EXPERIMENT--02-INTEFACING-A-DIGITAL-INPUT-TO-ARM-DEVELOPMENT-BOARD

Aim: To Interface a Digital Input (userpush button) to ARM development board and write a program to obtain the data and flash the led

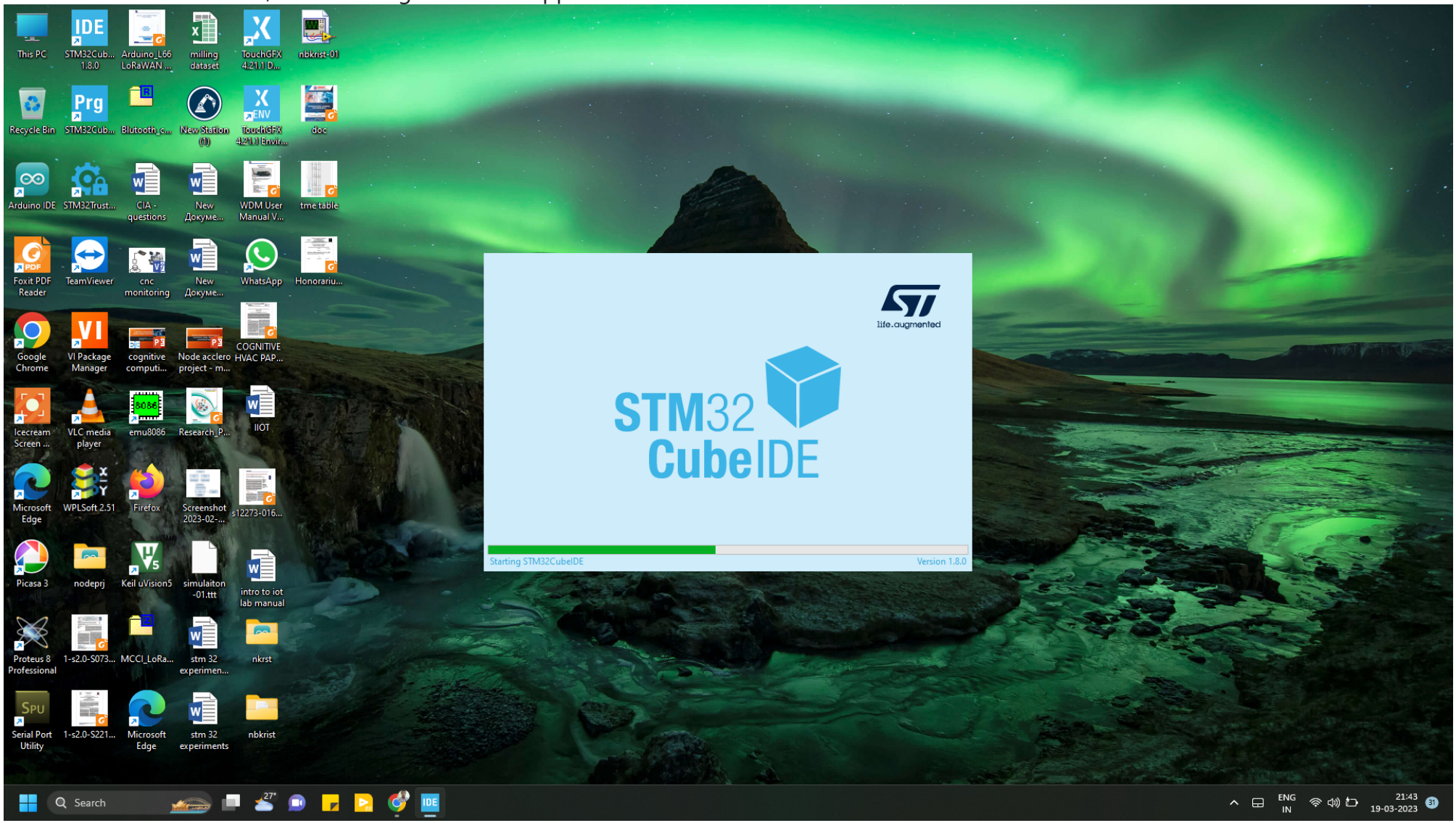
Components required: STM32 CUBE IDE, ARM IOT development board, STM programmer tool.

Theory

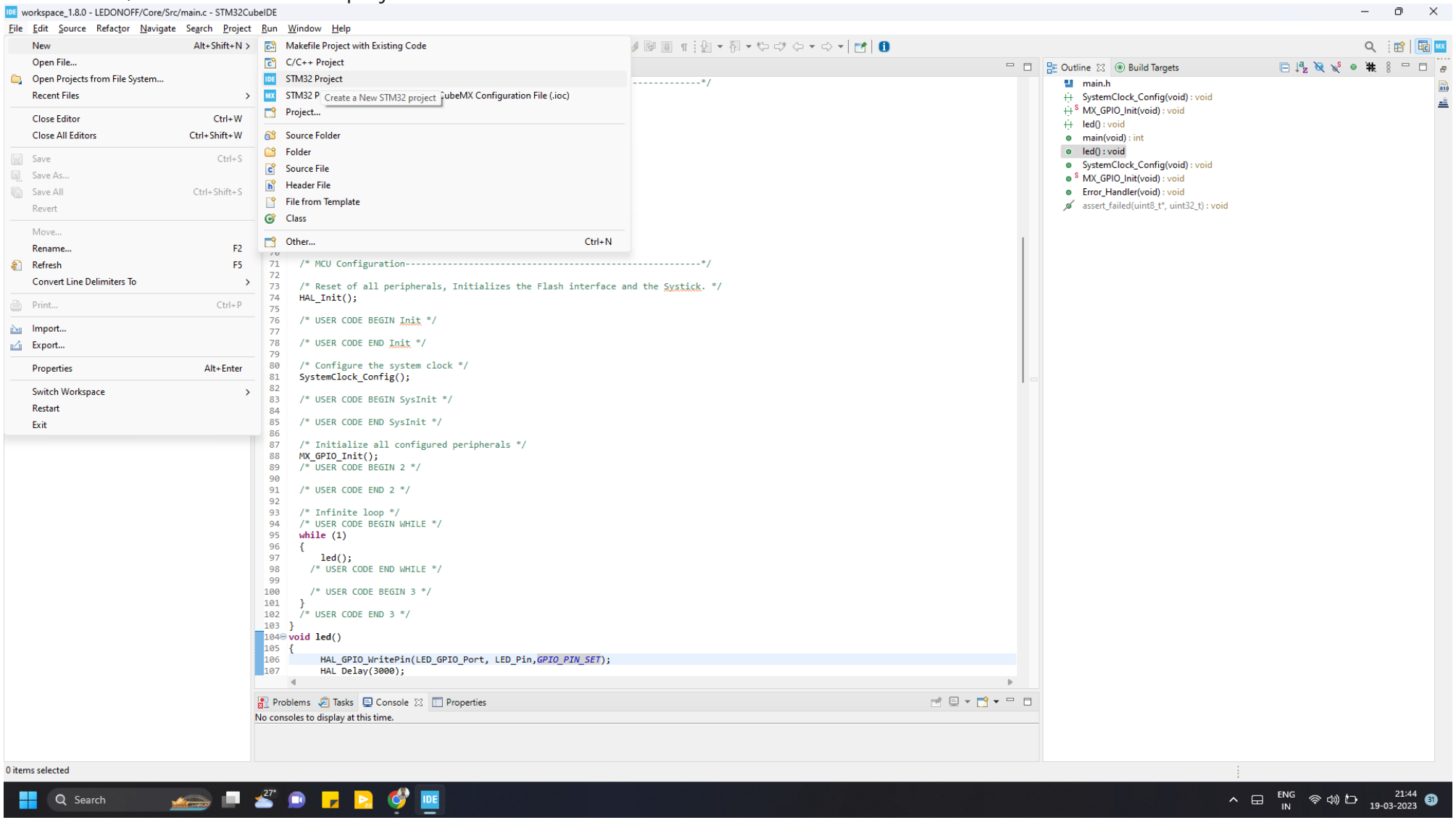
The full form of an ARM is an advanced reduced instruction set computer (RISC) machine, and it is a 32-bit processor architecture expanded by ARM holdings. The applications of an ARM processor include several microcontrollers as well as processors. The architecture of an ARM processor was licensed by many corporations for designing ARM processor-based SoC products and CPUs. This allows the corporations to manufacture their products using ARM architecture. Likewise, all main semiconductor companies will make ARM-based SOCs such as Samsung, Atmel, TI etc.

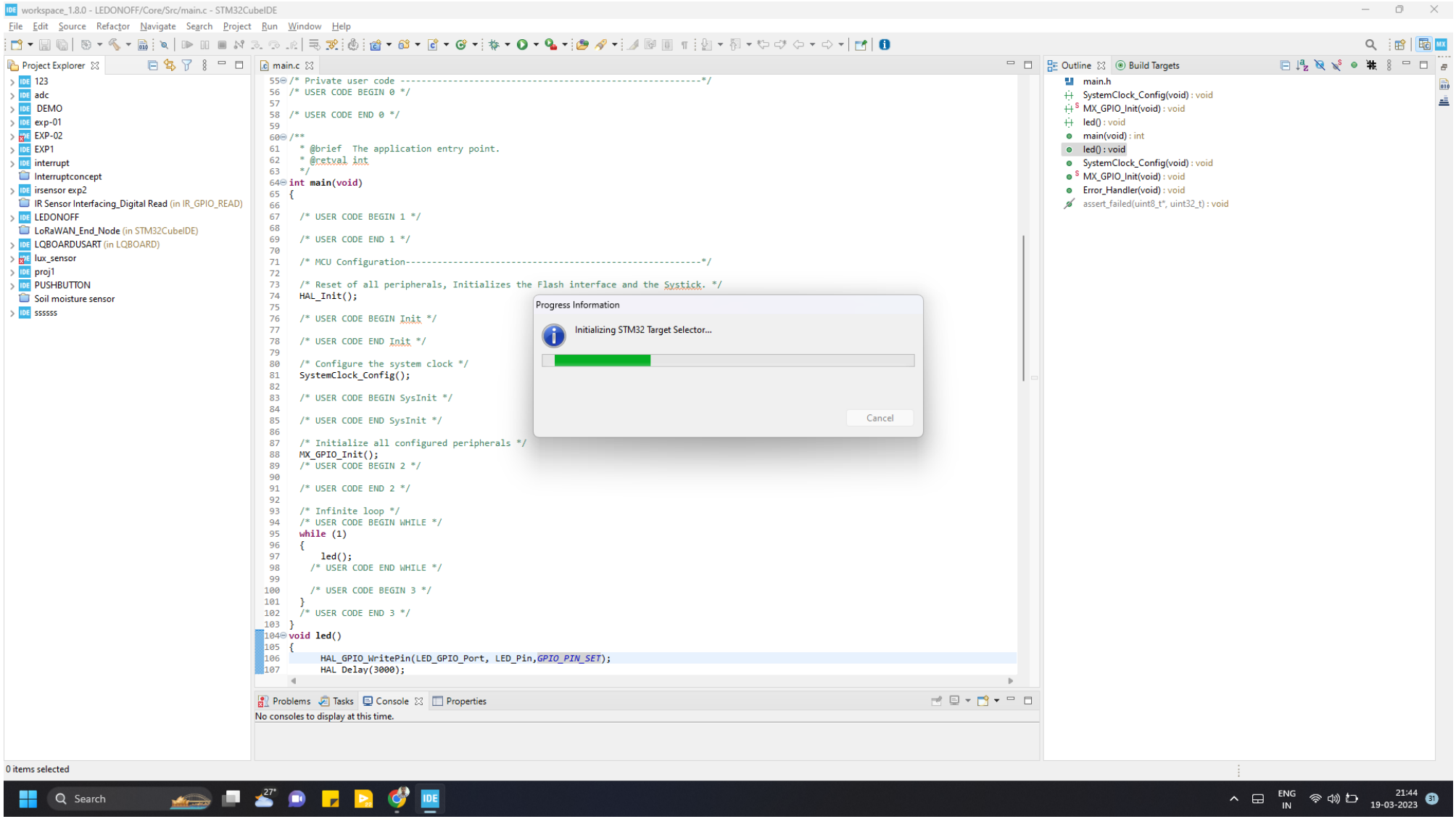
Procedure:

1. click on STM 32 CUBE IDE, the following screen will appear



2. click on FILE, click on new stm 32 project





3. select the target to be programmed as shown below and click on next

STM32 Project

Target Selection

Select STM32 target or STM32Cube example

IDE

MCU/MPU Selector | Board Selector | Example Selector | Cross Selector

MCU/MPU Filters

Part Number: STM32G071RB

Core

Check/Uncheck All

- [ ] Arm Cortex-A7 + Arm Cortex-M4
- [ ] Arm Cortex-M0
- [ ] Arm Cortex-M0+
- [ ] Arm Cortex-M3
- [ ] Arm Cortex-M4
- [ ] Arm Cortex-M4 + Arm Cortex-M0+
- [ ] Arm Cortex-M7
- [ ] Arm Cortex-M7 + Arm Cortex-M4
- [ ] Arm Cortex-M33

Series

Check/Uncheck All

- [ ] STM32F0
- [ ] STM32F1
- [ ] STM32F2
- [ ] STM32F3
- [ ] STM32F4
- [ ] STM32F7
- [ ] STM32G0
- [ ] STM32G4
- [ ] STM32H7
- [ ] STM32L0

Features | Block Diagram | Docs & Resources | Datasheet | Buy

STM32G0 Series

STM32G071RB

Mainstream Arm Cortex-M0+ MCU with 128 Kbytes of of Flash memory memory, 36 Kbytes RAM, 64 MHz CPU, 4x USART, timers, ADC, DAC, comm. I/F, 1.7-3.6V

ACTIVE Active
Product is in mass production

Unit Price for 10kU (US$) : **1.803**

Boards: NUCLEO-G071RB - STM32G071B-DISCO

LQFP64

The STM32G071x8/xB mainstream microcontrollers are based on high-performance Arm® Cortex®-M0+ 32-bit RISC core operating at up to 64 MHz frequency. Offering a high level of integration, they are suitable for a wide range of applications in consumer, industrial and appliance domains and ready for the Internet of Things (IoT) solutions.
The devices incorporate a memory protection unit (MPU), high-speed embedded memories (up to 128 Kbytes of Flash program memory with read protection, write protection, proprietary code protection, and securable area, and 36 Kbytes of SRAM), DMA and an extensive range of system functions, enhanced I/Os and peripherals. The devices offer standard communication interfaces (two I^2Cs, two SPIs / one I^2S, one HDMI CEC, and four USARTs), one 12-bit ADC (2.5 MSps) with up to 19 channels, one 12-bit DAC with two channels, two fast comparators, an internal voltage reference buffer, a low-power RTC, an advanced control PWM timer running at up to double the CPU frequency, five general-purpose 16-bit timers with one running at up to double the CPU frequency, a 32-bit general-purpose timer, two basic timers, two low-power 16-bit timers, two watchdog timers, and a SysTick timer. The STM32G071x8/xB devices provide a fully integrated USB Type-C Power Delivery controller.
The devices operate within ambient temperatures from -40 to 125°C. They can operate with supply voltages from 1.7 V to 3.6 V. Optimized dynamic consumption combined with a comprehensive set of power-saving modes, low-power timers and low-power UART, allows the design of low-power applications.

MCUs/MPUs List: 2 items

Display similar items | Export

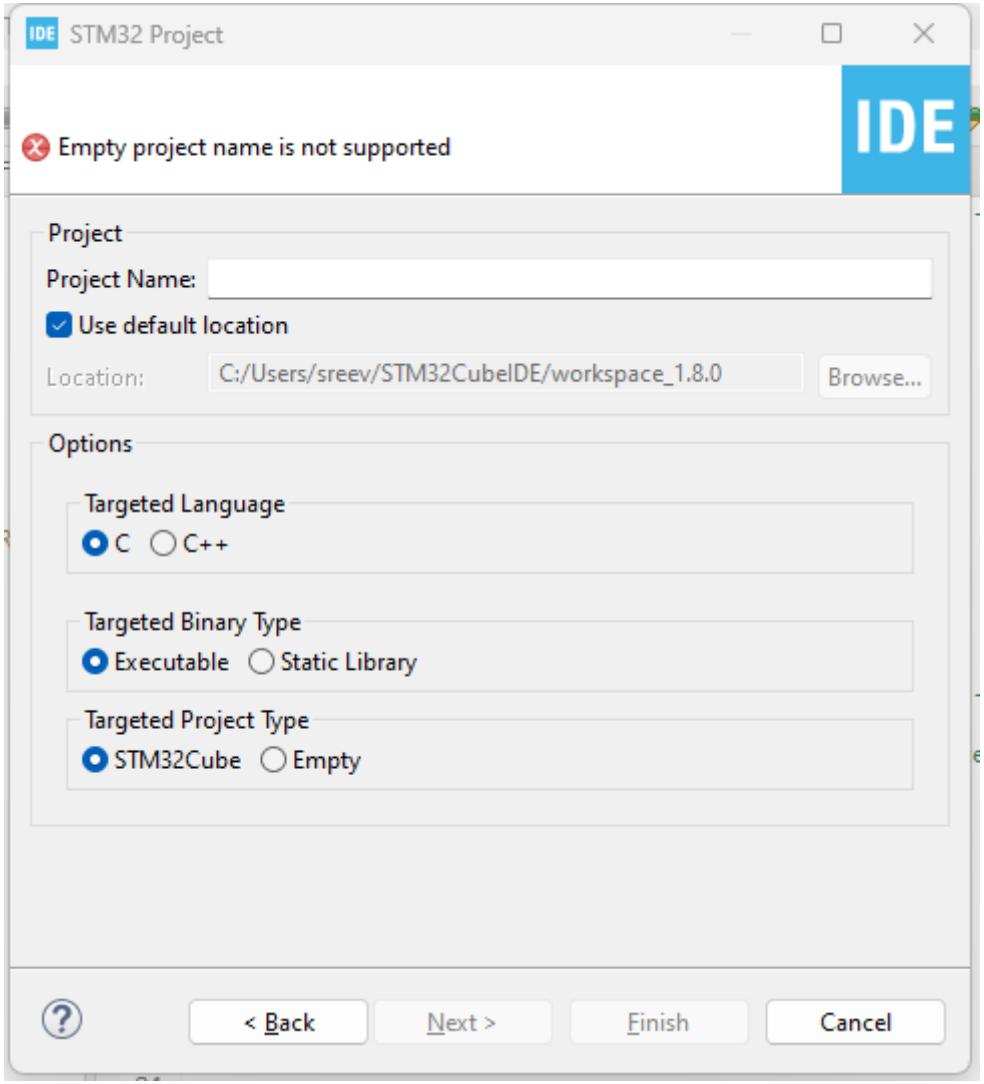
*	Part No	Reference	Marketing Status	Unit Price for 10kU (U...	Board	Package	Flash	RAM	IO	Freq.
☆	STM32G071RB	STM32G071...	Active	1.803		UFBGA64	128 kBytes	36 kBytes	60	64 MHz
☆		STM32G071...	Active	1.803	NUCLEO-G0...STM32G071...	LQFP64	128 kBytes	36 kBytes	60	64 MHz

< Back | Next > | Finish | Cancel

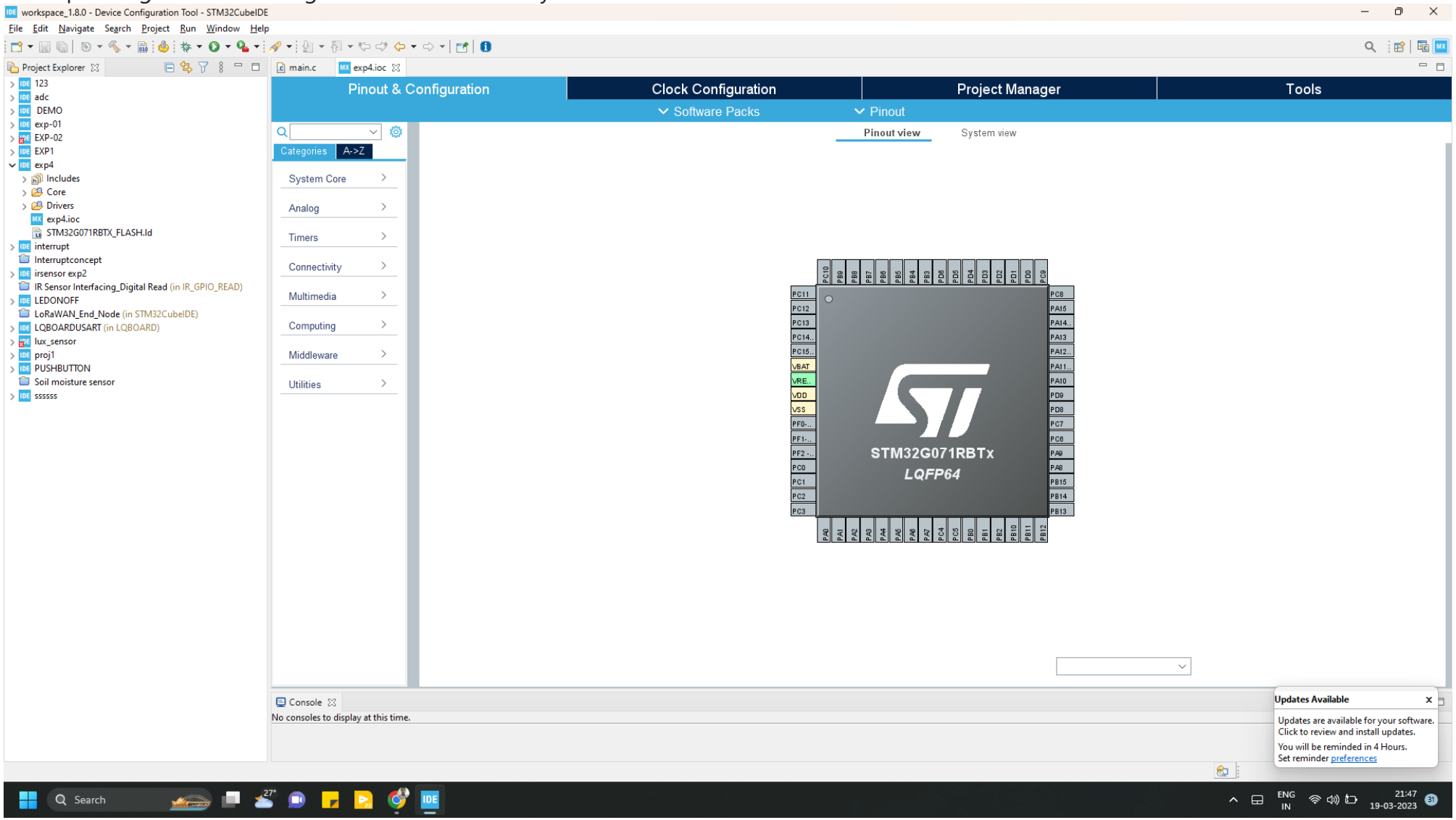
Search

ENG IN

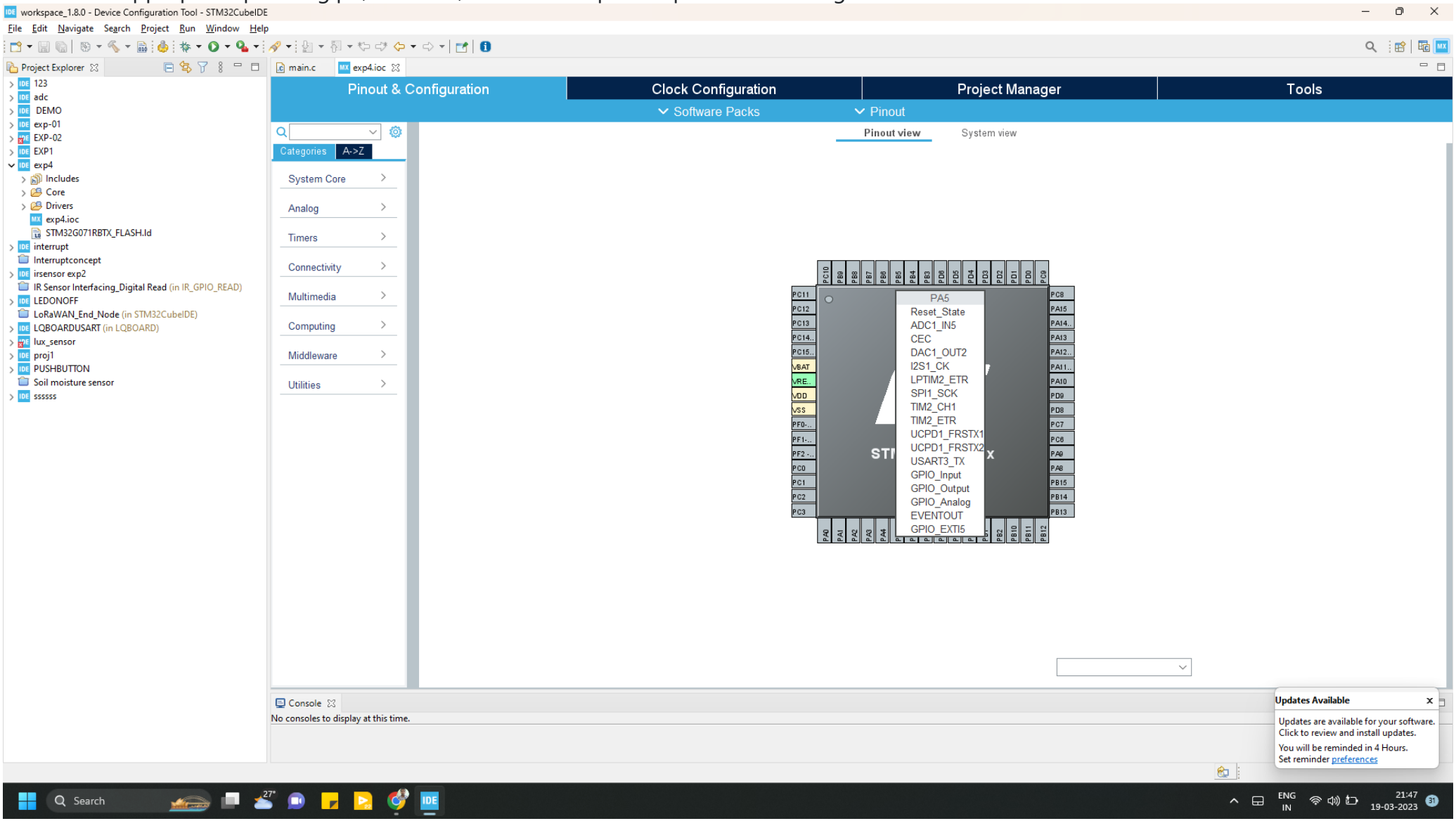
21:45
19-03-2023

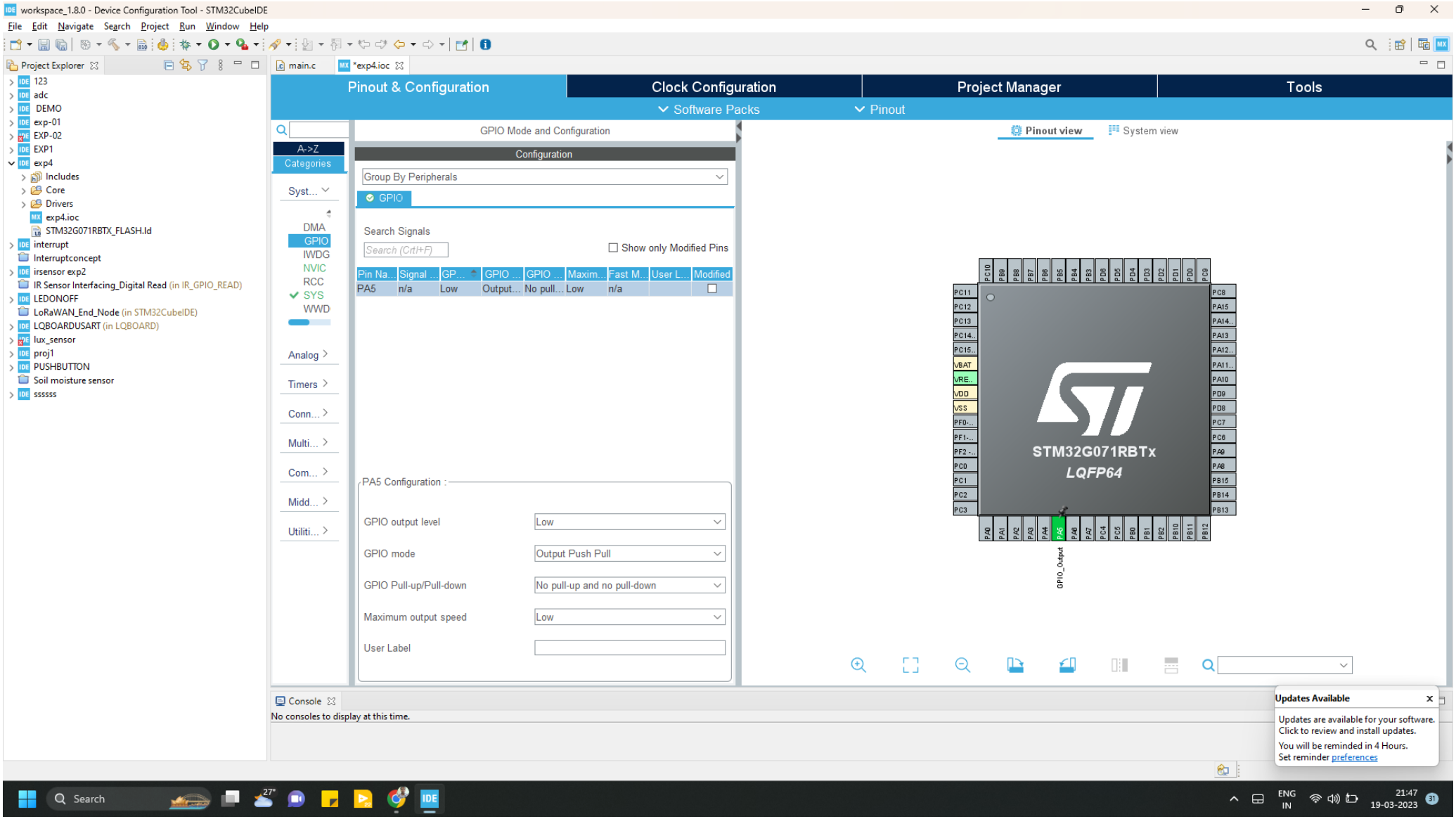
4.select the program name 

5. corresponding ioc file will be generated automatically

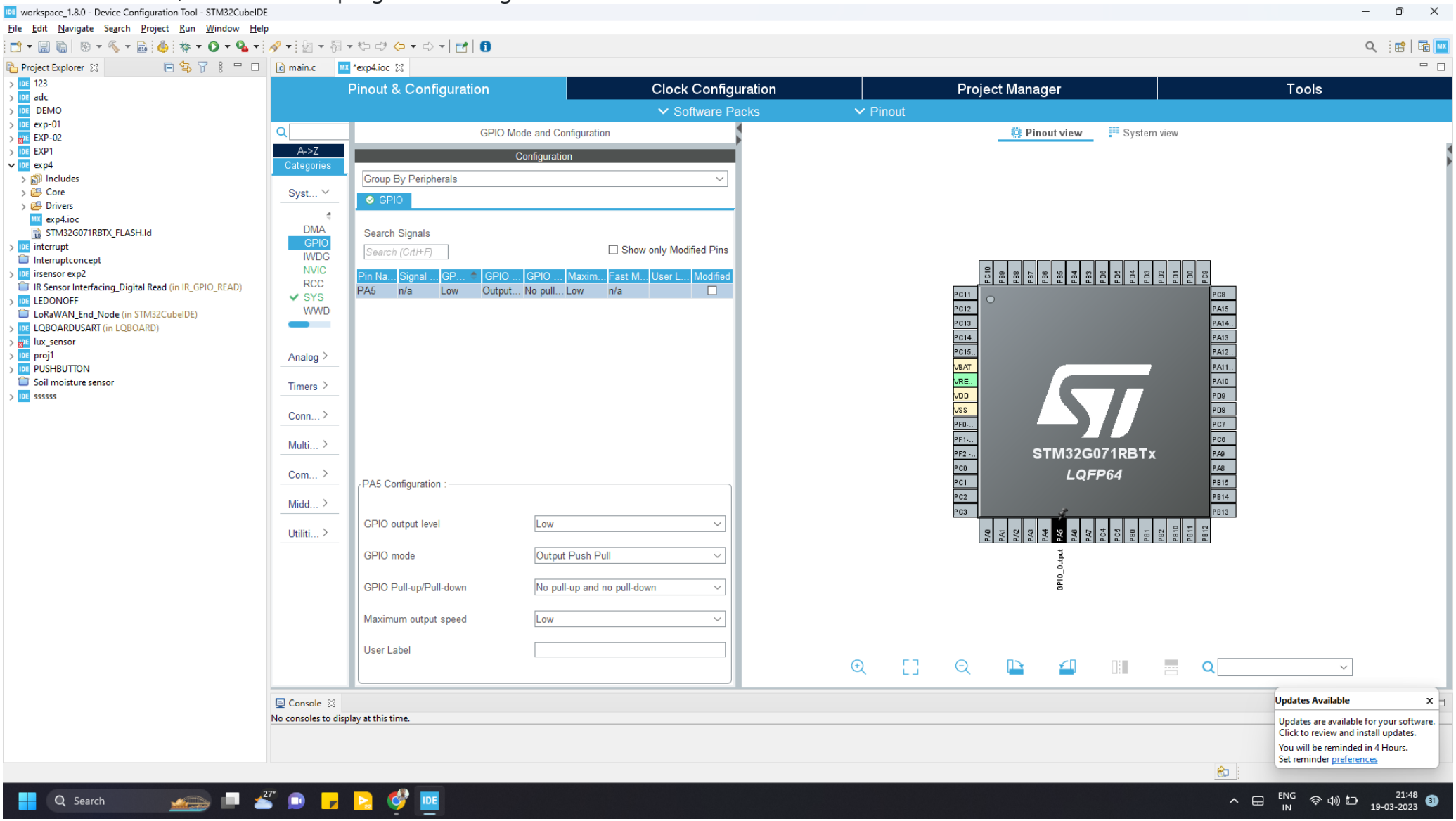


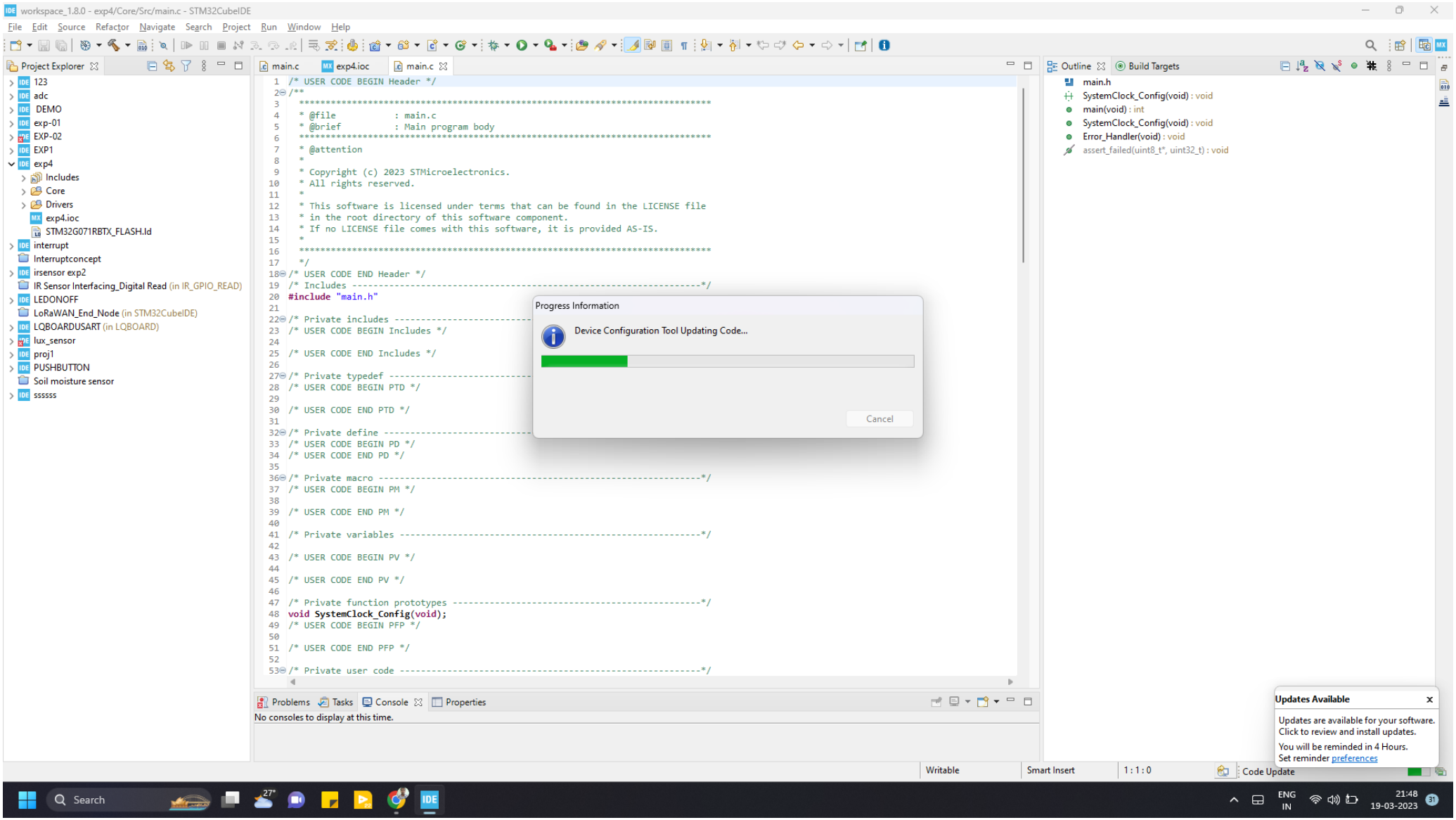
6.select the appropriate pins as gipo, in or out, USART or required options and configure



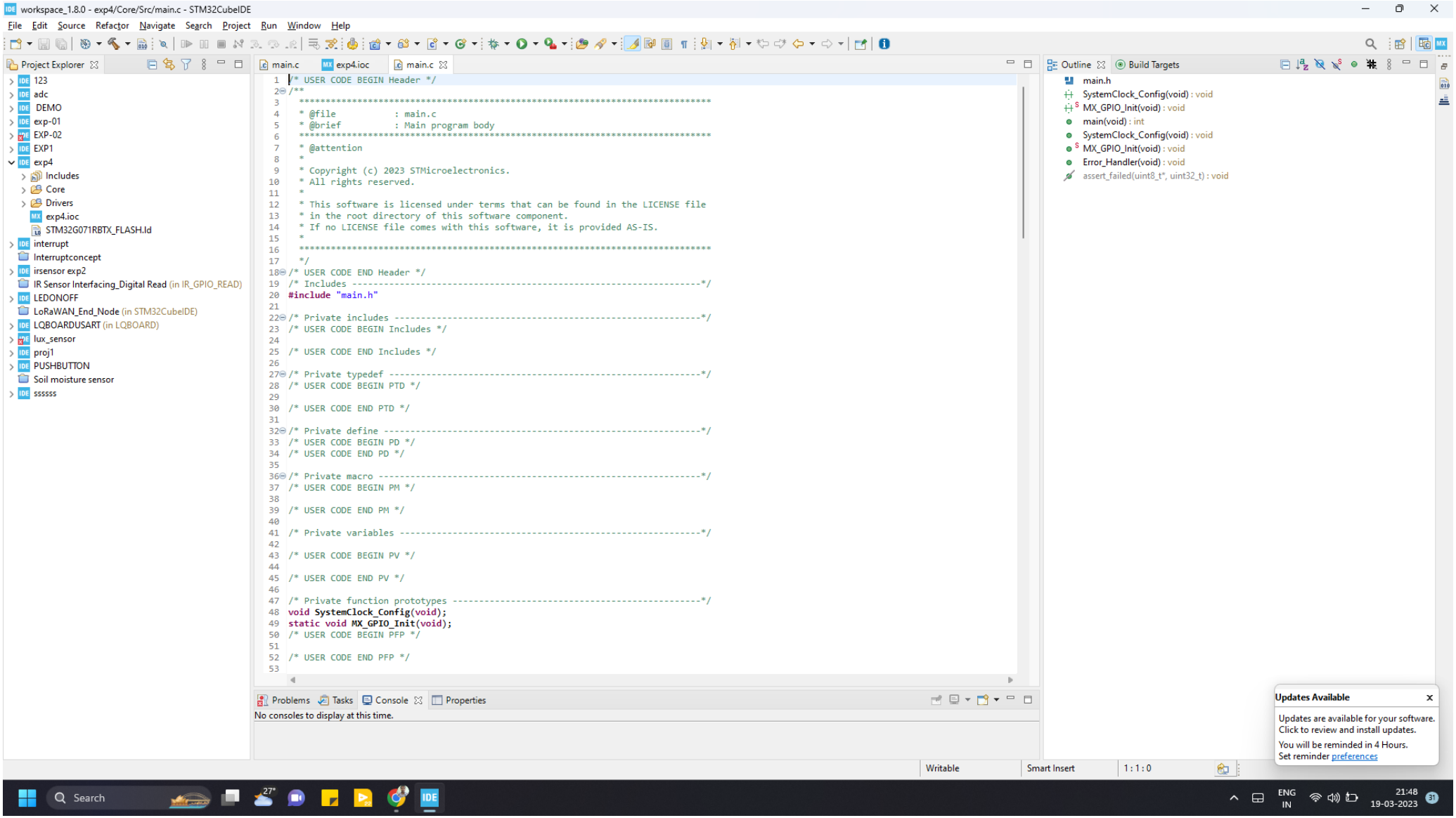

workspace_1.8.0 - Device Configuration Tool - STM32CubeIDE
File Edit Navigate Search Project Run Window Help
Project Explorer
123
adc
DEMO
exp-01
EXP-02
EXP1
exp4
Includes
Core
Drivers
exp4.ioc
STM32G071RBTX_FLASH.ld
interrupt
Interruptconcept
irsensor exp2
IR Sensor Interfacing_Digital Read (in IR_GPIO_READ)
LEDONOFF
LoRaWAN_End_Node (in STM32CubeIDE)
LQBOARDUSART (in LQBOARD)
lux_sensor
proj1
PUSHBUTTON
Soil moisture sensor
ssssss
main.c
*exp4.ioc
Pinout & Configuration
Clock Configuration
Project Manager
Tools
Software Packs
Pinout
GPIO Mode and Configuration
A->Z
Categories
Syst...
DMA
GPIO
IWDG
NVIC
RCC
SYS
WWD
Analog
Timers
Conn...
Multi...
Com...
Midd...
Utiliti...
Configuration
Group By Peripherals
GPIO
Search Signals
Search (Crtl+F)
Show only Modified Pins
Pin Na... Signal ... GP... GPIO ... GPIO ... Maxim... Fast M... User L... Modified
PA5 n/a Low Output... No pull... Low n/a
PA5 Configuration :
GPIO output level Low
GPIO mode Output Push Pull
GPIO Pull-up/Pull-down No pull-up and no pull-down
Maximum output speed Low
User Label
Pinout view
System view
STM32G071RBTx
LQFP64
GPIO_Output
Console
No consoles to display at this time.
Updates Available
Updates are available for your software.
Click to review and install updates.
You will be reminded in 4 Hours.
Set reminder preferences
Search
ENG
IN
21:47
19-03-2023

7.click on cntrl+S , automaticall C program will be generated

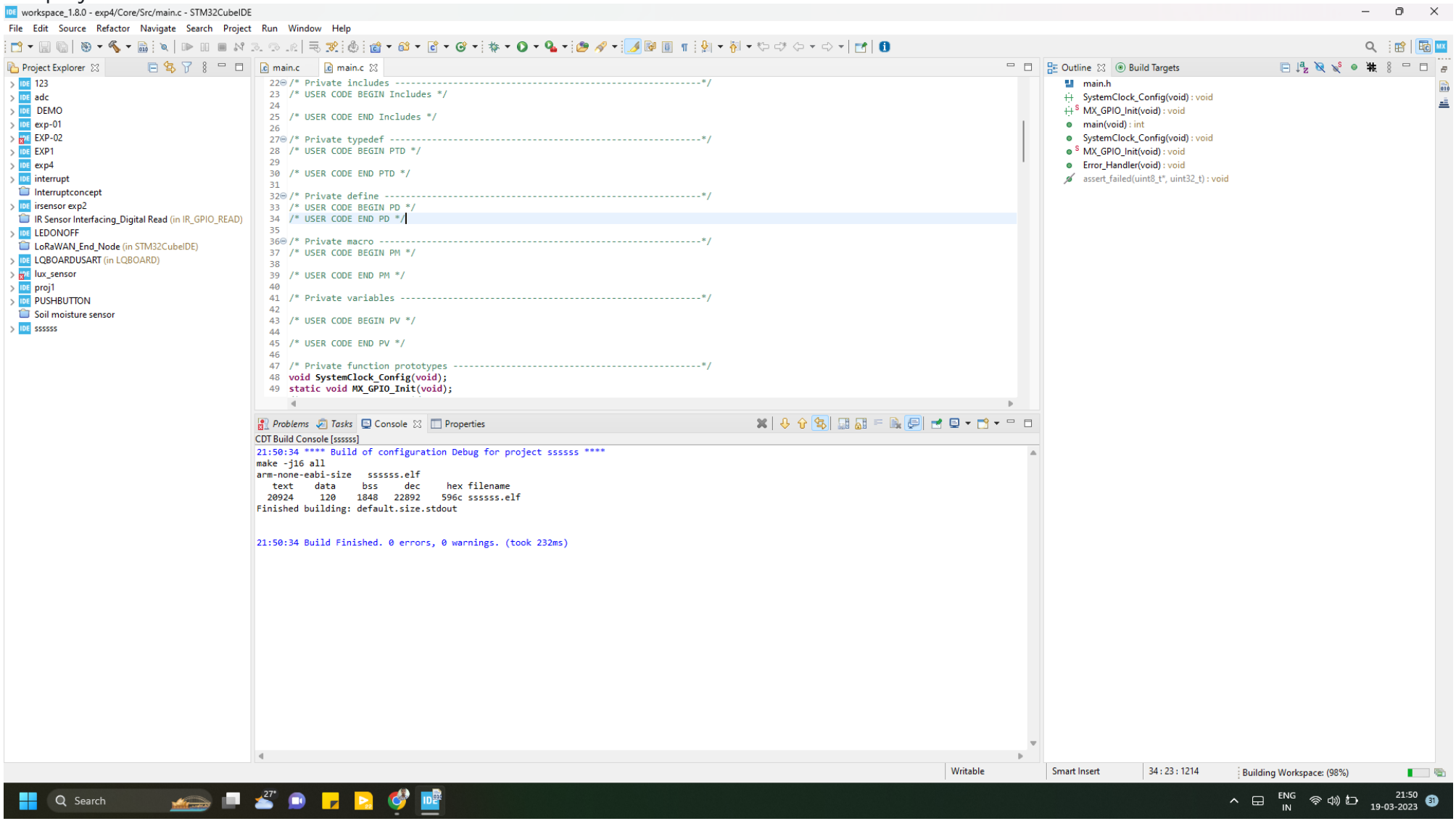




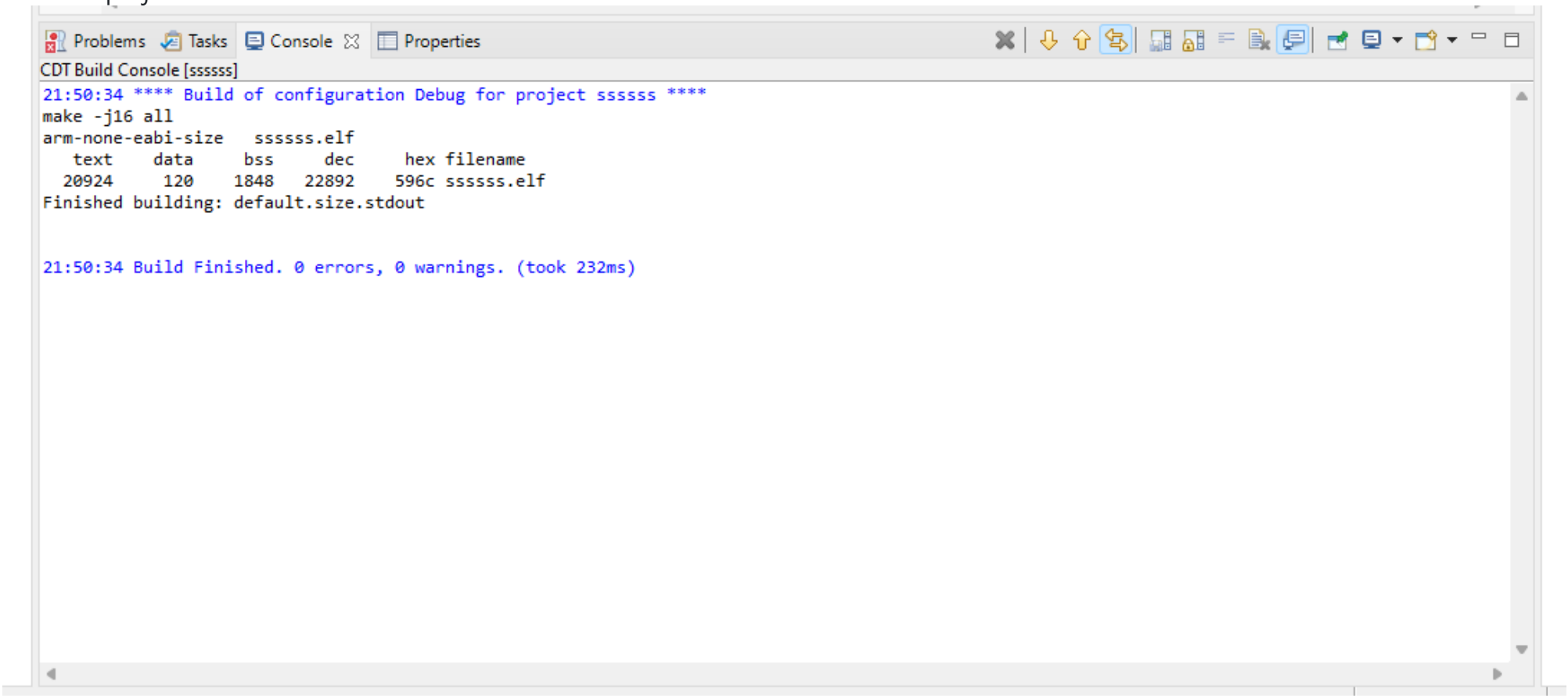
8. edit the program and as per required


workspace_1.8.0 - exp4/Core/Src/main.c - STM32CubeIDE
File Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
123
adc
DEMO
exp-01
EXP-02
EXP1
exp4
Includes
Core
Drivers
exp4.ioc
STM32G071RBTX_FLASH.ld
interrupt
Interruptconcept
irsensor exp2
IR Sensor Interfacing_Digital Read (in IR_GPIO_READ)
LEDONOFF
LoRaWAN_End_Node (in STM32CubeIDE)
LQBOARDUSART (in LQBOARD)
lux_sensor
proj1
PUSHBUTTON
Soil moisture sensor
ssssss
main.c
exp4.ioc
main.c
/* USER CODE BEGIN Header */
/**
* @file : main.c
* @brief : Main program body
* @attention
* Copyright (c) 2023 STMicroelectronics.
* All rights reserved.
* This software is licensed under terms that can be found in the LICENSE file
* in the root directory of this software component.
* If no LICENSE file comes with this software, it is provided AS-IS.
*/
/* USER CODE END Header */
/* Includes */
#include "main.h"
/* Private includes */
/* USER CODE BEGIN Includes */
/* USER CODE END Includes */
/* Private typedef */
/* USER CODE BEGIN PTD */
/* USER CODE END PTD */
/* 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 */
void SystemClock_Config(void);
static void MX_GPIO_Init(void);
/* USER CODE BEGIN PFP */
/* USER CODE END PFP */
Outline
Build Targets
main.h
SystemClock_Config(void) : void
MX_GPIO_Init(void) : void
main(void) : int
SystemClock_Config(void) : void
MX_GPIO_Init(void) : void
Error_Handler(void) : void
assert_failed(uint8_t*, uint32_t) : void
Problems Tasks Console Properties
No consoles to display at this time.
Updates Available
Updates are available for your software.
Click to review and install updates.
You will be reminded in 4 Hours.
Set reminder preferences
Writable
Smart Insert
1 : 1 : 0
Search
ENG
IN
21:48
19-03-2023

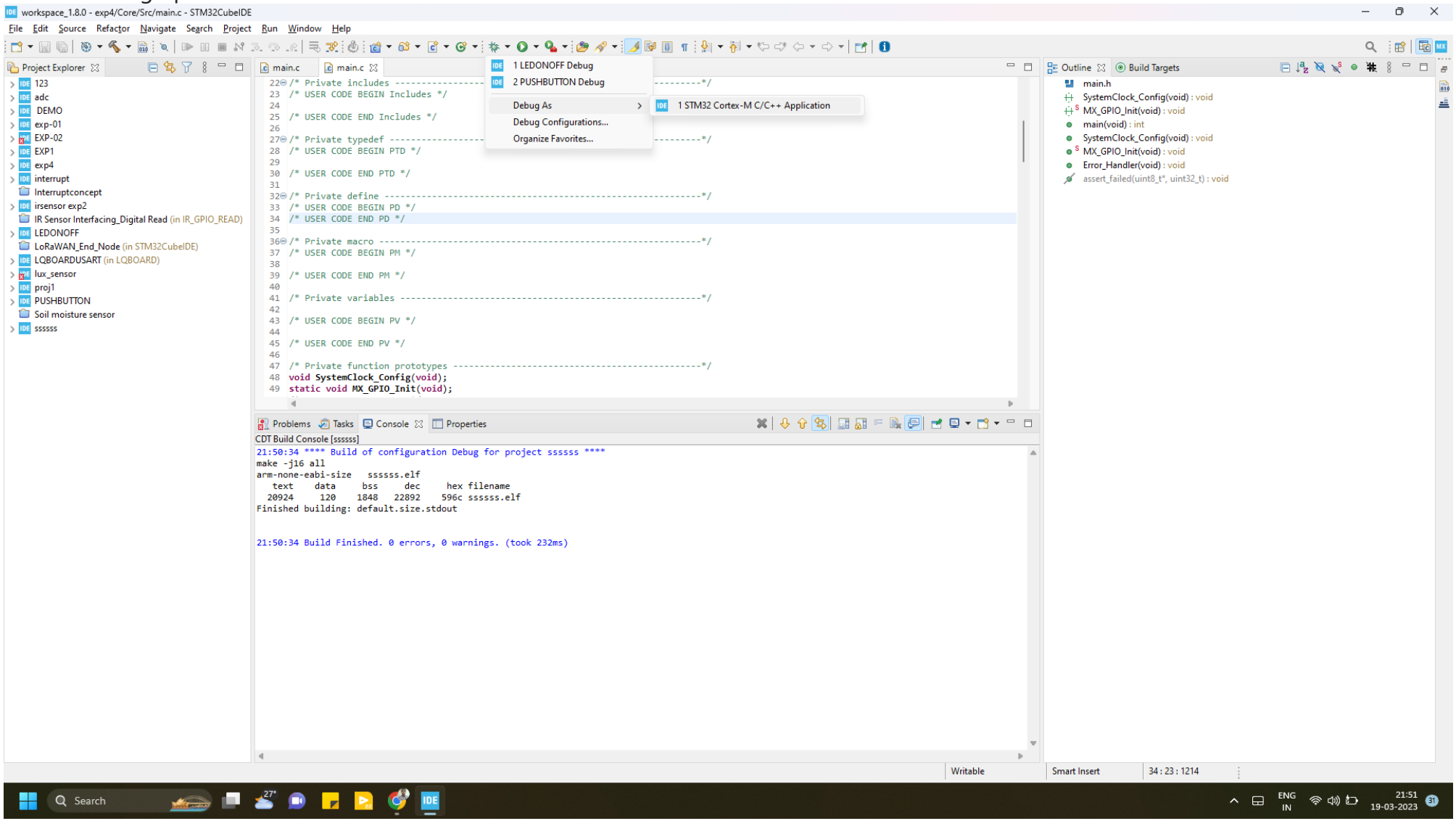
9. use project and build all



10. once the project is bulild



11. click on debug option



12. connect the ARM board to power supply and usb

13. check for execution of the output using run option

STM 32 CUBE PROGRAM :

```
#include "main.h"
#include "stdio.h"
#include "stdbool.h"
bool vs;

void SystemClock_Config(void);
static void MX_GPIO_Init(void);

int main(void)
{

  HAL_Init();
  SystemClock_Config();

  MX_GPIO_Init();

  while (1)
  {
          vs = HAL_GPIO_ReadPin(GPIOC,GPIO_PIN_13);
          if(vs==0)
          {
                  HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_SET);
                  HAL_Delay(50);
                  HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
                  HAL_Delay(50);
          }
          else
          {
                  HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
          }

  }
```

```
}


void SystemClock_Config(void)
{
  RCC_OscInitTypeDef RCC_OscInitStruct = {0};
  RCC_ClkInitTypeDef RCC_ClkInitStruct = {0};

  HAL_PWREx_ControlVoltageScaling(PWR_REGULATOR_VOLTAGE_SCALE1);

  RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSI;
  RCC_OscInitStruct.HSIState = RCC_HSI_ON;
  RCC_OscInitStruct.HSIDiv = RCC_HSI_DIV1;
  RCC_OscInitStruct.HSICalibrationValue = RCC_HSICALIBRATION_DEFAULT;
  RCC_OscInitStruct.PLL.PLLState = RCC_PLL_NONE;
  if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
  {
    Error_Handler();
  }

  RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
                              |RCC_CLOCKTYPE_PCLK1;
  RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_HSI;
  RCC_ClkInitStruct.AHBCLKDivider = RCC_SYSCLK_DIV1;
  RCC_ClkInitStruct.APB1CLKDivider = RCC_HCLK_DIV1;

  if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_0) != HAL_OK)
  {
    Error_Handler();
  }
}


static void MX_GPIO_Init(void)
{
```

```
  GPIO_InitTypeDef GPIO_InitStruct = {0};
  __HAL_RCC_GPIOC_CLK_ENABLE();
  __HAL_RCC_GPIOA_CLK_ENABLE();


  HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);

  GPIO_InitStruct.Pin = GPIO_PIN_13;
  GPIO_InitStruct.Mode = GPIO_MODE_INPUT;
  GPIO_InitStruct.Pull = GPIO_PULLUP;
  HAL_GPIO_Init(GPIOC, &GPIO_InitStruct);

  GPIO_InitStruct.Pin = GPIO_PIN_5;
  GPIO_InitStruct.Mode = GPIO_MODE_OUTPUT_PP;
  GPIO_InitStruct.Pull = GPIO_NOPULL;
  GPIO_InitStruct.Speed = GPIO_SPEED_FREQ_LOW;
  HAL_GPIO_Init(GPIOA, &GPIO_InitStruct);

}

void Error_Handler(void)
{
  __disable_irq();
  while (1)
  {
  }
}

#ifdef  USE_FULL_ASSERT

void assert_failed(uint8_t *file, uint32_t line)
{

}
#endif
```

Output :

NUCLEO-G071RB
NUG071RB$AU2
www.st.com/stm32nucleo
Technical Details
Temp. (max)
Material
Transmission
Length 1 Meter.
Micro USB Data Cable
Mobile USB Cable
USB Cable
120.00 (Inclusive of all taxes)
Oct. 2022
TECHNOLOGIES PVT. LTD.

NUCLEO-G071RB
NUG071RB$AU2
www.st.com/stm32nucleo
USER
RESET
STLINK
Saathi
chnical Details
Amp. (max)
terial
a Transmission
Length 1 Meter.
ro USB Data Cable
Mobile USB Cable
N Cable
120.00 (Inclusive of all taxes)
Oct. 2022

Result :

Interfacing a digital Input (Pushbutton) with ARM microcontroller based IOT development is executed and the results are verified.