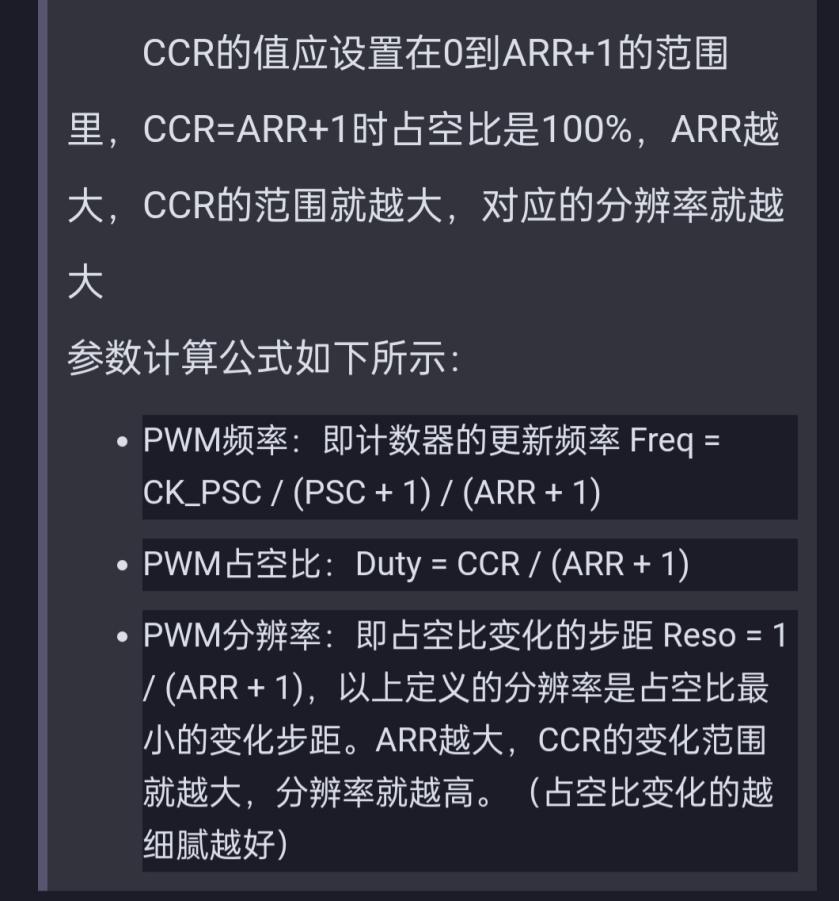
**请打开“导航窗格”以更清晰地查看文件结构**

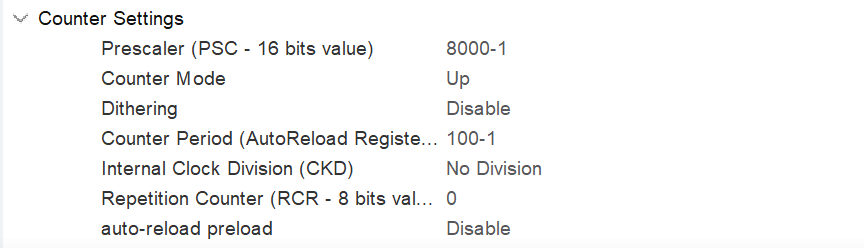
**0501PWM**

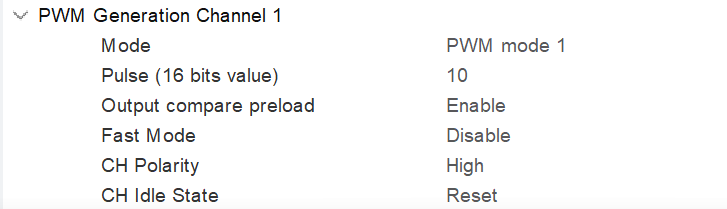
**1.配置**

**PWM**



* PA6（TIM16）、PA7（TIM17）为空引脚，可用于输出信号；配置为PWM Generation CH1

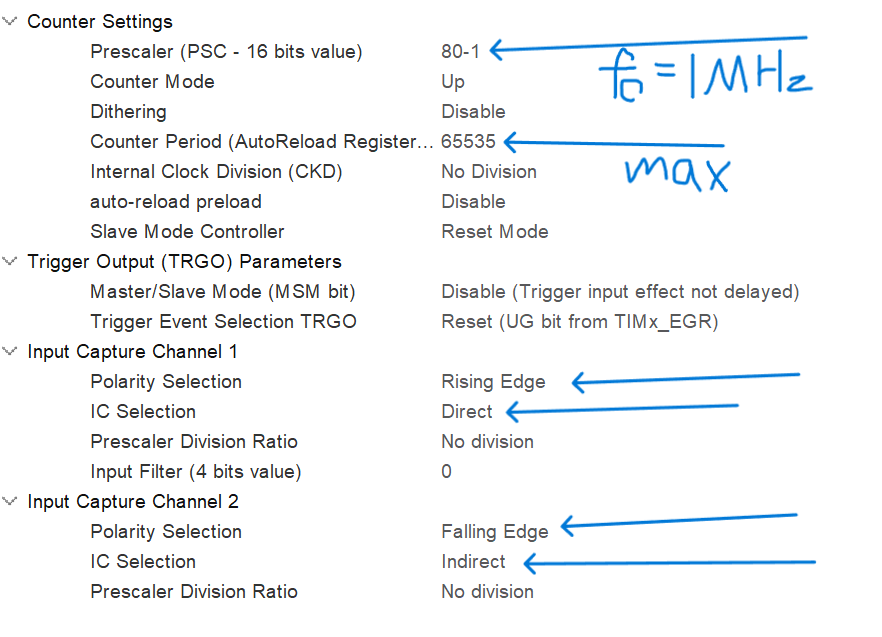


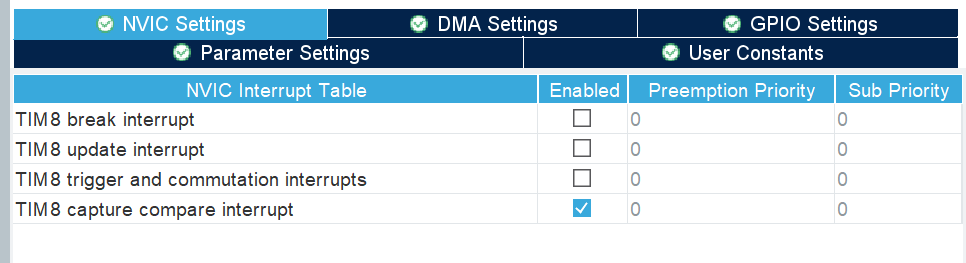


* Pulse为初始CCR的值，ARR影响周期（PWM的宽度）；(CCR/(ARR**+1**))\*100% = duty
* PWM生成不用开启中断
* ARR设置为100，这样compare(pulse)的值正好就是占空比（实际为'％'前的数字）
* frequency = pwm\_clk / (psc+1) / **(arr+1)**

**PWMI**

* PA15（TIM8）、PB4（TIM3）连接着信号发生器，用于测量信号
* PSC影响采样频率，ARR影响测量周期（设为最大值）
* 使能TIM3和TIM8，slave mode设为**reset mode**，Trigger source设为**TI1FP1**，channel1设为**input capture direct mode**，channel2设为**indirect mode**，NVIC中打开**capture compare中断**



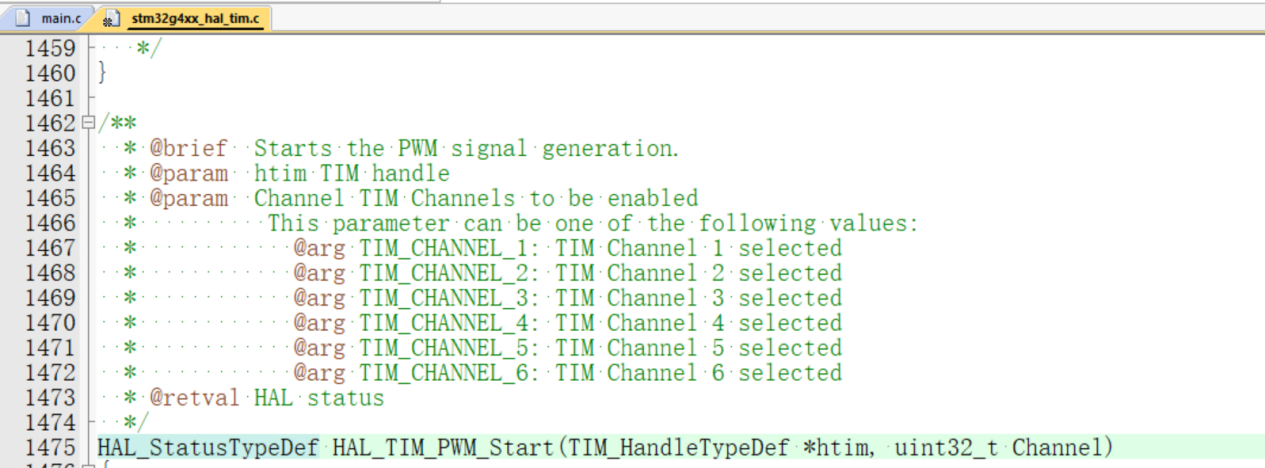


**2.函数**

**PWM Generate**

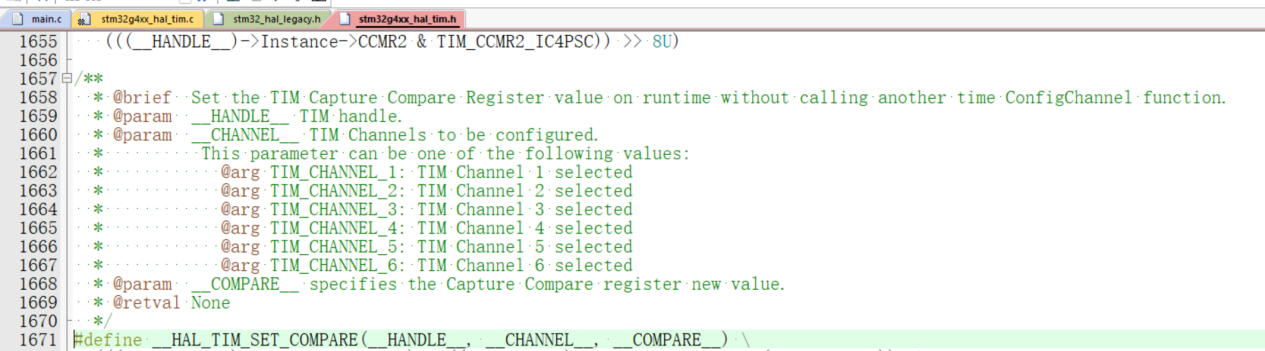
**HAL\_TIM\_PWM\_Start**

HAL\_TIM\_PWM\_Start(&htim16,TIM\_CHANNEL\_1);



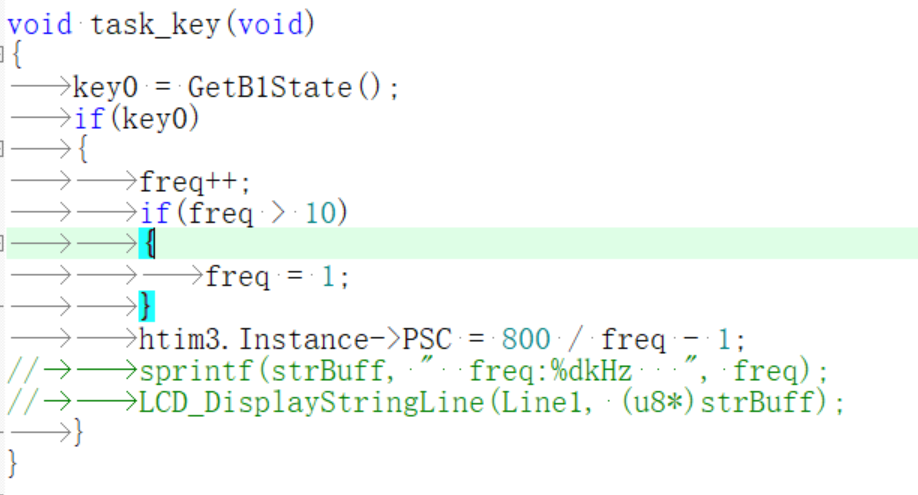
**\_\_HAL\_TIM\_SetCompare（调节占空比）**

\_\_HAL\_TIM\_SetCompare(&htim16,TIM\_CHANNEL\_1,duty1);



**调节频率**

* PSC最终值要减1
* 注意：不能只通过PSC++或--调节，频率可能不是均匀改变的



**PWMI**

1. 使用5个变量（具体类型根据函数返回值而定，未知就设为uint32\_t即可）

* 全周期捕获值capture\_full
* 下降沿捕获值capture\_half
* 定时器频率time\_freq
* PWM频率pwm\_freq
* 占空比duty

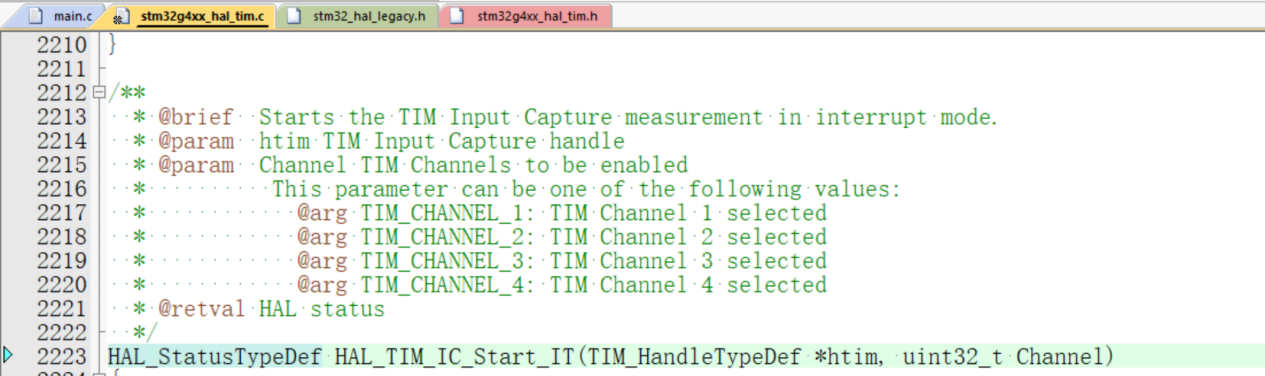
1. 捕获中断

* 判断对应的定时器
* 判断活动引脚：**ACTIVE**\_CHANNEL\_**1**时，获取全周期捕获值（**注意要加1**），计算定时器频率（注意除数**PSC要加1**），计算PWM的频率；ACTIVE\_CHANNLE\_**2**时，获取下降沿捕获值（注意加1），计算PWM占空比（下降沿捕获值\*100/全周期捕获值，\*100是为了取出%前的数，如果精度要求较高，也可以定义为double类型）

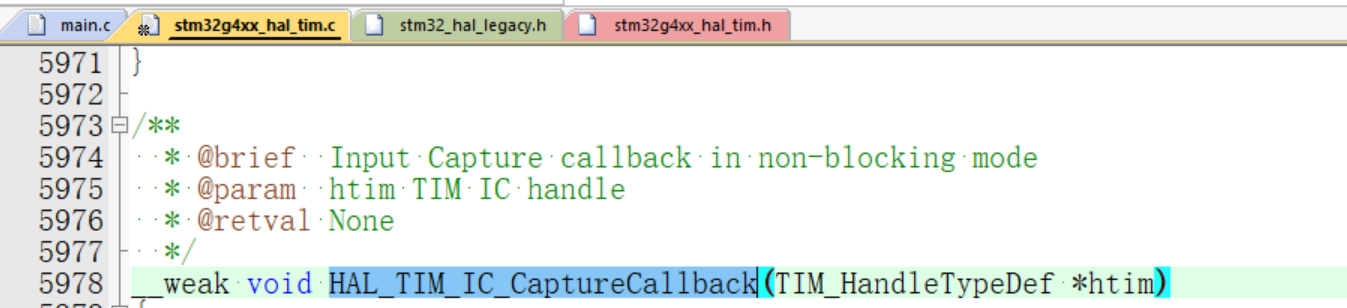
**HAL\_TIM\_IC\_Start\_IT**

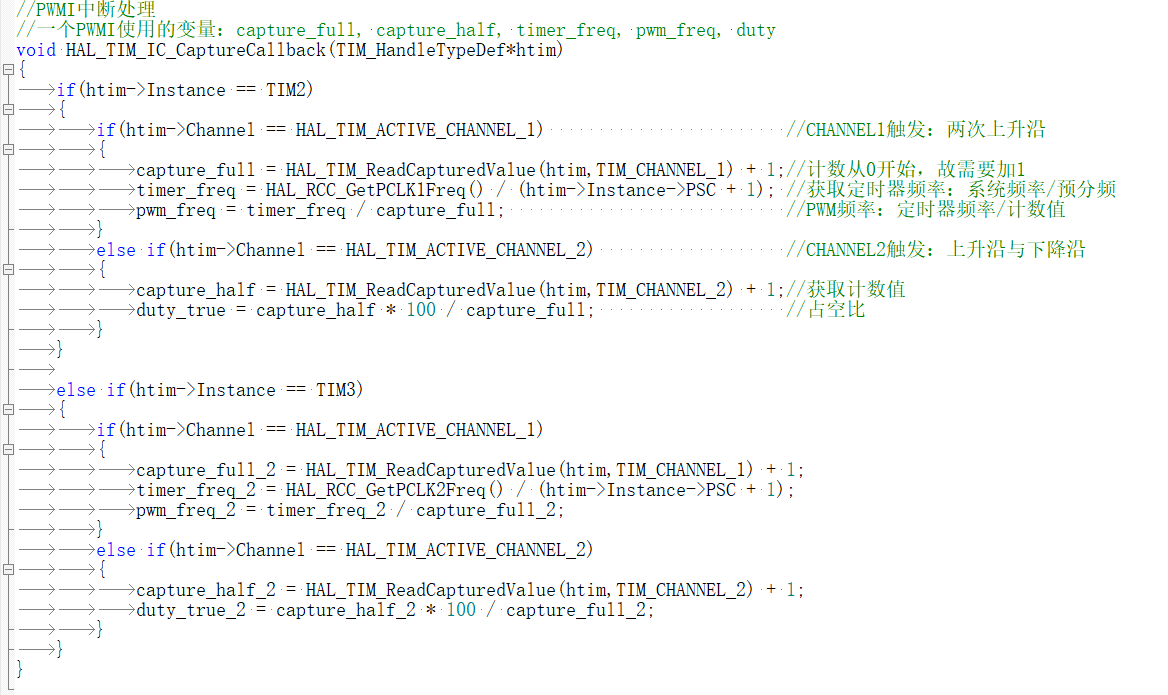
HAL\_TIM\_IC\_Start\_IT(&htim3,TIM\_CHANNEL\_1);

HAL\_TIM\_IC\_Start\_IT(&htim3,TIM\_CHANNEL\_2);//两个通道都要打开中断

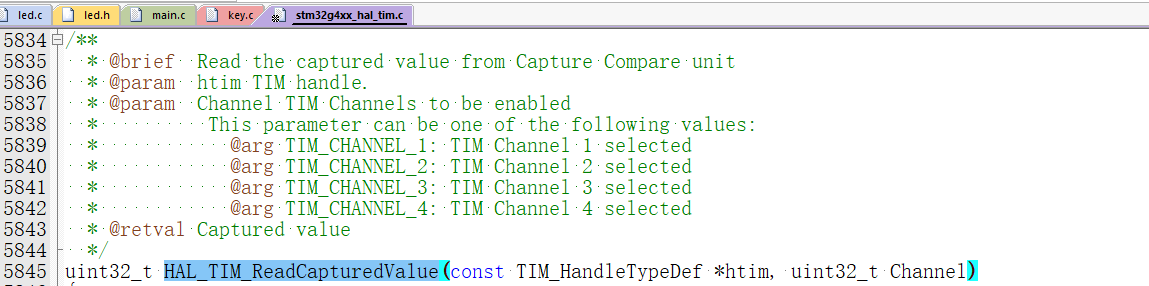


**HAL\_TIM\_IC\_CaptureCallback**



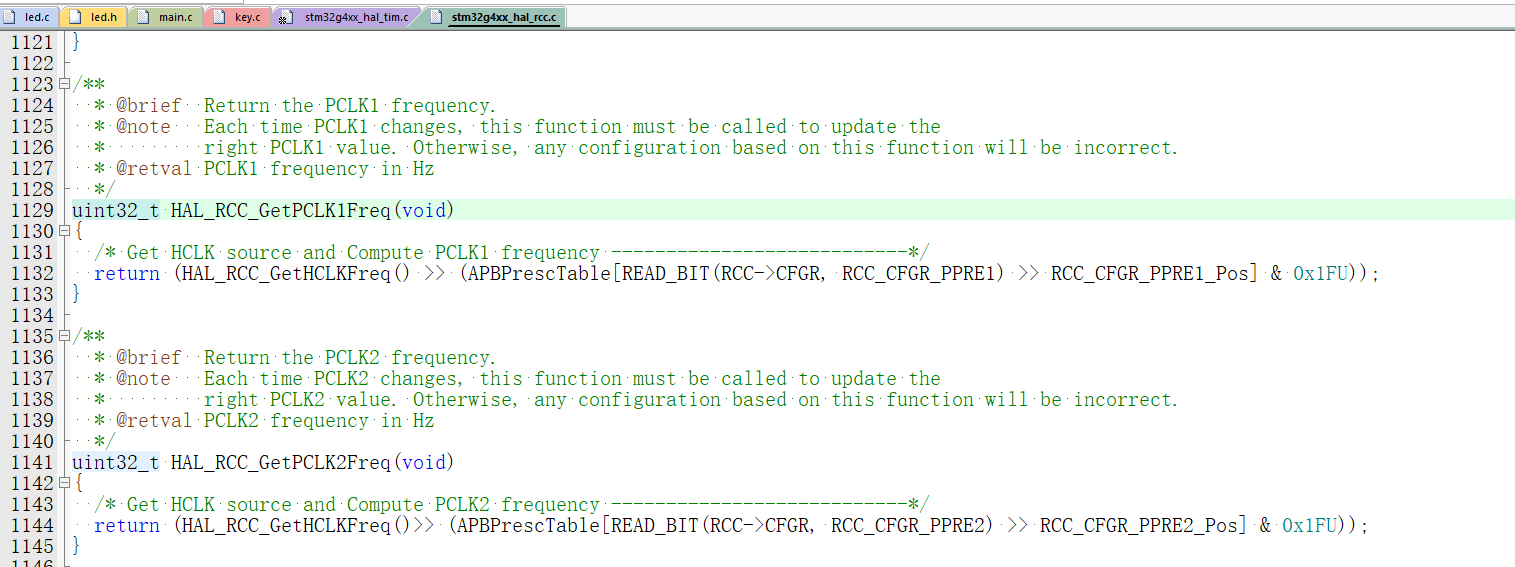


**HAL\_TIM\_ReadCapturedValue**



**HAL\_RCC\_GetPCLK1Freq**

* 注意：具体需要结合时钟树调用对应的函数，或者直接填写系统主频（cubemx中没有分频的前提下）



**htim->Instance->PSC**

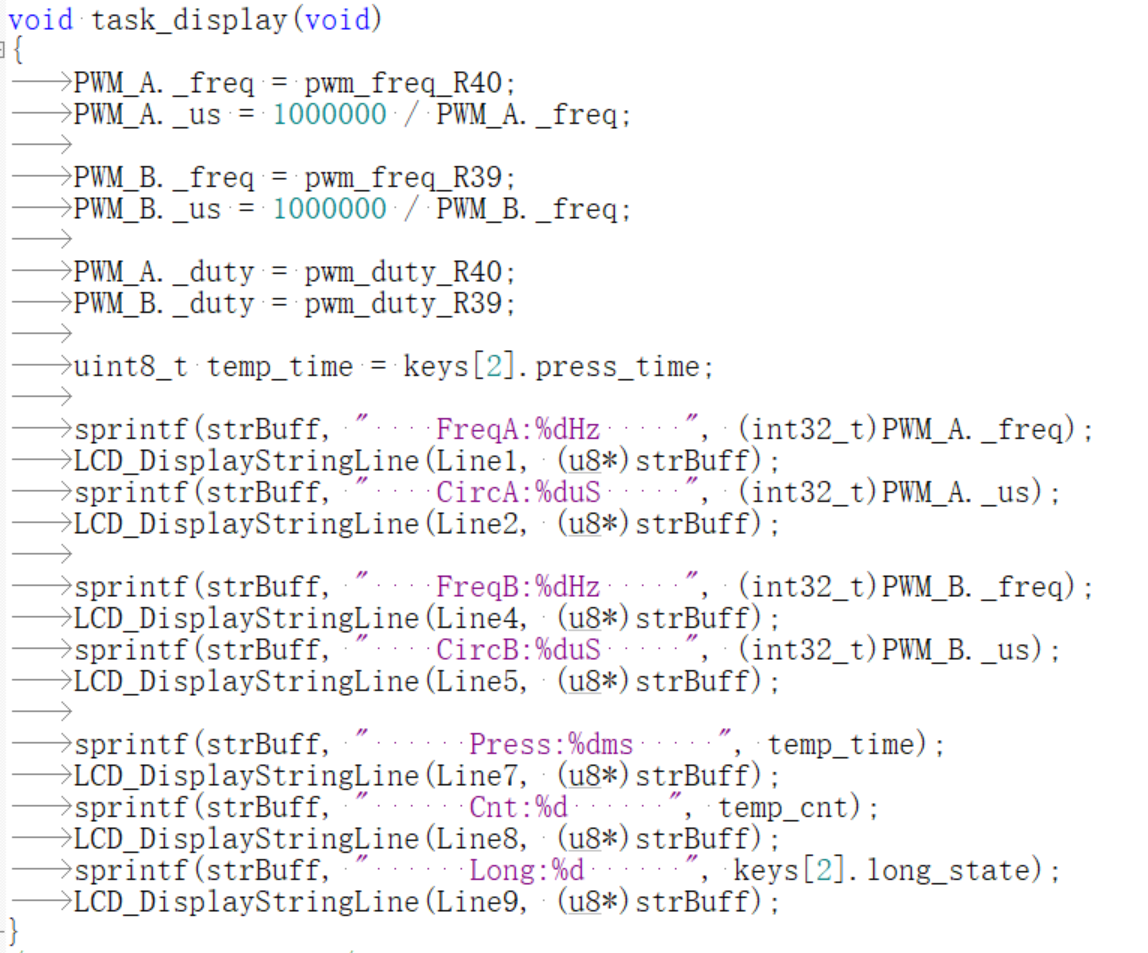
从0开始，使用时需要加1

**3.相关文件**

stm32g4xx\_hal\_tim.c、.h

**4.注意事项**

* 使用前要打开PWM输出/输入捕获中断，输入捕获中断需要打开两个通道（包括cubemx和代码）
* 配置时，需要打开**capture compare中断**
* 判断捕获的通道用HAL\_TIM\_**ACTIVE**\_CHANNEL\_x，不是TIM\_CHANNEL\_x
* LCD显示PWMI参数：先用其他变量接收，再用LCD显示

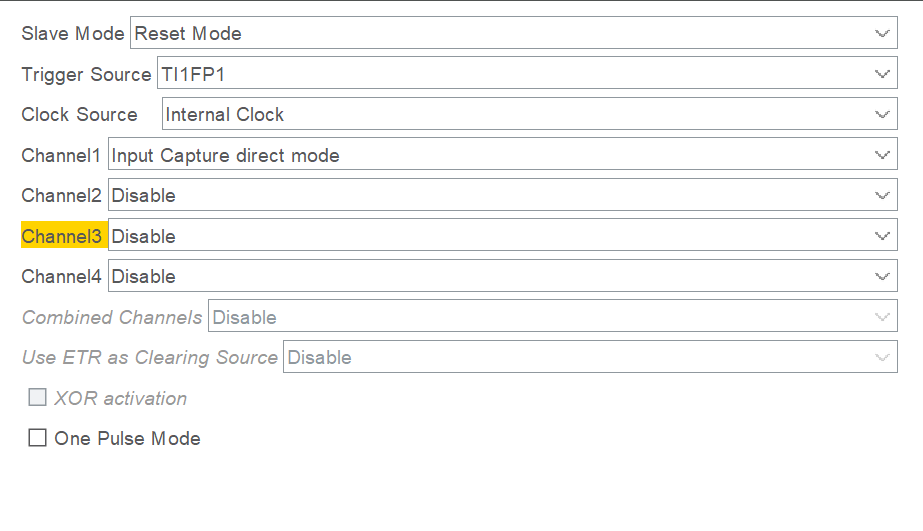


* 一个定时器可以同时测量频率相同的两路信号的占空比和频率（取消**从模式**，在其中一个通道的上升沿处理中清空计数即可）

**0502PWM只测量频率**

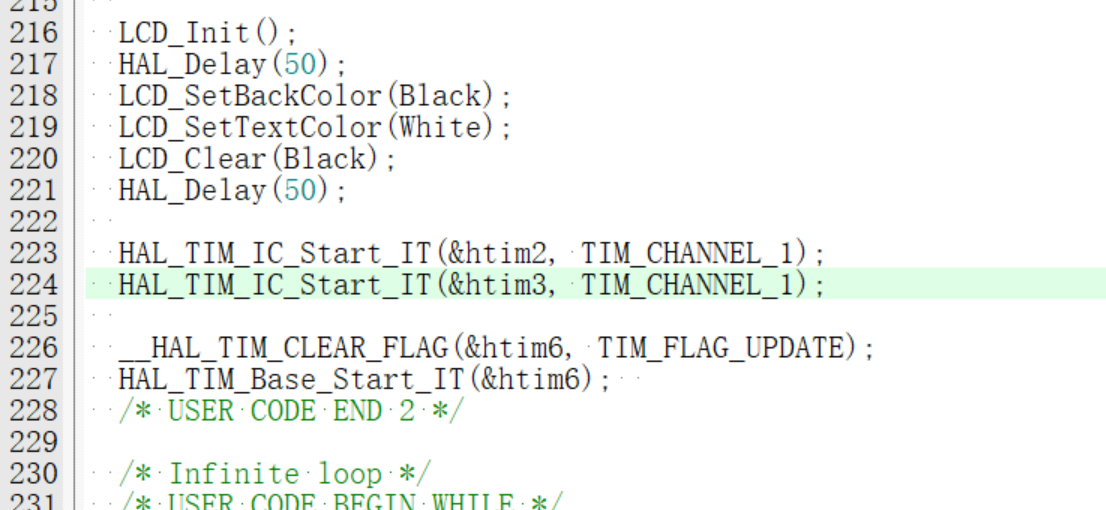
**1.配置**

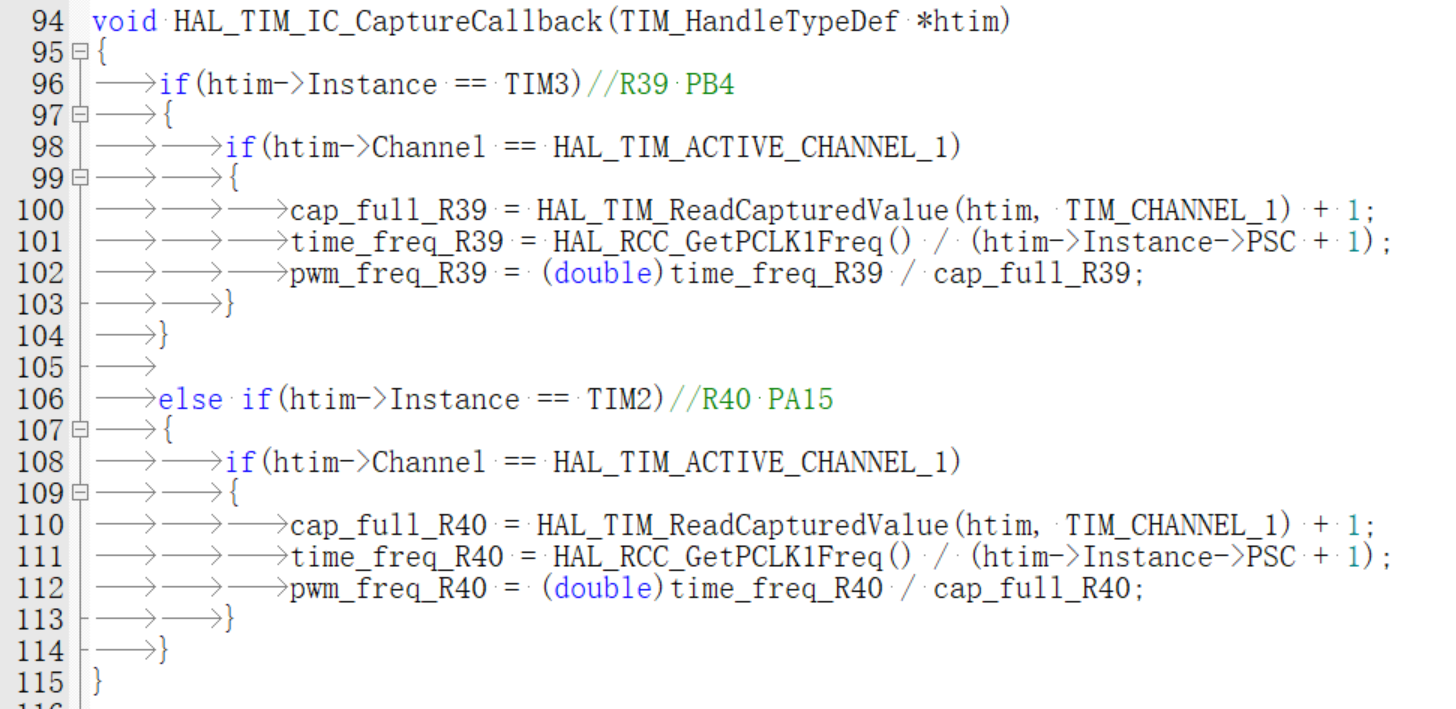
* CHANNEL2不用配置，设为Disable



**2.函数**

* CHANNEL2不用开启中断
* 中断处理中删去CHANNEL2分支即可



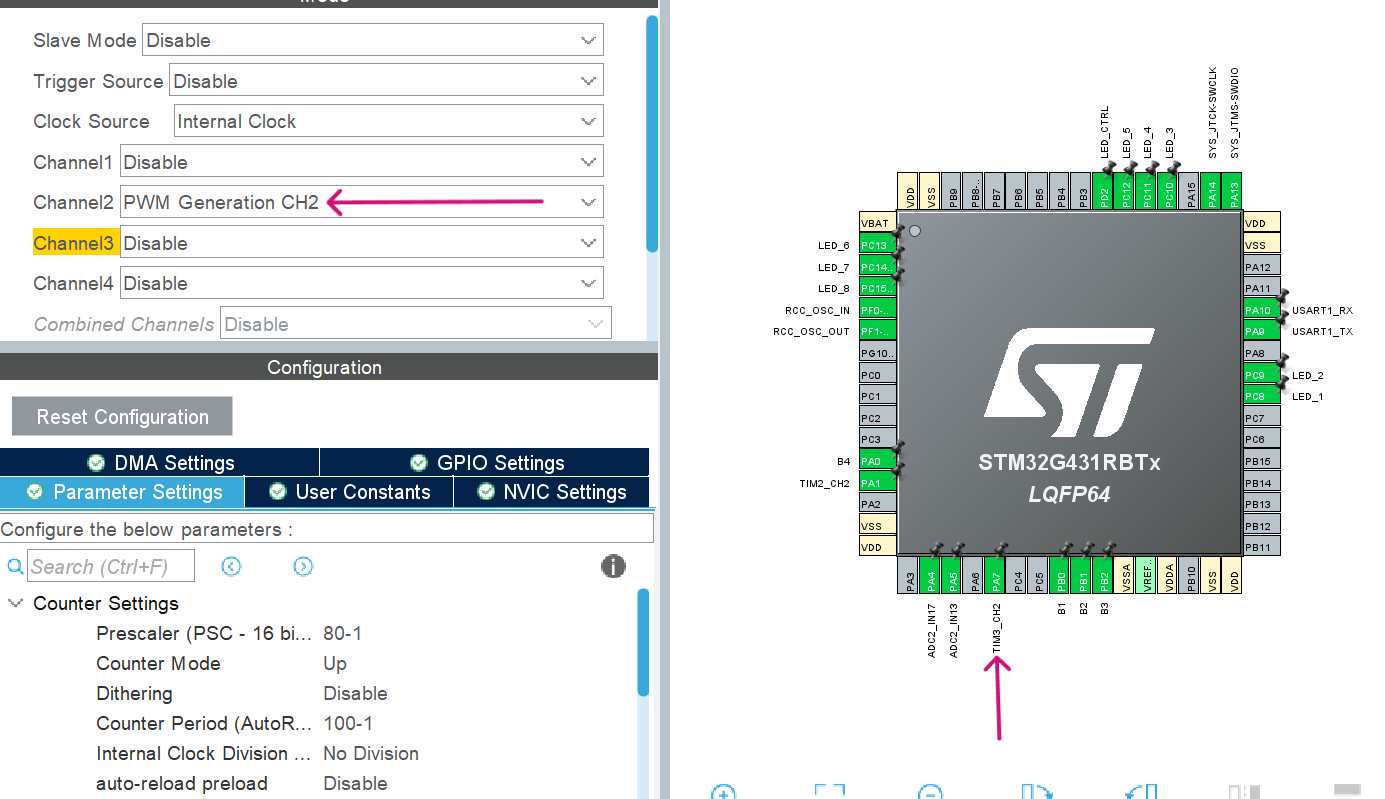


**3.注意事项**

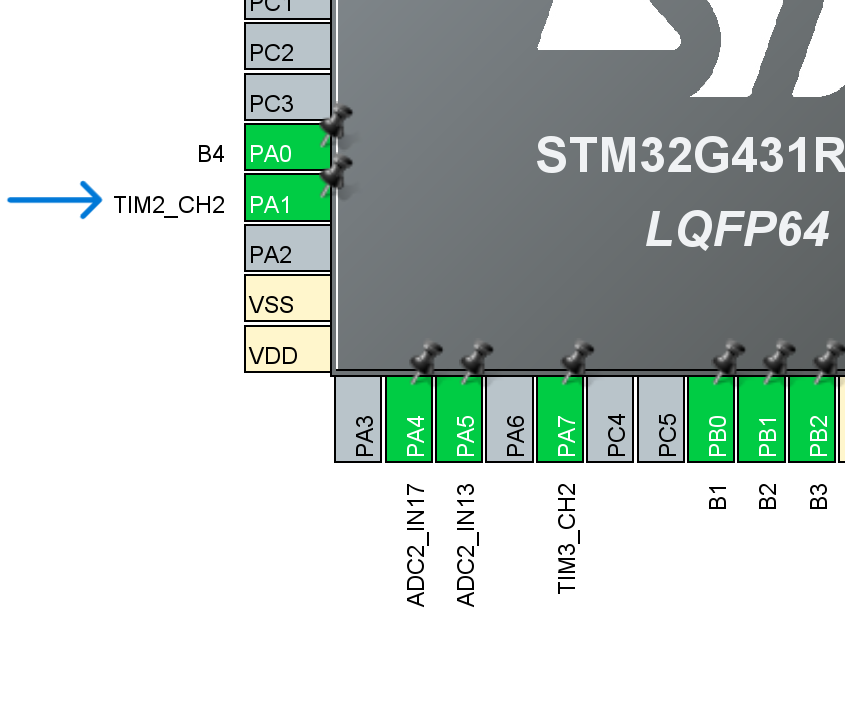
**0503PWM通道二产生信号与测量**

**1.配置**

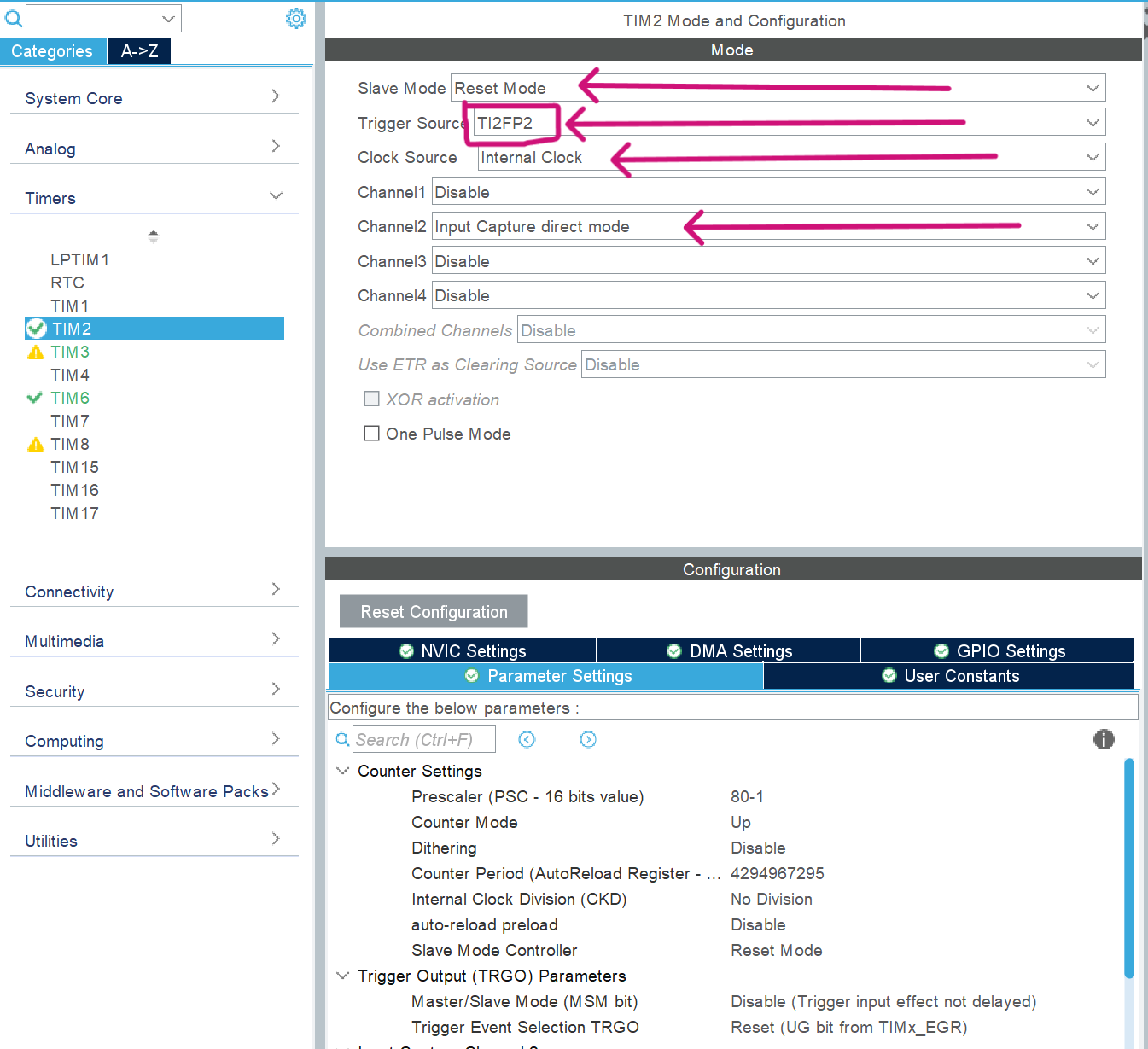
**PWM**

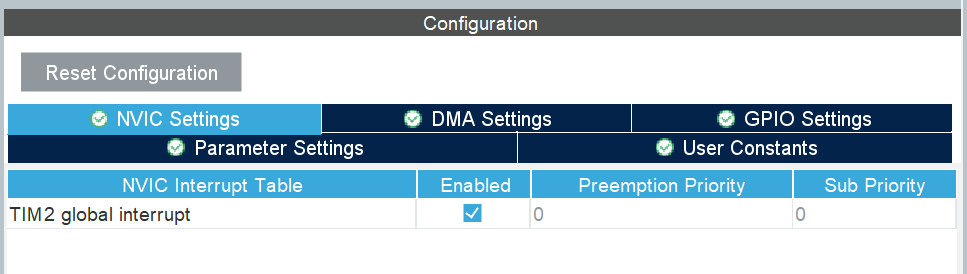


**PWMI**



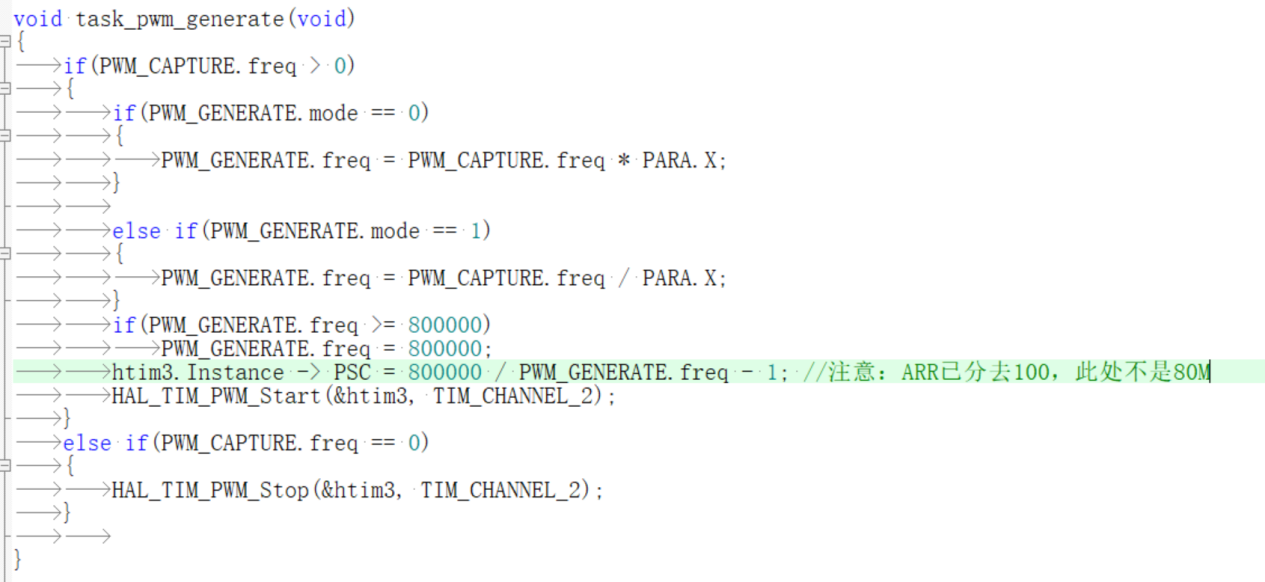
* Trigger Source需要改为TI2FP2（从通道2输入，输出到内部通道2）





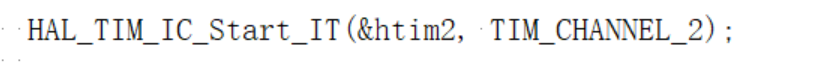
**2.函数**

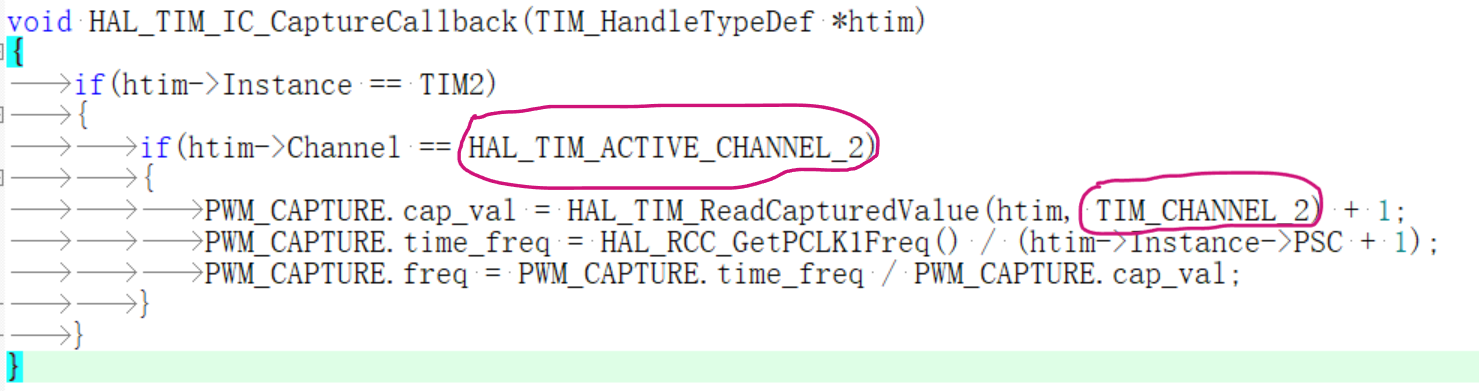
**PWM-频率调节**



**PWMI**

* 对应的通道由CHANNEL\_1改为CHANNEL\_2





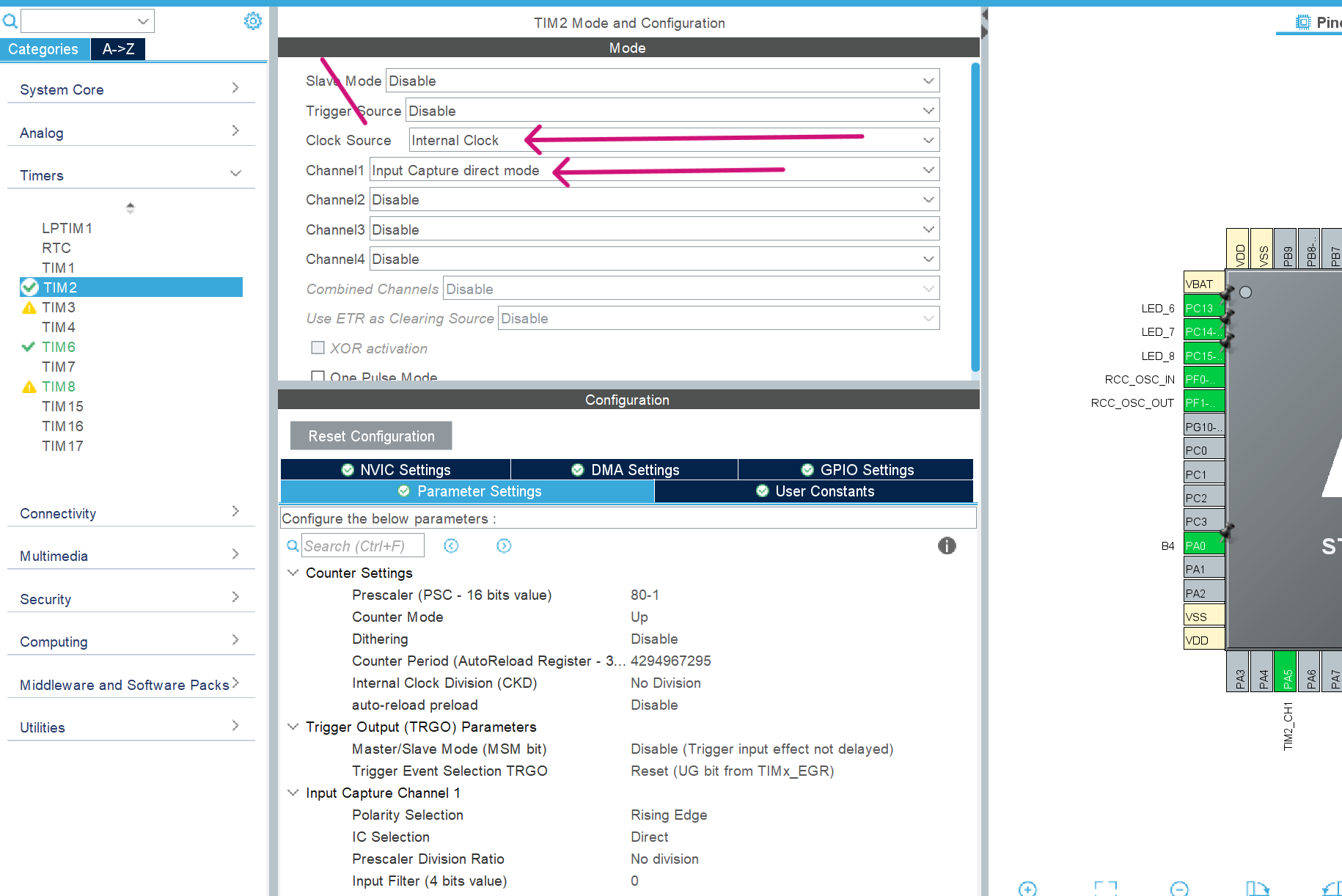
**3.注意事项**

* PWM\_Generation中，占空比需要大于0才能生成信号
* PWM\_Generation中，求PSC时**80M**需要先除以(ARR+1)
* 此处PWMI只实现了测量频率，若需要测量占空比，则打开通道一，再补齐代码

**0504PWMI-非从模式**

**1.配置**

* 去掉Slave Mode 和Trigger Source，其余不变

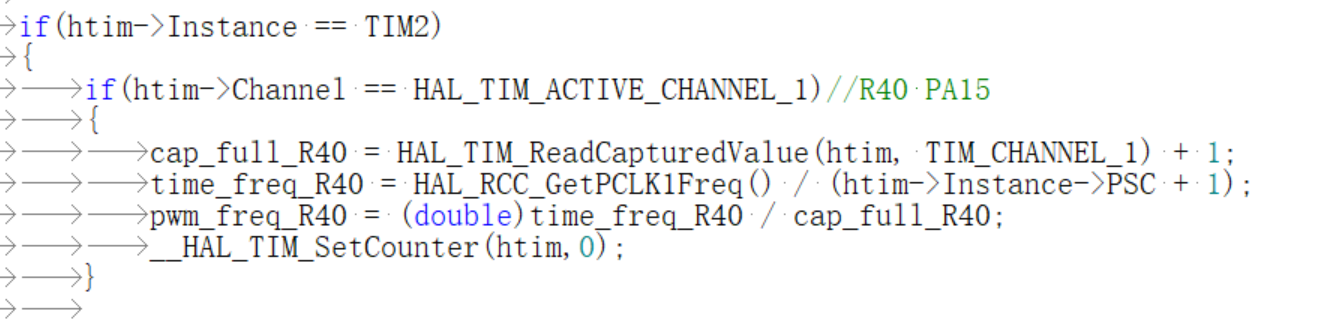


**2.函数**

* 初始化



* 中断函数处理：最后**将定时器计数清零**



**3.注意事项**

* 处理函数中注意清空计数

**0505PWM输出指定频率说明**

单位：Hz（通用）

已知：Freq = x Hz

**则 PSC = 80M/x/(ARR+1)-1**

单位：kHz（除多1000）

已知：Freq = x kHz

**则PSC = 80M/(x\*1000)/(ARR+1)-1**

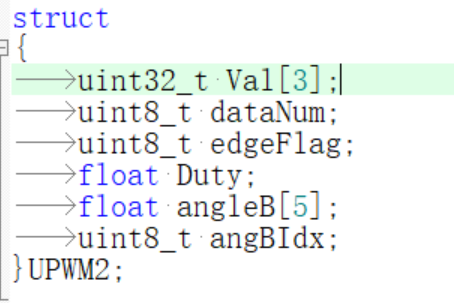
**0506单引脚测量占空比**

**1.配置**

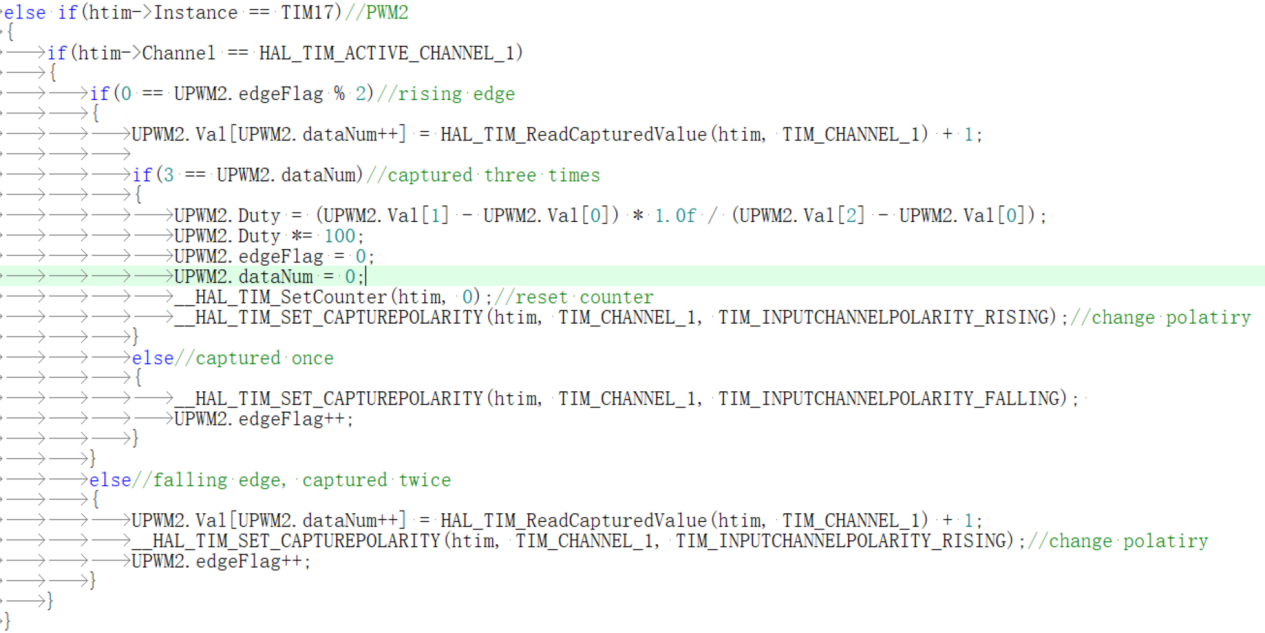
**触发方式选择上升沿触发**

**2.函数**

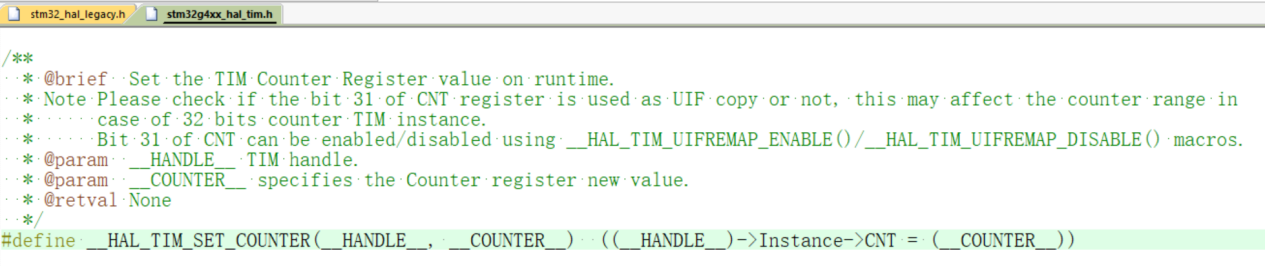
* 结构体属性：存放捕获值的容器（3个数据），当前捕获到的数据个数，边沿触发标志位，占空比



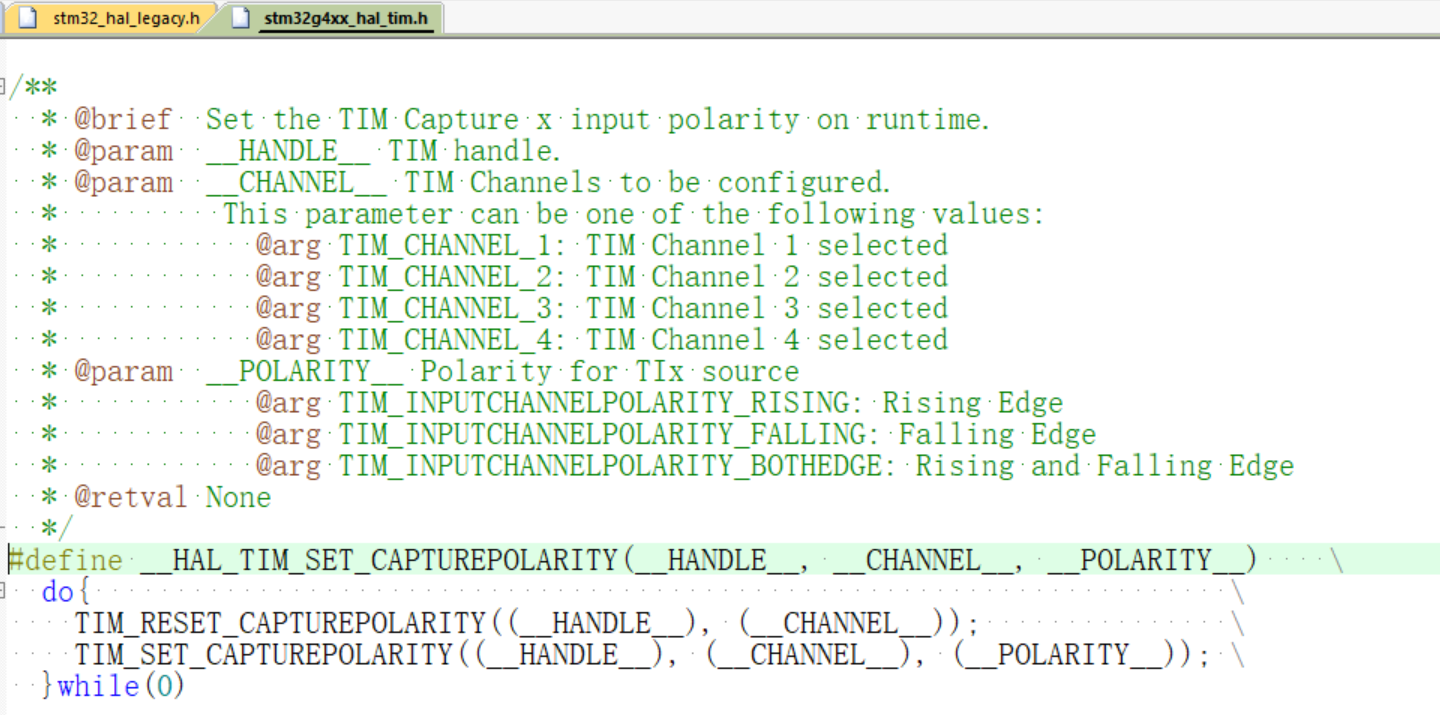
* edgeFlag%2为0时说明此次为上升沿捕获，为1说明此次为下降沿捕获
* 捕获次数到达3时，说明捕获到一个周期，进行计算和参数初始化，**清空定时器计数**
* 第一次捕获->切换到下降沿触发，第二次捕获（下降沿触发）/ 捕获完一个周期->切换到上升沿触发



* \_\_HAL\_TIM\_SetCounter



* \_\_HAL\_TIM\_SET\_CAPTUREPOLARITY



**3.注意事项**

* 需要在中断函数中重新指定触发模式
* cubemx中的配置还是选择上升沿触发