH31A0



TYH31A1



TYH31A2



TYH31A3



TYH31A4



TYH31A5

## TYH31B 系列 (面板安装)



TYH31B0



TYH31B1



TYH31B2



TYH31B4



TYH31B5

### 特点:

- 原理: 霍尔传感器, 采用德国进口霍尔传感器, 全温度范围线性校正, 磁性曲线线性校正, 保证输出的线性;
- 2 轴 (X Y), 14 位分辨率;
- 工作电压: 3.2V-5.5V, 可以稳定工作在 3.3V 或 5.0V
- 信号输出: USB、CAN、RS232、RS422、模拟电压  
模拟电压: 5.0V 供电 0.3~2.5~4.7V 或 6%Vdd~50Vdd~94%Vdd, 其它参数可定制。
- 重复定位精度: 小于 1.5%。
- 材料: 铝合金及 ABS+不锈钢, 精密结构工艺
- 防护等级: 面板以上部分 IP67
- 内置操作唤醒开关 (定货时注明):  
机械式开关, 不供电时也有效; 当操纵杆有动作时, 开关闭合, 回到中位时断开; 主要用于电池供电唤醒 CPU, 以降低功耗。

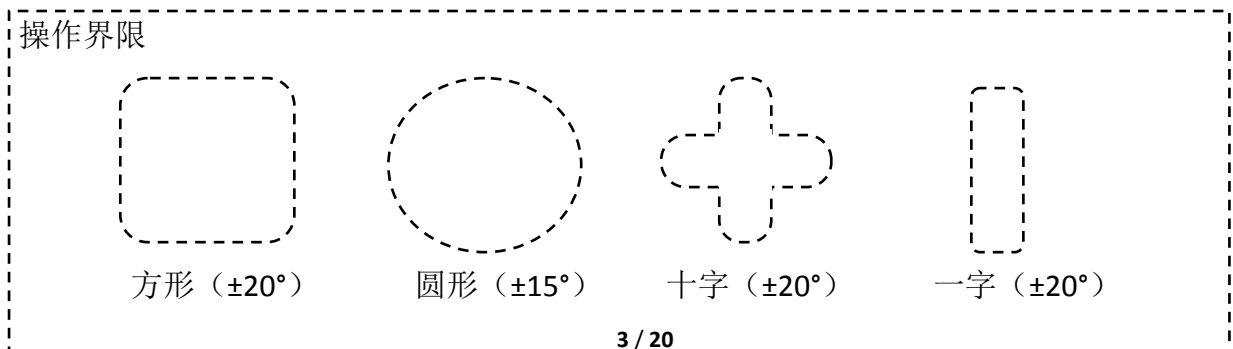
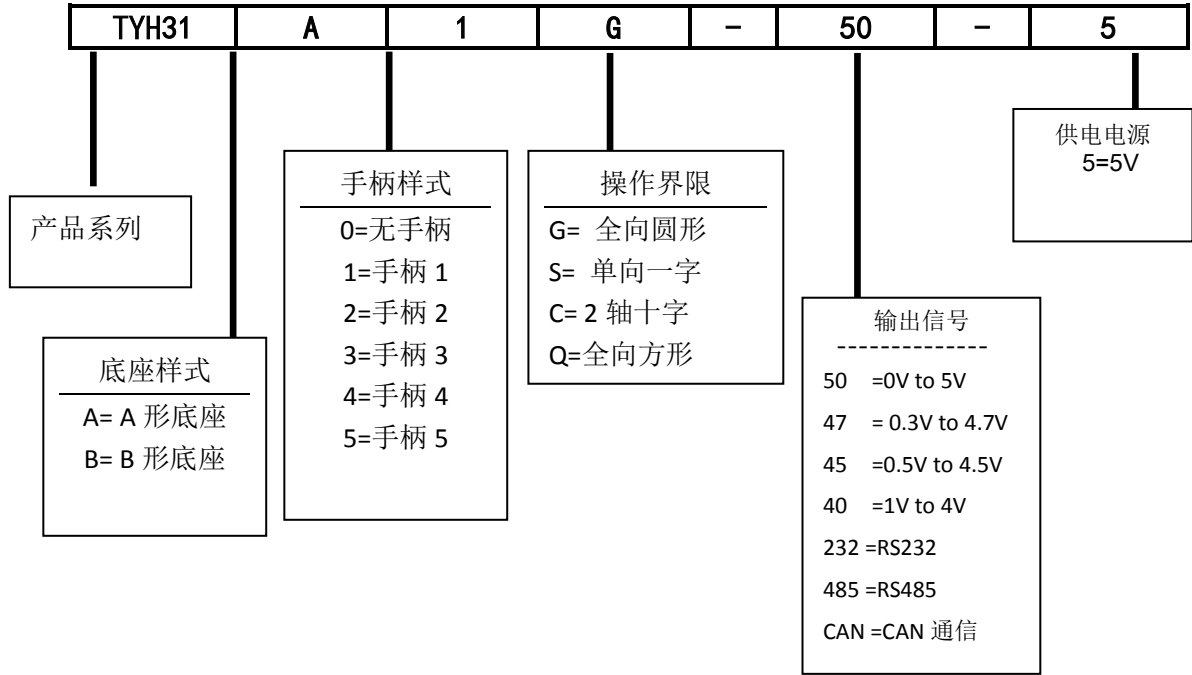
### 用途:

- 机器人
- 医疗设备
- 光学仪器

### 技术参数:

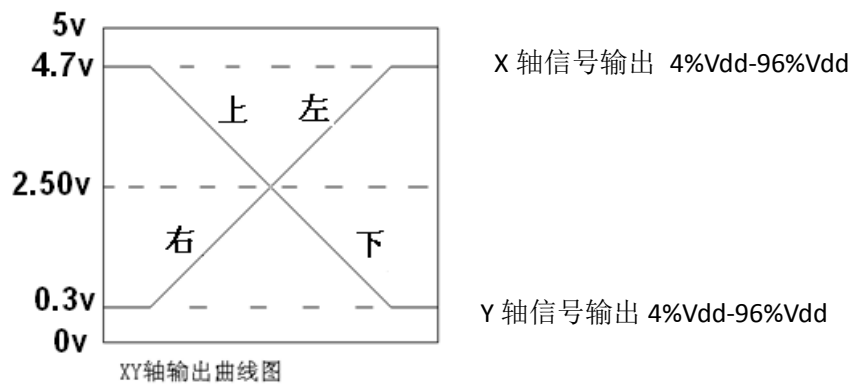
- 传感器: 霍尔传感器, 全量程多点线性校正;
- 信号输出: 线性模拟电压 4%Vdd-96%Vdd, 中心 50%Vdd (通信方式可选择)
- 转动角度:  $\pm 20^\circ$ , 对角线角度:  $\pm 28^\circ$
- 操作力度: 多种可选(或客户定制)
- 中心回位精度: 小于 1%
- 供电电压: DC3.3V 或 5V
- 工作电流: <15MA
- 唤醒开关: 机械式, 中心断开, 转动 5-10° 开关闭合, 功能选配。
- 操作温度:  $-40^\circ\text{C} \sim +70^\circ\text{C}$
- 贮藏温度:  $-50^\circ\text{C} \sim +80^\circ\text{C}$
- 防护等级: 面板以部分 IP67
- 尺寸: TYH31A 外形尺寸 76.2(H)X38(L)X38(W), 安装尺寸: 31X31  
TYH31B 外形尺寸 76.04(H)X47(L)X47(W), 安装尺寸: 35X35

产品型号参数选择

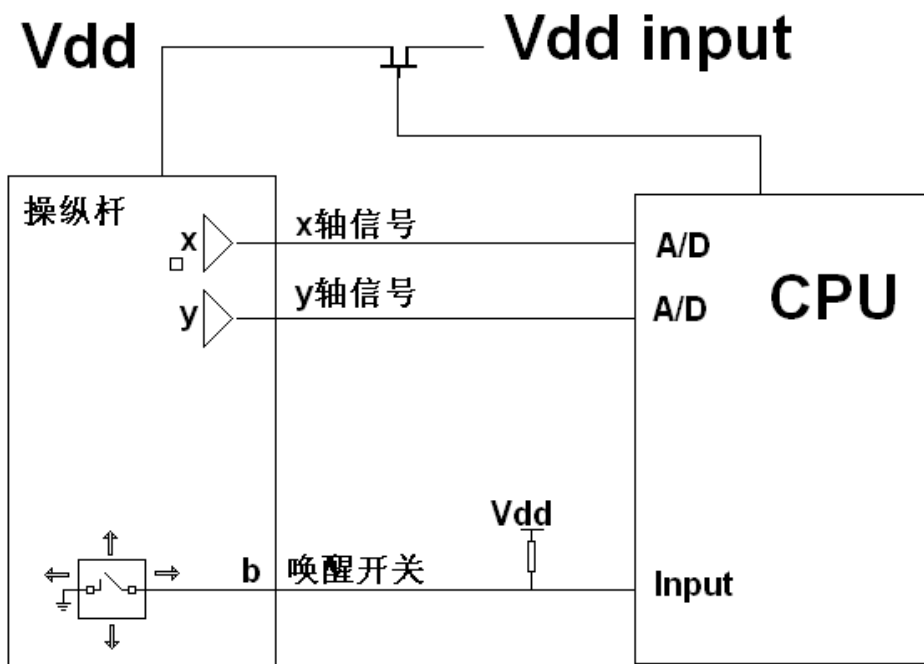




输出曲线图：



操纵杆接线图：



注：唤醒开关；当操纵杆离开中心位置 5-10° 左右时，开关闭合，当操纵杆回到中心时，开关断开，这个功能在不通电的情况下也有效。用于低功耗的产品中，当长时间不动时，CPU 给操纵杆断电，当操作时，用于唤醒 CPU，给操纵杆通电，这样可大大降低整机的待机功耗。

只有订制的操纵杆才有唤醒功能的产品才有这个功能；

模拟接口的引脚定义： (Vdd=5v)

引脚 1 + 5 V 红色	引脚 2 G N D 黑色	引脚 3 B u t t o n 黄色	引脚 4 N C 绿色	引脚 5 Y-Axis 蓝色	引脚 6 X-Axis 棕色
+5V	GND	唤醒开信号输出, 开关另一端已接地	备用	Y 轴 上 MAX=4.7V 中心: 2.5V 下 MIN=0.3V	X 轴 左 MAX=4.7V 中心 2.5V 右 MIN=0.3V

CAN+RS485 接口的引脚定义：

引脚 1 +5V 红色	引脚 2 GND 黑色	引脚 3 CAN-H 黄色	引脚 4 CAN-L 绿色	引脚 5 485A+ 蓝色	引脚 6 485B- 棕色
+5V	GND	CAN-H	CAN-L	A+	B-

CAN+RS232 接口的引脚定义：

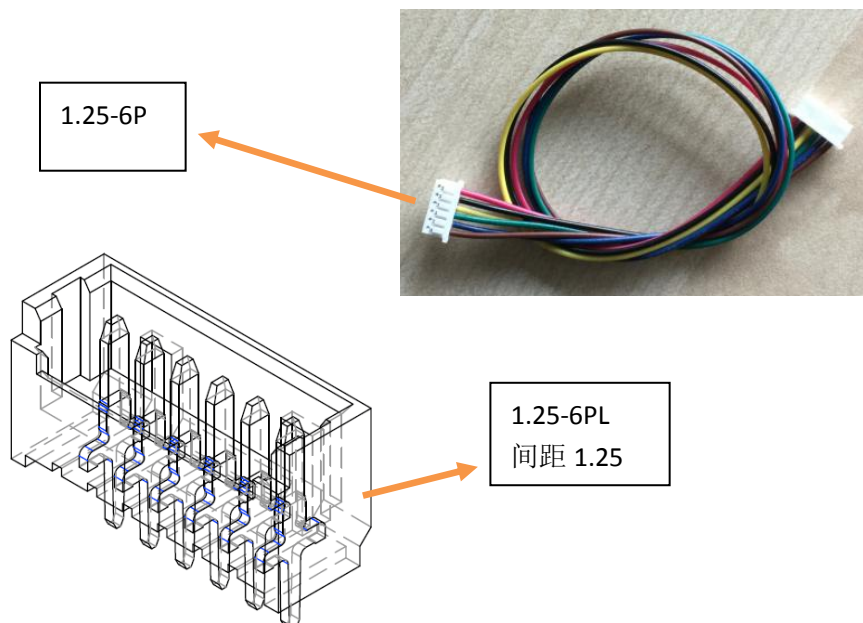
引脚 1 +5V 红色	引脚 2 GND 黑色	引脚 3 CAN-H 黄色	引脚 4 CAN-L 绿色	引脚 5 RS232TXD 蓝色	引脚 6 RS232RXD 棕色
+5V	GND	CAN-H	CAN-L	TXD RS232 发送	RXD RS232 接收

RS232 通信的 GND 与电源的 GND 共用，可从电源接口并接出来

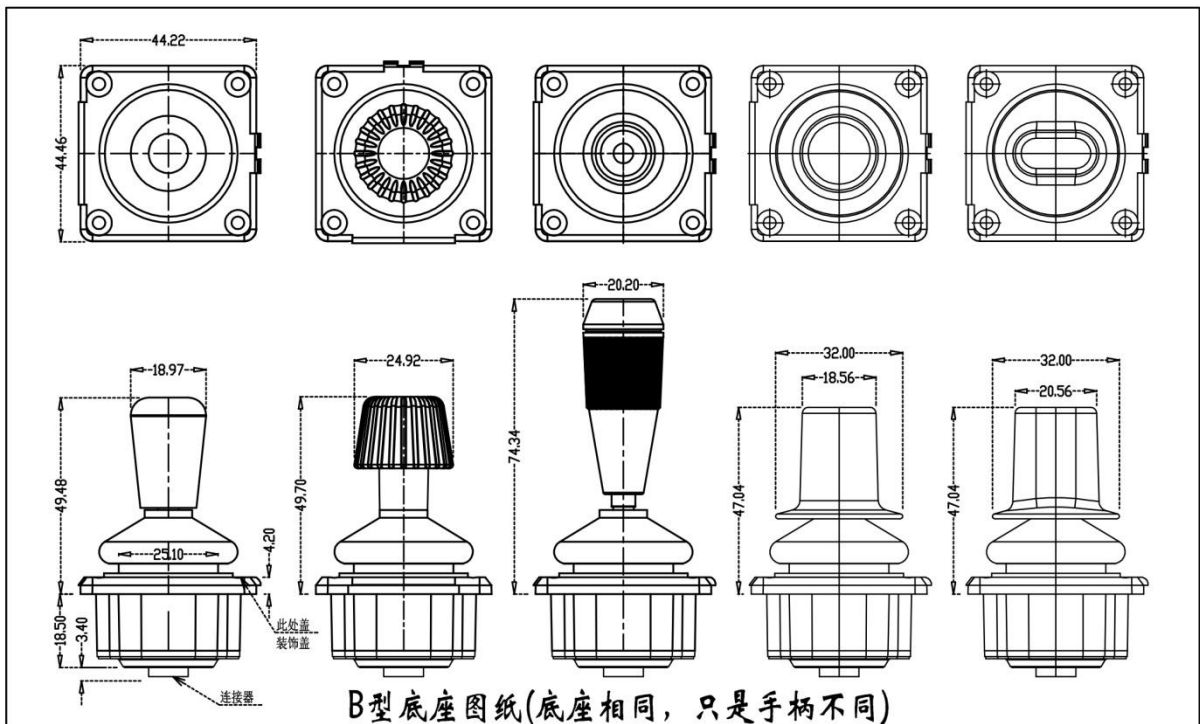
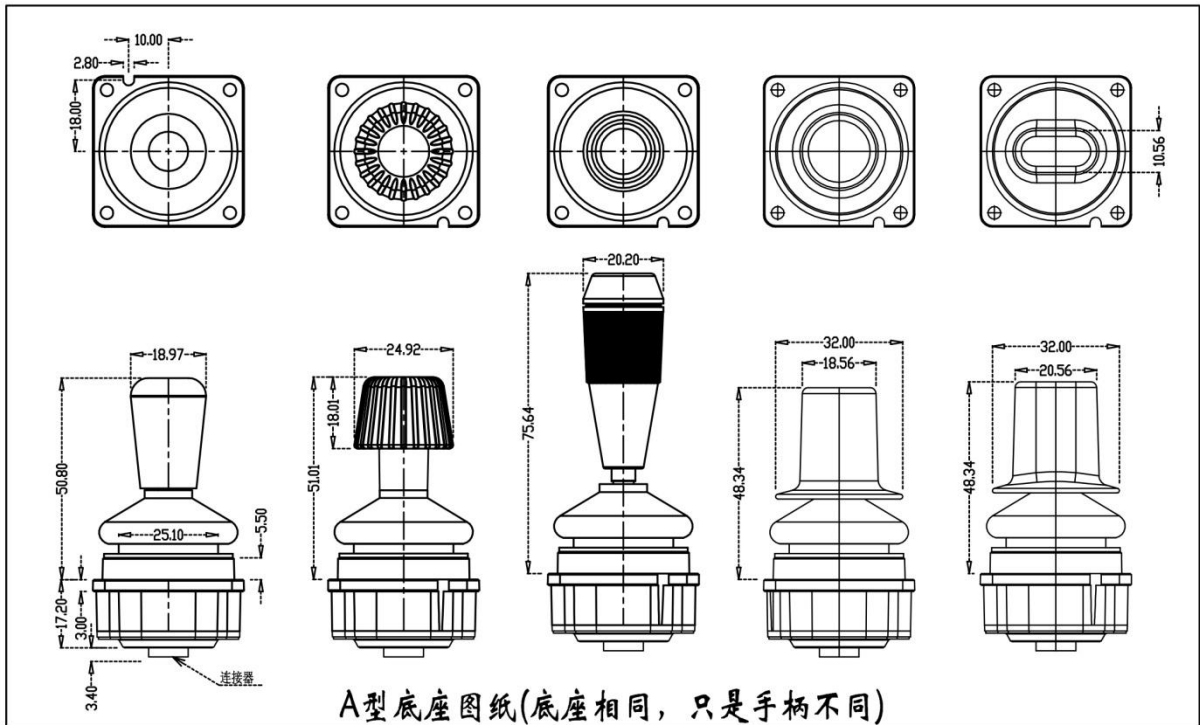
引线：

引线长度：180mm（含端子）

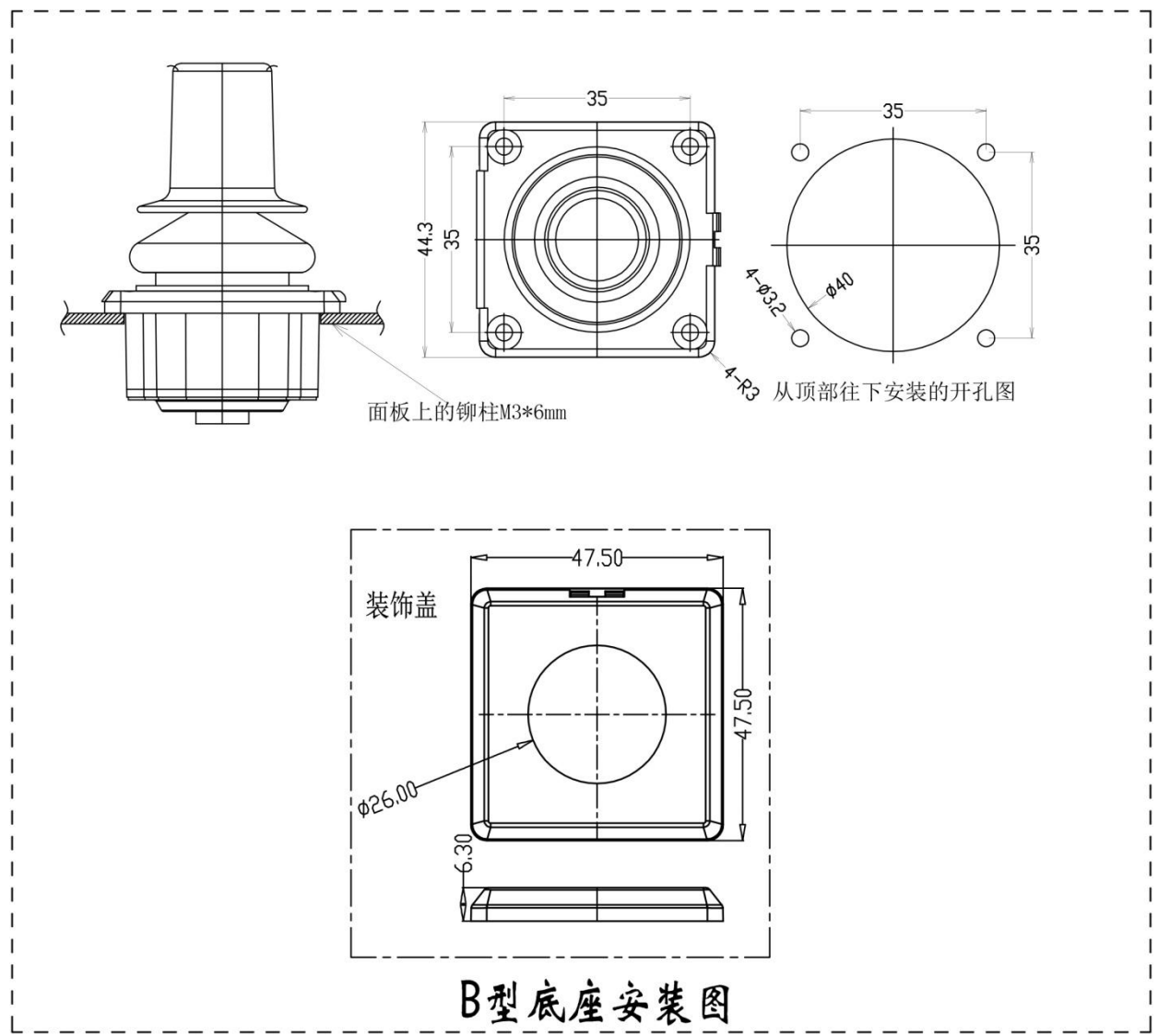
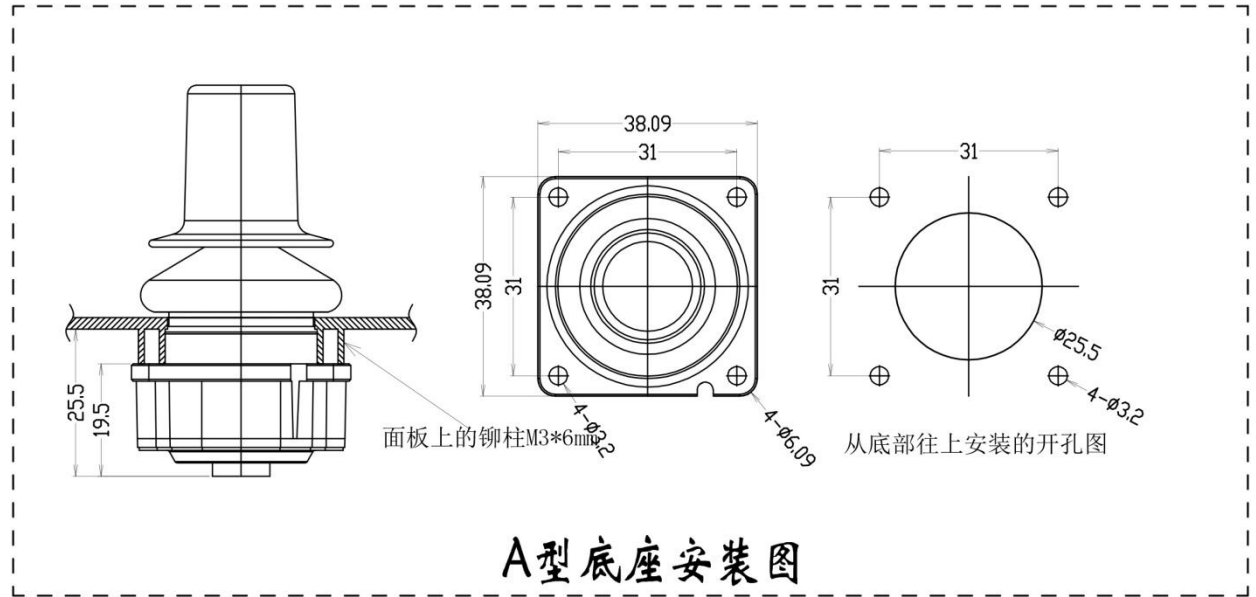
端子型号：1.25-6P



# 外形尺寸图



安装图



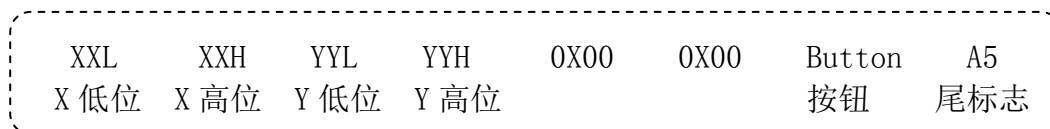


### CAN 总线通信方式:

- CAN2.0B
- 帧 ID: 标准帧 ID 和扩展帧 ID, 默认 ID=0X0101 (可通过 RS232 修改此 ID)
- 波特率: 125K, 250K, 500K, 1000K (默认 250K)
- 发送方式: 间隔 5-100ms 连续发送(默认 30ms)

操纵杆发送数据格式: (16 进制 HEX 数据)

BYTE0	XXL X 轴低位	X 轴数据 0X0020~0X0200~0X03E0
BYTE1	XXH X 轴高位	
BYTE2	YYL Y 轴低位	Y 轴数据 0X0020~0X0200~0X03E0
BYTE3	YYH Y 轴高位	
BYTE4	0X00	
BYTE5	0X00	
BYTE6	Button 按钮	按钮
BYTE7	0XA5	尾标志



YYYY Y 轴角度  
 XXXX X 轴角度  
 Button 按钮

#### X 轴参数

MAX	左	MIN	停止	MIN	右	MAX
0X0020-	-- --	0X01ff	0200	0X0201-	-- --	0X03E0

#### Y 轴参数

MAX	下	MIN	停止	MIN	上	MAX
0X0020-	-- --	0X01ff	0200	0X0201-	-- --	0X03E0

#### Button 按钮参数

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IN1	按钮 7	按钮 6	按钮 5	按钮 4	按钮 3	按钮 2	按钮 1
外部按钮 1		IN2 外部按钮 2					

button =1 有按键按下, 0 无按键按下

例如: 00 02 00 02 00 00 00 A5

# RS232/RS485 通信协议

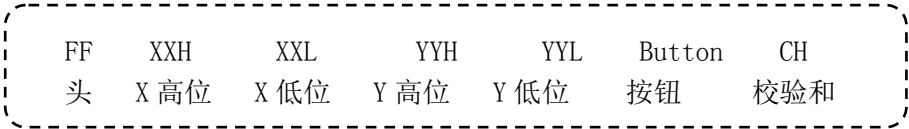
默认波特率 9600. 8. 1. N

出厂默认无地址位

功能: 发送操纵杆的各轴的位置参数

## 1、无地址位，操纵杆发送数据（7 字节）（操纵杆-PC）：

操纵杆发送数据格式：（16 进制 HEX 数据）



XXXX X 轴角度

YYYY Y 轴角度

Button joystick button 控制杆上的按钮

CH =XXH+XXL+YYH+YYL+Button 相加和的低位字节(00-FF)

X 轴参数

MAX	左	MIN	停止	MIN	右	MAX
0X0020-	-- --	0X01ff	0200	0X0201-	-- --	0X03E0

Y 轴参数

MAX	下	MIN	停止	MIN	上	MAX
0X0020-	-- --	0X01ff	0200	0X0201-	-- --	0X03E0

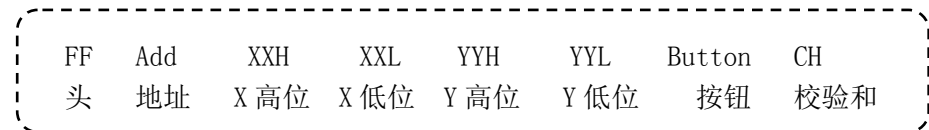
Button 按钮参数

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IN1	按钮 7	按钮 6	按钮 5	按钮 4	按钮 3	按钮 2	按钮 1
外部按钮 1		IN2 外部按钮 2					

button =1 有按键按下, 0 无按键按下

例如: FF 02 00 02 00 00 04

## 2. 带地址，操纵杆发送数据（10 字节）（操纵杆-PC）



Add 地址 0X01-0X40 与查巡的地址相同

其它与通用协议相同

CH =Add+XXH+XXL+YYH+YYL+Button (00-FF)

例如 FF 01 02 00 02 00 00 05

# Modbus 通信协议

## Modbus (RTU 模式)

### 操纵杆主站模式:

1. 波特率: 9600(可修改)
2. 数据位: 1 个起始位, 8 个数据位, 1 个停止位, 无效验位
3. 通信接口: RS485 和 RS232 只能选其一, 不能同时用
4. 数据格式: Modbus
5. 工作模式: 主站 (主站向从站 1 发送数据)
6. 工作模式: 操纵杆->从站
  - ◆ 定时发送数据, 帧间隔 17ms, 约 20HZ/帧(可设置刷新率);
  - ◆ 从站不做应答;

功能	数据	参数范围
设备地址	0x01	Modbus 站号
功能码	0x10	
第 1 个寄存器地址-高位	0x40	寄存器地址
第 1 个寄存器地址-低位	0x01	
寄存器数量-高位	0x00	
寄存器数量-低位	0x03	
数据长度	0x06	
按钮高位 Bit15-Bit8	0x00	1-16 个按钮 (Bit0=按钮 1) 1=ON, 0=OFF
按钮低位 Bit7-Bit0	0x00	
操纵杆 1 X 轴高位	0x02	0x0060-----0x07ff, 0x0800, 0x0801-----0x0fa0 (max 左 min)      停止      (min 右 max)
操纵杆 1 X 轴低位	0x00	
操纵杆 1 Y 轴高位	0x02	0x0060-----0x07ff, 0x0800, 0x0801-----0x0fa0 (max 下 min)      停止      (min 上 max)
操纵杆 1 Y 轴低位	0x00	
CRC 高位	E2	
CRC 低位	5C	

定时 25ms 发送一帧(可设置刷新率)

01 10 40 01 00 03 06 00 00 02 00 02 00 E2 5C

## 操纵杆从站模式：

### （主从方式）

1. 波特率： 9600/115200
2. 数据位： 1 个起始位， 8 个数据位， 1 个停止位， 无效验位
3. 通信接口： RS485 和 RS232 只能选其一， 不能同时用
4. 数据格式： Modbus
5. 工作模式： 从站
6. 工作模式： 主从方式（收到读指令， 回位 1 帧数据）

### Modbus 主机读取数据及从机应答格式（功能码 03）（PC→操纵杆）

字节 1	字节 2	字节 3	字节 4	字节 5	字节 6	字节 7	字节 8
地址	读命令	开始地址		寄存器数		CRC	
0x01	0x03	高	低	高	低	高	低

**实例： 01 03 40 01 00 04 00 09**

当操纵杆收到此指令， 操纵杆由主模式改为从模式。（停止主动发送数据， 只有收到这个读的数据， 操纵杆才响应如下的数据）；

### 操纵杆的响应（操纵杆→PC）

功能	数据	参数范围
设备地址	0x01	设备地址
功能码	0x03	
数据长度	0x08	
按钮高位 Bit15-Bit8	0x00	1-16 个按钮（Bit0=按钮 1） 1=ON, 0=OFF
按钮低位 Bit7-Bit0	0x00	
操纵杆 1 X 轴高位	0x02	0x0060-----0x07ff, 0x0800, 0x0801-----0x0fa0 (max 左 min) 停止 (min 右 max)
操纵杆 1 X 轴低位	0x00	
操纵杆 1 Y 轴高位	0x02	0x0060-----0x07ff, 0x0800, 0x0801-----0x0fa0 (max 下 min) 停止 (min 上 max)
操纵杆 1 Y 轴低位	0x00	
CRC 高位	21	
CRC 低位	AD	

例如：

主机→操纵杆：**01 03 40 01 00 04 00 09**

操纵杆→主机（响应）：**01 03 06 00 00 02 00 02 00 21 AD**

#### 4. USB 通信协议：USB 2.0 HID 人机介面协议标准

支持微软操作系统，免驱动；支持 directX 库  
 相关例程网上查“joystick directx input”

USB 键盘发送的数据格式 (HEX):

USB 键盘发送键盘的 3 轴操纵杆的角度参数和按键盘的状态值

byte1	byte2	byte3	byte4	byte5	byte6	byte7	byte8	byte9	byte10
XXL	XXH	YYL	YYH	RXL	RXH	RYL	RYH	BB1	BB2

X 轴 byte1 低字节 ,byte2 高字节

Y 轴 byte3 低字节 ,byte4 高字节

RX 轴 byte5 低字节 ,byte6 高字节

RY 轴 byte7 低字节 ,byte8 高字节

按钮 byte9 , byte10

XXXX: X 轴数据, 0000-03FF, (BYTE2 数据高位, BYTE1 数据低位)

0X0000-0X01FE 左

0X0200 停止

0X0200-0X03FF 右

YYYY: Y 轴数据, 0000-03FF, (BYTE4 数据高位, BYTE3 数据低位)

0X0000-0X01FE 下

0X0200 停止

0X0200-0X03FF 上

BB1:按钮

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IN1	按钮 7	按钮 6	按钮 5	按钮 4	按钮 3	按钮 2	按钮 1

BB2:按钮

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IN9	IN8	IN7	IN6	IN5	IN4	IN3	IN2

按钮 1-7 是操纵杆手柄的上的按钮部分，没有的为空位

IN1-9 是操纵杆的外部连接按钮，是用户另外安装的。

## 操纵杆通信参数设置

版本: Ver: 改 17. 11. 20

用户可能要对操纵杆的通信参数进行设置和修改 (包括 CAN, RS232, RS422);

上述所有的“参数修改”只能通过操纵杆的 RS422 接口或 RS232 接口进行, 包括 CAN 参数。

PC→操纵杆 (RS422、RS485 或 RS232) 上位机 (串口助手) 软件向操纵杆发送指令。  
(如串口助手软件没有, 可向我公司技术人员索要)

上位机 PC 如果没有 RS232 (DB9 9 针的连接) 在有一个 USB 转 RS232 的转换器 (标准的转换器, 不是 TTL 电平的转换器)。

操纵杆上的 RS422、RS485 或 RS232 通信接口, 出厂默认的波特率 9600. 8. 1. N

### 一、基本指令:

#### 1、ACK 确认 (操纵杆-PC)

AA 55 AF

表明操纵杆成功接收到地址设置指令, 并执行完成。

#### 2、设置操纵杆 ID 地址:

ID 是指 RS232/RS422 通信协议中的 ID, 或 CANopen 协议中的 ID  
(PC→操纵杆)

0xaf 0x0d 00 00 00 Add 0xf5  
头 命令 数据 1 数据 2 数据 3 数据 4 尾

Add=0x01~0x7F 地址 1-127

Add=0x00 无效 (地址=0 时, 在 RS232 或 RS422 时没有地址位)

例如:

设置 1 号地址 af 0d 00 00 00 01 f5 (HEX)

设置 2 号地址 af 0d 00 00 00 02 f5 (HEX)

操纵杆收到此指令, 执行后, 回复 ACK

#### 3、复位操纵杆 (PC→操纵杆)

0xaf 0x15 00 00 00 Add 0xf5  
头 命令 数据 1 数据 2 数据 3 数据 4 尾

Add=0x01~0x7f 地址要与操纵杆的地址一样才能复位

Add=0x00 复位所有地址的操纵杆, 任何地址都被复位

Add 范围不在 0-0x7f 无效

例如:

复位所有地址操纵杆: af 15 00 00 00 00 f5 (HEX)

复位 1 号地址操纵杆: af 15 00 00 00 01 f5 (HEX)

复位 2 号地址操纵杆: af 15 00 00 00 02 f5 (HEX)

#### 3、设置操纵杆的中心点 (用于校正中心点位置) (PC→操纵杆)

出厂时, 已经设置好, 可忽略此指令

PC 与操纵杆的 RS422 端口相连接, 波特率 9600

0xaf 0x09 00 00 00 00 0xf5

头 命令 数据 1 数据 2 数据 3 数据 4 尾

向操纵杆发送些数据, 重新设置操纵杆的停止位置(中心点)

例如: af 09 00 00 00 00 f5 (HEX)

#### 4、通信端口选择: (PC->操纵杆)

操纵杆通信端口 RS232, RS422, CAN 选其一; (出厂已经帮客户设置好了)

0xaf 0x05 XX 00 00 00 0xf5

头 命令 数据 1 数据 2 数据 3 数据 4 尾

XX=00 CAN 通信;

XX=01 RS232 通信

XX=02 RS422 通信

XX=03 RS485 通信 (标准 RS232/422/485 协议)

XX=04 RS485 Modbus RTU 通信

例如: af 05 00 00 00 00 f5 (HEX) CAN 通信

af 05 01 00 00 00 f5 (HEX) RS232 通信

af 05 02 00 00 00 f5 (HEX) RS422 通信

af 05 03 00 00 00 f5 (HEX) RS485 通信 (标准 485 协议)

af 05 04 00 00 00 f5 (HEX) RS485 Modbus RTU 通信

#### 5、刷新率设置 (PC->操纵杆)

刷新率=发送数据的帧间隔时间, 比如设置 20ms(每 20MS 向主机发送一帧数据)

0xaf 0x11 00 00 00 Ref 0xf5

头 命令 数据 1 数据 2 数据 3 数据 4 尾

Ref =0x0A~0x64 (10-100)ms, 单位是“毫秒”; (出厂默认:20ms )

设置此参数后, 复位或重启生效

例如: 设置操纵杆发送数据的刷新率 20MS(每 20MS 发送一帧数据, 1 秒发 50 次)

设置 20MS af 11 00 00 00 14 f5 (HEX)

设置 25MS af 11 00 00 00 19 f5 (HEX)

设置 33MS af 11 00 00 00 21 f5 (HEX)

设置 50MS af 11 00 00 00 32 f5 (HEX)

操纵杆收到此指令→回复 ACK→复位操纵杆

注意: 波特率较低, 相应的帧间隔时间就要长些

出厂默认:刷新率 20ms ( CAN 波特率 250K, RS232 和 RS422 波特率 9600)

## 6、通信模式（主从查询，定时自动发送, 包括 CAN 和 RS232/422 通信）（PC->操纵杆）

主从查询：操纵杆是从设备，只有收到主机的查询指令，才回送数据给主机。

定时自动发送：操纵杆开机就向主机发送数据，发送速率参考“刷新率设置”

此参数操纵杆永永储存（出厂已经帮客户设置好了）

格式：

0xaf 0x08 00 00 00 Mode 0xf5

头 命令 数据 1 数据 2 数据 3 数据 4 尾

Mode=00 定时发送

Mode=01 主从查询

例如：（PC->操纵杆）

定时发送模式 af 08 00 00 00 00 f5 (HEX)

主从查询模式 af 08 00 00 00 01 f5 (HEX)

设置成功后操纵杆返回 ACK (AA 55 AF ) (操纵杆->PC)

## 7、操纵杆信号轴的数量：（PC->操纵杆）只限厂家应用

出厂已经帮客户设置好，用户不用设置：

2 轴： af 0c 00 00 00 00 f5 (HEX)

3 轴： af 0c 01 00 00 00 f5 (HEX)

4 轴： af 0c 02 00 00 00 f5 (HEX)

## 8、查询设备信息：（PC->操纵杆）

af 20 00 00 00 00 f5 (HEX)

**操纵杆回复格式：操纵杆->PC**

FF 60 18 18 01 20 53 4D 43 36 30 41 3B  
头 型号 版本 日期 S M C 6 0 A 校验和

校验和 =除头 (FF) 外前面所有字节相加的和，取结果的低位字节

上述信息包含如下内容：

型号代码：60

版本：1.8

日期：2017-01-20

名称：TYH60A



## 二、RS232、RS422 和 RS485 的通信参数设置

### 9、设置 RS232、RS422 和 RS485 波特率(PC->操纵杆)

RS232 和 RS422 波特率一样，设置同时有效

```
0xaf  0x0b  00    00    00    Baud  0xf5
头    命令  数据 1  数据 2  数据 3  数据 4  尾
      Baud=0X00  波特率=9600
      Baud=0X01  波特率=19200
      Baud=0X02  波特率=57600
      Baud=0X03  波特率=115200
```

例如：

```
设置 9600      af 0b 00 00 00 00 f5    (HEX)
设置 19200     af 0b 00 00 00 01 f5    (HEX)
设置 57600     af 0b 00 00 00 02 f5    (HEX)
设置 115200    af 0b 00 00 00 03 f5    (HEX)
操纵杆收到此指令，执行后，回复 ACK
```

### 10、查巡操纵杆位置(PC->操纵杆)

只有当“主从查询”的能信模式时，这个指令才有效  
没有查询指令时操纵杆无任何数据输出，查一次操纵杆回一次。

```
0xaf  0x07  00    00    00    Addr  0xf5
头    命令  数据 1  数据 2  数据 3  数据 4  尾
```

- ◆ Addr 地址 = 0x01-0x7f 当地址正确时，回送
- ◆ 操纵杆收到这个数据就回送当前的位置，查一次操纵杆回一次，不查不发数据

例如 RS232 通信时查询：

```
(PC->操纵杆) af 07 00 00 00 01 f5    (HEX)
(操纵杆->PC) FF 01 08 00 70 00 00 00 00 79
操纵杆收到这个数据就回送当前的位置
```

### 三、CAN 通信的参数设置：CAN 的参数设置也要通过 RS232 或 RS422 端口；

#### 11、CAN 端口波特率：（PC->操纵杆）

0xaf	0x06	XX	00	00	00	0xf5
头	命令	数据 1	数据 2	数据 3	数据 4	尾
	XX=00	125K				
	XX=01	250K	(默认)			
	XX=02	500K				
	XX=03	1000K				
	XX=04	100K				

例如： af 06 00 00 00 00 f5 (HEX) CAN 波特率=125K  
af 06 01 00 00 00 f5 (HEX) CAN 波特率=250K (默认)  
af 06 02 00 00 00 f5 (HEX) CAN 波特率=500K  
af 06 03 00 00 00 f5 (HEX) CAN 波特率=1000K  
af 06 04 00 00 00 f5 (HEX) CAN 波特率=100K

#### 12、CAN 协议设置：（PC->操纵杆）

0xaf	0x0a	00	00	TP	SS	0xf5
头	命令	数据 1	数据 2	数据 3	数据 4	尾

SS=00 普通协议 ID=发送节点 ID, 参见(11)操纵杆发送节点 ID 设置) 默认

SS=01 CANopen 协议 ID=180+ID (参见 (2) 设置操纵杆 ID 地址)  
出厂已经帮客户设置好了

TP 是 CAN OPEN 协议中的 TPD0

TP=00: TPD01 发送 ID 0X0180+ID(参见一、2、设置操纵杆 ID 地址) 默认

TP=01: TPD02 发送 ID 0X0280+ID(参见一、2、设置操纵杆 ID 地址)

TP=02: TPD03 发送 ID 0X0380+ID(参见一、2、设置操纵杆 ID 地址)

TP=03: TPD04 发送 ID 0X0480+ID(参见一、2、设置操纵杆 ID 地址)

例如： af 0a 00 00 00 00 f5 (HEX) 普通协议  
af 0a 00 00 00 01 f5 (HEX) CANopen 协议 TPD01

#### 13、操纵杆“发送节点 ID”设置：（PC->操纵杆）

只适用于“普通协议”，CANopen 协议用不到这个指令

0xaf	0x01	D1	D2	D3	D4	0xf5
头	命令	数据 1	数据 2	数据 3	数据 4	尾

D1. 7=0 扩展帧 29 位

D1. 7=1 标准帧 11 位

- 29 位扩展帧：数据范围 0X0-0X0FFFFFFF，数据 D1-D4 对应“结点标识码”

例如： 设置发送结点标识码-扩展帧“0X00F0F101”

af 01 00 f0 f1 01 f5 (HEX)

- 11 位标准帧：数据范围 0X000-0X3FF，数据 D3-D4 对应“结点标识码”

例如： 设置发送结点标识码-标准帧“0X181”

af 01 80 00 01 81 f5 (HEX)

#### 14、操纵杆 “接收节点 ID” 设置： (PC->操纵杆)

只适用于“普通协议”，CANopen 协议用不到这个指令

0xaf 0x02 D1 D2 D3 D4 0xf5  
头 命令 数据 1 数据 2 数据 3 数据 4 尾

D1.7=0 扩展帧 29 位

D1.7=1 标准帧 11 位

- 29 位扩展帧：数据范围 0X0-0X0FFFFFFF，数据 D1-D4 对应“结点标识码”

例如： 设置接收结点标识码-扩展帧“0X00F0F101”

af 02 00 f0 f1 01 f5 (HEX)

- 11 位标准帧：数据范围 0X000-0X3FF，数据 D3-D4 对应“结点标识码”

例如： 设置接收结点标识码-标准帧“0X1E1”

af 02 80 00 01 E1 f5 (HEX)

#### 15、操纵杆 “屏蔽节点 ID” 设置： (PC->操纵杆)

0xaf 0x03 D1 D2 D3 D4 0xf5  
头 命令 数据 1 数据 2 数据 3 数据 4 尾

D1.7=0 扩展帧 29 位

D1.7=1 标准帧 11 位

- 29 位扩展帧：数据范围 0X0-0X0FFFFFFF，数据 D1-D4 对应“结点标识码”

例如： 设置屏蔽结点标识码-扩展帧“0X00002201”

af 03 00 00 22 01 f5 (HEX)

- 11 位标准帧：数据范围 0X000-0X3FF，数据 D3-D4 对应“结点标识码”

例如： 设置屏蔽结点标识码-标准帧“0X122”

af 03 80 00 01 22 f5 (HEX)

## 四、Modbus RTU (RS485) 通信的参数设置:

(修改时间: 2018-01-19)

16. 设置“设备地址“: 出厂默认 1  
参见《一、2、设置操纵杆 ID 地址》
17. 操纵杆工作模式 (PC->操纵杆):  
参见《一、6、通信模式》

当操纵杆接收到正确的“主机读取数据”指令 (01 03 40 01 00 04 00 09)  
工作模式也自动改为从站模式, 但是不永久存贮, 开机又恢复主站模式。

18. 设置寄存器地址 (PC->操纵杆):

数据格式:

0xaf	0x18	D1	D2	D3	D4	0xf5
头	命令	数据 1	数据 2	数据 3	数据 4	尾

D1: 寄存器地址高位

D2: 寄存器地址低位

(D3 和 D4=0x00)

出厂默认寄存器地址=0x4001

设置后永久贮存在操纵杆中

例如: 设置寄存器地址=0x4001(十六进制), 如果 8 进制要转成 16 进制。  
ID=0X4001      af 18 40 01 00 00 f5

## 五、USB 相关参数设置

1. USB 设置中心点指令 (通过 USB 下发指令)

f5 00 00 00 00 01 55 56

2. 设置 USB PID 的低位字节 (通过 USB 下发指令)

格式: f5 0A Pid 00 00 00 CH (CH 校验和: 2-6byte 相加和的低位字节)

例如: PID=0061      F5 0A 61 00 00 00 00 6b (默认)

PID=0062      F5 0A 62 00 00 00 00 6c

PID=0063      F5 0A 63 00 00 00 00 6d

深圳市小龙电器有限公司

[www.longcctv.com](http://www.longcctv.com)

电话: 0755-29671606

传真: 0755-29671575

EMAIL: [XL@LONGCCTV.COM](mailto:XL@LONGCCTV.COM)

技术支持: QQ: 4358032 李工 15914185789