

SAM4N LED 点灯程序

SAM4N 学习笔记

版本号：1.00

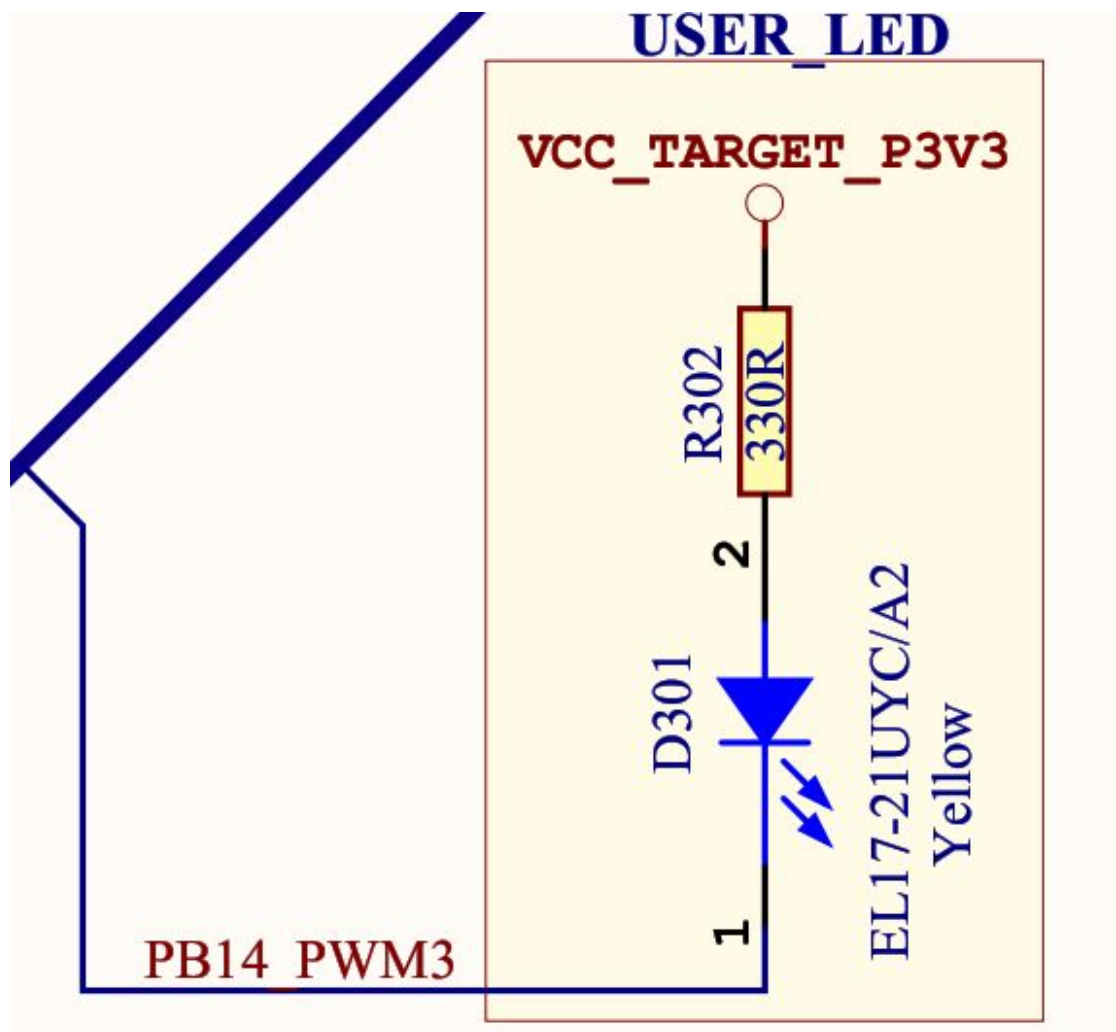
日期：2013/10/25

一、准备工作:

将上一节搭建的工程模板复制一份，命名为“1.blink”，这作为我们开发的第一个程序。

二、程序编写:

板上只有一个可控制的 LED，就是 LED0，在按键的旁边。打开原理图，可以看到 LED 的电路如下图：



通过原理图可以看出，LED 和 MCU 的 PB14 相连接，当 PB14 输出低电平的时候，LED 就会被点亮，为高电平的时候被熄灭。

下面开始编写 LED 驱动程序，首先在“Drivers”文件夹创建一个“drv_led.c”和“drv_led.h”文件，并加入到工程的“Drivers”分组下，在 drv_led.c 中编写 led 驱动程序，如下：

```
#include "drv_led.h"

void led_hw_init(void)
{
    /*禁止外设管理控制寄存器(PMC)写保护*/
```

```

PMC->PMC_WPMR = 0x504D4300;
/*使能 PIOB 时钟*/
PMC->PMC_PCER0 = (1UL << ID_PIOB);
/*使能外设管理控制寄存器(PMC)写保护*/
PMC->PMC_WPMR = 0x504D4301;
/*使能 LED 管脚*/
PIOB->PIO_PER|=(0x01<<LEDO_PIN);
/*使能 LED 管脚输出*/
PIOB->PIO_OER|=(0x01<<LEDO_PIN);
/*设置 LED 输出，即 LED 管脚为高电平*/
PIOB->PIO_SODR|=(0x01<<LEDO_PIN);
}

```

这里就写一个函数体，里面放置 led 的初始化代码，首先是要去开启 PIOB 的时钟，PMC 中的 PMC_PCER0 是控制各种外设时钟的，而且这个寄存器是被写保护的，在写之前，需要禁止写保护，这样写入数据才有效，给对应 PIOB 的 ID 位置 1 就算是开启了 PIOB 的时钟。讲到这里，也许有人注意到上一节搭建工程模板的时候这个没开启怎么就能成功点亮 LED 呢？是的，上次纯属侥幸。应该是芯片默认是开启这个 PIOB 时钟的，但对于我们编写驱动来说，这个步骤还是不要少的好，因为有可能再次之前这个别其他程序关掉了。

接下来是使能 PIOB14 管脚的 GPIO 功能，通过 PIOB 的 PIO_PER 寄存器去使能，对应的还有 PIO_PDR 是失能 GPIO 功能。使能了 PIOB14 的 GPIO 功能后，还要使能它的输出功能，接着就可以用过 PIO_SODR 和 PIO_CODR 去控制输出低电平还是高电平了。

在 drv_led.h 中主要是初始化函数的声明好 LED 控制相关的宏，如下：

```

#ifndef __DRV_LED_H
#define __DRV_LED_H
#include "sam4n.h"

#define LED0_PIN    14
#define led_hw_on() PIOB->PIO_CODR=(0x01UL<<LEDO_PIN)
#define led_hw_off() PIOB->PIO_SODR=(0x01UL<<LEDO_PIN)

void led_hw_init(void);

#endif

```

#endif

主要是防重复包含和 LED 控制的两个宏定义，接下来在 main.c 中程序如下：

```

int main(void)
{
    led_hw_init();
    while(1){
        led_hw_on();
        delay(300);
        led_hw_off();
        delay(300);
    }
}

```

```
}
```

主要是先初始化 **led**，然后就可以控制 **led** 的亮和灭了。效果我就不说了，就是简单的一亮一灭功能。主要是先熟悉一下 **SAM4N** 的 **GPIO** 是怎么驱动的。