

高精度温湿度采集模块 EG1010



用户手册

User Manual



目录

1. 概述.....	(3)
2. 技术参数.....	(4)
3. 通信协议.....	(5)
4. 实例程序.....	(7)
5. 外形尺寸.....	(12)
6. 设备安装.....	(12)
7. 注意事项.....	(12)



概述

传感,变送一体化设计,适用于暖通级室内环境的温湿度测量。

产品的自身定位为工业级产品。对于-40℃~+120℃范围内的温,湿度测量。

- 精度: 湿度: A级: $\pm 2\%RH$ (25℃, 5...95%), 带温度补偿
B级: $\pm 3\%RH$ (25℃, 5...95%), 带温度补偿
温度: 小于 $\pm 0.5^{\circ}C$
- 数字信号输出
- 多地址并联
- 485通讯接口传输方式
- 探头为高规格集成电路
- 响应时间快
- 一体壁挂式安装,方便快捷
- 探头内置,大大提高使用寿命

广泛用于室外测温、通讯机房、智能楼宇、地铁、商场、图书馆、过程控制等。特别用于需要高精度和高温的测量场合。特别针对高温,防尘,防水要求的工业应用场合。



技术参数

供电: 直流12VDC ($\pm 10\%$)

量程: 湿度: 0~100%RH

温度: -40~+120℃

A级精度: 湿度 $\pm 2\%$ RH (5~95%RH, 25℃)

温度 $\pm 0.5^\circ\text{C}$

B级精度: 湿度 $\pm 3\%$ RH (5~95%RH, 25℃)

温度 $\pm 0.5^\circ\text{C}$

电路工作温度: -20℃~+60℃

探头工作温度: -40℃~+120℃

长期稳定性: 湿度: < 1%RH/年

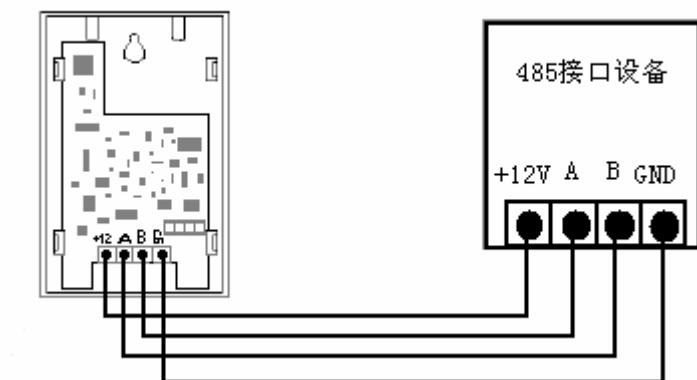
温度 < 0.1℃/年

响应时间: 湿度: < 4s (1m/s风速)

温度 < 15s (1m/s风速)

传输接口: RS485

安装方式: 壁挂式, 带安装架固定墙面





通信协议

基本格式:

序号	1	2	3	4	5	6	7	8	9
字节数	1	1	1	1	1	2	X	2	1
格式	SOI	VER	ADR	CID1	CID2	LENGT H	INFO	CHKSU M	EOI

序号	符号	表示意义	备注
1	SOI	起始标志位	(7EH)
2	VER	通讯协议版本号	**
3	ADR	设备地址描述	01-df
4	CID1	控制标识码(设备类型描述)	80
5	CID2	命令信息: 控制标识码(数据活动作类型描述)	81
6	LENGTH	INFO字节长度(包括LENID和LCHKSUM)	
7	INFO	命令信息: 控制数据信息 COMMAND INFO 应答信息: 应答数据信息 DATA INFO	
8	CHKSUM	校验和码	
9	EOI	结束码	CR (0DH)

数据格式

基本数据格式

在基本格式中的各项除SOI和EOI是以十六进制解释(SOI=7EH, EOI=0DH), 十六进制传输外, 其余各项都是以十六进制解释, 以十六进制—ASCII码的方式传输, 每个字节用两个ASCII码表示, 即高四位用一个ASCII码表示, 低四位用一个ASCII码表示。例如:

CID2=4BH, 传输时先传送34H, 再传送42H两个字节。

LENGTH数据格式

LENGTH的数据格式如下表所示。

高字节								低字节							
校验码LCHKSUM				LENID（表示INFO的传送中ASCII码字节数）											
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0



LENGTH共2个字节，由LENID和LCHKSUM组成，LENID表示INFO项的ASCII码字节数，当LENID=0时，INFO为空，即无该项。LENGTH传输中先传高字节，再传低字节，分四个ASCII码传送。

校验码的计算： $D11D10D9D8 + D7D6D5D4 + D3D2D1D0$ ，求和后模16的余数取反加1。
例如：

INFO项的ASCII码字节数为18，即LENID = 0000 0001 0010。

$D11D10D9D8 + D7D6D5D4 + D3D2D1D0 = 0000 + 0001 + 0010 = 0011$ ，模16余数为0011H，0011H取反加1就是1101H，即LCHKSUM为1101H。可得：

LENGTH为 1101 0000 0001 0010，即D012H。

CHKSUM数据格式

CHKSUM的计算是除SOI、EOI和CHKSUM外，其他字符ASCII码值累加求和，所得结果模65535余数取反加1。例：

收到或发送的字节序列是：“~1203400456ABCDFEFC72\R”，则最后五个字符“FC72\R”中的FC72是CHKSUM，计算方法是：

$$\begin{aligned} & '1' + '2' + '0' + \dots + 'A' + 'B' + \dots + 'F' + 'E' \\ &= 31H + 32H + 30H + \dots + 41H + 42H + \dots + 46H + 45H \\ &= 038EH \end{aligned}$$

其中‘1’表示1的ASCII码值，‘E’表示E的ASCII码值。038EH模65536余数是038EH，038EH取反加1就是‘FC72’。

表 1 命令信息

序号	1	2	3	4	5	6	7	8	9
字节数	1	1	1	1	1	2	LENID/2	2	1
格式	SOI	ver	adr	80H	81H	LENGTH		CHKSUM	EOI

注：LENID=00H

表.2 响应信息

序号	1	2	3	4	5	6	7	8	9
字节数	1	1	1	1	1	2	LENID/2	2	1
格式	SOI	ver	adr	80H	00	LENGTH	DATAINFO	CHKSUM	EOI

注：DATAINFO 由 RUNSTATE 组成，RUNSTATE 为运行状态。



实例程序

```
/** Beginheader SM_RecPro */
int SM_RecPro( char ComPort, char *RecBuf, unsigned int RecBufLen, unsigned int tmout );
/** endheader */

/* START FUNCTION DESCRIPTION *****/
SM_RecPro          <SU_SM.LIB>
SYNTAX:            int SM_RecPro( char ComPort, char *RecBuf, unsigned int
RecBufLen, unsigned int tmout );
DESCRIPTION:       串口类监控模块 SM 应答协议接收
                   ComPort      串口号    1=SerA 2=SerB 3=SerC 4=SerD 5=SerE 6=SerF
RETURN VALUE:      1=接收到有效协议
                   0=接收协议错误

END DESCRIPTION *****/
int SM_RecPro( char ComPort, char *RecBuf, unsigned int RecBufLen, unsigned int tmout )
{
    int nFlag,i;
    int RecByteNum;           //等待总的从接收缓存读取的字节数
                                int RecStatus;
    int tempLength;          //协议包及数据项的总长
                                int ProRecPoint;
                                unsigned short int tempCRC,recCRC;
    unsigned char FindProHead_Count; //查找协议头次数，达到一定数没有找到协议头后认为错误，
强行结束此次握手

                                FindProHead_Count = 0;
                                //等待接收协议报文
                                ProRecPoint = 0;
                                RecByteNum = SM_PACKAGE_HEAD_LEN; //设置开始接收的协
议头的字节数

                                RecStatus = 0;
                                while(1)
                                {
                                    if ( (RecByteNum > (RecBufLen-ProRecPoint)) || (RecByteNum < 0) )
                                    {
                                        #ifdef EW15TSU_DEBUG_SMPRO
                                            //程序不应执行到这
                                            printf(".....Program error,SMrecnum=%d\n",RecByteNum);
                                        #endif
                                    }

                                }

    //
    nFlag = -1;
    nFlag = SerDRead( RecBuf+ProRecPoint, RecByteNum, tmout );
                                if ( FindProHead_Count >= 10 )//多次找不到协议头则强行结束本次握手
```



```
{
    break;
}
if ( nFlag != RecByteNum )
{
    if (nFlag == WAITTIMEOUT)
    {
        //接收任务在设定的时间内没有接收到协议报文
        break;
    }
}

if ( nFlag < 0 )
{
    break;
}

}

ProRecPoint = ProRecPoint + RecByteNum;
//分析接收协议报文

for (;;)
{
    if (RecStatus == 0)
    {
        //找 SM 协议包的协议头
        nFlag = SMPRO_FindProHead(RecBuf, &ProRecPoint);
        if (nFlag >= 0)
        {
            //找到协议头
            tempLength = nFlag;
            RecStatus = 1;
            if ( ProRecPoint >= tempLength )
                RecByteNum = 0; //已经接收到了整条协议报文
            else
                RecByteNum = tempLength - ProRecPoint; //设置
还需要接收的字节数

            //
            if ( (tempLength <= 0) || (tempLength > RecBufLen) )
            {
                //如果 tempLength 超出协议可能的最大长度，则重找协议头
                *RecBuf = 0;
                RecStatus = 0;
            }
            else
                break;
        }
    }
}
```




```
else
{
    //没找到协议头，还需接收数据
    RecStatus = 0;
    RecByteNum = abs(nFlag); //设置还需要接收的字节数
    FindProHead_Count++;
    break;
}
}
else
{
    //判断整条协议的完整性
    if ( *(RecBuf+tempLength-1) == SM_EOI )
    {
        //找到对应的结束符，检查校验和
        nFlag = tempLength - 1 - 4 - 1;
        tempCRC = PRO_chksum( RecBuf+1, nFlag );
        nFlag = ASCII_to_HEX( RecBuf+tempLength-4-1 );
        recCRC = nFlag * 0x100;
        nFlag = ASCII_to_HEX( RecBuf+tempLength-4-1+2 );
        recCRC = recCRC + nFlag;
        if ( recCRC != tempCRC )
        {
            //校验和错误，重找协议头
            *RecBuf = 0;
            RecStatus = 0;
        }
        else
        {
            //协议语法和校验码都正确，进行接收协议报文处理
            RecStatus = 2;
            break;
        }
    }
}
else
{
    //没有对应的结束符，重找协议头
    *RecBuf = 0;
    RecStatus = 0;
}
}
} //endfor 分析接收协议报文
if (RecStatus == 2)
```



```
        break;
    }//endwhile

    if ( RecStatus == 2 )

        return 1;
    else
        return 0;
}

-----
if ( tSMSet->DeviceType == enumENVIRONMENT )
{
    tBufOffset = tBuf+SM_PACKAGE_HEAD_LEN+2;
    //温度传感器数量
    tCount = ASCII_to_HEX( tBufOffset );
    tBufOffset = tBufOffset + 2;
    for (i=0;i<tCount; i++)
    {
        tFchar[0] = ASCII_to_HEX( tBufOffset );
        tBufOffset = tBufOffset + 2;
        tFchar[1] = ASCII_to_HEX( tBufOffset );
        tBufOffset = tBufOffset + 2;
        tFchar[2] = ASCII_to_HEX( tBufOffset );
        tBufOffset = tBufOffset + 2;
        tFchar[3] = ASCII_to_HEX( tBufOffset );
        tBufOffset = tBufOffset + 2;
        memcpy( &tTempFloat, &tFchar[0], 4);
    }
    if ( SM485_RunBuf->Pro_cstate == 0 )
        tTempFloat = tTempFloat - 3;
    tIDLong = ID_GetSP( tSMSet->ID, 0x100 + i );
    SM_SPSample( tIDLong, C_SPTAIC, SM_THSTATUS_RS485, &tTempFloat );
    #ifdef EW15TSU_DEBUG_RS485
    printf("Temp=%.2f\n", tTempFloat);
    #endif
}
//湿度传感器数量
tCount = ASCII_to_HEX( tBufOffset );
tBufOffset = tBufOffset + 2;
for (i=0;i<tCount; i++)
{
    tFchar[0] = ASCII_to_HEX( tBufOffset );
    tBufOffset = tBufOffset + 2;
    tFchar[1] = ASCII_to_HEX( tBufOffset );
    tBufOffset = tBufOffset + 2;
    tFchar[2] = ASCII_to_HEX( tBufOffset );
```



```
tBufOffset = tBufOffset + 2;
tFchar[3] = ASCII_to_HEX( tBufOffset );
tBufOffset = tBufOffset + 2;
memcpy( &tHumiFloat, &tFchar[0], 4);

if ( SM485_RunBuf->Pro_cstate == 0 )
{
    tHumiFloat = tHumiFloat*100/3;

    //tHumiFloat = tHumiFloat - (tTempFloat-25)*0.4;
    if (tHumiFloat<0)
        tHumiFloat = 0;
    if (tHumiFloat>100)
        tHumiFloat = 100;
}

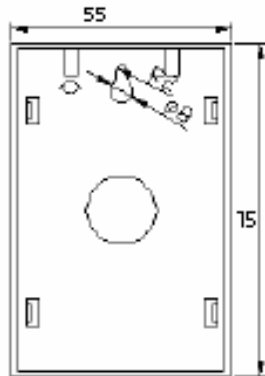
tIDLong = ID_GetSP( tSMSet->ID, 0x200 + i );
SM_SPSample( tIDLong, C_SPTAIC, SM_THSTATUS_RS485, &tHumiFloat );
#ifdef EW15TSU_DEBUG_RS485
    printf("Humi=%.2f\n", tHumiFloat);
#endif
}

sys_selfcheck.sys_envi    =    sys_selfcheck.sys_envi    &
~SYS_ENVI_LINK;

}
```



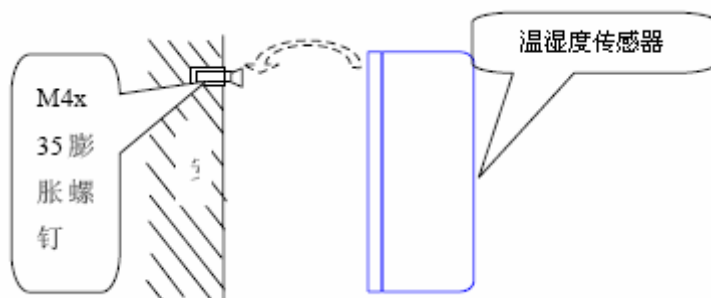
外形尺寸



底板及外形尺寸图

ABS 白色 75x55x28mm

设备安装



注意事项

- 1、使用前请认真阅读说明书、确保接线正确：任何错误接线均有可能对变送器造成不可逆伤害。
- 2、安装使用避免在易于传热且直接造成与待测区域温差地带安装，否则会造成温湿度测量不准确。
- 3、安装在环境稳定的区域,避免直接光照,远离窗口,空调,暖气等设备,避免直对窗口、房门
- 4、安装高度为人体坐高或主要要求环境条件的设备高度
- 5、防止化学试剂,油、粉尘等直接侵害传感器,勿在结露,结冰,高温下使用。请勿进行冷、热冲击。
- 6、传感器作为计量器具,检定周期为一年,请在检定后按修正值使用。