PWM 使用手册

南京博芯电子技术有限公司

2009-04

版权说明

否认书：

南京博芯电子技术有限公司保留对文档随时进行修改的权利，无须任何申明。南京博芯电子技术有限公司所提供的信息是精确可靠的。对于它的应用以及由于应用而导致违反专利权或是第三方的其他权利，本公司不负任何责任。

版本历史

日期	版本	描述	备注
2009-04	1.0	初稿	Jack

目　录

一. PWM在SEP4020 中的位置

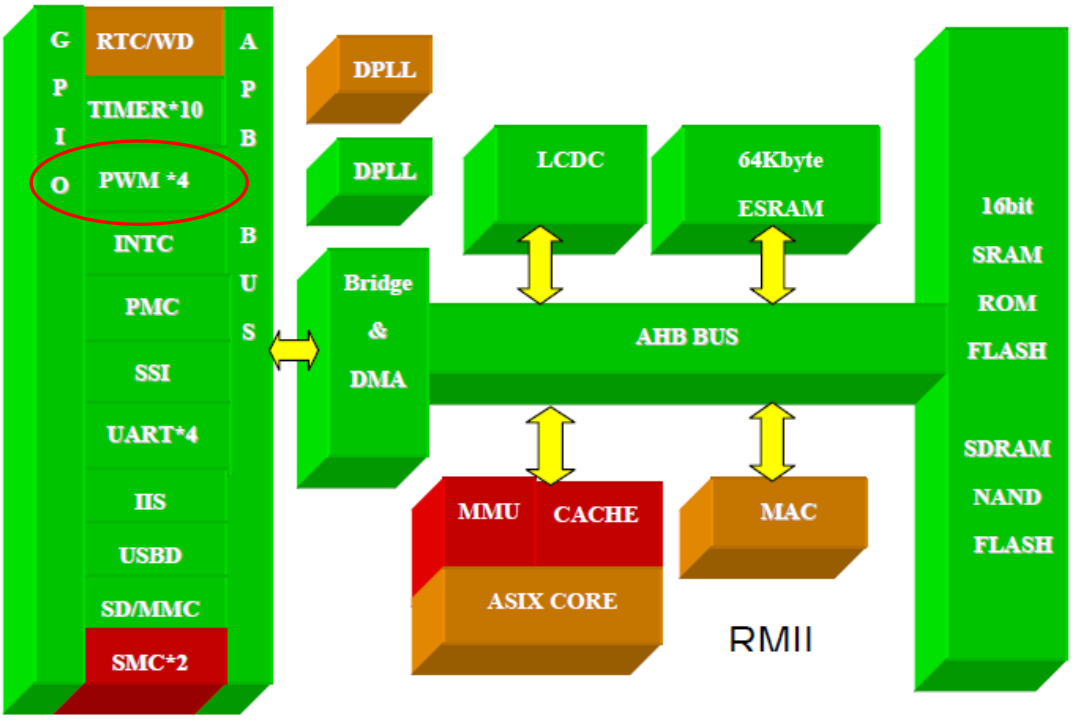


图 1　　PWM 在 SEP4020 中的位置

二. PWM介绍

2.1 功能介绍

PWM（Pulse Width Modulator，脉宽调制）共有4 个通道，每个通道相互独立。当通道内部的16 位计数器计到设定的值后输出高低电平，从而产生相应的波形，计数时钟从总线时钟分频而来，分频值可配为 1/2/4/6/8/10/……/4094。

PWM 可工作在3 种模式：

- PWM 模式：周期可配，占空比根据输入数据动态变化
- 高速 GPIO 模式：数据直接移位输出或者采样输入
- TIMER 模式：用作计时器

2.1.1 使用

2.1.1.1PWM 模式

该模式下 PWM 输出可变周期的波形，但是每个周期的高电平占空比可根据输入的数据动态变化。

波形周期由周期寄存器决定，大小可配置为 2～65535，单位为计数时钟周期。

占空比的大小由向数据 FIFO 写入的数据决定，一个数据决定一个周期的高电平占空比，数据的大小最小为1，最大为65535（FIFO 数据大于或等于计数周期时，计数周期输出全高电平），单位为计数时钟周期。

每个周期开始的波形输出为高电平，当计数器的值和当前占空比数据相等的时候，波形输出变为低电平，数据FIFO 弹出当前占空比数据，计数器继续计数，当计数器的值和周期寄存器的值相等时，波形输出为高电平，计数器清零，开始下一个波形周期。当FIFO 中没有数据时，PWM 会按照最后一个占空比数据重复输出相同占空比的波形。如果要输出固定周期和占空比的波形，只需配一次周期寄存器和一次数据FIFO。

当数据 FIFO 半满、空时会发出中断。

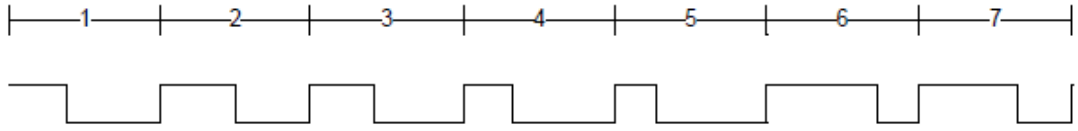


图2 动态变化占空比的输出波形

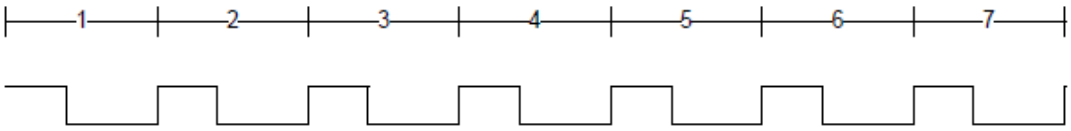


图3 固定占空比和周期的输出波形

2.1.1.2高速GPIO 模式

该模式下 PWM 相当于高速的GPIO，可以用作输入或者输出。

用作 GPIO 输出时，PWM 将数据FIFO 中的数据直接从高位至低位移位输出，波形随着移位寄存器输出的数据变化，移位时钟就是计数时钟，当输入数据全部移位输出后（FIFO 空，移位寄存器也为空），输出波形电平保持在最后一个移位的数据。当数据FIFO 半空和空的时候会发出中断。高速GPIO 可以一次连续输出64 个数据。

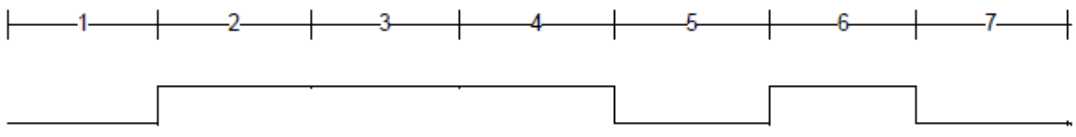


图4 数据“0111010”的输出波形

用作 GPIO 输入时，PWM 将用计数时钟去采样外部数据，采样后的数据按高到低的顺序移位后被存入数据FIFO，等待CPU 读取。当数据FIFO 半满和满的时候会发出中断。

2.1.1.3TIMER 模式

该模式下，PWM 就是一个16 位的TIMER，计数器可以工作在内部计数模式。

内部计数时，当计数器的值等于周期寄存器时，发出中断，可以输出高电平、低电平、翻转电平或者保持输出不变，当前计数结束。

一次计数：计数器清零，停止计数，等待下一次启动，同时发出中断。

循环计数：计数器清零，重新开始计数，同时发出中断

2.2 寄存器介绍

脉宽调制模块的基址：0x10004000

名称	偏移地址	复位值	描述
PWM0_CTRL	0x00	0x00000000	PWM0 控制寄存器
PWM0_DIV	0x04	0x00000000	PWM0 分频寄存器
PWM0_PERIOD	0x08	0x0000ffff	PWM0 周期寄存器
PWM0_DATA	0x0c	--	PWM0 数据寄存器
PWM0_CNT	0x10	0x00000000	PWM0 计数寄存器
PWM0_STATUS	0x14	0x00000003	PWM0 状态寄存器
PWM1_CTRL	0x20	0x00000000	PWM1 控制寄存器
PWM1_DIV	0x24	0x00000000	PWM1 分频寄存器
PWM1_PERIOD	0x28	0x0000ffff	PWM1 周期寄存器
PWM1_DATA	0x2c	--	PWM1 数据寄存器
PWM1_CNT	0x30	0x00000000	PWM1 计数寄存器
PWM1_STATUS	0x34	0x00000003	PWM1 状态寄存器
PWM2_CTRL	0x40	0x00000000	PWM2 控制寄存器
PWM2_DIV	0x44	0x00000000	PWM2 分频寄存器
PWM2_PERIOD	0x48	0x0000ffff	PWM2 周期寄存器
PWM2_DATA	0x4c	--	PWM2 数据寄存器
PWM2_CNT	0x50	0x00000000	PWM2 计数寄存器
PWM2_STATUS	0x54	0x00000003	PWM2 状态寄存器
PWM3_CTRL	0x60	0x00000000	PWM3 控制寄存器
PWM3_DIV	0x64	0x00000000	PWM3 分频寄存器
PWM3_PERIOD	0x68	0x0000ffff	PWM3 周期寄存器
PWM3_DATA	0x6c	--	PWM3 数据寄存器
PWM3_CNT	0x70	0x00000000	PWM3 计数寄存器
PWM3_STATUS	0x74	0x00000003	PWM3 状态寄存器
PWM_INTMASK	0x80	0x0000000f	PWM 中断屏蔽寄存器
PWM_INT	0x84	0x00000000	PWM 中断寄存器
PWM_ENABLE	0x88	0x00000000	PWM 使能寄存器

三. 实现原理

3.1 硬件原理

3.1.1 SEP4020 PWM部分输出管脚定义

序号	管脚名	方向	描述	驱动电流（mA）	属性	复位值
27	PWM0	I/O	NAND FLASH ready/busy	4		1'H0
26	PWM1	I/O	NAND FLASH命令锁存	4		1'H0
24	PWM2	I/O	NAND FLASH 地址锁存	4		1'H0
23	PWM3	I/O	NAND FLASH 片选	4		1,H1

3.2 软件原理

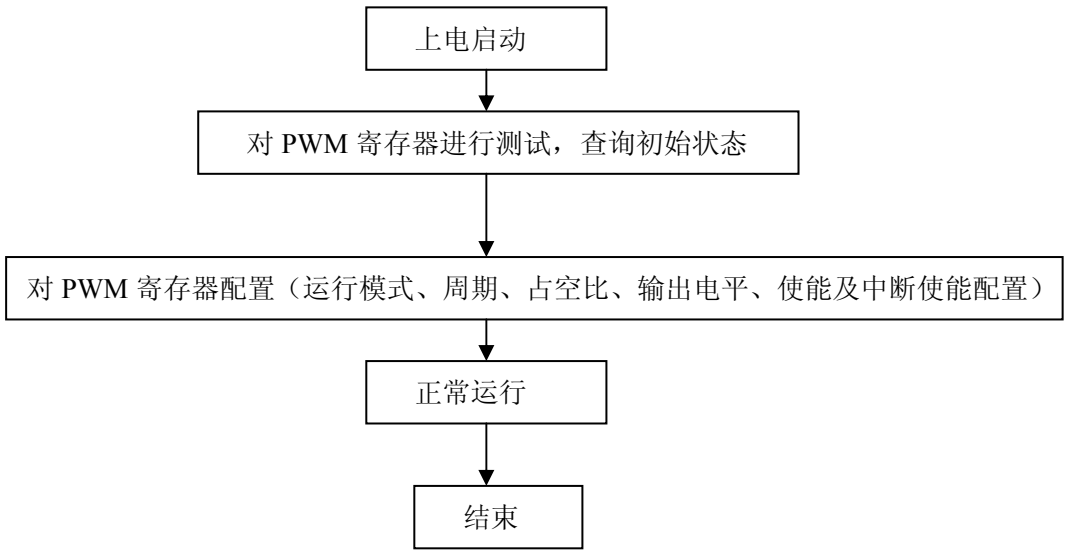
3.2.1 头文件定义说明

```
#include    <stdio.h>          标准输入输出函数库
#include    "intc.h"           INTC 模块的中断源，中断处理，定义中断的向量结构体
#include    "sep4020.h"        sep402寄存器定义文件
#include    "pwm.h"            对PWM的操作函数
```

3.2.2 核心数据结构声明

INT_VECTOR　中断向量结构体包括：中断号和中断处理函数

3.2.3 代码实现流程图



3.2.4 主要函数及参数，返回值介绍

（1）void PWMInit(U32 PwmNum,U32 MatchOut,U32 GpioDirection,U32 Function)

描述：对PWM寄存器的配置

输入：

U32 PwmNum PWM通道号

U32 MatchOut TIMER模式下，当计数值等于周期寄存器时输出状态

00：输出低电平

01：输出高电平

10：保持输出不变

11：翻转输出电平

U32 GpioDirection GPIO输入/输出选择

0：输出

1：输入

U32 Function 工作模式

00：PWM模式

01：高速GPIO模式

10：TIMER模式，一次计数

11：TIMER模式，循环计数

输出：无

使用位置：

```
main()
  {
……..
  //初始化
  mask_irq(INTSRC_PWM);
  PWMIntDisable(INTSRC_PWM);
  PWMInit(0,0x00,0x0,0x00);        /*PWM0，输出低电平（未用），GPIO输
                                     出，PWM方式*/
  PWMSet(0,0xFF,20,50);          /*PWM0，0xFF分频，周期4，占空比90
                                     %*/
  PWMEnable(0);                    /*使能PWM0*/
  unmask_irq(INTSRC_PWM);
```

```
            irq_enable(INTSRC_PWM);

            PWMIntEnable(0);                /*PWM0中断使能*/
          ……..

            }
```

（2）void PWMEnable(U32 PwmNum)

描述：通道使能函数

输入： PwmNun

0 PWM0通道使能

1 PWM1通道使能

2 PWM2通道使能

3 PWM3通道使能

输出：无

使用位置：

```
main()
            {
          ……..
            //初始化
            mask_irq(INTSRC_PWM);
            PWMIntDisable(INTSRC_PWM);
            PWMInit(0,0x00,0x0,0x00);       /*PWM0，输出低电平（未用），GPIO输
                                              出，PWM方式*/
            PWMSet(0,0x5FFF,20,50);         /*PWM0，0xFF分频，周期4，占空比90
                                              %*/
            PWMEnable(0);                   /*使能PWM0*/
            unmask_irq(INTSRC_PWM);

            irq_enable(INTSRC_PWM);

            PWMIntEnable(0);                /*PWM0中断使能*/
          ……..

            }
```

（3）void PWMDisable(U32 PwmNum)

描述：不使能对应通道的配置函数

输入：

PwmNun	
0	PWM0通道使能
1	PWM1通道使能
2	PWM2通道使能
3	PWM3通道使能

输出：无

使用位置：

```
main()
        {
      ……..
        //初始化
        mask_irq(INTSRC_PWM);
        PWMIntDisable(INTSRC_PWM);
        PWMInit(0,0x00,0x0,0x00);          /*PWM0，输出低电平（未用），GPIO输
                                             出，PWM方式*/
        PWMSet(0,0x5FFF,20,50);            /*PWM0，0xFF分频，周期4，占空比90
                                             %*/
        PWMEnable(0);                      /*使能PWM0*/
        unmask_irq(INTSRC_PWM);

        irq_enable(INTSRC_PWM);

        PWMIntEnable(0);                   /*PWM0中断使能*/
      ……..

        }
```

（4）void PWMSet(U32 PwmNum,U32 PwmDiv,U32 PwmPeriod,U32 PwmDuty)

描述：配置PWM模块，输出占空比可调的方波

输入：U32 PwmNum　　PWM通道号

U32 PwmDiv　　PWM分频因子

U32 PwmPeriod　　PWM周期

U32 PwmDuty　　PWM占空比

输出：无

使用位置：

```
main()
        {
      ……..
        //初始化
        mask_irq(INTSRC_PWM);
        PWMIntDisable(INTSRC_PWM);
        PWMInit(0,0x00,0x0,0x00);          /*PWM0，输出低电平（未用），GPIO输
                                             出，PWM方式*/
        PWMSet(0,0x5FFF,20,50);            /*PWM0，0xFF分频，周期4，占空比90
                                             %*/
        PWMEnable(0);                      /*使能PWM0*/
        unmask_irq(INTSRC_PWM);

        irq_enable(INTSRC_PWM);

        PWMIntEnable(0);                   /*PWM0中断使能*/
      ……..

        }
```

（5）void PWMTimerSet(U32 PwmNum,U32 TimerDiv,U32 TimerPeriod)

描述：配置PWM Timer模式函数

输入：U32 PwmNum　　PWM通道号

U32 PwmDiv　　PWM分频因子

U32 PwmPeriod　　PWM周期

输出：　无

使用位置：

```
main()
        {
      ……..
        //初始化
```

```
mask_irq(INTSRC_PWM);
PWMIntDisable(INTSRC_PWM);
PWMInit(0,0x00,0x0,0x00);          /*PWM0，输出低电平（未用），GPIO输
                                     出，PWM方式*/
PWMTimerSet(0,0xFF,4);             /*PWM0，0xFF分频，周期4*/
PWMEnable(0);                      /*使能PWM0*/
unmask_irq(INTSRC_PWM);

irq_enable(INTSRC_PWM);

PWMIntEnable(0);                   /*PWM0中断使能*/
……..

 }
```

（6）void PWMIntEnable(U32 PwmNum)

描述：配置使能通道中断使能

输入：PwmNun

0　PWM0通道使能

1　PWM1通道使能

2　PWM2通道使能

3　PWM3通道使能

输出：无

使用位置：

```
main()
 {
……..
//初始化
mask_irq(INTSRC_PWM);
PWMIntDisable(INTSRC_PWM);
PWMInit(0,0x00,0x0,0x00);          /*PWM0，输出低电平（未用），GPIO输
                                     出，PWM方式*/
PWMSet(0,0x5FFF,20,50);            /*PWM0，0xFF分频，周期4，占空比90
                                     %*/
PWMEnable(0);                      /*使能PWM0*/
```

```
            unmask_irq(INTSRC_PWM);

            irq_enable(INTSRC_PWM);

            PWMIntEnable(0);                    /*PWM0中断使能*/
          ……..

            }
```

（7）void PWMIntDisable(U32 PwmNum)

描述：配置使能通道中断不是能

输入：PwmNun

0　PWM0通道使能

1　PWM1通道使能

2　PWM2通道使能

3　PWM3通道使能

输出：无

使用位置：

```
main()
            {
          ……..
            //初始化
            mask_irq(INTSRC_PWM);
            PWMIntDisable(INTSRC_PWM);
            PWMInit(0,0x00,0x0,0x00);          /*PWM0，输出低电平（未用），GPIO输
                                                 出，PWM方式*/
            PWMSet(0,0x5FFF,20,50);            /*PWM0，0xFF分频，周期4，占空比90
                                                 %*/
            PWMEnable(0);                      /*使能PWM0*/
            unmask_irq(INTSRC_PWM);

            irq_enable(INTSRC_PWM);

            PWMIntEnable(0);                   /*PWM0中断使能*/
          ……..
```

}

（8）void PWMIntHandler(void)

描述：中断服务函数

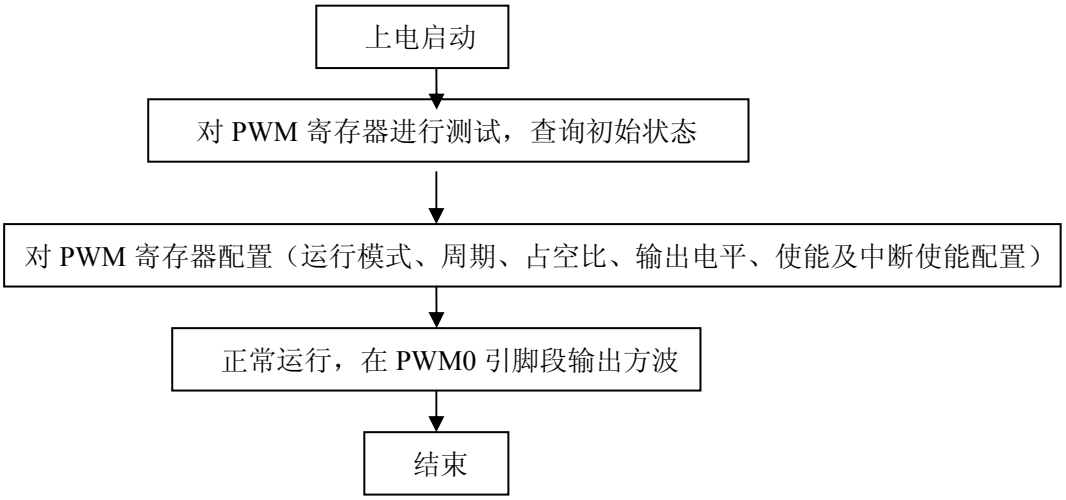
输入：无

输出：无

使用位置：将其指针放入中断对应的向量表里面

四.测试说明

4.1测试流程图



4.2测试结果

连接号蜂鸣器的短接帽，正确运行蜂鸣器会响，也可以用示波器测试PWM0的输出引脚，有方波输出。

中断开启的时候，在console窗口中会输出：

```
PWM3 interrupted,the STATUS is 63
PWM2 interrupted,the STATUS is 43
PWM1 interrupted,the STATUS is 23
PWM0 interrupted,the STATUS is 3
```