

KF8F3132——CMP2 样例程序

引言

本应用笔记提供了 KF8F3132—CMP2 相关的配置信息以及如何能够快速的理解并上手使用该模块的一些配置方式。

本应用笔记须与 KF8F3132 数据手册结合使用。

C2CAL（比较器 2 校准寄存器）

寄存器13.6: C2CAL: 比较器2校准寄存器（116H）

复位值 0000 0000		bit7						bit0	
		C2DT7	C2DT6	C2DT5	C2DT4	C2DT3	C2DT2	C2DT1	C2DT0
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

C1CTL（比较器 1 控制寄存器）

寄存器13.1: C1CTL: 比较器1控制寄存器(地址: 19H)

复位值 0000 0000		bit7						bit0	
		C1OE	C1MOD2	C1MOD1	C1MOD0	C1CALEN	C1CALSET	C1OUT	C1EN
		R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

C2CTL（比较器 2 控制寄存器）

寄存器13.5: C2CTL: 比较器2控制寄存器(地址: 112H)

复位值 0000 0000		bit7						bit0	
		C2M3	C2M2	C2M1	C2M0	C2CALEN	C2CALSET	C2MOD2	C2EN
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

C2FILTCTL（比较器 2 滤波控制寄存器）

寄存器13.7: C2FILTCTL: 比较器2滤波控制寄存器（14CH）

复位值 0000 0000		bit7						bit0	
		INV2	WEN2	FEN2	-	-	F2CNT2	F2CNT1	F2CNT0
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

C2FILTPRE（比较器 2 滤波采样时钟分频寄存器）

寄存器13.8: C2FILTPRE: 比较器2滤波采样时钟分频寄存器(地址: 14DH)

复位值 0000 0000		bit7						bit0	
		FP27	FP26	FP25	FP24	FP23	FP22	FP21	FP20
		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

图注: R=可读 W=可写 -=未用 U=未实现位

位使用说明:

8 位单片机支持对寄存器的位进行直接的操作，因此在使用的过程中不仅可以通过给寄存器赋值来达到想要的配置，同时还可以直接对位进行操作来达到需要的配置。

以下是对程序中使用到的位进行说明：

T0IF（T0 中断标志位）

C2CALEN（比较器 2 校准功能使能位）

C2CALSET（比较器 2 校准设置位）

C2OUT（比较器 2 输出控制位）

C2EN（比较器 2 使能位）

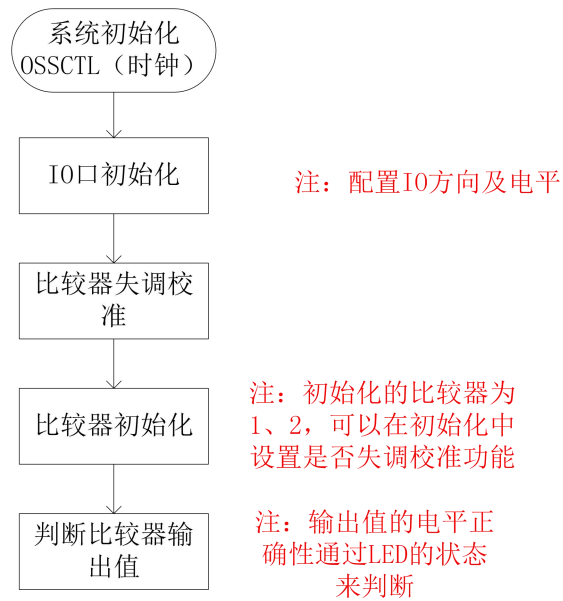
C2IE（比较器 2 中断使能位）

PUIE（外设中断使能位）

AIE（全局中断使能位）

C2IF（比较器 2 中断标志位）

CMP2 比较器样例程序框图



注：失调校准的程序代码及校准方式有固定的格式，可以直接使用。

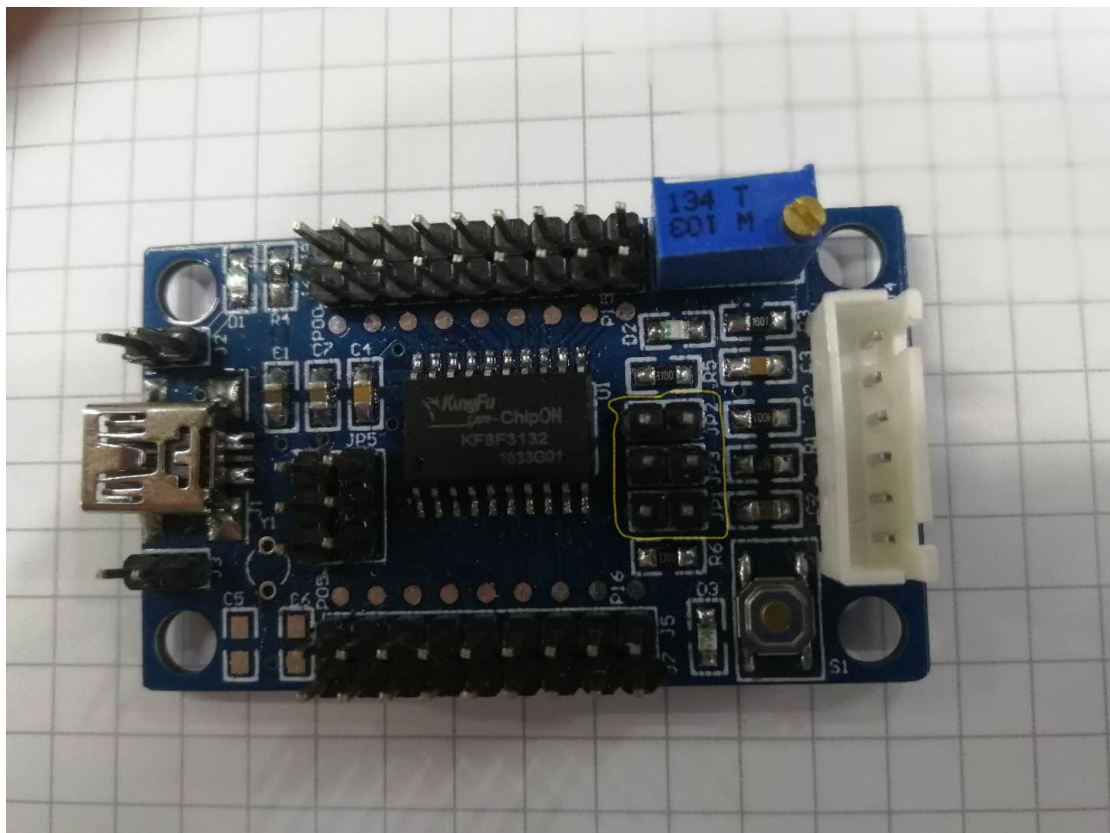
CMP2 比较器样例简述:

开发工具: ChipON IDE

功能简述: 比较器 1 负端配置为 I/O 口, 轮询比较器的结果并在 P02 输出比较器结果。比较器 2 负端配置为内部参考电压, 比较器的结果发生变化是, 触发中断, 比较结果在 P14 输出。

实验现象: 将 P10 和电位器的输出脚相连, 调节电位器, 如果电位器的电压大于 $0.5V_{DD}$, LED2 点亮, P14 输出高电平。反之, LED2 熄灭, P14 输出低电平。P00 接 VDD、P01 接 GND, 比较器 1 的输出脚 P02 将输出高电平, LED3 点亮。P00 接 GND、P01 接 VDD, 比较器 1 的输出脚 P02 将输出高电平, LED3 熄灭。

硬件连接: 连接 JP1、JP2、JP3 (图片中黄色框中的插针接跳线帽)



CMP2 比较器样例程序:

MCU 初始化:

```
void Init_fun()
{
    OSCCTL = 0x70;    //晶振设置为16M

    OPTR=0x08;//T0不分频

    TR12 =0;
    LED2=0;
    TR11=0;
    LED3=0;
}
```

延时函数:

```
void Delay_30us()
{
    T0=136;
    T0IF=0;
    while(!T0IF);
}
```

C2 比较器失调校准:

```
void Calbriation_C2()    //C2比较器失调校准函数
{
    uchar i, aa, bb;
    C2CALEN = 1;    //使能比较器2的校准功能
    C2CALSET = 1;    //比较器2正负端串接1/2 VDD
    C2OUT = 0;
    C2EN = 1;    //使能C2比较器
    Delay_30us();
    C2CAL = 0B01111111;
    Delay_30us();
    if(C2OUT)
    {
        C2DT7 = 1;
    }
    else
    {
        C2DT7 = 0;
    }
    aa = 0B10111111;
    bb = 0B01000000;
    for(i = 0; i <= 6; i++)    //C2CAL第6~0位校准
    {
        C2CAL &= aa;
        Delay_30us();
        if(C2OUT)
        {
            C2CAL |= bb;
        }
        else
        {
            C2CAL &= aa;
        }
        aa >> 1;
        bb >> 1;
    }
}
```

CMP1 初始化:

```
void CMP1_Init()
{
    TR00=1;//比较器1的正向输入端—P00配置为输入模式
    TR01=1;//比较器1的负向输入端—P01配置为输入模式
    TR02=0;//配置比较器1的输出引脚,
    P0LR2=0;
    C1CTL =0x81;//CMP1输出使能, 负端输入选择P01, 不校准, 使能CMP1
}
}
```

CMP2 初始化:

```
void CMP2_Init()
{
    TR10=1;//比较器2的正端输入—P10配置为输入模式
    TR21 = 0;//配置比较器2的输出引脚
    P2LR1=0;
    C2CTL=0x93;//电阻分压器选择0.5VDD, 负端输入选择内部VDAC, 使能比较器2
    C2FILTCTL =0x67;//正极性输出, 范围控制使能, 滤波使能, 滤波采样数量为7
    C2FILTPRE =0x10;//滤波采样时钟16分频
    COUT =0x20;//分压电路的电压源为VDD, 输出使能
    C2IE =1;
    PUIE=1;
    AIE=1;
}
}
```

中断函数:

```
void int_fun0() __interrupt (0)
{
    // 比较器中断
    if(C2IF)
    {
        LED2 = !C2OUT;
        C2IF=0;
    }
}
}
```

主函数:

```
void main()
{
    Init_fun();//初始化晶振和公用IO口
    Calbriation_C2();
    CMP1_Init();//比较器1的初始化
    CMP2_Init();//比较器2的初始化

    while(1)
    {
        if(C1OUT)
        {
            LED3 = 0;
        }
        else
        {
            LED3 = 1;
        }
    }
}
}
```


注意事项:

1、使能比较器中断，用户在进入中断后必须先读取相应比较器的输出位，然后再清除中断标志位。

2、比较器滤波采样时钟频率计算方式

★ 式13.1: 比较器1滤波采样时钟频率 = $\frac{\text{SCLK}}{\text{FP1}<7:0>+1}$

3、当使能比较器范围控制功能时，利用系统时钟作为筛选时钟，当时钟电平为高时，比较器结果输出；当时钟电平为低时，比较器结果保持。

4、比较输出需要滤波的原因：内部和外部的寄生效应，和信号线、电源线与其他系统部分间的信号耦合造成了比较器的输出震荡，因此需要进行滤波处理。