TUTORIAL: Digital Out

Name: 김은찬 ID: 21801017

# I. Introduction

In this lab, we will learn how to control the digital Output of GPIOs of the MCU board to turn on/off an LED.

The objectives of this lab are to learn how to

* Read and configure registers of digital GPIO of MCU
* Program firmware to control digital input/output pins

### Hardware

NUCLEO -F411RE

### Software

Keil uVision IDE, CMSIS, EC\_HAL

### Documentation

[STM32 Reference Manual](https://ykkim.gitbook.io/ec/stm32-m4-programming/hardware/nucleo-f411re#manual-documentation)

# II. Basics of GPIO Out

## A. Bit Operation

Fill in the blanks. You should write the answer in both hexa-decimal and binary number.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Bit operation | Description | Answer |
|  | 0xA & 0x2 | AND | b1010 & b0010 = b0010 = 0x2 |
|  | 0x3 | 0xC | OR | bx0011 | bx1100 = bx1111 = 0xF |
|  | ~0x7 | NOT | ~b0111 = b1000 = 0x8 |
|  | 0xF ^ 0x5 | XOR(Toggle) | b1111 ^ b0101 = b1010 = 0xA |
|  | 0x1 << 3 | Shift left | b0001 << 3 = b1000 = 0x8 |
|  | 0xC >> 2 | Shift right | b1100 >> 2 = b0011 = 0x3 |
|  | 0x11 |= 1<<3 | Set bit | b1111 = 0xF |
|  | 0xFF &= ~1<<4 | Clear bit | b1110 1111 = 0xEF |
|  | val=0x0F &1<<3 | Read bit | b0000 1111 & b1000 = b0000 1000 = 0x8 |

## B. GPIO Digital Out Register

List GPIO registers for this LAB

|  |  |  |
| --- | --- | --- |
| Type | Register Name | Description |
| GPIO | GPIOx\_MODER | Mode: Output/Input/Analog |
|  | GPIOx\_OTYPER | Output Type: Opendrain/Push-Pull |
|  | GPIOx\_OSPEEDR | Output Speed: |
|  | GPIOx\_PUPDR | Pull-Up Pull-Down: |
|  | GPIOx\_ODR | Output Data Register |

Schematic

텍스트, 도표, 스크린샷, 평면도이(가) 표시된 사진

자동 생성된 설명

Process of GPIOx register initiation

|  |
| --- |
| 0. Enable Peripheral Clock (**AHB1ENR**)  1. Configure as Digital Output (**MODER**)  2. Configure pull-up/down resistors (**PUPDR**)  3. For Output: Configure Output Type (**OTYPE**)  4. For Output: Configure Output Speed (**OSPEEDR**)  5. Output Data **(ODR)** |

22

# III. Tutorial

## A. Register Configuration

**1. GPIO: Digital Out - Pin Initialization & Set LED**

Port A Pin 5 / Output / Push-Pull / No Pull-Up & No Pull-Down / Default Speed

use **#define LED\_PIN 5**

* **MODER:** Output (MODER5[1:0]=01)

|  |
| --- |
| 텍스트, 스크린샷, 폰트, 번호이(가) 표시된 사진  자동 생성된 설명 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Mask** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **Value** | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 | 1 | x | x | x | x | x | x | x | x | x | x |

GPIOA->MODER &= ~(3<<( LED\_PIN \*2)); // clear bits at both [10] and [11]

GPIOA->MODER |= 1<<(LED\_PIN \*2); // set bit at [10]

* **OTYPER:** Push-Pull (OT5=0)

|  |
| --- |
| 텍스트, 폰트, 스크린샷, 라인이(가) 표시된 사진  자동 생성된 설명 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **1** | **0** | **1** | **1** | **1** | **1** | **1** |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **x** | **0** | **x** | **x** | **x** | **x** | **x** |

GPIOA->OTYPER &= ~(1<< LED\_PIN)

* **PUPDR:** no pull-up no pull-down (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

|  |
| --- |
| *Register map from reference manual* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

GPIOA->PUPDR &= ~(3<<( LED\_PIN \*2));

GPIOA->PUPDR \_\_\_\_\_= \_\_\_\_\_

* **ODR:** Set LED (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

|  |
| --- |
| *Register map from reference manual* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Register** | **31** | **30** | **29** | **28** | **27** | **26** | **25** | **24** | **23** | **22** | **21** | **20** | **19** | **18** | **17** | **16** | **15** | **14** | **13** | **12** | **11** | **10** | **9** | **8** | **7** | **6** | **5** | **4** | **3** | **2** | **1** | **0** |
| **Mask** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

GPIOA->ODR \_\_= ;

## B. Programming

**Preparation**

* + Open the program ‘Keil uVision5’ and create a new project.
  + “ /repos/EC/Tutorial/TU\_GPIO\_Digital\_Out\_LED/”
  + Name the project as ‘**TU\_GPIO\_Digital\_Out\_LED**’.
  + Create a new item(file) and name it as ‘**TU\_GPIO\_Digital\_Out\_LED.c’**
  + Copy and paste from the source code ‘[**TU\_GPIO\_Digital\_Out\_LED\_student.c**](https://github.com/ykkimhgu/EC-student/tree/main/tutorial/tutorial-student)’.
  + Download [ecRCC\_student.h and ecRCC\_student.c](https://github.com/ykkimhgu/EC-student/tree/main/include/lib-student). Then, change the file names as **ecRCC.h** and **ecRCC.c**.
  + Save them in \lib folder: “repos/EC/include”
  + Include provided **ecRCC.h** and **ecRCC.c** library files in your project.

**Exercise**

This is an example code of turning ON LED *without* button input. Do not worry if you do not understand what this code means yet. You will learn one by one in the following few weeks.

Fill in the empty spaces in the code. Then, compile(F7) and flash(F8) the source code onto the MCU board. Verify the program by checking if the LED is turned ON.

텍스트이(가) 표시된 사진

자동 생성된 설명

## Appendix

[See here for MCU resources](https://ykkim.gitbook.io/ec/resource/nucleo-f411re)

1. Pin Configuration of NUCLE-F401RE

**텍스트, 스크린샷, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명텍스트, 번호, 평행, 폰트이(가) 표시된 사진

자동 생성된 설명**

1. LED/Button Circuit Diagram

텍스트, 도표, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

텍스트, 도표, 지도, 평면도이(가) 표시된 사진

자동 생성된 설명