7200- Embedded Systems

Dhamini Vootkuri- Name

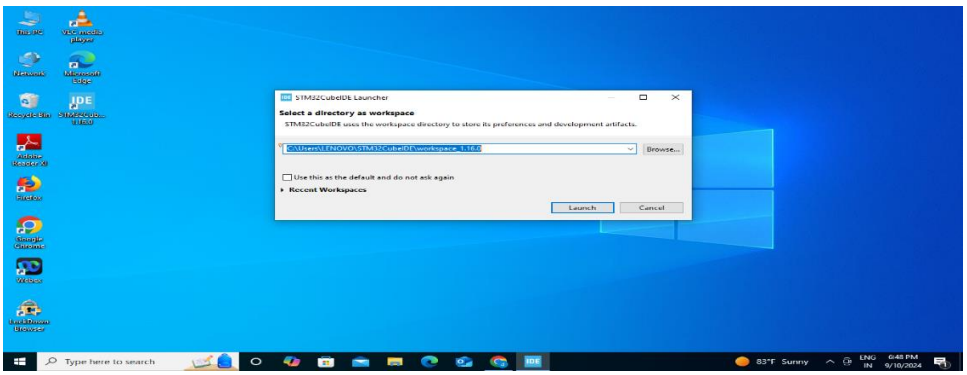
U01111434-UID

vootkuri.3@wright.edu

LAB-1- REPORT:

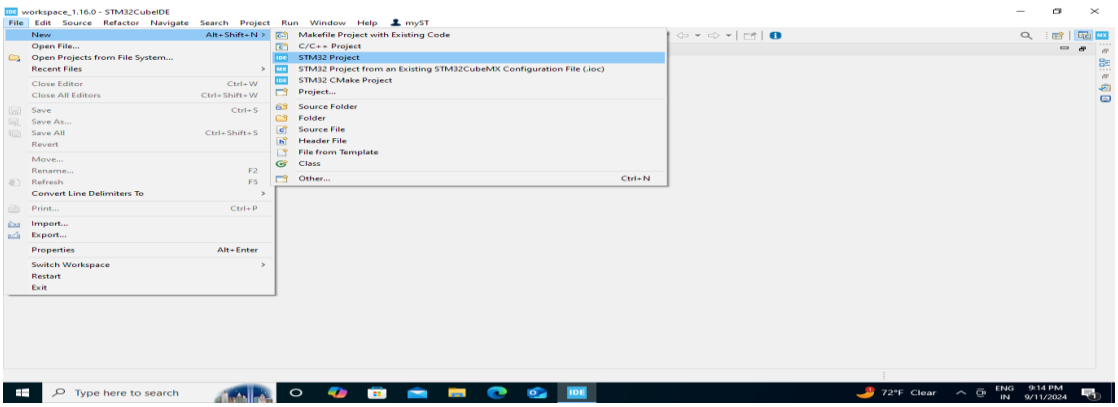
1. A. The steps are taken to set up the project, from initialization to configuration and completion.

Download the STM32CubeIDE and follow the instructions for installation of it.

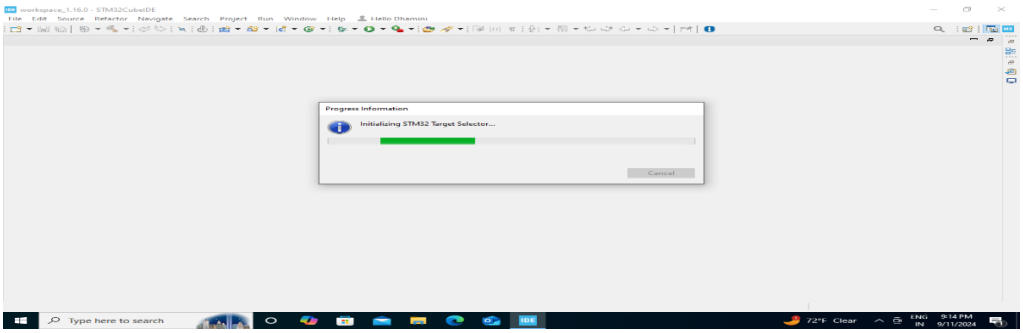


2. Include screenshots of important steps (such as configuring PA5 and the code in main.c).

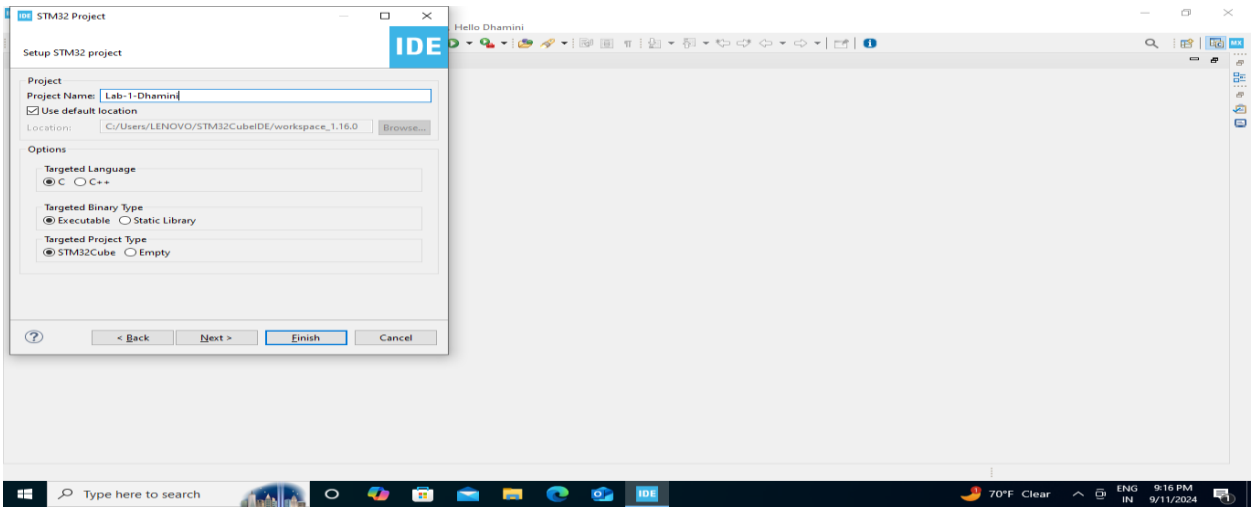
B. Showing how to create a new STM32 project, navigate to File > New > STM32 Project.



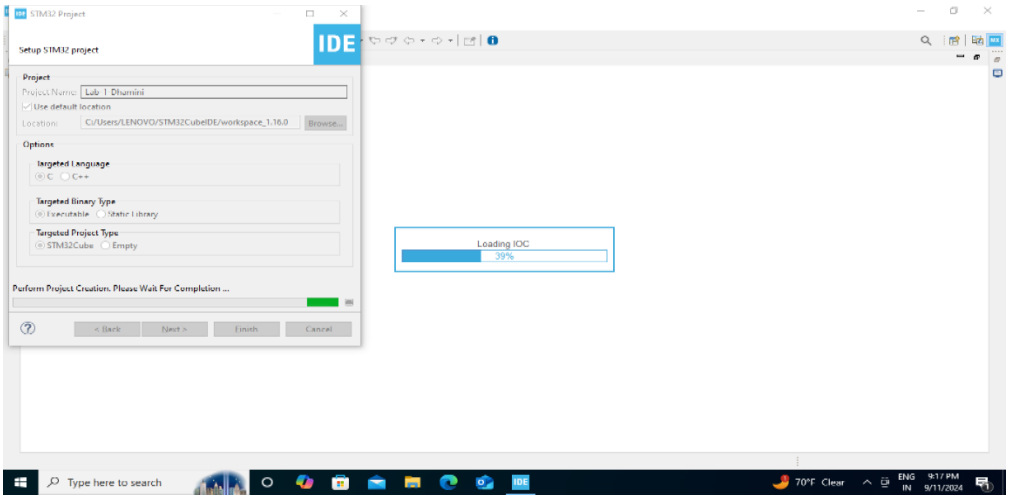
C. It gets initialized, then select the board NUCLEO-F411RE

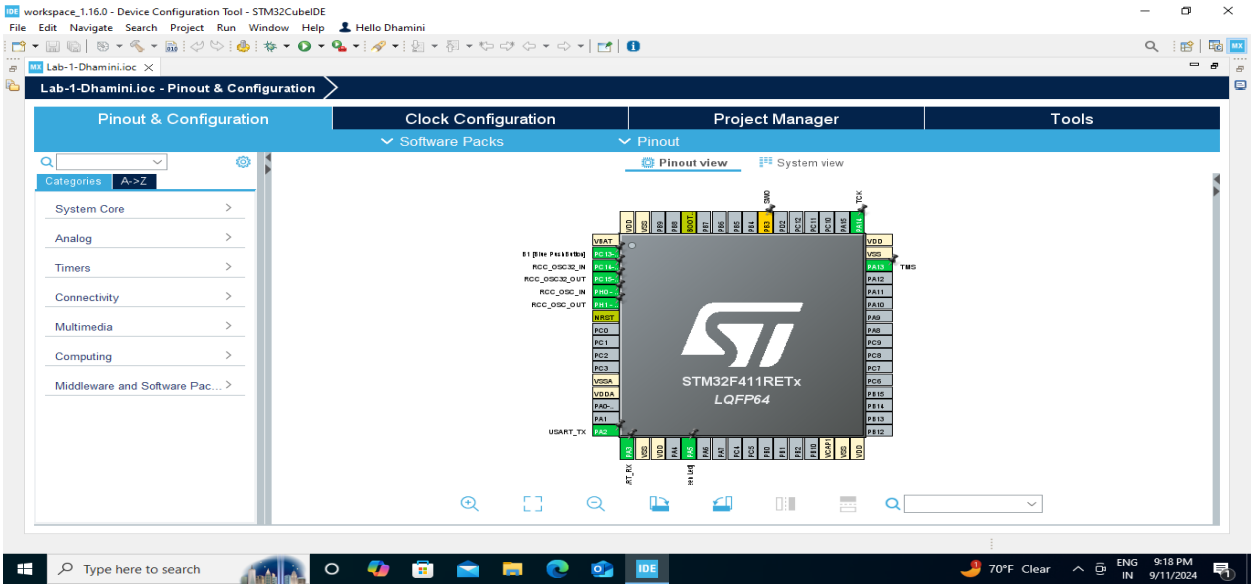


D. Next, configure the project settings, including the project name and location. I named the project "Lab-1-Dhamini."

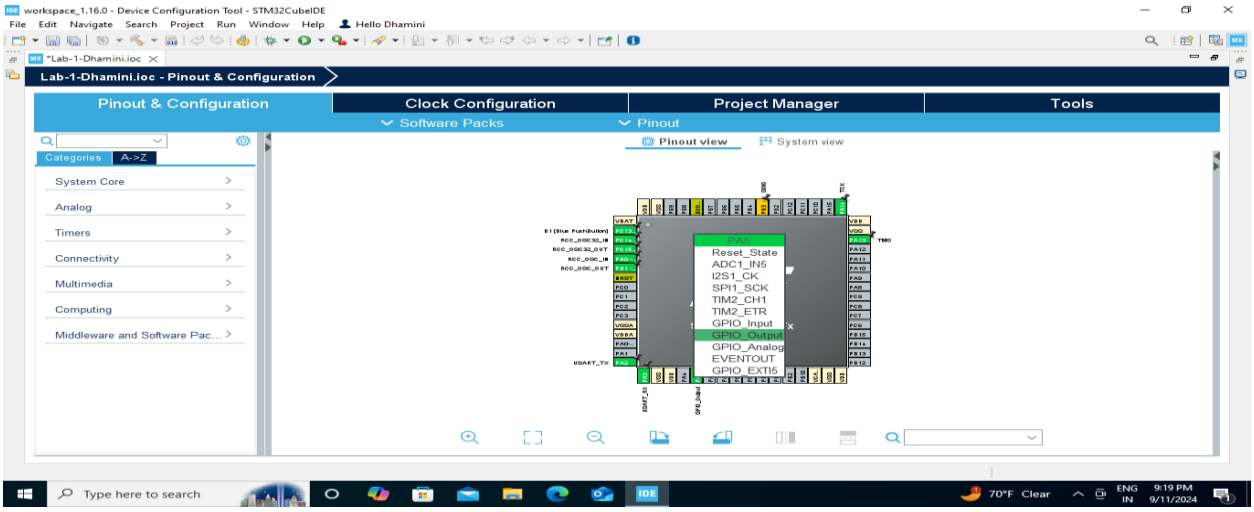


E. initializes the project and provides the pinout configuration.

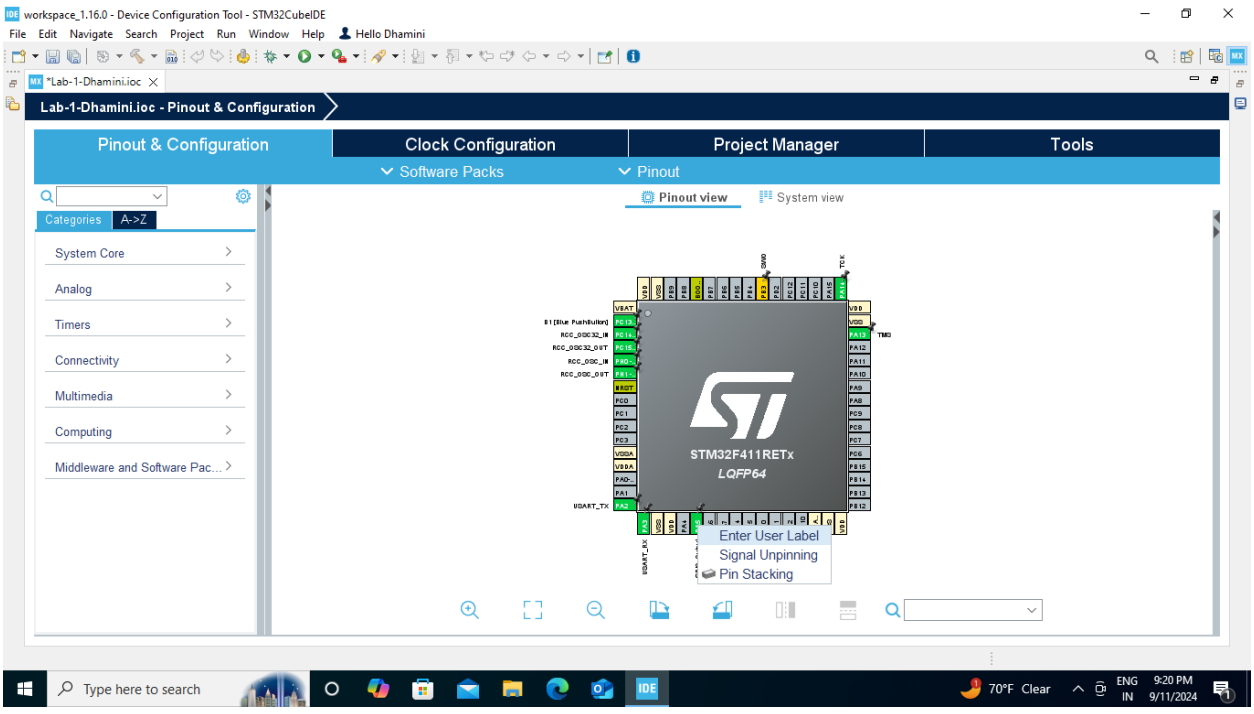




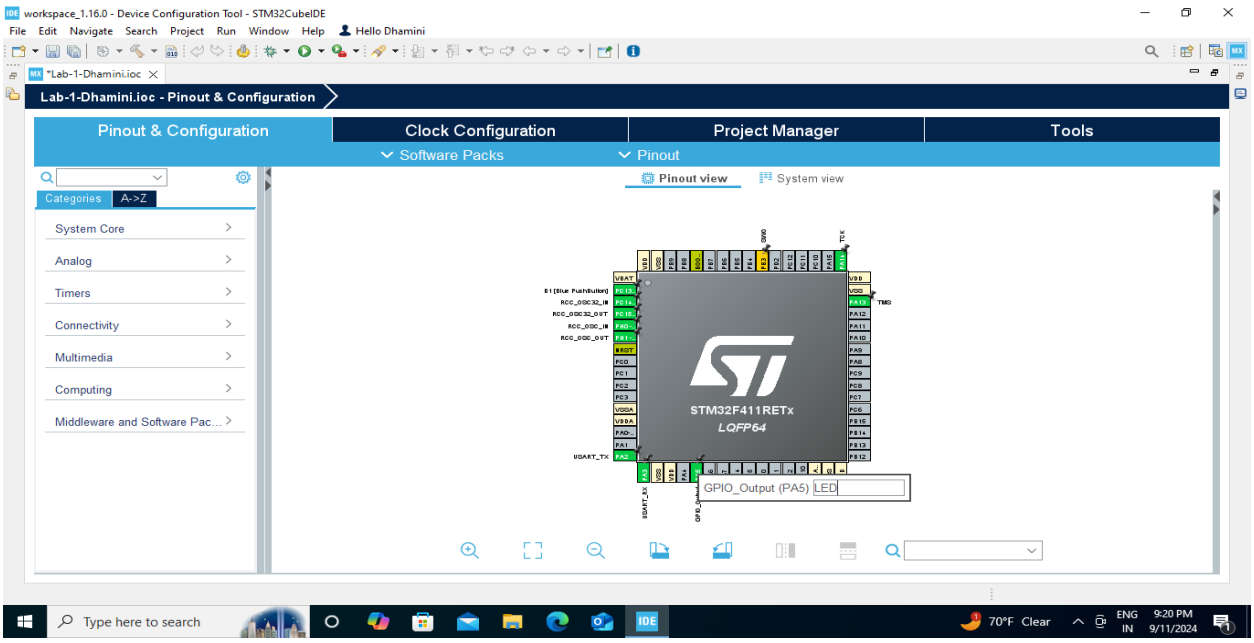
F. Make the PA5 as GPIO_Output



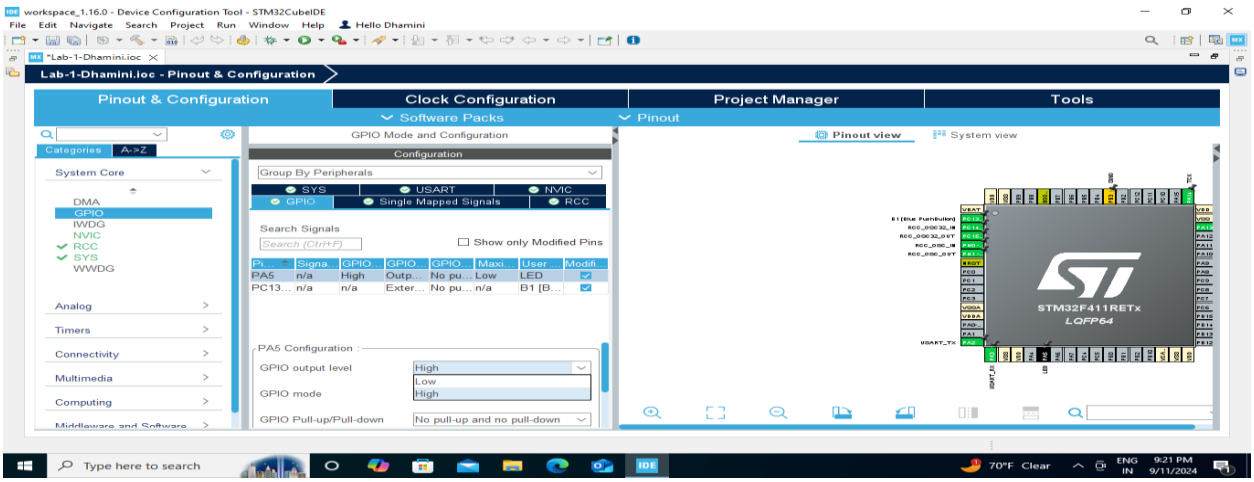
G. Give the right click after choosing GPIO_Output, then choose the option “Enter User Label”

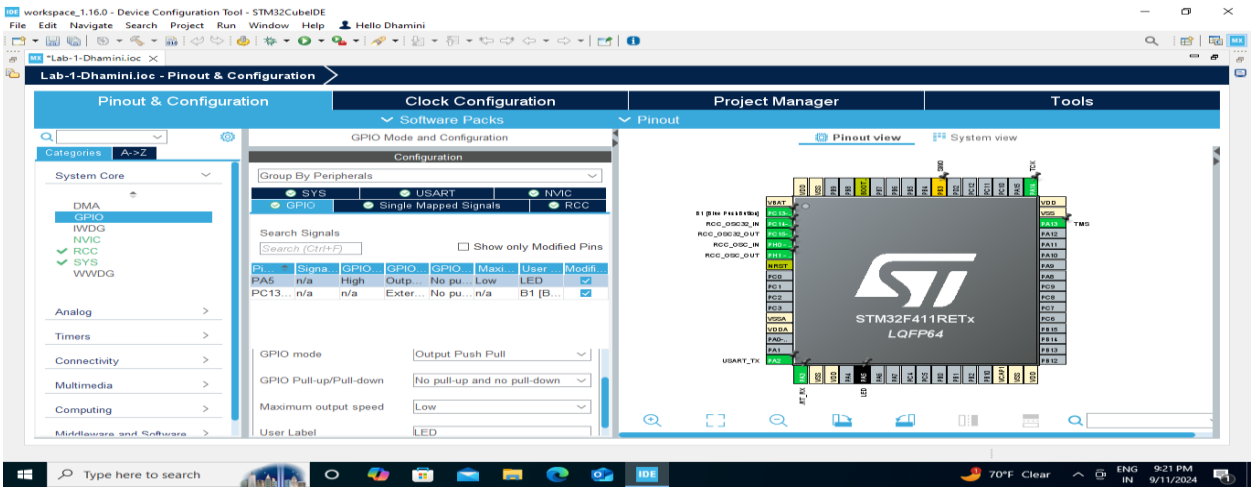


H.Name GPIO_Output(PA5) as LED



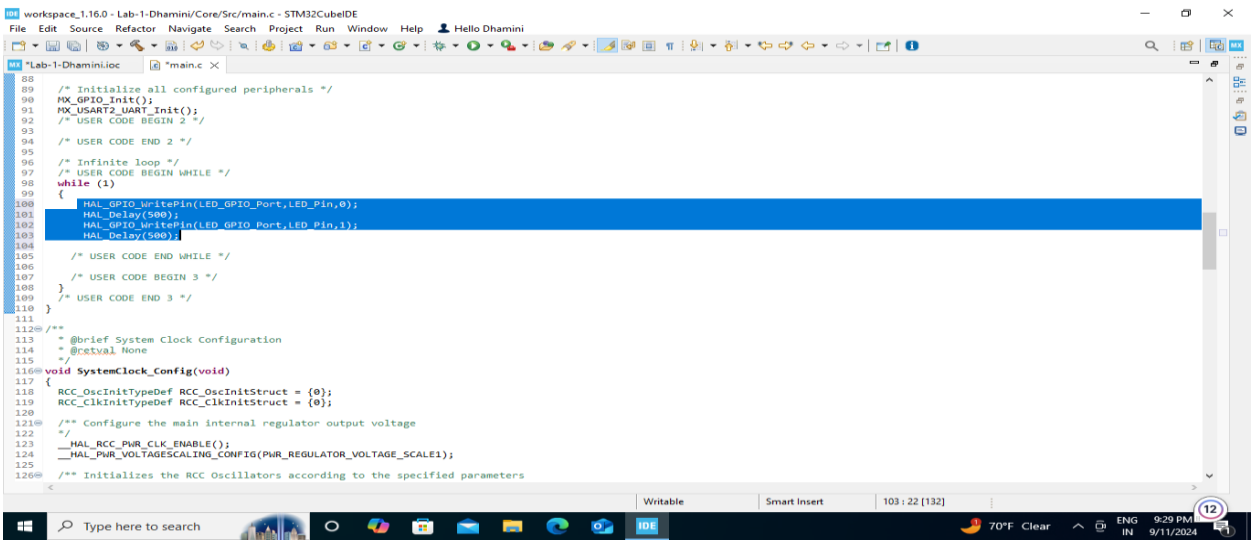
I.Identify the GPIO pin associated with GPIO_OUTPUT, which is PA5 Pin and label it as LED. You can also customize settings such as output state (low or high) and configure the pin mode (push or pull).



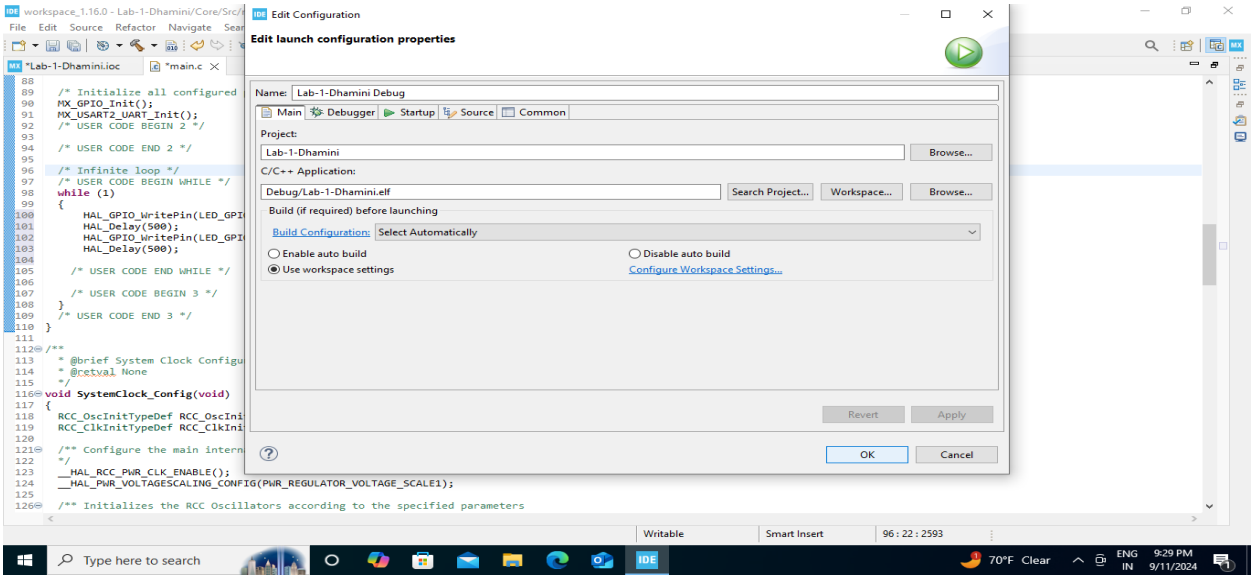


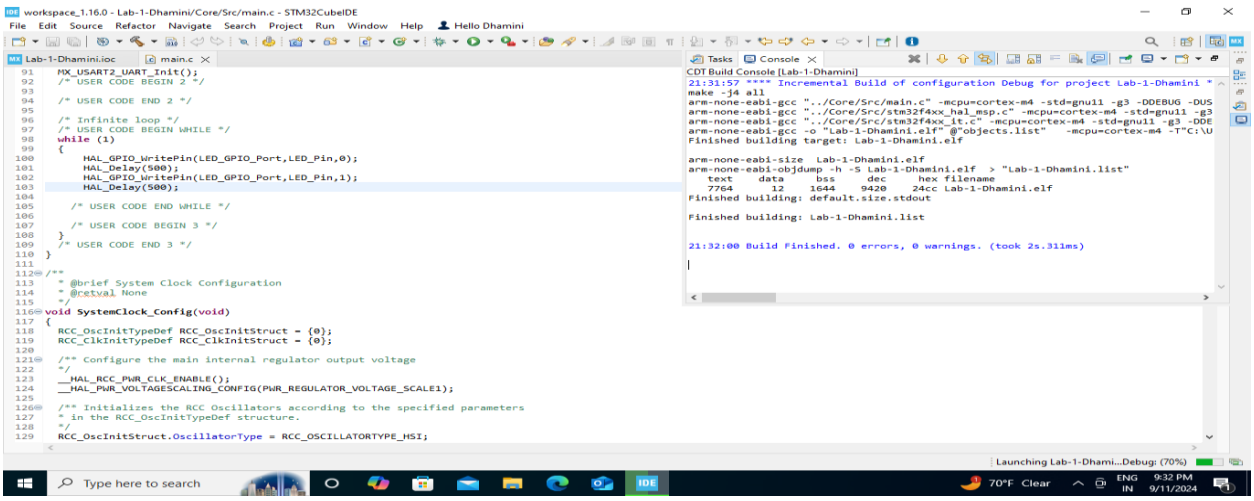
J. Open the main.c file and write the code to control the LED as shown below:

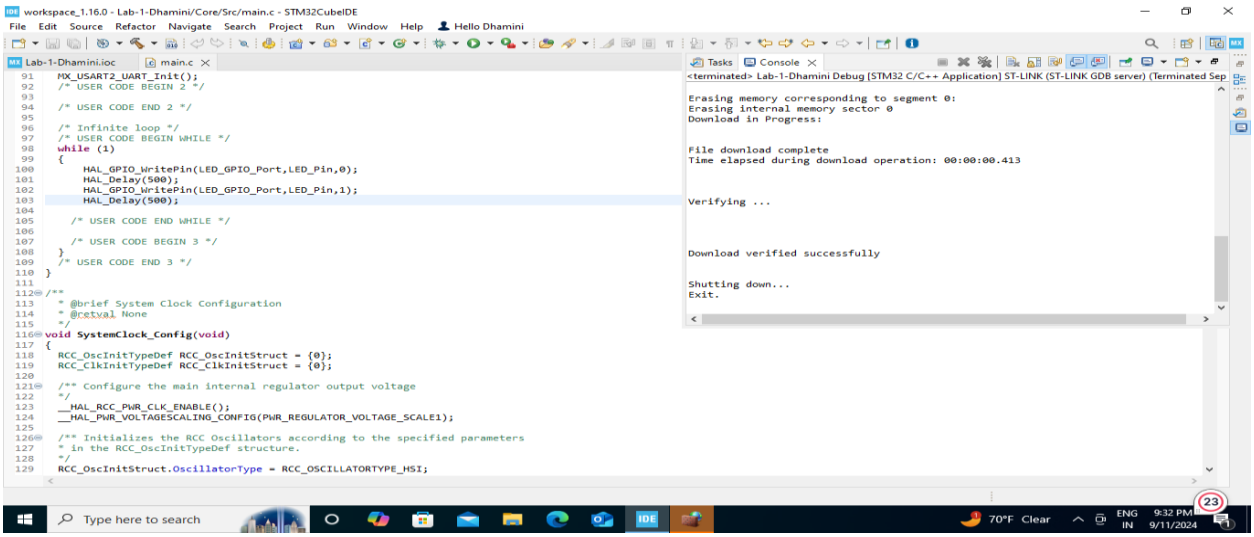
```
while (1)
{
 HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,0);
 HAL_Delay();
HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,1);
 HAL_Delay();
}
```

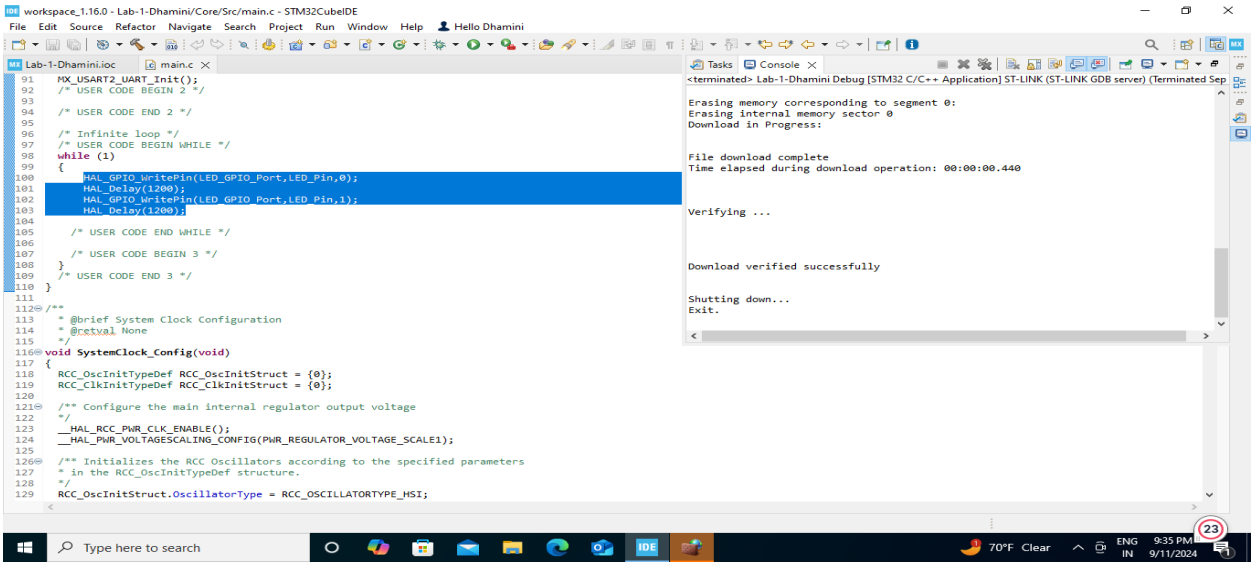


debug the project






workspace_1.16.0 - Lab-1-Dhamini/Core/Src/main.c - STM32CubeIDE
File Edit Source Refactor Navigate Search Project Run Window Help Hello Dhamini
Lab-1-Dhamini.ioc
main.c
MX_USART2_UART_Init();
/* USER CODE BEGIN 2 */
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,0);
HAL_Delay(500);
HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,1);
HAL_Delay(500);
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}
/**
* @brief System Clock Configuration
* @retval None
*/
void SystemClock_Config(void)
{
RCC_OscInitTypeDef RCC_OscInitStruct = {0};
RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
/** Configure the main internal regulator output voltage
*/
__HAL_RCC_PWR_CLK_ENABLE();
__HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
/** Initializes the RCC Oscillators according to the specified parameters
* in the RCC_OscInitTypeDef structure.
*/
RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
Tasks Console
<terminated> Lab-1-Dhamini Debug [STM32 C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Sep
Erasing memory corresponding to segment 0:
Erasing internal memory sector 0
Download in Progress:
File download complete
Time elapsed during download operation: 00:00:00.413
Verifying ...
Download verified successfully
Shutting down...
Exit.
Type here to search
70°F Clear
ENG IN
9:32 PM 9/11/2024


workspace_1.16.0 - Lab-1-Dhamini/Core/Src/main.c - STM32CubeIDE
File Edit Source Refactor Navigate Search Project Run Window Help Hello Dhamini
Lab-1-Dhamini.ioc
main.c
MX_USART2_UART_Init();
/* USER CODE BEGIN 2 */
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
while (1)
{
HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,0);
HAL_Delay(1200);
HAL_GPIO_WritePin(LED_GPIO_Port,LED_Pin,1);
HAL_Delay(1200);
/* USER CODE END WHILE */
/* USER CODE BEGIN 3 */
}
/* USER CODE END 3 */
}
/**
* @brief System Clock Configuration
* @retval None
*/
void SystemClock_Config(void)
{
RCC_OscInitTypeDef RCC_OscInitStruct = {0};
RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};
/** Configure the main internal regulator output voltage
*/
__HAL_RCC_PWR_CLK_ENABLE();
__HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
/** Initializes the RCC Oscillators according to the specified parameters
* in the RCC_OscInitTypeDef structure.
*/
RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
Tasks Console
<terminated> Lab-1-Dhamini Debug [STM32 C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Sep
Erasing memory corresponding to segment 0:
Erasing internal memory sector 0
Download in Progress:
File download complete
Time elapsed during download operation: 00:00:00.440
Verifying ...
Download verified successfully
Shutting down...
Exit.
Type here to search
70°F Clear
ENG IN
9:35 PM 9/11/2024

3. The result of compiling and executing the code.

Figure showing the outcome is the blinking of an LED for 500 Milliseconds:

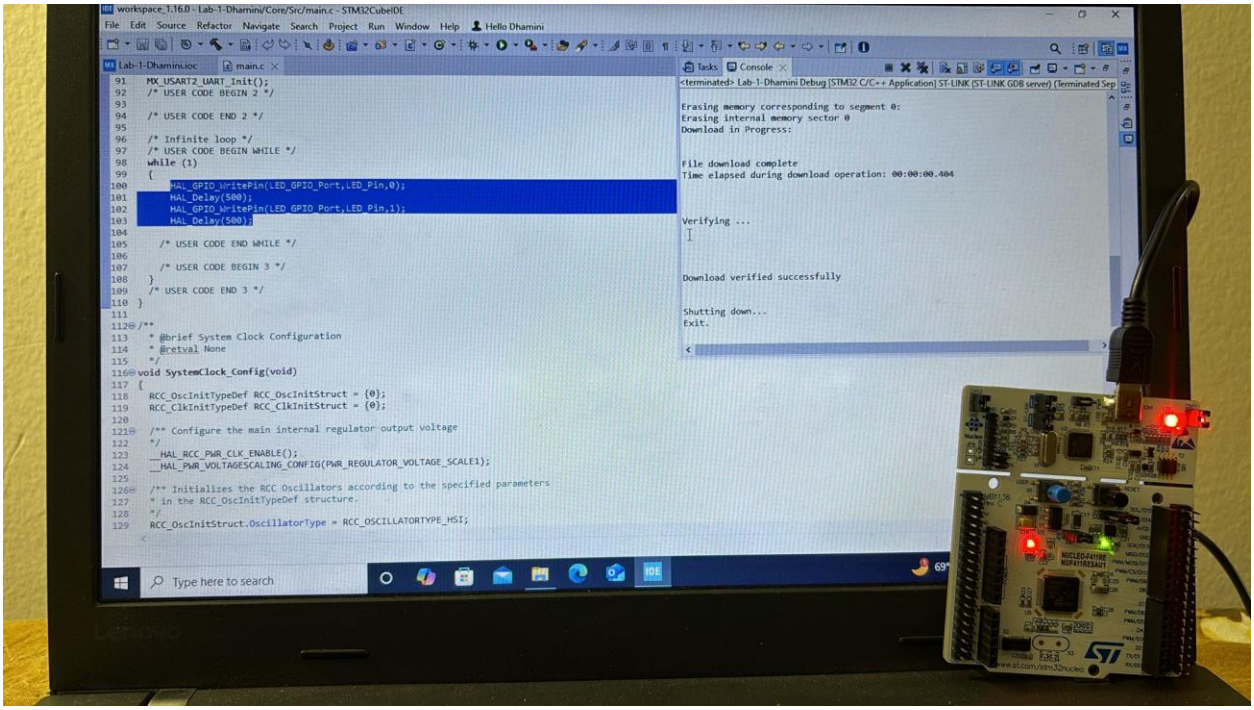
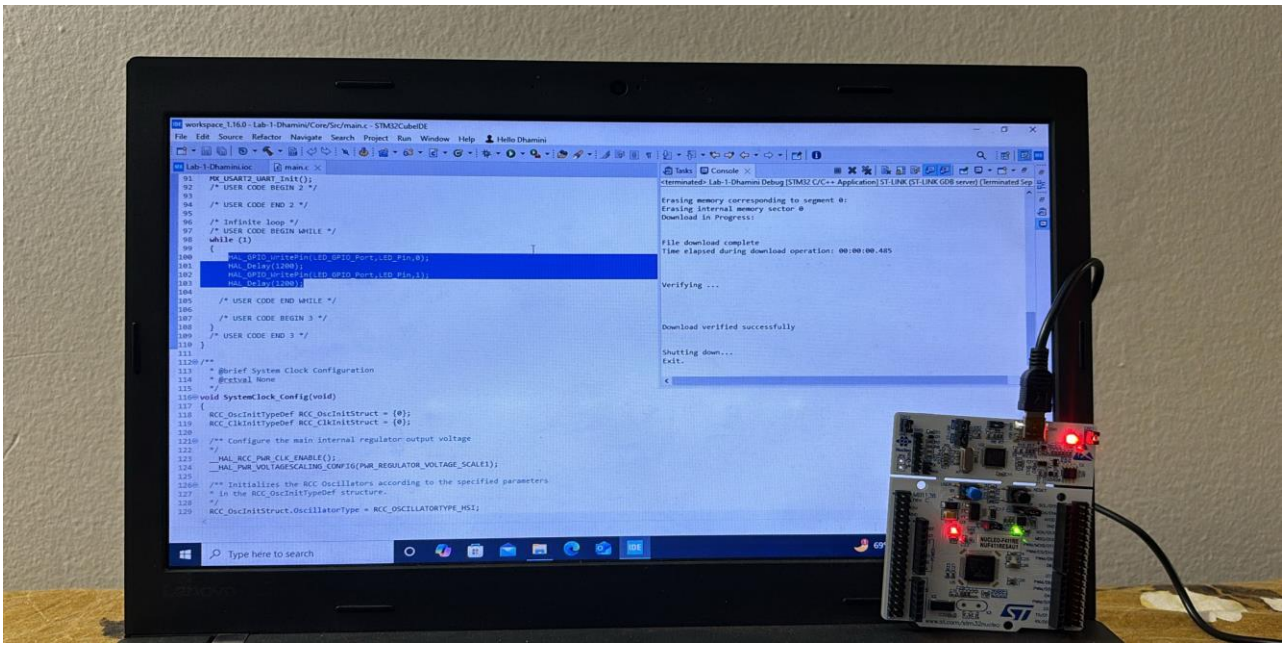
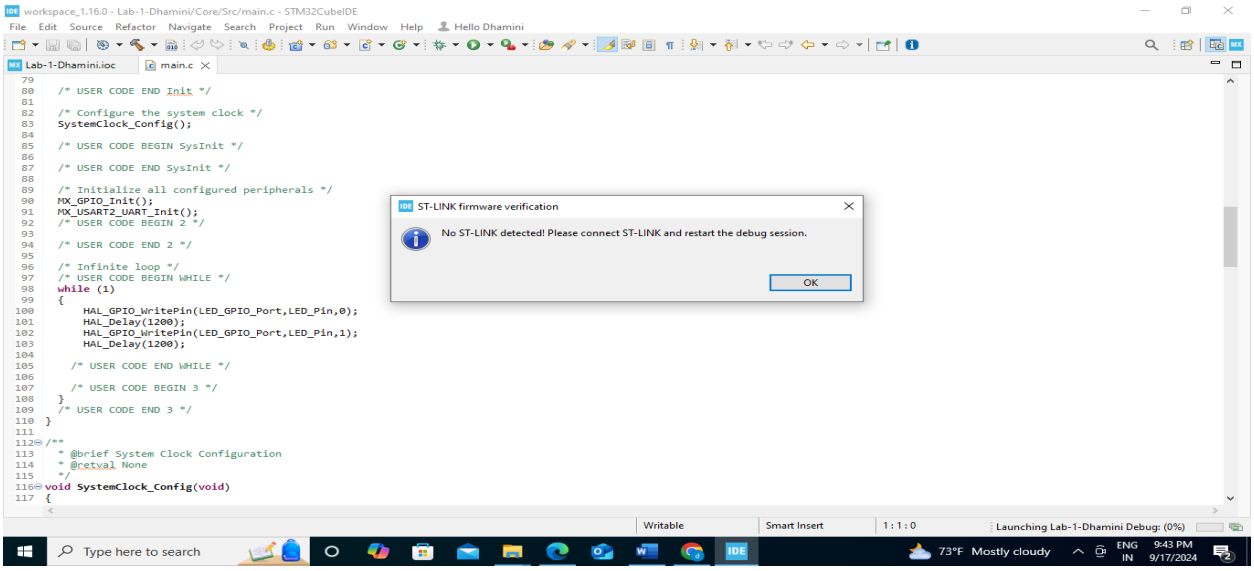


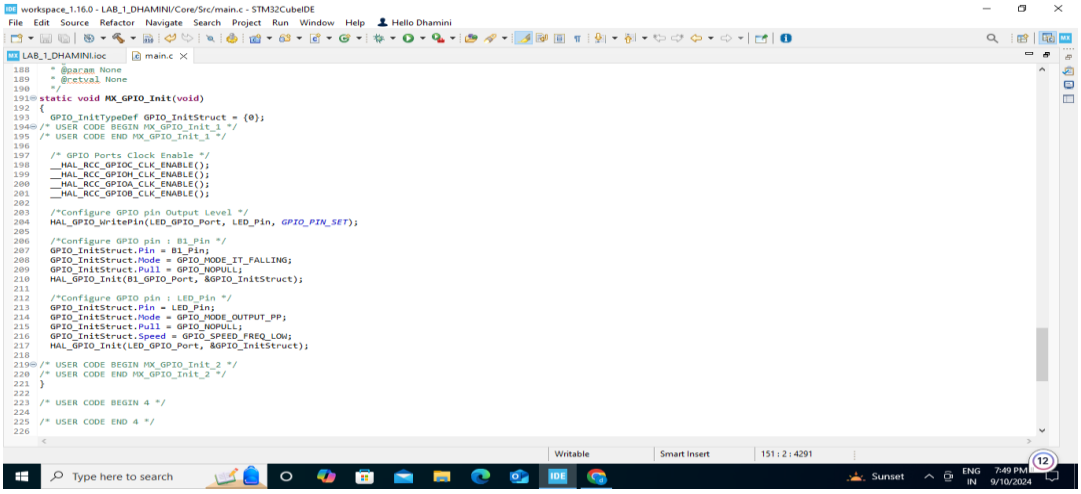
Figure showing the outcome is blinking of an LED for 1200 Milliseconds :



4. "I encountered an error during debugging but resolved it by linking the STM properly."



5. The process involved configuring the pin to light up the LED:

Once your project is initialized, STM32Cube helps configure the microcontroller's peripherals and pins. In the Pinout tab, you can see a visual layout of the microcontroller's pins.

- Where you locate the GPIO pin designated as GPIO_OUTPUT and enter the user label as LED. Then modify the settings as the default output state (low/high) and adjust the pin mode (push-pull /pull-up).
- After the settings are done STM32Cube will automatically generate the required initialization code, including the configuration for the GPIO pin. As shown below: