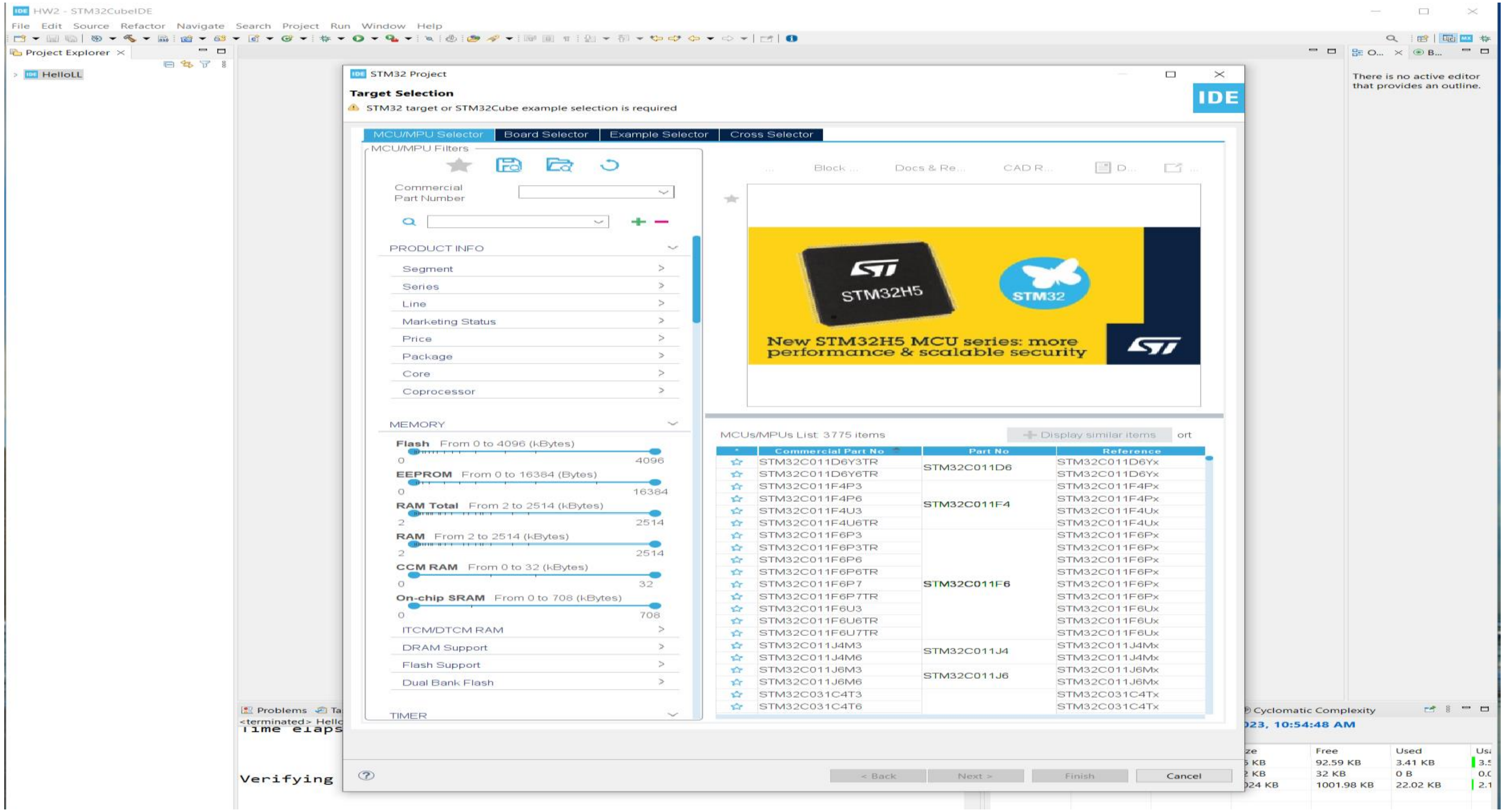
UCSD Embedded RTOS Assignment 8

By

Hsuankai Chang

hsuankac@umich.edu

Step 1. Startup STM32CubeIDE and create new STM32 project



STM32 Project

Target Selection

⚠ STM32 target or STM32Cube example selection is required

MCU/MPU Filters

Commercial Part Number

PRODUCT INFO
- Segment >
- Series >
- Line >
- Marketing Status >
- Price >
- Package >
- Core >
- Coprocessor >

MEMORY

Flash From 0 to 4096 (kBytes) — 0 … 4096

EEPROM From 0 to 16384 (Bytes) — 0 … 16384

RAM Total From 2 to 2514 (kBytes) — 2 … 2514

RAM From 2 to 2514 (kBytes) — 2 … 2514

CCM RAM From 0 to 32 (kBytes) — 0 … 32

On-chip SRAM From 0 to 708 (kBytes) — 0 … 708

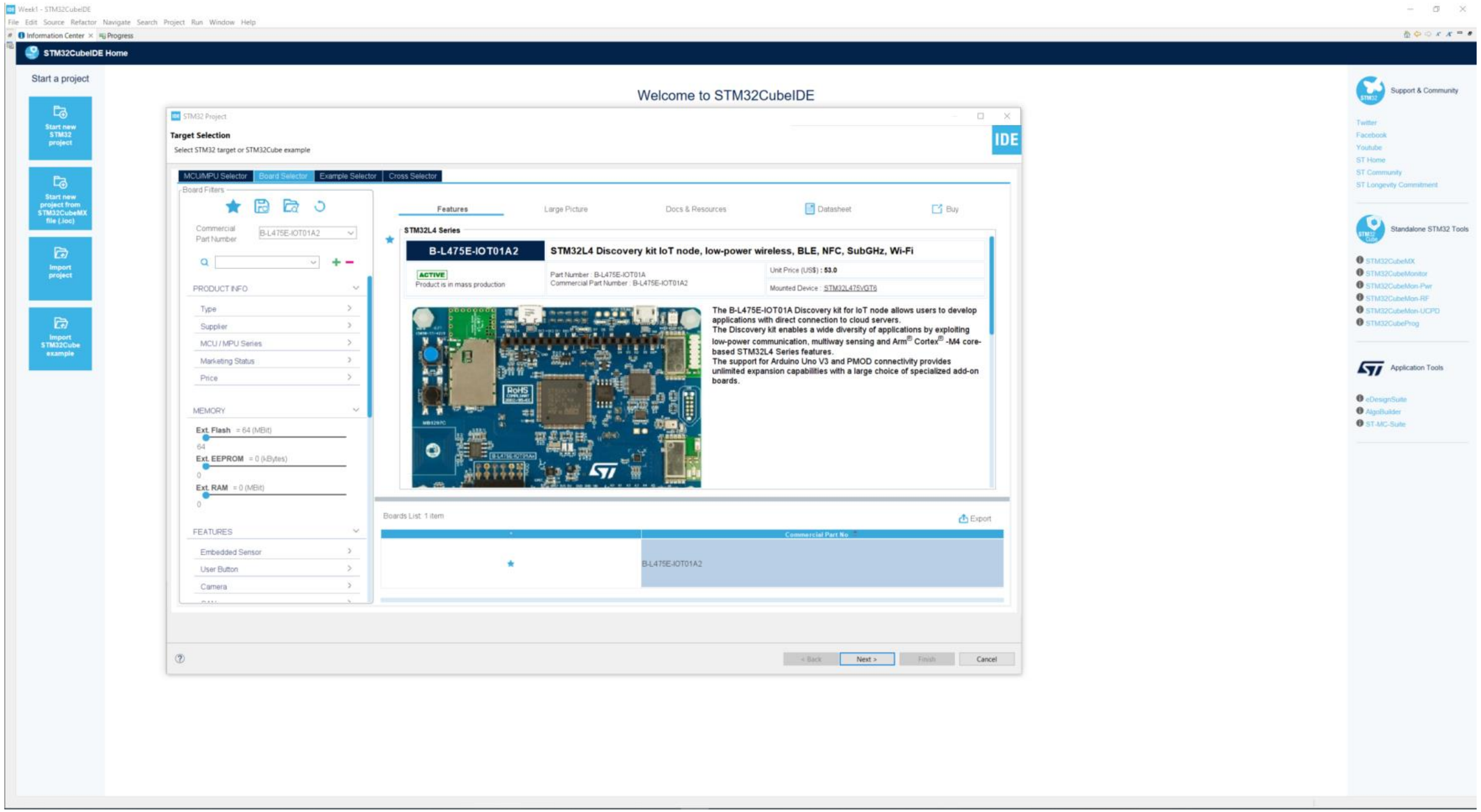
- ITCM/DTCM RAM >
- DRAM Support >
- Flash Support >
- Dual Bank Flash >

TIMER

MCUs/MPUs List: 3775 items

*	Commercial Part No	Part No	Reference
☆	STM32C011D6Y3TR	STM32C011D6	STM32C011D6Yx
☆	STM32C011D6Y6TR		STM32C011D6Yx
☆	STM32C011F4P3	STM32C011F4	STM32C011F4Px
☆	STM32C011F4P6		STM32C011F4Px
☆	STM32C011F4U3		STM32C011F4Ux
☆	STM32C011F4U6TR		STM32C011F4Ux
☆	STM32C011F6P3	STM32C011F6	STM32C011F6Px
☆	STM32C011F6P3TR		STM32C011F6Px
☆	STM32C011F6P6		STM32C011F6Px
☆	STM32C011F6P6TR		STM32C011F6Px
☆	STM32C011F6P7		STM32C011F6Px
☆	STM32C011F6P7TR		STM32C011F6Px
☆	STM32C011F6U3		STM32C011F6Ux
☆	STM32C011F6U6TR		STM32C011F6Ux
☆	STM32C011F6U7TR		STM32C011F6Ux
☆	STM32C011J4M3	STM32C011J4	STM32C011J4Mx
☆	STM32C011J4M6		STM32C011J4Mx
☆	STM32C011J6M3	STM32C011J6	STM32C011J6Mx
☆	STM32C011J6M6		STM32C011J6Mx
☆	STM32C031C4T3		STM32C031C4Tx
☆	STM32C031C4T6		STM32C031C4Tx

Step 2. Access board selector and type in the board you use, click Next



Welcome to STM32CubeIDE

Target Selection

Select STM32 target or STM32Cube example

Commercial Part Number: B-L475E-IOT01A2

STM32L4 Series

B-L475E-IOT01A2 **STM32L4 Discovery kit IoT node, low-power wireless, BLE, NFC, SubGHz, Wi-Fi**

ACTIVE
Product is in mass production

Part Number : B-L475E-IOT01A
Commercial Part Number : B-L475E-IOT01A2

Unit Price (US$) : 53.0

Mounted Device : STM32L475VGT6

The B-L475E-IOT01A Discovery kit for IoT node allows users to develop applications with direct connection to cloud servers.
The Discovery kit enables a wide diversity of applications by exploiting low-power communication, multiway sensing and Arm® Cortex® -M4 core-based STM32L4 Series features.
The support for Arduino Uno V3 and PMOD connectivity provides unlimited expansion capabilities with a large choice of specialized add-on boards.

Ext. Flash = 64 (MBit)

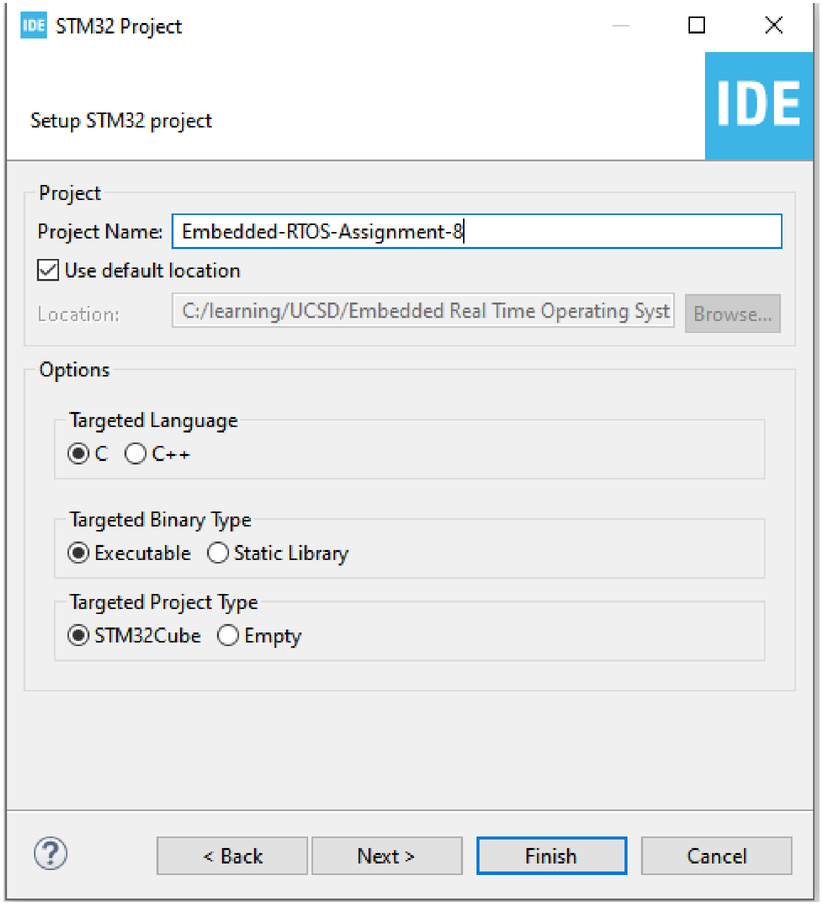
Ext. EEPROM = 0 (kBytes)

Ext. RAM = 0 (MBit)

Boards List: 1 item

	Commercial Part No
★	B-L475E-IOT01A2

Step 3. Enter the project name then click Next



STM32 Project

Setup STM32 project

Project

Project Name: Embedded-RTOS-Assignment-8

☑ Use default location

Location: C:/learning/UCSD/Embedded Real Time Operating Syst

Options

Targeted Language

● C ○ C++

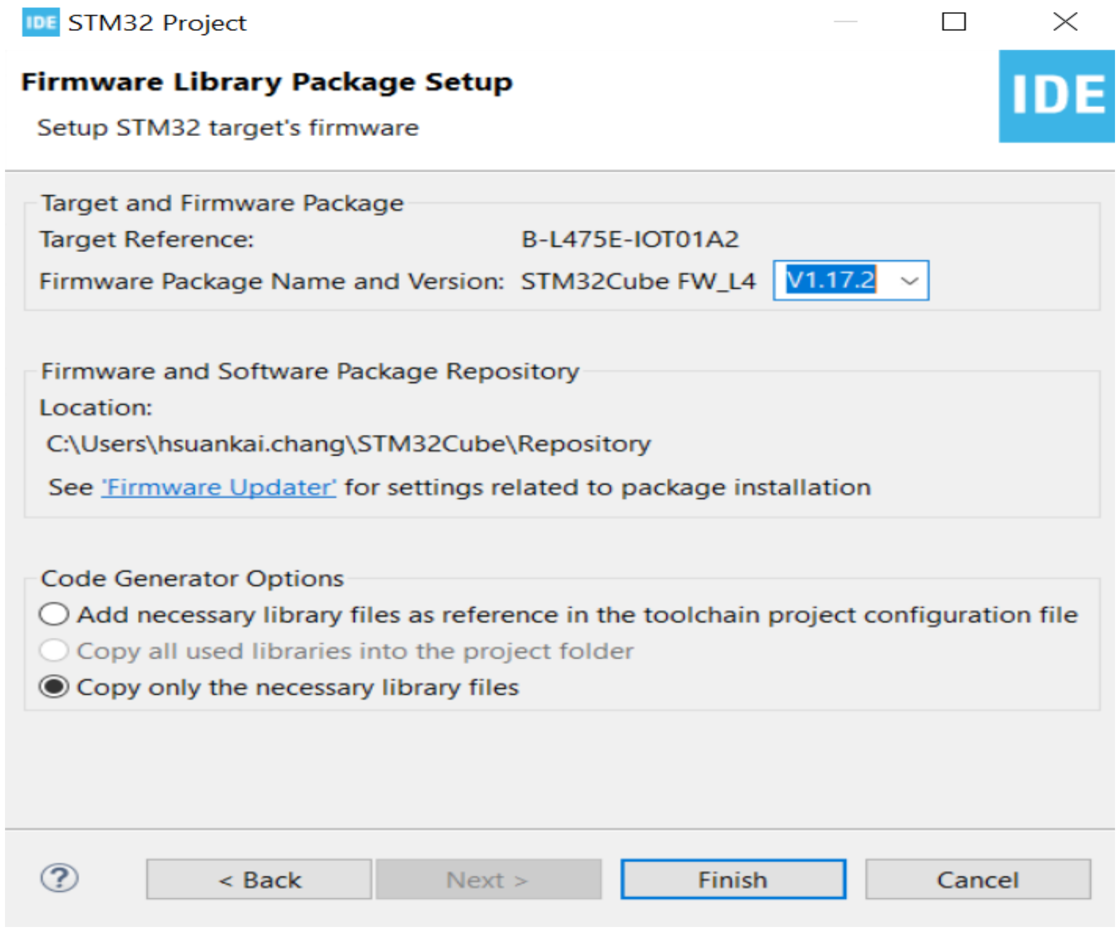
Targeted Binary Type

● Executable ○ Static Library

Targeted Project Type

● STM32Cube ○ Empty

Step 4. See the firmware package name and version



STM32 Project

Firmware Library Package Setup

Setup STM32 target's firmware

Target and Firmware Package

Target Reference: B-L475E-IOT01A2

Firmware Package Name and Version: STM32Cube FW_L4 V1.17.2

Firmware and Software Package Repository

Location:

C:\Users\hsuankai.chang\STM32Cube\Repository

See 'Firmware Updater' for settings related to package installation

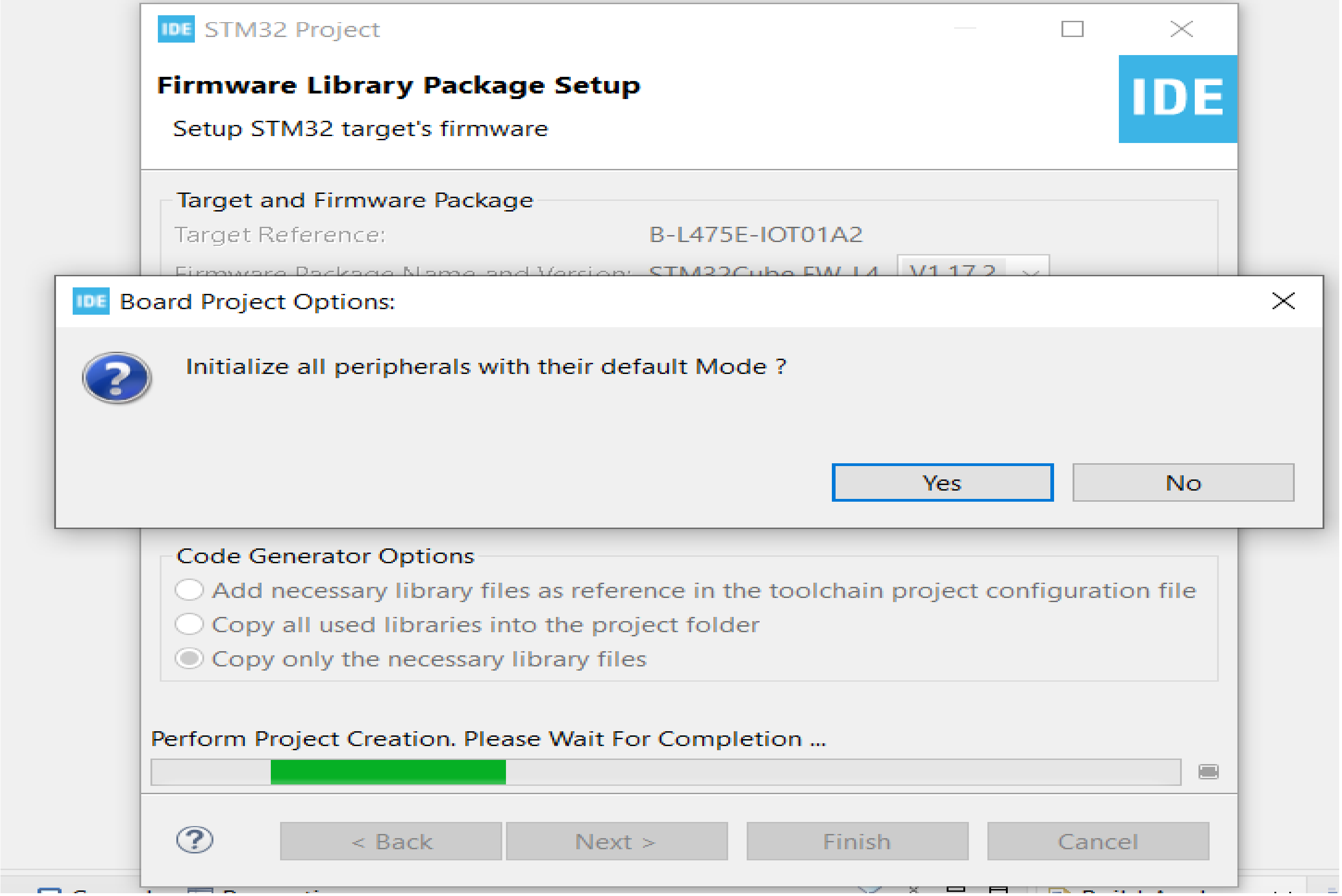
Code Generator Options

○ Add necessary library files as reference in the toolchain project configuration file

○ Copy all used libraries into the project folder

● Copy only the necessary library files

Step 5. Click yes to initialize all peripherals to default



STM32 Project

Firmware Library Package Setup

Setup STM32 target's firmware

Target and Firmware Package

Target Reference: B-L475E-IOT01A2

Firmware Package Name and Version: STM32Cube FW_L4 V1.17.2

Board Project Options:

Initialize all peripherals with their default Mode ?

Code Generator Options

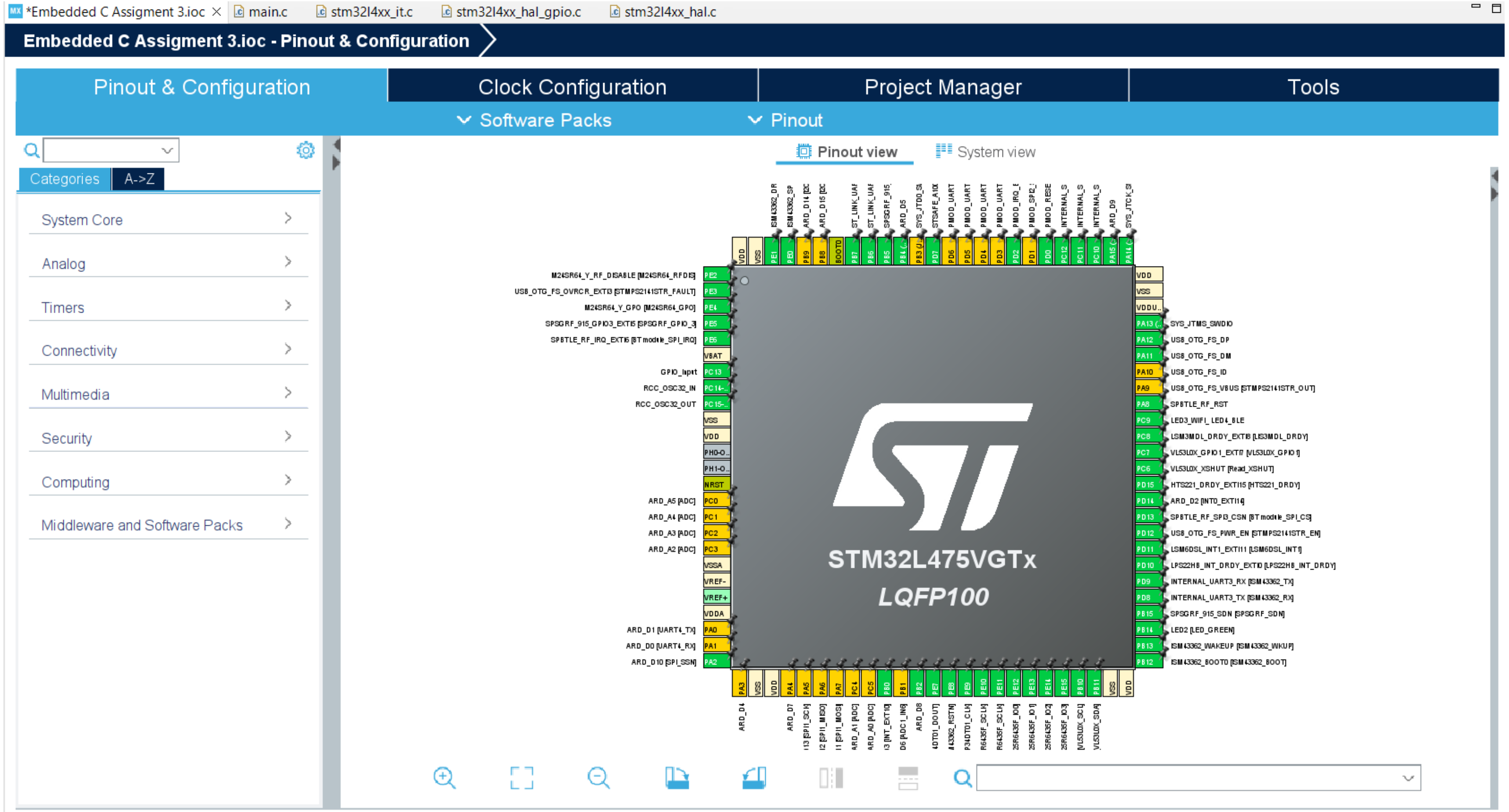
○ Add necessary library files as reference in the toolchain project configuration file

○ Copy all used libraries into the project folder

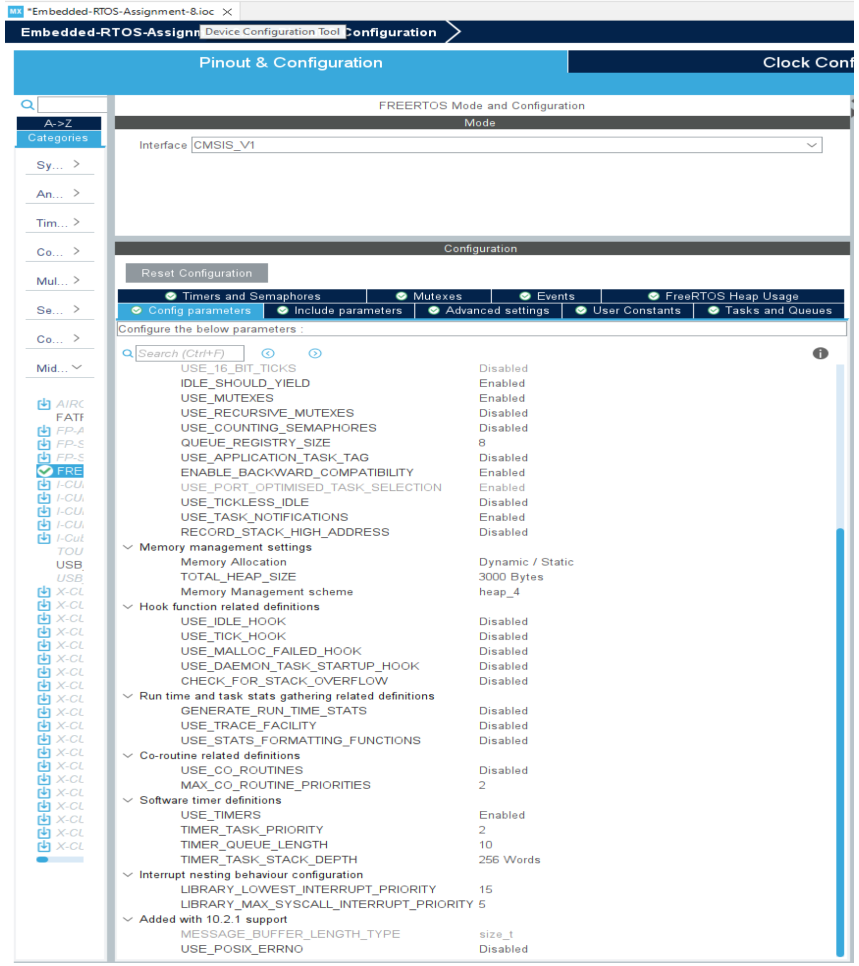
● Copy only the necessary library files

Perform Project Creation. Please Wait For Completion ...

Step 6. When in .ioc file, click Pinout & Configurations


*Embedded C Assigment 3.ioc
main.c
stm32l4xx_it.c
stm32l4xx_hal_gpio.c
stm32l4xx_hal.c
Embedded C Assigment 3.ioc - Pinout & Configuration
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Software Packs
Pinout
Categories
A->Z
System Core
Analog
Timers
Connectivity
Multimedia
Security
Computing
Middleware and Software Packs
Pinout view
System view
STM32L475VGTx
LQFP100
M24SR64_Y_RF_DISABLE [M24SR64_RFDIS]
USB_OTG_FS_OVRCR_EXTI3 [STMPS2141STR_FAULT]
M24SR64_Y_GPO [M24SR64_GPO]
SPSGRF_915_GPIO3_EXTI5 [SPSGRF_GPIO_3]
SPBTLE_RF_IRQ_EXTI6 [BT module_SPI_IRQ]
GPIO_Input
RCC_OSC32_IN
RCC_OSC32_OUT
ARD_A5 [ADC]
ARD_A4 [ADC]
ARD_A3 [ADC]
ARD_A2 [ADC]
ARD_D1 [UART4_TX]
ARD_D0 [UART4_RX]
ARD_D10 [SPI_SSN]
SYS_JTMS_SWDIO
USB_OTG_FS_DP
USB_OTG_FS_DM
USB_OTG_FS_ID
USB_OTG_FS_VBUS [STMPS2141STR_OUT]
SPBTLE_RF_RST
LED3_WIFI_ LED4_BLE
LSM3MDL_DRDY_EXTI8 [LIS3MDL_DRDY]
VL53L0X_GPIO1_EXTI7 [VL53L0X_GPIO1]
VL53L0X_XSHUT [Read_XSHUT]
HTS221_DRDY_EXTI15 [HTS221_DRDY]
ARD_D2 [INT0_EXTI14]
SPBTLE_RF_SPI3_CSN [BT module_SPI_CS]
USB_OTG_FS_PWR_EN [STMPS2141STR_EN]
LSM6DSL_INT1_EXTI11 [LSM6DSL_INT1]
LPS22HB_INT_DRDY_EXTI0 [LPS22HB_INT_DRDY]
INTERNAL_UART3_RX [ISM43362_TX]
INTERNAL_UART3_TX [ISM43362_RX]
SPSGRF_915_SDN [SPSGRF_SDN]
LED2 [LED_GREEN]
ISM43362_WAKEUP [ISM43362_WKUP]
ISM43362_BOOT0 [ISM43362_BOOT]

Step 7. Enable FreeRTOS CMSIS_V1 and enable timers



FREERTOS Mode and Configuration

Mode

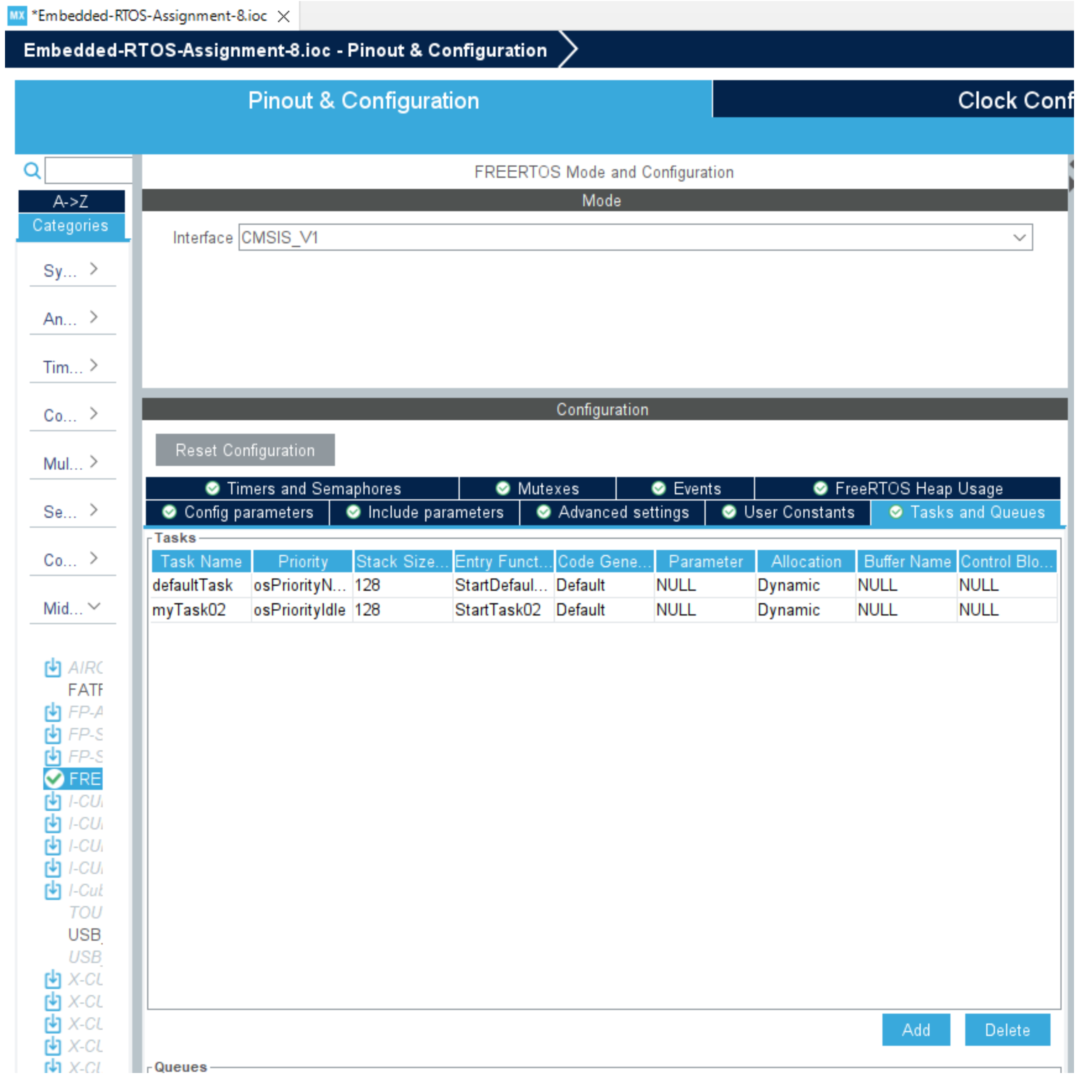
Interface CMSIS_V1

Configuration

Configure the below parameters :

```
        USE_16_BIT_TICKS                          Disabled
        IDLE_SHOULD_YIELD                         Enabled
        USE_MUTEXES                               Enabled
        USE_RECURSIVE_MUTEXES                     Disabled
        USE_COUNTING_SEMAPHORES                   Disabled
        QUEUE_REGISTRY_SIZE                       8
        USE_APPLICATION_TASK_TAG                  Disabled
        ENABLE_BACKWARD_COMPATIBILITY             Enabled
        USE_PORT_OPTIMISED_TASK_SELECTION         Enabled
        USE_TICKLESS_IDLE                         Disabled
        USE_TASK_NOTIFICATIONS                    Enabled
        RECORD_STACK_HIGH_ADDRESS                 Disabled
  Memory management settings
        Memory Allocation                         Dynamic / Static
        TOTAL_HEAP_SIZE                           3000 Bytes
        Memory Management scheme                  heap_4
  Hook function related definitions
        USE_IDLE_HOOK                             Disabled
        USE_TICK_HOOK                             Disabled
        USE_MALLOC_FAILED_HOOK                    Disabled
        USE_DAEMON_TASK_STARTUP_HOOK              Disabled
        CHECK_FOR_STACK_OVERFLOW                  Disabled
  Run time and task stats gathering related definitions
        GENERATE_RUN_TIME_STATS                   Disabled
        USE_TRACE_FACILITY                        Disabled
        USE_STATS_FORMATTING_FUNCTIONS            Disabled
  Co-routine related definitions
        USE_CO_ROUTINES                           Disabled
        MAX_CO_ROUTINE_PRIORITIES                 2
  Software timer definitions
        USE_TIMERS                                Enabled
        TIMER_TASK_PRIORITY                       2
        TIMER_QUEUE_LENGTH                        10
        TIMER_TASK_STACK_DEPTH                    256 Words
  Interrupt nesting behaviour configuration
        LIBRARY_LOWEST_INTERRUPT_PRIORITY         15
        LIBRARY_MAX_SYSCALL_INTERRUPT_PRIORITY 5
  Added with 10.2.1 support
        MESSAGE_BUFFER_LENGTH_TYPE                size_t
        USE_POSIX_ERRNO                           Disabled
```

Step 8. Add second task

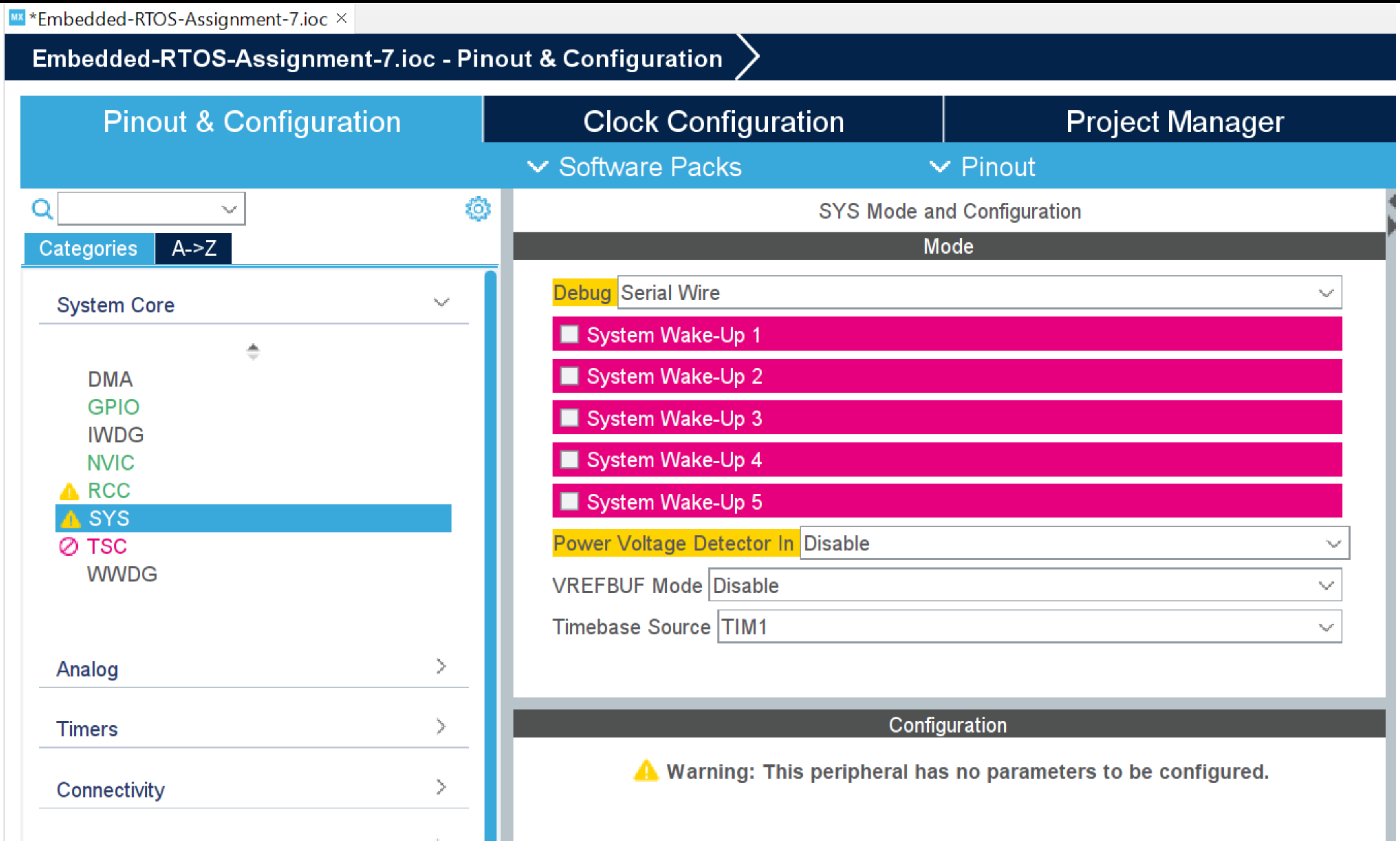


Interface CMSIS_V1

Tasks

Task Name	Priority	Stack Size...	Entry Funct...	Code Gene...	Parameter	Allocation	Buffer Name	Control Blo...
defaultTask	osPriorityN...	128	StartDefaul...	Default	NULL	Dynamic	NULL	NULL
myTask02	osPriorityIdle	128	StartTask02	Default	NULL	Dynamic	NULL	NULL

Step 9. Change time base source to TIM1



SYS Mode and Configuration

Mode

Debug Serial Wire

□ System Wake-Up 1

□ System Wake-Up 2

□ System Wake-Up 3

□ System Wake-Up 4

□ System Wake-Up 5

Power Voltage Detector In Disable

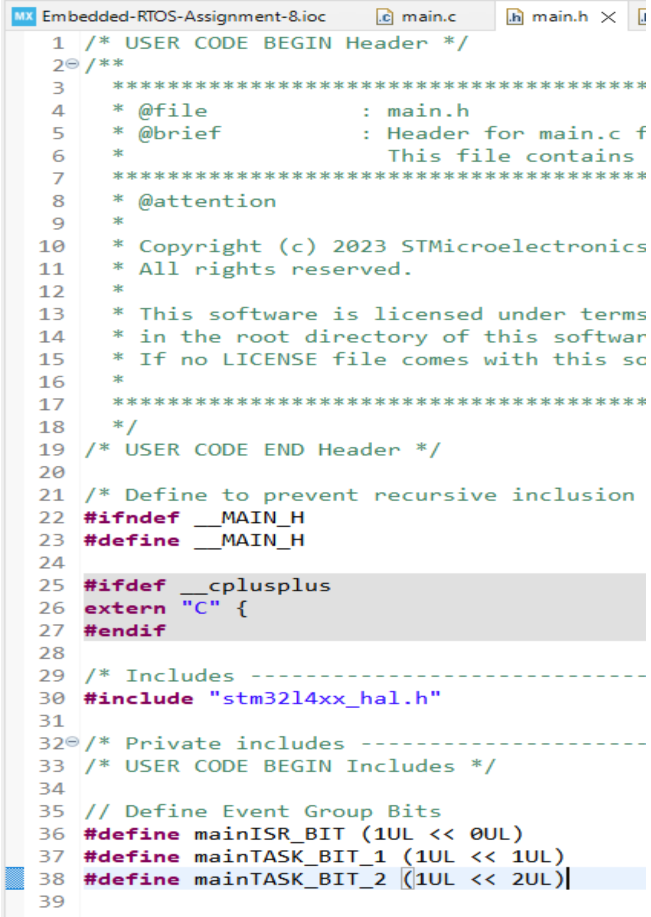
VREFBUF Mode Disable

Timebase Source TIM1

Configuration

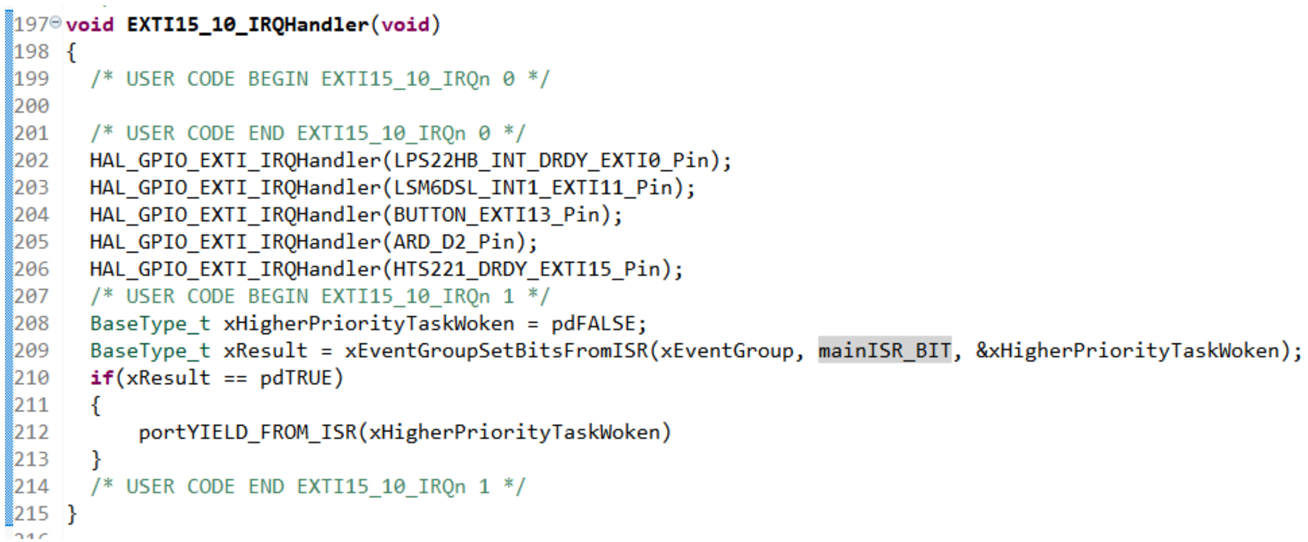
⚠ Warning: This peripheral has no parameters to be configured.

Step 10. Add event group bits in main.h file

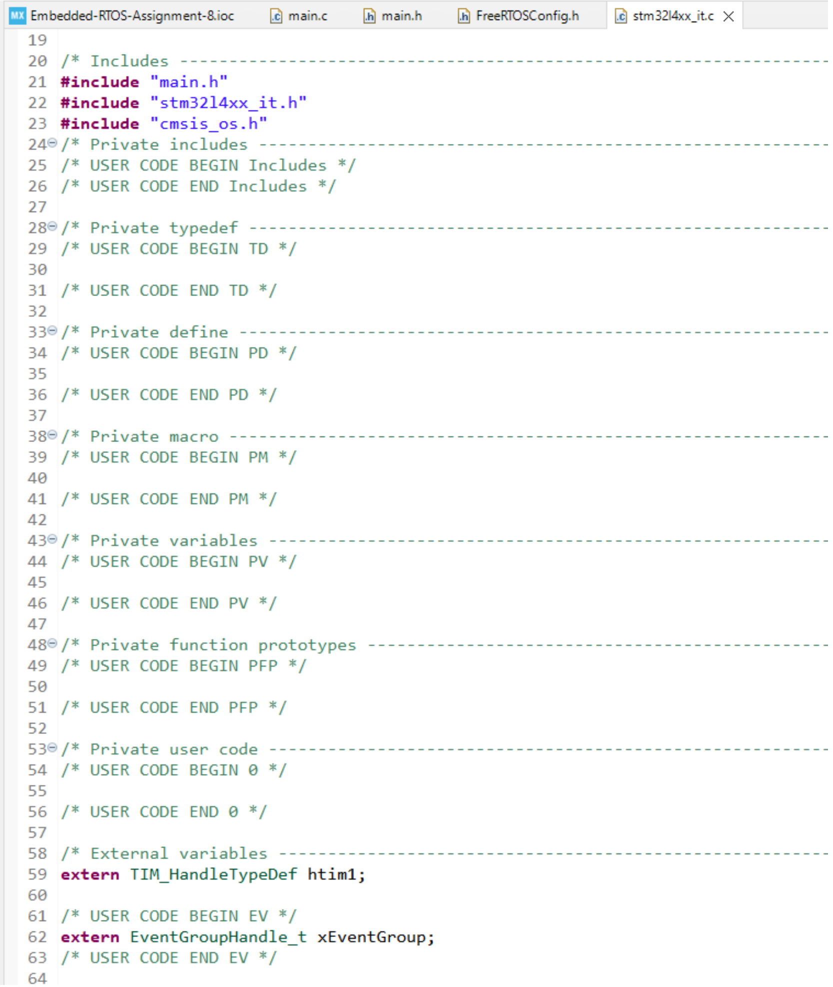


```c
/* USER CODE BEGIN Header */
/**
  ***************************************
  * @file           : main.h
  * @brief          : Header for main.c f
  *                   This file contains
  ***************************************
  * @attention
  *
  * Copyright (c) 2023 STMicroelectronics
  * All rights reserved.
  *
  * This software is licensed under terms
  * in the root directory of this softwar
  * If no LICENSE file comes with this sc
  *
  ***************************************
  */
/* USER CODE END Header */

/* Define to prevent recursive inclusion
#ifndef __MAIN_H
#define __MAIN_H

#ifdef __cplusplus
extern "C" {
#endif

/* Includes -----------------------------
#include "stm32l4xx_hal.h"

/* Private includes --------------------
/* USER CODE BEGIN Includes */

// Define Event Group Bits
#define mainISR_BIT (1UL << 0UL)
#define mainTASK_BIT_1 (1UL << 1UL)
#define mainTASK_BIT_2 (1UL << 2UL)

```

Step 11. Add interrupt code for button interrupt

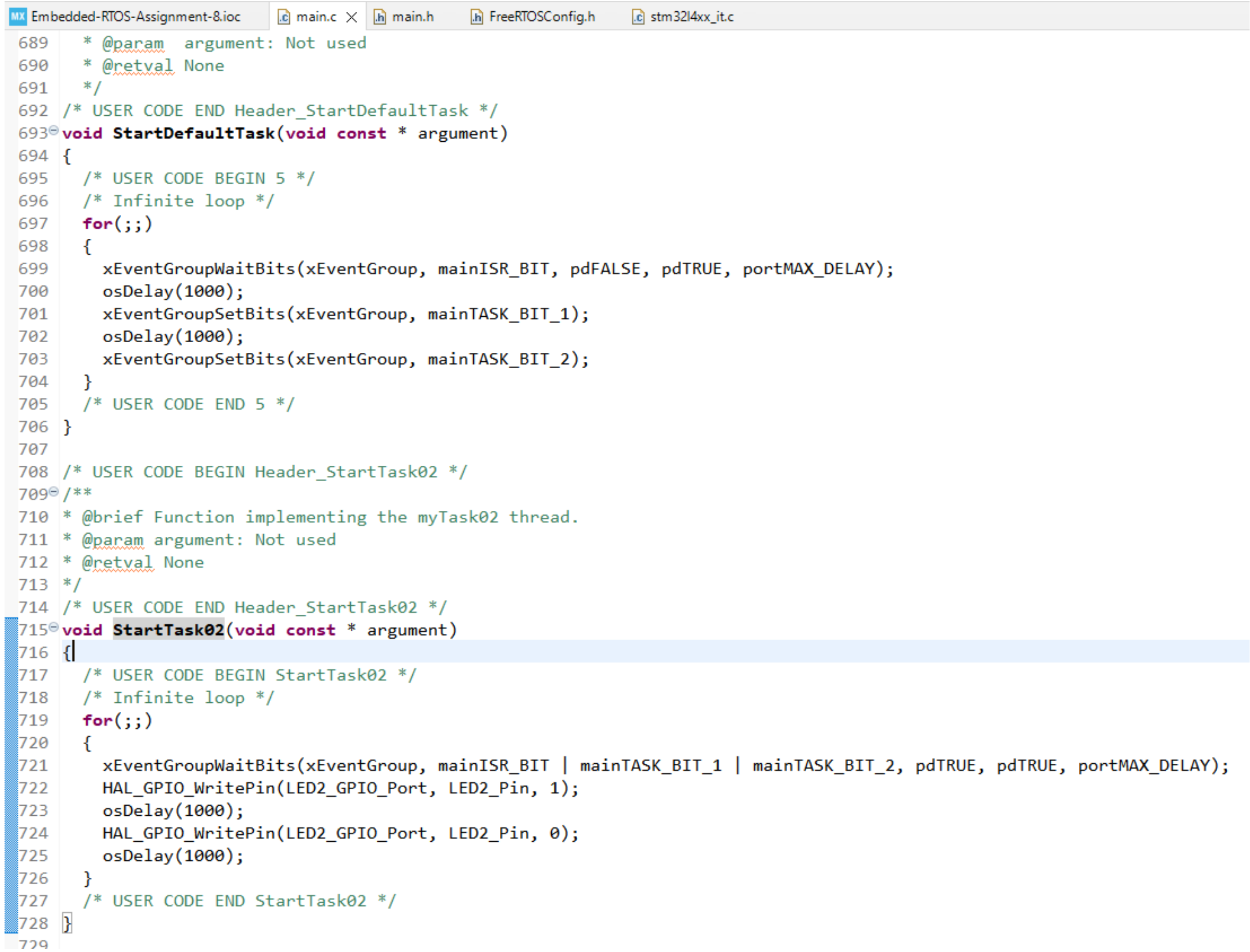


```
void EXTI15_10_IRQHandler(void)
{
  /* USER CODE BEGIN EXTI15_10_IRQn 0 */

  /* USER CODE END EXTI15_10_IRQn 0 */
  HAL_GPIO_EXTI_IRQHandler(LPS22HB_INT_DRDY_EXTI0_Pin);
  HAL_GPIO_EXTI_IRQHandler(LSM6DSL_INT1_EXTI11_Pin);
  HAL_GPIO_EXTI_IRQHandler(BUTTON_EXTI13_Pin);
  HAL_GPIO_EXTI_IRQHandler(ARD_D2_Pin);
  HAL_GPIO_EXTI_IRQHandler(HTS221_DRDY_EXTI15_Pin);
  /* USER CODE BEGIN EXTI15_10_IRQn 1 */
  BaseType_t xHigherPriorityTaskWoken = pdFALSE;
  BaseType_t xResult = xEventGroupSetBitsFromISR(xEventGroup, mainISR_BIT, &xHigherPriorityTaskWoken);
  if(xResult == pdTRUE)
  {
      portYIELD_FROM_ISR(xHigherPriorityTaskWoken)
  }
  /* USER CODE END EXTI15_10_IRQn 1 */
}
```



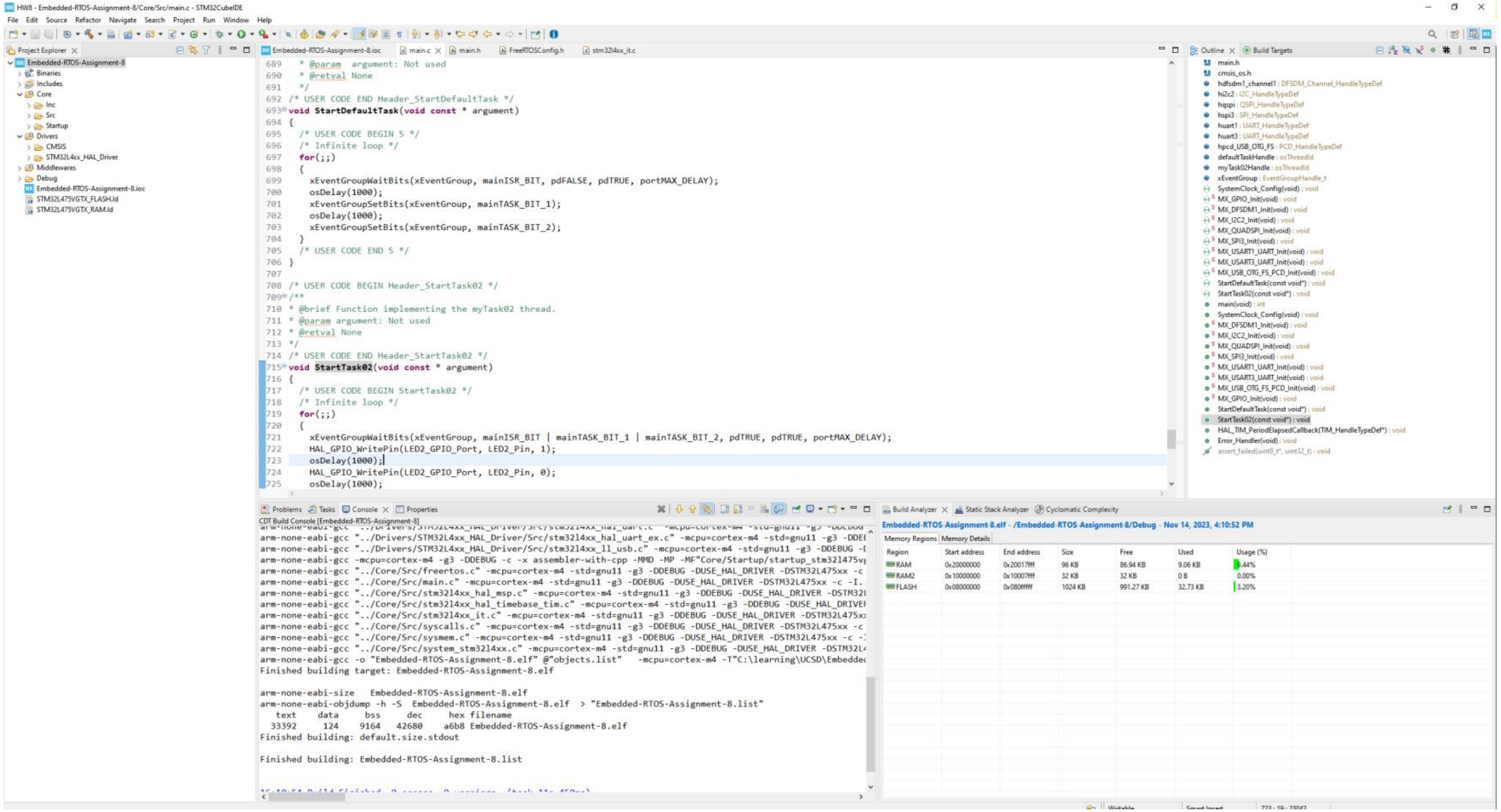
```

/* Includes ------------------------------------------------------------------*/
#include "main.h"
#include "stm32l4xx_it.h"
#include "cmsis_os.h"
/* Private includes ----------------------------------------------------------*/
/* USER CODE BEGIN Includes */
/* USER CODE END Includes */

/* Private typedef -----------------------------------------------------------*/
/* USER CODE BEGIN TD */

/* USER CODE END TD */

/* Private define ------------------------------------------------------------*/
/* USER CODE BEGIN PD */

/* USER CODE END PD */

/* Private macro -------------------------------------------------------------*/
/* USER CODE BEGIN PM */

/* USER CODE END PM */

/* Private variables ---------------------------------------------------------*/
/* USER CODE BEGIN PV */

/* USER CODE END PV */

/* Private function prototypes -----------------------------------------------*/
/* USER CODE BEGIN PFP */

/* USER CODE END PFP */

/* Private user code ---------------------------------------------------------*/
/* USER CODE BEGIN 0 */

/* USER CODE END 0 */

/* External variables --------------------------------------------------------*/
extern TIM_HandleTypeDef htim1;

/* USER CODE BEGIN EV */
extern EventGroupHandle_t xEventGroup;
/* USER CODE END EV */

```

Step 12. Code the task1 and task2



```c
  * @param  argument: Not used
  * @retval None
  */
/* USER CODE END Header_StartDefaultTask */
void StartDefaultTask(void const * argument)
{
  /* USER CODE BEGIN 5 */
  /* Infinite loop */
  for(;;)
  {
    xEventGroupWaitBits(xEventGroup, mainISR_BIT, pdFALSE, pdTRUE, portMAX_DELAY);
    osDelay(1000);
    xEventGroupSetBits(xEventGroup, mainTASK_BIT_1);
    osDelay(1000);
    xEventGroupSetBits(xEventGroup, mainTASK_BIT_2);
  }
  /* USER CODE END 5 */
}

/* USER CODE BEGIN Header_StartTask02 */
/**
* @brief Function implementing the myTask02 thread.
* @param argument: Not used
* @retval None
*/
/* USER CODE END Header_StartTask02 */
void StartTask02(void const * argument)
{
  /* USER CODE BEGIN StartTask02 */
  /* Infinite loop */
  for(;;)
  {
    xEventGroupWaitBits(xEventGroup, mainISR_BIT | mainTASK_BIT_1 | mainTASK_BIT_2, pdTRUE, pdTRUE, portMAX_DELAY);
    HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, 1);
    osDelay(1000);
    HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, 0);
    osDelay(1000);
  }
  /* USER CODE END StartTask02 */
}
```

Step 13. Build and run the code, test is successful



```c
   * @param  argument: Not used
   * @retval None
   */
/* USER CODE END Header_StartDefaultTask */
void StartDefaultTask(void const * argument)
{
  /* USER CODE BEGIN 5 */
  /* Infinite loop */
  for(;;)
  {
    xEventGroupWaitBits(xEventGroup, mainISR_BIT, pdFALSE, pdTRUE, portMAX_DELAY);
    osDelay(1000);
    xEventGroupSetBits(xEventGroup, mainTASK_BIT_1);
    osDelay(1000);
    xEventGroupSetBits(xEventGroup, mainTASK_BIT_2);
  }
  /* USER CODE END 5 */
}

/* USER CODE BEGIN Header_StartTask02 */
/**
* @brief Function implementing the myTask02 thread.
* @param argument: Not used
* @retval None
*/
/* USER CODE END Header_StartTask02 */
void StartTask02(void const * argument)
{
  /* USER CODE BEGIN StartTask02 */
  /* Infinite loop */
  for(;;)
  {
    xEventGroupWaitBits(xEventGroup, mainISR_BIT | mainTASK_BIT_1 | mainTASK_BIT_2, pdTRUE, pdTRUE, portMAX_DELAY);
    HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, 1);
    osDelay(1000);
    HAL_GPIO_WritePin(LED2_GPIO_Port, LED2_Pin, 0);
    osDelay(1000);
```

```
arm-none-eabi-gcc "../Drivers/STM32L4xx_HAL_Driver/Src/stm32l4xx_hal_uart_ex.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEI
arm-none-eabi-gcc "../Drivers/STM32L4xx_HAL_Driver/Src/stm32l4xx_ll_usb.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -I
arm-none-eabi-gcc -mcpu=cortex-m4 -g3 -DDEBUG -c -x assembler-with-cpp -MMD -MP -MF"Core/Startup/startup_stm32l475v
arm-none-eabi-gcc "../Core/Src/freertos.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L475xx -c
arm-none-eabi-gcc "../Core/Src/main.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L475xx -c -I..
arm-none-eabi-gcc "../Core/Src/stm32l4xx_hal_msp.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L
arm-none-eabi-gcc "../Core/Src/stm32l4xx_hal_timebase_tim.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVEI
arm-none-eabi-gcc "../Core/Src/stm32l4xx_it.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L475x
arm-none-eabi-gcc "../Core/Src/syscalls.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L475xx -c
arm-none-eabi-gcc "../Core/Src/sysmem.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L475xx -c -
arm-none-eabi-gcc "../Core/Src/system_stm32l4xx.c" -mcpu=cortex-m4 -std=gnu11 -g3 -DDEBUG -DUSE_HAL_DRIVER -DSTM32L4
arm-none-eabi-gcc -o "Embedded-RTOS-Assignment-8.elf" @"objects.list"   -mcpu=cortex-m4 -T"C:\learning\UCSD\Embedde
Finished building target: Embedded-RTOS-Assignment-8.elf

arm-none-eabi-size   Embedded-RTOS-Assignment-8.elf
arm-none-eabi-objdump -h -S  Embedded-RTOS-Assignment-8.elf  > "Embedded-RTOS-Assignment-8.list"
   text    data     bss     dec     hex filename
  33392     124    9164   42680    a6b8 Embedded-RTOS-Assignment-8.elf
Finished building: default.size.stdout

Finished building: Embedded-RTOS-Assignment-8.list
```

Embedded-RTOS-Assignment-8.elf - /Embedded-RTOS-Assignment-8/Debug - Nov 14, 2023, 4:10:52 PM

Region	Start address	End address	Size	Free	Used	Usage (%)
RAM	0x20000000	0x20017fff	96 KB	86.94 KB	9.06 KB	9.44%
RAM2	0x10000000	0x10007fff	32 KB	32 KB	0 B	0.00%
FLASH	0x08000000	0x080fffff	1024 KB	991.27 KB	32.73 KB	3.20%