

STM32L5 - LPTIM

Low-power timer

Revision 1.0



Hello, and welcome to this presentation of the STM32L5 Low-power timer (LPTIM). It covers the features of this peripheral, which offers a set of timing features and can generate waveforms even in low-power modes.

Overview

2

- The LPTIM is a 16-bit timer. Thanks to its diversity of clock sources, the LPTIM is able to keep running in most of the available low-power modes of the STM32L5 microcontroller.

Features summary

- Asynchronous running capability
- Ultra-low power consumption
- Timeout function for wakeup from low-power modes



The low-power timer peripheral embedded in the STM32L5 microcontroller provides a 16-bit timer that is able to run even in low-power modes. This is made possible thanks to a flexible clocking scheme. The low-power timer peripheral provides basic general-purpose timer functions. One major function of the low-power timer is its capability to keep running even when no internal clock source is active when configured in asynchronous counting mode.

Key features

3

- Flexible clocking scheme through many selectable clock sources:
 - Internal clock sources: LSE, LSI, HSI16 or APB clocks
 - External clock source over LPTIM “LPTIM_IN1” input (works even when on-chip oscillator is not running, used by Pulse Counter applications)
- Up to 8 external triggers
 - With configurable active edges: rising edge, falling edge and both edges
 - With digital glitch filter to avoid spurious triggers
- 2 operating modes: continuous and one-shot



The low-power timer's main feature is its ability to keep running even in low-power mode when almost all clock sources are turned off. The low-power timer has a very flexible clocking scheme. It can be clocked from on-chip clock sources: LSE, LSI, HSI16 or APB clocks. Or it can be clocked from an external clock source over the low-power timer's “LPTIM_IN1” input. This latter feature is used for building “Pulse Counter” applications and is a key function for metering applications like gas-meters, etc.

The low-power timer features up to 8 external trigger sources with configurable polarity. External trigger inputs feature digital filters to cancel-out faulty triggers that could be raised in noisy operating environments.

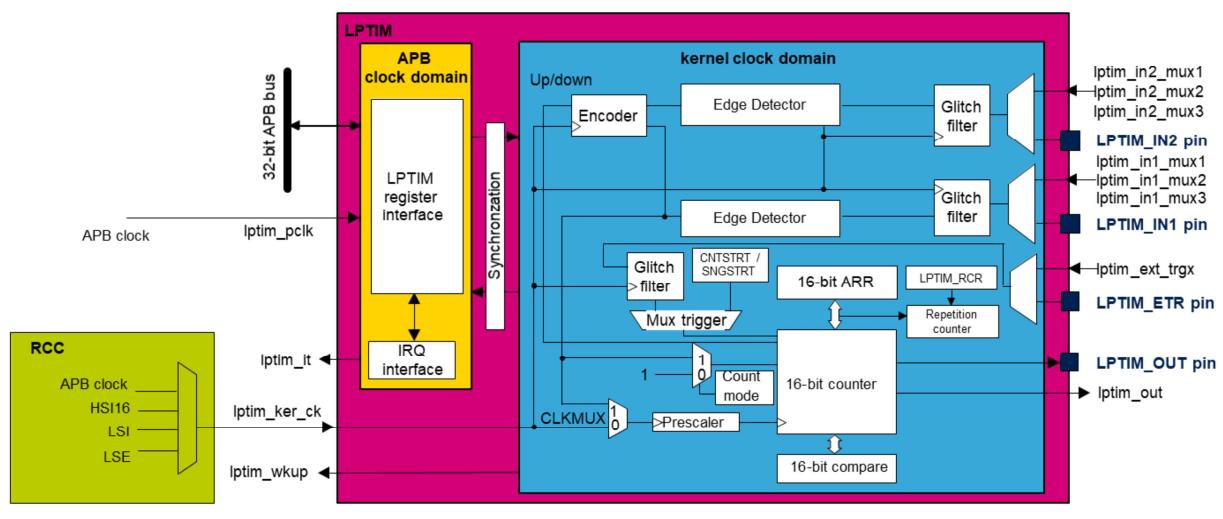
The low-power timer can be configured to run either in Continuous or One-shot mode. One-shot mode is used for generating pulse waveforms while Continuous mode is

used to generate PWM waveforms.

Block diagram

4

LPTIM block diagram



The low-power timer is a peripheral with two clock domains. The APB clock domain contains the peripheral's APB interface. The kernel clock domain contains the low-power timer peripheral's core functions. The kernel clock domain can be clocked by internal clock sources or by an external clock source through the timer's "LPTIM_IN1" input.

The low-power timer peripheral embeds a 16-bit counter that is fed through a power-of-two prescaler. The low-power timer peripheral features a 16-bit Auto-reload register and a 16-bit Compare register that are used to set the period and duty-cycle, respectively, for a PWM waveform signal output on the timer's "LPTIM_OUT" output.

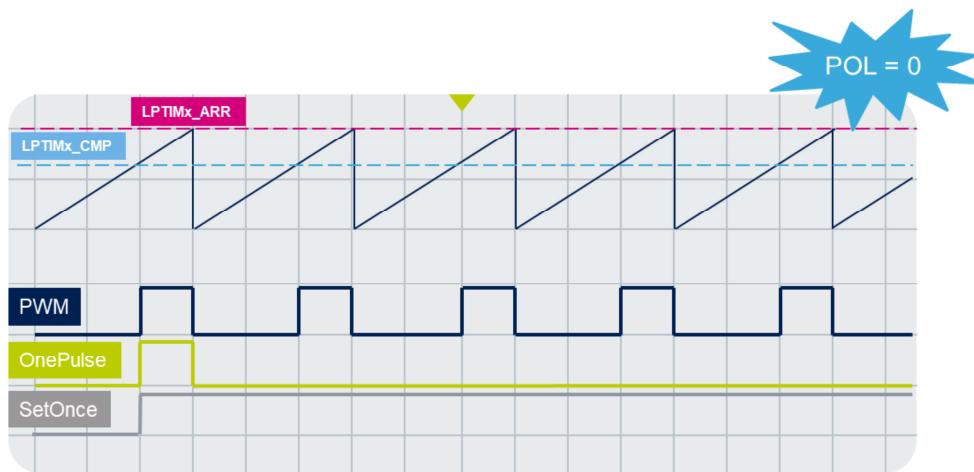
The low-power timer features a repetition counter which allows to adjust the counter roll-over.

The low-power timer features an Encoder mode function that can be used to interface with incremental quadrature encoder sensors using the peripheral's "lptim_in1_mux" and "lptim_in2_mux" inputs. Both inputs feature glitch-filtering circuitry.

Up to 3 configurable waveforms

5

- PWM, one-pulse and set-once waveforms



The LPTIM_CMP and LPTIM_ARR registers in conjunction with the bit-fields 'WAVE' from the LPTIM_CFGR register and 'SNGSTRT' from the LPTIM_CR register are used to control the output waveform.

The output waveform is either a typical PWM signal with its period and duty-cycle controlled by the LPTIM_ARR and LPTIM_CMP registers, respectively. Or it is a single pulse with the last output state defined by the configured waveform.

If the last output state is the same as the one at the waveform's beginning, then One-pulse mode is configured. If not, then SetOnce mode is configured.

The low-power timer's output polarity is controlled through the 'WAVPOL' bit-field in the LPTIM_CFGR register.

Timer counter reset

- Timer counter reset resets the content of the LPTIM_CNT register
- Two counter reset mechanisms are possible
 - Synchronous counter reset mechanism
 - When the COUNTRST bit of the LPTIM_CR register is set to '1', the content of the LPTIM_CNT register is reset. This reset only takes place after a synchronization delay of 3 kernel clock cycles (the lptim_ker_ck kernel clock signal may be different from APB clock).
 - Asynchronous counter reset mechanism
 - When the RSTARE bit of the LPTIM_CR register is set to '1', any read access to LPTIM_CNT register will asynchronously reset the contents of the LPTIM_CNT register.



The low-power timer features a counter reset function used to reset to '0' the content of the LPTIM_CNT register.

Two counter reset mechanisms are possible: The synchronous counter reset mechanism and the asynchronous counter reset mechanism.

A synchronous counter reset is performed by setting the COUNTRST bit. Due to the synchronous nature of this reset, it only takes place after a synchronization delay of 3 LPTIM kernel clock cycles.

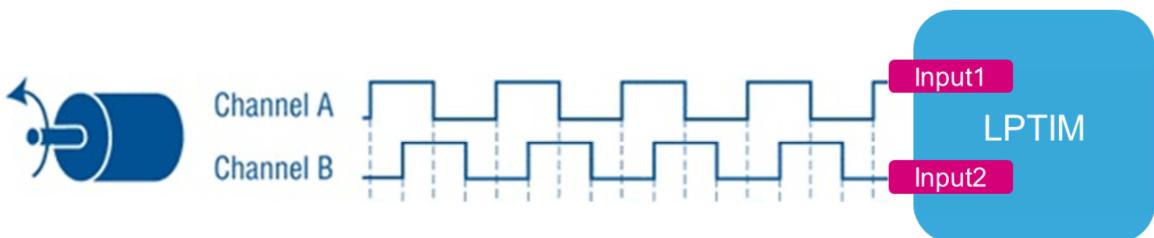
When the RSTARE bit is set, an asynchronous counter reset is performed on the next APB read access to the LPTIM_CNT register.

Encoder mode

7

Encoder Mode

- Same operating mode as Encoder mode on general-purpose timers
- Only available when LPTIM is running in Continuous mode



The low-power timer features an Encoder mode function that can interface with the incremental quadrature encoder sensors using the peripheral's "Input1" and "Input2" inputs. Both inputs feature glitch-filtering circuitry.

The Encoder function is similar to the one embedded in the general-purpose timers.

In order to use the Encoder mode function, the low-power timer must be running in Continuous mode.

One important thing to note is that only low-power timers 1 and 2 embed the Encoder mode function.

Interrupt event	Description
Compare match	Interrupt flag is raised when the Counter register's (LPTIM_CNT) content matches the Compare register's (LPTIM_CMP) content.
Auto-reload match	Interrupt flag is raised when the Counter register's (LPTIM_CNT) content matches the Auto-reload register's (LPTIM_ARR) content.
External trigger event	Interrupt flag is raised when an external trigger is detected.
Auto-reload register write completed	Interrupt flag is raised when the write action into the LPTIM_ARR register is completed.
Compare register write completed	Interrupt flag is raised when the write action into the LPTIM_CMP register is completed.
Direction change	Used for Encoder mode, two interrupt flags are embedded to highlight direction change: Up flag to highlight up-counting direction change and Down flag to highlight down-counting direction change.
Update Event	Interrupt flag is raised when the repetition counter underflows (or contains zero) and the LPTIM counter overflows.
Repetition register write completed	REPOK is set by hardware to inform application that the APB bus write operation to the LPTIM_RCR register has been successfully completed.



The low-power timer peripheral features 9 interrupt sources.

- The “Compare match” interrupt is raised once the content of Counter register LPTIM_CNT matches or is greater than the Compare register LPTIM_CMP content.
- The “Auto-reload match” interrupt is raised when the Counter register's content matches the Auto-reload register's content.
- The “External trigger event” interrupt is raised when a valid external trigger is detected.
- The “Auto-reload register write completed”, the “Compare register write completed” and the “Repetition register write completed” interrupts are raised when the transfer of the content of the LPTIM_ARR register, the LPTIM_CMP register and the LPTIM_RCR register,

respectively, is completed from the peripheral's APB interface logic into the peripheral's core logic which are contained by two different clock domains. These three interrupts are useful in mitigating the overhead of polling on the status of writing to these two registers when the peripheral core clock is too slow compared to the APB interface clock.

- The “Up and Down Direction change” interrupts are raised when the Encoder mode function is enabled and the counting direction is changed from up to down or vice-versa. The counting direction of the low-power timer's counter reflects the rotation direction of the quadrature sensor.
- The “Update Event” is raised when the repetition counter underflows and the LPTIM counter overflows.

Low-power modes

9

Mode	Description
Sleep	Active. Peripheral interrupts cause the device to exit Sleep mode.
Low-power sleep	Active. Peripheral interrupts cause the device to exit Low-power sleep mode
Stop 0/Stop 1	Active when LPTIM is clocked by LSE or LSI. LPTIM interrupts cause the device to exit Stop 0 and Stop 1.
Standby	Powered-down. The peripheral must be reinitialized after exiting Standby mode.
Shutdown	Powered-down. The peripheral must be reinitialized after exiting Shutdown mode.



The low-power timer peripheral is active in Sleep and Stop power modes.

The low-power timer is able to wake up the microcontroller from either Sleep or Stop modes.

Features of STM32L5 instances

10

LPTIM features (1)	LPTIM1	LPTIM2	LPTIM3
Encoder mode	X	-	-
External input clock	X	X	X
Wakeup from Stop 0 + Stop 1	(2)	(3)	(2)

1. X = supported
2. Wakeup supported from Stop 0, Stop 1 and Stop 2 modes.
3. Wakeup supported from Stop 0 and Stop 1 modes.



The STM32L5 devices embed three LPTIM peripherals, where only LPTIM1 instance embeds the encoder mode. Wakeup from Stop 0 and Stop 1 modes is supported by all of LPTIM instances while wakeup from Stop 2 is only supported by LPTIM1 and LPTIM 3.

References

11

- For more details and additional information, refer to the following:
 - Application note AN4865: Low-power timer (LPTIM) applicative use-cases on STM32 MCUs



For more details, please refer to the following documentation available on our website.