TUTORIAL: Timer Interrupt

Timer and Time Delay

Name: 김은찬 ID: 21801017

# I. Introduction

In this lab, you will learn how to set up MCU Timers for Timer Interrupt and PWM output.

Objectives of this lab are learning how to

* Configure registers of Timers (TIMx)
* Generate Timer Interrupt

### 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 Timer Interrupt

## A. Register List

List of TIMx registers for this tutorial.

|  |  |  |
| --- | --- | --- |
| Type | Register Name | Description |
| TIMx | TIMx\_ CR1 | TIMx control register 1 |
|  | TIMx\_ PSC | TIMx prescaler register |
|  | TIMx\_ARR | TIMx auto-reload register |
|  | TIMx\_DIER | TIMx DMA Interrupt Enable register |
|  | TIMx\_BDTR | TIM1(only) break and dead-time register |

Schematic

텍스트, 도표, 스크린샷, 라인이(가) 표시된 사진

자동 생성된 설명

## B. Register Setting

**System Clock setting**

1. RCC setting (PLL)

**Timer setting**

1. Enable Timer Peripheral Clock (**RCC🡪APB1ENR**)

2. Set Timer Clock Pre-scaler value (**TIMx🡪PSC : PSC[15:0]**)

3. Set Auto-reload value (**TIMx🡪ARR : ARR**)

4. Set Counting Direction (**TIMx🡪CR1 : DIR**)

5. Enable Timer DMA/Interrupt. **(TIMx🡪DIER : UIE)**

6. Enable counter (**TIMx🡪CR1 : CEN**)

**NVIC setting**

* + 1. Enable TIMx Interrupt: NVIC\_EnableIRQ(TIMx\_IRQn)
    2. Set interrupt Priority NVIC\_SetPriority(TIMx\_IRQn,2)

# III. Tutorial

## A. Register Configuration

Fill in the table

|  |  |  |
| --- | --- | --- |
| **Port/Pin** | **Description** | **Register setting** |
| RCC | PLL Initialization | RCC\_PLL\_init();  EC\_SYS\_CLK =EC\_PLL= 84,000,000 |
|  | Enable Timer Peripheral Clock: TIM2 | RCC->APB1ENR |=1≪0 |
| TIM2 | TIM2 counting direction: DIR0 | TIM2->CR1 & = ~(1<<5) |
|  | Set Timer Clock Pre-scaler value  84MHz To 100kHz | TIM2->PSC = 839 |
|  | Set Auto-reload value:  With 100kHz, counting of 1kHz | TIM2->ARR = 99 |
|  | Enable Timer DMA/Interrupt | TIM2->DIER |= 1<<0 |
|  | Enable Counter | TIM2->CR1 |= 1<<0 |
| NVIC | Set TIM2\_IRQn with Priority 2, and enable | NVIC\_SetPriority(TIM2\_IRQn, 2);  NVIC\_EnableIRQ(TIM2\_IRQn); |

## B. Programming

**Procedure**

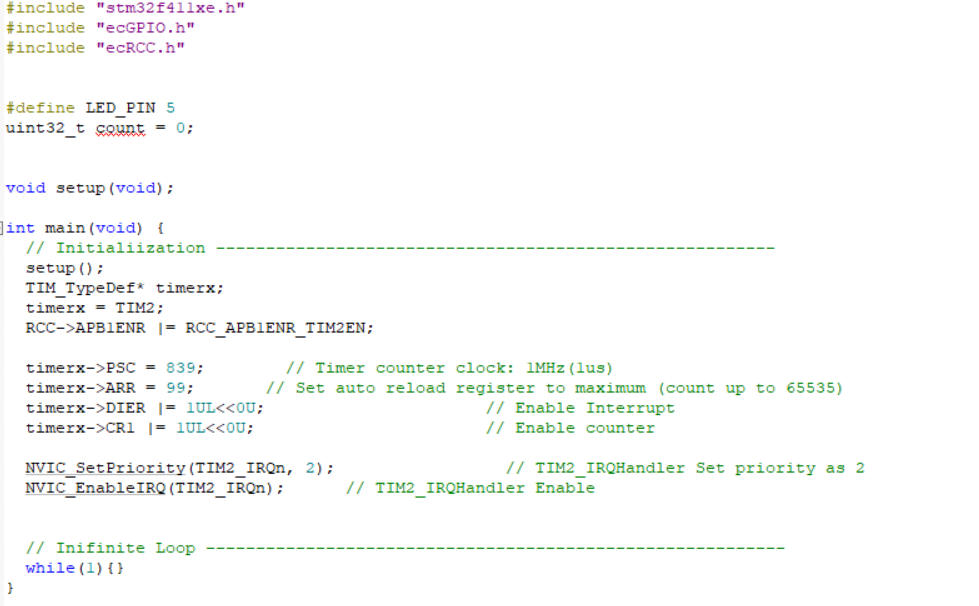
* Create a new folder ‘**EC/Tutorial/TU\_TimerInterrupt/**’
* Open the program ‘Keil uVision5’ and create a new project.
* Name the project as ‘**TU\_TimerInterrupt**.
* Create a new item called ‘**TU\_TimerInterrupt.c**’
* Use the given source code of ‘**TU\_TimerInterrupt\_student.c**’ [Click to download](https://github.com/ykkimhgu/EC-student/tree/main/tutorial/tutorial-student)
* Fill in the empty spaces in the code.
* Run the program and check your result.
* Your tutorial report must be submitted to LMS

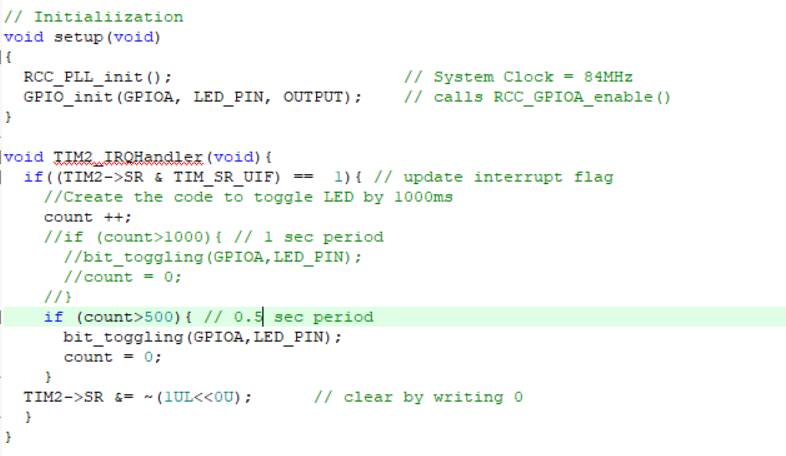
**Exercise**

* Create a simple program that turns LED on/off at 1 second period.
* System CLK is PLL 84MHz for STM32F411RE
* Use Counter of TIM2: Up-counting, Timer2\_CLK = 100 kHz, COUNT\_CLK = 1 kHz
* Use timer interrupt “void TIM2\_IRQHandler(void)”
* Now, change the LED blinking at 0.5 sec period

텍스트, 폰트, 라인, 스크린샷이(가) 표시된 사진

자동 생성된 설명

****



## Appendix

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

텍스트, 도표, 평행, 평면도이(가) 표시된 사진

자동 생성된 설명

* TIM\_TypeDef : <stm32f411xe.h>

텍스트, 스크린샷, 메뉴, 폰트이(가) 표시된 사진

자동 생성된 설명

텍스트, 폰트, 스크린샷, 화이트이(가) 표시된 사진

자동 생성된 설명

텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명