مرکز آموزش نیـــرا سیستم

nirasystem.com

Timer (TIM)

Ali Mirghasemi

Introduction

- The Timer peripheral in STM32F407 is a fundamental feature that provides precise timing and timekeeping capabilities for various applications.
- Timers can generate accurate time delays, measure time intervals, and trigger actions at specific time intervals.
- The STM32F407 microcontroller offers multiple timers with distinct features and functionalities, making it highly versatile for time-sensitive tasks.

Modes

- **Basic Timer Mode**
  - Also known as "General-Purpose Timer" (GPT), it provides simple timing functions like generating time delays or periodic events.
- **Input Capture Mode**
  - This mode enables the timer to measure the time between external events, allowing precise time interval measurements.
- **Output Compare Mode**
  - In this mode, the timer can generate output signals or trigger actions based on comparing the timer value with a predefined value.
- **PWM Generation Mode**
  - The timer can generate Pulse Width Modulation (PWM) signals, useful for motor control, LED dimming, and other applications requiring variable duty cycles.

Types

- **General-Purpose Timers (TIM2, TIM3, TIM4, etc.)**
  - These are 16-bit and 32-bit timers suitable for basic timing and event generation tasks.
- **Advanced Timers (TIM1, TIM8)**
  - These are 16-bit and 32-bit timers with additional advanced features, such as complementary PWM outputs and input capture capabilities.
- **Basic Timer (TIM6, TIM7, etc.)**
- **Low-Power Timers (LPTIM1, LPTIM2)**
  - These timers are optimized for low-power applications, where precise timing is required with minimal power consumption.

Applications

- **Real-Time Control**
  - Timers are used in real-time control systems to schedule tasks, trigger events, and synchronize operations with precise timing.
- **Pulse Width Modulation (PWM)**
  - Timers generate PWM signals for motor control, LED dimming, and other applications where variable duty cycles are necessary.
- **Time Measurement**
  - The input capture mode enables accurate time measurement between external events, such as measuring the frequency of a signal or the speed of a motor.
- **Periodic Tasks**
  - Timers can trigger periodic tasks, such as updating display content or acquiring sensor data at fixed intervals.
- **Timekeeping**
  - Timers are used for timekeeping and scheduling tasks in real-time operating systems (RTOS).

Examples

- **Generating Delays**
  - Using a basic timer to create precise time delays for controlling external devices.
- **PWM Signal Generation**
  - Utilizing timers to generate PWM signals for controlling the speed of a motor.
- **Input Capture**
  - Measuring the frequency of an external signal by capturing the time between rising edges.
- **RTOS Scheduling**
  - Employing timers in an RTOS to schedule tasks at specific intervals.

Registers

- **TIMx_CR1**
  - Control register 1, which controls the timer's basic settings, such as clock division, direction, and one-pulse mode.
- **TIMx_ARR**
  - Auto-reload register, sets the value at which the timer reloads.
- **TIMx_CNT**
  - Counter register, holds the current value of the timer.
- **TIMx_CCRx**
  - Capture/Compare registers, used for input capture, output compare, and PWM signal generation.

Formula

- $Tim_{Tick} = \frac{Tim_{Freq(Hz)} \times Tim_{Period(s)}}{Tim_{Prescaler} \times Tim_{Divider}} - 1$

- $Tim_{Tick} =$ **TIMx_ARR**