Lab 2: EXTI, NVIC and RCC

Activity 1: EXTI Configuration

Aim: Configuring GPIO as an external interrupt on STM32L476VG MCU.

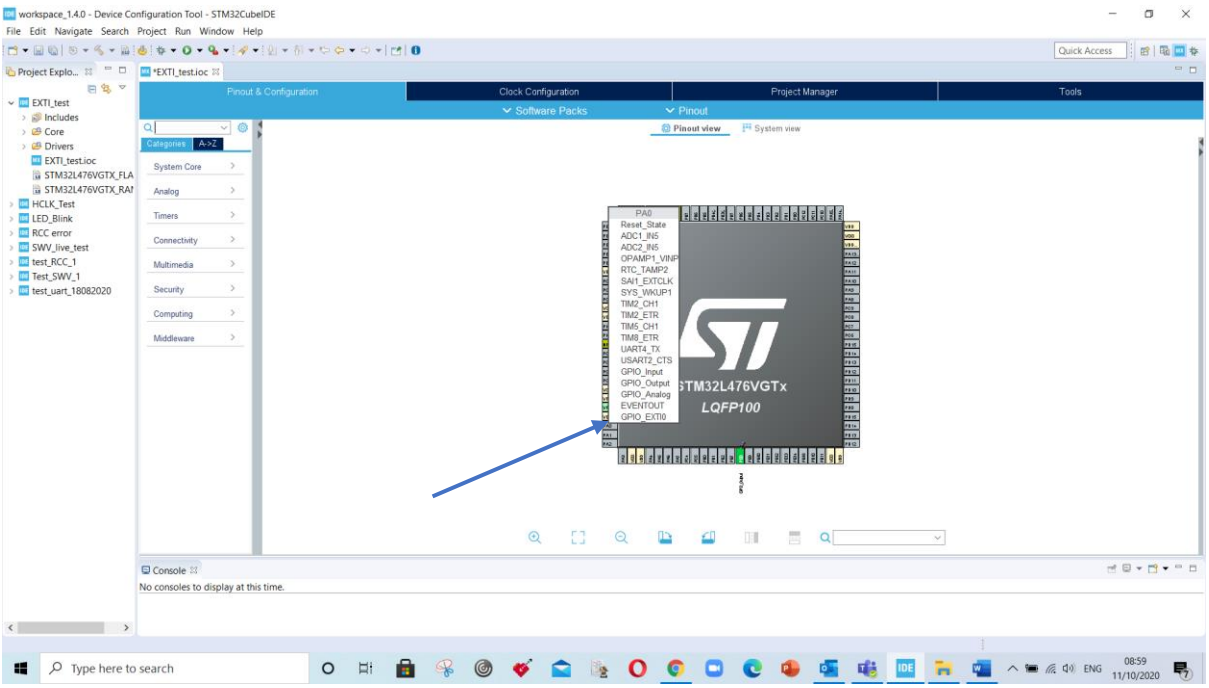
Objectives:

1- Learn how to configure the external interrupt controller for a GPIO
2- Declaring the callback function and its prototype
3- Testing the functionality by toggling LED using external interrupt

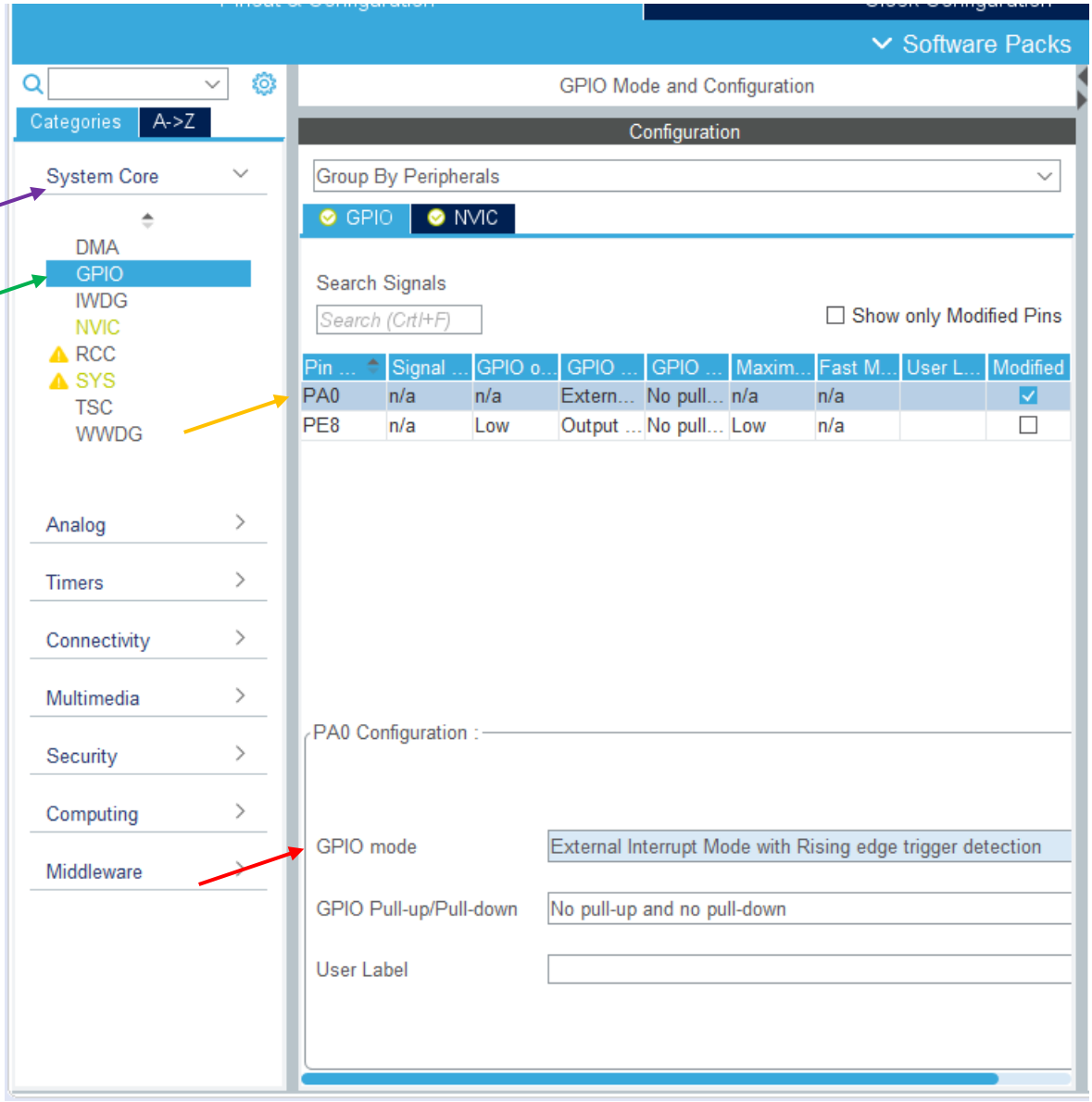
---

Step 1: Create a project in STM32CubeIDE and select 'PE8' as output mode, refer to previous lab if you are not sure how to do it.

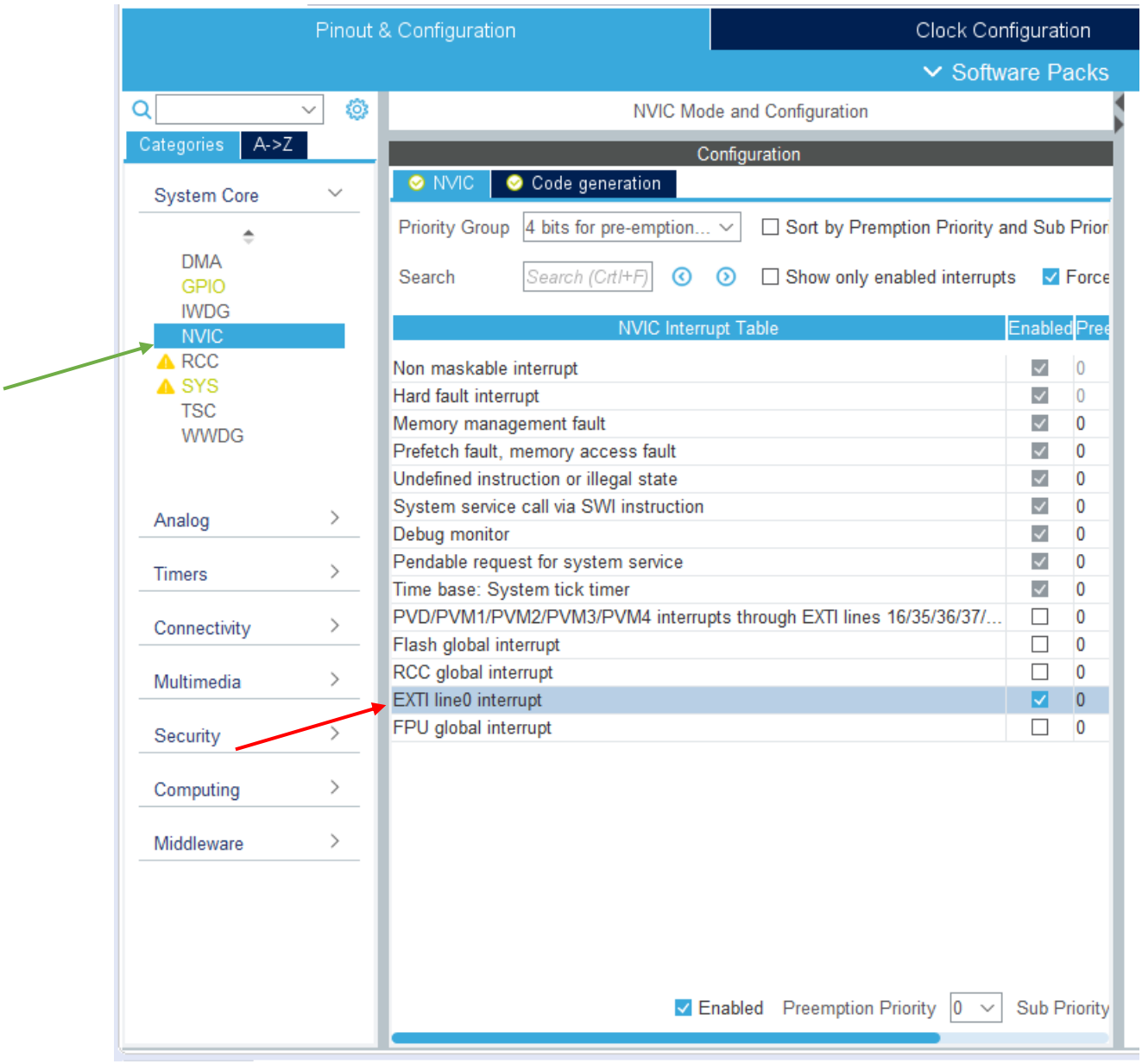
Step 2: Left click on 'PA0' and select 'GPIO_EXTI0' to configure it as external interrupt. This PIN is attached to center button of joystick.



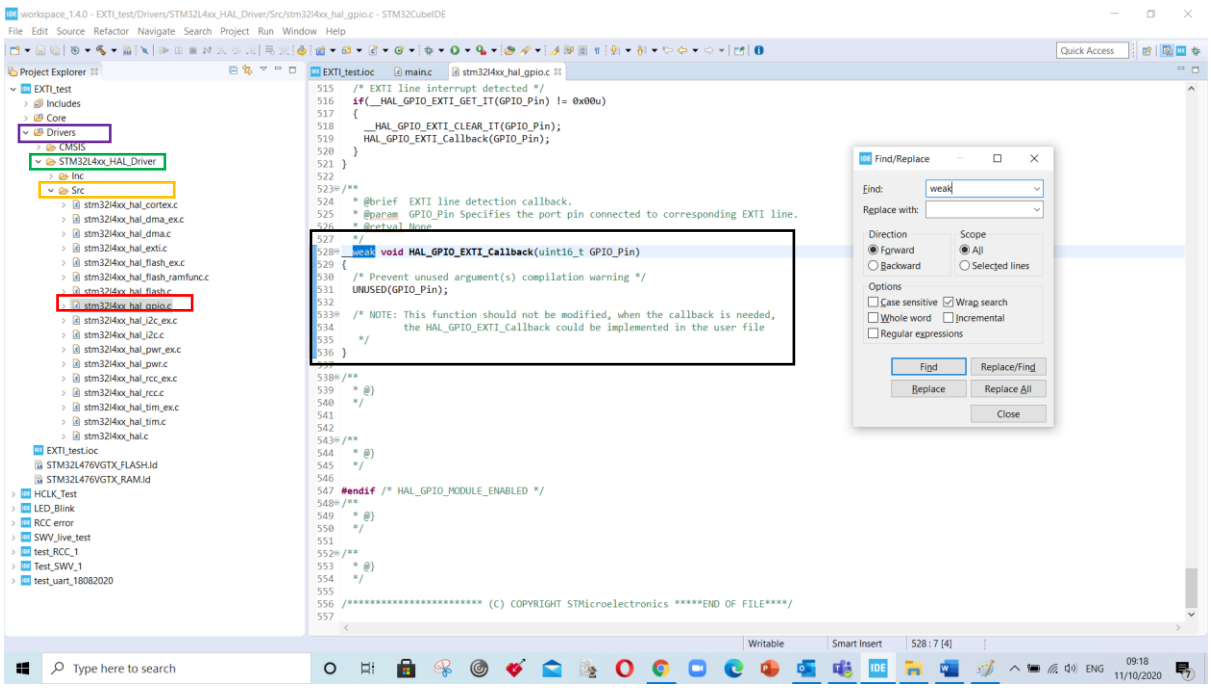
Step 3: Go to 'System Core', select 'GPIO', choose pin 'PA0' and in '**GPIO mode'** select 'External Interrupt Mode with Rising edge trigger detection'



Step 4: From 'System Core' menu, select 'NVIC' and check the 'EXTI line0 interrupt'



Step 5: *Generate the code* and in files select folders 'Driver', then select folder 'STM32L4xx_HAL_Driver', then select 'Src' and finally select 'Stm32l4xx_hal_gpio.c'. Now find the *WEAK* of instance function '**HAL_GPIO_EXTI_callback'**.



Copy and paste the code in the 'main.c' file in area of /* USER CODE BEGIN 4 */, as below:

```
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{

}
```

Create/Declare the function prototype in main.c file, as shown below. (See Appedix A)

```
/* USER CODE BEGIN PFP */
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin);
```

In external interrupt function callback add the GPIO PE8 toggle code: (See Appedix B)

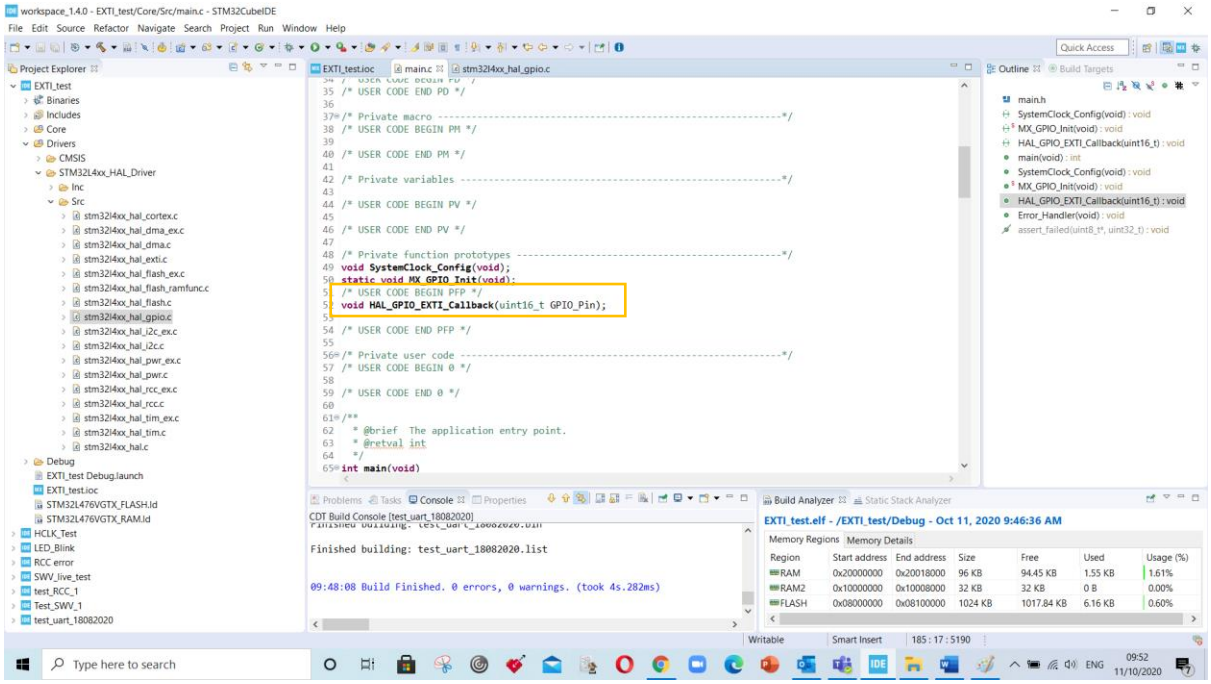
```
if (GPIO_Pin == GPIO_PIN_0)
{
        HAL_GPIO_TogglePin(GPIOE, GPIO_PIN_8);
}
```

Note: We have use if statement to verify that this call back was generated by our desired pin GPIO PA0.

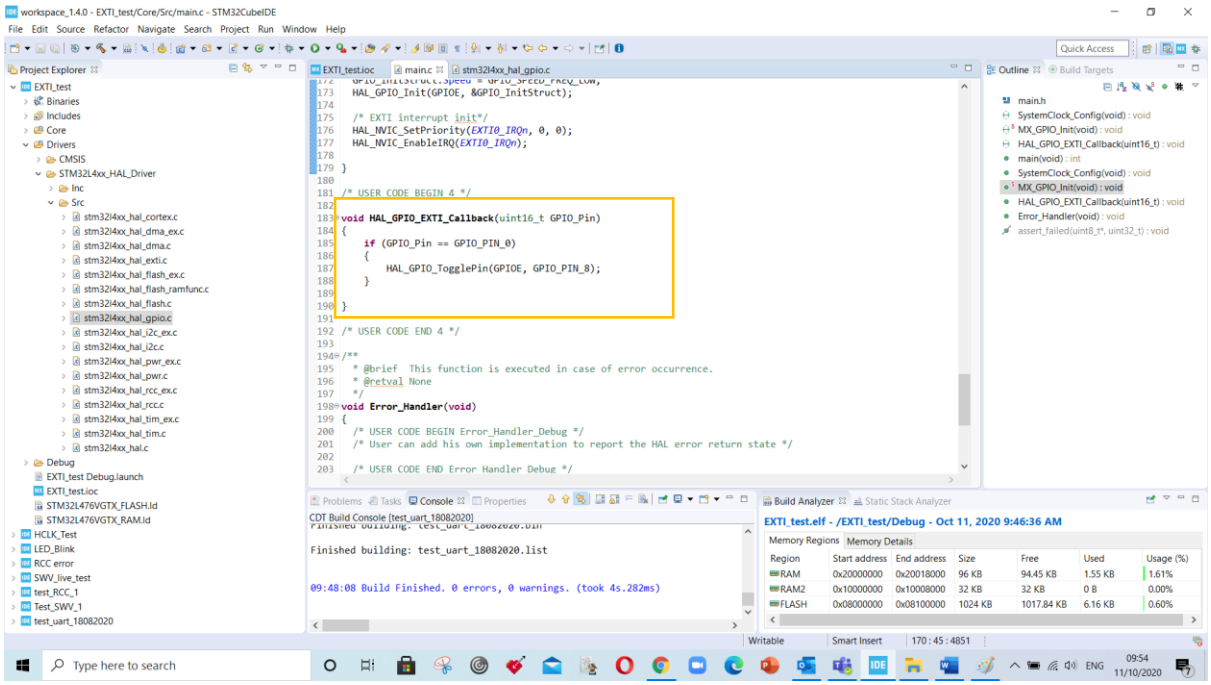
Run '  ' and build the code. You will not find any error or warning messages. Now test your code by toggling the LED using joystick middle button.

To do: Blink the Red LED on STM32L476_DISCO board with delay of 500ms and toggle the Green LED using External interrupt through joystick.

Appendix A: Prototype function declaration



Appendix B: External interrupt call back function



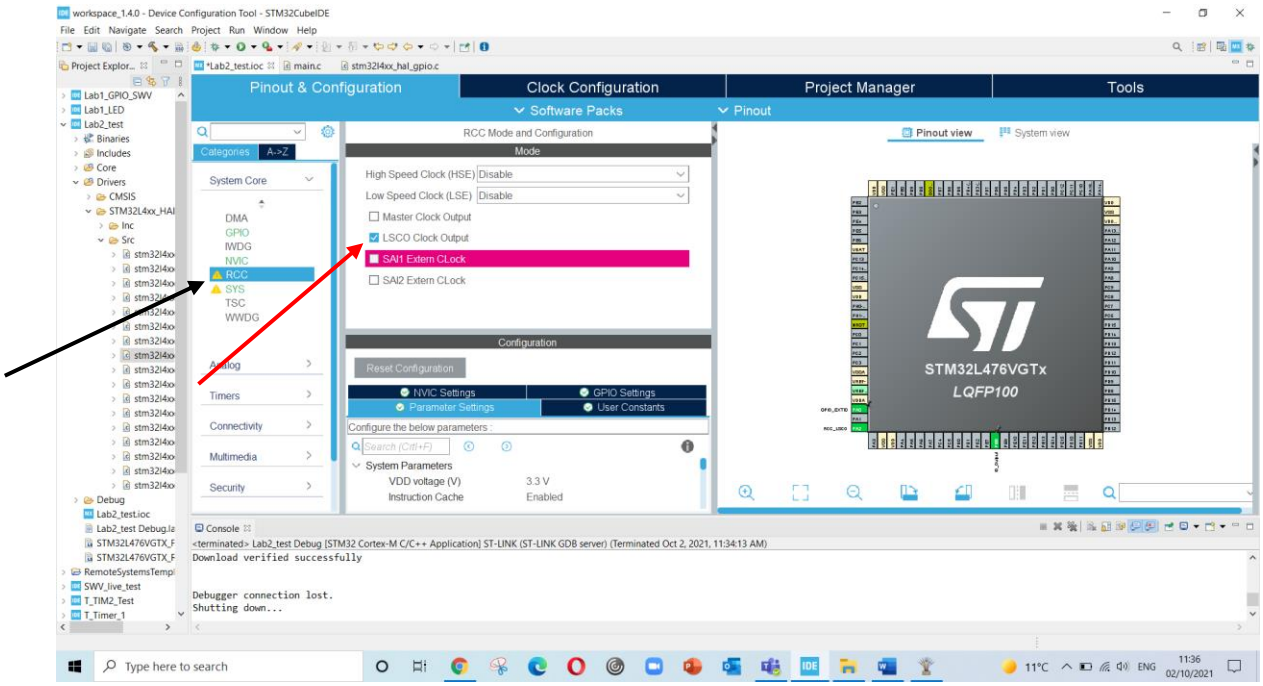
Activity 2: RCC configuration

Aim: To configure Reset and clock controller to generate a Low speed clock output (LSCOCO) on STM32L476VG MCU.

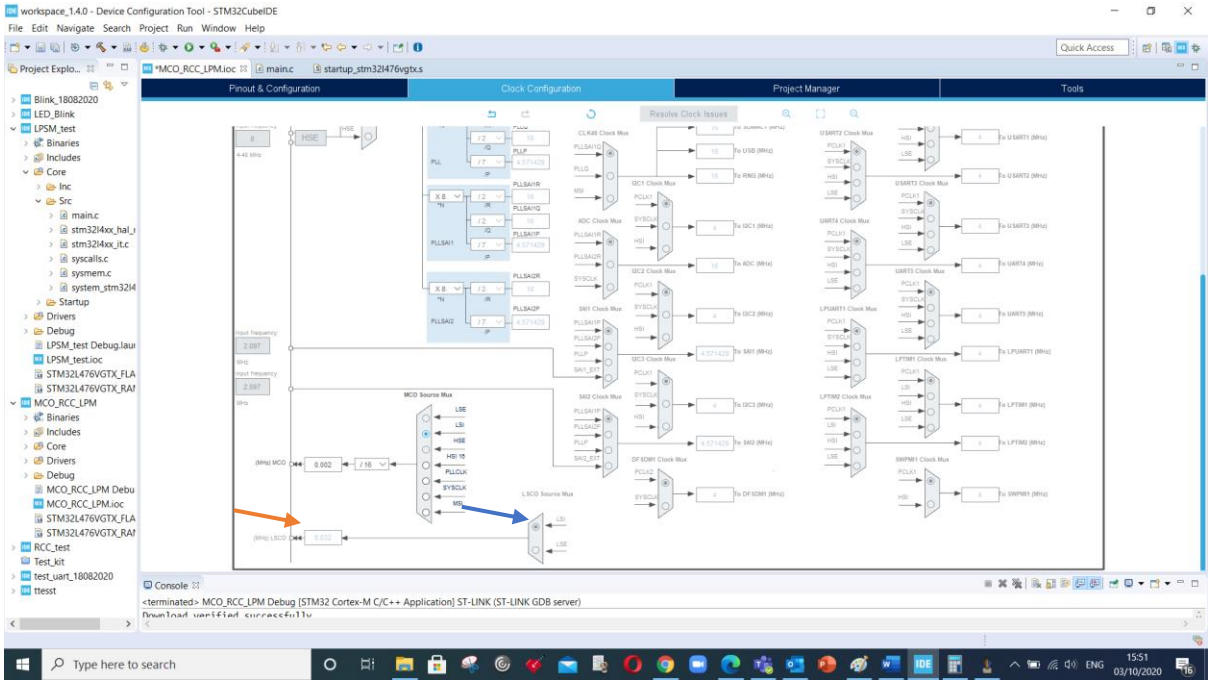
Objectives:

1- Generate a clock output on Pin PA2 for external circuitry
2- Observer the Waveform on function Generator

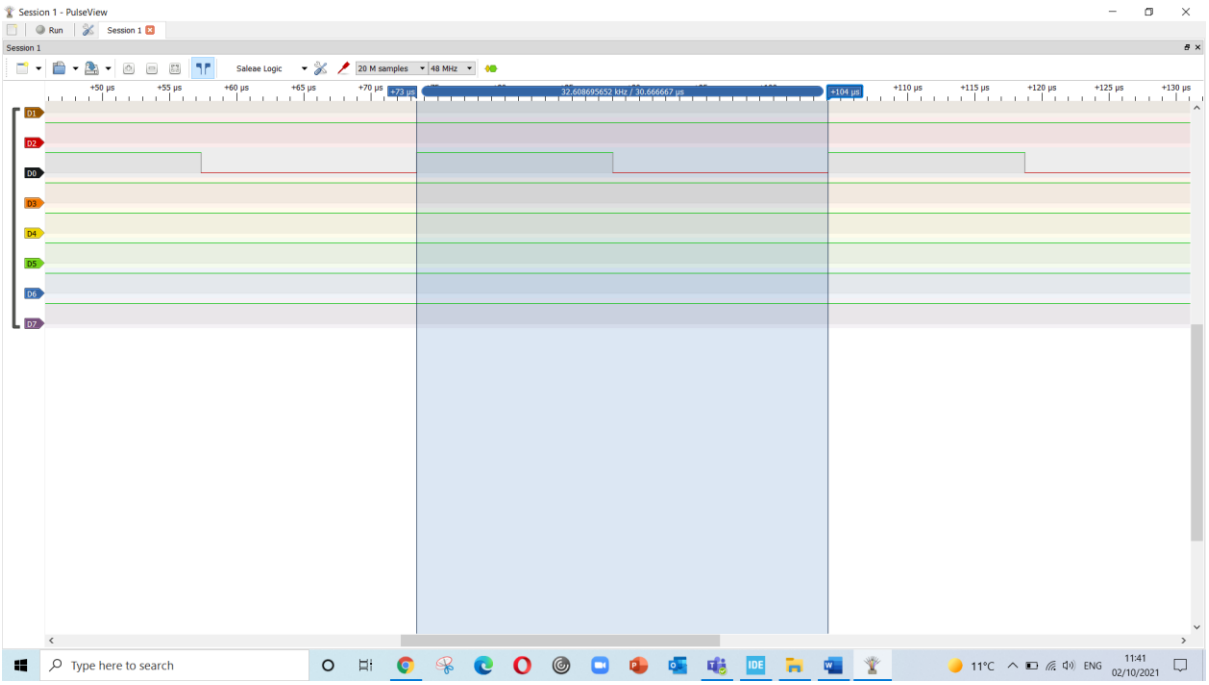
Step 1: In the 'System Core' select the '**RCC**' and check the 'LSCO Clock Output'



Step 2: By default, ''LSI' will be clock source for LSCO at frequency of 32kHz.



To do: Capture the waveform in logic analyzer on pin ‘PA2‘ and measure the waveform frequency.



Activity 3: SWV trace analysis

Aim: To debug the signal on STM32L476VG MCU inside STM32CubeIDE.

Objectives:

1- Generate a clock output on Pin PE8.
2- Configure SWV trace analysis
3- Observer the Waveform inside STM32CubeIDE

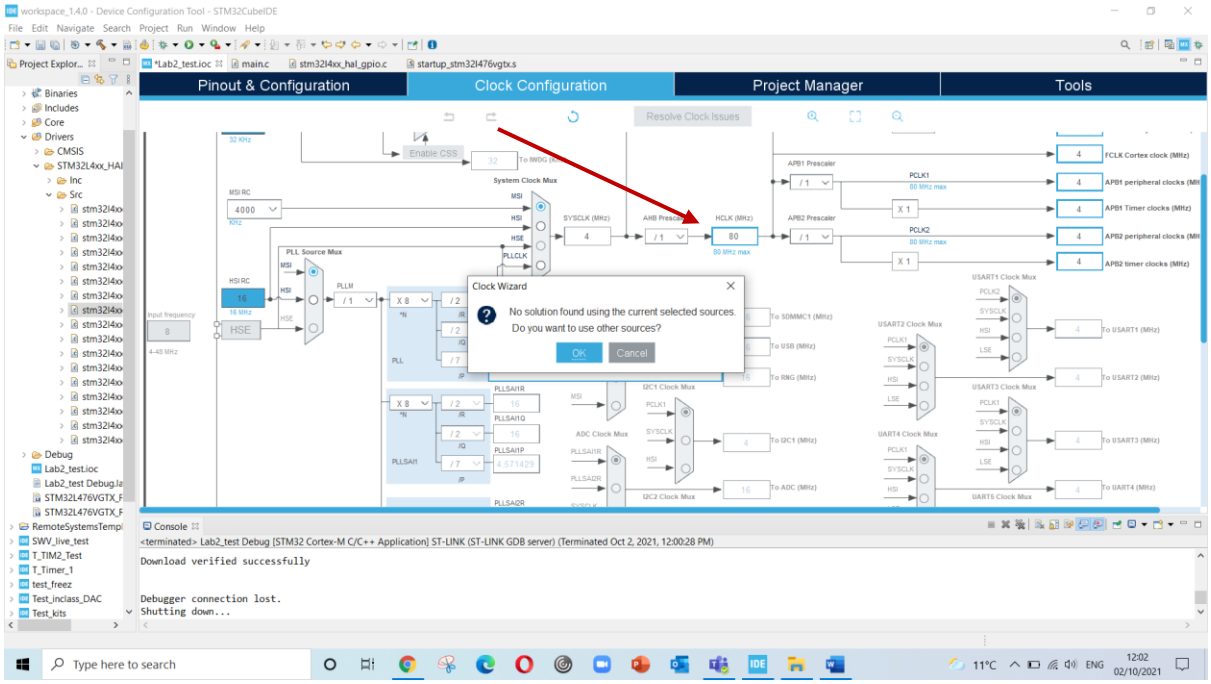
Step 1: Configure GPIO PB2 as output and connect SWO and PB3 on STM32L4 DISCO board.

Step 2: use the following code to toggle led and once the code is generated.

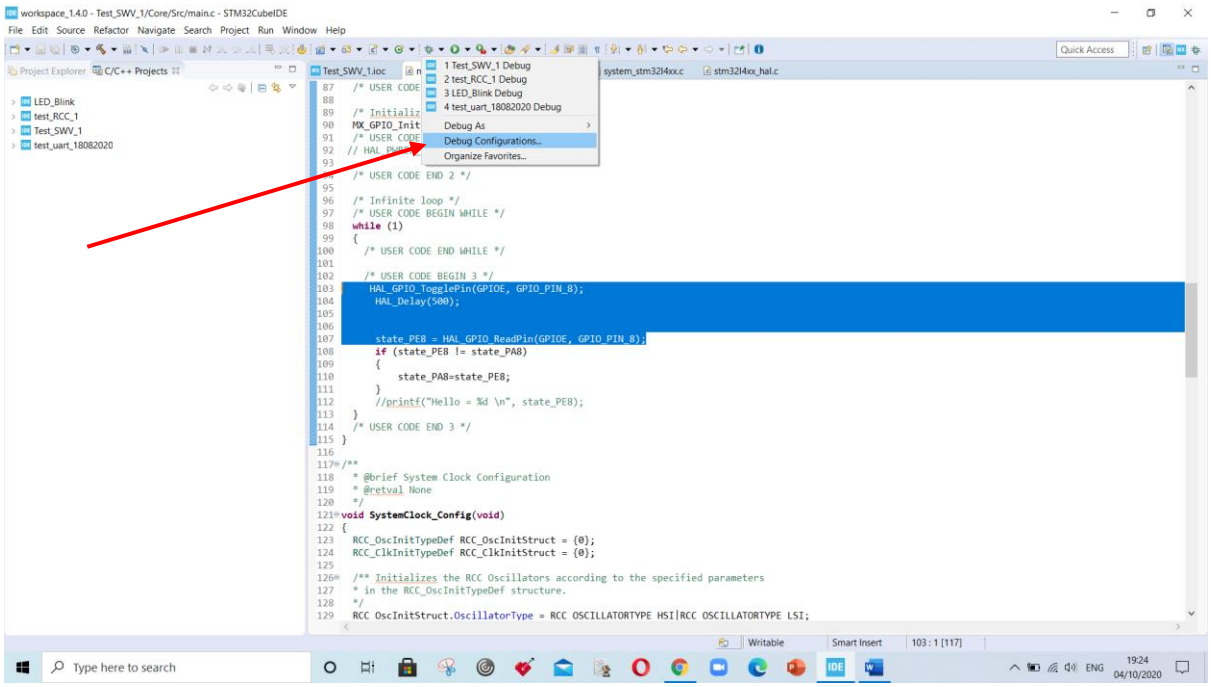
```
/* USER CODE BEGIN 1 */
    int state_PB2 = 0;
```

```
/* USER CODE BEGIN 3 */
    HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_2);
    HAL_Delay(500);
  state_PB2 = HAL_GPIO_ReadPin(GPIOB, GPIO_PIN_2);
```

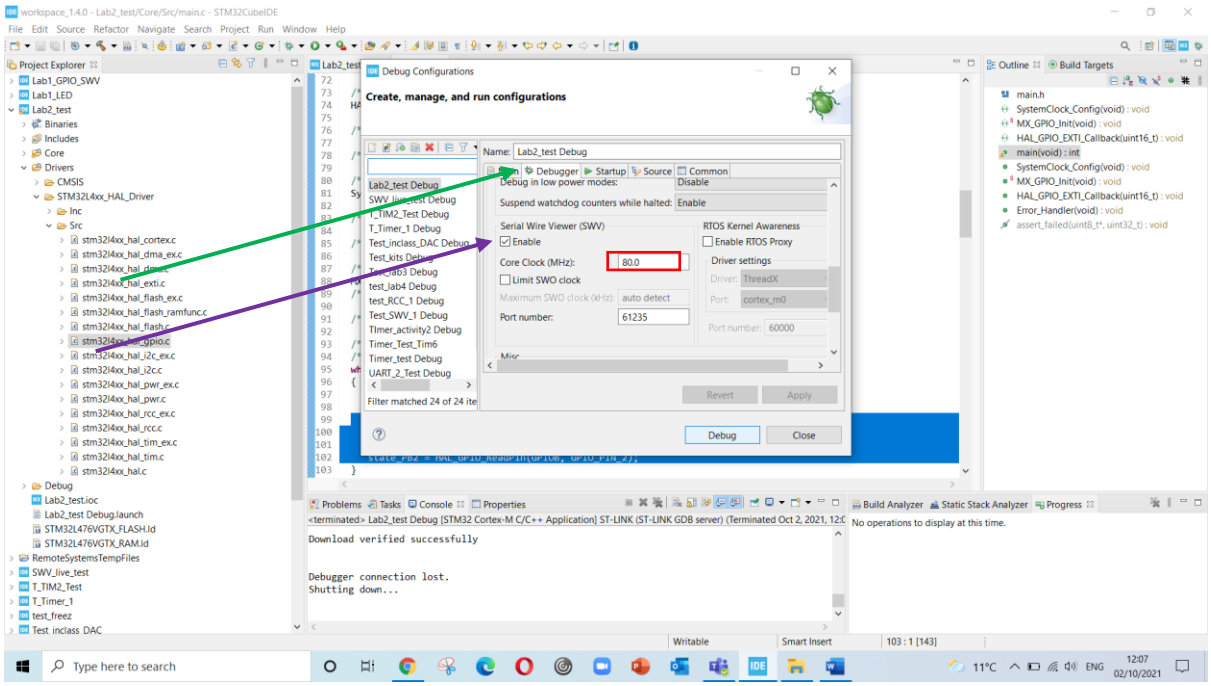
Step 3: Set HCLK to 80 MHz and press Enter Key then it will prompt the ‘CLOCK WIZARD’ window and press OK.

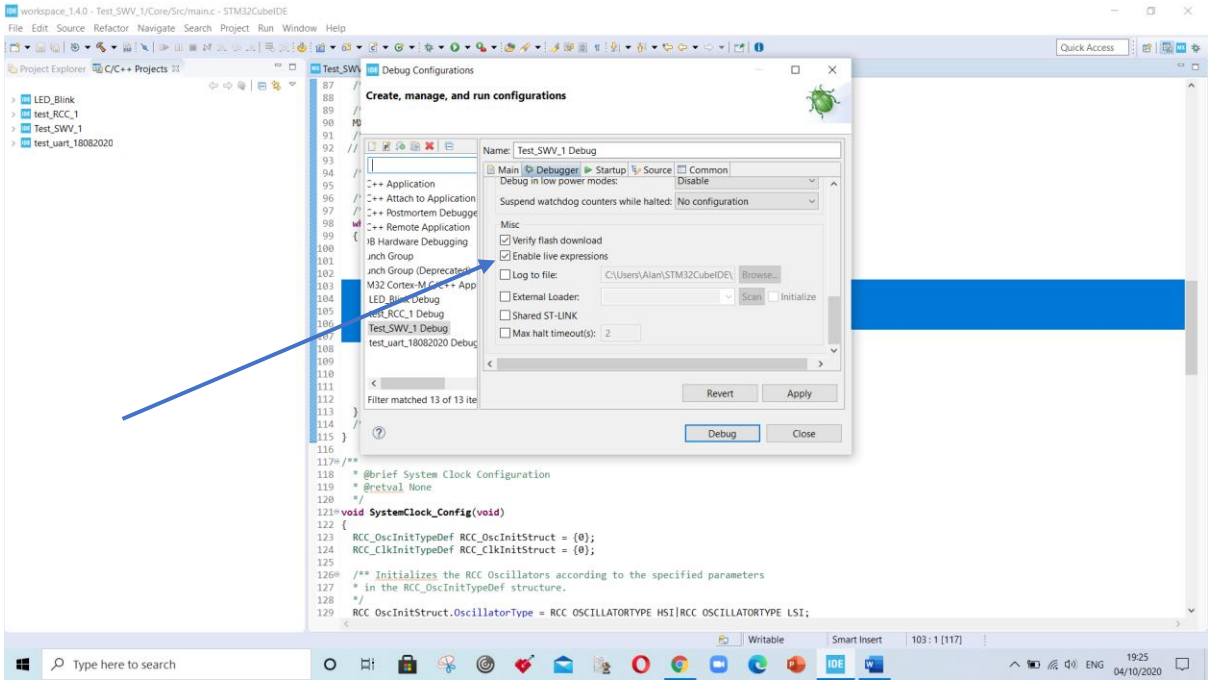


Build the project by using Run button then go to ‘Debug Configuration’.

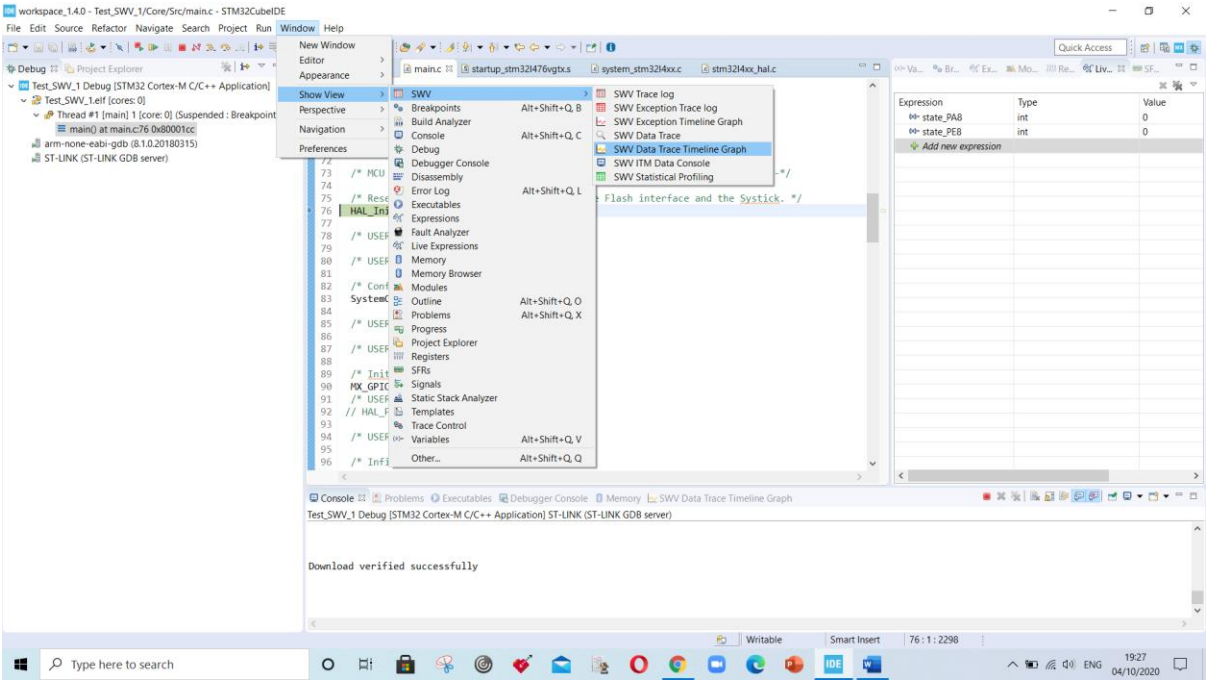


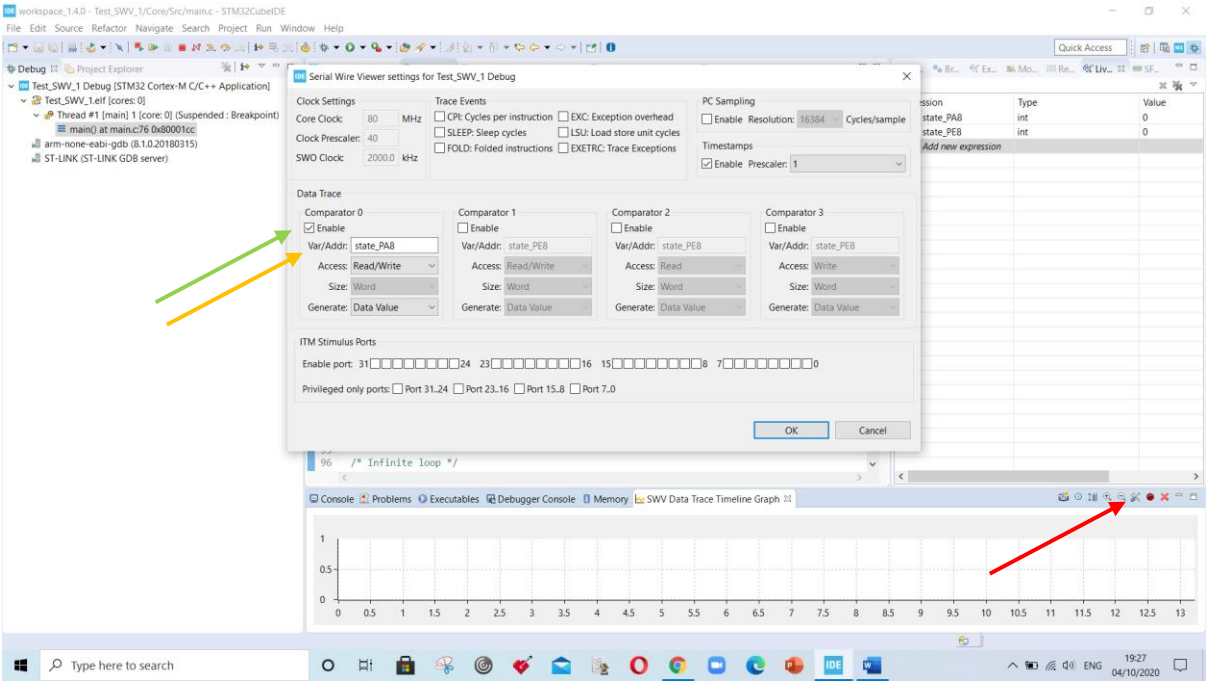
Step 4: Go to Debugger tab, scroll down to Serial Wire Viewer then check Enable and add the 80 in core clock. Finally check Enable Live Expresssion option scrolling down the window.



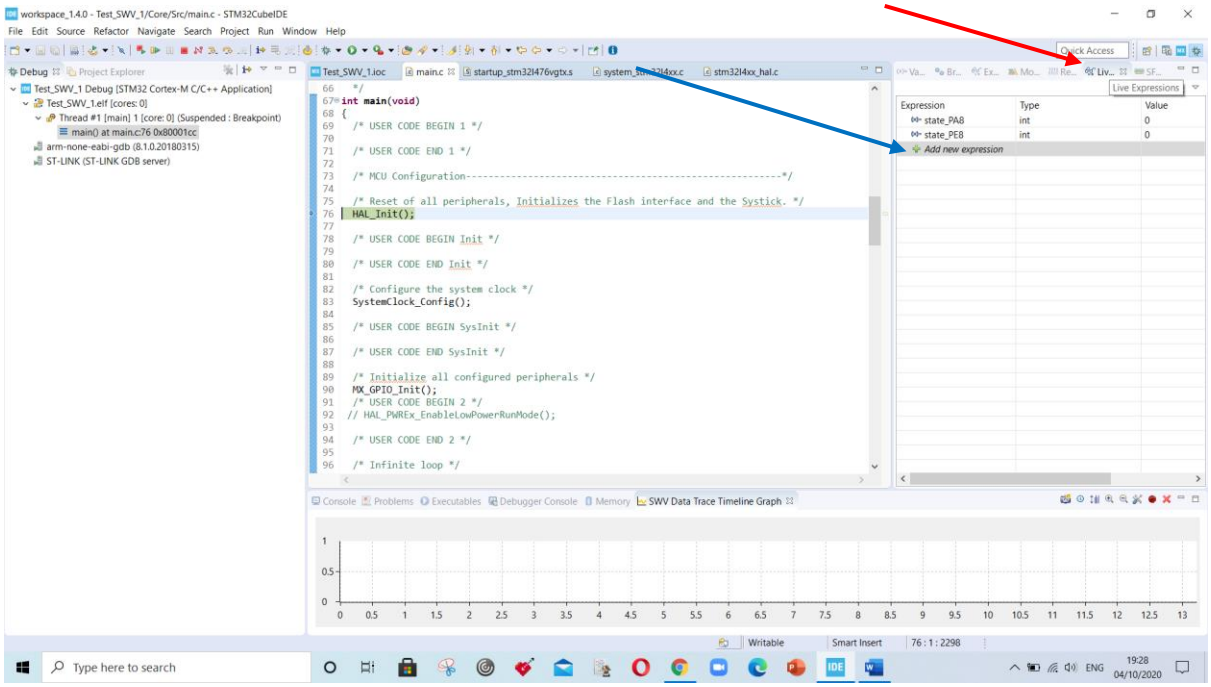


Step 5: Select Window->Show View->SWV -> SWV Data Trace Timeline graph. Then Click on configuration option and Enable comparator 0 and add variable name **'state_PB2'**.

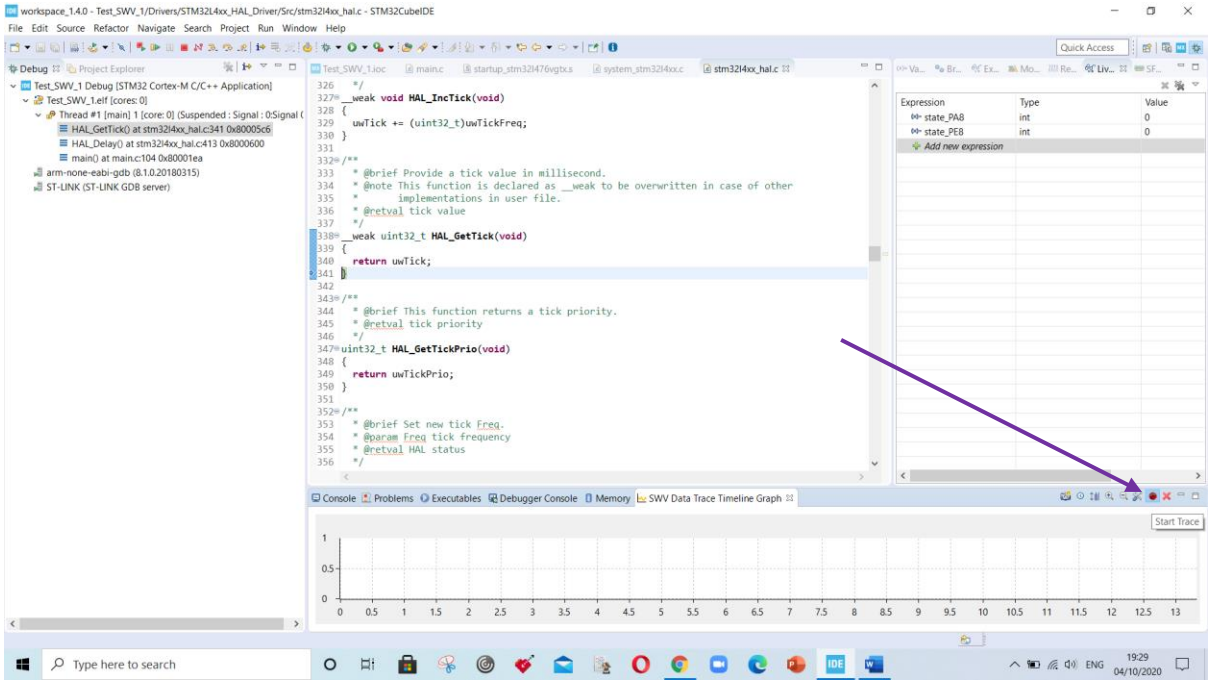

workspace_1.4.0 - Test_SWV_1/Core/Src/main.c - STM32CubeIDE
File Edit Source Refactor Navigate Search Project Run Window Help
New Window
Editor
Appearance
Show View
Perspective
Navigation
Preferences
SWV
Breakpoints Alt+Shift+Q, B
Build Analyzer
Console Alt+Shift+Q, C
Debug
Debugger Console
Disassembly
Error Log Alt+Shift+Q, L
Executables
Expressions
Fault Analyzer
Live Expressions
Memory
Memory Browser
Modules
Outline Alt+Shift+Q, O
Problems Alt+Shift+Q, X
Progress
Project Explorer
Registers
SFRs
Signals
Static Stack Analyzer
Templates
Trace Control
Variables Alt+Shift+Q, V
Other... Alt+Shift+Q, Q
SWV Trace log
SWV Exception Trace log
SWV Exception Timeline Graph
SWV Data Trace
SWV Data Trace Timeline Graph
SWV ITM Data Console
SWV Statistical Profiling
Debug
Test_SWV_1 Debug [STM32 Cortex-M C/C++ Application]
Test_SWV_1.elf [cores: 0]
Thread #1 [main] 1 [core: 0] (Suspended : Breakpoint
main() at main.c:76 0x80001cc
arm-none-eabi-gdb (8.1.0.20180315)
ST-LINK (ST-LINK GDB server)
Expression Type Value
state_PA8 int 0
state_PE8 int 0
Add new expression
Console Problems Executables Debugger Console Memory SWV Data Trace Timeline Graph
Test_SWV_1 Debug [STM32 Cortex-M C/C++ Application] ST-LINK (ST-LINK GDB server)
Download verified successfully
Writable Smart Insert 76 : 1 : 2298
Type here to search
ENG 19:27 04/10/2020


Serial Wire Viewer settings for Test_SWV_1 Debug
Clock Settings
Core Clock: 80 MHz
Clock Prescaler: 40
SWO Clock: 2000.0 kHz
Trace Events
CPI: Cycles per instruction
EXC: Exception overhead
SLEEP: Sleep cycles
LSU: Load store unit cycles
FOLD: Folded instructions
EXETRC: Trace Exceptions
PC Sampling
Enable Resolution: 16384 Cycles/sample
Timestamps
Enable Prescaler: 1
Data Trace
Comparator 0
Enable
Var/Addr: state_PA8
Access: Read/Write
Size: Word
Generate: Data Value
Comparator 1
Var/Addr: state_PE8
Access: Read/Write
Size: Word
Generate: Data Value
Comparator 2
Var/Addr: state_PE8
Access: Read
Size: Word
Generate: Data Value
Comparator 3
Var/Addr: state_PE8
Access: Write
Size: Word
Generate: Data Value
ITM Stimulus Ports
Enable port: 31 24 23 16 15 8 7 0
Privileged only ports: Port 31..24 Port 23..16 Port 15..8 Port 7..0
OK
Cancel
/* Infinite loop */
Console Problems Executables Debugger Console Memory SWV Data Trace Timeline Graph
1
0.5
0
0 0.5 1 1.5 2 2.5 3 3.5 4 4.5 5 5.5 6 6.5 7 7.5 8 8.5 9 9.5 10 10.5 11 11.5 12 12.5 13
Type here to search
ENG 19:27 04/10/2020

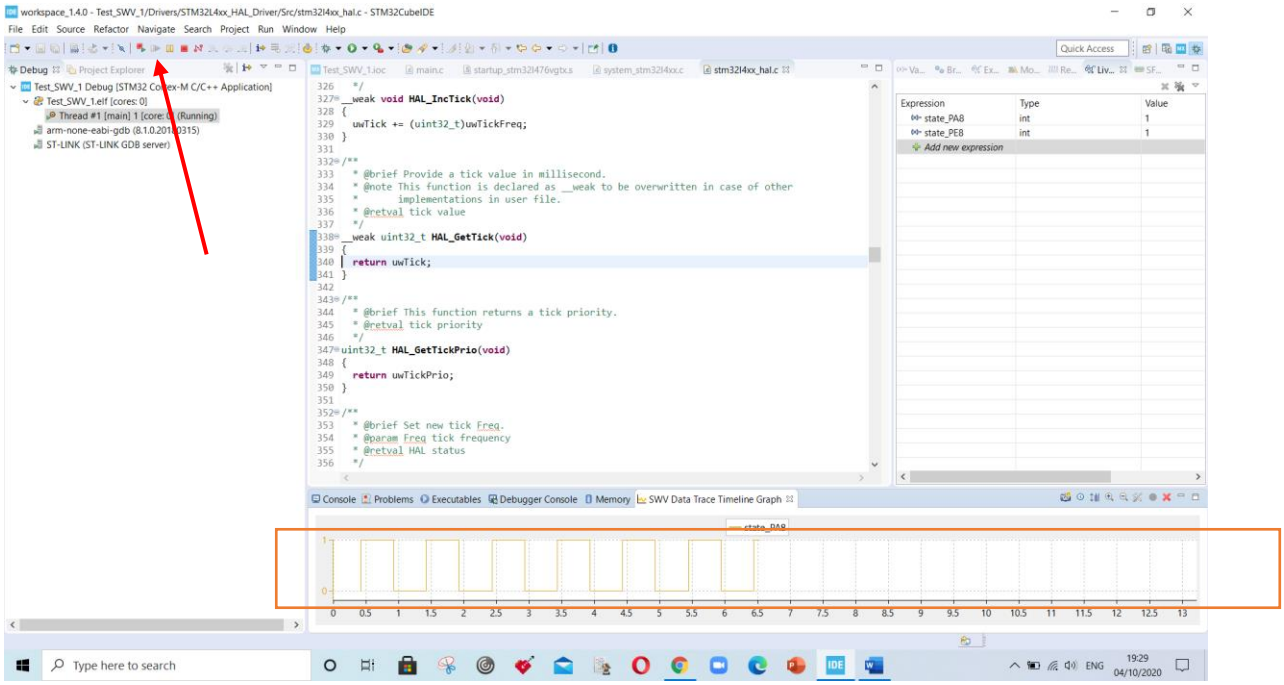
Step 6: Select Live expression window and Add the variable to monitor **'state_PB2'**.



Select start trace option the press the resume button to run the debug mode.



You can view the square wave as below:



To do: Create the light house flashing pattern in SWV Data Trace Timeline Graph.