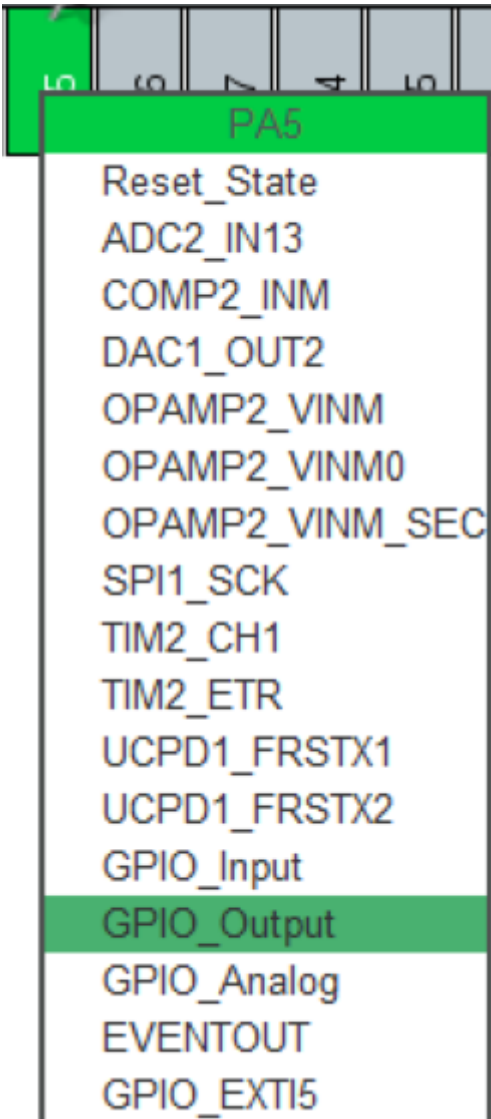# 实验2：中断与GPIO

1. CubeMX配置

1. 配置LD2引脚PA5（虽然默认配置已经配置好了，但还是再做演示）

选中PA5引脚（具体每个GPIO引脚具有的功能可查看datasheet），选择 `GPIO_Output`

Pin N... ⬆	Signal o...	GPIO ou...	GPIO m...	GPIO Pu...	Maximu...	Fast Mode	User Label	Modified
PA5	n/a	Low	Output ...	No pull-u...	Low	n/a	LD2 [gre...	☑
PC13	n/a	n/a	External...	No pull-u...	n/a	n/a	B1 [blue ...	☑

┌ PA5 Configuration :

GPIO output level	Low ⌄
GPIO mode	Output Push Pull ⌄
GPIO Pull-up/Pull-down	No pull-up and no pull-down ⌄
Maximum output speed	Low ⌄
User Label	LD2 [green]

　　进入左侧System->GPIO界面，这里有所有选中的引脚，选择PA5，配置同默认配置
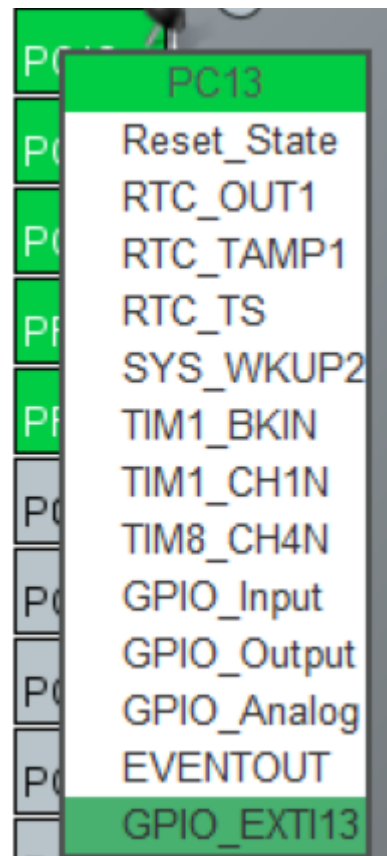
2. 配置PC13外部中断

选择PC13，选中功能GPIO_EXIT13

Pin N... ⬆	Signal o...	GPIO ou...	GPIO m...	GPIO Pu...	Maximu...	Fast Mode	User Label	Modified
PA5	n/a	Low	Output ...	No pull-u...	Low	n/a	LD2 [gre...	☑
PC13	n/a	n/a	External...	No pull-u...	n/a	n/a	B1 [blue ...	☑

PC13 Configuration :

GPIO mode	External Interrupt Mode with Rising edge trigger detection ⌄
GPIO Pull-up/Pull-down	No pull-up and no pull-down ⌄
User Label	B1 [blue push button]

同上配置PC13，基本如默认配置

GPIO mode	External Interrupt Mode with Rising edge trigger detection ⌄

External Interrupt Mode with Rising edge trigger detection 上升沿触发
External Interrupt Mode with Falling edge trigger detection 下降沿触发
External Interrupt Mode with Rising/Falling edge trigger detection 边沿触发
External Event Mode with Rising edge trigger detection
External Event Mode with Falling edge trigger detection
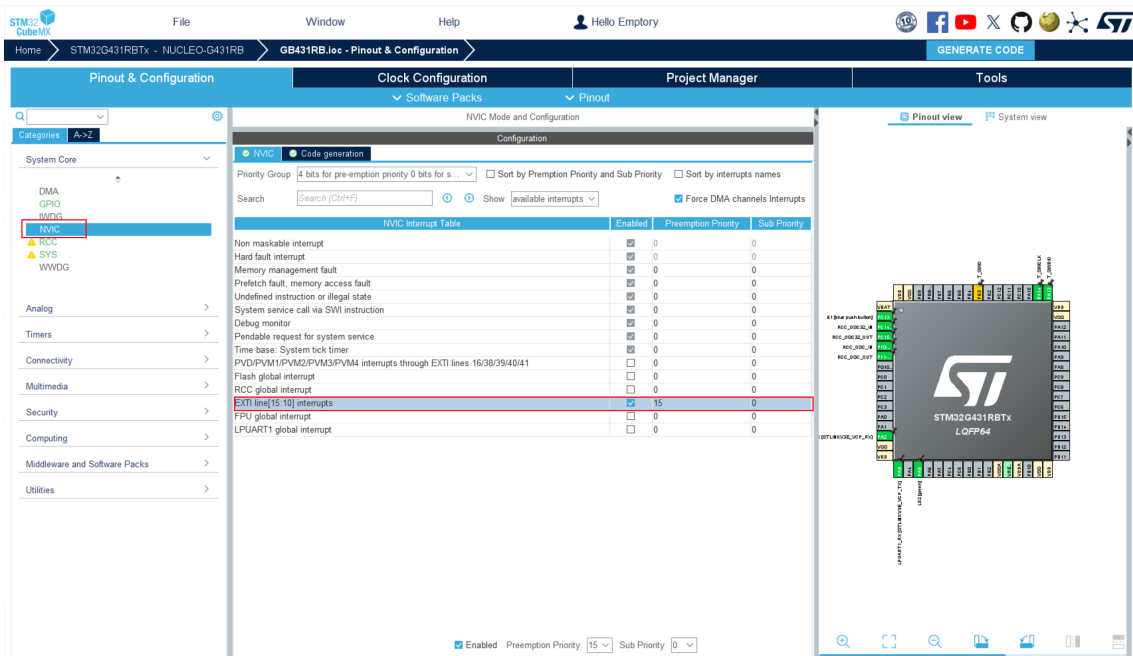External Event Mode with Rising/Falling edge trigger detection

外部中断

GPIO mode 可以自己选择别的mode试试
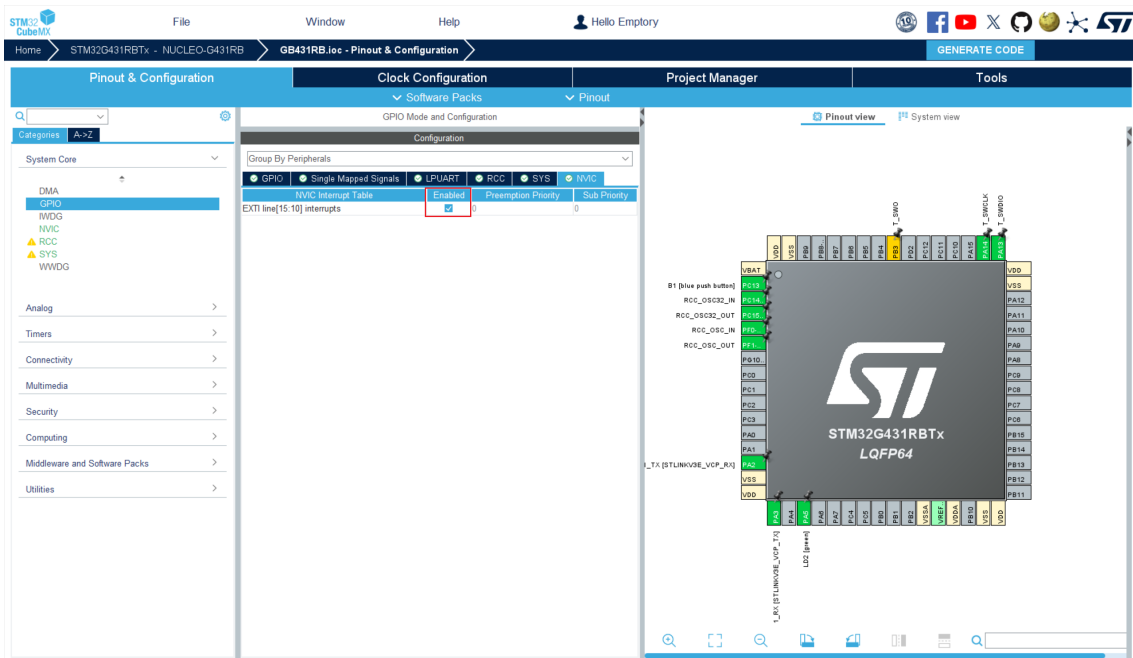
进入GPIO->NVIC中启用EXIT13的NVIC通道

进入NVIC，对EXTI13的NVIC通道优先级进行配置

> 优先级的数值越小，则优先级越高，也就是说优先级设置为0，那么优先级将是最高的，反之
> 255优先级最小，但是这里只取4 bits，这也是可以再Priority Group中设置的

3. Generate Code后使用MDK进行编写

2. 可能用到的hal库函数

```
1  GPIO_PinState HAL_GPIO_ReadPin(GPIO_TypeDef *GPIOx, uint16_t GPIO_Pin)
2  /*
3      读取某GPIO引脚电平值
4  */
5  //示例
6  HAL_GPIO_ReadPin(B1_GPIO_Port,B1_Pin) == GPIO_PIN_SET
```

```
1  void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
2  /*
```

```
 3        外部中断函数
 4        原型
 5
 6        __weak void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
 7
 8        __weak标识符代表其可被user重载
 9  */
10  //示例，只需在main.c或者其他文件处，调用HAL_GPIO_EXTI_Callback后重写具体函数内容
    即可
11  void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
12  {
13      if(GPIO_Pin == B1_Pin)
14      {
15          HAL_Delay(10);
16          {
17              if(HAL_GPIO_ReadPin(B1_GPIO_Port,B1_Pin) == GPIO_PIN_SET)
18              {
19                  HAL_GPIO_TogglePin(LD2_GPIO_Port,LD2_Pin);
20              }
21          }
22
23      }
24  }
```