TI DSP, MCU, Xilinx Zynq FPGA 프로그래밍 전문가 과정

GPIO based on STM32F407

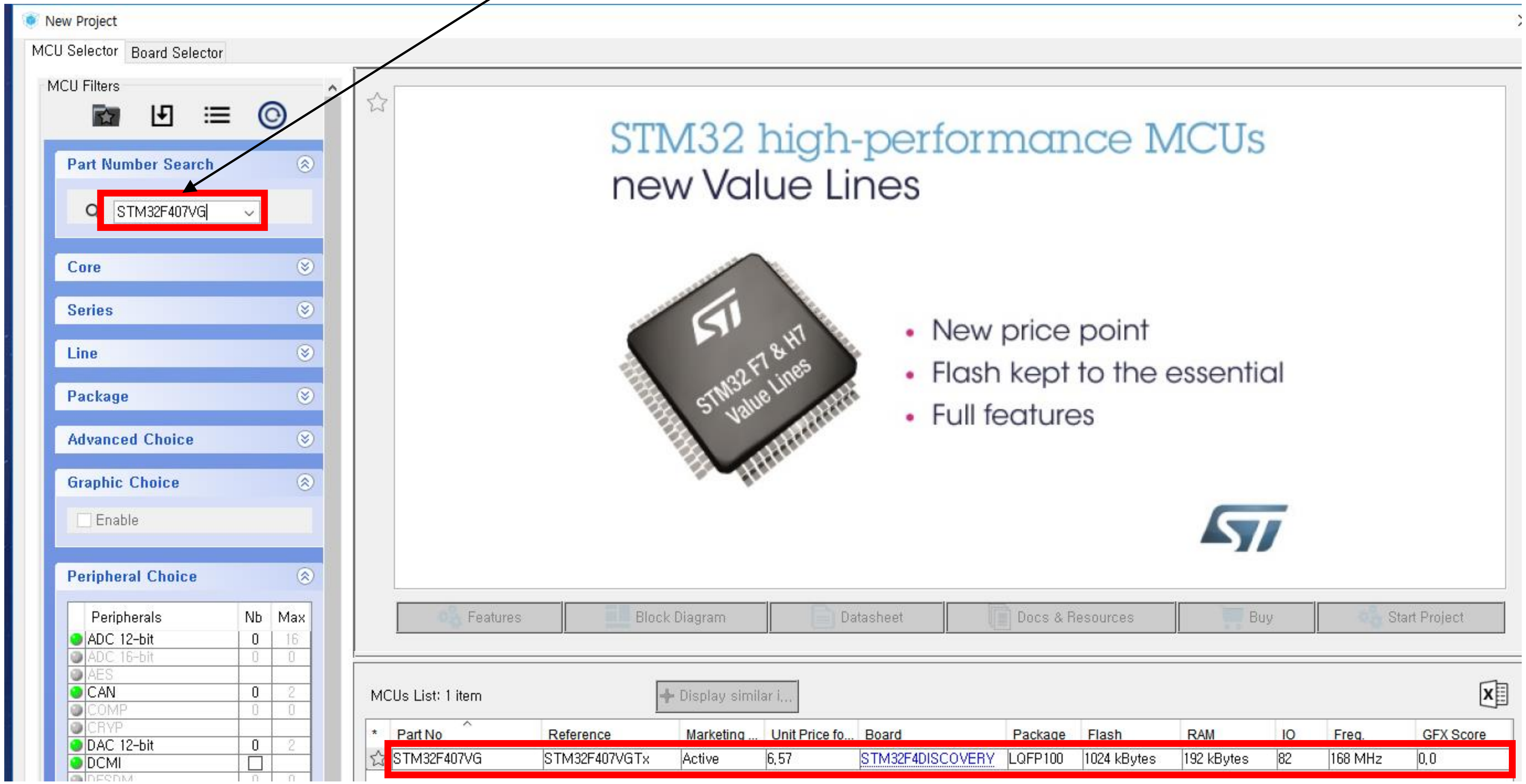
강사 – Innova Lee(이상훈)
gcccompil3r@gmail.com

학생 – 안상재
sangjae2015@naver.com

1. LED 깜빡이기 (GPIO OUTPUT)

- CubeMX 설정

필자는 **STM32F407VGT6** 이라는 MCU가 탑재됨 보드(STM32F4-DISC1)를 사용함



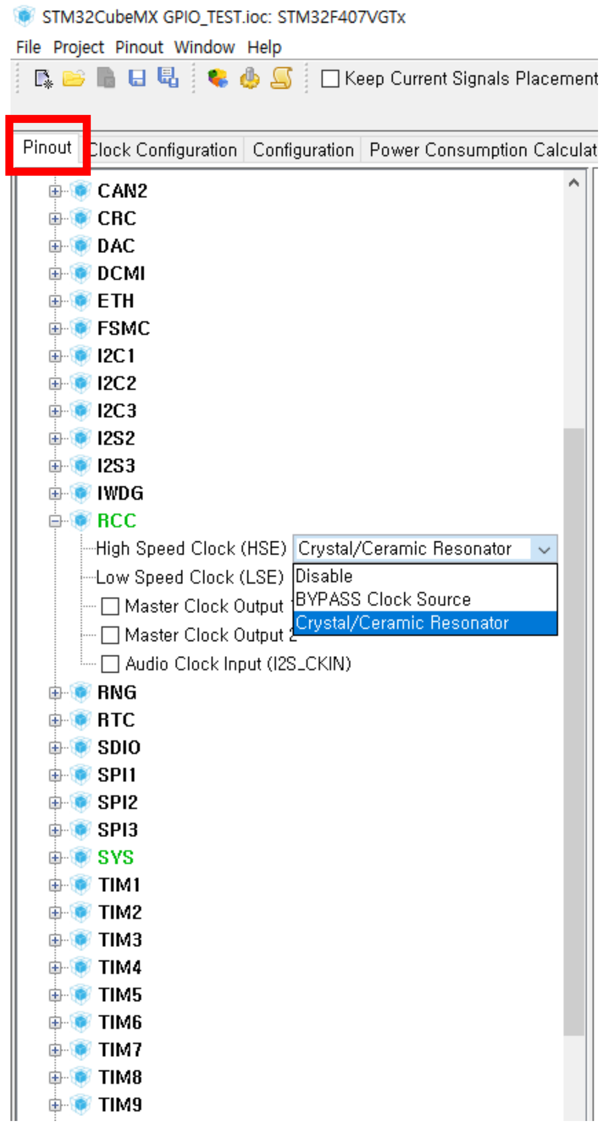
STM32F407VG

	Peripherals	Nb	Max
●	ADC 12-bit	0	16
●	ADC 16-bit	0	0
●	AES		
●	CAN	0	2
●	COMP	0	0
●	CRYP		
●	DAC 12-bit	0	2
●	DCMI	□	
●	DFSDM	0	0

MCUs List: 1 item

*	Part No	Reference	Marketing ...	Unit Price fo...	Board	Package	Flash	RAM	IO	Freq.	GFX Score
☆	STM32F407VG	STM32F407VGTx	Active	6,57	STM32F4DISCOVERY	LQFP100	1024 kBytes	192 kBytes	82	168 MHz	0,0

더블클릭으로 프로젝트 생성



STM32CubeMX GPIO_TEST.ioc: STM32F407VGTx

File Project Pinout Window Help

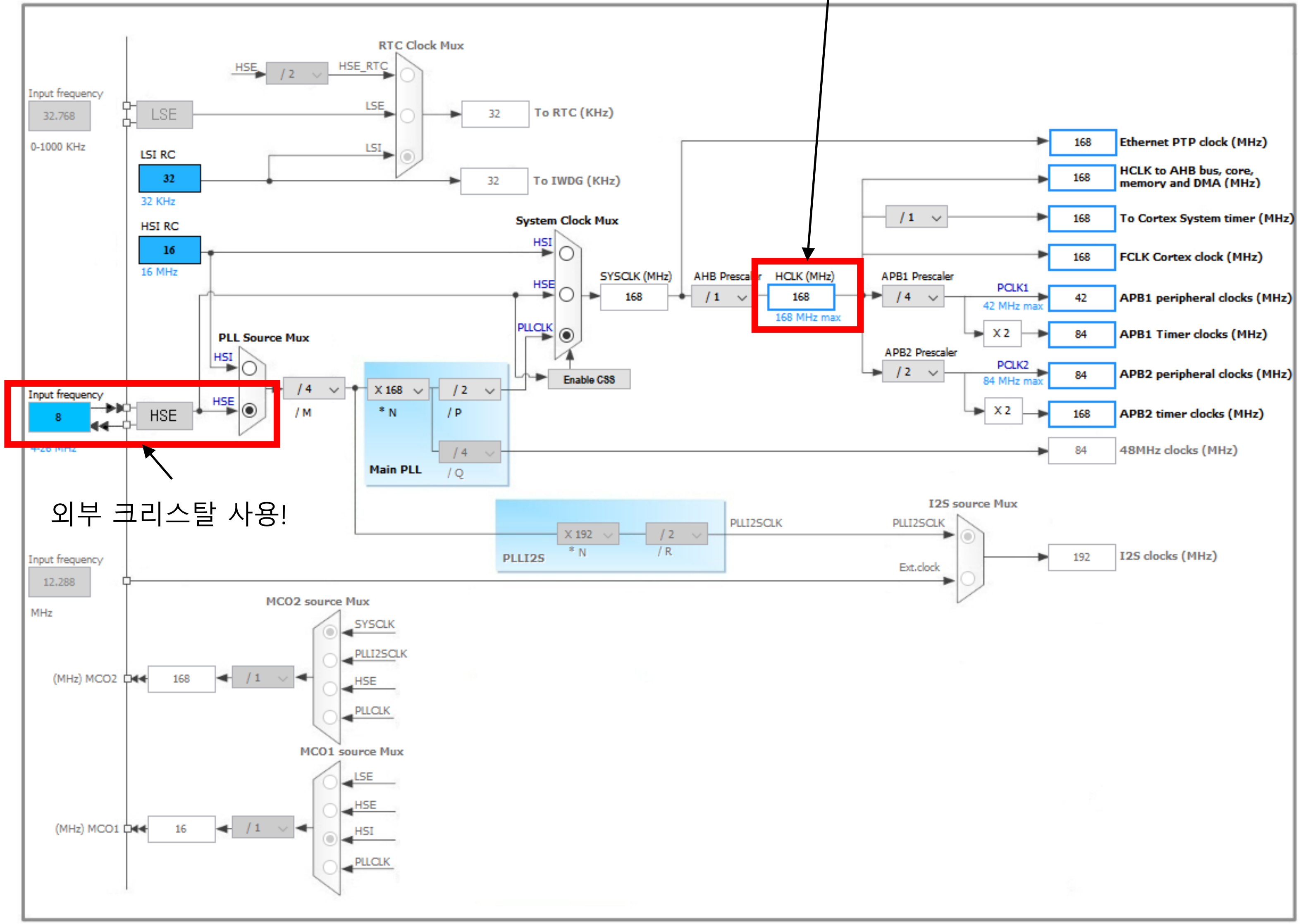
□ Keep Current Signals Placement

Pinout | Clock Configuration | Configuration | Power Consumption Calculat

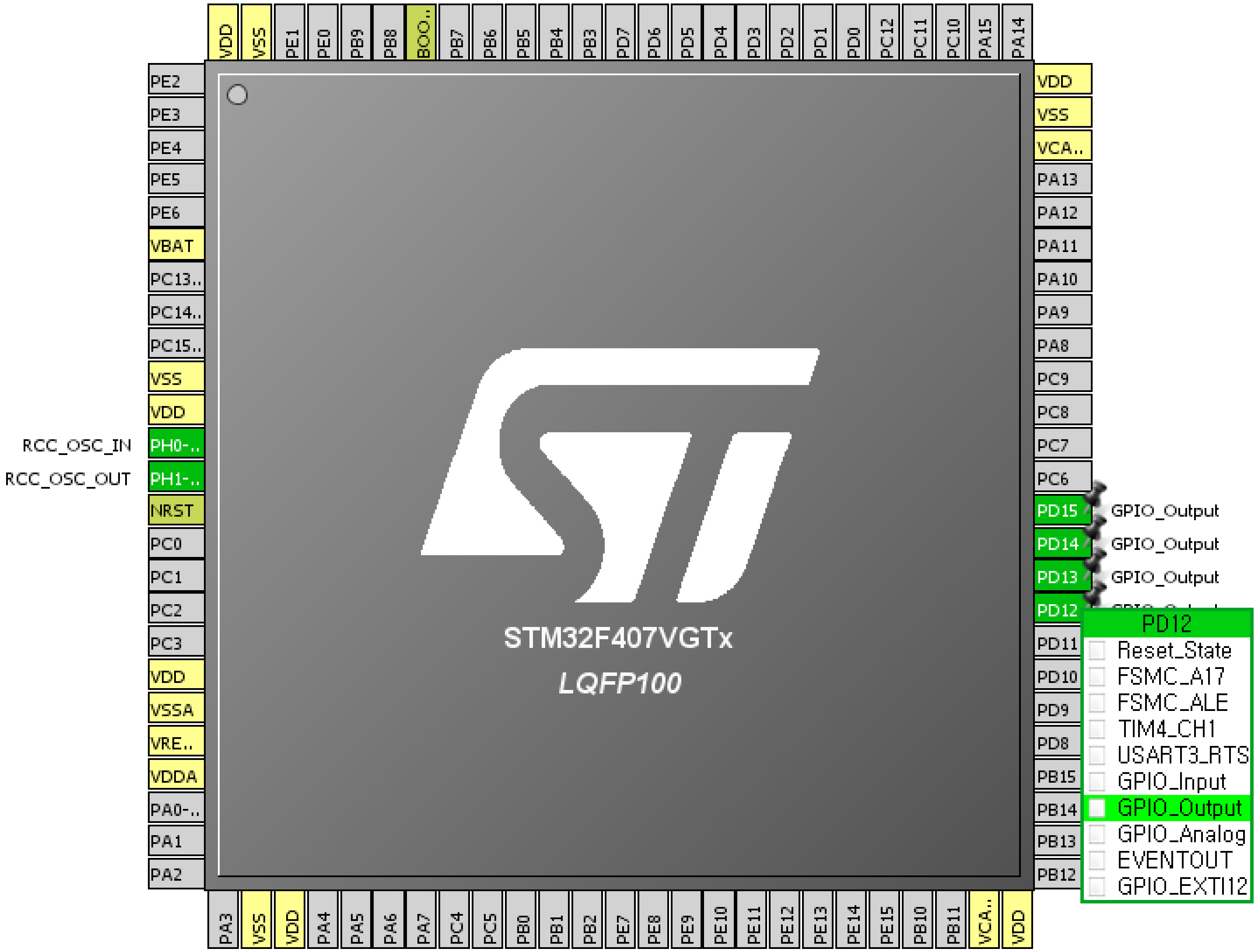
- CAN2
- CRC
- DAC
- DCMI
- ETH
- FSMC
- I2C1
- I2C2
- I2C3
- I2S2
- I2S3
- IWDG
- RCC
  - High Speed Clock (HSE): Crystal/Ceramic Resonator
  - Low Speed Clock (LSE): Disable / BYPASS Clock Source / Crystal/Ceramic Resonator
  - □ Master Clock Output 1
  - □ Master Clock Output 2
  - □ Audio Clock Input (I2S_CKIN)
- RNG
- RTC
- SDIO
- SPI1
- SPI2
- SPI3
- SYS
- TIM1
- TIM2
- TIM3
- TIM4
- TIM5
- TIM6
- TIM7
- TIM8
- TIM9

Clock Configuration 탭

MCU가 허용하는 최대 주파수로 맞춰줌!



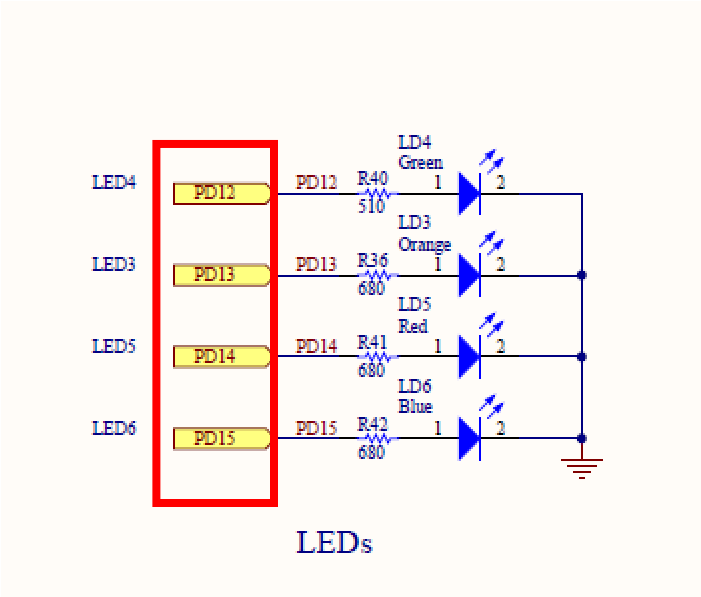
외부 크리스탈 사용!



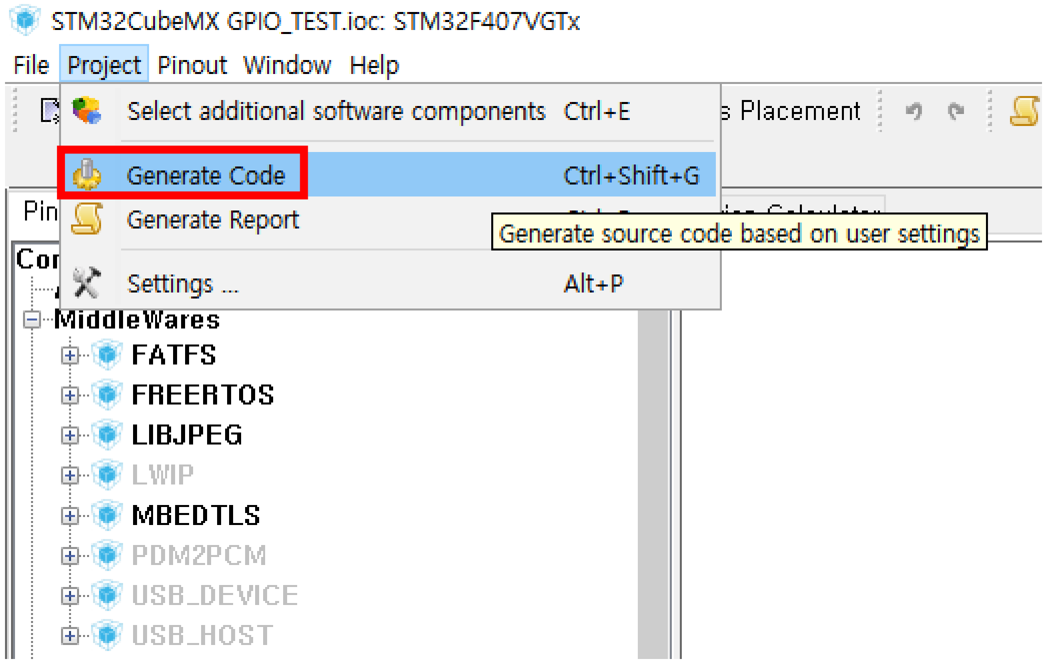
PD15 GPIO_Output
PD14 GPIO_Output
PD13 GPIO_Output
PD12 GPIO_Output

PD12
- □ Reset_State
- □ FSMC_A17
- □ FSMC_ALE
- □ TIM4_CH1
- □ USART3_RTS
- □ GPIO_Input
- □ GPIO_Output
- □ GPIO_Analog
- □ EVENTOUT
- □ GPIO_EXTI12

CubeMX Pinout



STM32F407-DISC1 보드 회로도



STM32CubeMX GPIO_TEST.ioc: STM32F407VGTx
File
Project
Pinout
Window
Help
Select additional software components Ctrl+E
Generate Code Ctrl+Shift+G
Generate Report
Settings ... Alt+P
Generate source code based on user settings
Placement
MiddleWares
FATFS
FREERTOS
LIBJPEG
LWIP
MBEDTLS
PDM2PCM
USB_DEVICE
USB_HOST

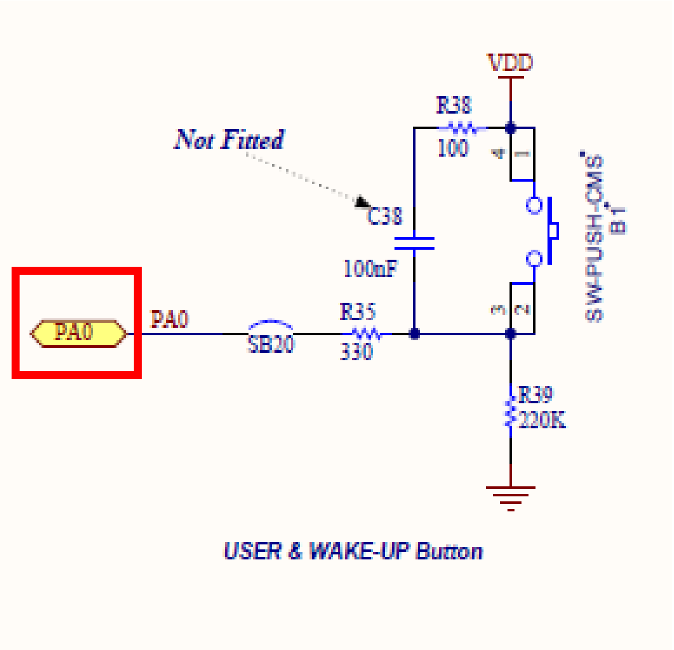
Generate Code 클릭 후, 프로젝트 생성

- 소스 코드

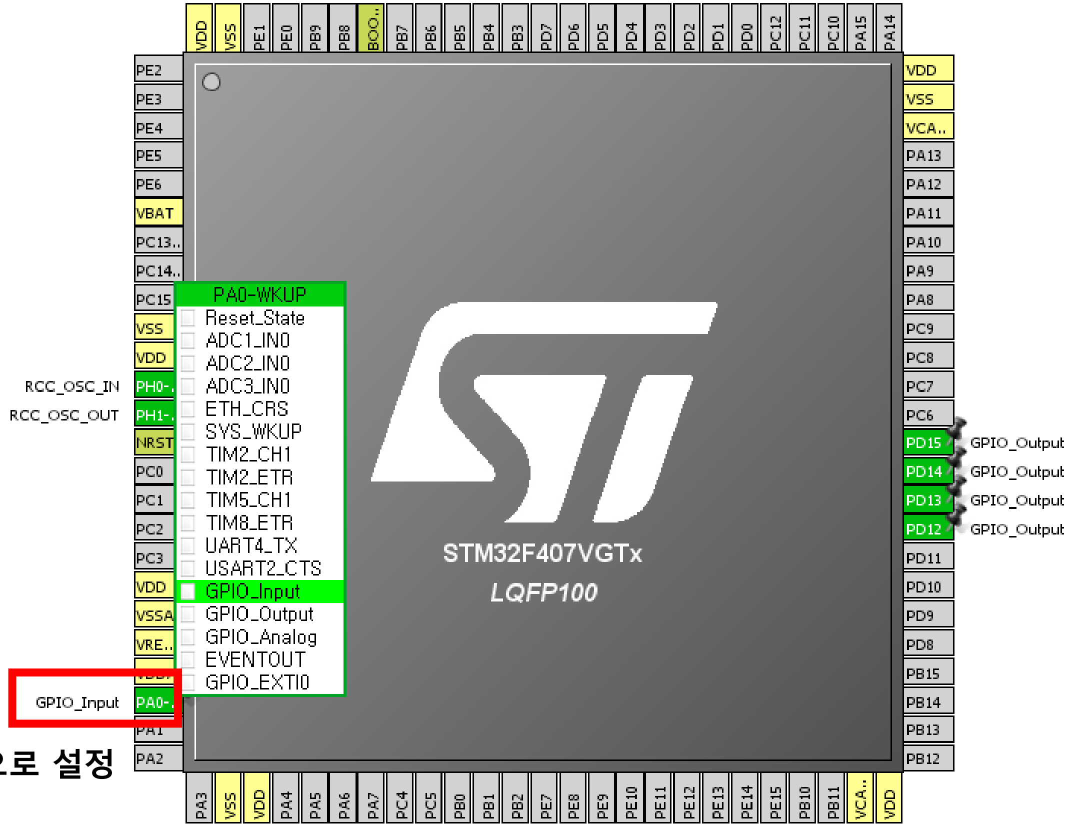
```c
while (1)
  {
  /* USER CODE END WHILE */

  /* USER CODE BEGIN 3 */
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, GPIO_PIN_SET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_15, GPIO_PIN_SET);

  HAL_Delay(1000);// milisecond 단위

  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, GPIO_PIN_RESET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, GPIO_PIN_RESET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, GPIO_PIN_RESET);
  HAL_GPIO_WritePin(GPIOD, GPIO_PIN_15, GPIO_PIN_RESET);

  HAL_Delay(1000);// milisecond 단위
  /* USER CODE BEGIN 3 */
  }
```

```c
void HAL_GPIO_WritePin(GPIO_TypeDef* GPIOx, uint16_t GPIO_Pin,
GPIO_PinState PinState)
{
  /* Check the parameters */
  assert_param(IS_GPIO_PIN(GPIO_Pin));
  assert_param(IS_GPIO_PIN_ACTION(PinState));

  if(PinState != GPIO_PIN_RESET)
  {
    GPIOx->BSRR = GPIO_Pin;
  }
  else
  {
    GPIOx->BSRR = (uint32_t)GPIO_Pin << 16U;
  }
}
```

2. 스위치 입력 (GPIO INPUT)



스위치 주변 회로

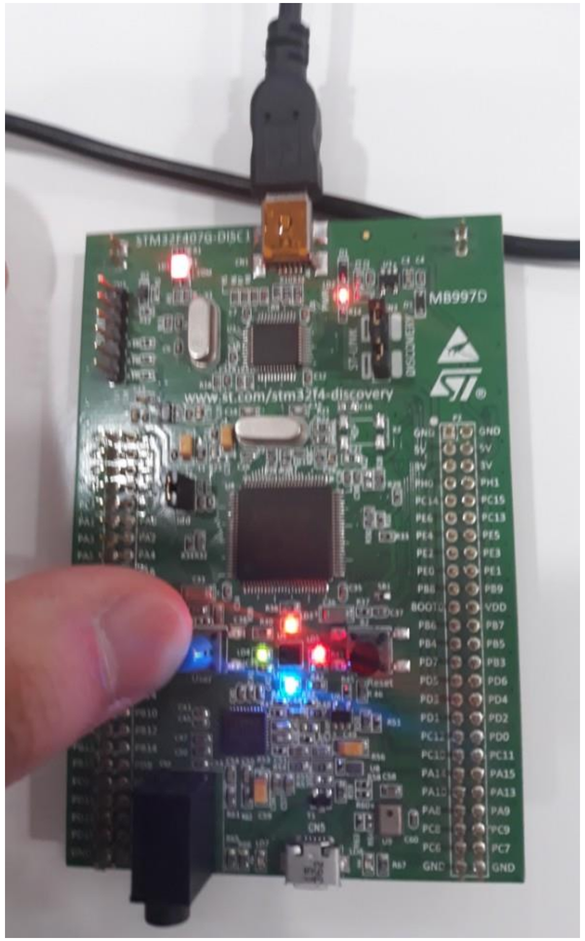


GPIOA0 번 핀을 Input으로 설정

CubeMX Pinout

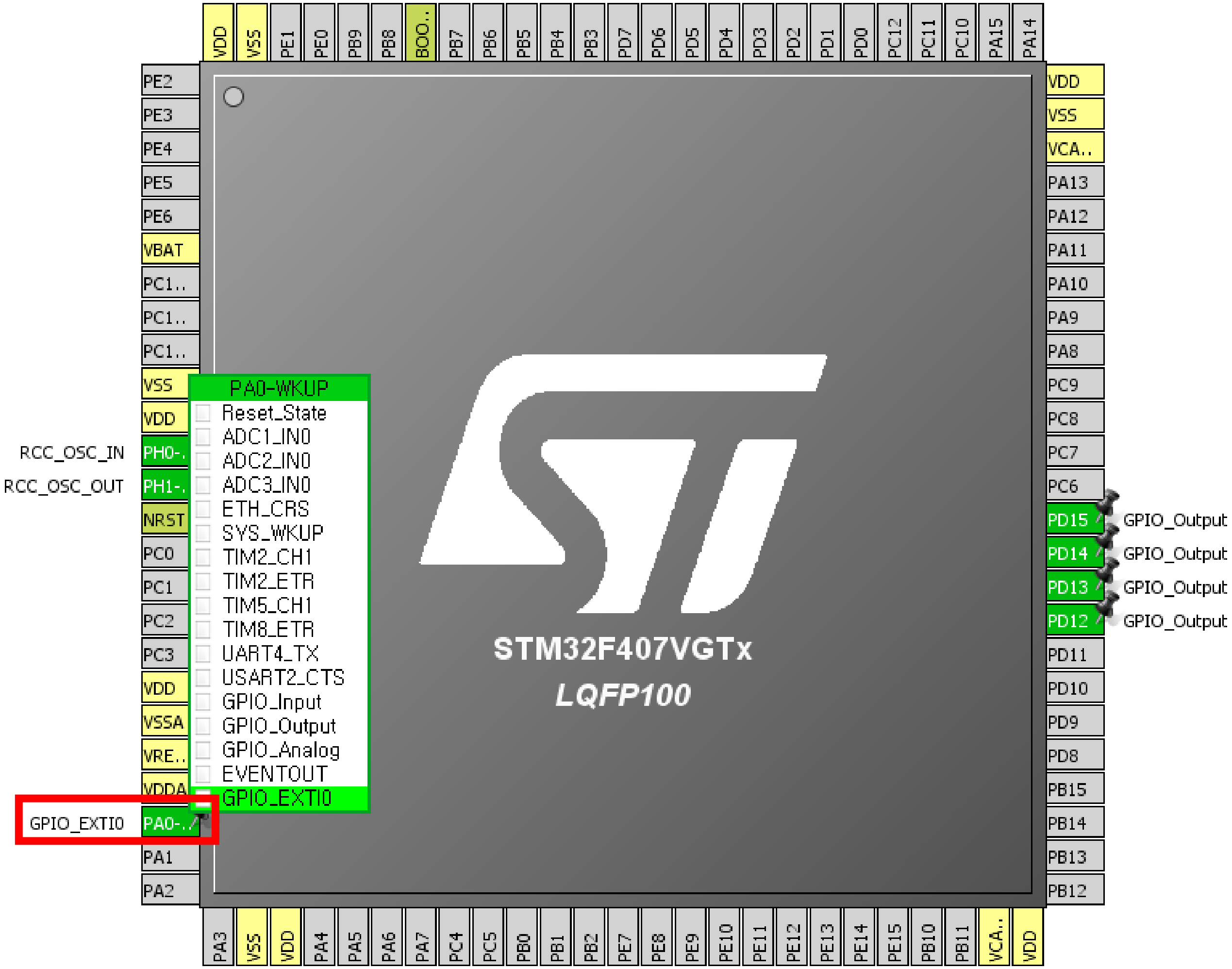
```
while (1)
 {
 /* USER CODE END WHILE */

 /* USER CODE BEGIN 3 */
      if(HAL_GPIO_ReadPin(GPIOA, GPIO_PIN_0) == 1)
      {
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, SET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, SET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, SET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_15, SET);
      }
      else
      {
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_12, RESET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_13, RESET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_14, RESET);
            HAL_GPIO_WritePin(GPIOD, GPIO_PIN_15, RESET);
      }
 }
```

스위치 입력 확인!

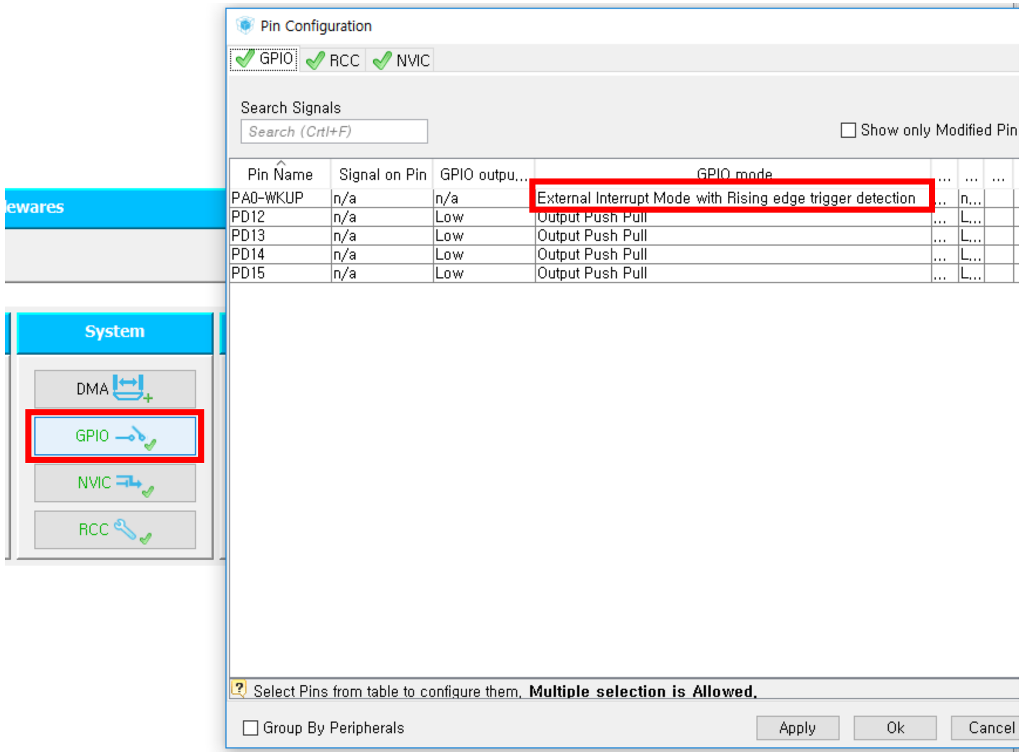


스위치 눌렀을 때 LED가 켜짐

3. 외부 인터럽트 (External Interrupt)



인터럽트가 설정된 것을 확인!



Pin Configuration

✓ GPIO ✓ RCC ✓ NVIC

Search Signals

Search (Crtl+F)

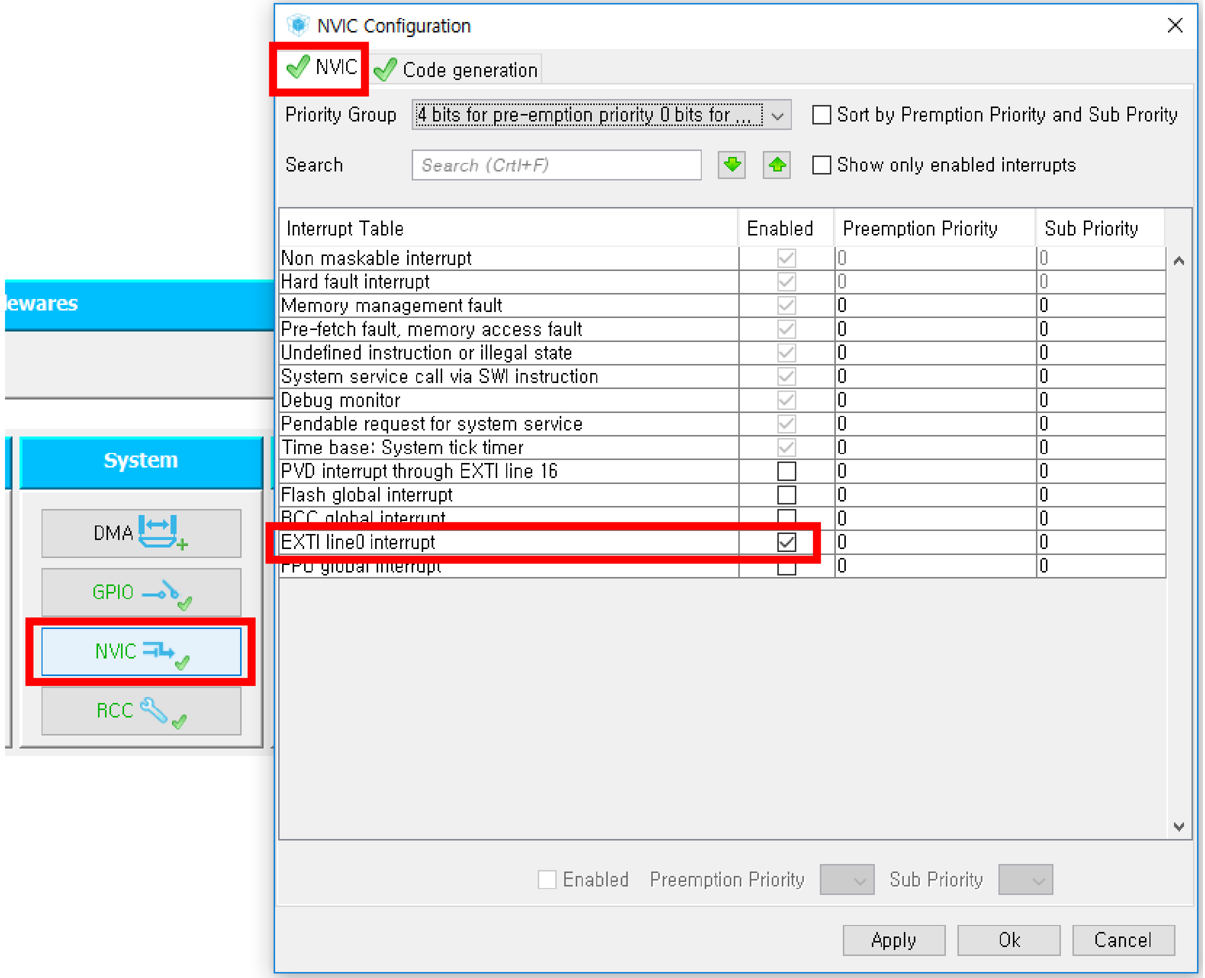
☐ Show only Modified Pin

Pin Name	Signal on Pin	GPIO outpu...	GPIO mode	...	...	...
PA0-WKUP	n/a	n/a	External Interrupt Mode with Rising edge trigger detection	...	n...	
PD12	n/a	Low	Output Push Pull	...	L...	
PD13	n/a	Low	Output Push Pull	...	L...	
PD14	n/a	Low	Output Push Pull	...	L...	
PD15	n/a	Low	Output Push Pull	...	L...	

Select Pins from table to configure them. **Multiple selection is Allowed.**

☐ Group By Peripherals

Apply Ok Cancel



Priority Group: 4 bits for pre-emption priority 0 bits for ...

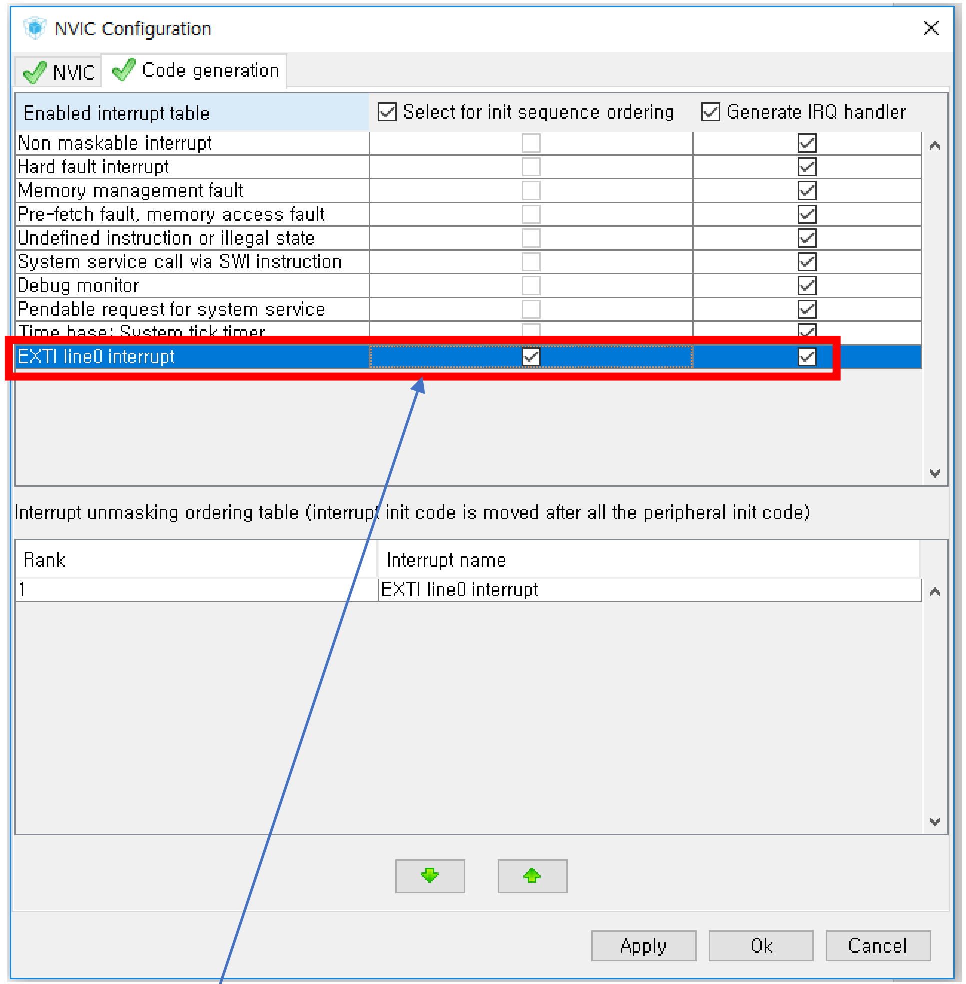
☐ Sort by Premption Priority and Sub Prority

Search: Search (Crtl+F)

☐ Show only enabled interrupts

Interrupt Table	Enabled	Preemption Priority	Sub Priority
Non maskable interrupt	☑	0	0
Hard fault interrupt	☑	0	0
Memory management fault	☑	0	0
Pre-fetch fault, memory access fault	☑	0	0
Undefined instruction or illegal state	☑	0	0
System service call via SWI instruction	☑	0	0
Debug monitor	☑	0	0
Pendable request for system service	☑	0	0
Time base: System tick timer	☑	0	0
PVD interrupt through EXTI line 16	☐	0	0
Flash global interrupt	☐	0	0
RCC global interrupt	☐	0	0
EXTI line0 interrupt	☑	0	0
FPU global interrupt	☐	0	0

☐ Enabled Preemption Priority Sub Priority

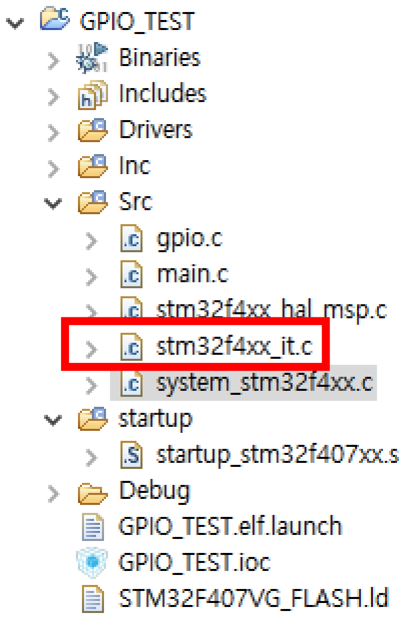


Enabled interrupt table	☑ Select for init sequence ordering	☑ Generate IRQ handler
Non maskable interrupt	☐	☑
Hard fault interrupt	☐	☑
Memory management fault	☐	☑
Pre-fetch fault, memory access fault	☐	☑
Undefined instruction or illegal state	☐	☑
System service call via SWI instruction	☐	☑
Debug monitor	☐	☑
Pendable request for system service	☐	☑
Time base: System tick timer	☐	☑
EXTI line0 interrupt	☑	☑

Interrupt unmasking ordering table (interrupt init code is moved after all the peripheral init code)

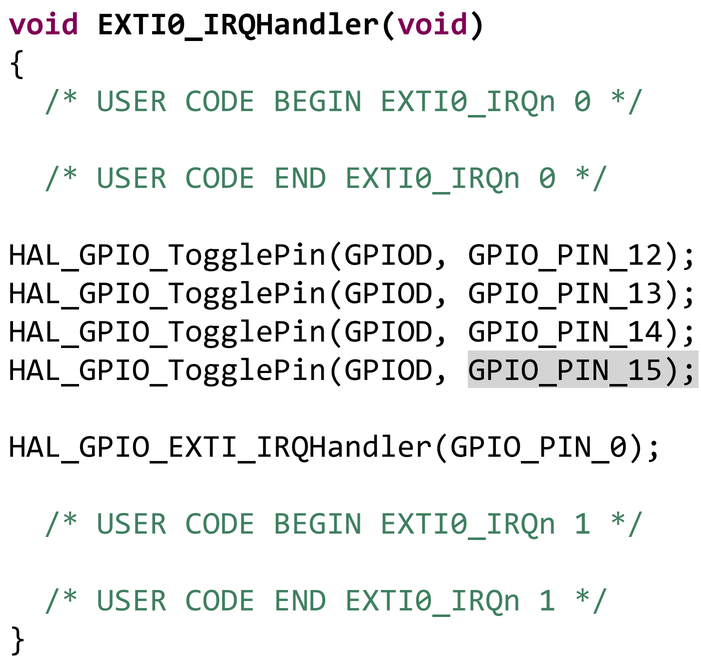
Rank	Interrupt name
1	EXTI line0 interrupt

EXTI0_IRQHandler() 생성



- GPIO_TEST
  - Binaries
  - Includes
  - Drivers
  - Inc
  - Src
    - gpio.c
    - main.c
    - stm32f4xx_hal_msp.c
    - stm32f4xx_it.c
    - system_stm32f4xx.c
  - startup
    - startup_stm32f407xx.s
  - Debug
  - GPIO_TEST.elf.launch
  - GPIO_TEST.ioc
  - STM32F407VG_FLASH.ld

stm32f4xx_it.c 파일에 EXTI0에 대한 인터럽트 서비스 루틴이 생성됨!



```c
void EXTI0_IRQHandler(void)
{
  /* USER CODE BEGIN EXTI0_IRQn 0 */

  /* USER CODE END EXTI0_IRQn 0 */

HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_12);
HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_13);
HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_14);
HAL_GPIO_TogglePin(GPIOD, GPIO_PIN_15);

HAL_GPIO_EXTI_IRQHandler(GPIO_PIN_0);

  /* USER CODE BEGIN EXTI0_IRQn 1 */

  /* USER CODE END EXTI0_IRQn 1 */
}
```