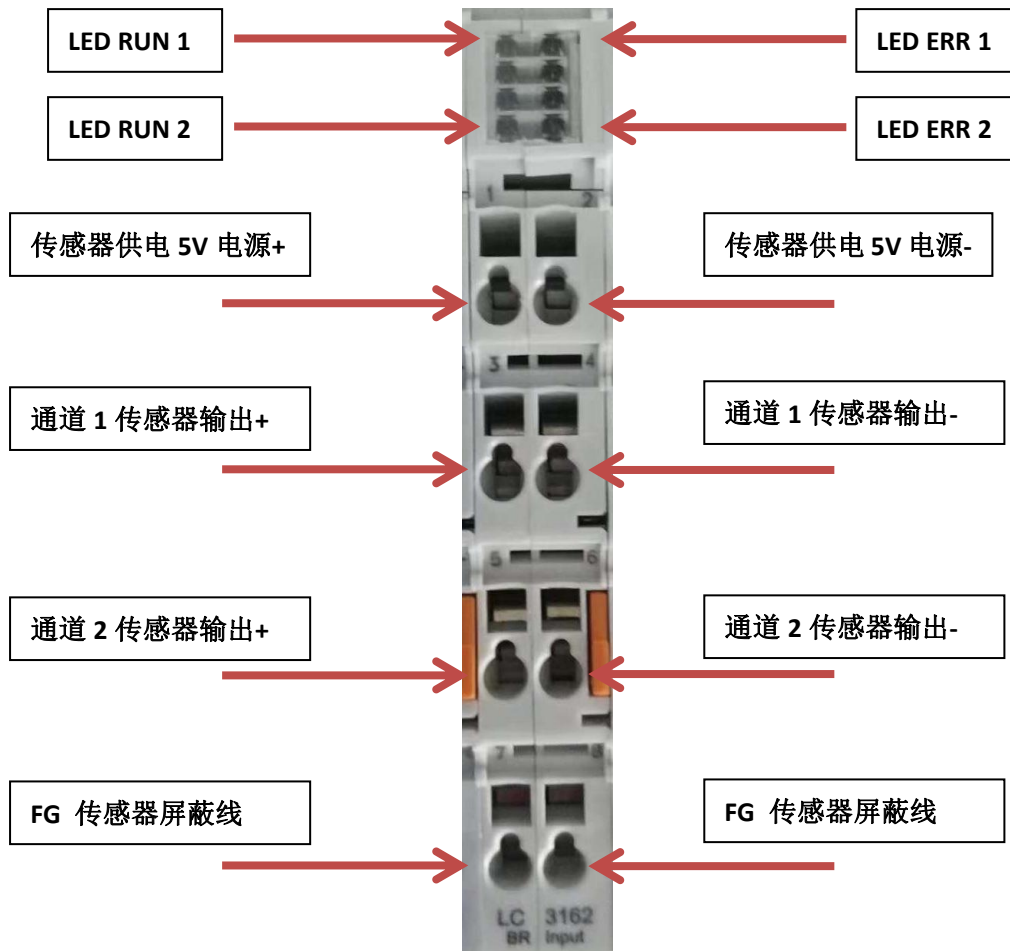
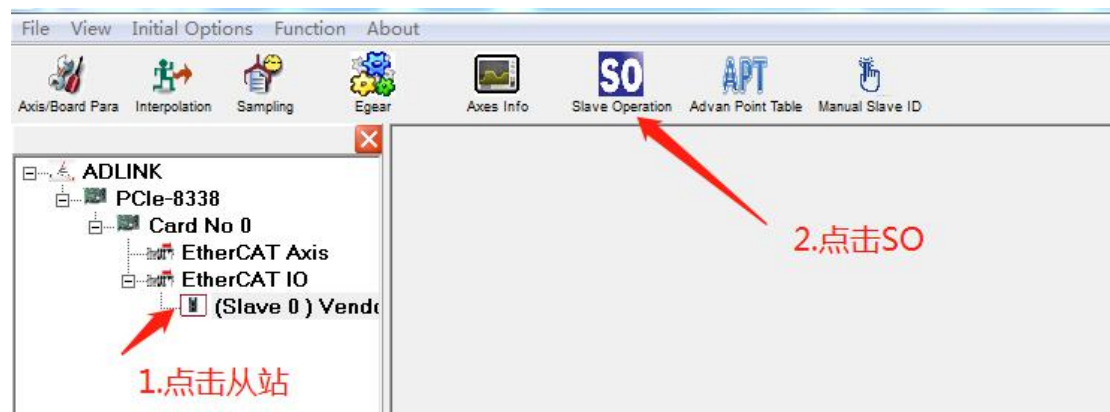


1. LC3162 接线



2. 操作说明



我们以两片 LC3162 为例,目前从站连接一个耦合器带两张 LC3162,而每个 LC3162 有两个通道。

TX (只写寄存器) :

Transmit PDO Mapping (TxPDO)					
SubModule No.	Index	Name	Data Type Number	Data Type Name	Bit Length
1	0	Module 1 (LC3162) Bridge RxPDO-Mapping BR Command	6	Unsigned16	16
2	0	Module 2 (LC3162) Bridge RxPDO-Mapping BR Command	6	Unsigned16	16

TxPDO Configuration

SubModule	Index	Value	Write

通过 SubModule 和 Index 选择不同模块的控制字进行输入. SubModule 为 1 时, 控制对象为第一片 LC3162 的控制字。 SubModule 为 2 时控制第二片的控制字。我们通过向控制字中写入协议值来进行校准、采集、停止等控制。例如 SubModule=2, index=0, value=288 (0X120 的十进制数), 即为开始采集传感器数值。

协议如下:

状态值	符号	说明	指示灯状态
0x0000	S_Idle	空闲状态	LED RUN1 与 LED RUN2 常亮
0x0100	S_Sampling	正常采样状态	LED RUN1 与 LED RUN2 闪烁
0x0101	S_1CZeroCal	1 通道零电平校准完成	LED RUN1 闪烁
0x0102	S_1CFullCal	1 通道满量程校准完成	LED RUN1 常亮
0x0103	S_2CZeroCal	2 通道零电平校准完成	LED RUN2 闪烁
状态值	符号	说明	
0x0104	S_2CFullCal	2 通道满量程校准完成	LED RUN2 常亮
0x0109	S_Caling	校准过程中	维持现状
0x0111	S_1CErr	1 通道校准出错	LED Error1 常亮
0x0112	S_2CErr	2 通道校准出错	LED Error2 常亮
0x0115	S_CommErr	错误指令	LED Error1 与 LED Error2 闪烁
0x0116	S_InitErr	从站初始化错误	LED Error1 闪烁
0x0117	S_UnCal	从站未校准	LED Error2 闪烁
0x0118	S_OverTime	访问从站超时	LED Error1 与 LED Error2 常亮
注: 未做说明即不亮。			
命令值	符号	说明	
0x0120	C_Idle	从站空闲命令	
0x000	C_Sampling	正常采样命令	
0x0121	C_1CZeroCal	1 通道零电平校准命令	
0x0122	C_1CFullCal	1 通道满量程校准命令	
0x0123	C_2CZeroCal	2 通道零电平校准命令	
0x0124	C_2CFullCal	2 通道满量程校准命令	

RX（只读寄存器）：

Recieve PDO Mapping (RxPDO)						
SubModule No.	Index	Name	Data Type Number	Data Type Name	Bit Length	Value
1	0	Module 1 (LC3162).Bridge TxPDO-Mapping.BR State	6	Unsigned16	16	0
1	1	Module 1 (LC3162).Bridge TxPDO-Mapping.BR Value CH1	6	Unsigned16	16	0
1	2	Module 1 (LC3162).Bridge TxPDO-Mapping.BR Value CH2	6	Unsigned16	16	0
2	0	Module 2 (LC3162).Bridge TxPDO-Mapping.BR State	6	Unsigned16	16	0
2	1	Module 2 (LC3162).Bridge TxPDO-Mapping.BR Value CH1	6	Unsigned16	16	0
2	2	Module 2 (LC3162).Bridge TxPDO-Mapping.BR Value CH2	6	Unsigned16	16	0
3	0	Module 3 (Alarm).Error FeedBack.Error	7	Unsigned32	32	0

RxPDO Configuration			
SubModule	Index	Value	Read
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Read"/>

SubModule 对应的是第几片 LC3162，index 对应的则是该片 LC3162 下面不同的寄存器地址。当我们需要读取第二片 LC3162 上的第二通道时，我们需要在 SubModule 中输入 2，在 Index 中输入 2。这样我们读到的就是压力值的数字量。

3. 数字量换算

我们的内部数字电路为 16 位的电压采集电路，以 5KG 的量程，2mv/v 的精度压力传感器为例。LC3162 的激励电压是 5V，那么根据公式：

$$V_{OUT} = V_{IN} \times 2\text{mv/v}$$

得到最大的输出电压是 10mv。LC3162 的内部 AD 芯片的最大量程也是 10mv。根据公式：

1. 判断数字量为正值还是负值，如果数字量 D 大于 $2^{15} \times (\frac{V_{out}}{10})$ 则为负值，小于 32768 则为正值。

2. 正值公式：

$$P_{out} = \frac{D \times P_{max}}{2^{15} \times (\frac{V_{out}}{10})}$$

3. 负值公式：

$$D_{\text{补}} = (D \text{ xor } 0111111111111111) + 1$$

$$P_{out} = \frac{(D_{\text{补}} - 2^{15} \times (\frac{V_{out}}{10})) \times (-P_{max})}{(2^{15} - 1) \times (\frac{V_{out}}{10})}$$

其中 P_{out} 是测得的压力值，D 是采集得到的数字量， P_{max} 是 Loadcell 的量程， V_{out} 是计算得到的 Loadcell 最大量程的电压， V_{admax} 是 AD 芯片的最大量程，我们这里默认

10mV。

4. C#程序编写

我们通过程序写入和读取 SO 的值，采用 `APS_set_field_bus_od_data()` 以及 `APS_get_field_bus_od_data()` 两条函数。

```
int APS168.APS_set_field_bus_od_data(int Board_ID, int BUS_No, int MOD_No, int SubMOD_No, int ODIndex, uint RawData)
```

其中主要讲一下最后三个参数：

`SubMOD_No` 即为上文讲到的 `SubModule`，一般指的是第几片模块。

`ODIndex` 即为上文讲到的 `Index`，指的是第几个寄存器。

`RawData` 就是我们读写的数据。

5. 举例说明

例：使用 8338 接 LC1100 从站带两片 LC3162，在第二片第一通道上接有 5KG,2mv/V 的 loadcell 传感器。以界面操作和程序操作为例。

1) 开始测量



SubModule	Index	Value	Write
2	0	288	Write

```
APS168.APS_set_field_bus_od_data(0, 0, 0, 2, 0, 0);
```

此时可以看到第二片 LC3162 的 led 灯不停闪烁。

2) 读取数据



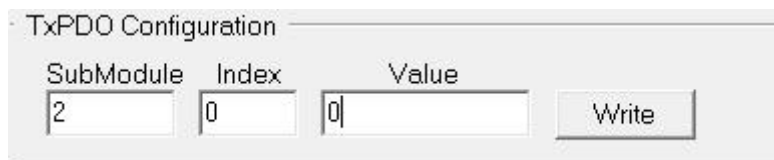
SubModule	Index	Value	Read
2	1	2249	Read

```
APS168.APS_get_field_bus_od_data(0, 0, 0, 2, 1, ref out1);
```

3) 换算数据

$$P_{\text{out}} = \frac{2249 \times 5}{2^{16} \times (1)} = 0.172 \text{ (kg)}$$

4) 关闭通道



SubModule	Index	Value	Write
2	0	0	Write

```
APS168.APS_set_field_bus_od_data(0, 0, 0, 2, 0, 0);
```

关闭后，LED 灯恢复常亮。