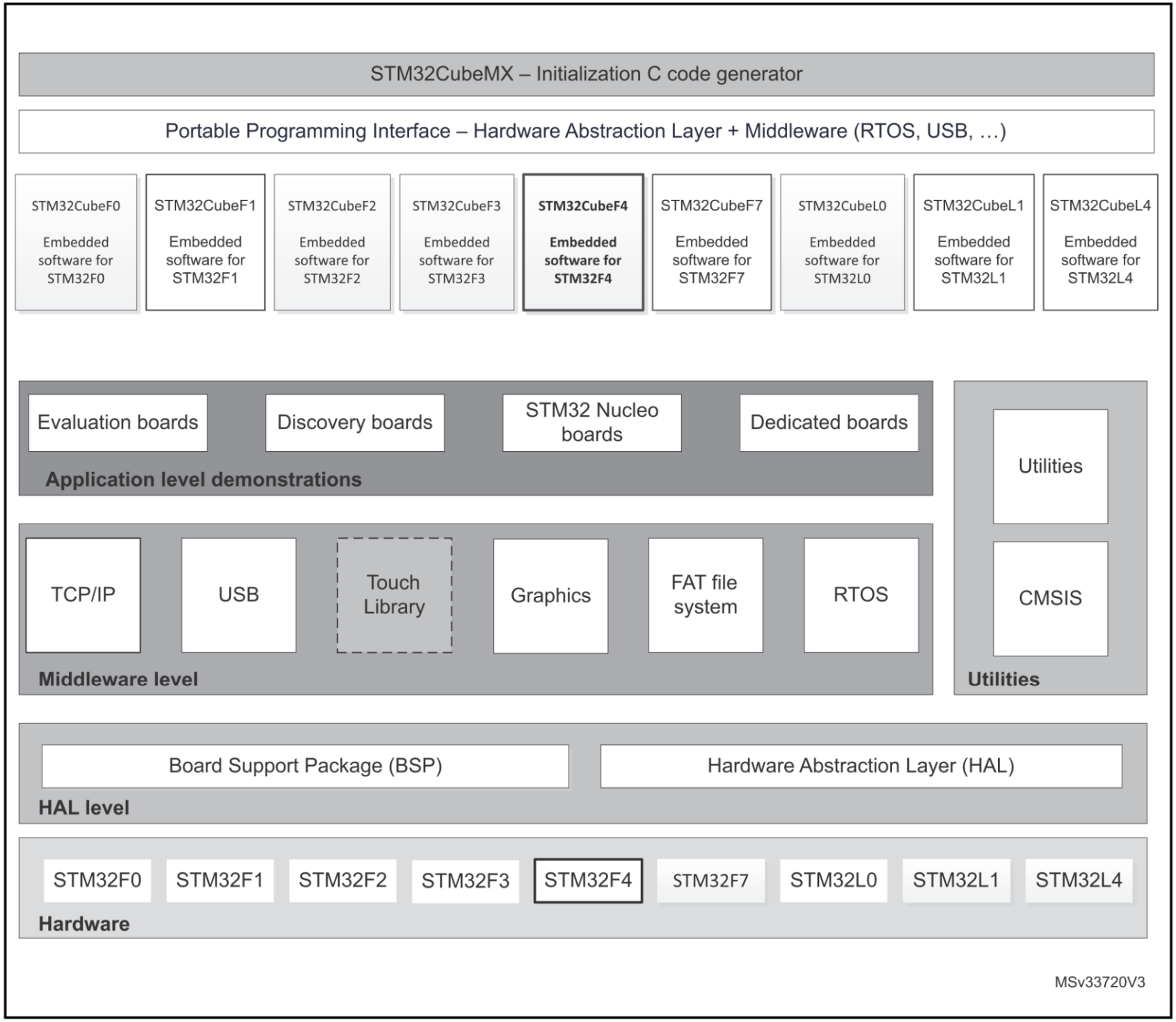
Lesson 5
STM32Cube

Lecturer: Harvard Tseng


STM32CubeMX – Initialization C code generator
Portable Programming Interface – Hardware Abstraction Layer + Middleware (RTOS, USB, …)
STM32CubeF0
Embedded software for STM32F0
STM32CubeF1
Embedded software for STM32F1
STM32CubeF2
Embedded software for STM32F2
STM32CubeF3
Embedded software for STM32F3
STM32CubeF4
Embedded software for STM32F4
STM32CubeF7
Embedded software for STM32F7
STM32CubeL0
Embedded software for STM32L0
STM32CubeL1
Embedded software for STM32L1
STM32CubeL4
Embedded software for STM32L4
Evaluation boards
Discovery boards
STM32 Nucleo boards
Dedicated boards
Application level demonstrations
Utilities
CMSIS
Utilities
TCP/IP
USB
Touch Library
Graphics
FAT file system
RTOS
Middleware level
Board Support Package (BSP)
Hardware Abstraction Layer (HAL)
HAL level
STM32F0
STM32F1
STM32F2
STM32F3
STM32F4
STM32F7
STM32L0
STM32L1
STM32L4
Hardware
MSv33720V3

What does STM32Cube contain?

- Hundreds of examples.
- Middleware level driver.
- HAL driver.
- Low layer driver.
- CMSIS

Middleware level

- TCP/IP
- USB
- Graphics
- …

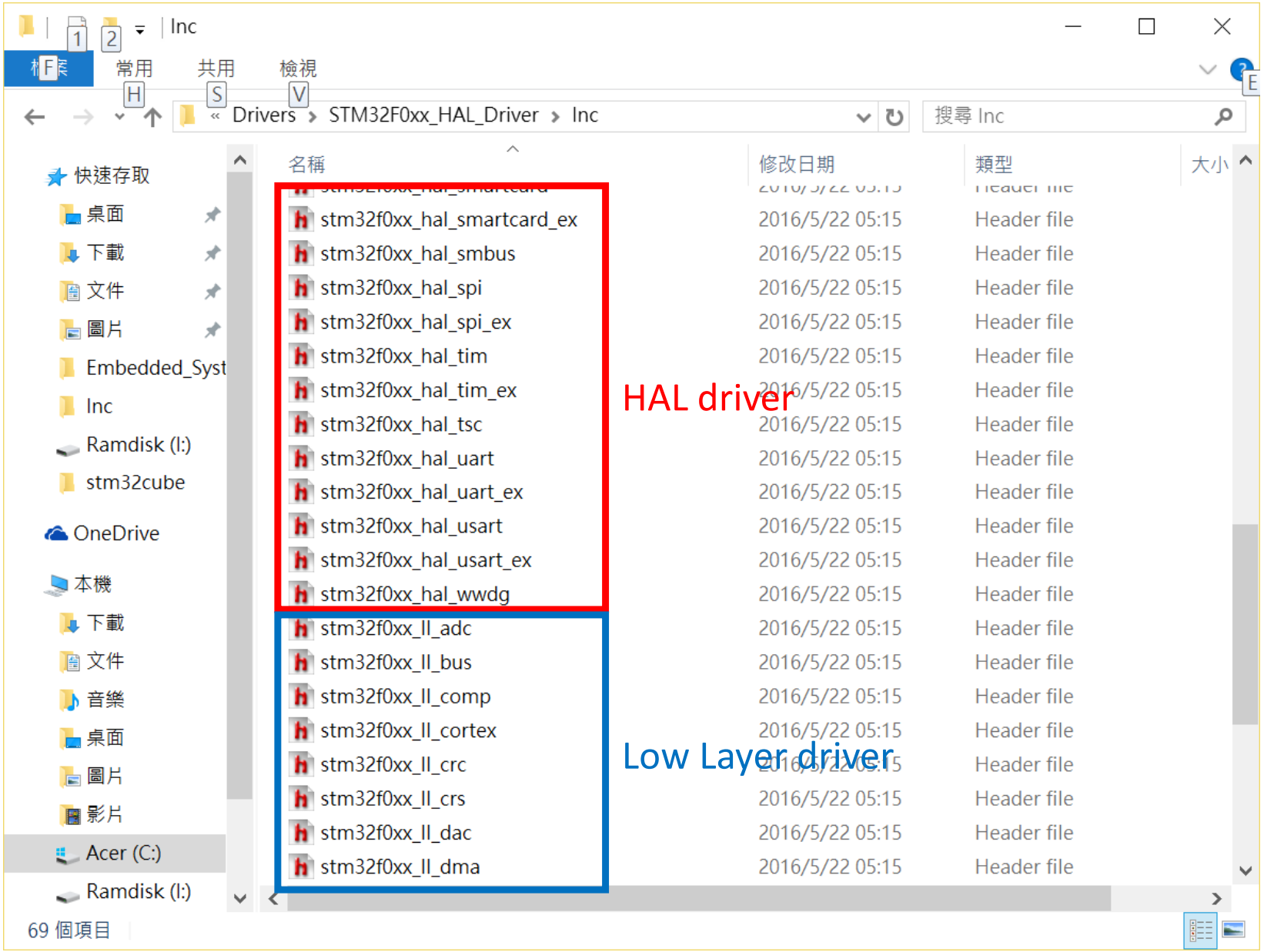
Hardware Abstraction Layer(HAL)

- Easy to use.
- Separate hardware and software.
- Enabling portability between different STM32 devices.

Low layer APIs

- Light-weight
- Optimized
- Expert oriented
- Closer to the hardware than the HAL.

Files


Inc
常用
共用
檢視
Drivers › STM32F0xx_HAL_Driver › Inc
搜尋 Inc
快速存取
桌面
下載
文件
圖片
Embedded_Syst
Inc
Ramdisk (I:)
stm32cube
OneDrive
本機
下載
文件
音樂
桌面
圖片
影片
Acer (C:)
Ramdisk (I:)
名稱
修改日期
類型
大小
stm32f0xx_hal_smartcard_ex
stm32f0xx_hal_smbus
stm32f0xx_hal_spi
stm32f0xx_hal_spi_ex
stm32f0xx_hal_tim
stm32f0xx_hal_tim_ex
stm32f0xx_hal_tsc
stm32f0xx_hal_uart
stm32f0xx_hal_uart_ex
stm32f0xx_hal_usart
stm32f0xx_hal_usart_ex
stm32f0xx_hal_wwdg
stm32f0xx_ll_adc
stm32f0xx_ll_bus
stm32f0xx_ll_comp
stm32f0xx_ll_cortex
stm32f0xx_ll_crc
stm32f0xx_ll_crs
stm32f0xx_ll_dac
stm32f0xx_ll_dma
2016/5/22 05:15
Header file
HAL driver
Low Layer driver
69 個項目

HAL Driver Setup

Step1. Download STM32CubeF4

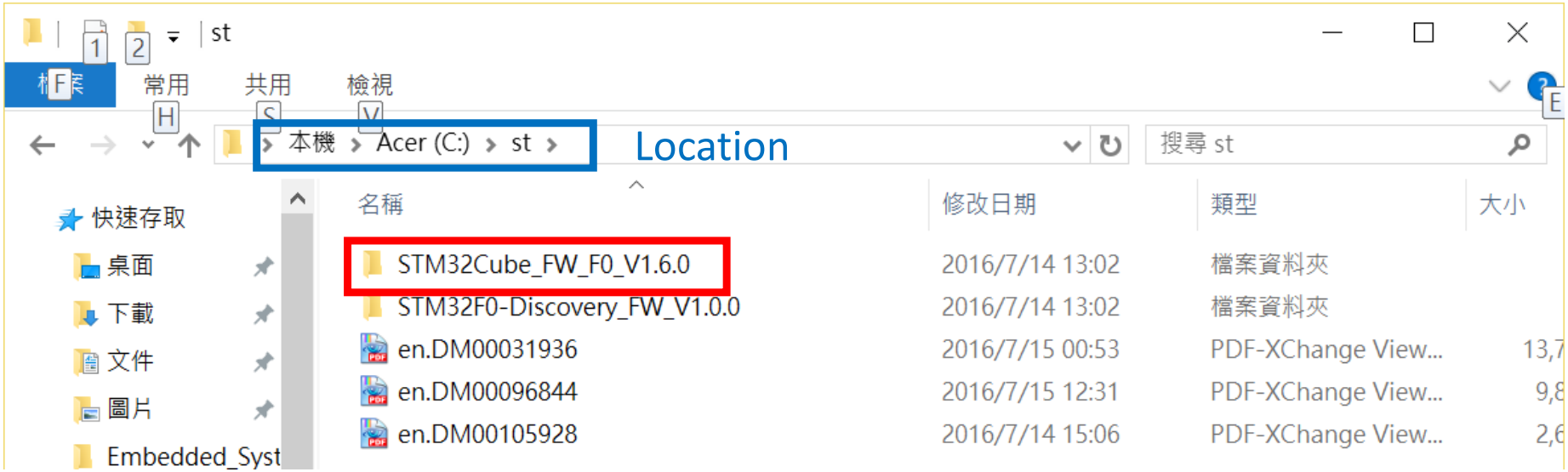
- Download from ST.

GET SOFTWARE

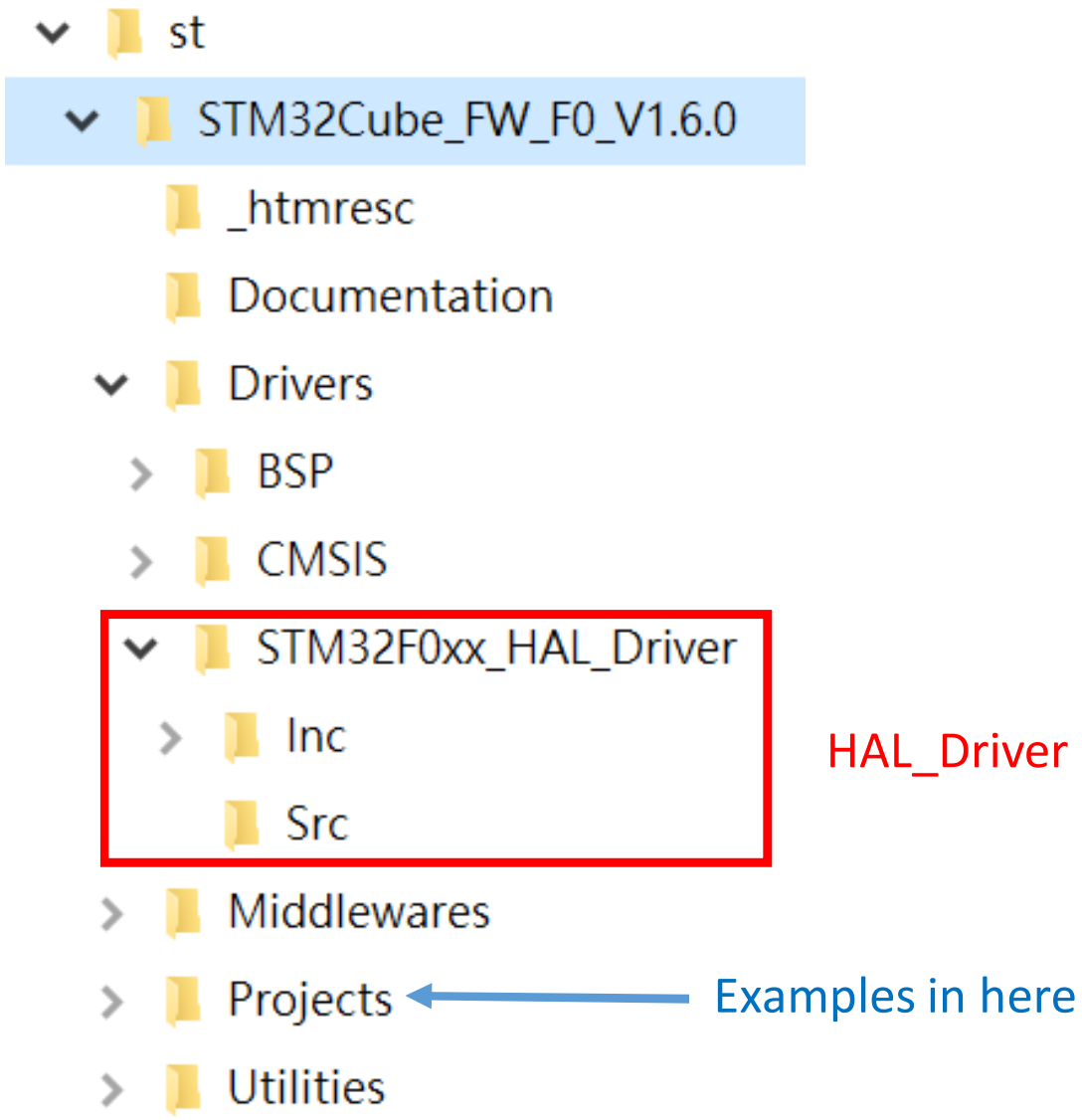
Part Number	Software Version	Marketing Status	Supplier	Order from ST
STM32CubeF4	1.13.0	Active	ST	Get Software

Step2. Extract

- Extract to folder C:\st\

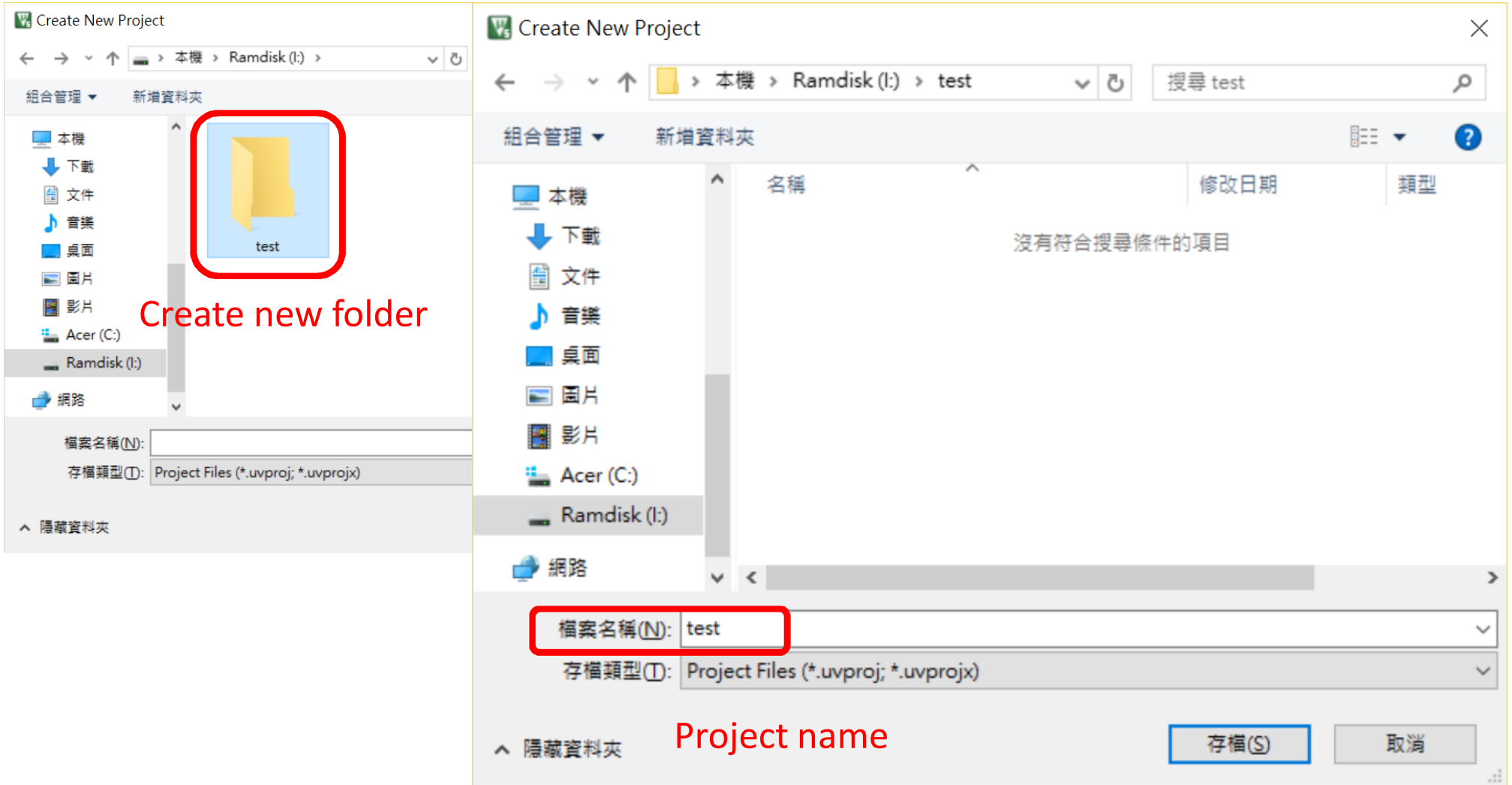


File tree


st
STM32Cube_FW_F0_V1.6.0
_htmresc
Documentation
Drivers
BSP
CMSIS
STM32F0xx_HAL_Driver
Inc
Src
HAL_Driver
Middlewares
Projects
Examples in here
Utilities

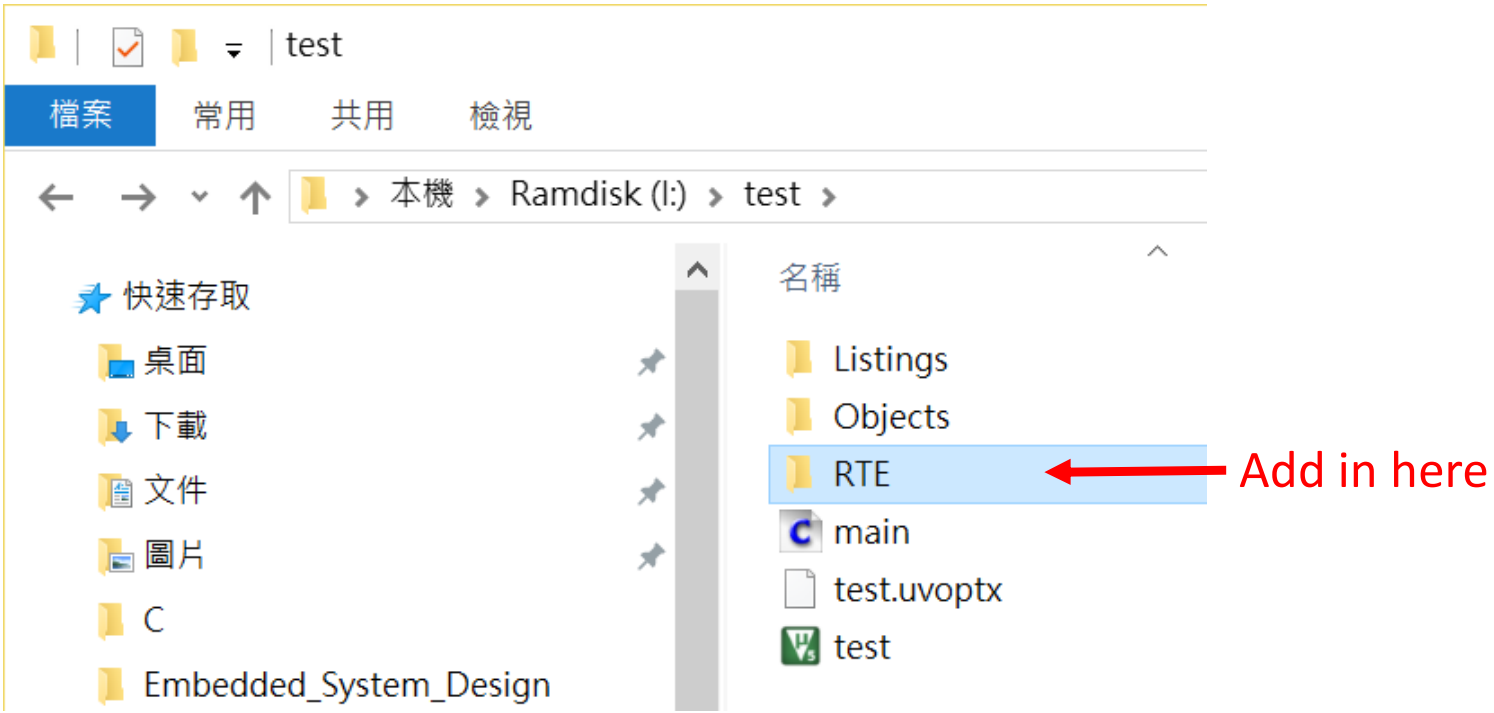
Step3. Create new project

- Setup as usual.

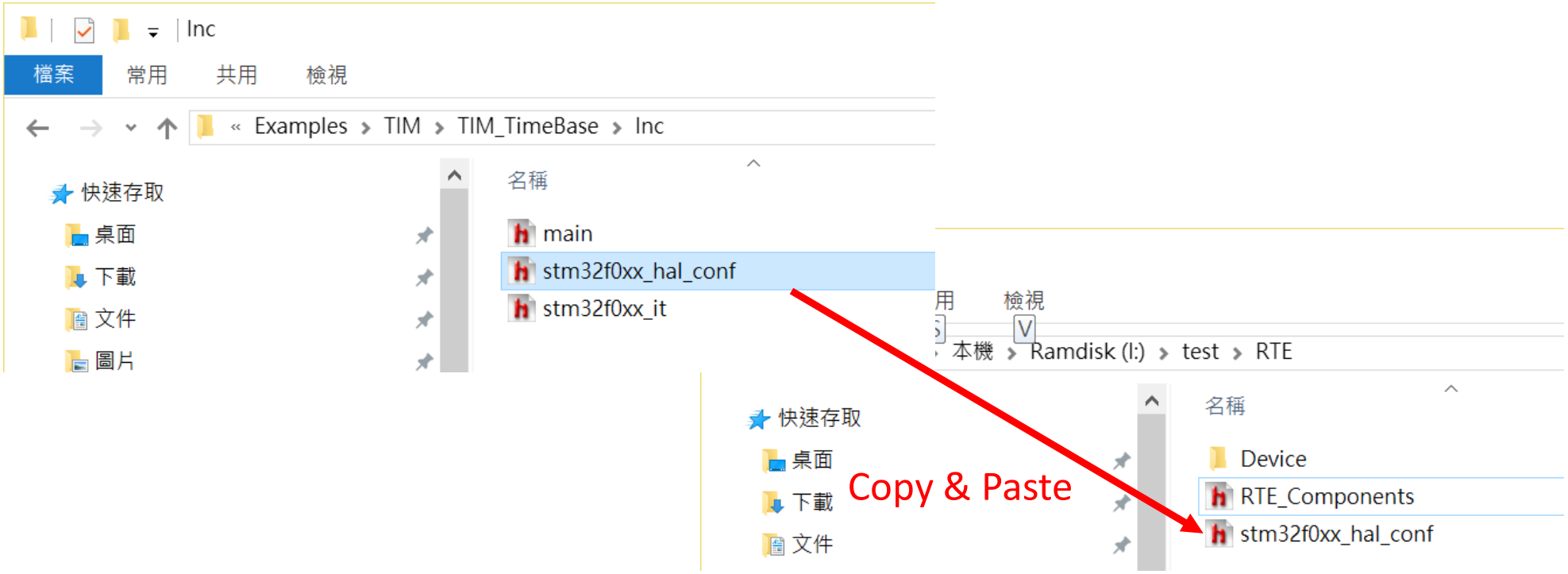


Step4. Add stm32f4xx_hal_conf.h

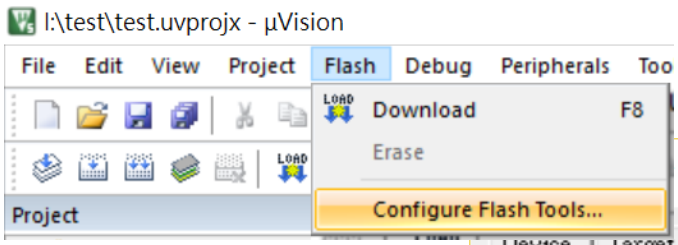
- This header can be found in any example.
- Copy and paste to our project folder/RTE.



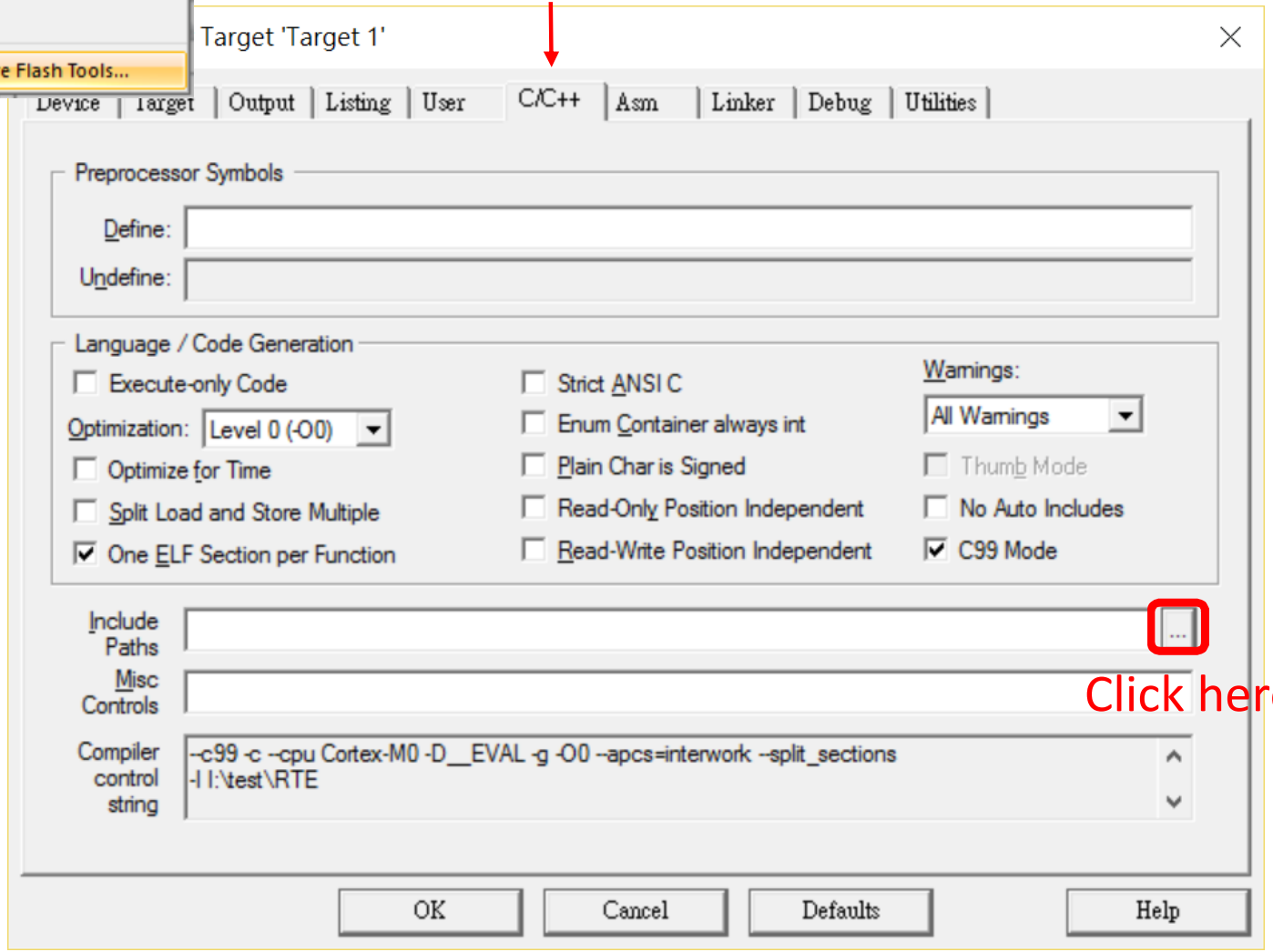
Step4. Add stm32f4xx_hal_conf.h



Step5. Add include path

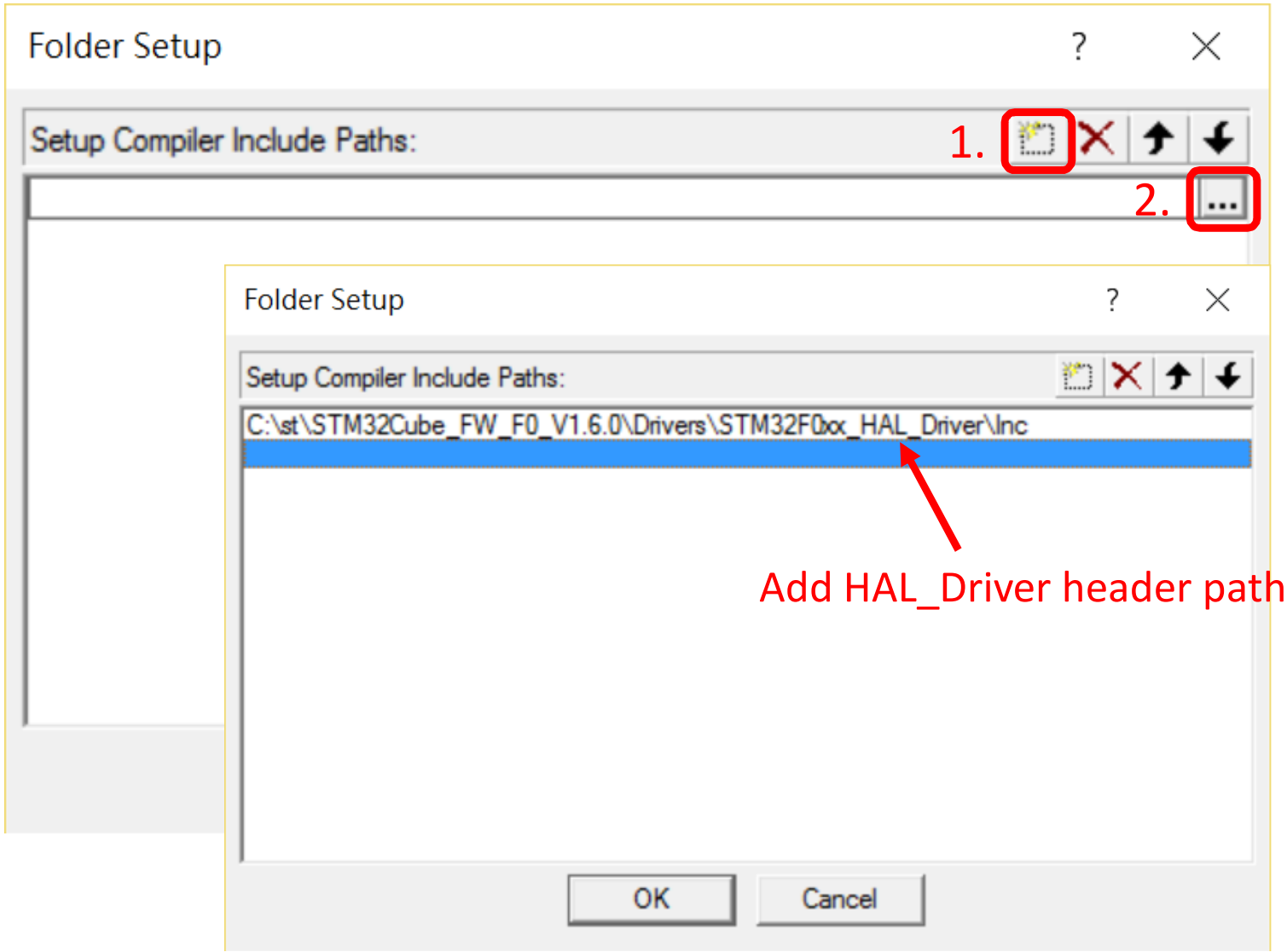


C/C++



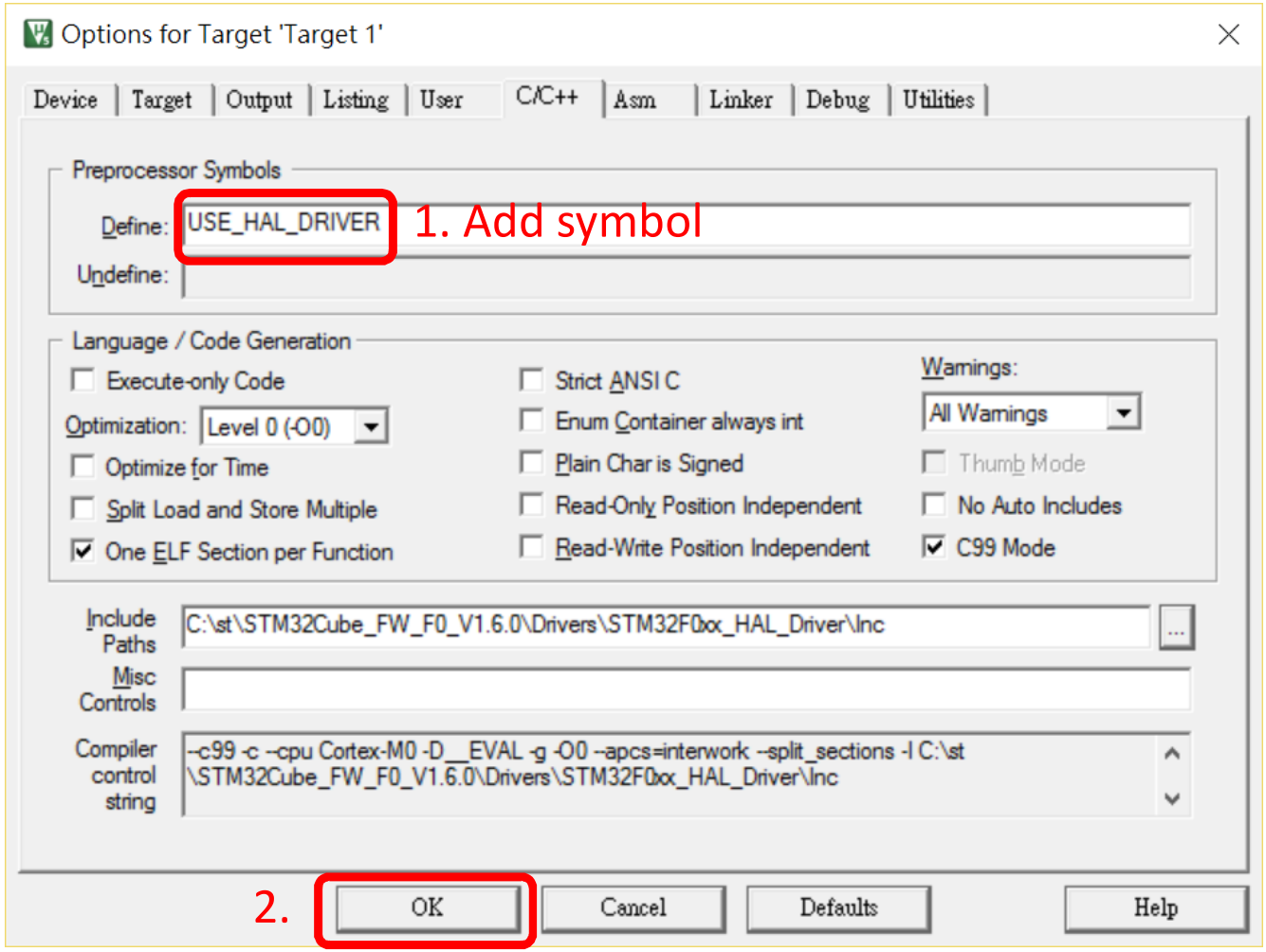
Click here

Step5. Add include path



Step6. Add Preprocessor Symbol

- Type USE_HAL_DRIVER.



Step7. Include HAL header

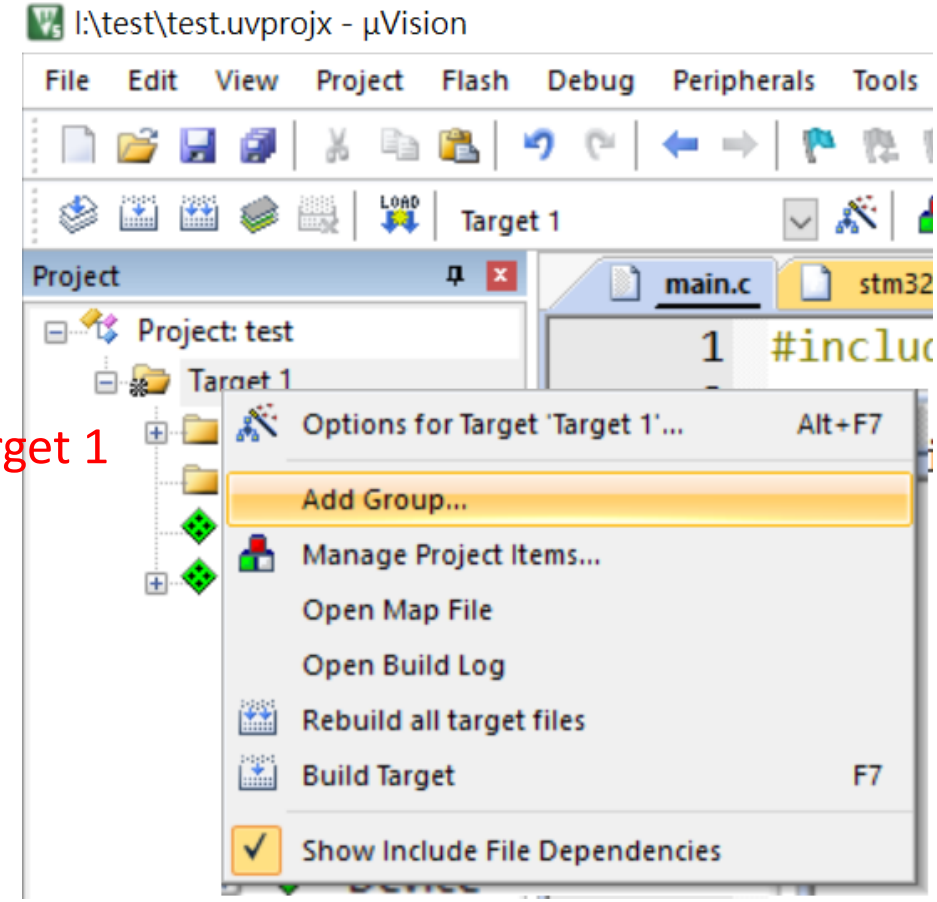
main.c*

```
1  #include "stm32f0xx.h"
2  #include "stm32f0xx_hal_rcc.h"
3  #include "stm32f0xx_hal_gpio.h"
4
5  int main(void){
6
7  }
8
```

Step8. Add source file

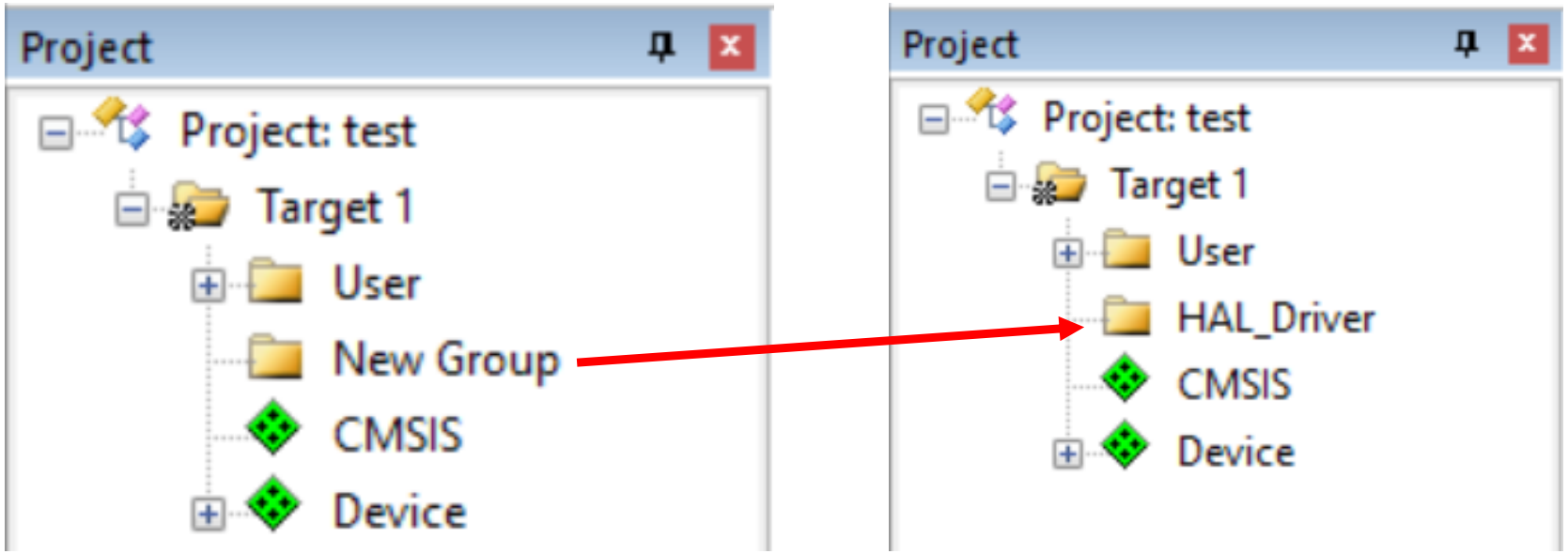
- Add group.

Right click on Target 1

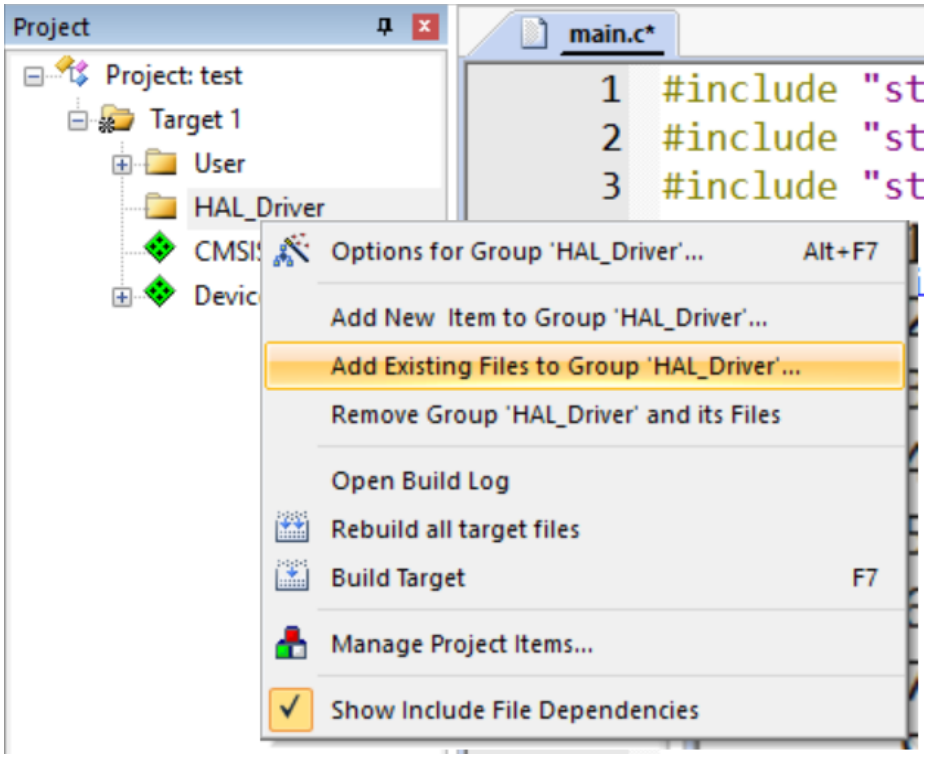


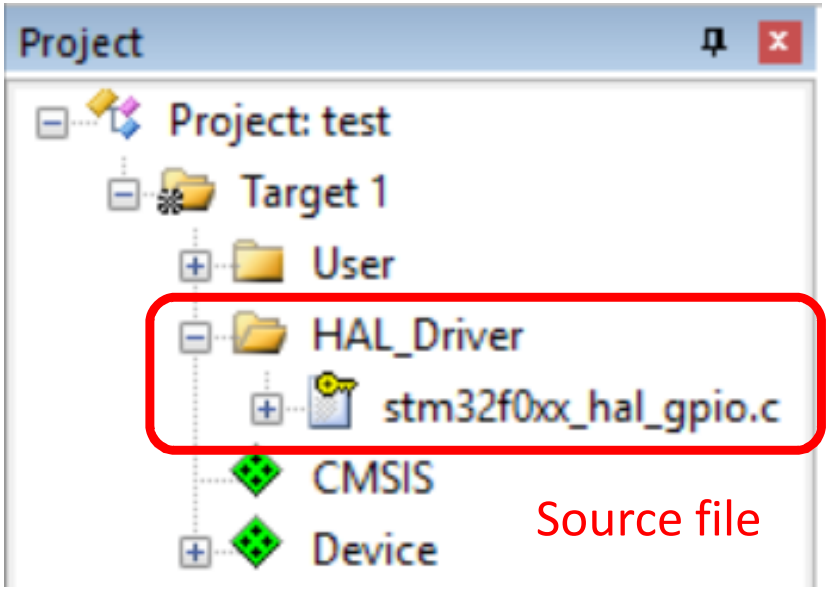
Step8. Add source file

- Rename New Group to HAL_Driver.



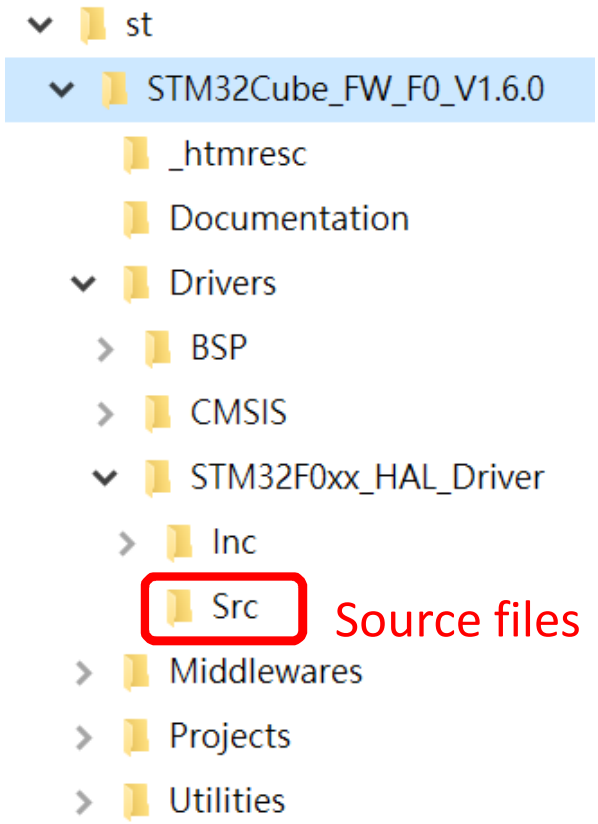
Step8. Add source file





Step8. Add source file

- Source files will be in here.



Step9. Test

- Blinky LED.

main.c

```c
#include "stm32f0xx.h"                          // Device header
#include "stm32f0xx_hal_rcc.h"
#include "stm32f0xx_hal_gpio.h"

int main(void){
    __HAL_RCC_GPIOC_CLK_ENABLE();
    GPIO_InitTypeDef GPIO_Init;
    GPIO_Init.Pin = GPIO_PIN_8 | GPIO_PIN_9;
    GPIO_Init.Mode = GPIO_MODE_OUTPUT_PP;
    HAL_GPIO_Init(GPIOC, &GPIO_Init);

    while(1){
        HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_9);
        for(int i=0; i<800000; i++);
    }
}
```

Exercise

Transfer previous code into HAL API

main.c | stm32f0xx.h | stm32f051x8.h

```c
#include "stm32f0xx.h"                    // Device header

int main(void){
    RCC->AHBENR |= RCC_AHBENR_GPIOCEN;

    GPIOC->MODER |= (0x1 << 2*8) | (0x1 << 2*9);
    while(1){
        GPIOC->BSRR = (0x1 << 9);
        for(int i=0; i<800000; i++);
        GPIOC->BSRR = (0x1 << (9+16));
        for(int i=0; i<800000; i++);
    }
}
```

main.c

```c
#include "stm32f0xx.h"                    // Device header
#include "stm32f0xx_hal_rcc.h"
#include "stm32f0xx_hal_gpio.h"

int main(void){
    __HAL_RCC_GPIOC_CLK_ENABLE();
    GPIO_InitTypeDef GPIO_Init;
    GPIO_Init.Pin = GPIO_PIN_8 | GPIO_PIN_9;
    GPIO_Init.Mode = GPIO_MODE_OUTPUT_PP;
    HAL_GPIO_Init(GPIOC, &GPIO_Init);

    while(1){
        HAL_GPIO_TogglePin(GPIOC, GPIO_PIN_9);
        for(int i=0; i<800000; i++);
    }
}
```