**LAB: GPIO Digital I/O**

**- LED Toggle with Push-Button -**

**I. Overview**

In this lab, we will learn how to control digital I/O of GPIOs of the MCU board to turn on/off an LED with a push-button input. The LED should be turned on when the button is pressed, and vice versa.

The objectives of this lab are learning how to

* Read and configure registers of digital GPIO of MCU
* Program firmware to control digital input/output pins
* Create your own functions for GPIOs

**Preparation**:

* You need to read about the following registers: GPIO, ‘STM Reference Manual pg. 145-163’
* Install Keil uVision

**II. Pre-Lab**

**A. Bit Operation**

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

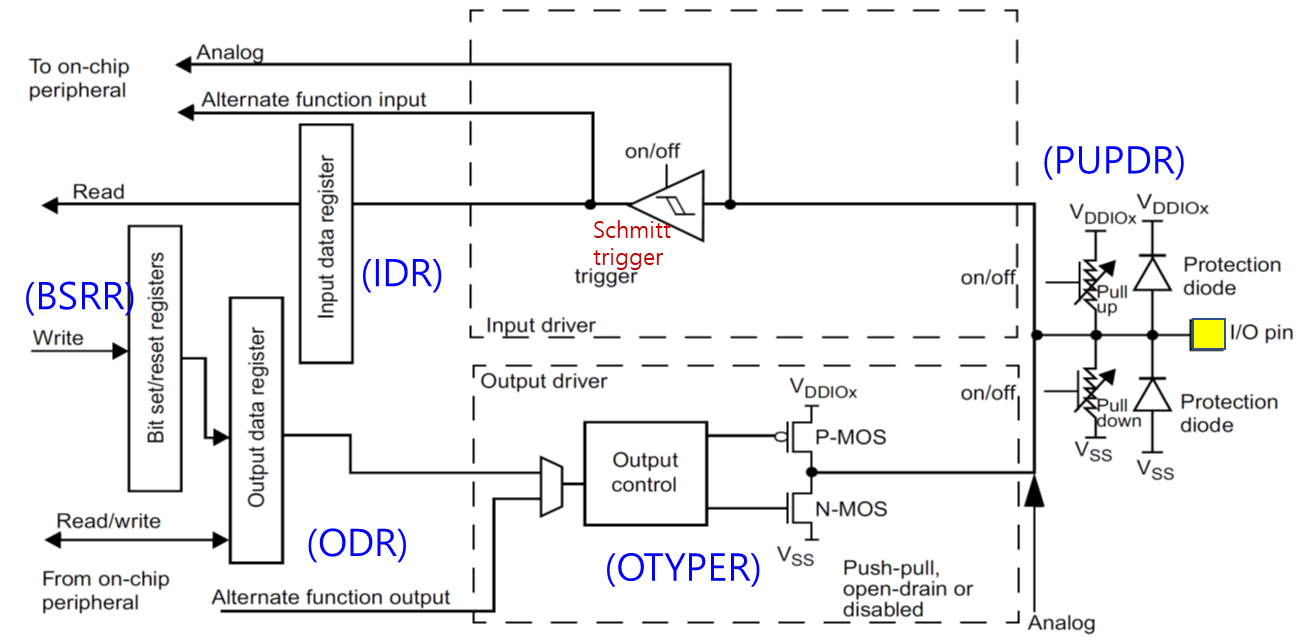
|  |  |  |  |
| --- | --- | --- | --- |
|  | Bit operation | Description | Answer |
|  | 0xA & 0x2 | AND |  |
|  | 0x3 | 0xC | OR |  |
|  | ~0x7 | NOT |  |
|  | 0xF ^ 0x5 | XOR(Toggle) |  |
|  | 0x1 << 3 | Shift left |  |
|  | 0xC >> 2 | Shift right |  |
|  | 0x11 |= 1<<3 | Set bit |  |
|  | 0xFF &= ~1<<4 | Clear bit |  |
|  | val=0x0F &1<<3 | Read bit |  |

**B. GPIO 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\_IDR | Input Data Register |
|  | GPIOx\_ODR | Output Data Register |

* Schematic



* Process of GPIO register initiation

|  |
| --- |
| 0. Enable Peripheral Clock (**AHB1ENR**)  1. Configure as Digital Output or Input (**GPIOx\_MODER**)  2. Configure pull-up/down resistors (**PUPDR**)  3. For Output: Configure Output Type (**OTYPE**)  4. For Output: Configure Output Speed (**OSPEEDR**)  5. Read Data or Output Data **(ORD / IDR)** |

**C. Register Setting**

**1. Pin Initialization & Set LED:** Port A Pin 5 / Output / Push-Pull / No Pull-Up& No Pull-Down

* **GPIOx\_MODER:** Output

|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 | 1 |  |  |  |  |  |  |  |  |  |  |

GPIOA\_MODER |= 3<<(5\*2)

* **GPIOx\_OTYPER:** Push-Pull

|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 |  |  |  |  |  |
| **Value** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 0 |  |  |  |  |  |

GPIOA\_OTYPER &= ~(1<<5)

* **GPIOx\_PUPDR:** no pull-up no pull-down

|  |
| --- |
| *Register map goes here* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 =

* **GPIOx\_ODR:** Set LED

|  |
| --- |
| *Register map goes here* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

*Code goes here*

**2. Pin Initialization & Read PushButton:** Port C Pin 13 / Input // Pull-Up

* **GPIOx\_MODER:** Input

|  |
| --- |
|  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **Value** |  |  |  |  | **1** | **1** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

GPIOC\_MODER |= 3<<(13\*2)

* **GPIOx\_PUPDR:** pull-up

|  |
| --- |
| *Register map goes here* |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

GPIOC\_PUPDR =

* **GPIOx\_IDR:** Read Push-Button Value

|  |
| --- |
| *Register map goes here* |

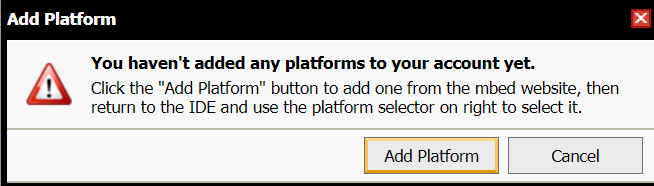
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

*Code goes here*

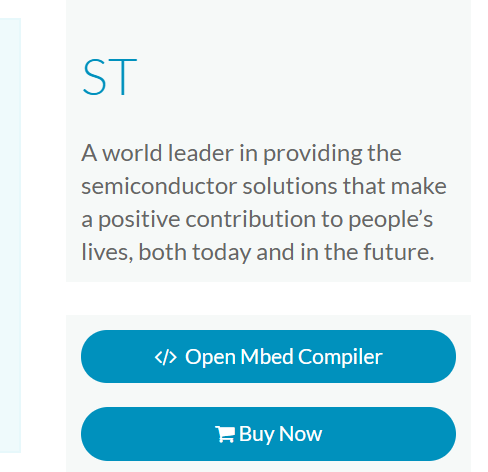
**III. Tutorial**

**A. Mbed online compiler**

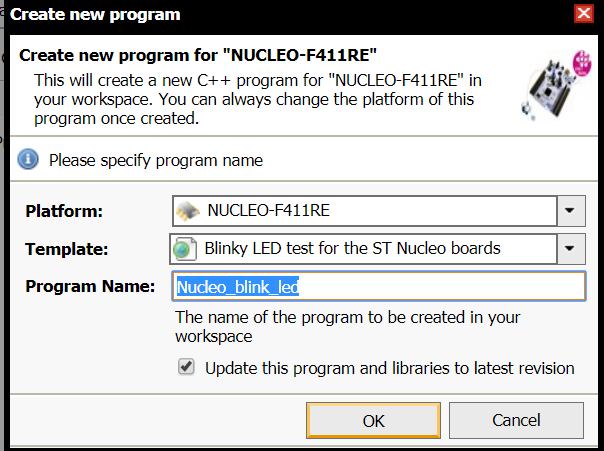
* Open <https://www.mbed.com/en/>
* Click on Compiler
* Create new program. If it asks for ‘add new platform’



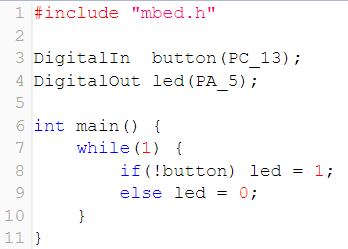
then ‘**Add Platform**’ 🡪 search for ‘NUCLEO-F411RE’ board 🡪 Click ‘**Open Mbed Complier**’



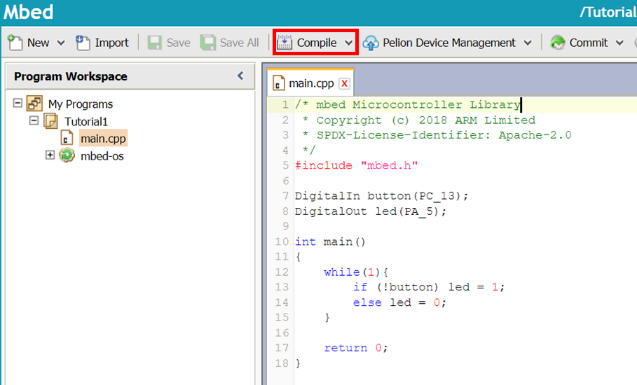
* Then, create new program as ‘ **Tutorial1\_LED**’



* Open ‘main.cpp’ and delete the example codes.
* Write the following code on ‘main.cpp’.



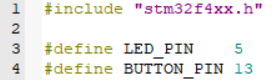
* To compile the program, click on ‘**Compile’** button

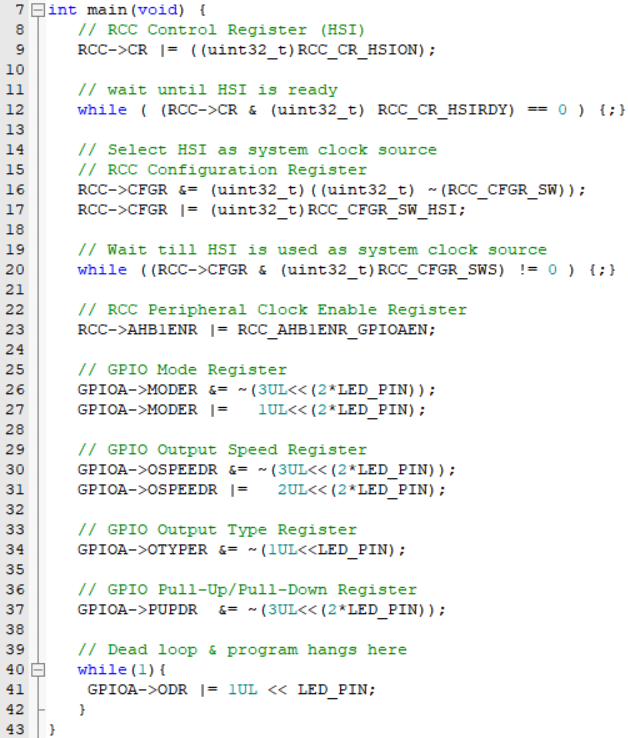


* Then, the binary file “Tutorial1\_LED.NUCLEO\_F411RE.bin” of the project will be created and downloaded on your computer.
* Connect the MCU board to your PC via USB cable and check if the drive of “ NODE\_F411RE (E:)” is created.
* To load the program onto the MCU, copy the downloaded binary file to the drive “ NODE\_F411RE (E:)”. If the program is loaded successfully then LED(LD1) will be green light.
* LED (LD2) should be turned ON when the button (USER B1) is pressed.

**B. GPIO Register tutorial**

* Open the program ‘Keil uVision5’ and create a new project. Name the project as ‘**Tutorial\_LED**’. Refer to ‘Tutorial0\_Project Start’ if you need help.
* Create a new item called ‘main.c’
* Copy and paste the given source code on ‘**main.c**’. This is an example code of turning ON LED *without* button input. You will modify this code to include the push-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.





* Compile(F7) and flash(F8) the source code on to the MCU board.
* Verify the program by checking if the LED is turned ON.

**IV. Exercise/Demo**

1. **Modify the code to turn ON LED for different speed and check any difference**
   * Low High / Very High Speed
2. **Modify the code to turn Toggle LED with a push-button input.** 
   * Button Input: Port C Pin 15 / Pull-up
   * Draw a flowchart or pseducode for this program

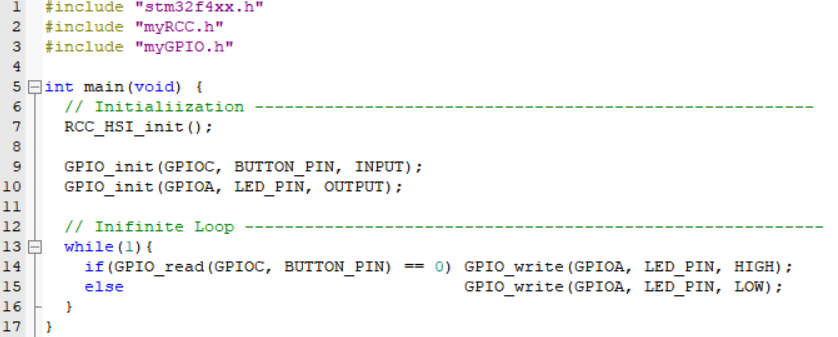
|  |
| --- |
|  |

* + Check the output pin with oscilloscope and observe how the signals change with input button

1. **Create user defined function****s for GPIO control (due by end of next lab)**

* Create your own function to initialize the clock setting, and control GPIO. Below are examples of the functions. Attach your own codes in Appendix.

|  |  |
| --- | --- |
| **Source File** | **Function** |
| myRCC.h | void RCC\_HSI\_init () |
| myGPIO.h | void GPIO\_init(GPIO\_TypeDef \*Port, int Pin, int I/OMode) |
| void GPIO\_otype(GPIO\_TypeDef \*Port, int Pin, int Otype) |
| void GPIO\_pupd(GPIO\_TypeDef \*Port, int Pin, int Pupd) |
| void GPIO\_write(GPIO\_TypeDef \*Port, int Pin, uint8 Out) |
| uint8 GPIO\_read(GPIO\_TypeDef \*Port, int Pin) |

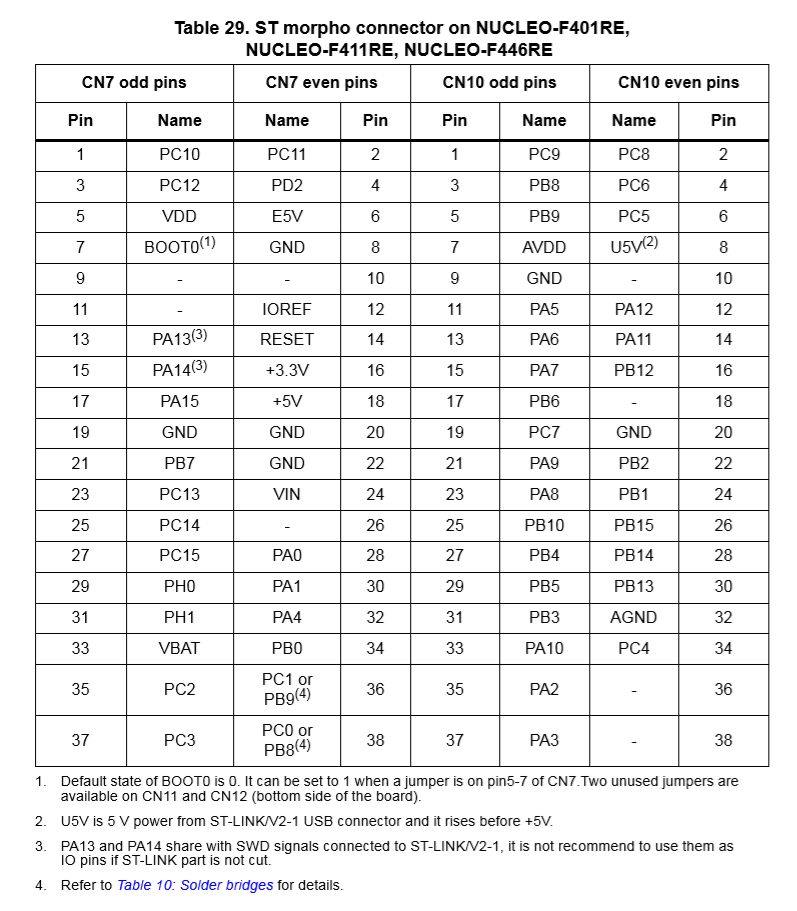
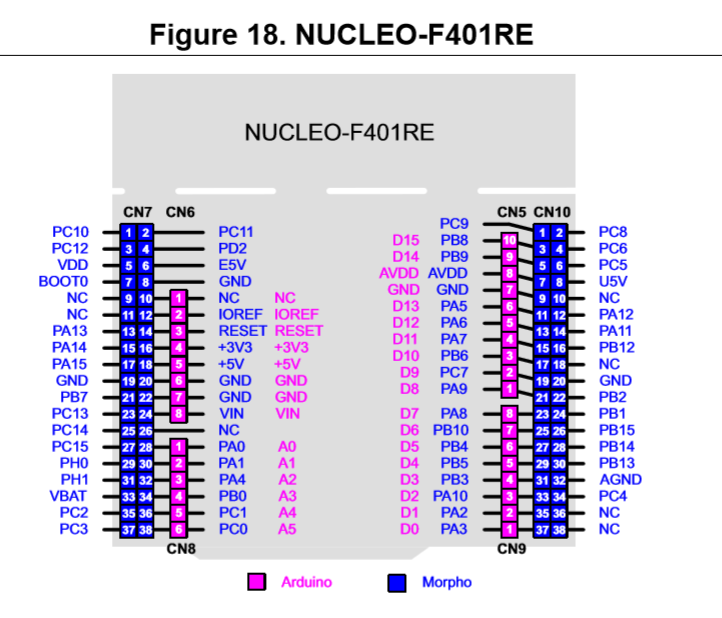
****

**V. Questions**

1. Find out a typical solution for software debouncing and hardware debouncing. What method did this NUCLEO board used for the push-button(B1)?
2. What are the differences between open-drain and Push-pull for output pin?
3. Which method of pullup/pulldown did this NUCLEO board used for LED and Button?
4. What is the GPIO output speed? Explain the advantage of low speed for LED output than high speed, if there is any.

**Appendix**

1. Pin Configuration of NUCLE-F401RE

****

1. LED/Button Circuit Diagram

