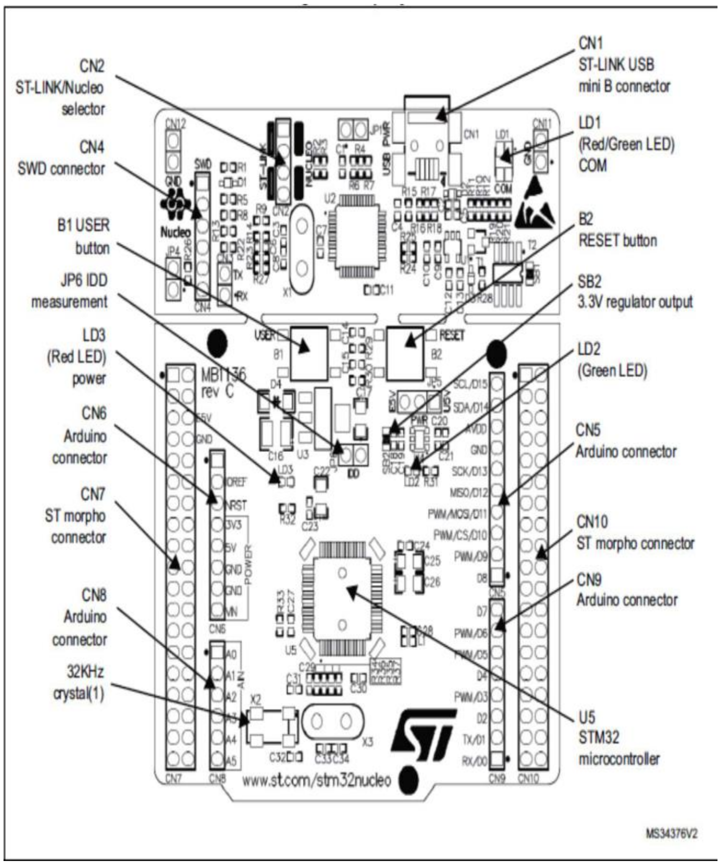
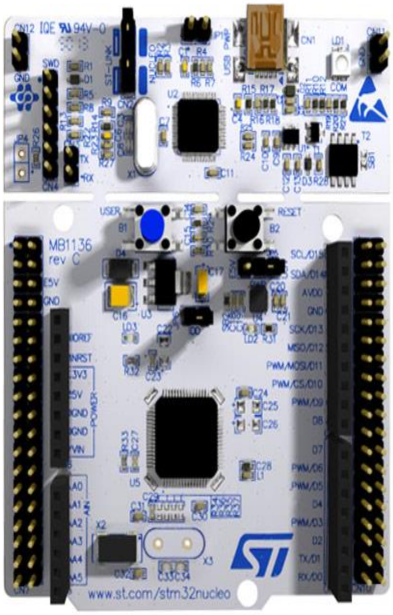
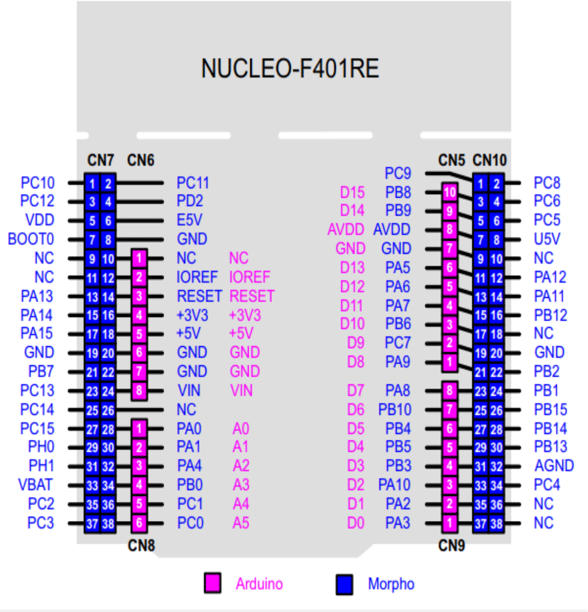
Introduction to

STM32CubeMX

Features:







STM32 Nucleo F401RE Development Board

STM32 Nucleo F401RE Pinout

- ➢ Flexible board power supply
  - ✓USB or external source
- ➢ Integrated ST-Link/V2-1 debugger
  - ✓Drag & drop device flash programming
  - ✓Virtual COM port
- ➢ For user application
  - ✓1 LED
  - ✓Push button (blue)
- ➢ STM32F401RE microcontroller
- ➢ Connectors
  - ✓Arduino Uno
  - ✓ST Morpho Extension -direct access to all
  - ✓MCU I/Os

Specifications:

Microcontroller	STM32F401RET6 (32-bit)
Architecture	ARM Cortex M4 CPU with FPU
Power consumption	2.4uA at standby without RTC
CPU Frequency	84 MHz
Crystal Oscillator Range	4 to 26 MHz
MCU Operating Voltage (VDD)	1.7V to 3.6V
Board Operating Voltage (VIN)	7V to 15V
Flash Memory	512KB
SRAM	96 KB
GPIO Pins	50
ADC	12-bit 16Channel
RTC	In-built 32kHz with calibration
Timers	16-bit (6) 32-bit (2)
Watchdog Timers	2
USART/UART Communication	4
I2C Communication	3
SPI Communication	3
USB2.0 Support	Yes
Internal Crystal Oscillator	Yes, 16MHz
On Board Debugger	Yes, Serial Wire and JTAG

Software to be installed:

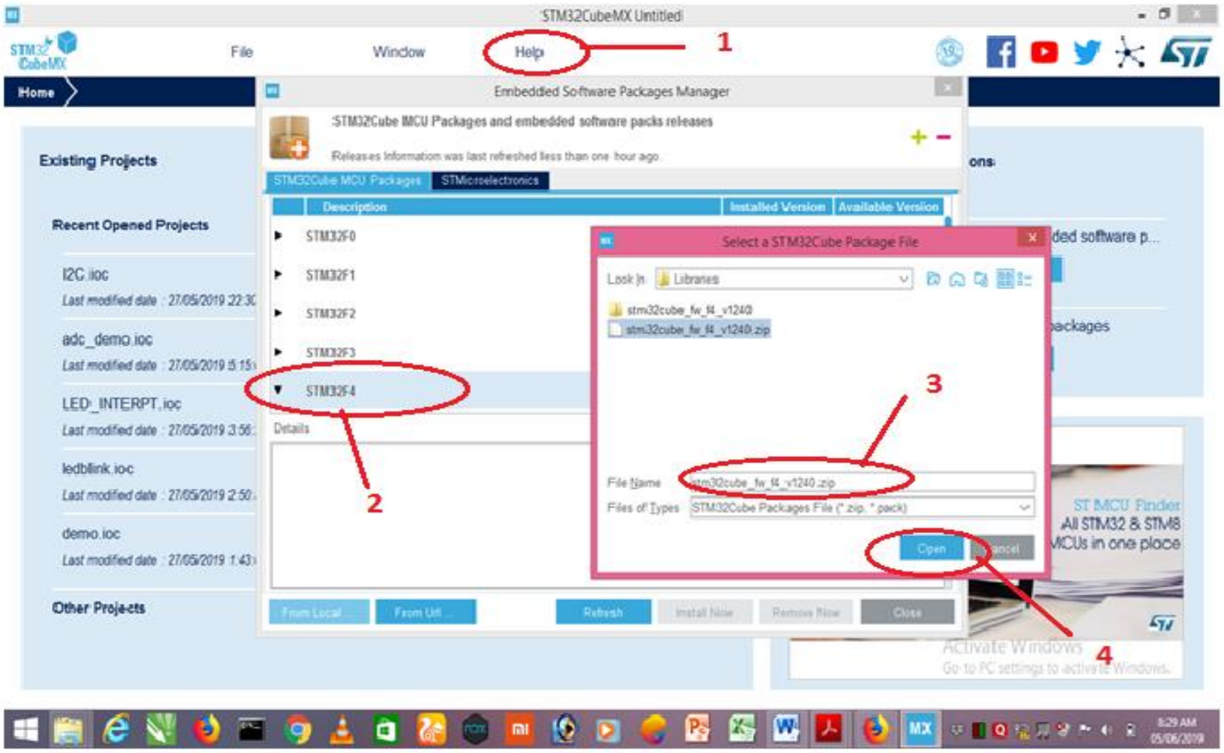
- STM32CUBEMX
  https://www.st.com/en/development-tools/stm32cubemx.html
- TRUESTUDIO
  https://atollic.com/resources/download/

STM32CubeF4 in STM32CubeMX Repository

- BeforeusingtheSTM32CubeMX,makesurethattheSTM32CubeF4 librarypackageisinstalledintheSTM32CubMXrepositoryfolder

- ThiscandonebydownloadingviaSTM32CubeMX(clickoncheckboxandthenclickonInstallNow.Requiresinternetconnection)

- For faster way, install from local.
  ✓Go to "Help->Manage embedded software packages"
  ✓Click on "From Local"...
  ✓Browse to...\STM32Cube\STM32CubeF4
  ✓Selectthefileen.stm32cubeF4.zip
  ✓Click"Open"to install.



STM32CubeMX Untitled

Embedded Software Packages Manager

STM32Cube MCU Packages and embedded software packs releases

Releases Information was last refreshed less than one hour ago.

Existing Projects

Recent Opened Projects

I2C.ioc
Last modified date : 27/05/2019 22:30

adc_demo.ioc
Last modified date : 27/05/2019 5:15

LED_INTERPT.ioc
Last modified date : 27/05/2019 3:56

ledblink.ioc
Last modified date : 27/05/2019 2:50

demo.ioc
Last modified date : 27/05/2019 1:43

Other Projects

Select a STM32Cube Package File

Look In: Libraries

stm32cube_fw_f4_v1240
stm32cube_fw_f4_v1240.zip

File Name: stm32cube_fw_f4_v1240.zip
Files of Types: STM32Cube Packages File (*.zip, *.pack)

ST MCU Finder
All STM32 & STM8 MCUs in one place

Activate Windows
Go to PC settings to activate Windows.

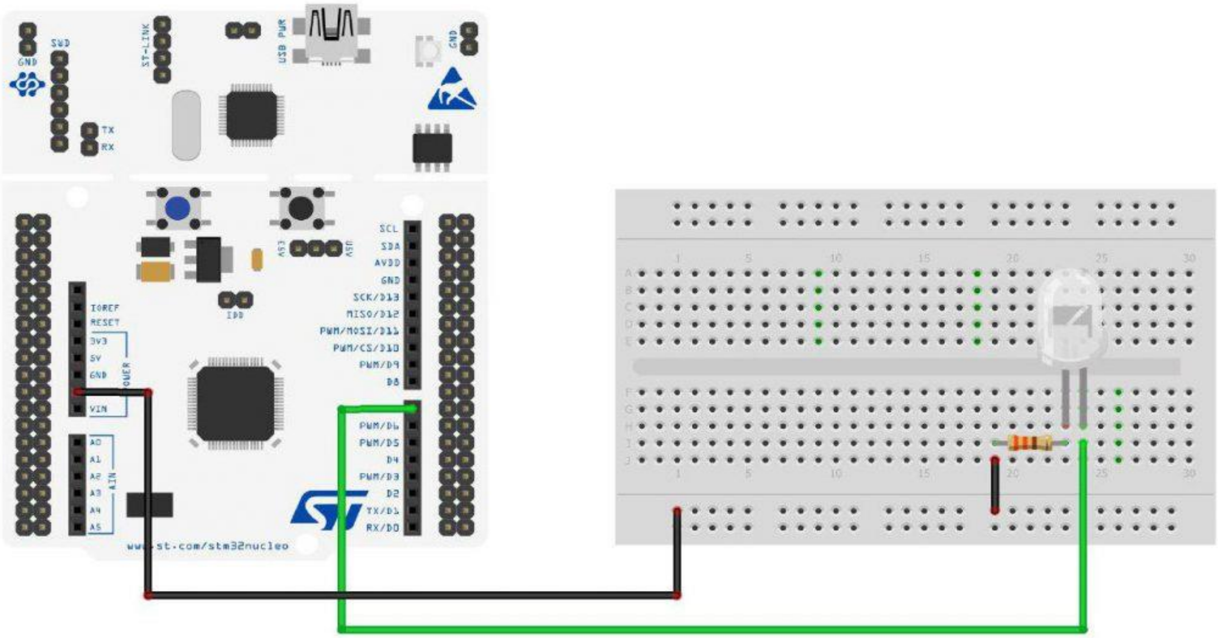
8:29 AM
05/06/2019

Practical and demonstration example

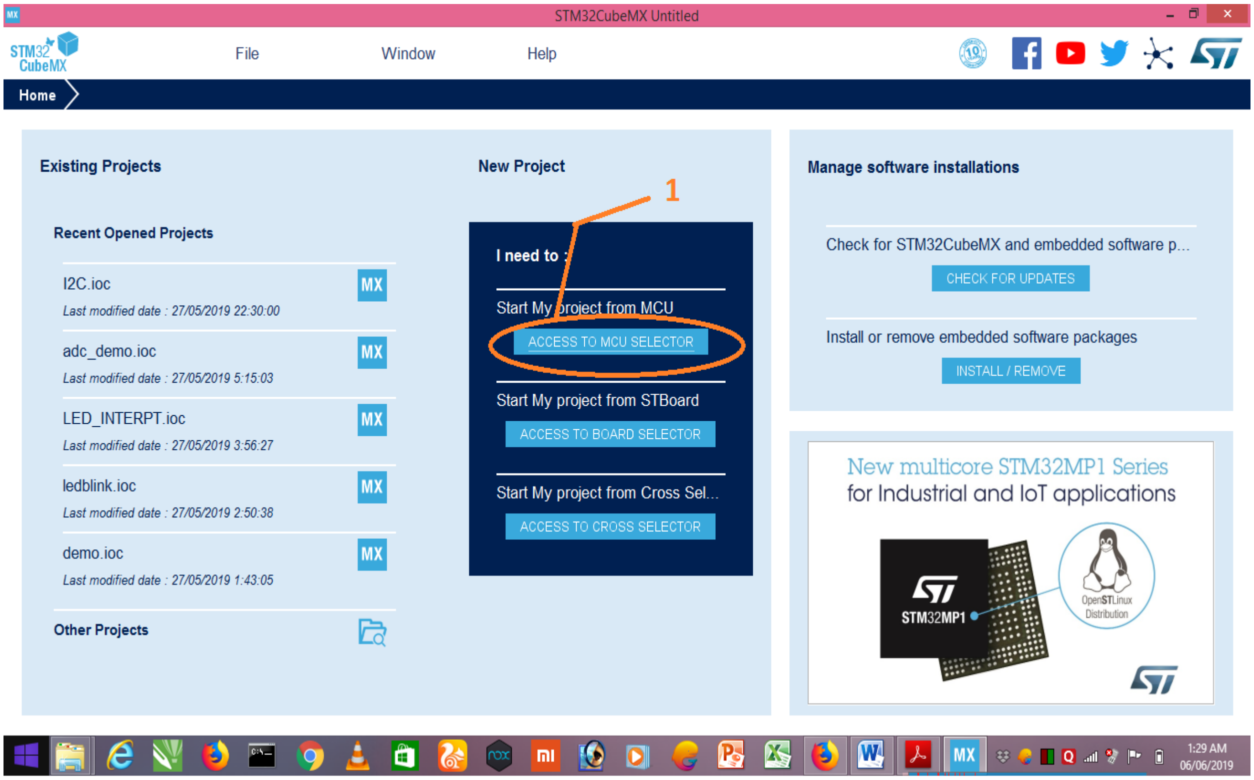
GPIO using LED blink

(General Purpose Input Output)

- ➢ This hands-on describes how to use the GPIO HAL APIs. The User push button, configured as input will be used to change the states of the LEDs.
- ➢ STM32CubeMX will be used to generate the initialization codes for the GPIO and System clock.
- ➢ This process will speed up the development as the initialization codes are generated by the STM32CubeMX tool. The user then will only need to add the user codes as per application.



Step 1: Open New Project



I2C.ioc
Last modified date : 27/05/2019 22:30:00

adc_demo.ioc
Last modified date : 27/05/2019 5:15:03

LED_INTERPT.ioc
Last modified date : 27/05/2019 3:56:27

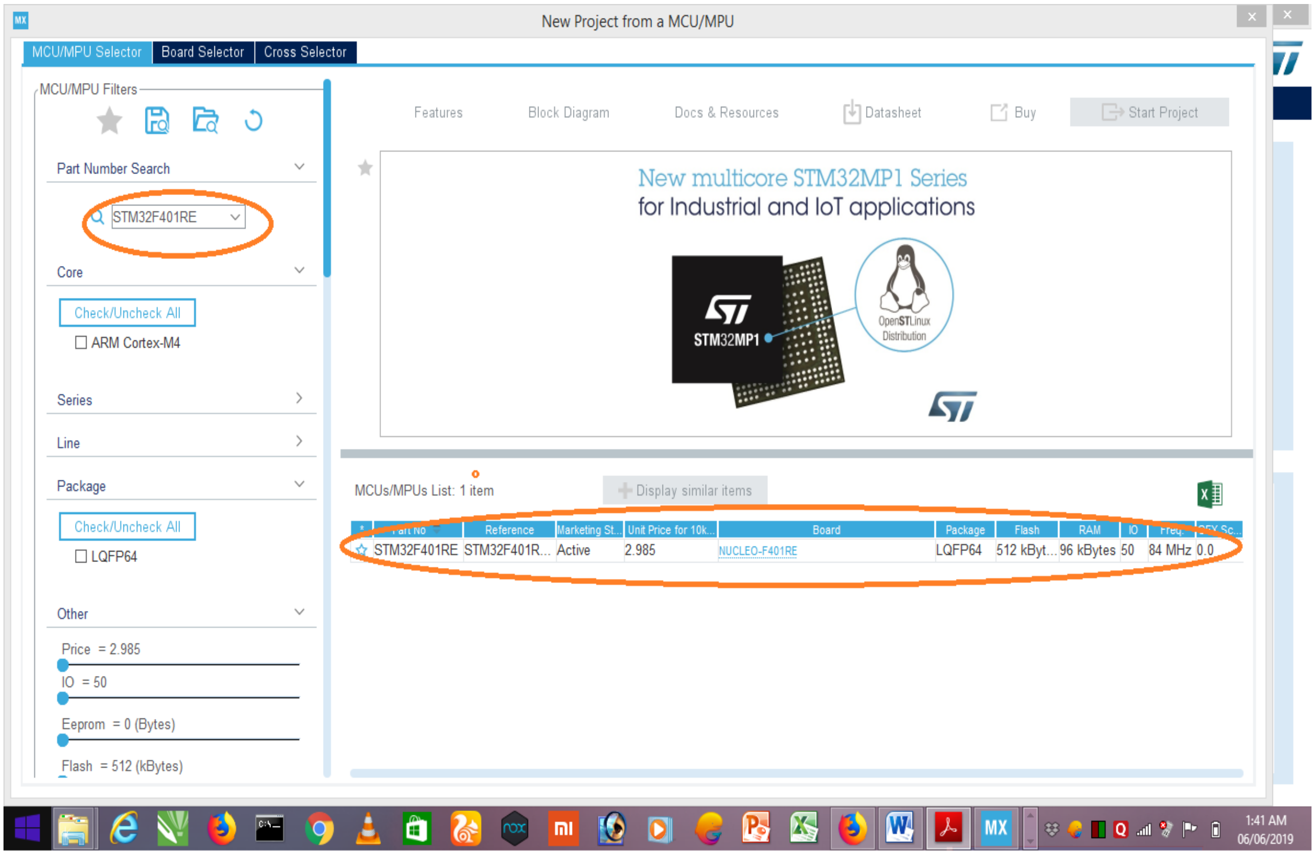
ledblink.ioc
Last modified date : 27/05/2019 2:50:38

demo.ioc
Last modified date : 27/05/2019 1:43:05

- Open **STM32CubeMX.**
- Click on **ACCESS MCU SELECTOR**
- Click No to the prompt box.
- From MCU Selector

- Select [Part No.-> STM32F401RE (LQFP64)]
- Click [Start Project] or double click [STM32F401RE] to continue

Step 2: Select MCU



Part Number Search: STM32F401RE

Core: ☐ ARM Cortex-M4

Series

Line

Package: ☐ LQFP64

Other

Price = 2.985

IO = 50

Eeprom = 0 (Bytes)

Flash = 512 (kBytes)

MCUs/MPUs List: 1 item

*	Part No	Reference	Marketing St...	Unit Price for 10k...	Board	Package	Flash	RAM	IO	Freq.	SFX Sc...
☆	STM32F401RE	STM32F401R...	Active	2.985	NUCLEO-F401RE	LQFP64	512 kByt...	96 kBytes	50	84 MHz	0.0

Use [MCU Selector] to select STM32F401RE device

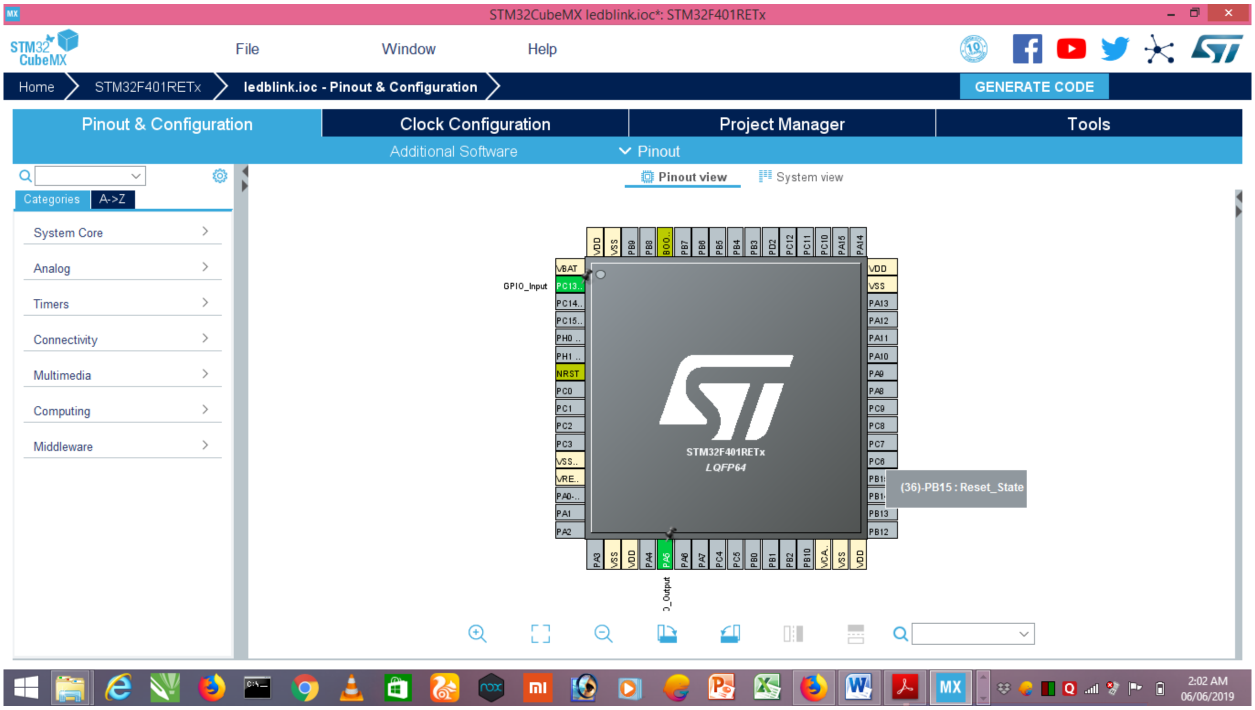
▶MCU Filter -Type “STM32F401RE” in [Part Number Search]

▶MCU List

▶Search for part No STM32F401RE (LQFP64)

▶Click [Start Project] or double click [STM32F401RE] to continue.

Step 3: Configure GPIO



Configure project settings

- ➢ [Pinout] tab

  Left-click pin PA5 and set to [GPIO_Output] mode
  Left-click pin PC13 and set to [GPIO_Input] mode

- ➢ Note: Drive LED

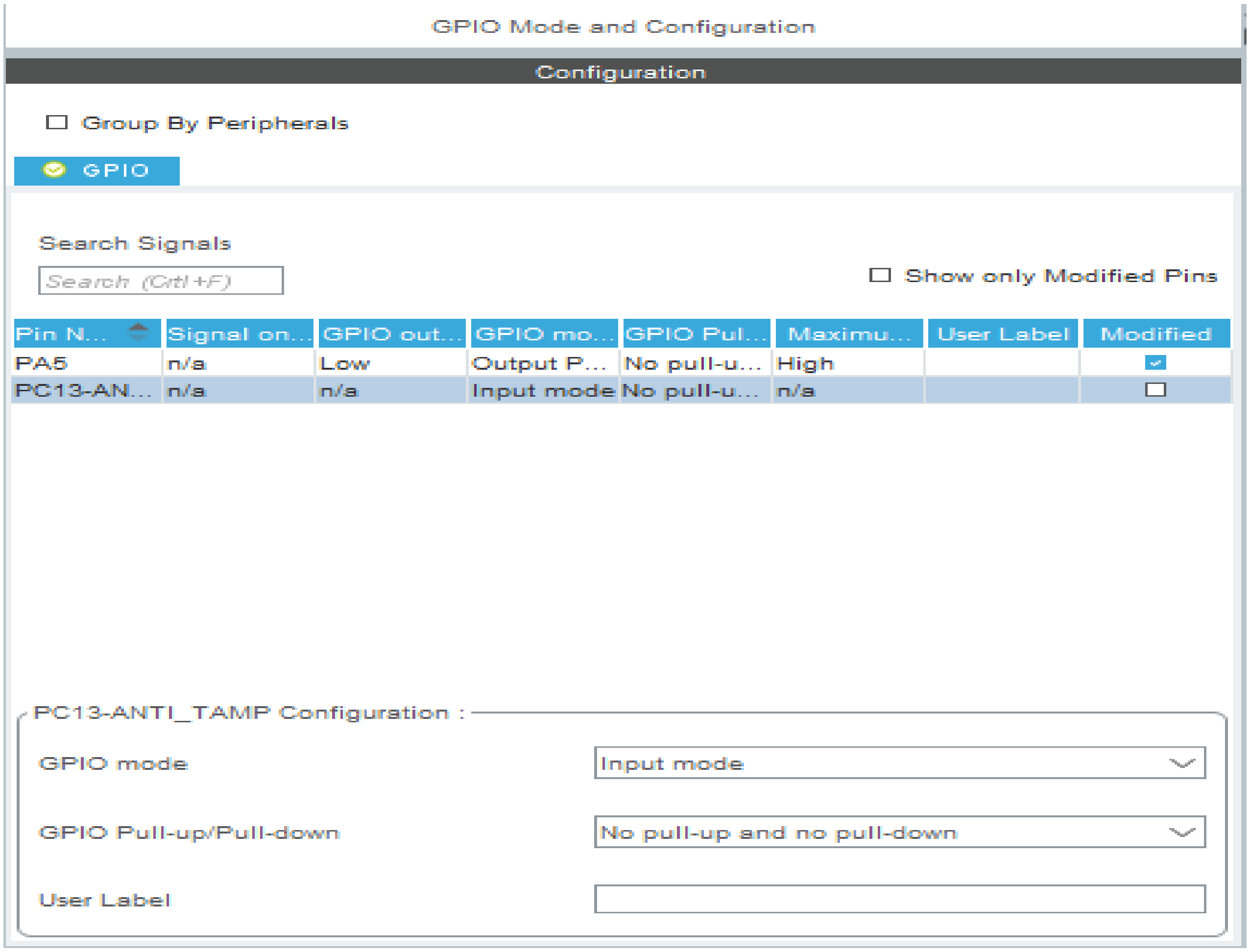
  Turn OFF –GPIO is LOW
  Turn ON –GPIO is HIGH

- ➢ Note: USER button (Blue)

  Button not press –GPIO is LOW
  Button press –GPIO is HIGH

- ➢ Hint –Pin PC7 can also be found by using [Find] feature in STM32CubeMx

Step 4 : Peripheral Configuration



GPIO Mode and Configuration

Configuration

□ Group By Peripherals

GPIO

Search Signals

Search (Crtl+F)

□ Show only Modified Pins

Pin N...	Signal on...	GPIO out...	GPIO mo...	GPIO Pul...	Maximu...	User Label	Modified
PA5	n/a	Low	Output P...	No pull-u...	High		☑
PC13-AN...	n/a	n/a	Input mode	No pull-u...	n/a		□

PC13-ANTI_TAMP Configuration :

GPIO mode: Input mode

GPIO Pull-up/Pull-down: No pull-up and no pull-down

User Label:

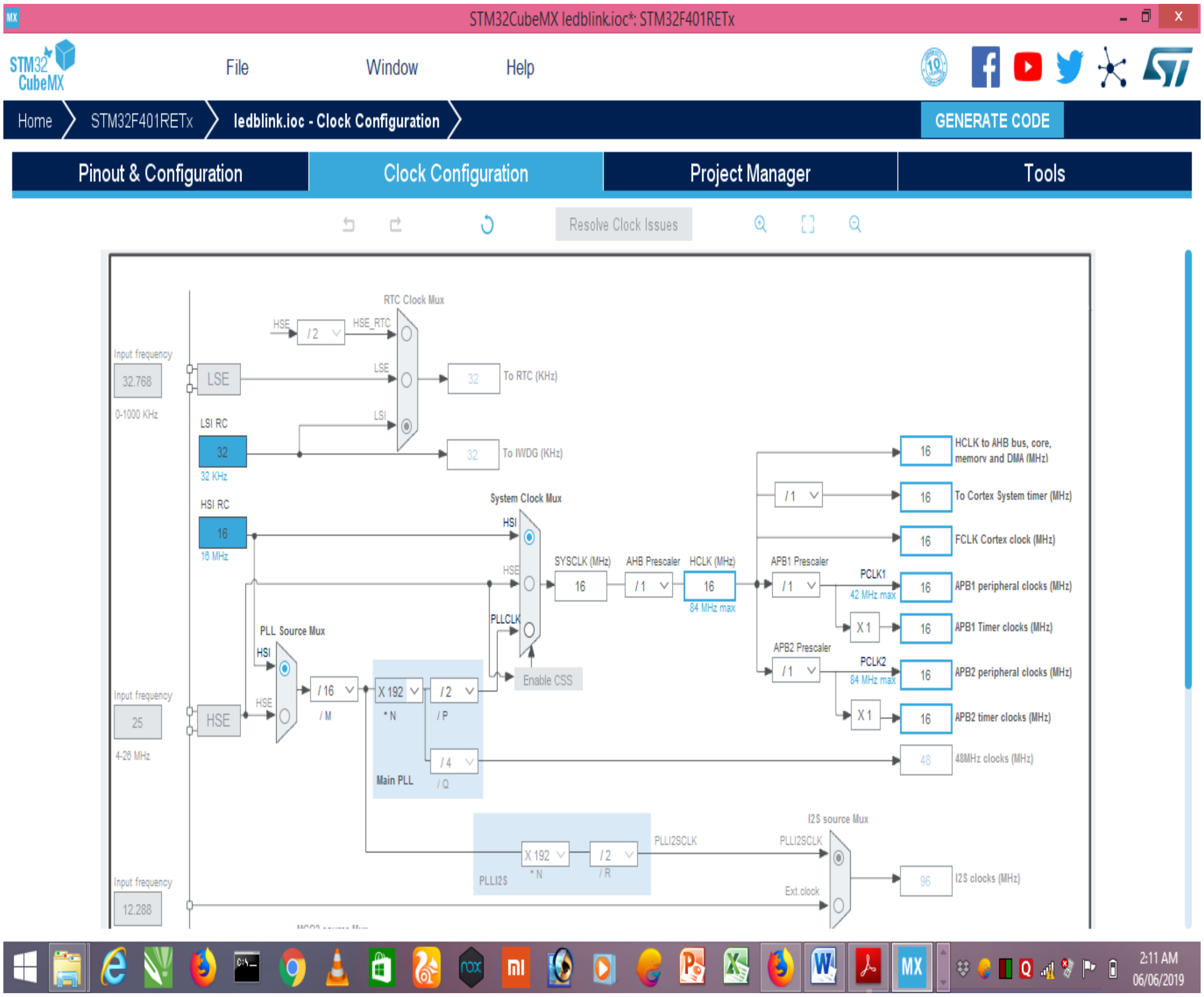
SystemView

- Select [GPIO]

- **Configure PA5**
- [GPIO Mode] : Output Push Pull
- [Maximum output speed] : High
- Other settings use default

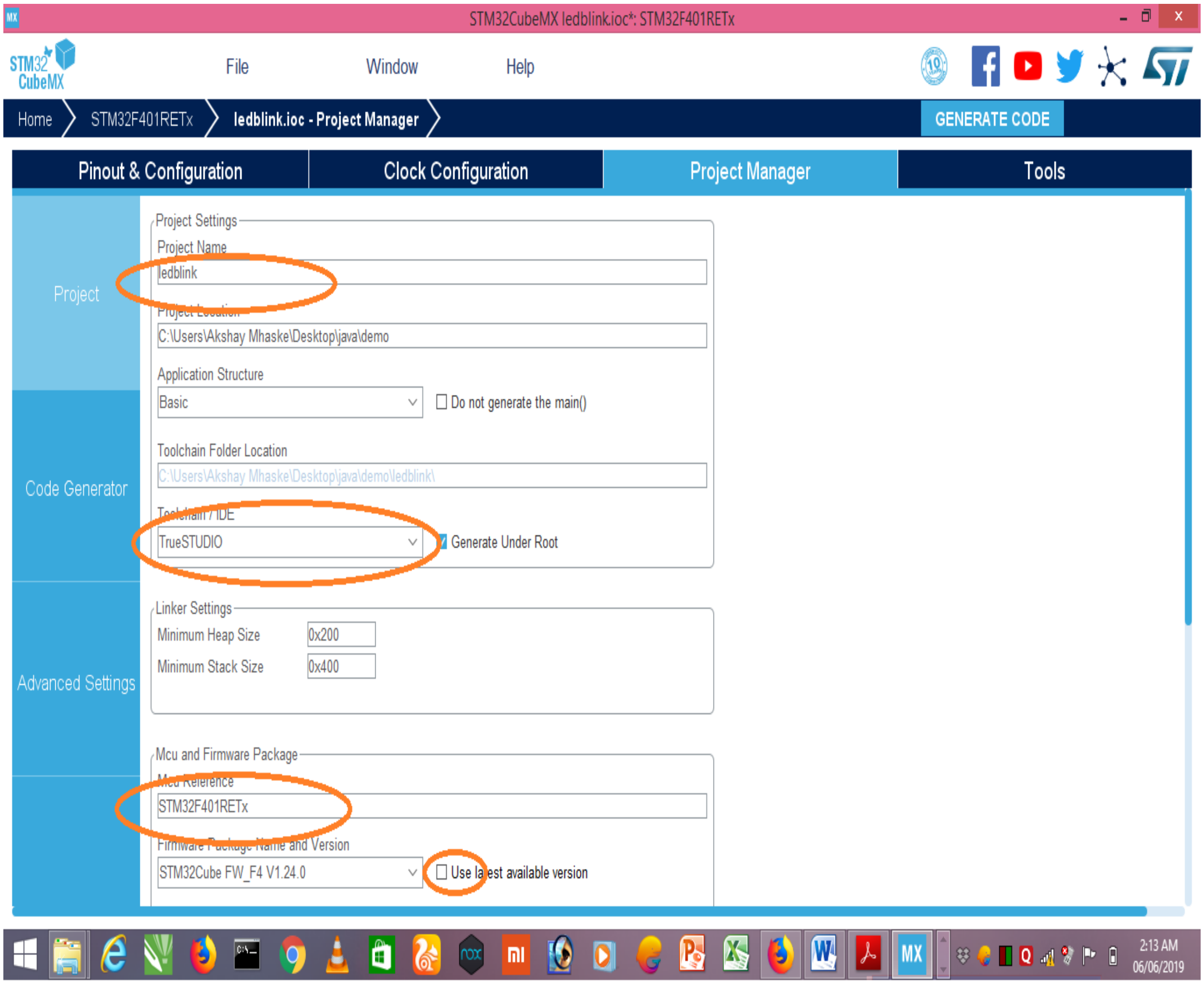
- **Configure PC13**
- [GPIO Mode] : Input Mode
- Other settings use default
- Click [Apply] and [OK]

Step 5 : Clock Configuration



- ➢ Click OK when [Clock Wizard] message pop out to automatically find the correct clock sources
- ➢ The appropriate clock source and PLL values will be set automatically

Step 6 : Project Settings



Project Settings

Project Name: ledblink

Project Location: C:\Users\Akshay Mhaske\Desktop\java\demo

Application Structure: Basic □ Do not generate the main()

Toolchain Folder Location: C:\Users\Akshay Mhaske\Desktop\java\demo\ledblink\

Toolchain / IDE: TrueSTUDIO ☑ Generate Under Root

Linker Settings

Minimum Heap Size: 0x200

Minimum Stack Size: 0x400

Mcu and Firmware Package

Mcu Reference: STM32F401RETx

Firmware Package Name and Version: STM32Cube FW_F4 V1.24.0 □ Use latest available version

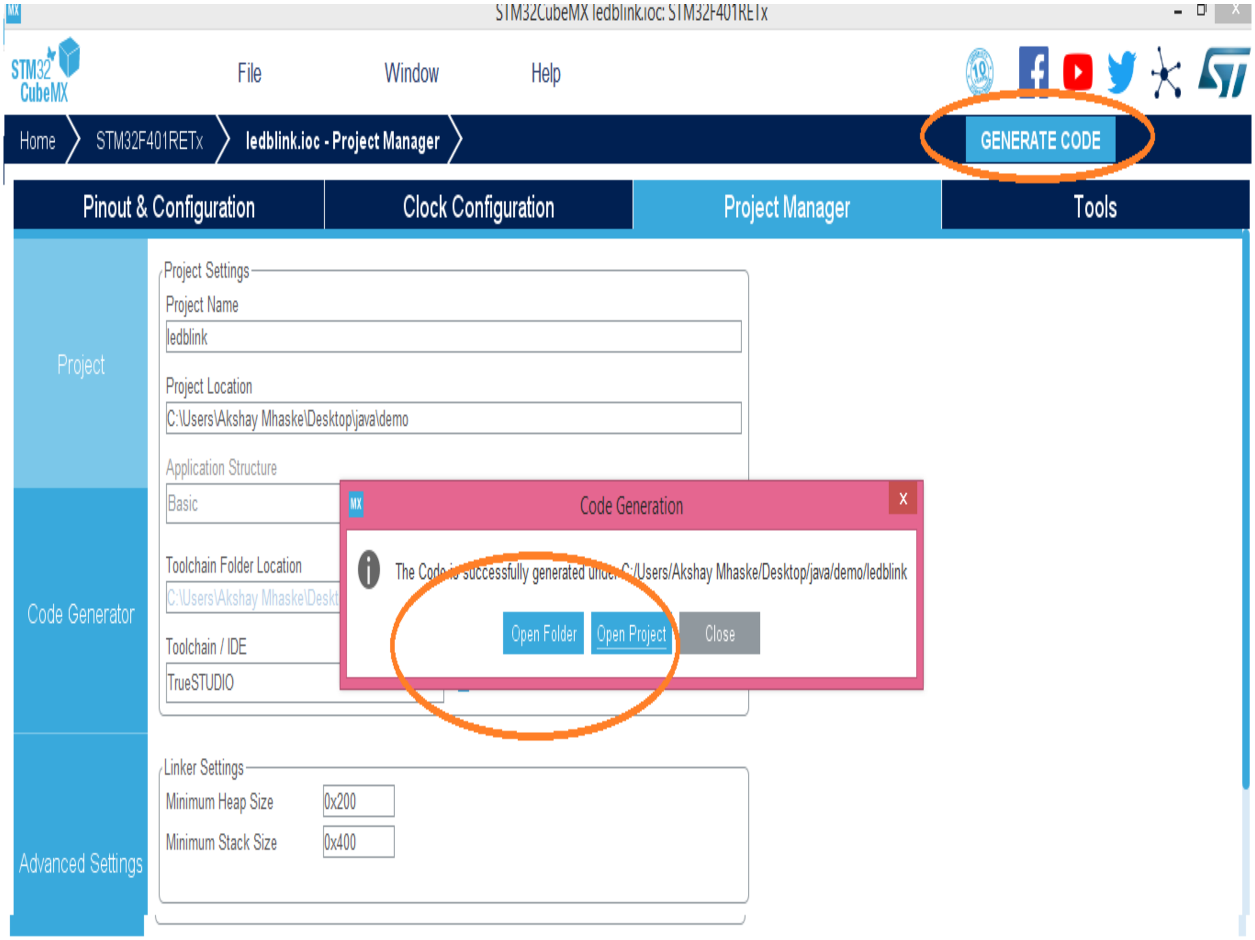
Configure project settings

- Select [Project->Settings…]
- [Project] tab
  - [Project Name]:Any name. For example LED_Toggle
  - [ProjectLocation]:Locationtostoreprojectfolders.InthecaseofAtollicTrueSTUDIO,theworkspacefolderlocation.ForexampleC:\STM32F4+Workshop\workspace
  - [Application Main Location]:Src

•[Tool chain Folder Location]:Will automatically begenerated
•[Tool chain/IDE]:True STUDIO
•[Generate Under Root]:Default

- [Code Generator] tab
  •Keep default configuration
- Click [OK] to finish

Step 7 : Generate Code



Project Settings

Project Name: ledblink

Project Location: C:\Users\Akshay Mhaske\Desktop\java\demo

Application Structure: Basic

Toolchain Folder Location: C:\Users\Akshay Mhaske\Deskt

Toolchain / IDE: TrueSTUDIO

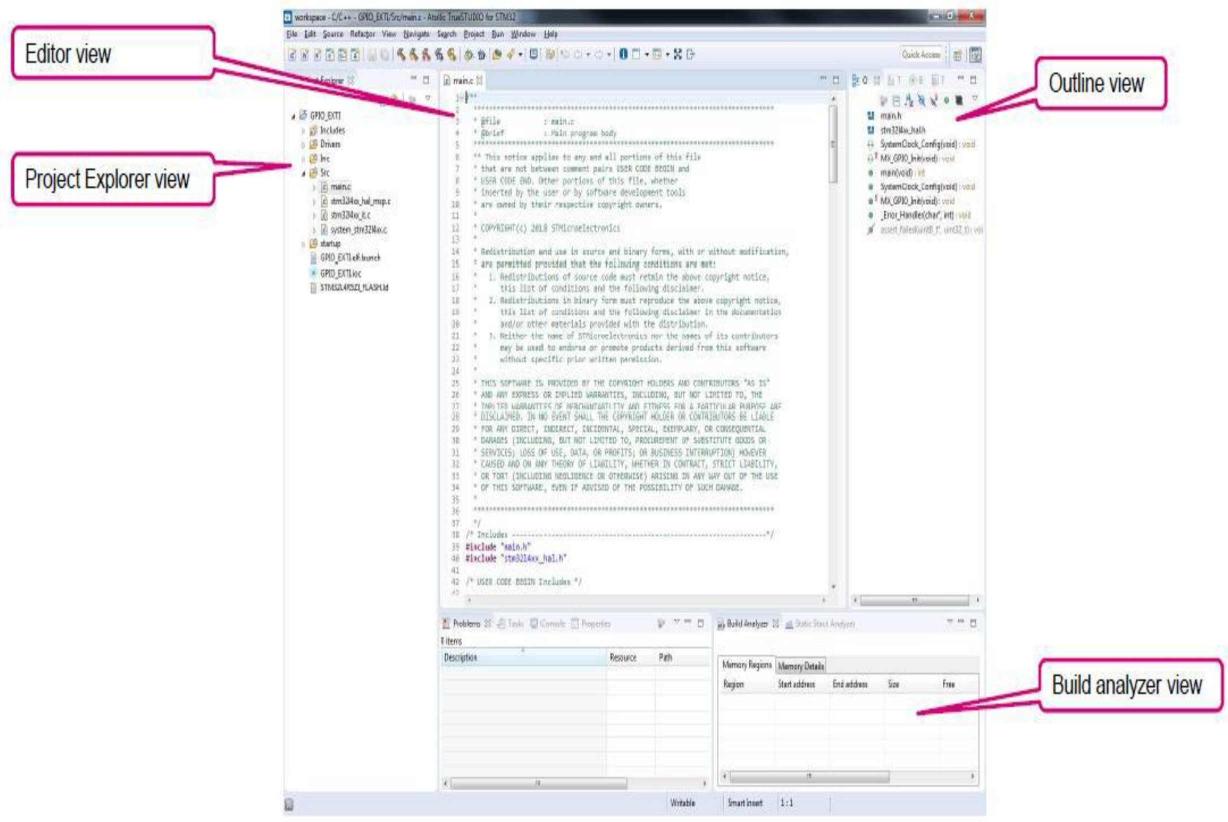
The Code is successfully generated under C:/Users/Akshay Mhaske/Desktop/java/demo/ledblink

Linker Settings

Minimum Heap Size: 0x200

Minimum Stack Size: 0x400

Step 8 : Starting TrueStudio



```
/**
  ******************************************************************************
  * @file           : main.c
  * @brief          : Main program body
  ******************************************************************************
  ** This notice applies to any and all portions of this file
  * that are not between comment pairs USER CODE BEGIN and
  * USER CODE END. Other portions of this file, whether
  * inserted by the user or by software development tools
  * are owned by their respective copyright owners.
  *
  * COPYRIGHT(c) 2018 STMicroelectronics
  *
  * Redistribution and use in source and binary forms, with or without modification,
  * are permitted provided that the following conditions are met:
  *   1. Redistributions of source code must retain the above copyright notice,
  *      this list of conditions and the following disclaimer.
  *   2. Redistributions in binary form must reproduce the above copyright notice,
  *      this list of conditions and the following disclaimer in the documentation
  *      and/or other materials provided with the distribution.
  *   3. Neither the name of STMicroelectronics nor the names of its contributors
  *      may be used to endorse or promote products derived from this software
  *      without specific prior written permission.
  *
  * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
  * AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
  * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
  * DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
  * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
  * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
  * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
  * CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
  * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
  * OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
  *
  ******************************************************************************
  */
/* Includes ------------------------------------------------------------------*/
#include "main.h"
#include "stm32f4xx_hal.h"

/* USER CODE BEGIN Includes */
```

▶Open the Project

▶Go to src

▶Double Click the main.cfile

▶You will get a window as shown.

▶This is the code generated by STN32CubeMX

▶We need to modify the code for hands on for GPIO.

Open project folder name→src folder→main.c

Step 9 : Modify Generated Code

[Note: Do changes in main.c]

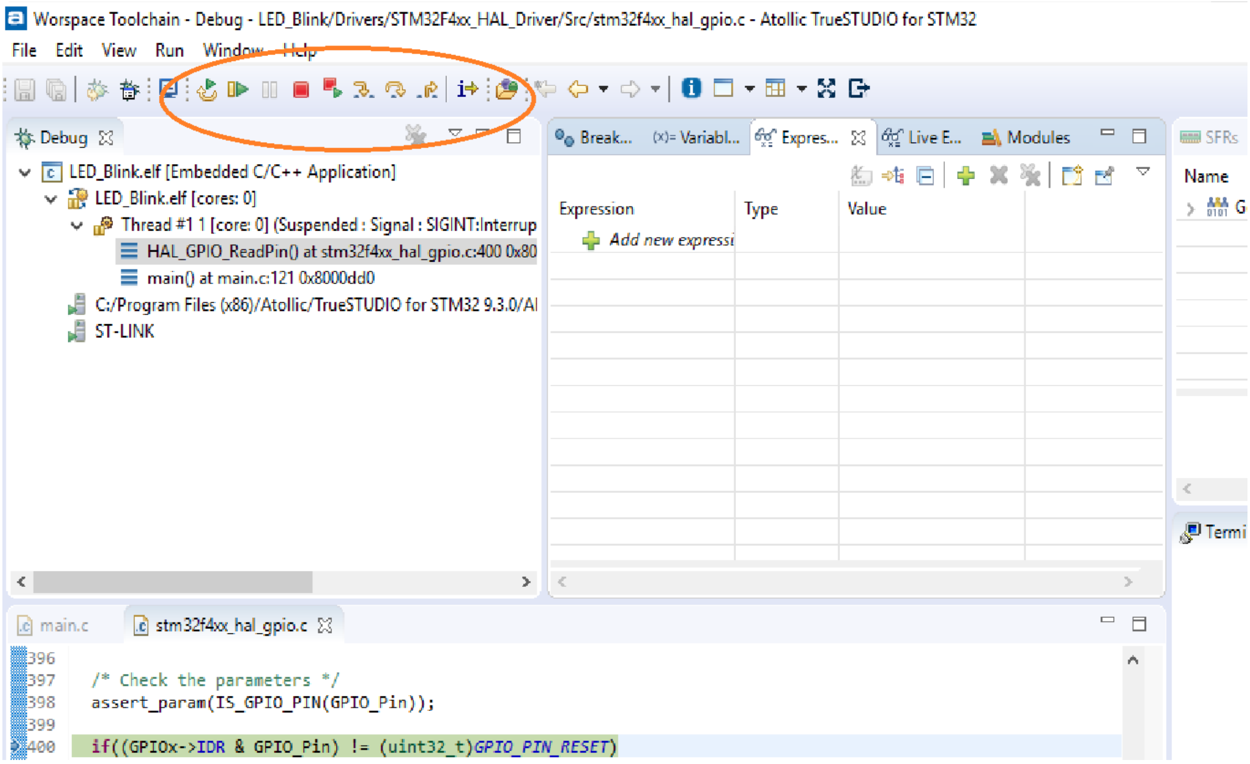
```
***************************************************************************
***************************************************************************
/* USER CODE BEGIN 2 */
// Resetingthe On Board LED which is on PA5
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_5, GPIO_PIN_RESET);
/* USER CODE END 2 */
/* Infinite loop */
/* USER CODE BEGIN WHILE */
while(1)
{
//monitoring the status of on board user button connected to PC13
GPIO_PinStatebtn_state= HAL_GPIO_ReadPin(GPIOC, GPIO_PIN_13);
if(btn_state== GPIO_PIN_RESET)
{
HAL_GPIO_TogglePin(GPIOA, GPIO_PIN_5);
HAL_Delay(100);
}
/* USER CODE END WHILE */
                          /* USER CODE BEGIN 3 */

                                     }

***************************************************************************
***************************************************************************
```

- Please take care about the beginning and ending of USER CODE block mentioned in the code.
- Code should be modified within these blocks as shown >>>>>>>
- Save the code by Ctrl+S
- Click on Project Menu and select Build Project option from drop down menu.
- Once completed, click on Run Menu and select Debug to start debugging session.
- Result will be reflected on the board after clicking Resume button.
- Press the Blue button on the board an see the state of LED.

Step 10 -Debugging Controls



Worspace Toolchain - Debug - LED_Blink/Drivers/STM32F4xx_HAL_Driver/Src/stm32f4xx_hal_gpio.c - Atollic TrueSTUDIO for STM32

```
LED_Blink.elf [Embedded C/C++ Application]
  LED_Blink.elf [cores: 0]
    Thread #1 1 [core: 0] (Suspended : Signal : SIGINT:Interrup
      HAL_GPIO_ReadPin() at stm32f4xx_hal_gpio.c:400 0x80
      main() at main.c:121 0x8000dd0
  C:/Program Files (x86)/Atollic/TrueSTUDIO for STM32 9.3.0/Al
  ST-LINK
```

Expression	Type	Value
Add new expressi		

```

  /* Check the parameters */
  assert_param(IS_GPIO_PIN(GPIO_Pin));

  if((GPIOx->IDR & GPIO_Pin) != (uint32_t)GPIO_PIN_RESET)
```

Thank You…

Special Thanks to Nikhil Sir (Founder of IoTIoT.in) and also Tevatron Technologies Pvt.LTD

All efforts done by Akshay Mhaske and IoTIoT-BLE team. Documents are always open source to Tech Enthusiats