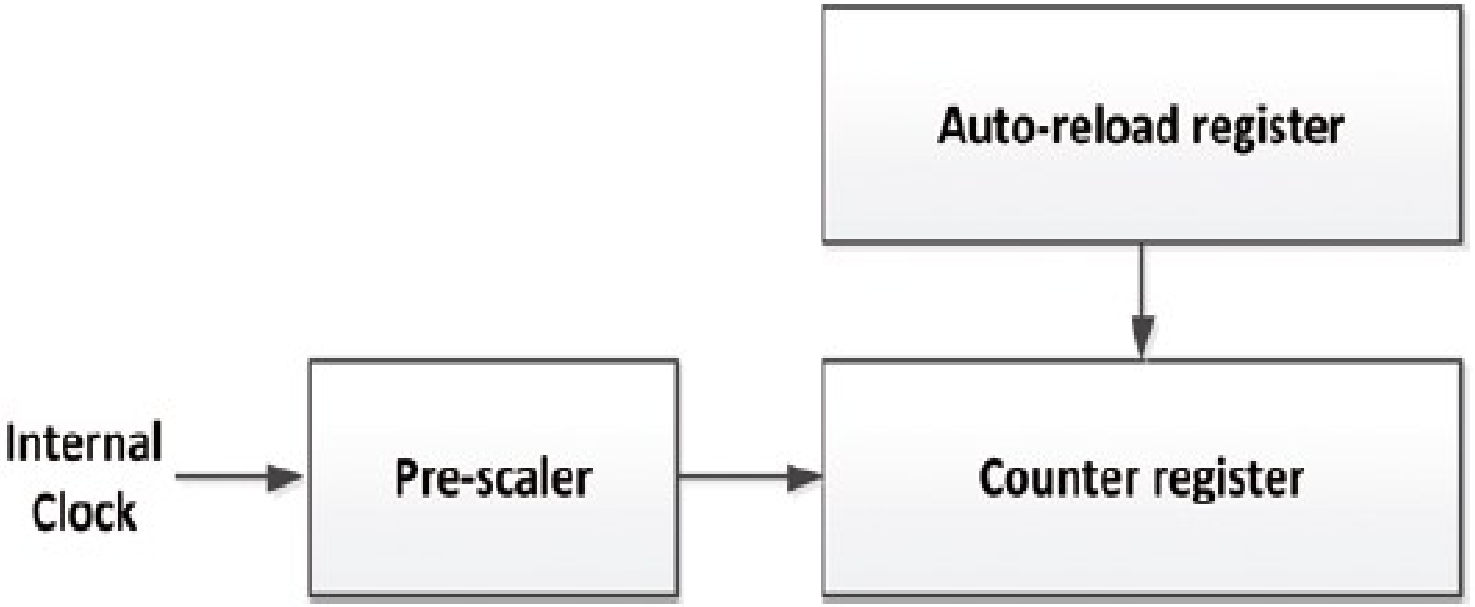
Tajmerski prekid



Tajmerski prekid

The following equation can be used to determine the values to be loaded to generate interrupts at specified times:

$$I = \frac{(P+1)(A+1)}{f}$$

or,

$$A = \frac{f \times I}{(P+1)} - 1$$

where

A is the value to be loaded into the auto-reload register;
I is the required interrupt time in seconds;
f is the clock period in Hz;
P is the prescaler value (1 to 65535).

Tajmerski prekid

The following equation can be used to determine the values to be loaded to generate interrupts at specified times:

$$I = \frac{(P+1)(A+1)}{f}$$

or,

$$A = \frac{f \times I}{(P+1)} - 1$$

where

A is the value to be loaded into the auto-reload register;
I is the required interrupt time in seconds;
f is the clock period in Hz;
P is the prescaler value (1 to 65535).

For example, with 80 MHz clock, for 1-second timer interrupts, assuming *P* = 39999, we have:

$$A = \frac{80,000,000 \times 1}{(39999+1)} - 1 = 1999$$

Podesiti za svoj MCU I max frekvenciju!!!

Timer Interrupt to Flash LED Every Second

Description
In this project the on-board LED (at port PA5) is flashed every second using timer interrupts.

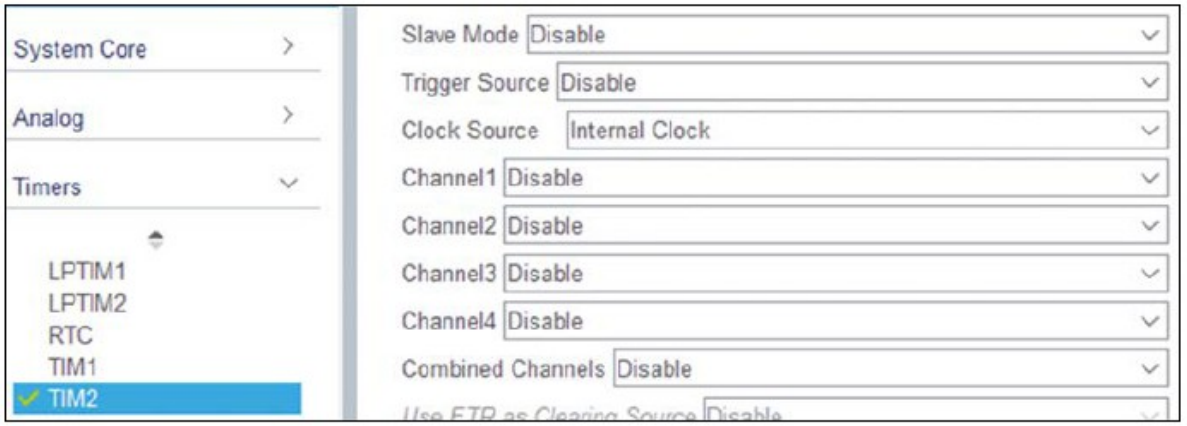
The aim
The aim of this project is to show how timer interrupts can be used in a program.

- Configure PA5 as digital output.
- Configure the MCU clock as 80 MHz and make sure that the APB1 clock is set to 80 MHz.
- Click **Timers**, then click TIM2, and select the **Clock Source** as **Internal clock**

Obratiti paznju na max clock za konkretan MCU!!!

Timer Interrupt to Flash LED Every Second

- Click **Timers**, then click TIM2, and select the **Clock Source** as **Internal clock** (Figure 6.3).



Timer Interrupt to Flash LED Every Second

- Under **Configuration**, click on **Prescaler (PSC – 16-bit value)** and make sure it is set to **Decimal** and change its value to 39999.
- Change the **Counter Mode** to **Down**.
- Click on **Counter Period (Auto reload register)** and make sure it is set to **Decimal**. Change its value to 1999.
- Click **Internal Clock Division** and set it to 2. Notice that this is not clock division, but it relates to the filter at the input sampler.
- Click **auto-reload preload** and click to **Enable** it.
- Click **System Core** followed by **NVIC**. Click to enable **TIM2 global interrupt**.
- Click **File**, followed by **Save** and click **YES** to generate code.

Timer Interrupt to Flash LED Every Second

You should now open the generated code. Notice that the following code is included for the timer interrupt (comments are removed for clarity):
The following timer related code is included in the program:

```
TIM_HandleTypeDef htim2;
```

You should add the following code just before the **while** loop in the main program to enable timer interrupts:

```
HAL_TIM_Base_Start_IT(&htim2);   // Enable the timer
```

Timer Interrupt to Flash LED Every Second

Also, the timer interrupt service routine should be added before the main program as shown below. Notice that the LED control code to toggle the LED is inside the timer interrupt service routine.

```
void HAL_TIM_PeriodElapsedCallback(TIM_HandleTypeDef *htim)
{
	HAL_GPIO_TogglePin(GPIOA, LED);
}
```

Timer Interrupt to Flash LED Every Second

```
int main(void)
{
  HAL_Init();

  SystemClock_Config();

  MX_GPIO_Init();
  MX_TIM2_Init();
  HAL_TIM_Base_Start_IT(&htim2);                // Enable the timer

  while (1)
  {
  }
}
```