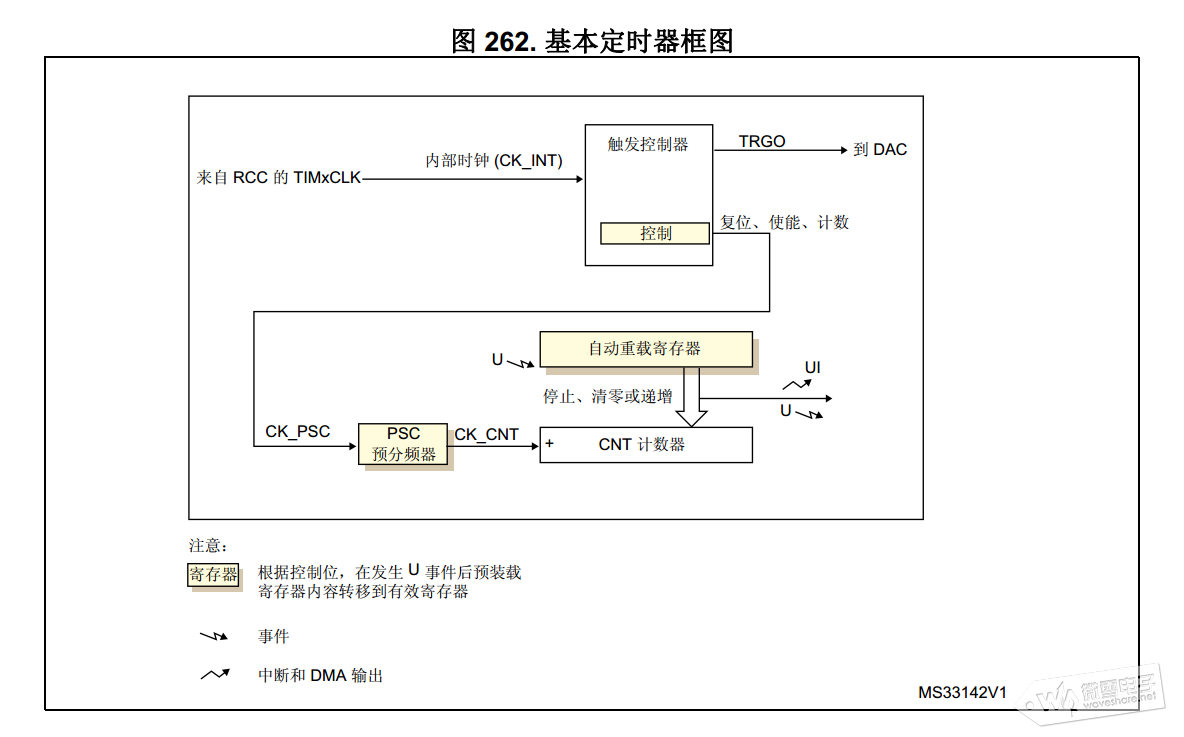
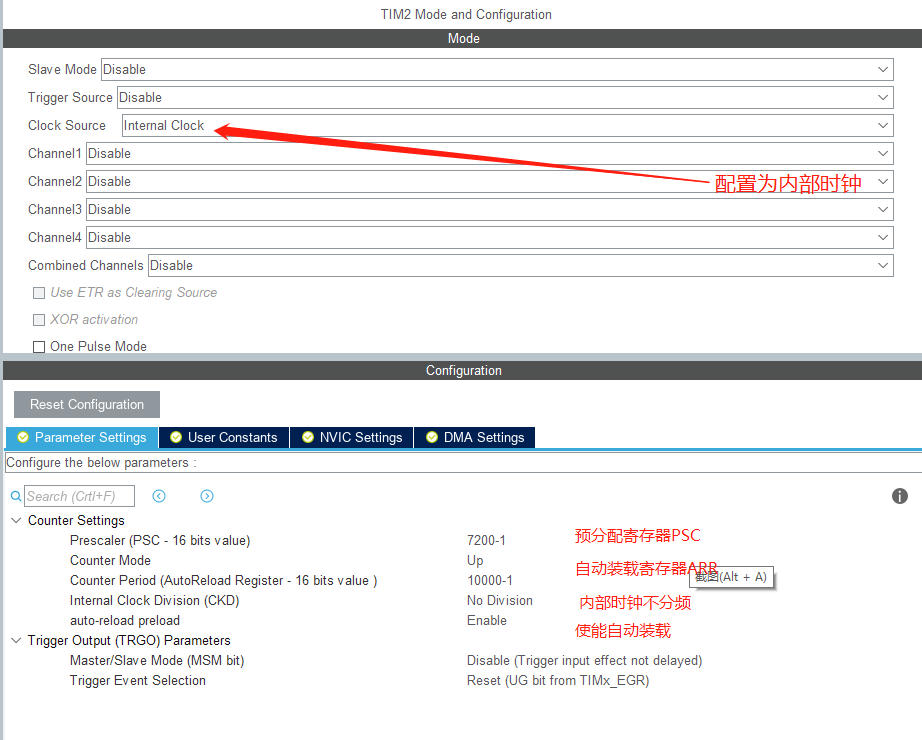
通用器定时中断配置

1. 定时器说明
   1. 定时器类型分为高级定时器、通用定时器和基本定时器，我们的开发板f103c8芯片有TIM1高级定时器和TIM2~4通用定时器；
   2. 定时器的功能
      1. 定时中断
      2. 输入捕获
      3. 输出比较
      4. PWM生成(边缘或中间对齐模式)
      5. 单脉冲模式输出
   3. 本章介绍如何配置通用定时器的定时中断，使用的是TIM2定时器；
   4. 定时器的结构如下：

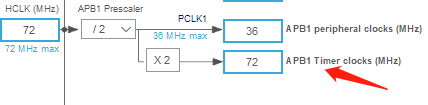


定时器时钟源设为内部时钟，首先查看参考手册看定时器挂载在哪个总线上，确定总线时钟频率，内部时钟可进行分频；然后过PSC预分频器分频，计数器按照分频后的频率递增或递减，当计数器寄存器 (TIMx\_CNT)加到自动重载寄存器 (TIMx\_ARR)或减为0时，会对计数器寄存器重置，同时产生中断信号，由此产生定时器定时中断，具体定时中断频率计算方式下面结合具体配置进行详细介绍。

1. 定时器中断配置
   1. 定时器配置

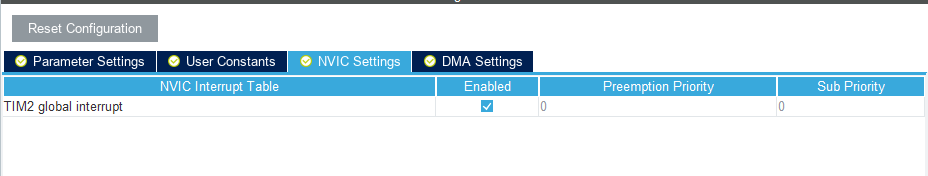


TIM2挂载在APB1总线上，时钟频率为72M，在Clock Configuration中可以看到：

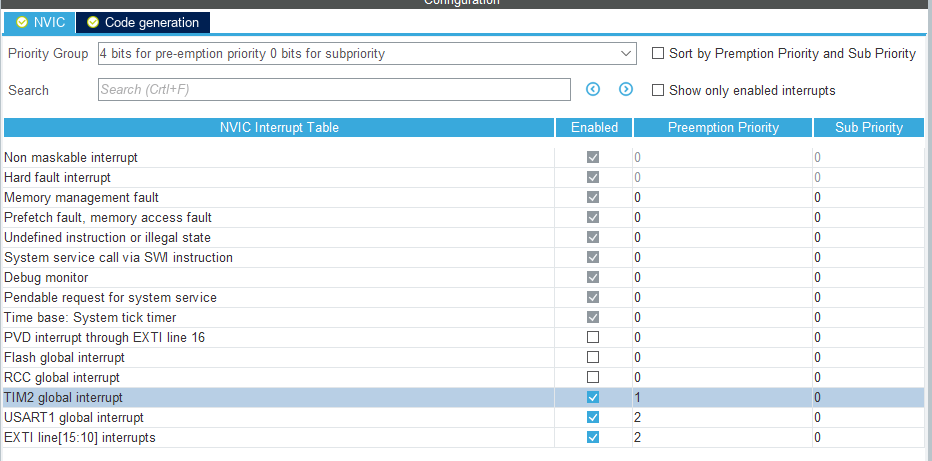


内部时钟不分频，则CK\_PSC的时钟频率等于APB1的时钟频率72MHz,即72000 000Hz。若要定时时间为1s,则即可设置7200分频（预分频器寄存器 (TIMx\_PSC)的值为7200-1），定时器的时钟CK\_CNT的频率为10000Hz.则自动重载寄存器 (TIMx\_ARR)设置为10000-1即定时为1s.TRGO为触发输出，可以触发内部ADC/DAC,这里我们没有用到这个功能，参数为默认设置。

* 1. 开启中断



* 1. NVIC配置中断优先级



1. 完成中断回调函数
   1. 生成代码后，在main函数中定时器初始化结束开中断

HAL\_TIM\_Base\_Start\_IT(&htim2);

* 1. timer.c中定义中断回调函数

void HAL\_TIM\_PeriodElapsedCallback(TIM\_HandleTypeDef \*htim){

if(htim ->Instance== TIM2){

……

}

}

作业：

1. 完成1s的定时中断，在中断中将LED翻转；
2. 串口中接收定时中断的频率或周期，更改定时中断的频率，通过LED的闪烁频率查看是否成功更改；（提示：通过更改ARR更改中断频率，可以通过重新初始化的方式实现更改，也可以通过直接更改ARR寄存器实现，后者更快更简洁）