

KF8F3132——PWM5 样例程序

引言

本应用笔记提供了 KF8F3132—PWM5 单输出、半桥输出、全桥输出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。

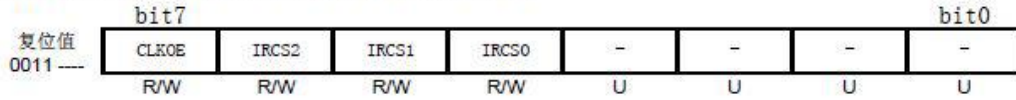
本应用笔记须与 KF8F3132 数据手册结合使用。

寄存器

寄存器使用说明

OSCCTL: 系统控制寄存器

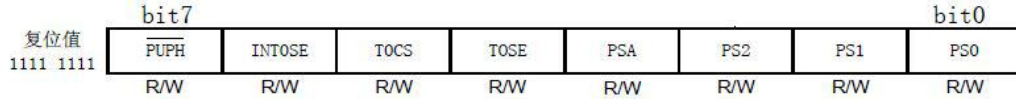
寄存器OSCCTL: 系统频率控制寄存器(地址:2FH)



图注: R = 可读 W = 可写 P = 可编程 U = 未使用
- = 读为0 x = 状态未知

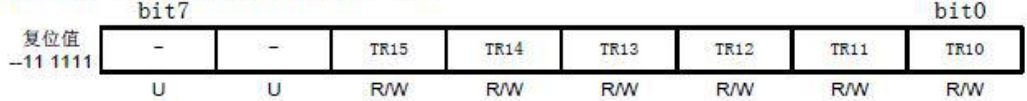
OPTR: 选择寄存器

寄存器OPTR: 选择寄存器(地址: 21H)



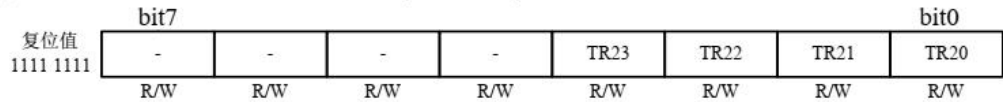
TR1: P1 口方向控制寄存器

TR1: P1口方向控制寄存器(地址: 27H)



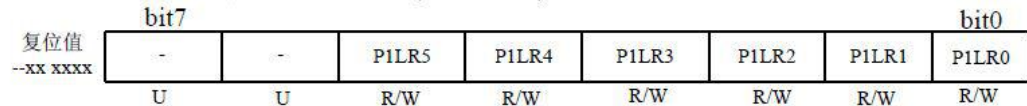
TR2: P2 口方向控制寄存器

寄存器2.13: TR2: P2口方向控制寄存器(地址: 26H)



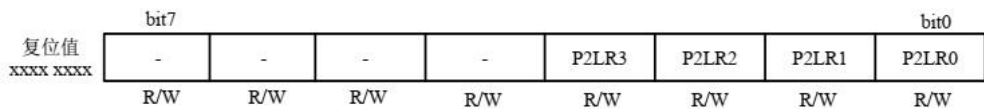
P1LR: P1 口输出锁存寄存器

寄存器P1LR: P1口输出锁存寄存器(地址: 47H)



P2LR: P2 口输出锁存寄存器

寄存器2.12: P2LR: P2口输出锁存寄存器(地址: 46H)



PSTRCTL: PWM5 脉冲转向控制寄存器 0

寄存器9.8: PSTRCTL: PWM5脉冲转向控制寄存器0(地址: 5DH)

复位值 ---0 0001	bit7							bit0
	-	-	-	STRSYNC	STREND	STRENC	STRENB	STRENA
	U	U	U	R/W	R/W	R/W	R/W	R/W

PP5H: PWM5 周期寄存器高 8 位

PP5L: PWM5 周期寄存器低 8 位

PWM5H0: PWM5 占空比寄存器高 8 位

PWM5L0: PWM5 占空比寄存器低 8 位

T2H: 定时器 2 高字节寄存器

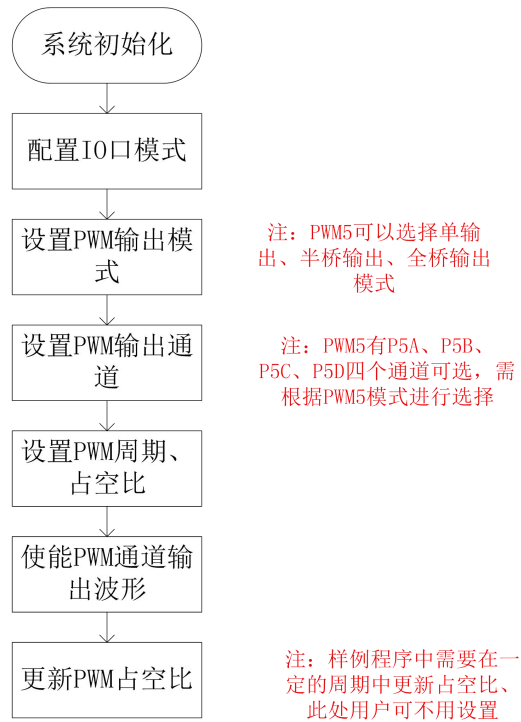
T2L: 定时器 2 低字节定时器

T2CTL0: T2 控制寄存器 0

T2CTL1: T2 控制寄存器 1

T0: 定时/计数器寄存器 0

PWM5 样例程序框图



注：PWM3 有三种输出方式，分别为单输出、半桥输出、全桥输出，以上三种模式均可以采用样例程序框图的流程来达到测试及配置的目的。

PWM5 单输出样例简述:

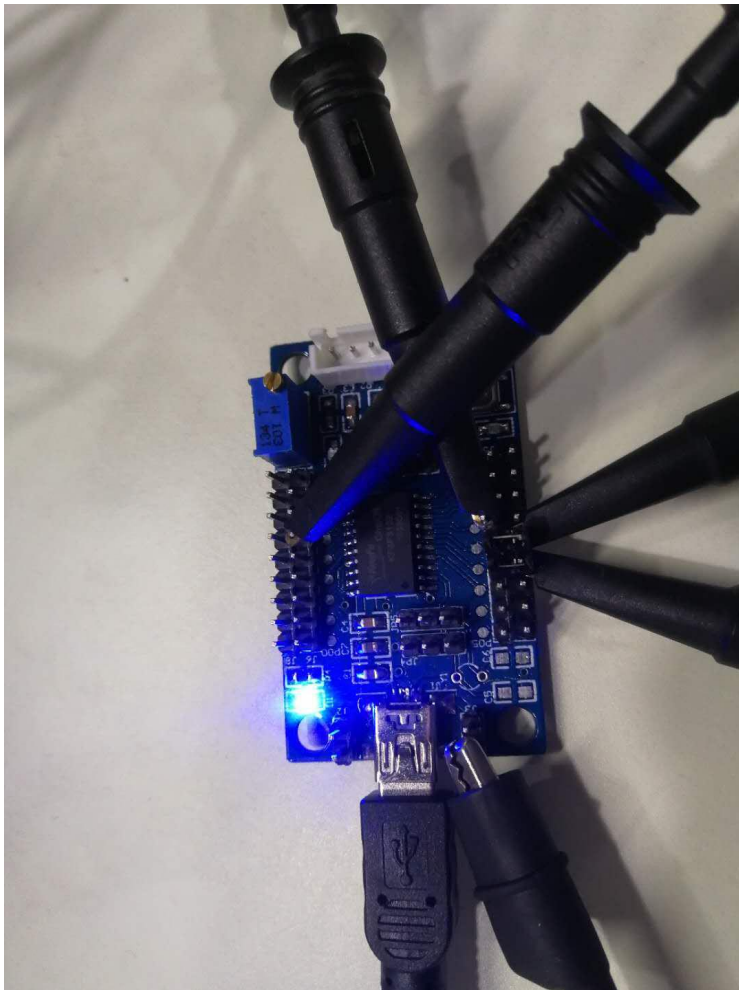
开发环境: ChipON IDE

功能简述: PWM5 周期设置为 100us, 占空比是每 10ms 增加 1%, 满占空比后清零继续增加。占空比 1s 一个周期, 从 0%到 100%循环扫描。

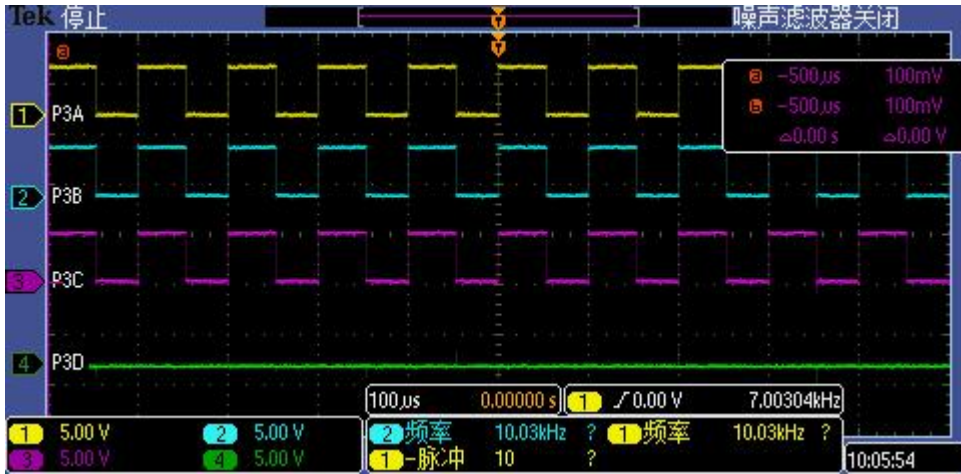
P3A、P3B、P3C、P3D 以 2s 为周期循环累加的方式打开。例如: 先打开 P3A 的 PWM 输出, 1s 后再把 P3B 打开, 依次循环直至全部打开输出保持 1s 后关闭全部输出, 等待 1s 后再依次打开各个 PWM 输出通道。

硬件连接: P3A、P3B、P3C、P3D 分别接示波器的一个通道, 观察 PWM5 输出。

下图为 PWM 波形输出示波器连接通道:



下图为 PWM5 单输出波形：

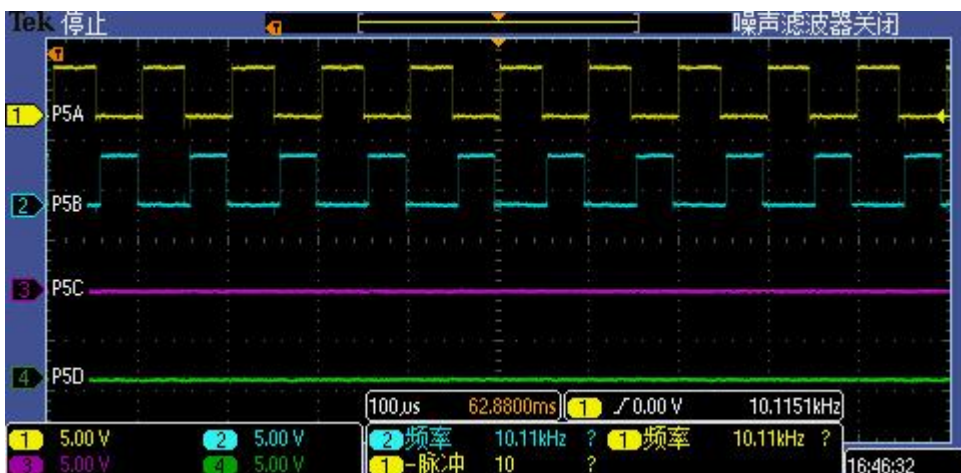


PWM5 半桥输出样例简述：

功能简述：PWM5 周期设置为 100µs，占空比是每 10ms 增加 1%，满占空比后清零继续增加。占空比 1s 一个周期，从 0%到 100%循环扫描。PWM5 设置为半桥输出模式，P3A 和 P3B 输出互补的 PWM 波形。死区延时开启，死区延时设置为 3µs。

硬件连接：示波器 P3A、P3B 管脚，观察 PWM3 输出。

下图为 PWM3 半桥输出波形：



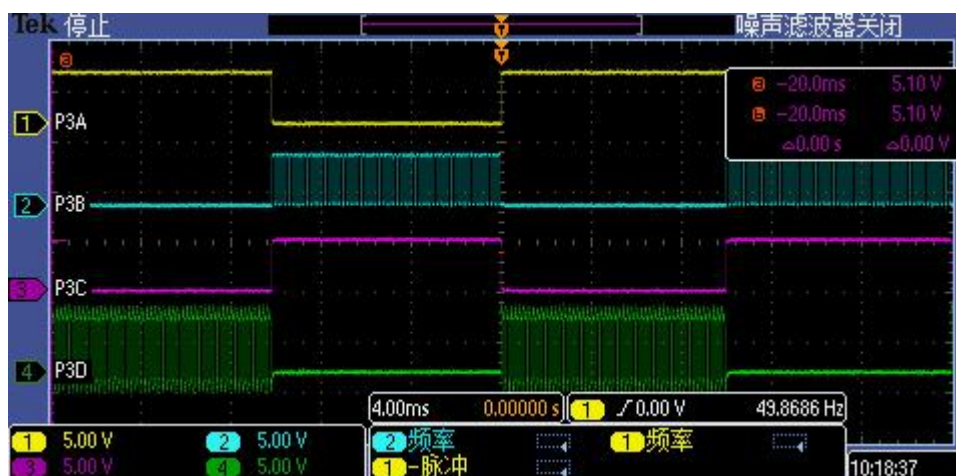
PWM5 全桥输出样例简述：

功能简述：PWM5 周期设置为 200µs，占空比是每 20ms 增加 1%，满占

空比后清零继续增加。占空比 2s 一个周期，从 0%到 100%循环扫描。每隔 10ms，PWM5 换向一次。PWM5 使能自动关断模式和自动重启模式，关断源是 INT0 管脚低电平。INT0 管脚高电平 PWM5 正常工作，INT0 低电平 PWM5 关闭。

硬件连接：示波器接 P3A、P3B、P3C、P3D 管脚，观察 PWM5 输出。INT0 接 GND，PWM5 关闭，INT0 脚取消接地，PWM5 恢复输出。

下图为 PWM5 全桥输出波形：



PWM5 单输出样例参考程序:

初始化时钟及 IO 口;

```
void Mcu_Init()  
{  
    OSCCTL =0x70; //系统时钟设为16M  
    OPTR=0X04; //T0设为定时模式, 32分频  
  
    TR1=0; //P1口全部配置为输出口  
    P1LR=0; //P1口初始为输出低电平  
  
    TR2=0;  
    P2LR=0;  
}
```

PWM5 输出模式及通道设置函数:

```

void PWM5_Init()
{
    PWM5CTL0 =0x0C;//单输出模式，P5A/C、P5B/D均为高电平有效
    PWM5OC =0;//P5A/B均为PWM输出模式
    PWM5PC =0;//P5A/B均为高电平有效
    PWM5FC =0;//强制输出，改为默认值
    PWM5CTL1 =0;//关闭自动重启功能，死区延时设置清零
    P5ASCTL =0;//关断状态位清零，禁止自动关闭，P5A/B/C/D关闭态默认是0
    PSTRCTL =0x1F;//命令周期后立即输出转向更新，P5A输出PWM波形
    PWM5CTL2 =0x01;//关闭熔丝位设置，设置更新使能

    PP5H=0;
    PP5L=99;//周期是(99+1)*0.5uS=100us，10K主频。
    PWM5H0=0;
    PWM5L0=50;

    T2H=0;
    T2L=0;
    T2CTL0 =0x07; //T2时钟源是晶振频率，预分频比1:16，后分频的值是1，T2打开
    T2CTL1 =0;//向上计数，禁止触发AD
}

```

延时函数（使用 T0 定时器）:

```

void Delay_ms(unsigned int k)
{
    for (;k>0;k--)
    {
        T0=132;
        TOIF=0;
        while(!TOIF);
    }
}

```

主函数:

```

void main()
{
    unsigned char i=0;
    Mcu_Init();

    PWM5_Init();
    while(1)
    {
        Delay_ms(10);
        i++;
        if(i>100)
        {
            i=0;
            if(PSTRCTL<0x0F) //依次打开各个PWM3输出通道
            {
                PSTRCTL=PSTRCTL<<1;
                PSTRCTL++;
            }
            else //关闭全部PWM输出
            {
                PSTRCTL=0;
            }
        }
        PWM5L0 =i;//PWM3占空比更新
    }
}

```

PWM5 半桥输出样例参考程序:

MCU 初始化:

```
void Mcu_Init()
{
    OSCCTL =0x70;//系统时钟设为16M
    OPTR=0X04; //T0设为定时模式, 32分频

    TR1=0;//P1口全部配置为输出口
    P1LR=0;//P1口初始为输出低电平

    TR2=0;
    P2LR=0;
}
```

PWM 模块初始化:

```
void Mcu_Init()
{
    OSCCTL =0x70;//系统时钟设为16M
    OPTR=0X04; //T0设为定时模式, 32分频

    TR1=0;//P1口全部配置为输出口
    P1LR=0;//P1口初始为输出低电平

    TR2=0;
    P2LR=0;
}
```

延时函数:

```
void Delay_ms(unsigned int k)
{
    for (;k>0;k--)
    {
        T0=132;
        TOIF=0;
        while (!TOIF);
    }
}
```

主函数:

```
void main()
{
    unsigned char i=0;
    Mcu_Init();

    PWM5_Init();
    while(1)
    {
        Delay_ms(10);
        i++;
        if(i>100)
        {
            i=0;
        }
        PWM5L0 =i;//PWM5占空比更新
    }
}
```

PWM5 全桥输出样例参考程序:

MCU 初始化 (时钟、IO 口):

```
void Mcu_Init()
{
    OSCCTL =0x70;//系统时钟设为16M
    OPTR=0x04; //T0设为定时模式, 32分频

    TR02 =1;

    TR1=0;//P1口全部配置为输出口
    P1LR=0;//P1口初始为输出低电平

    TR2=0;
    P2LR=0;
}
```

PWM 模块初始化:

```
void PWM5_Init()
{
    PWM5CTL0 =0x4C;//全桥正向输出模式, P5A/C、P5B/D均为高电平有效
    PWM5OC =0;//P5A/B均为PWM输出模式
    PWM5PC =0;//P5A/B均为高电平有效
    PWM5FC =0;//强制输出, 改为默认值
    PWM5CTL1 =0x80;//使能自动重启功能
    P5ASCTL =0x40;//关断状态位清零, 使能INT0低电平自动关闭, P5A/B/C/D关闭态默认是0
    PSTRCTL =0x1F;//下个PWM周期后输出转向更新, P5A/B/C/D输出PWM波形
    PWM5CTL2 =0x01;//关闭熔丝位设置, 设置更新使能

    PP5H=0;
    PP5L=99;//周期是(99+1)*0.5uS=100us, 10K主频。
    PWM5H0=0;
    PWM5L0=0;

    T2H=0;
    T2L=0;
    T2CTL0 =0x07; //T2时钟源是晶振频率, 预分频比1:16, 后分频的值是1, T2打开
    T2CTL1 =0;//向上计数, 禁止触发AD
}
```

延时函数 (T0 定时器):

```
void Delay_ms(unsigned int k)
{
    for(;k>0;k--)
    {
        T0=132;
        T0IF=0;
        while(!T0IF);
    }
}
```

主函数:

```
void main()
{
    unsigned char i=0;
    Mcu_Init();

    PWM5_Init();
    while(1)
    {
        PWM5CTL0 =0x4C;//全桥正向模式
        Delay_ms(10);//保持10ms
        i++;
        if(i>100)//i溢出后归零
        {
            i=0;
        }
        PWM5L0 =i;//占空比更新
        PWM5CTL0 =0xCC;//全桥反向模式
        Delay_ms(10);
    }
}
```

PWM5 模块应用注意事项:

1、PWM5 边沿对齐模式周期计算公式:

$$\star \text{式9.1: PWM周期}=(\langle \text{PP5H:PP5L} \rangle + 1) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比})$$

2、PWM5 中心对齐模式周期计算公式:

$$\star \text{式9.2: PWM周期}=(\langle \text{PP5H:PP5L} \rangle) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比}) \cdot 2$$

注: T_{INTHF} 内部高频振荡器时钟周期

3、PWM5 边沿对齐模式脉冲宽度、占空比计算公式:

$$\star \text{式9.3: 脉冲宽度}=(\text{PWM5H0:PWM5L0}) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比})$$

$$\star \text{式9.4: 占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM5H0:PWM5L0}}{\text{PP5} + 1}$$

4、PWM5 中心对齐模式脉冲宽度、占空比计算公式:

$$\star \text{式9.5: 脉冲宽度}=(\text{PWM5H0:PWM5L0}) \cdot T_{\text{INTHF}} \cdot (\text{T2预分频比}) \cdot 2$$

$$\star \text{式9.6: 占空比} = \frac{\text{脉冲宽度}}{\text{PWM周期}} = \frac{\text{PWM5H0:PWM5L0}}{\text{PP5}}$$

由上述公式可知, 占空比为 0%和 100%满足条件如下:

1、边沿对齐模式:

0%: 占空比寄存器为 0

100%: 占空比寄存器为大于等于 (PP5+1) (其中 PP5 为周期寄存器)

2、中心对齐模式:

0%: 占空比寄存器为 0

100%: 占空比寄存器为大于等于 PP5 (其中 PP5 为周期寄存器)

在半桥模式下, 占空比为 0%或者 100%时, 只有在边界 PWM 变

化时插入死区，否则不插入死区。

5、当 T2 启动时占空比寄存器立即更新。占空比寄存器的值更新到占空比缓冲器中时，才真正完成了占空比寄存器的设置。

6、PWM5 分辨率计算公式：

$$\star \text{式9.7: 分辨率} = \frac{\text{Log}[4(\langle \text{PP5H:PP5L} \rangle + 1)]}{\text{log}2} \quad \text{位}$$

7、死区延时时间计算公式：

$$\star \text{式9.8: 延时时间} = \text{PDC}\langle 6:0 \rangle \cdot T_{\text{INTHF}}$$

8、通过选择 T2 的计数方式，从而产生边沿对齐和中心对齐的 PWM 信号。当 T2 向上计数时，产生边沿对齐的 PWM 信号；当 T2 向上向下计数时，产生中心对齐的 PWM 信号。

9、半桥输出模式具有死区延时功能，由于外部电路中的开关管等原件导通和截止时间存在差异，可用来防止在半桥驱动电路中产生直通电流，损坏相关电路。

10、若切换前电平和切换后输出电平无变化时不插入死区，若切换前输出电平和切换后输出电平有变化时，则插入死区。

11、影响关断源触发的信号是电平信号而不是边沿信号，只要关断电平存在关断状态就会保持。