

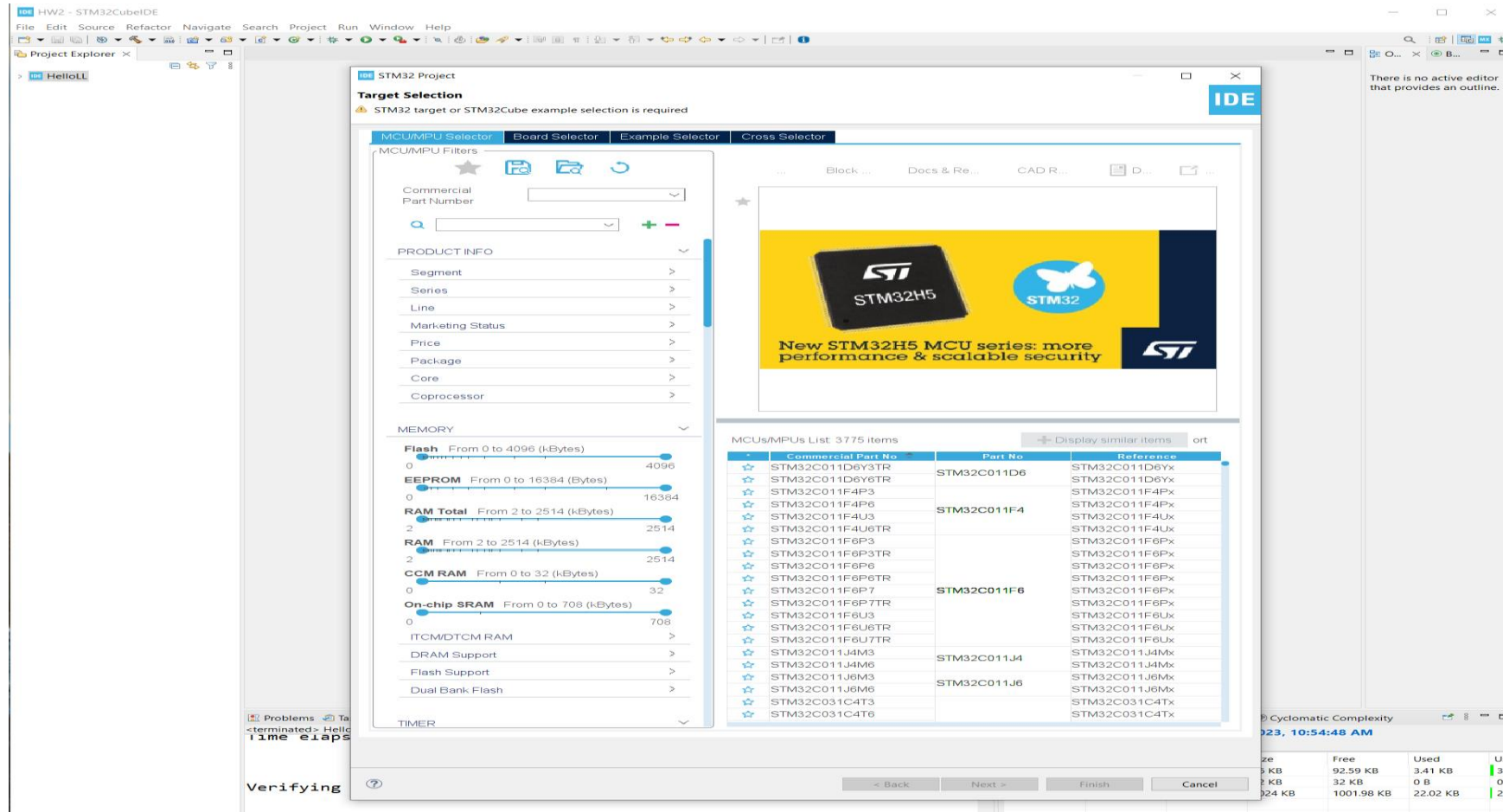
# UCSD Embedded RTOS Assignment 3

By

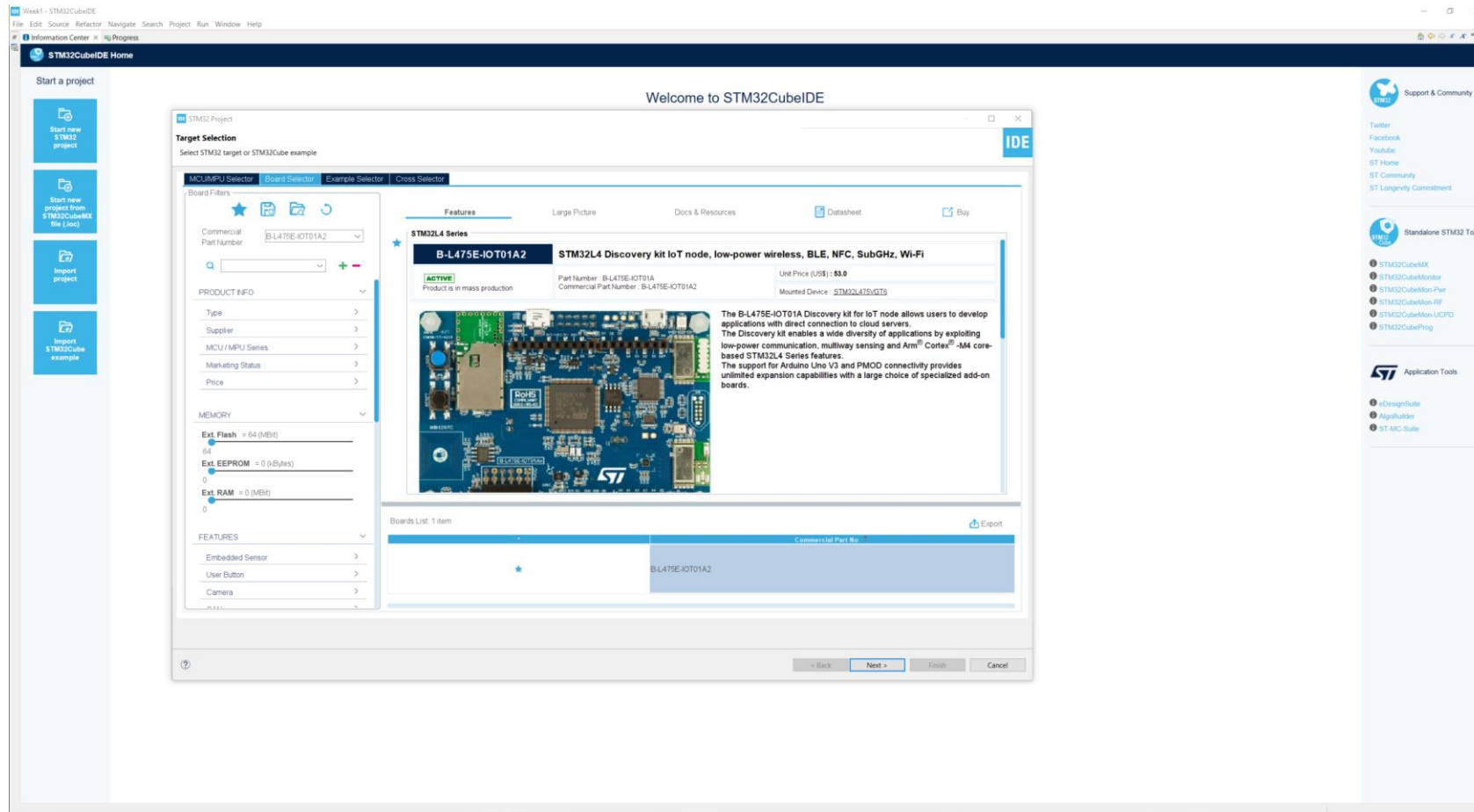
Hsuankai Chang

[hsuankac@umich.edu](mailto:hsuankac@umich.edu)

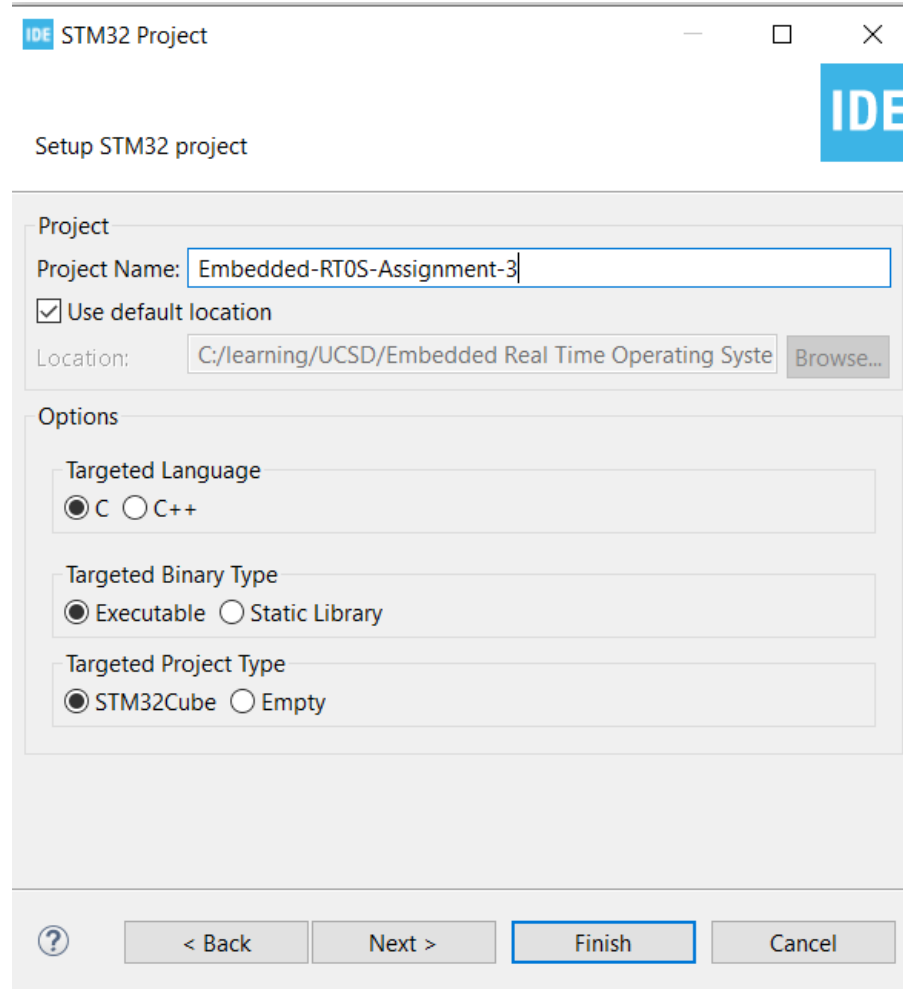
## Step 1. Startup STM32CubeIDE and create new STM32 project



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



Step 3. Enter the project name then click Next



The image shows a 'Setup STM32 project' dialog box from the IDE. The window title is 'IDE STM32 Project'. The dialog is titled 'Setup STM32 project'. It has two main sections: 'Project' and 'Options'. In the 'Project' section, the 'Project Name' field contains 'Embedded-RTOS-Assignment-3'. Below it, the 'Use default location' checkbox is checked. The 'Location' field shows 'C:/learning/UCSD/Embedded Real Time Operating System' with a 'Browse...' button next to it. The 'Options' section contains three groups of radio buttons: 'Targeted Language' with 'C' selected, 'Targeted Binary Type' with 'Executable' selected, and 'Targeted Project Type' with 'STM32Cube' selected. At the bottom, there are four buttons: a help button (question mark), '< Back', 'Next >', and 'Finish' (which is highlighted with a blue border). A 'Cancel' button is also present.

IDE STM32 Project

Setup STM32 project

**Project**

Project Name: Embedded-RTOS-Assignment-3

☒ Use default location

Location: C:/learning/UCSD/Embedded Real Time Operating System Browse...

**Options**

Targeted Language

☒ C ☐ C++

Targeted Binary Type

☒ Executable ☐ Static Library

Targeted Project Type

☒ STM32Cube ☐ Empty

? < Back Next > Finish Cancel

Step 4. See the firmware package name and version



The image shows a screenshot of the 'Firmware Library Package Setup' dialog box in the STM32 Project IDE. The dialog box has a title bar with the IDE logo and the text 'STM32 Project'. The main title is 'Firmware Library Package Setup' and the subtitle is 'Setup STM32 target's firmware'. The dialog is divided into three sections: 'Target and Firmware Package', 'Firmware and Software Package Repository', and 'Code Generator Options'. In the 'Target and Firmware Package' section, the 'Target Reference' is 'B-L475E-IOT01A2' and the 'Firmware Package Name and Version' is 'STM32Cube FW\_L4 V1.17.2'. In the 'Firmware and Software Package Repository' section, the 'Location' is 'C:\Users\hsuankai.chang\STM32Cube\Repository' and there is a link to 'Firmware Updater'. In the 'Code Generator Options' section, there are three radio buttons: 'Add necessary library files as reference in the toolchain project configuration file', 'Copy all used libraries into the project folder', and 'Copy only the necessary library files'. The 'Finish' button is highlighted with a blue border.

IDE 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

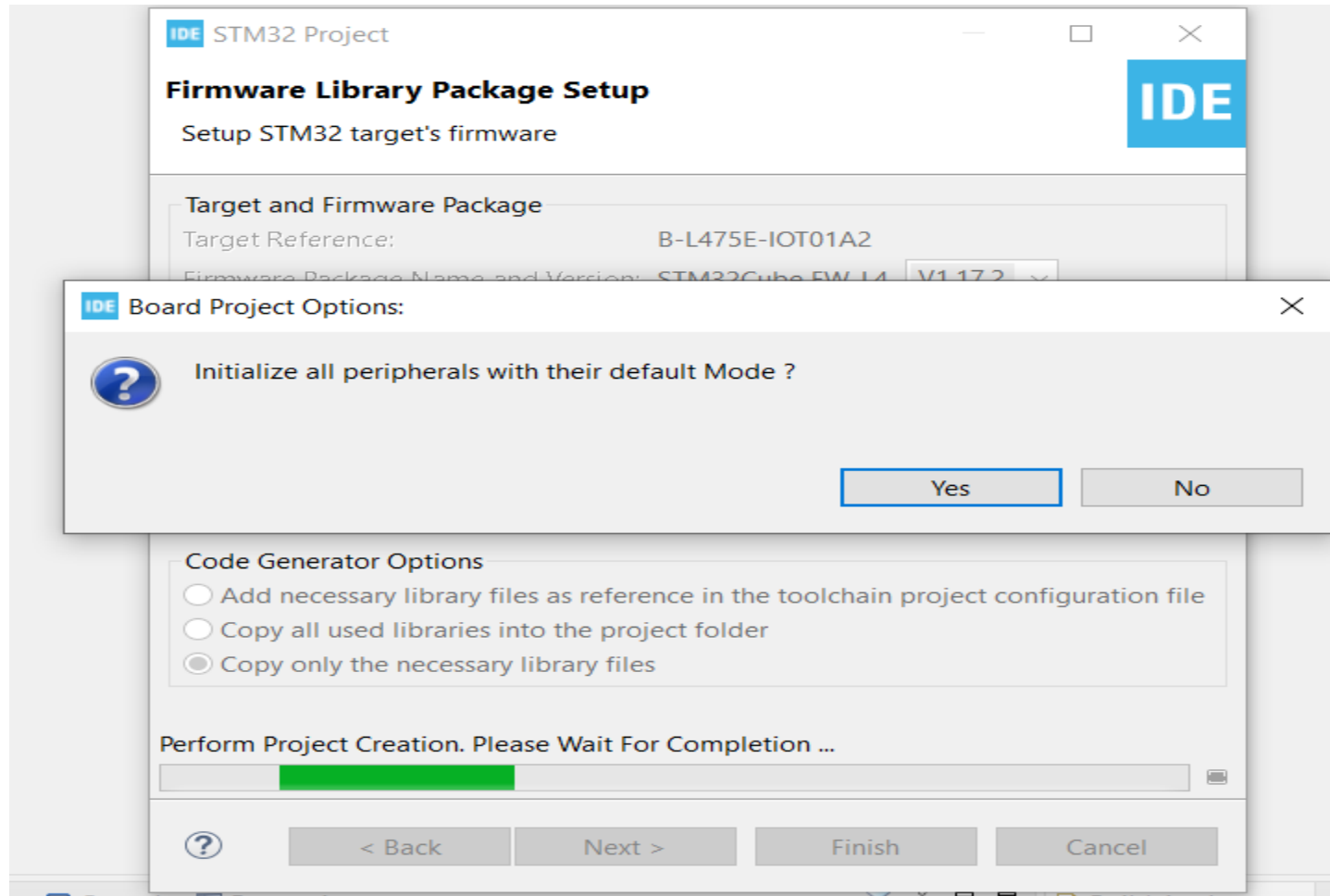
☐ Add necessary library files as reference in the toolchain project configuration file

☐ Copy all used libraries into the project folder

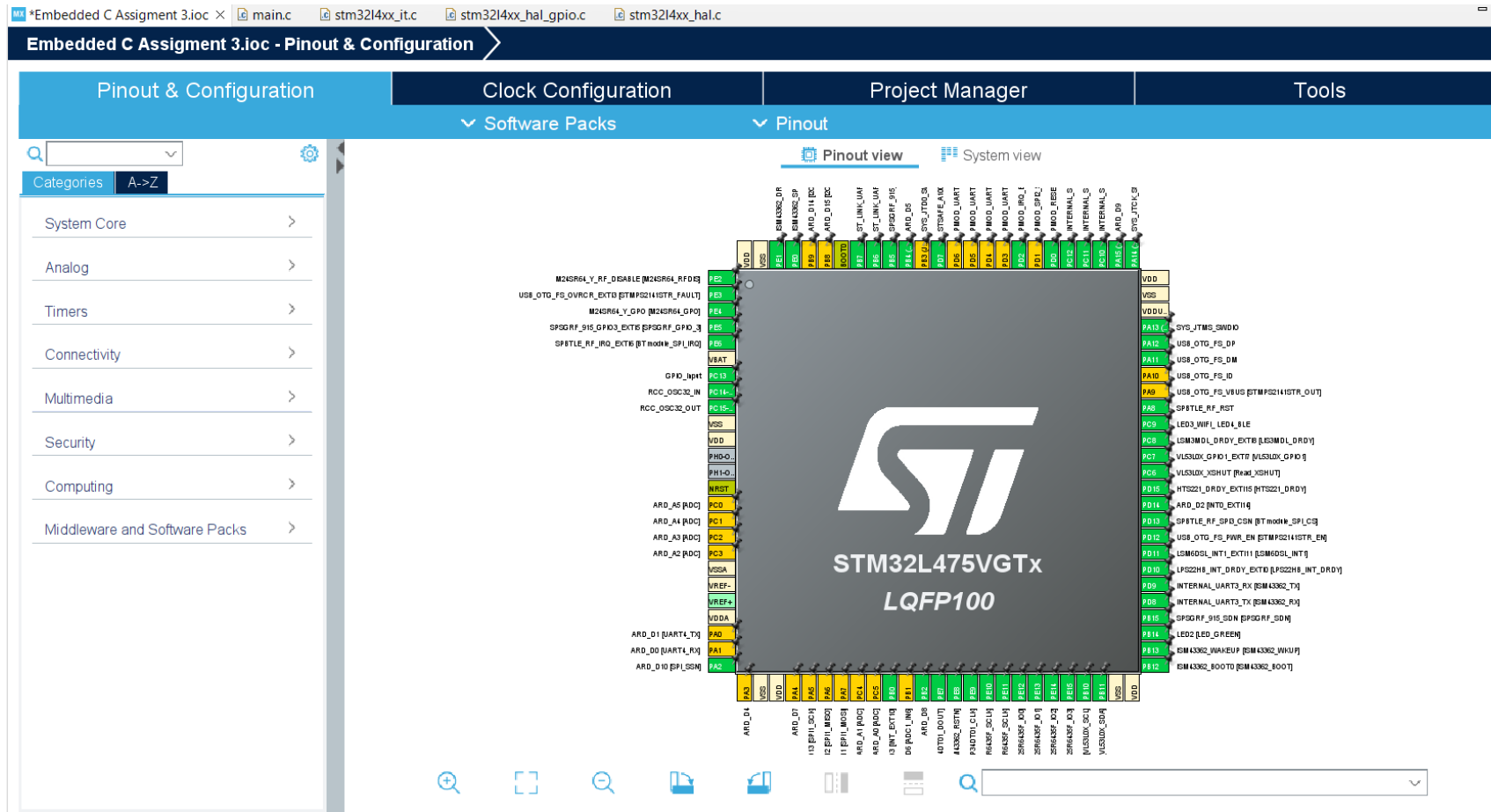
☒ Copy only the necessary library files

? < Back Next > Finish Cancel

Step 5. Click yes to initialize all peripherals to default



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



Step 7. Enable the CMSIS\_V1 RTOS, and add two more tasks. Set different priority to test

Embedded-RTOS-Assignment-3.ioc - Pinout & Configuration

Pinout & Configuration | Clock Configuration | Project Manager | Tools

Software Packs | Pinout

FREERTOS Mode and Configuration

Interface: CMSIS\_V1

Configuration

Reset Configuration

Tasks and Queues | Timers and Semaphores | Mutexes | Events | FreeRTOS Heap Usage

Config parameters | Include parameters | Advanced settings | User Constants

Tasks

Task Name	Priority	Stack	Code Gen.	Parameter	Allocation	Buffer Name	Control Blo.
defaultTask	osPriorityNormal	128	...	Default	NULL	Dynamic	NULL
myTask02	osPriorityLow	128	...	Default	NULL	Dynamic	NULL
myTask03	osPriorityIdle	128	...	Default	NULL	Dynamic	NULL

Queues

Queue Name	Queue Size	Item Size	Allocation	Buffer Name	Control Block Na.
------------	------------	-----------	------------	-------------	-------------------

Pinout view | System view

STM32L475VGTx LQFP100

https://sw-center.st.compositeArtifacts.xml



## Step 8. Change Timebase from systick to TIM1

\*Embedded-RTOS-Assignment-1.ioc X

### Embedded-RTOS-Assignment-1.ioc - Pinout & Configuration

#### Pinout & Configuration

Categories: A-Z

System Core

- DMA
- GPIO
- IWDG
- NVIC
- ⚠ RCC
- ⚠ **SYS**
- ⊗ TSC
- WWDG

Analog >

Timers >

Connectivity >

Multimedia >

Security >

Computing >

Middleware and... >

#### Software Packs

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

VREBUF Mode Disable

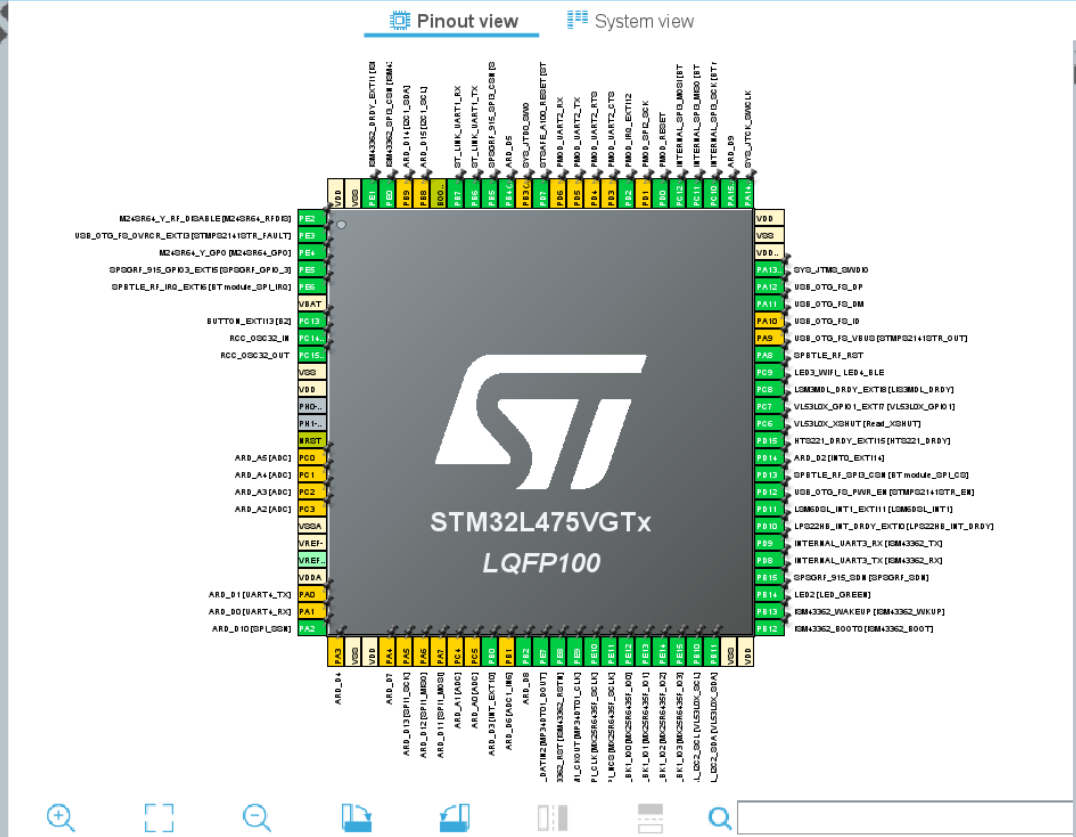
Timebase Source TIM1

Configuration

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

#### Pinout

Pinout view System view



Step 9. In your default task, blink the LED2 every 250 ms

```
MX Embedded-RTOS-Assignment-3.ioc  main.c ×
690
691 /* USER CODE END 4 */
692
693 /* USER CODE BEGIN Header_StartDefaultTask */
694 /**
695  * @brief Function implementing the defaultTask threa
696  * @param argument: Not used
697  * @retval None
698  */
699 /* USER CODE END Header_StartDefaultTask */
700 void StartDefaultTask(void const * argument)
701 {
702     /* USER CODE BEGIN 5 */
703     /* Infinite loop */
704     for(;;)
705     {
706         HAL_GPIO_TogglePin(LED2_GPIO_Port, LED2_Pin);
707         osDelay(250);
708     }
709     /* USER CODE END 5 */
710 }
```

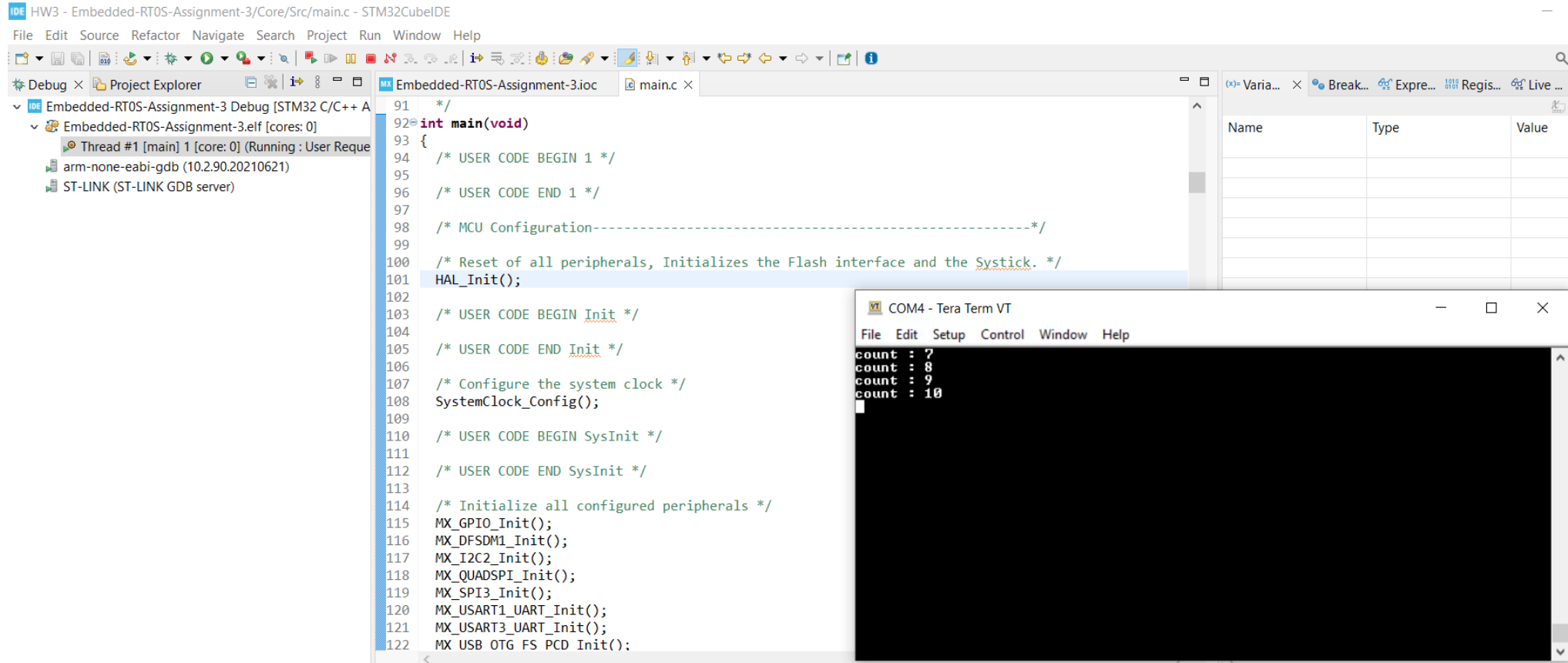
Step 10. In task 2, blinks LED3\_WIFI\_LED4\_BLE every 1000 msec

```
MX Embedded-RTOS-Assignment-3.ioc  main.c ×
712 /* USER CODE BEGIN Header_StartTask02 */
713 /**
714  * @brief Function implementing the myTask02 thread.
715  * @param argument: Not used
716  * @retval None
717  */
718 /* USER CODE END Header_StartTask02 */
719 void StartTask02(void const * argument)
720 {
721     /* USER CODE BEGIN StartTask02 */
722     /* Infinite loop */
723     for(;;)
724     {
725         HAL_GPIO_TogglePin(LED3_WIFI__LED4_BLE_GPIO_Port, LED3_WIFI__LED4_BLE_Pin);
726         osDelay(1000);
727     }
728     /* USER CODE END StartTask02 */
729 }
730
```

Step 11. Create "Task 3" that displays a counter (e.g. "count = 1", "count = 2", etc.) on the STM32 Virtual Console.

```
MX Embedded-RTOS-Assignment-3.ioc  main.c ×
730
731 /* USER CODE BEGIN Header_StartTask03 */
732 /**
733  * @brief Function implementing the myTask03 thread.
734  * @param argument: Not used
735  * @retval None
736  */
737 /* USER CODE END Header_StartTask03 */
738 void StartTask03(void const * argument)
739 {
740     /* USER CODE BEGIN StartTask03 */
741     int count = 0;
742     /* Infinite loop */
743     for(;;)
744     {
745         count++;
746         char buf[100];
747         snprintf(buf, sizeof(buf), "count : %d\n\r", count);
748         HAL_UART_Transmit(&huart1, (uint8_t*) buf, strlen(buf), 1000);
749         osDelay(4000);
750     }
751     /* USER CODE END StartTask03 */
752 }
753
```

Step 12. Debug and run the code, you can see LED2 and LED3 blinking, and console prints out the desired message



The screenshot displays the STM32CubeIDE environment. The main editor window shows the `main.c` file with the following code:

```
91  /*  
92  int main(void)  
93  {  
94      /* USER CODE BEGIN 1 */  
95      /* USER CODE END 1 */  
96      /* MCU Configuration-----*/  
97      /* Reset of all peripherals, Initializes the Flash interface and the Systick. */  
98      HAL_Init();  
99      /* USER CODE BEGIN Init */  
100     /* USER CODE END Init */  
101     /* Configure the system clock */  
102     SystemClock_Config();  
103     /* USER CODE BEGIN SysInit */  
104     /* USER CODE END SysInit */  
105     /* Initialize all configured peripherals */  
106     MX_GPIO_Init();  
107     MX_DFSDM1_Init();  
108     MX_I2C2_Init();  
109     MX_QUADSPI_Init();  
110     MX_SPI3_Init();  
111     MX_USART1_UART_Init();  
112     MX_USART3_UART_Init();  
113     MX_USB_OTG_FS_PCD_Init();
```

The Project Explorer on the left shows the project structure, including the `main.c` file. The Tera Term VT window in the foreground displays the following output:

```
count : 7  
count : 8  
count : 9  
count : 10
```