

```

#include "HT66F018.h"
#define u8 unsigned char
#define u16 unsigned int
#pragma vector Timer0_isr @ 0x0C //用於定时
void Write_EEDATA(u8 addr, u8 data);
u16 Get_AD();
u8 read_eeprom(u8 addr);
#pragma rambank0
bit flagon ;
#pragma norambank
void delay(long t);
void sendchar(u8 ch);
void DIS(unsigned int n,unsigned int m);
const unsigned char led[10]={0xfc,0x60,0xda,0xf2,0x66,0xb6,0xbe,0xe0,0xfe,0xf6};
#define DAT _pb6
#define CLK _pb4
#define wei1 _pa3
#define wei2 _pa2
#define wei3 _pa1
#define wei4 _pa0
#define wei8 _pa4
#define wei7 _pa5
#define wei6 _pa6
#define wei5 _pa7
u16 m,n,t,ii;
u8 AD_Count,num,DIAN,BUFFER;
unsigned long AD_NEW;
u16 AD_Mun[10];
u16 temp,temp1;
u16 AD_Last,MAXADdianya,MAXADdianliu,XSdianya,XSdianliu;
u16 AD_Value,ADH,ADL;

void Init()
{
    _wdtc=0b10101000; //关闭看门狗
    _cpc = 0x08; //关闭比较器

    for(_tblp=0xff;_tblp>0x9f;_tblp--) //上电清除所有 RAM
    {
        _mp0=_tblp;
        _iar0=0;
    }
    for(_tblp=0xff;_tblp>0x9f;_tblp--)
    {

```

```

    _bp = 1;
    _mp1 = _tblp;
    _iar1 = 0;
}

_t2cp = 1; //设置 PWM 输出脚 pb3, 第 12 脚
_tm2c0 = 0b00010000; //SYS 默认为 8M
_tm2c1 = 0b10100000; //PWM 模式或单脉冲输出模式
_tm2rp = 2; //频率为 8M/ (256*2)
_tm2al = read_eeprom(0x01);
_tm2ah = read_eeprom(0x00); //占空比为(_tm2ah*256+_tm2ala)/ (256*2)
_emi = 1; //总中断控制位
_t2on = 1; //开始计数

_pbc3 = 0; //太重要了!!!, 要不 PWM 的输出极性总在变化
_pbc6 = 0;
_pbc4 = 0;
_pac = 0x00;
_papu = 0xff;
_pa = 0xff; // 熄灭所有

_pbc5 = 0; //控制 3843 的电源光耦, 输出为 1 时 3843 才有电
_pbpu5 = 1;
_pb5 = 0;

_pbc2 = 1; //相当于开关
_pbpu2 = 1;

_pcc = 0xff; //数码电位器的 AB 片接点, 为输入组合数据
_pcpu = 0xff;

_adcr0 = 0b00010001; //AD 通道 0 高字节 8 位+低字节高四位保存 AD 值
_adcr1 = 0b00000001; //内部 ADC 电源作为参考电压
_acerl = 0b00000011; //选择 AN0,AN1

    _tm0c0 = 0b00000000; //SYS1/4 默认为 8M/4
    _tm0c1 = 0b11000001; //定时器, 且与比较器 A 比较
    _tm0al = 0xf5;
    _tm0ah = 0x0f;
    _t0ae = 1; //比较器 A 中断允许使能
    _mf0e = 1; //多功能中断允许使能
    _emi = 1; //总中断允许使能
    _t0on = 1; //定时器开始计数器使能
}

```

```

void main()
{
    u16 numm;
    u8 k,ch,i;
    bit flagon;
    Init();
    ii=read_eeprom(k)*256+read_eeprom(k+1);//取出存储的数值
    BUFFER=0x00;
    k=0;
    numm=0;
    flagon=0;

    MAXADdianya=Get_AD();
    _acs0=0;
    MAXADdianliu=Get_AD();

    while(1)
    {
        if(!flagon)
        {
            _pb5=0;
            _pa=0xff;
            _t0on=0;
        }
        else
        {
            _pb5=1;
            _t0on=1;
            _acs0=1;
            AD_NEW = Get_AD();
            if((AD_NEW-5)>MAXADdianya || (AD_NEW+5)<MAXADdianya)
            {
                MAXADdianya=AD_NEW;//数据较上次变化较大，更新数据
                XSdianya=5212*AD_NEW/4096;
            }
            _acs0=0;
            AD_NEW = Get_AD();
            if((AD_NEW-5)>MAXADdianliu || (AD_NEW+5)<MAXADdianliu)
            {
                MAXADdianliu=AD_NEW;//数据较上次变化较大，更新数据
                XSdianliu=581*AD_NEW/4096;
            }
            if(XSdianliu>580 || XSdianya>2600)//电流不大于 5.8 安，电压不超过 26 伏
            {
                _pb5=0;
                ii=4;
            }
        }
    }
}

```

```

        _tm2al=ii&0x00ff;
        _tm2ah=ii>>8;
        flagon=0;
    }
    if(_pc2==0)//电位器自带按钮开关
    {    delay(1000);
        if(_pc2==0)
        {    while(_pc2==0)
            {    numm++;
                delay(15000);
            }
            if(numm<20)//判断是短按
            { k=k+2;
                if(k>8)
                k=0;
                ii=read_eeprom(k)*256+read_eeprom(k+1);
                _tm2al=ii&0x00ff;
                _tm2ah=ii>>8;
            }
            else//判断是长按,返回到一个固定的值
            {    Write_EEDATA(k, ii>>8);
                Write_EEDATA(k+1, ii&0x00ff);
                delay(1000);
            }
            numm=0;//判断是长按还是短按后, num 返回初始值。
        }
    }
    DIS(XSdianya,XSdianliu);
}

if(_pb2==0)
{    delay(1000);
    if(_pb2==0)
    {    while(_pb2==0);
        flagon=~flagon;
    }
}

}

void sendchar(u8 ch)
{    u8 i;
    for(i=0;i<8;i++)//8 位数据传输完毕后才给数码管显示
    {    DAT=ch&0x01;

```

```

        CLK=0;
        CLK=1;
        ch>>=1;
    }
}
void DIS(unsigned int n,unsigned int m)
{ if(n/1000>0)
    {sendchar(led[n/1000]);
      wei4=0;
      _delay(1000);
      _pa =0xff ;// 熄灭所有
    }

    sendchar(0x01);
    wei4=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

    sendchar(led[n/100%10]);
    wei3=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

    sendchar(led[n/10%10]);

    wei2=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

    sendchar(led[n%10]);

    wei1=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

    //电流
    wei1=1;
    if(m/1000>0)
    {sendchar(led[m/1000]);
      wei8=0;
      _delay(1000);
      _pa =0xFF ;// 熄灭所有
    }
}

```

```

sendchar(0x01);
    wei8=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

sendchar(led[m/100%10]);
    wei7=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

sendchar(led[m/10%10]);

    wei6=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有

    sendchar(led[m%10]);

    wei5=0;
    _delay(1000);
    _pa =0xFF ;// 熄灭所有
}
void Write_EEDATA(u8 addr, u8 data)
{
    _eea = addr;
    _eed = data;
    _emi = 0;
    _clrwdt();
    #asm
    set [04].0
    mov a,40h
    mov [03],a
    set [02].3
    set [02].2
    sz  [02].2
    jmp $-1
    #endasm
    _clrwdt();
    _emi = 1;
    _eec = 0;
}

u8 read_eeprom(u8 addr)

```

```

{
    _eea = addr;
    _emi = 0;
    _clrwdt();
    #asm
    set [04].0
    mov a,40h
    mov [03],a
    set [02].1
    set [02].0
    sz [02].0
    jmp $-1
    #endasm
    _clrwdt();
    _emi = 1;
    _eec = 0;
    return _eed;
}
void delay(long t)
{
    while(t--);
}
u16 Get_AD()
{
    u8 i,j,AD_Count;
    u16 AD;

    for(AD_Count=0;AD_Count<10;AD_Count++)
    {
        _delay(10); //适当延时

        _start=0;
        _start=1;
        _start=0; //启动 AD 转换
        while(!_eocb); //等待转换完成
        ADL= _adrl;
        ADH= _adrh;
        //AD_Mun[AD_Count]=((u16)ADH<<4) + ((ADL>>4)&0x0f); //处理 AD 的结果
        AD_Mun[AD_Count]=ADL+ADH*256;

    }

    for(j=1;j<10;j++) //从小到大排序 冒泡排序

```

```

    {
        for(i=0;i<10-j;i++)
        {
            if(AD_Mun[i]>AD_Mun[i+1])
            {
                temp=AD_Mun[i];
                AD_Mun[i]=AD_Mun[i+1];
                AD_Mun[i+1]=temp;
            }
        }
    }
    AD=AD_Mun[4];                //取第 5 个值(中值)
    return AD;

}

void Timer0_isr()                //0.56ms
{
    if((BUFFER&0x03)!=(0x03&_pc))//数码电位器有转动时
    {
        BUFFER=(BUFFER<<2)|(0x03&_pc);
        if(BUFFER==0x87)//往左旋转时
        {
            ii=ii-2;
            if(ii<2)
                ii=512;
            _tm2al=ii&0x00ff;
            _tm2ah=ii>>8;
        }
        if(BUFFER==0x4b)//往右旋转时
        {
            ii=ii+2;
            if(ii>512)
                ii=2;
            _tm2al=ii&0x00ff;
            _tm2ah=ii>>8;
        }
    }
    _t0af = 0;
}

```