# 基于Stm32

**闪烁灯代码**

while (1)

{

HAL\_GPIO\_TogglePin( led\_GPIO\_Port, led\_Pin);

HAL\_Delay(200);

}

## 使用pwm调制制作呼吸灯

Pwm：脉冲宽度调制

PWM 频率：是指1秒钟内信号从高电平到低电平再回到高电平的次数（一个周期）;也就是说一秒钟PWM有多少个周期单位

Fpwm = Tclk /(( arr +1)\*( psc +1))（单位： Hz )

arr 是计数器值

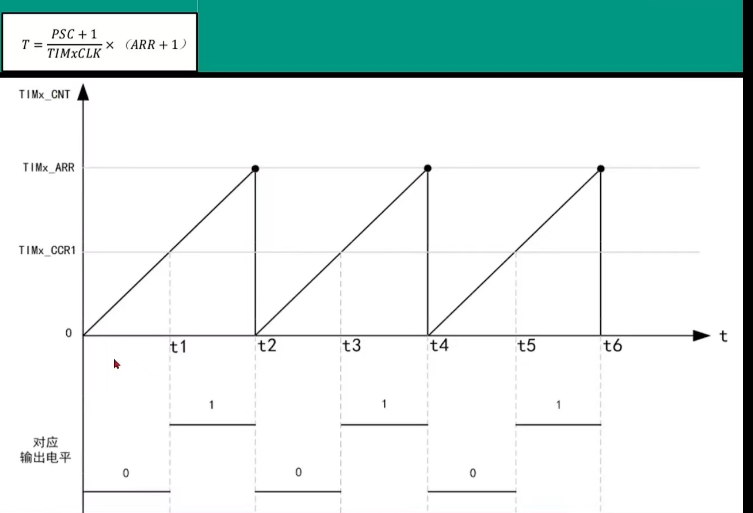
psc 是预分频值

占空比：是一个脉冲周期内，高电平的时间与整个周期时间的比例：

duty circle =TIM2->CCR2/arr 单位：%)

TIM2->CCR2用户设定值

**总的来说：PWM就是在合适的信号频率下，通过一个周期里改变占空比的方式来改变输出的有效电压。**PWM频率越大，相应越快，



举例：定时器频率 Tclk =72Mhz arr =499 psc =71

720000/500/72=2000Hz，即2KHz

arr =499 TIM2->CCR2=250

那么 PWM 频率就是

则 pwm 的占空比为50%

while (1)

{

while (pwmVal< 500)

{

pwmVal++;

\_\_HAL\_TIM\_SetCompare(&htim2, TIM\_CHANNEL\_2, pwmVal);

HAL\_Delay(1);

}

while (pwmVal)

{

pwmVal--;

\_\_HAL\_TIM\_SetCompare(&htim2, TIM\_CHANNEL\_2, pwmVal);

HAL\_Delay(1);

}

HAL\_Delay(200);

}

龙邱Tc264核心板

led0闪烁灯

**while** (1)

{

LED\_Ctrl(*LED1*,*RVS*);

delayms(100);

}

led1呼吸灯

**while** (1)

{

**for**(i=0;i<1000;i++)

{

LED\_Ctrl(*LED0*,0);

**for**(m=0;m<t;m++);

LED\_Ctrl(*LED0*,1);

**for**(n=0;n<1000-t;n++);

t++;

**if**(t>=1000)

{

**for**(i=0;i<1000;i++)

{

LED\_Ctrl(*LED0*,0);

**for**(m=0;m<t;m++);

LED\_Ctrl(*LED0*,1);

**for**(n=0;n<1000-t;n++);

t--;

}

}

}

}

}