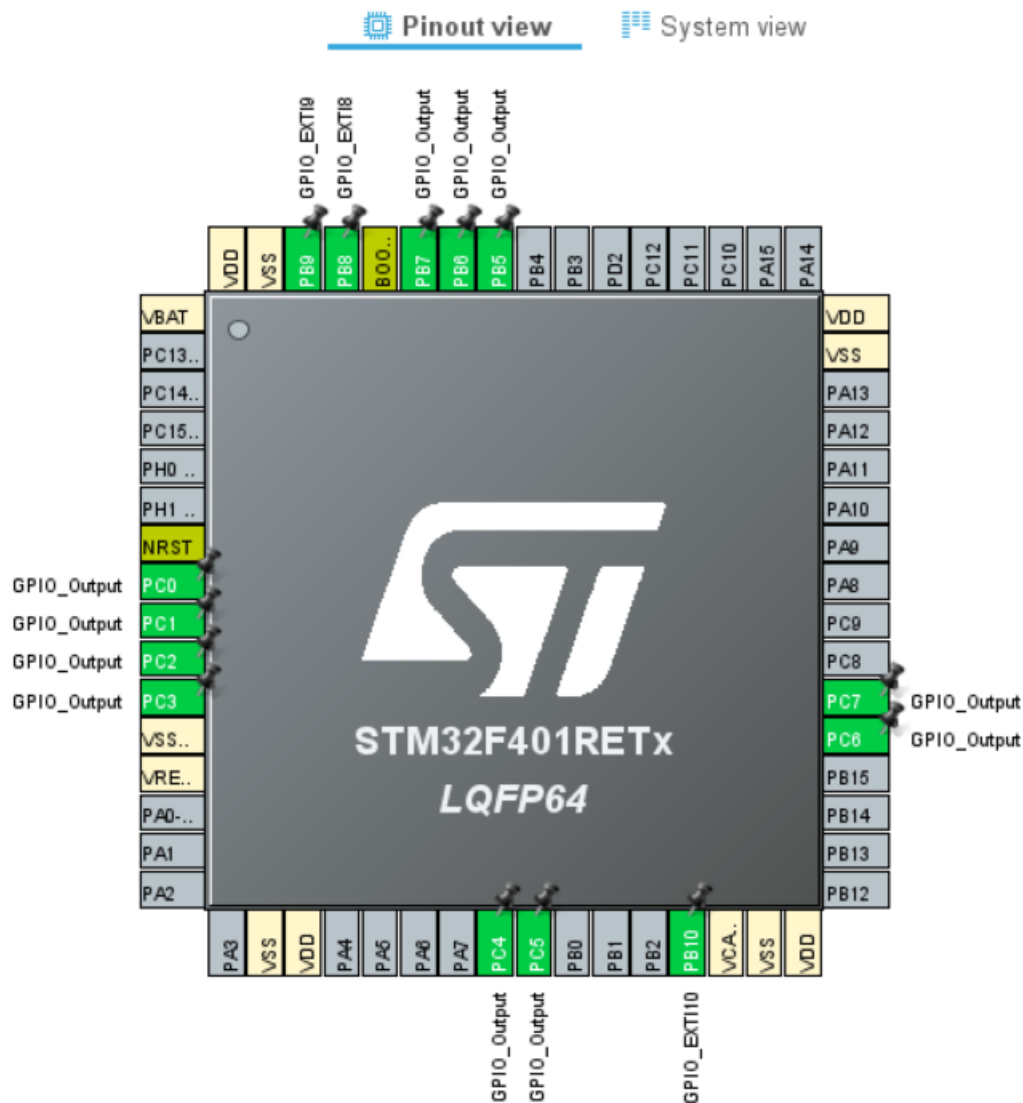


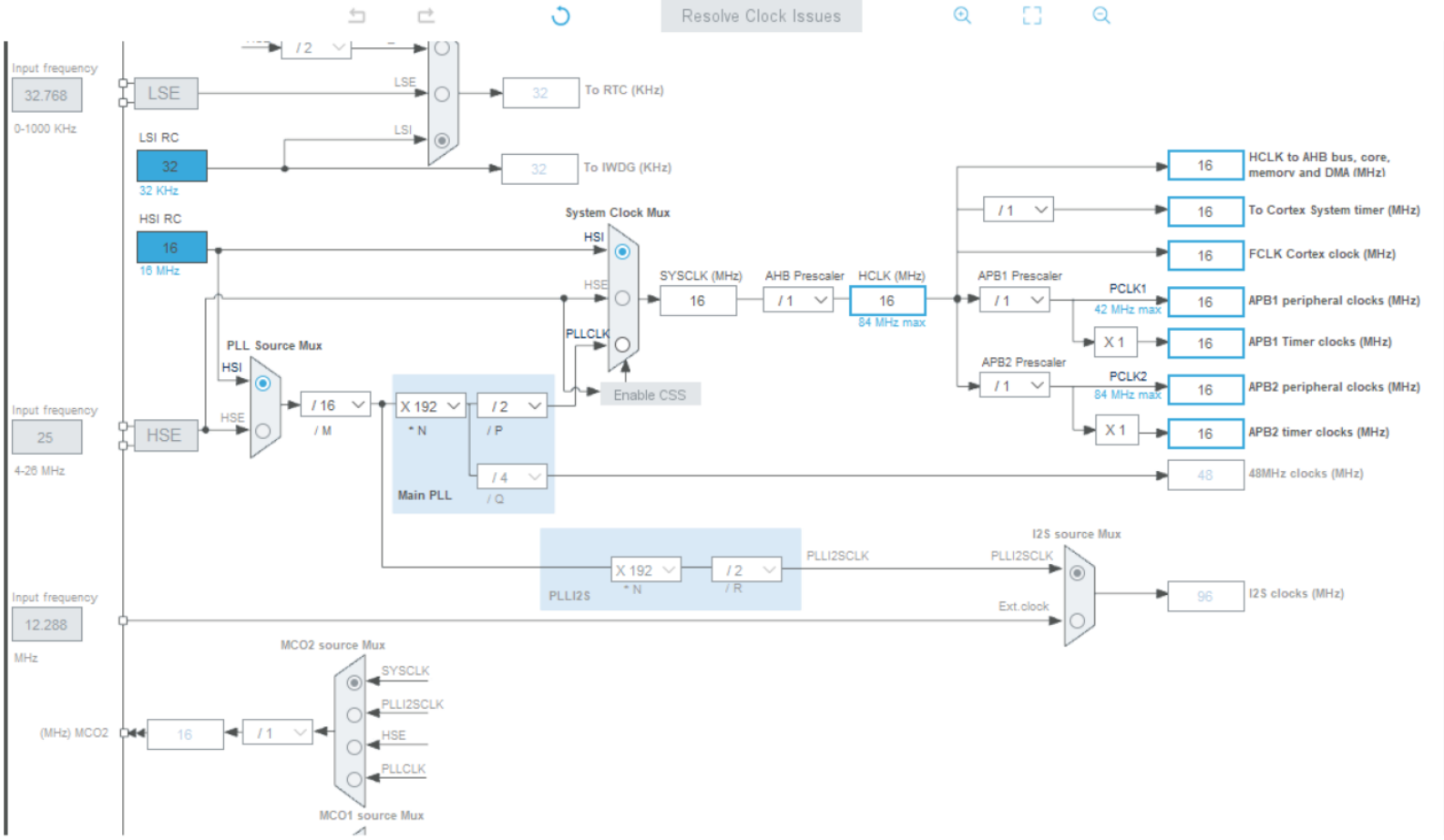


در ادامه با توجه به پروژه قبلی پین های مورد نظر برای gpio و interrupt و timer را تنظیم میکنیم که هرکدام به چه صورت باید باشد ، به عنوان مثال : pull up یا pull down یا غیره .

حال عکسی از هر بخش و پین های مورد نظر را در ادامه قرار میدهیم :



## : clock configuration عكس



در ادامه عکسهای از configuration ها داریم :

## 1) Gpio configuration :

The screenshot shows the 'Pinout & Configuration' window. On the left, the 'System Core' category is expanded, showing 'GPIO' selected. The main area is titled 'GPIO Mode and Configuration'. It has a 'Group By Peripherals' dropdown and tabs for 'GPIO' and 'NVIC'. Below the tabs is a 'Search Signals' field. A table lists various pins with their configurations.

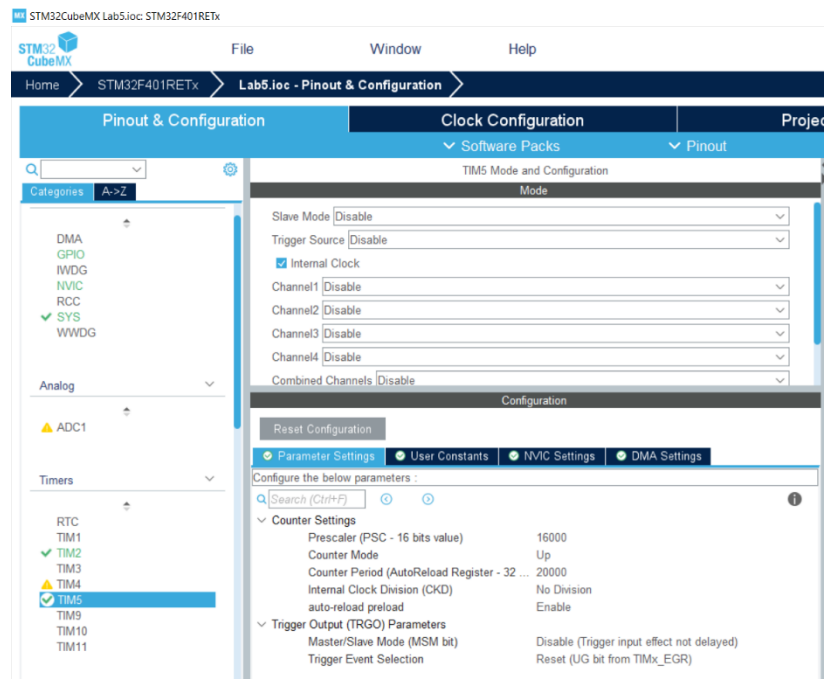
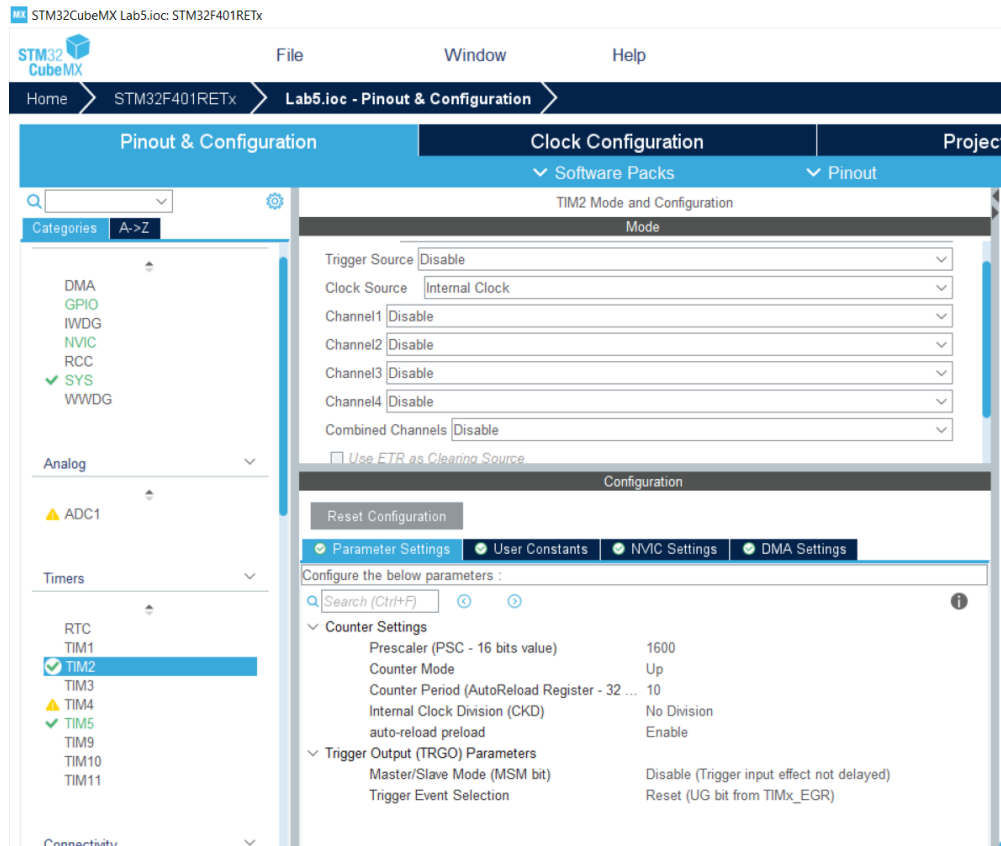
Pin Name	Signal on Pin	GPIO output	GPIO mode	GPIO Pull-up	Maximum	User Label	Modified
PB5	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PB6	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PB7	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PB8	n/a	n/a	External In...	Pull-up	n/a		<input checked="" type="checkbox"/>
PB9	n/a	n/a	External In...	Pull-up	n/a		<input checked="" type="checkbox"/>
PB10	n/a	n/a	External In...	Pull-up	n/a		<input checked="" type="checkbox"/>
PC0	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PC1	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PC2	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>
PC3	n/a	Low	Output Pu...	No pull-up ...	Low		<input type="checkbox"/>

## 2) Nested vector configuration :

The screenshot shows the 'NVIC Mode and Configuration' window. On the left, the 'System Core' category is expanded, showing 'NVIC' selected. The main area is titled 'Configuration'. It has tabs for 'NVIC' and 'Code generation'. Below the tabs are settings for 'Priority Group', 'Sort by Preemption Priority and Sub Priority', 'Sort by interrupts names', 'Search', 'Show', and 'Force DMA channels interrupts'. A table lists various interrupts with their configurations.

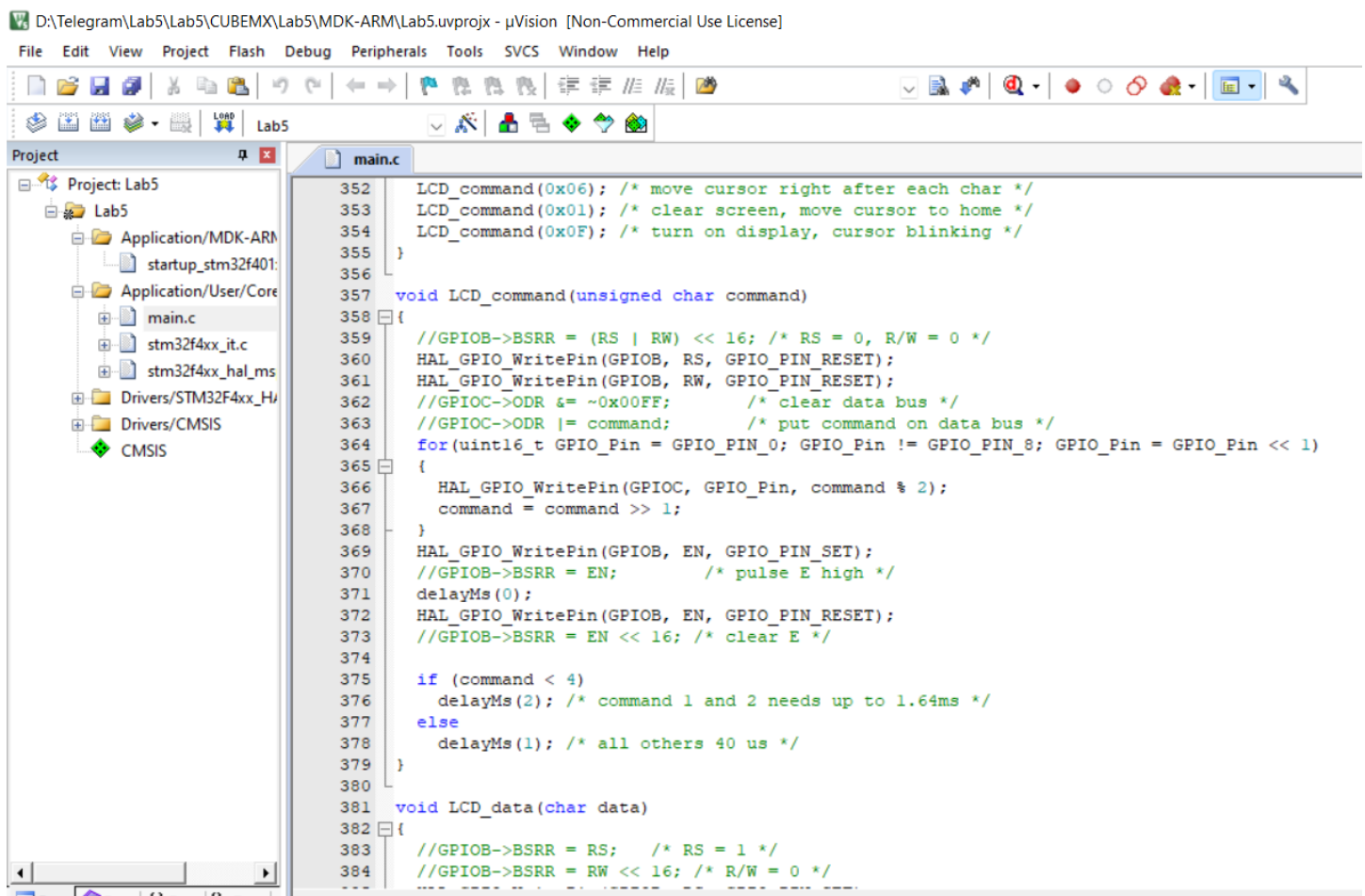
NVIC Interrupt Table	Enabled	Preemption Priority	Sub Priority
Non maskable interrupt	<input checked="" type="checkbox"/>	0	0
Hard fault interrupt	<input checked="" type="checkbox"/>	0	0
Memory management fault	<input checked="" type="checkbox"/>	0	0
Pre-fetch fault, memory access fault	<input checked="" type="checkbox"/>	0	0
Undefined instruction or illegal state	<input checked="" type="checkbox"/>	0	0
System service call via SWI instruction	<input checked="" type="checkbox"/>	0	0
Debug monitor	<input checked="" type="checkbox"/>	0	0
Pendable request for system service	<input checked="" type="checkbox"/>	0	0
Time base: System tick timer	<input checked="" type="checkbox"/>	0	0
PVD interrupt through EXTI line 16	<input type="checkbox"/>	0	0
Flash global interrupt	<input type="checkbox"/>	0	0
RCC global interrupt	<input type="checkbox"/>	0	0
EXTI line[9:5] interrupts	<input checked="" type="checkbox"/>	0	0
TIM2 global interrupt	<input checked="" type="checkbox"/>	0	0
EXTI line[15:10] interrupts	<input checked="" type="checkbox"/>	0	0
TIM5 global interrupt	<input type="checkbox"/>	0	0
FPU global interrupt	<input type="checkbox"/>	0	0

## : Timer 2 and Timer 5 configuration (3



در ادامه برای کد keil کلیت کد همانند دستور کار قبلی بوده و فقط مواردی تفاوت وجود دارد که به علت تغییر library از cmsis به HAL میباشد که تعدادی را در ادامه ی گزارش آورده ایم :

(1) اولین مورد تبدیل توابع cmsis به HAL میباشد که cmsis ان را کامنت گذاری کرده ایم که یک نمونه به عنوان مثال آورده ایم :



```
352 LCD_command(0x06); /* move cursor right after each char */
353 LCD_command(0x01); /* clear screen, move cursor to home */
354 LCD_command(0x0F); /* turn on display, cursor blinking */
355 }
356
357 void LCD_command(unsigned char command)
358 {
359     //GPIOB->BSRR = (RS | RW) << 16; /* RS = 0, R/W = 0 */
360     HAL_GPIO_WritePin(GPIOB, RS, GPIO_PIN_RESET);
361     HAL_GPIO_WritePin(GPIOB, RW, GPIO_PIN_RESET);
362     //GPIOC->ODR &= ~0x00FF; /* clear data bus */
363     //GPIOC->ODR |= command; /* put command on data bus */
364     for(uint16_t GPIO_Pin = GPIO_PIN_0; GPIO_Pin != GPIO_PIN_8; GPIO_Pin = GPIO_Pin << 1)
365     {
366         HAL_GPIO_WritePin(GPIOC, GPIO_Pin, command % 2);
367         command = command >> 1;
368     }
369     HAL_GPIO_WritePin(GPIOB, EN, GPIO_PIN_SET);
370     //GPIOB->BSRR = EN; /* pulse E high */
371     delayMs(0);
372     HAL_GPIO_WritePin(GPIOB, EN, GPIO_PIN_RESET);
373     //GPIOB->BSRR = EN << 16; /* clear E */
374
375     if (command < 4)
376         delayMs(2); /* command 1 and 2 needs up to 1.64ms */
377     else
378         delayMs(1); /* all others 40 us */
379 }
380
381 void LCD_data(char data)
382 {
383     //GPIOB->BSRR = RS; /* RS = 1 */
384     //GPIOB->BSRR = RW << 16; /* R/W = 0 */
```

(2) یک تغییر دیگر نحوه صدا زدن تابع وقفه میباشد که syntax آن تغییر کرده و به صورت ذیل است :

`void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)`

3) و ما سه تابع دیگر با نام های :

`void SystemClock_Config(void)`

`static void MX_TIM2_Init(void)`

`static void MX_TIM5_Init(void)`

`static void MX_GPIO_Init(void)`

داریم که خود `cubemx` انها را برای `initialize` کردن اولیه تعریف میکند.