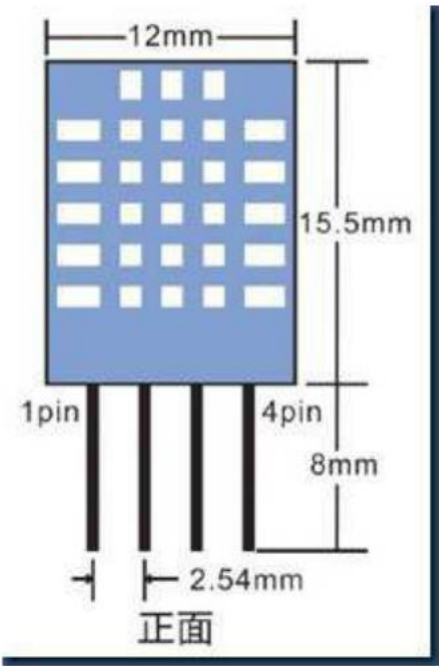


DHT11 是数字的湿度传感器... 6 元一个买的... 与它的通信方式类似于 I2C 总线通信.



Pin	名称	注释
1	VDD	供电 3—5.5VDC
2	DATA	串行数据，单总线
3	NC	空脚，请悬空
4	GND	接地，电源负极

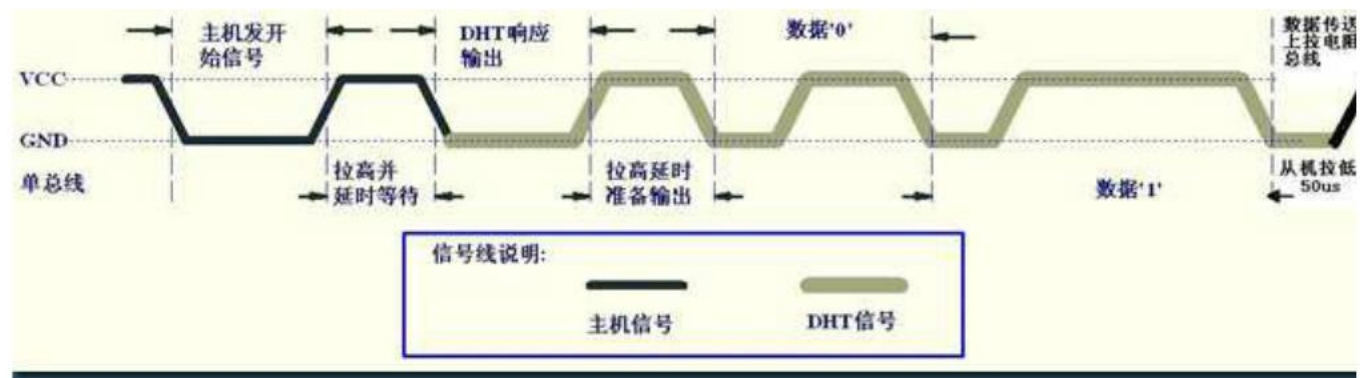
串行数据一次返回 40 位.

数据格式:8bit 湿度整数数据+8bit 湿度小数数据
+8bi 温度整数数据+8bit 温度小数数据
+8bit 校验和

数据传送正确时校验和数据等于“8bit 湿度整数数据+8bit 湿度小数数据
+8bi 温度整数数据+8bit 温度小数数据”所得结果的末 8 位。

如果模块离芯片距离近, 建议在数据线接一 5k 电阻上拉.

时序图如下:



第一次拉低要求至少 18ms, 之后拉高 20-40us, 之后 DHT 拉低 80us 后再拉高 80us. . 之后进入数据阶段. 每个数据, 0, 1 以高电平的时间长短区分. 0 数据高电平时间为 26-28us, 1 为 70us. . 不论 1 或者 0, 之前都先输出 50us 的低电平. . 数据传输完成后拉高总线.

在此次的程序中用 P2.1 作为信息口.

程序如下

```
#include <msp430x14x.h>
#define HIGH P2OUT|=BIT1;
#define LOW P2OUT&=~BIT1;
void DelayNus(int n)
{
    CCR0 = n;
    TACTL |= MC_1; //增计数到 CCR0
    while(!(TACTL & BIT0)); //等待
    TACTL &= ~MC_1; //停止计数
    TACTL &= ~BIT0; //清除中断标志
}
char receive(void) //接受函数
{
    char tem,cnt=0; //临时变量用于存储接受数据
    while(cnt<8)
    {
        while(!(P2IN&BIT1)); //等待 50us 的低电平结束
        DelayNus(45);
        if(P2IN&BIT1) //长于 30us 定义为 1
        {
            tem++;
            tem<=1;
            while(P2IN&BIT1); //结束高电平
        }
        else
```

```

{
tem<<=1;
}
if(cnt!=7)
while(!(P2IN&BIT1)); //最后一次给函数返回留下时间
cnt++;
}
return tem;
}

```

```

int main( void )
{
int i;
char temph, templ, humdh, humdl, check, cal;
WDTCTL=WDTPW+WDTHOLD;
_DINT();
BCSCTL1&=~XT2OFF;
do
{
IFG1&=~OFIFG;
for(i=0xff;i>0;i--);
}
while(IFG1&OFIFG);
_EINT();
BCSCTL2|=SELM_2+SELS; //SMCLK 8 分频
P2DIR|=BIT1;
P2OUT|=BIT1;
TACTL|=TASSEL_2+ID_3;
//inited
LOW;
DelayNus(18000);
HIGH;
DelayNus(30);
P2DIR&=~BIT1;
while(P2IN&BIT1);
while(!(P2IN&BIT1));
while(P2IN&BIT1);
//Data comes
humdh=receive();
humdl=receive();
temph=receive();
templ=receive();
check=receive();
cal=humdh+humdl+temph+templ;

```

```
if (check==cal)
{
LPM0;
}
```

```
return 0;
}
```