Lab 5: Timer and PWM

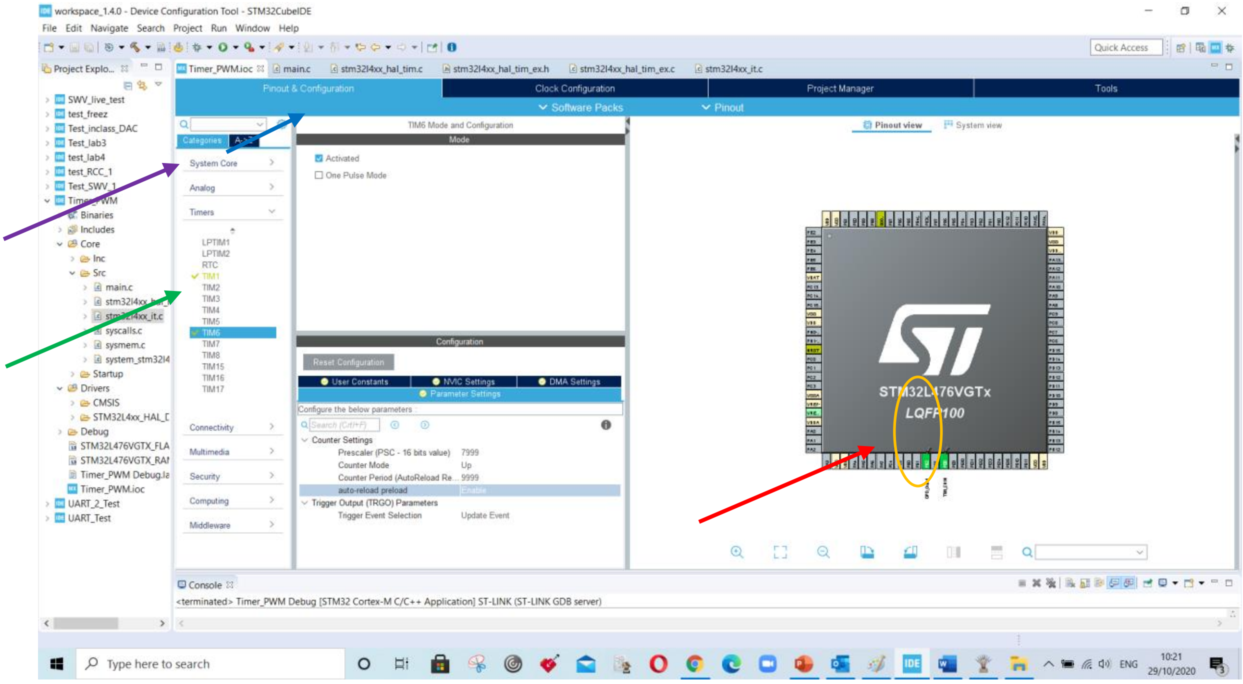
Activity 1: Timer Event Generation

Aim: Learn to how to configure Timer to generate train of software events.

Objectives:

1- Learn how to configure the Timer to generate software event at rate of 1Hz.
2- Using the software event interrupt callback to toggle LED.
3- Testing the functionality by PulseView software.

Step 1: Create a project in STM32CubeIDE. Expand 'Timers' menu and select 'TIM6' and in **'Mode and configuration'** check the 'Activated' box then declare PB2 as an output.



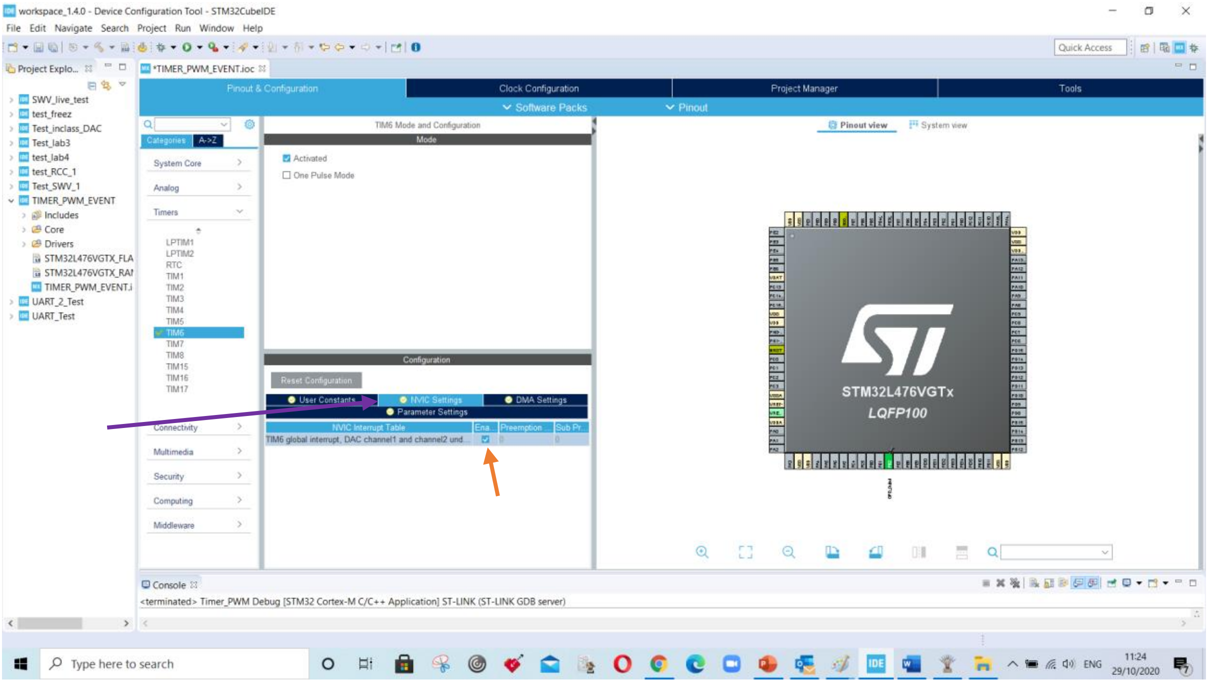
☑ Activated
☐ One Pulse Mode

Configure the below parameters :

Counter Settings
Prescaler (PSC - 16 bits value) 7999
Counter Mode Up
Counter Period (AutoReload Re... 9999
auto-reload preload Enable
Trigger Output (TRGO) Parameters
Trigger Event Selection Update Event

<terminated> Timer_PWM Debug [STM32 Cortex-M C/C++ Application] ST-LINK (ST-LINK GDB server)

Then go to **Configuration** window select 'NVIC settings' and check the enable box for 'TIM 6 global interrupt'.


workspace_1.4.0 - Device Configuration Tool - STM32CubeIDE

TIM6 Mode and Configuration

Mode

☑ Activated

☐ One Pulse Mode

Configuration

Reset Configuration

User Constants | NVIC Settings | DMA Settings | Parameter Settings

NVIC Interrupt Table	Ena...	Preemption ...	Sub Pr...
TIM6 global interrupt, DAC channel1 and channel2 und...	☑	0	0

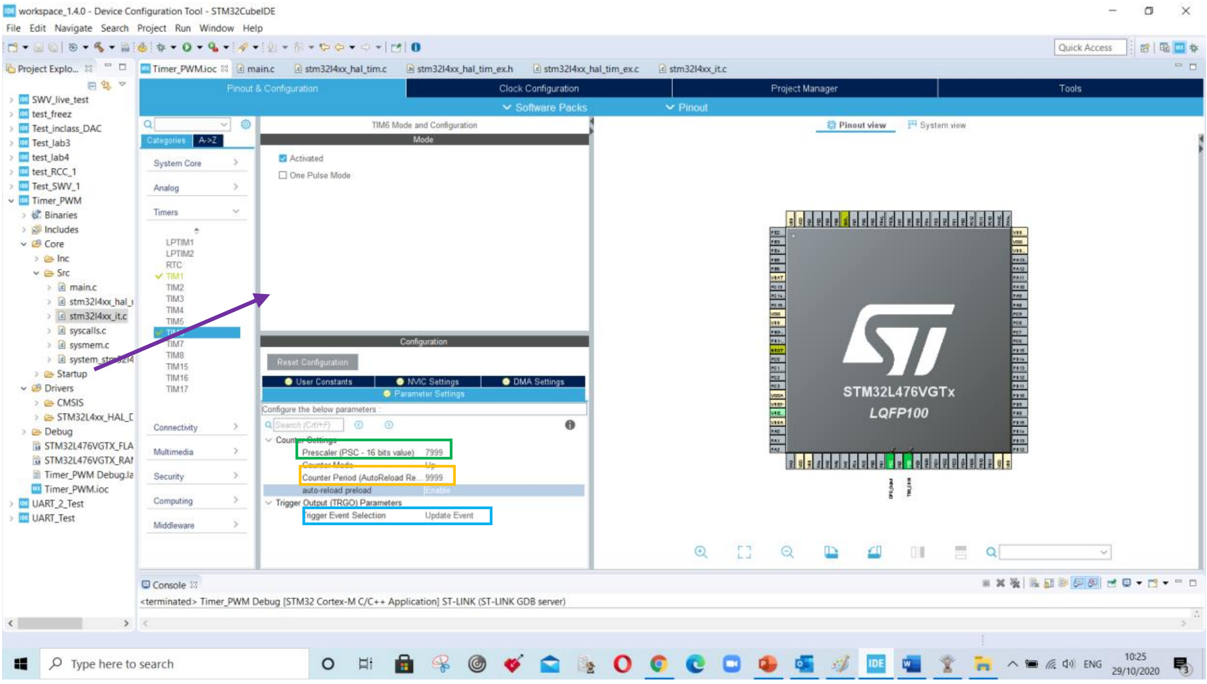
<terminated> Timer_PWM Debug [STM32 Cortex-M C/C++ Application] ST-LINK (ST-LINK GDB server)

Step 2: Set the **HCLK to 80MHz** and to create 1Hz event go to **Configuration** window select ‘parameter setting’ window assign ‘Prescaler value of 7999’ ‘Counter period value of 9999’ and select ‘update event’ in Trigger event selection. Finally **Enable** ‘auto reload Pre-load’ to create a sequential repetition.

Formula: { (**HCLK** / **Prescaler**) - **Control period** }

80MHz/8000=10000 KHz means 10000 cycles per second, so 9,999 counter period means clock speed 1 Hz

*The values are starting from zero so a subtraction of 1 is performed


workspace_1.4.0 - Device Configuration Tool - STM32CubeIDE

TIM6 Mode and Configuration

Mode

☑ Activated

☐ One Pulse Mode

Configuration

Reset Configuration

User Constants | NVIC Settings | DMA Settings | Parameter Settings

Configure the below parameters :

Counter Settings

Prescaler (PSC - 16 bits value) 7999

Counter Mode Up

Counter Period (AutoReload Re... 9999

auto-reload preload Enable

Trigger Output (TRGO) Parameters

Trigger Event Selection Update Event

<terminated> Timer_PWM Debug [STM32 Cortex-M C/C++ Application] ST-LINK (ST-LINK GDB server)

Step 3: Generate the code and add the code section given below:

Start the Timer:

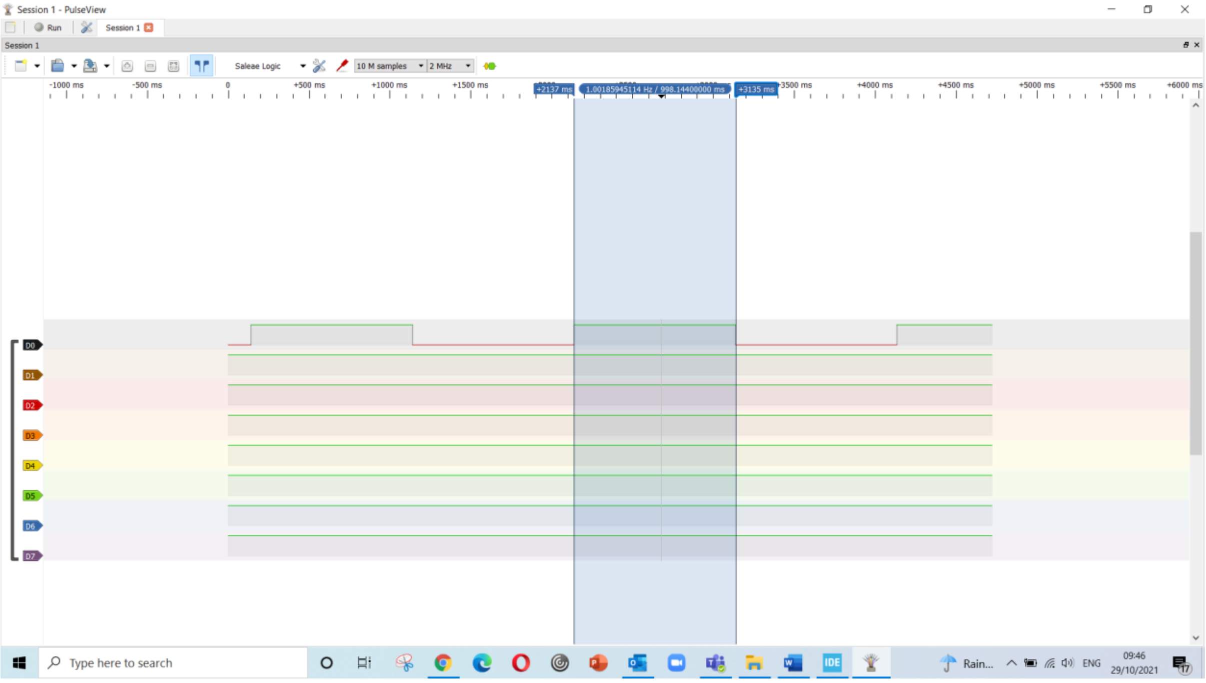
```
/* USER CODE BEGIN 2 */

HAL_TIM_Base_Start_IT(&htim6);
```

In source file: core->src-> stm32l4_it.c add the toggle statement to Timer interrupt handler function:

```
void TIM6_DAC_IRQHandler(void)
{
  /* USER CODE BEGIN TIM6_DAC_IRQn 0 */

  /* USER CODE END TIM6_DAC_IRQn 0 */
  HAL_TIM_IRQHandler(&htim6);
  /* USER CODE BEGIN TIM6_DAC_IRQn 1 */
  HAL_GPIO_TogglePin(GPIOB, GPIO_PIN_2);

  /* USER CODE END TIM6_DAC_IRQn 1 */
}
```

-

Step 4: Build and Run the code. You will be ablet to view the toggling of RED LED on STM32L4 DISCO Kit. Now capture waveform in PulseView Software we can verify the timing sequence with value of 1 Hz/998 ms approx (Note: LED toggle is taking place on each timer event)


Session 1 - PulseView

Saleae Logic

10 M samples

2 MHz

+2137 ms

1.00185945114 Hz / 998.14400000 ms

+3135 ms

To do: Generate software event with frequency of 2Hz.

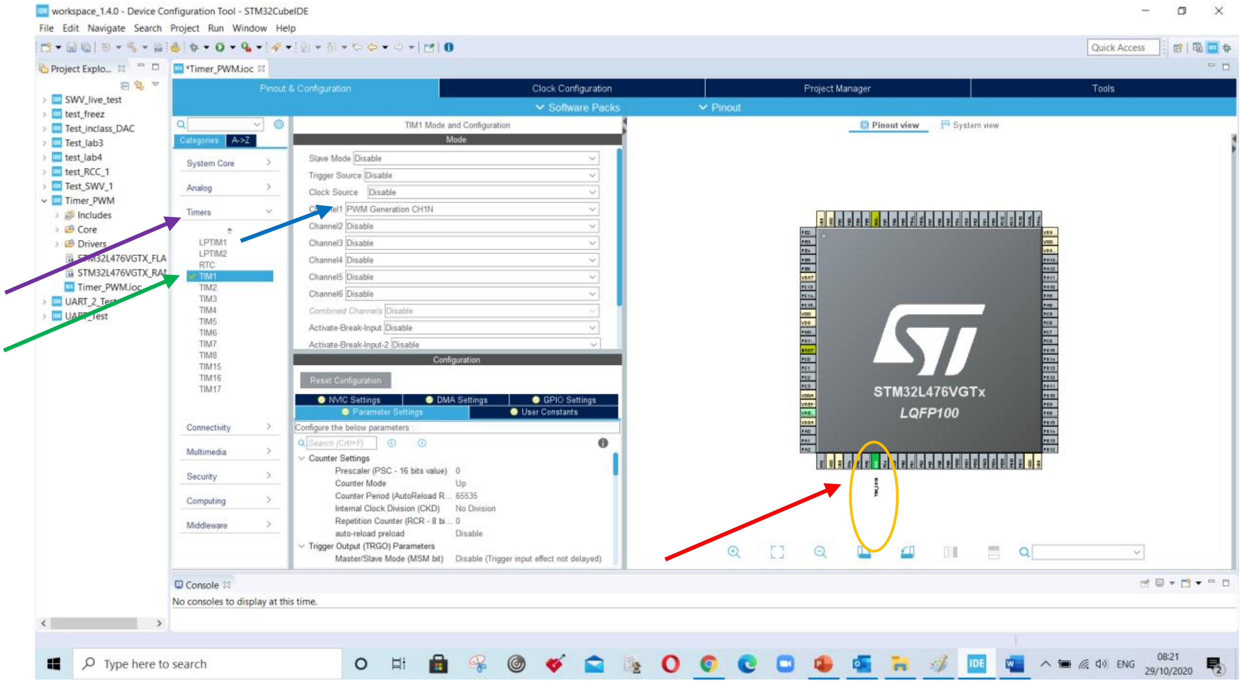
Activity 2: Timer for PWM Generation

Aim: Learn to how to configure Timer to generate the PWM waveform.

Objectives:

1- Learn how to configure the Timer as PWM.
2- Generate PWM waveform with duty cycle of 50% with frequency of 1Hz to blink the LED.
3- Testing the functionality by PulseView Software.

Step 1: Create a project in STM32CubeIDE. Expand 'Timers' menu and select 'TIM1' and in **Mode and configuration** select 'PWM Generation CH1N' for **Channel 1** then relocate the green highlighted pins by mouse while holding **CTRL** key on keyboard to Pin 'PE8'.



workspace_1.4.0 - Device Configuration Tool - STM32CubeIDE

TIM1 Mode and Configuration

Mode

Slave Mode: Disable
Trigger Source: Disable
Clock Source: Disable
Channel1: PWM Generation CH1N
Channel2: Disable
Channel3: Disable
Channel4: Disable
Channel5: Disable
Channel6: Disable
Combined Channels: Disable
Activate-Break-Input: Disable
Activate-Break-Input-2: Disable

Configuration

Configure the below parameters :

Counter Settings
Prescaler (PSC - 16 bits value): 0
Counter Mode: Up
Counter Period (AutoReload R...: 65535
Internal Clock Division (CKD): No Division
Repetition Counter (RCR - 8 bi...: 0
auto-reload preload: Disable

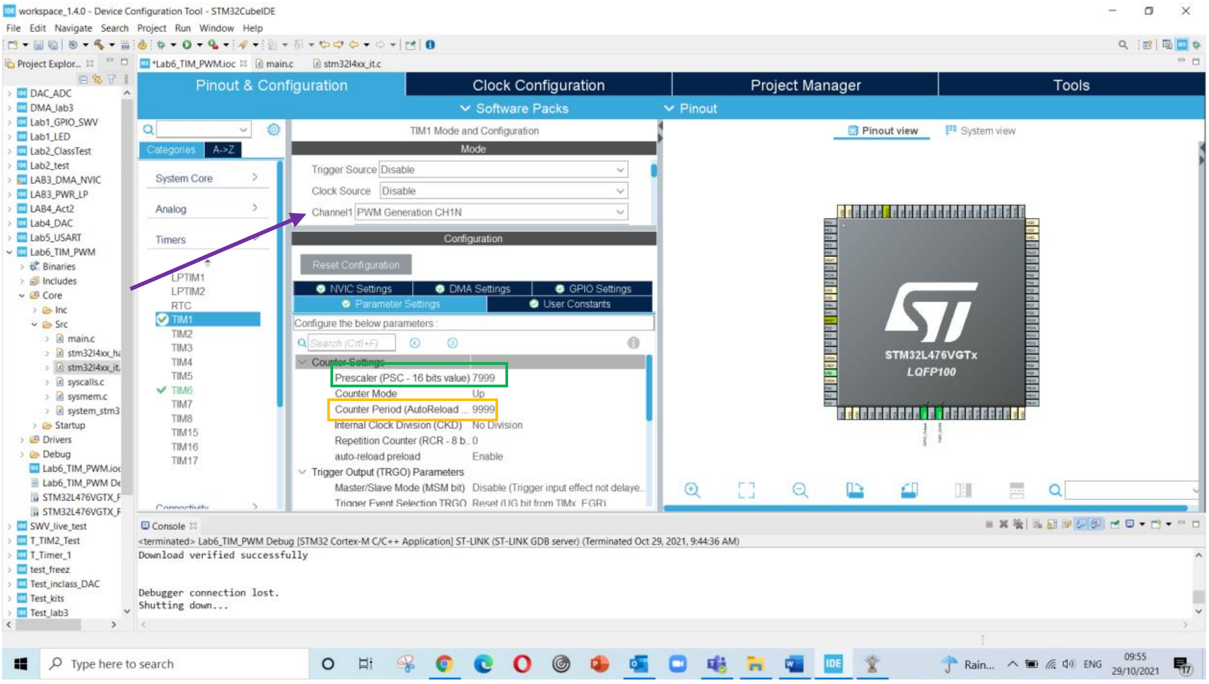
Trigger Output (TRGO) Parameters
Master/Slave Mode (MSM bit): Disable (Trigger input effect not delayed)

No consoles to display at this time.

Step 2: Set the **HCLK to 80MHz** and In configuration window for 'counter setting' select 'Parameter setting' window assign 'Prescaler value of 7999' 'Control period value of 9999' and, Enable 'auto reload pre-load'. In PWM Generation CH1N drop down give 'Pulse (16 bit value) = 5000**'** to generate the pulse of 1Hz

Formula: { (HCLK / prescaler*) - Control period* } x Pulse (duration)

*The values are starting from zero so a subtraction of 1 is performed.



TIM1 Mode and Configuration

Mode

Trigger Source: Disable

Clock Source: Disable

Channel1: PWM Generation CH1N

Configuration

Reset Configuration

NVIC Settings | DMA Settings | GPIO Settings | Parameter Settings | User Constants

Configure the below parameters :

Counter Settings
- Prescaler (PSC - 16 bits value) 7999
- Counter Mode Up
- Counter Period (AutoReload ... 9999
- Internal Clock Division (CKD) No Division
- Repetition Counter (RCR - 8 b.. 0
- auto-reload preload Enable

Trigger Output (TRGO) Parameters
- Master/Slave Mode (MSM bit) Disable (Trigger input effect not delaye...
- Trigger Event Selection TRGO Reset (UG bit from TIMx_EGR)

Console

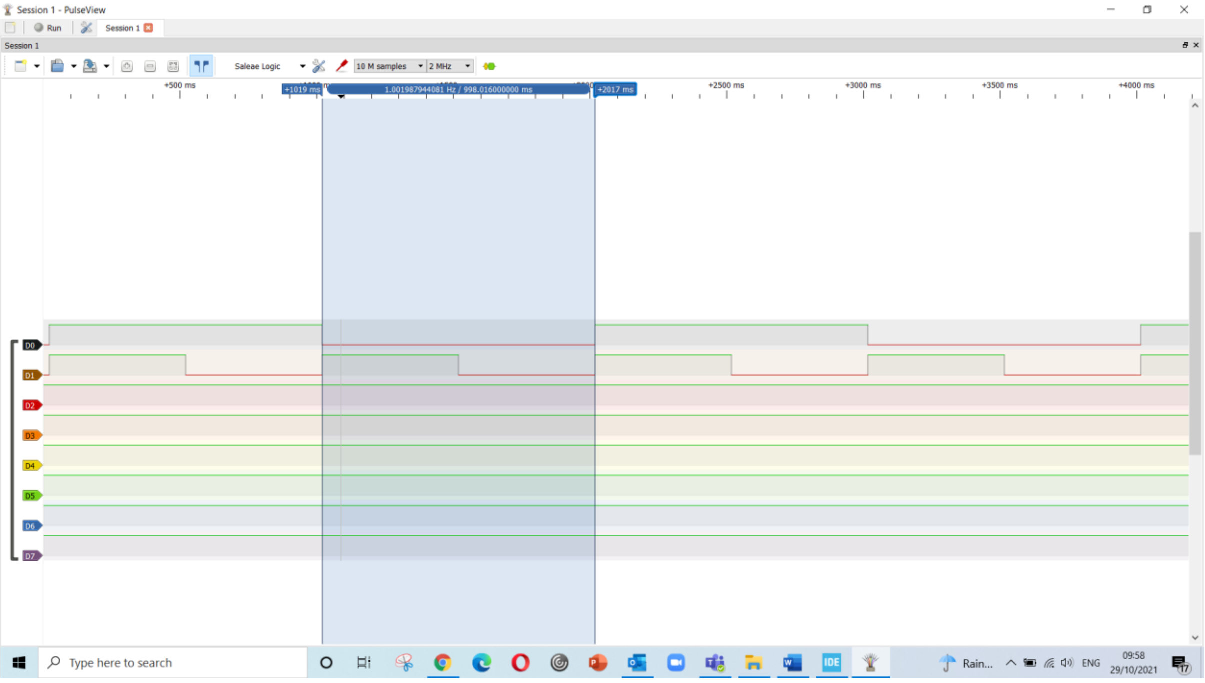
```
<terminated> Lab6_TIM_PWM Debug [STM32 Cortex-M C/C++ Application] ST-LINK (ST-LINK GDB server) (Terminated Oct 29, 2021, 9:44:36 AM)
Download verified successfully


Debugger connection lost.
Shutting down...
```

Step 3: Generate the code and add the code section given below:

Start the Timer to generate PWM waveform:

```
/* USER CODE BEGIN 2 */
HAL_TIMEx_PWMN_Start(&htim1, TIM_CHANNEL_1);
```

Step 4: Build and Run the code. You will be ablet to view the toggling of Green LED on STM32L4 DISCO Kit and on capturing waveform in **PulseView** software we can verify the timing of PWM signal. Total duration is 1Hz/998 ms with duty cycle of 50%.



Session 1 - PulseView

Saleae Logic

10 M samples

2 MHz

+1019 ms

1.001987944081 Hz / 998.016000000 ms

+2017 ms

To do: Create PWM waveform with duty cycle of 25% and 75%.