

## EU9单相电参数测量仪操作说明书



## 特点:

- ⊙ 精度等级为 $\pm 0.5\%F.S.$ 。
- ⊙ 同时测量电压、电流、功率因数、有功功率、无功功率、视在功率、电度7个电参数。
- ⊙ 输入/输出全隔离。
- ⊙ 具有电压、电流、功率因数、有功功率、无功功率、视在功率、电度值上下限可设定的两路报警。
- ⊙ 对电压、电流、功率因数、有功功率、无功功率、视在功率进行4-20mA变送输出。
- ⊙ RS485通信接口, Modbus RTU通信协议。
- ⊙ 真有效值测量。
- ⊙ 显示切换与菜单设置操作方便, 运行状态和电度值有掉电保护。

为了您的安全, 在使用前请仔细阅读以下内容!

### ■ 注意安全

※ 在使用前请认真阅读说明书

※ 请遵守下面的要点

⚠ 警告 如果不按照说明操作会发生意外。

⚠ 注意 如果不按照说明操作会导致产品毁坏。

※ 操作说明书中的符号说明如下。

⚠ 在特殊情况下会出现意外或危险。

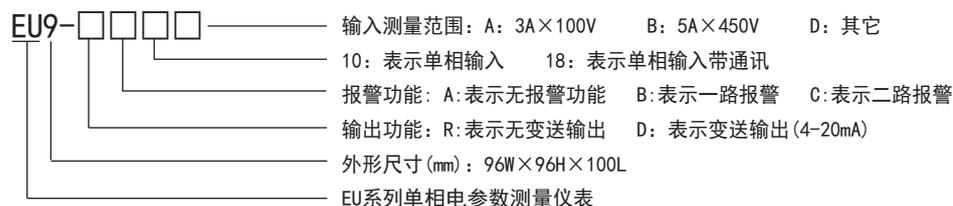
### ⚠ 警告

1. 在以下情况下使用这个设备, 如(核能控制、医疗设备、汽车、火车, 飞机、航空、娱乐或安全装置等), 需要安装安全保护装置, 或联系我们索取这方面的资料, 否则会引起严重的损失, 火灾或人身伤害。
2. 必须要安装面板, 否则可能会发生触电。
3. 在供电状态中不要接触接线端子, 否则可能会发生触电。
4. 不要随意拆卸和改动这个产品, 如确实需要请联系我们, 否则会引起触电和火灾。
5. 请在连接电源线或信号输入时检查端子号, 否则会引起火灾。

### ⚠ 注意

1. 这个装置不能使用在户外, 否则会缩短此产品的使用寿命或发生触电事故。
2. 当电源输入端或信号输入端接线时, No. 20AWG ( $0.50\text{mm}^2$ ) 螺丝拧到端子上的力矩为 $0.74\text{n}\cdot\text{m} - 0.9\text{n}\cdot\text{m}$ , 否则可能会发生损坏或连接端子起火。
3. 请遵守额定的规格, 否则会缩短这个产品的寿命后发生火灾。
4. 清洁这个产品时, 不要使用水或油性清洁剂, 否则会发生触电或火灾, 也将损坏本产品。
5. 在易燃易爆, 潮湿, 太阳光直射, 热辐射, 振动等场所应避免使用这个单元, 否则可能会引起仪表不能正常工作。
6. 在这个单元中不能有流尘或沉淀物, 否则可能会引起火灾或机械故障。
7. 不要用汽油, 化学溶剂清洁仪表外壳, 使用这些溶剂会损害仪表外壳, 请用柔软的湿布(水或酒精)清洁塑料外壳。

## 一、仪表型号

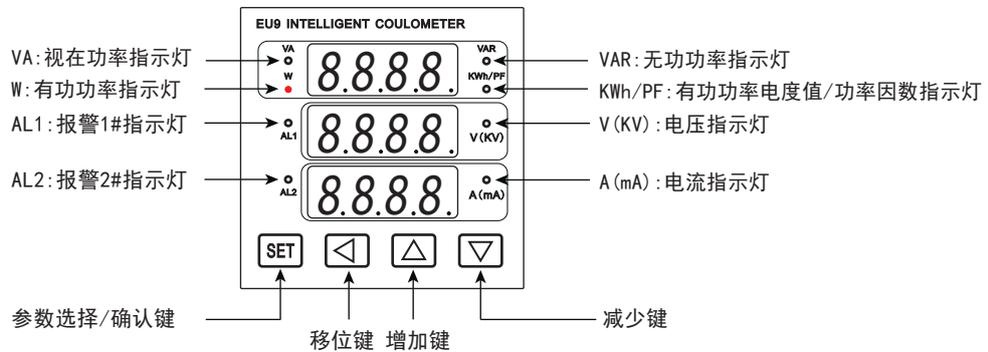


型号	报警方式	变送功能	通讯	输入范围
EU9-A10B	无	无	无	B: 5A×450V D: 其它型号 需订做
EU9-RB10B	一路	无	无	
EU9-RC10B	二路	无	无	
EU9-RC18B	二路	无	RS485	
EU9-DA10B	无	4-20mA	无	
EU9-DC10B	二路	4-20mA	无	
EU9-DC18B	二路	4-20mA	RS485	

## 二、主要技术参数

测量功能	视在功率、无功功率、有功功率、电压、电流、功率因数、电度
输入阻抗	电压输入阻抗: $\geq 300K\Omega$ (450V) 电流输入阻抗: $\leq 0.02\Omega$ (直接输入0~5A时)
直接输入量程范围	电压: AC 6-450V 电流: AC 0.015-5A
显示方式	双排数码管显示
电流变比	1.0-1999可自由设定
测量精度	电压: $\pm 0.5\%F.S \pm 2\text{Digits}$ 电流: $\pm 0.5\%F.S \pm 2\text{Digit}$
采样速度	约2次/秒
电源	AC/DC 100~240V
耐压	DC 2000V/1min
通讯	RS485通信接口, 采用Modbus RTU协议
变送输出	DC 4-20mA变送输出, 精度: $\pm 0.5\%F.S$ 负载能力 $\leq 600\Omega$
绝缘阻抗	100M $\Omega$
使用环境	0~50 $^{\circ}\text{C}$ , 相对湿度 $\leq 85\%RH$
外形尺寸(mm)	96W*96H*100L

## 三、面板名称



NO	面板文字	内容说明
1	数码管	显示测量值/参数设定内容
2	AL1	报警1#指示灯
3	AL2	报警2#指示灯
4	V (KV)	电压V指示灯 (常亮) /KV指示灯 (闪动)
5	A (mA)	电流A指示灯 (常亮) /mA指示灯 (闪动)
6	VA	视在功率指示灯 (常亮) /指示灯闪动时为KVA
7	VAR	无功功率VAR指示灯 (常亮) /指示灯闪动时为KVAR
8	KWh、PF	功率因数PF指示灯 (闪动) /有功电度值指示灯 (常亮)
9	W	有功功率W指示灯 (常亮) /指示灯闪动时为KW
10	SET	参数选择/确认键
11	◀	移位键
12	▼	减少键
13	▲	增加键

注: 面板上视在功率, 有功功率, 无功功率, 电度值/功率因数处有指示灯常亮或闪动时, 表示当前主测量窗口显示为指示灯所指示处的单位测量值

## 四、操作流程



### 参数方式设定



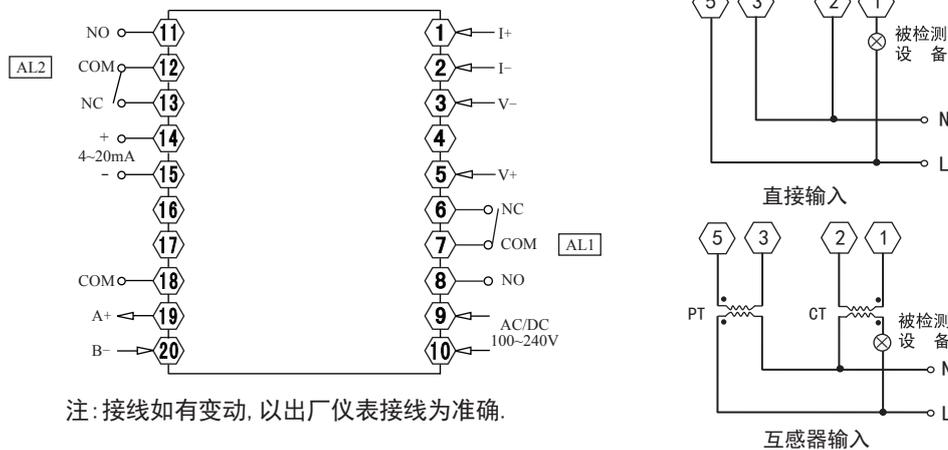
以上各菜单操作流程及设定方式说明如下:

- 1、按SET键3秒, 可进入参数设置菜单。
- 2、按SET键循环选择欲修改参数, 按◀键闪动移至所需修改值处点动▼/▲键修改为欲设参数值再按SET键确认, 在参数闪动可修改状态下, 同时按◀键+▲键可进行小数点移位操作, 若欲往下看则继续按SET键即可。
- 3、在修改设定状态下, 长按SET键>3秒, 则返回到测量状态。
- 4、电度值清零方法: 在主测量窗口显示为电度值状态下同时按SET键+▲键>3秒即可。

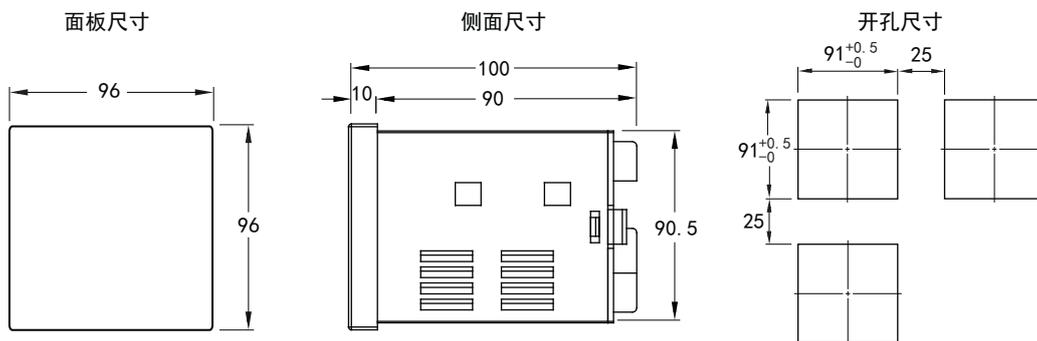
各报警方式实际显示参数对应表

NO	显示参数	上限报警	下限报警
1	电 流	RH	AL
2	电 压	UH	UL
3	功率因数	PFH	PFL
4	无功功率	Q-H	Q-L
5	视在功率	S-H	S-L
6	有功功率	P-H	P-L
7	有功电度	EHH	EHL

## 五、接线图



## 六、外形尺寸



## 七、使用保存注意事项

- 1、适宜使用环境温度0~50℃, 相对湿度85%RH以下。
- 2、本仪表校准时间间隔为一年。
- 3、注意防止震动和冲击, 不要在有超量灰尘和超量有害化学药品和气体等地方使用。
- 4、若长期存放未使用时, 请每三个月通电一次, 通电时间一次不少于4小时。
- 5、长期保存应避免直射光线, 宜存放在0~50℃, 湿度60%RH以下的地方。切勿和有机溶剂或油物接触。

## 八、通讯协议及参数

EU9 表使用 Modbus RTU 通信协议，进行 RS485 半双工通信，读功能号 0x03，写功能号 0x10，采用 16 位 CRC 校验，仪表对校验错误不返回。

**数据帧格式：**

起始位	数据位	停止位	校验位
1	8	1	无

**通信异常处理：**

异常应答时，将功能号的最高位置 1。例如：主机请求功能号是 0x04, 则从机返回的功能号对应项为 0x84。

错误类型码：

0x01——功能码非法：仪表不支持接收到的功能号。

0x02——数据位置非法：主机指定的数据位置超出仪表的范围。

0x03——数据值非法：主机发送的数据值超出超出仪表对应的数据范围。

### 1 读多寄存器

例：主机读取浮点数 AL1（第一路报警值 241.5）。

AL1 的地址编码是 0x0000，因为 AL1 是浮点数（4 字节），占用 2 个数据寄存器。十进制浮点数 241.5 的 IEEE-754 标准 16 进制内存码为 0x0080 7143。

主机请求（读多寄存器）							
1	2	3	4	5	6	7	8
表地址	功能号	起始地址 高位	起始地址 低位	数据字长 高位	数据字长 低位	CRC 码的低 位	CRC 码的高 位
0x01	0x03	0x00	0x00	0x00	0x02	0xC4	0x0B

从机正常应答（读多寄存器）								
1	2	3	4	5	6	7	8	9
表地址	功能号	数据字节 数	数据 1 高 位	数据 1 低 位	数据 2 高 位	数据 2 低 位	CRC 码的 低位	CRC 码的 高位
0x01	0x03	0x04	0x00	0x80	0x71	0x43	0x9E	0x7A

功能号异常应答：（例如主机请求功能号为 0x04）。

从机异常应答（读多寄存器）				
1	2	3	8	9
表地址	功能号	错误码	CRC 码的 低位	CRC 码的 高位
0x01	0x84	0x01	0x82	0xC0

### 2 写多路寄存器

例：主机写浮点数 HY1（第一路报警值回差 20.5）。

HY1 的地址编码是 0x0001, 因为 HY1 是浮点数 (4 字节), 占用 2 个数据寄存器。十进制浮点数 20.5 的 IEEE-754 标准 16 进制内存码为 0x0000A441。

主机请求 (写多寄存器)												
1	2	3	4	5	6	7	8	9	10	11	12	13
表地址	功能号	起始地址高位	起始地址低位	数据字长高位	数据字长低位	数据字节长度	数据 1 高位	数据 1 低位	数据 2 高位	数据 2 低位	CRC 低位	CRC 高位
0x01	0x10	0x00	0x01	0x00	0x02	0x04	0x00	0x00	0xA4	0x41	0x88	0x93

从机正常应答 (写多寄存器)							
1	2	3	4	5	6	7	8
表地址	功能号	起始地址高 8 位	起始地址低 8 位	数据字长高位	数据字长低位	CRC 码的低位	CRC 码的高位
0x01	0x10	0x00	0x01	0x00	0x02	0x10	0x08

数据位置错误应答 (例如: 主机请求写地址索引为 0x0050)。

从机异常应答 (读多寄存器)				
1	2	3	8	9
表地址	功能号	错误码	CRC 码的低位	CRC 码的高位
0x01	0x90	0x02	0xCD	0xC1

### 3 相关参数地址映射表

序号	地址映射	变量名称	字长	取值范围	读写允许	备注
0	0x0000	第 1 路报警值 AL1	2	-1999~9999	R/W	
1	0x0001	第 1 路报警回差 HY1	2	-1999~9999	R/W	
2	0x0002	第 2 路报警值 AL2	2	-1999~9999	R/W	
3	0x0003	第 2 路报警回差 HY2	2	-1999~9999	R/W	
4	0x0004	电流系数 Ct	2	-1999~9999	R/W	
5	0x0005	变送上限值 rH	2	-1999~9999	R/W	
6	0x0006	变送下限值 rL	2	-1999~9999	R/W	
7	0x0007	电压修正值 VPS	2	-1999~9999	R/W	
8	0x0008	电流修正值 APS	2	-1999~9999	R/W	
9	0x0009	电压满量程 FSV	2	0.000~9999	R	
10	0x000A	电流满量程 FSA	2	0.000~9999	R	
11	0x000B	电压值	2	0.000~9999	R	
12	0x000C	电流值	2	0.000~9999	R	
13	0x000D	功率因数	2	0.000~1.000	R	
14	0x000E	有功功率值	2	0.000~9999	R	
15	0x000F	无功功率值	2	0.000~9999	R	
16	0x0010	视在功率值	2	0.000~9999	R	

17	0x0011	电能值	2	0.000~9999	R	
保留						
20	0x0014	第1路报警模式 Ad1	1	0~13	R/W	注①
21	0x0015	第2路报警模式 Ad2	1	0~13	R/W	
22	0x0016	变送模式 brM	1	0~5	R/W	注②
23	0x0017	菜单锁定码 LCK	1	0~255	R/W	
24	0x0018	波特率 bAd	1	0~1	R	注③
25	0x0019	表地址 Add	1	0~255	R	
26	0x001A	测量状态指示	1	0~255	R	注④
27	0x001B	表名称	1	0xE9	R	
保留						

R: 只读; R/W: 可读写.

**注①: 报警模式**

上限报警	通信数值	下限报警	通信数值	报警内容
VH	0	VL	1	电压
AH	2	AL	3	电流
PFH	4	PFL	5	功率因数
P-H	6	P-L	7	有功功率
Q-H	8	Q-L	9	无功功率
S-H	10	S-L	11	视在功率
KWhH	12	KWhL	13	电度值

**注②: 变送模式**

通信数值	0	1	2	3	4	5
菜单显示	V	A	PF	P	Q	S
变送内容	电压	电流	功率因数	有功功率	无功功率	视在功率

**注③: 波特率**

通信数值	0	1
菜单显示	4.8	9.6

**注④: 测量状态指示**

D7	D6	D5	D4	D3	D2	D1	D0
有功功率	视在功率	无功功率	电流	AL1	AL2	电压	功率因数 (电度值)

**4、字节字符内码表示的浮点数转化成十进制浮点数的程序**

```
float BytesToFloat(unsigned char *pch)
{
    float result;
    unsigned char *p;
```

```

    p=(unsigned char *)&result;
    *p=*pch; *(p+1)=*(pch+1); *(p+2)=*(pch+2); *(p+3)=*(pch+3);
    return result;
}

```

### 十进制浮点数按 IEEE-754 标准转化成 4 字节字符内码表示的程序

```

void FloatToChar(float Fvalue, unsigned char *pch)
{
    unsigned char *p;
    p=(unsigned char*)&Fvalue;
    *pch=*p; *(pch+1)=*(p+1); *(pch+2)=*(p+2); *(pch+3)=*(p+3);
}

```

### 16 位 CRC 校验码获取程序

```

unsigned int Get_CRC(uchar *pBuf, uchar num)
{
    unsigned i, j;
    unsigned int wCrc = 0xFFFF;
    for(i=0; i<num; i++)
    {
        wCrc ^= (unsigned int)(pBuf[i]);
        for(j=0; j<8; j++)
        {
            if(wCrc & 1)
            {
                wCrc >>= 1; wCrc ^= 0xA001;
            }
            else
                wCrc >>= 1;
        }
    }
    return wCrc;
}

```