Hello, and welcome to this presentation of the STM32H5 Low-Power Timers. It covers the main features of these ultra-low-power timers.

Overview

- The LPTIM is a 16-bit timer
  - Thanks to its diversity of clock sources, the LPTIM runs in most of the available low-power modes of the STM32H5 microcontroller

Features summary

- Asynchronous running capability
- Ultra-low power consumption
- Fully functional in stop mode
- PWM & Input Capture mode
- Timeout function for wakeup from low-power modes

life.augmented

2

The low-power timer peripheral embedded in the STM32H5 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

- Flexible clocking scheme through many selectable clock sources:
  - Internal configurable clock sources (APB clock, PER_CK, PLL output p, LSI, LSE)
  - External clock source over LPTIM “LPTIM_IN1” input (works even when on-chip oscillator is not running, useful for 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
- Two operating modes: continuous and one-shot
- Configurable channels
  - Each channel is configurable independently as an input capture or PWM output
  - Input Capture with configurable active edges and digital glitch filters
- DMA requests for input capture and update events

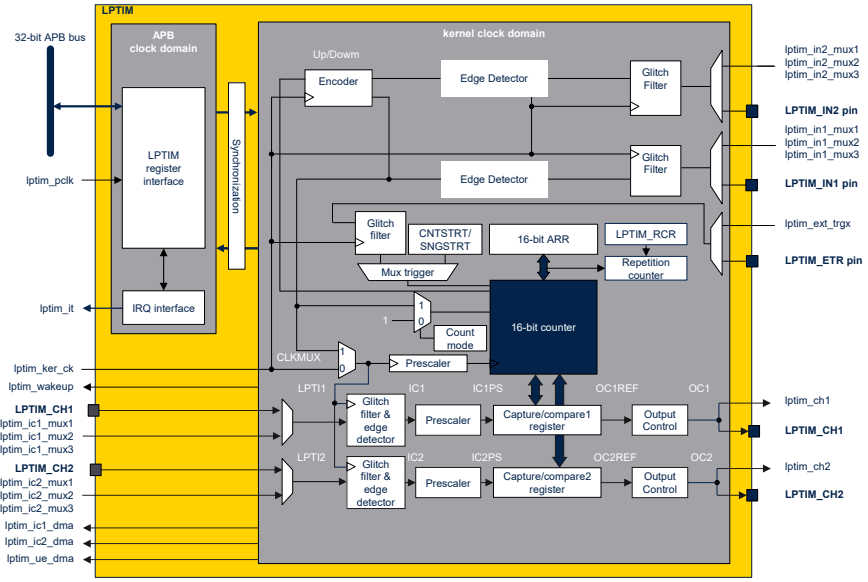
life.augmented

3

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: APB clock, Peripheral clock (PER_CK), LSE, LSI, PLL output p. 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 glitch 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.
LPTIMs with implemented channels, can be configured as either input capture or output compare.
Two different events can trigger DMA requests: input capture and update event.

Block diagram

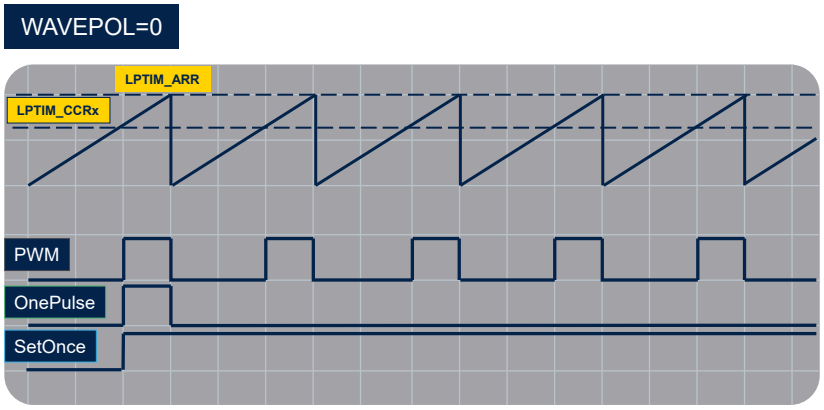


The low-power timer is a peripheral split into two separate clock domains.
The APB clock domain on the left of the figure contains the peripheral's APB interface, while the kernel clock domain contains the low-power timer peripheral core functions.
The kernel clock domain on the right of the figure can 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,

steered to either LPTIM_CH1 or LPTIM_CH2 output, according to the PWM channel number.
The low-power timer features a repetition counter which allows to adjust the counter roll-over.
At last, 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 to select the direction. These input signals are visible on the upper right of this figure.
All timer inputs support a glitch-filtering circuitry.
Note that LPTIM4 has a simpler block diagram.

Up to 3 configurable waveforms

PWM, one-pulse and set-once waveforms



life.augmented

5

The LPTIM_CCRx and LPTIM_ARR registers in conjunction with the bit-fields WAVE from the LPTIM_CFGR register and Single Start 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_CCRx 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. Otherwise SetOnce mode is configured.

The low-power timer's output polarity is controlled through

the WAVPOL bit-field in the LPTIM_CFGR register.
By setting the polarity bit, the default state of the output is high level, and the waveforms are one’s complemented with respect to the ones represented in the figure.

PWM mode

PWM mode

- Each channel can be configured to the PWM output with the following features:
  - Configurable duty cycle as defined by the LPTIM_CCRx register
  - Configurable period defined by the LPTIM_ARR register (common for all channels)
  - Programmable output polarity
- An update event (interrupt or DMA) is issued on counter overflow and repetition counter underflow
  - The PWM period or duty cycle can be changed regularly by the CPU or DMA at each update event
- The update interrupt issuing rate is adjustable with a repetition counter

life.augmented

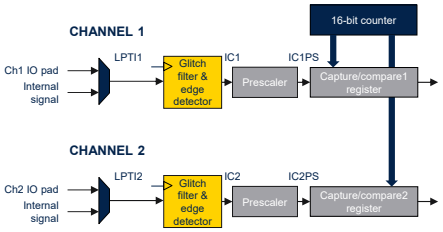
6

The PWM mode generates a signal with a frequency determined by the value of the LPTIM_ARR register and a duty cycle determined by the value of the LPTIM_CCRx register.
The LPTIM is able to generate PWM in edge-aligned mode.
Dynamic modifications of the PWM period or duty cycle can be performed by CPU or DMA accesses.
The following PWM events can be used to raise an interrupt request: counter overflow, repetition counter underflow.
The 8-bit repetition counter decrements at each counter overflow and triggers an interrupt when it reaches zero, which is convenient to adjust the interrupt rate.

Input capture

Input capture mode

- Each channel can be configured individually as input capture with the following features:
  - Programmable edge-sensitivity (rising / falling / both)
  - Event prescaler (1 capture every 1, 2, 4, 8 events)
  - Digital filter (for debouncing and noise removal)
- A capture event causes the counter value to be transferred into the capture register and triggers an interrupt or a DMA request
  - The captured value can be transferred by CPU or DMA
  - The over-capture flag is set if the capture register is overwritten without having been read



life.augmented

7

This slide describes the input capture features.
Each channel can be individually configured as input capture with various signal conditioning options.
The edge sensitivity is programmable and can be rising edge, falling edge or both.
An event prescaler captures of one event every 2, 4 or 8 events, as programmed in the prescaler.
Spurious transitions due to noise or bounces can be removed using a programmable digital filter.
Once the capture trigger is issued, the timer's counter value is transferred into the capture register and an interrupt or a DMA request can be issued.
If a new capture occurs before the previous one has been read, the capture register is over-written and an over-

capture flag is set for the software to manage this condition if needed.

Timer counter reset

Timer counter reset

- Timer counter reset clears the content of the LPTIM_CNT register
- Two counter reset mechanisms are implemented:
  - 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 the APB clock)
  - Asynchronous counter reset mechanism
    - When the RSTARE bit of the LPTIM_CR register is set to '1', any read access to the LPTIM_CNT register will asynchronously reset the contents of the LPTIM_CNT register

life.augmented

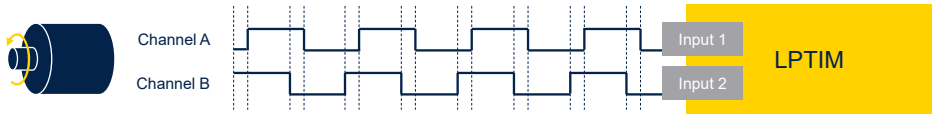
8

The low-power timer features a counter reset function used to reset to ‘0’ the contents 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

Encoder Mode

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



life.augmented

9

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.
This mode allows handles signals from quadrature encoders used to detect the angular position of rotary elements.
Encoder interface mode simply acts as an external clock with direction selection. This means that the counter just counts continuously between 0 and the auto-reload value programmed in the LPTIM_ARR register.
From the two external input signals, Input1 and Input2, a clock signal is generated to clock the LPTIM counter. The phase between these two signals determines the counting direction.

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.

Interrupts and DMA

Event	Interrupt	DMA	Description
Capture	Yes	Yes	Generated when a compare match or a capture is triggered
Compare	Yes	No	
Auto-reload match	Yes	No	Raised when an Auto-reload match is triggered
External trigger	Yes	No	Raised when an external trigger is detected
ARR, CCRx, RCR register write OK	Yes	No	ARROK, CMPxOK or REPOK flag is raised when the write action to the LPTIM_ARR, LPTIM_CCRx or LPTIM_RCR register, respectively, is completed
Direction change	Yes	No	Used for Encoder mode: Up flag to highlight up-counting direction change and Down flag to highlight down-counting direction change
Update Event	Yes	Yes	Raised when the repetition counter underflows (or contains zero) and the LPTIM counter overflows

life.augmented

This table lists the interrupts and DMA request sources.

- The “Capture” interrupt or DMA request is generated once the contents of the Counter register LPTIM_CNT matches or is greater than the Compare register LPTIM_CCRx contents.
- The “Compare match” interrupt or DMA request is generated once the contents of Counter register LPTIM_CNT matches or is greater than the Compare register LPTIM_CCRx contents.
- The “Auto-reload match” interrupt is raised when the Counter register’s contents matches the Auto-reload register’s contents.
- The “External trigger event” interrupt is raised when a valid external trigger is detected.
- The “Auto-reload register write OK”, the “Compare register write OK” and the “Repetition register write OK”

interrupts are raised when the transfer of the contents of the LPTIM_ARR register, the LPTIM_CCRx register or the LPTIM_RCR register, respectively, is completed from the peripheral's APB interface logic into the peripheral's core logic which are contained in two different clock domains. These three interrupts are useful in mitigating the overhead of polling on the status or writing to these status registers when the peripheral core clock is much slower than 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" interrupt or DMA request is generated when the repetition counter underflows and the LPTIM counter overflows.

Low-power modes

Mode	Description
Sleep	Active ➢ Peripheral interrupts cause the device to exit Sleep mode
Stop	If the LPTIM is clocked by an oscillator available in Stop mode, LPTIM is functional, and the interrupts cause the device to exit Stop mode
Standby	Powered-down ➢ The peripheral must be reinitialized after exiting Standby mode

life.augmented

The low-power timer peripheral is active in Sleep and Stop power modes.
It is powered-down and therefore not functional in Standby mode.
The low-power timer wakes up the microcontroller from either Sleep or Stop modes.
All DMA requests must be disabled, before entering Sleep, Stop and Standby modes.

References

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

life.augmented

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

Thank you

© STMicroelectronics - All rights reserved.
ST logo is a trademark or a registered trademark of STMicroelectronics International NV or its affiliates in the EU and/or other countries.
For additional information about ST trademarks, please refer to www.st.com/trademarks.
All other product or service names are the property of their respective owners.

life.augmented

In addition to this presentation, you can refer to the following presentations:

-Reset and Clock Controller (RCC)

-Power Control (PWR).