Mark	1/11

Team name:	*B5*		
Homework number:	*HOMEWORK 02*		
Due date:	01/10/24		
Contribution	NO	Partial	Full
Marenghi Manuela			*x*
Fellegara Tommaso			*x*
Giammusso Samuele			*x*
Cattani Luca			*x*
Csata Dániel			*x*
Notes: none			

Project name	Test		
Not done	Partially done (major problems)	Partially done (minor problems)	Completed
			x

Part 1 a - Modify the status (switch on / off) of the NUCLEO green LED, every time you snap your fingers

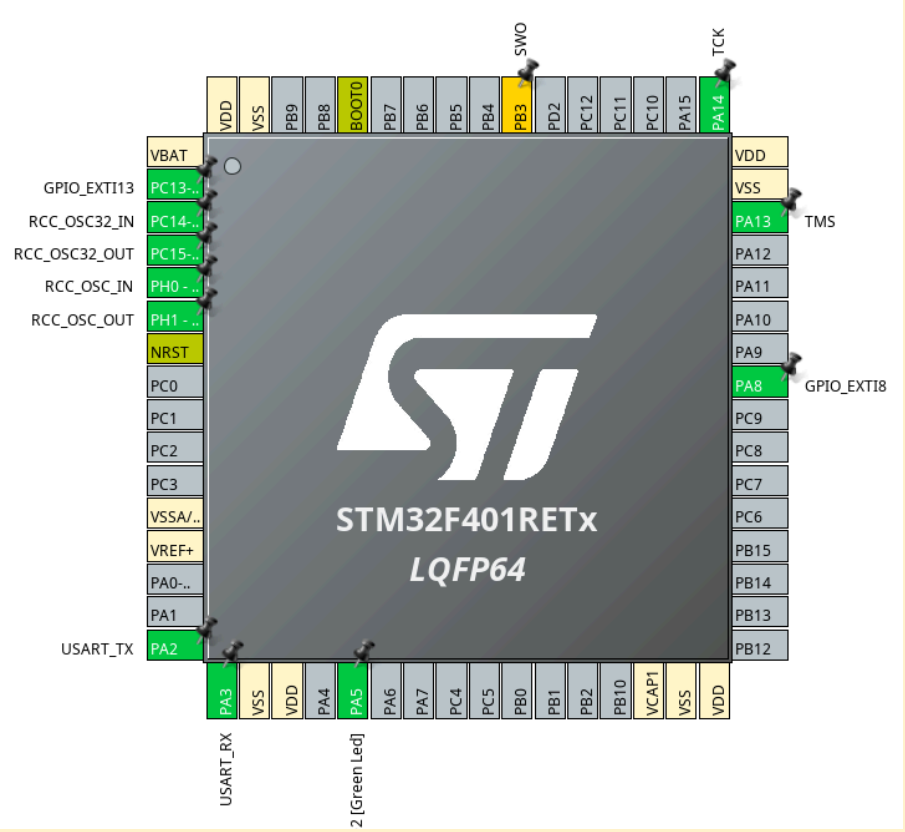
- To find the pin of the microphone:

we opened the pdf “Hands on lab schematic”:
search “microphone” and find that is connected to SND_DETECT
search SND_DETECT and find that is connected to SND_IN
search SND_IN and find that the pin is PA8.

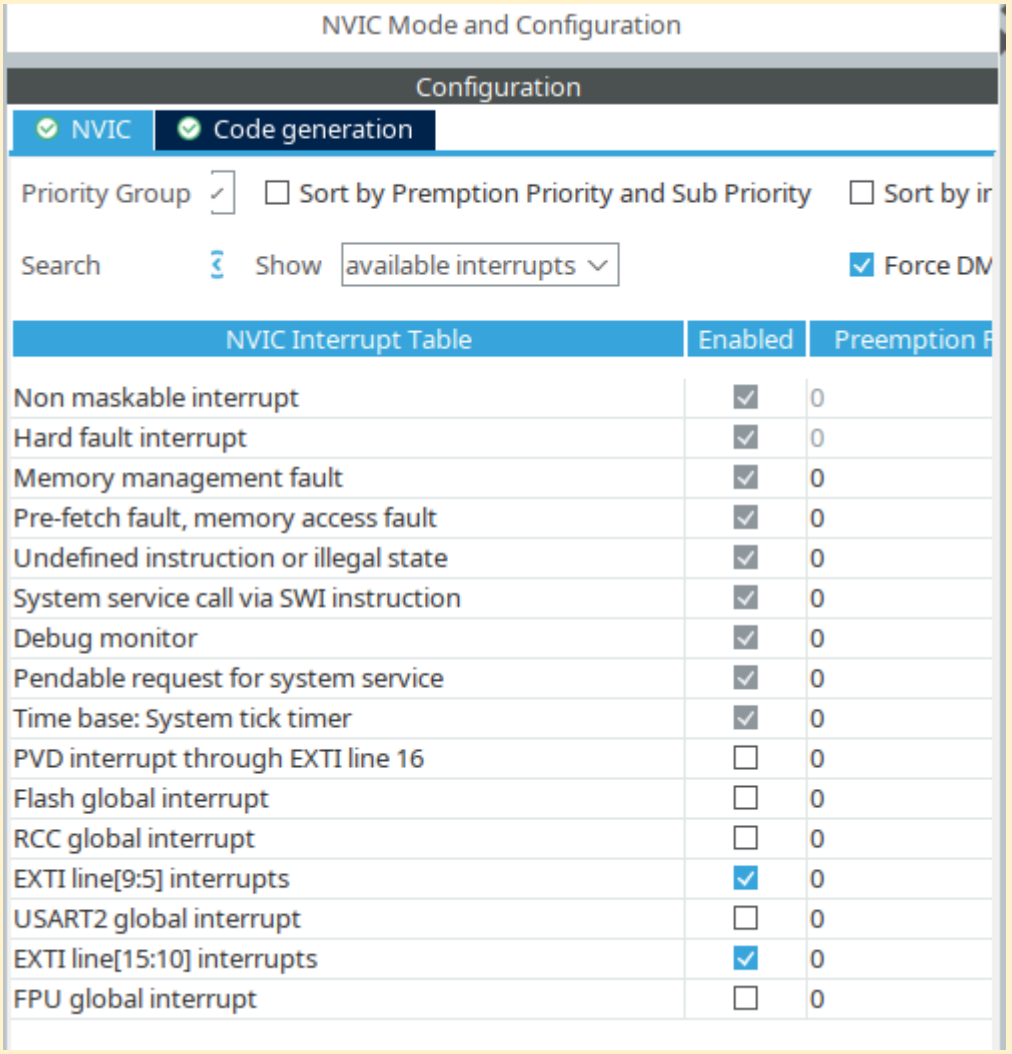
So the pin is PA8

Another way is to search directly for SND_IN (as specified in the homework assignment) and follow the trace of the circuit until PA8

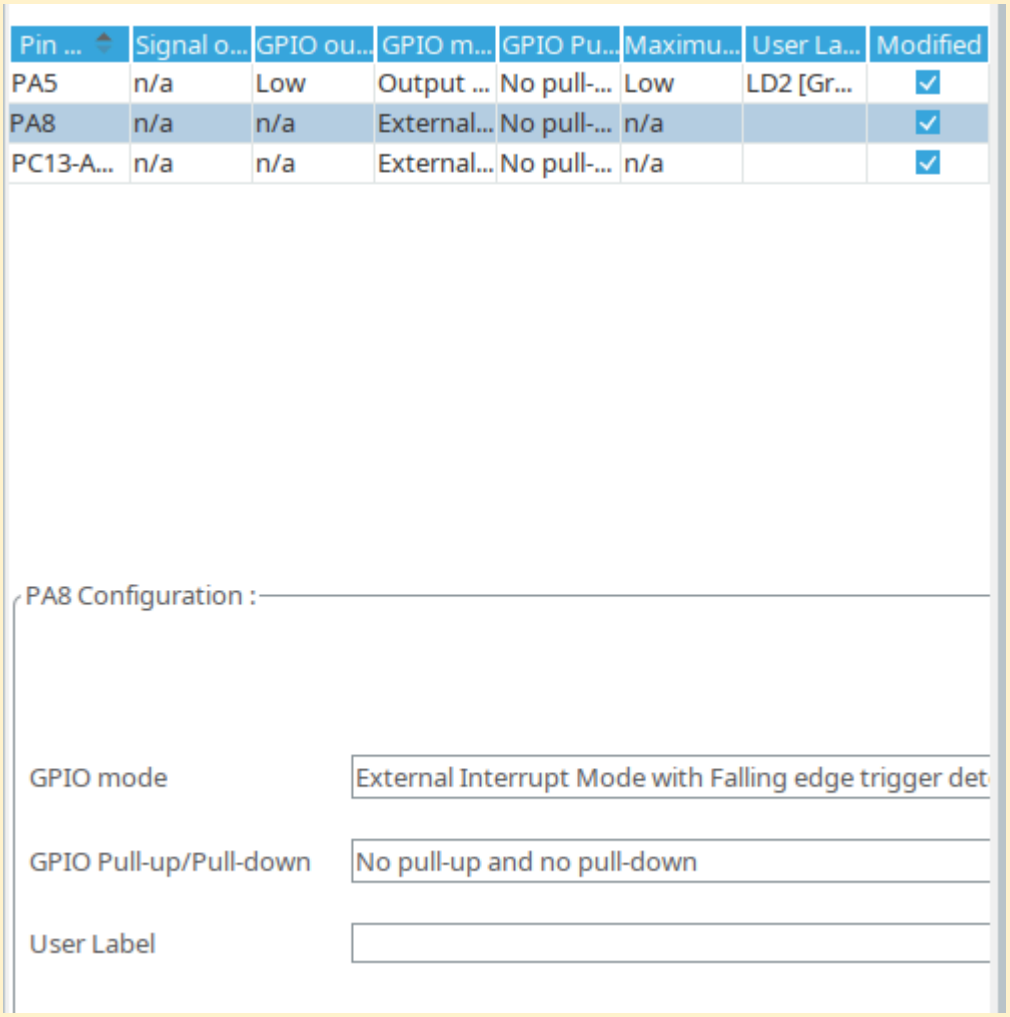
- We first set the PA8 pin to GPIO_EXTI8



- We then enable the EXTI line 15 to 10 and also the EXTI line 9 to 5 interrupts line in the NVIC



- Then we set the GPIO mode of the pin PA8 to External Interrupt mode with Falling edge trigger detection, we decided to use the falling edge to toggle the state of the LED when snapping our fingers (instead of making the LED blink)

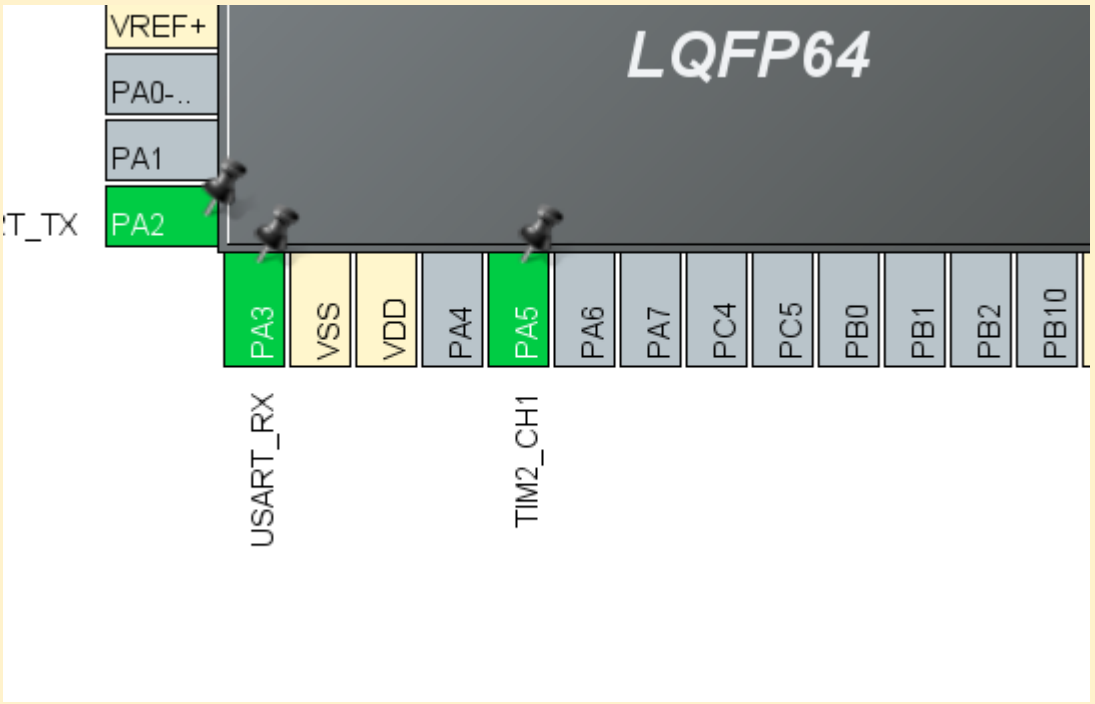


- Saving the previous controller configuration on the IDE creates the code for the ISR, which calls the HAL_GPIO_EXTI_Callback function, passing as argument GPIO_Pin == GPIO_PIN_8, this allows us to extended the EXTI callback code developed in class to toggle the status of the LED on microphone input as follows (added the second switch case):

```
/* USER CODE BEGIN 0 */
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    switch(GPIO_Pin){
    case GPIO_PIN_13: // keep the led on (or off) while the button is press
        GPIO_PinState pushbutton = HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_13);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, !pushbutton);
        break;
    case GPIO_PIN_8: // toggle the status of the led
        GPIO_PinState ledStatus = HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_5);
        HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, !ledStatus);
        break;
    default:
        break;
    }
}
```

Part 1 b - Make the NUCLEO green LED blink at a 1 Hz rate using PWM generation on the corresponding channel

- Set the PA5 pin (the green led pin) to TIM2_CH1



- To achieve a 1 Hz, use this formula:

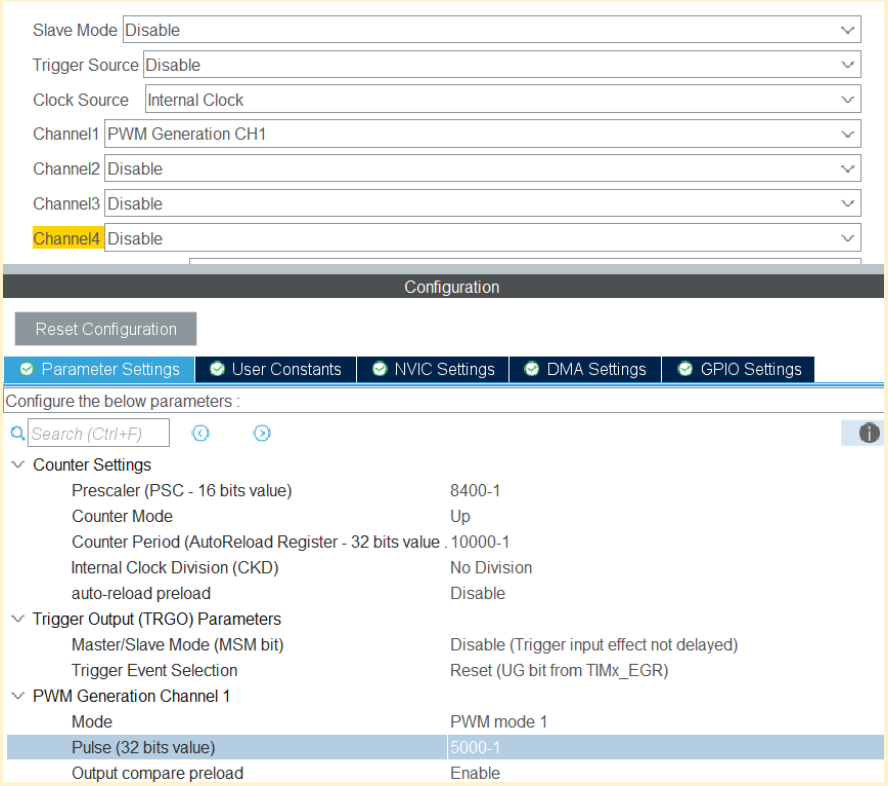
$$f_{PWM} = \frac{f_{TIM}}{(ARR+1)\cdot(PSC+1)} = 1\,Hz$$

f_{TIM}: 84 MHz

$ARR+1$: $10'000 - 1$

$PSC+1$: $8400 - 1$

- Then set up the timer as seen in class:



- Then add this HAL function to activate the PWM

```
96  /* USER CODE BEGIN 2 */
97
98  HAL_TIM_PWM_Start(&htim2,TIM_CHANNEL_1);
99
100 /* USER CODE END 2 */
```

- Then build and run the code to see the led blinking

Professor comments: