21 High-Resolution Timer (HRTIM)

21.1 Introduction

The high-resolution timer can generate up to 10 digital signals with highly accurate timings. It is primarily intended to drive power conversion systems such as switch mode power supplies or lighting systems, but can be of general purpose usage, whenever a very fine timing resolution is expected.

Its modular architecture allows to generate either independent or coupled waveforms. The wave-shape is defined by self-contained timings (using counters and compare units) and a broad range of external events, such as analog or digital feedbacks and synchronization signals. This allows to produce a large variety of control signal (PWM, phase-shifted, constant Ton,...) and address most of conversion topologies.

For control and monitoring purposes, the timer has also timing measure capabilities and links to built-in ADC and DAC converters. Last, it features light-load management mode and is able to handle various fault schemes for safe shut-down purposes.

21.2 Main features

- High-resolution timing units
  - 217 ps resolution, compensated against voltage and temperature variations
  - High-resolution available on all outputs, possibility to adjust duty-cycle, frequency and pulse width in triggered one-pulse mode
  - 6 16-bit timing units (each one with an independent counter and 4 compare units)
  - 10 outputs that can be controlled by any timing unit, up to 32 set/reset sources per channel
  - Modular architecture to address either multiple independent converters with 1 or 2 switches or few large multi-switch topologies
- Up to 10 external events, available for any timing unit
  - Programmable polarity and edge sensitivity
  - 5 events with a fast asynchronous mode
  - 5 events with a programmable digital filter
  - Spurious events filtering with blanking and windowing modes
- Multiple links to built-in analog peripherals
  - 4 triggers to ADC converters
  - 3 triggers to DAC converters
  - 3 comparators for analog signal conditioning
- Versatile protection scheme
  - 5 fault inputs can be combined and associated to any timing unit
  - Programmable polarity, edge sensitivity, and programmable digital filter
  - dedicated delayed protections for resonant converters
- Multiple HRTIM instances can be synchronized with external synchronization inputs/outputs
- Versatile output stage
  - High-resolution Deadtime insertion (down to 868 ps)
  - Programmable output polarity
  - Chopper mode
- Burst mode controller to handle light-load operation synchronously on multiple converters
- 7 interrupt vectors, each one with up to 14 sources
- 6 DMA requests with up to 14 sources, with a burst mode for multiple registers update

21.3 Functional description

21.3.1 General description

The HRTIM can be partitioned into several sub entities:

- The master timer
- The timing units (Timer A to Timer E)
- The output stage
- The burst mode controller
- An external event and fault signal conditioning logic that is shared by all timers
- The system interface

The master timer is based on a 16-bit up counter. It can set/reset any of the 10 outputs via 4 compare units and it provides synchronization signals to the 5 timer units. Its main purpose is to have the timer units controlled by a unique source. An interleaved buck converter is a typical application example where the master timer manages the phase-shifts between the multiple units.

The timer units are working either independently or coupled with the other timers including the master timer. Each timer contains the controls for two outputs. The outputs set/reset events are triggered either by the timing units compare registers or by events coming from the master timer, from the other timers or from external events.

The output stage has several duties

- Addition of deadtime when the 2 outputs are configured in complementary PWM mode
- Addition of a carrier frequency on top of the modulating signal
- Management of fault events, by asynchronously asserting the outputs to a predefined safe level

The burst mode controller can take over the control of one or multiple timers in case of light-load operation. The burst length and period can be programmed, as well as the idle state of the outputs.

The external event and fault signal conditioning logic includes:

- The input selection MUXes (for instance for selecting a digital input or an on-chip source for a given external event channel)
- Polarity and edge-sensitivity programming
- Digital filtering (for 5 channels out of 10)

The system interface allows the HRTIM to interact with the rest of the MCU:

- Interrupt requests to the CPU
- DMA controller for automatic accesses to/from the memories, including an HRTIM specific burst mode
- Triggers for the ADC and DAC converters

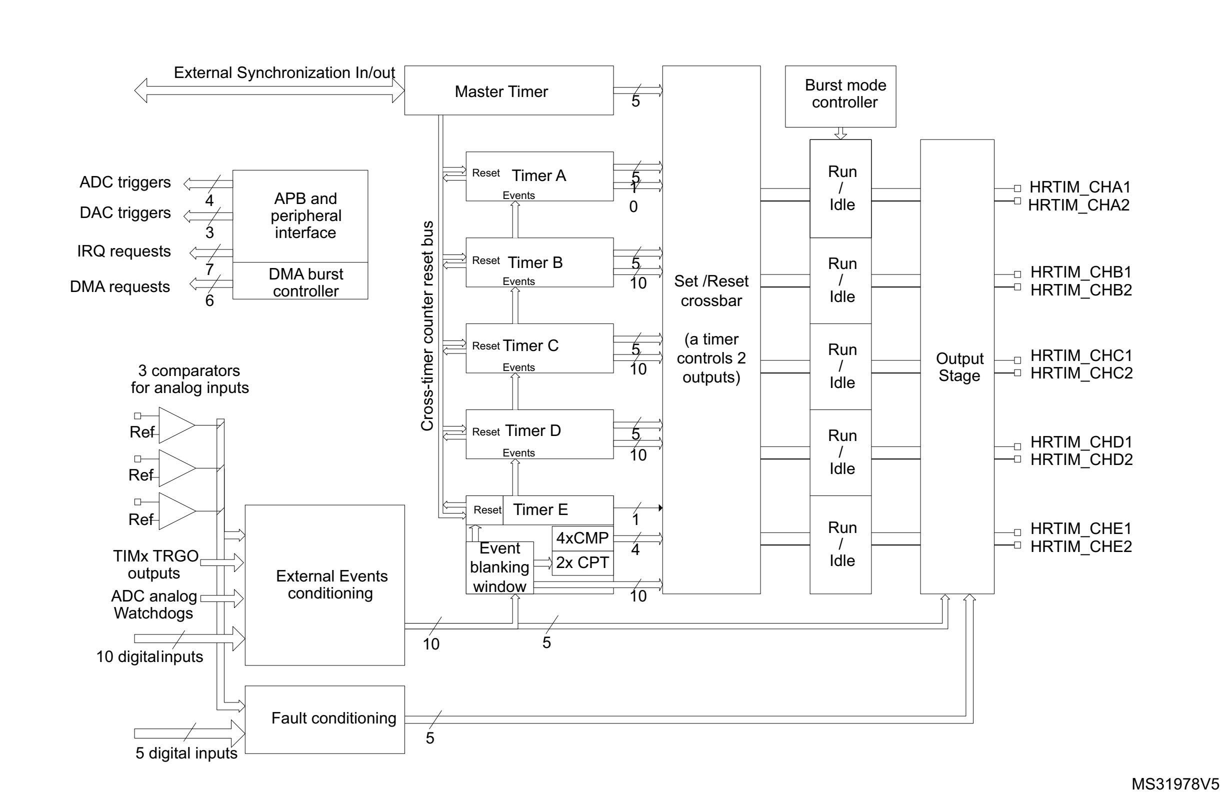
The HRTIM registers are split into 7 groups:

- Master timer registers
- Timer A to Timer E registers
- Common registers for features shared by all timer units

Note: As a writing convention, references to the 5 timing units in the text and in registers are generalized using the “x” letter, where x can be any value from A to E.

The block diagram of the timer is shown in *Figure 244*.

Figure 244. High-resolution timer block diagram



21.3.2 HRTIM pins and internal signals

The table here below summarizes the HRTIM inputs and outputs, both on-chip and off-chip.

Table 80. HRTIM Input/output summary

Signal name	Signal type	Description
HRTIM_CHA1, HRTIM_CHA2, HRTIM_CHB1, HRTIM_CHB2, HRTIM_CHC1, HRTIM_CHC2, HRTIM_CHD1, HRTIM_CHD2, HRTIM_CHE1, HRTIM_CHE2	Outputs	Main HRTIM timer outputs. They can be coupled by pairs (HRTIM_CHx1 & HRTIM_CHx2) with deadtime insertion or work independently.
HRTIM_FLT[5:1], HRTIM_FLT_in[5:1]	Digital input	Fault inputs: immediately disable the HRTIM outputs when asserted (5 on-chip inputs and 5 off-chip HRTIM_FLTx inputs).

Table 80. HRTIM Input/output summary (continued)

Signal name	Signal type	Description
SYSFLT	Digital input	System fault gathering MCU internal fault events (Clock security system, SRAM parity error, Cortex®-M4 lockup (HardFault), PVD output).
HRTIM_SCIN[3:1]	Digital Input	Synchronization inputs to synchronize the whole HRTIM with other internal or external timer resources: HRTIM_SCIN1: reserved HRTIM_SCIN2: the source is a regular TIMx timer (via on-chip interconnect) HRTIM_SCIN3: the source is an external HRTIM (via the HRTIM_SCIN input pins)
HRTIM_SCOUT[2:1]	Digital output	The purpose of this output is to cascade or synchronize several HRTIM instances, either on-chip or off-chip: HRTIM_SCOUT1: reserved HRTIM_SCOUT2: the destination is an off-chip HRTIM or peripheral (via HRTIM_SCOUT output pins)
HRTIM_EEV1[4:1]	Digital input	External events. Each of the 10 events can be selected among 4 sources, either on-chip (from other built-in peripherals: comparator, ADC analog watchdog, TIMx timers, trigger outputs) or off-chip (HRTIM_EEVx input pins)
HRTIM_EEV2[4:1]		
HRTIM_EEV3[4:1]		
HRTIM_EEV4[4:1]		
HRTIM_EEV5[4:1]		
HRTIM_EEV6[4:1]		
HRTIM_EEV7[4:1]		
HRTIM_EEV8[4:1]		
HRTIM_EEV9[4:1]		
HRTIM_EEV10[4:1]		
UPD_EN[3:1]	Digital input	HRTIM register update enable inputs (on-chip interconnect) trigger the transfer from shadow to active registers
BMtrig	Digital input	Burst mode trigger event (on-chip interconnect)
BMClk[4:1]	Digital input	Burst mode clock (on-chip interconnect)
ADCtrigOut[4:1]	Digital output	ADC start of conversion triggers
DACtrigOut[3:1]	Digital output	DAC conversion update triggers
IRQ[7:1]	Digital output	Interrupt requests
DMA[6:1]	Digital output	DMA requests

21.3.3 Clocks

The HRTIM must be supplied by the t_{HRTIM} system clock to offer a full resolution. The t_{HRTIM} clock period is evenly divided into up to 32 intermediate steps using an edge positioning logic. All clocks present in the HRTIM are derived from this reference clock.

Definition of terms

f_{HRTIM}: main HRTIM clock . All subsequent clocks are derived and synchronous with this source.

f_{HRCK}: high-resolution equivalent clock. Considering the f_{HRTIM} clock period division by 32, it is equivalent to a frequency of 144 x 32 = 4.608 GHz.

f_{DTG}: deadtime generator clock. For convenience, only the t_{DTG} period ($t_{DTG} = 1/f_{DTG}$) is used in this document.

f_{CHPFRQ}: chopper stage clock source.

f_{1STPW}: clock source defining the length of the initial pulse in chopper mode. For convenience, only the t_{1STPW} period ($t_{1STPW} = 1/f_{1STPW}$) is used in this document.

f_{BRST}: burst mode controller counter clock.

$f_{SAMPLING}$: clock needed to sample the fault or the external events inputs.

f_{FLTS}: clock derived from f_{HRTIM} which is used as a source for $f_{SAMPLING}$ to filter fault events.

f_{EEVS}: clock derived from f_{HRTIM} which is used as a source for $f_{SAMPLING}$ to filter external events.

Timer clock and prescaler

Each timer in the HRTIM has its own individual clock prescaler, which allows you to adjust the timer resolution. (See *Table 81*).

Table 81. Timer resolution and min. PWM frequency for f_{HRTIM} = 144 MHz

CKPSC[2:0]	Prescaling ratio	f_{HRCK} equivalent frequency	Resolution	Min PWM frequency
000	1	144 x 32 MHz = 4.608 GHz	217 ps	70.3 kHz
001	2	144 x 16MHz = 2.304 GHz	434 ps	35.1 kHz
010	4	144 x 8MHz = 1.152 GHz	868 ps	17.6 kHz
011	8	144 x 4MHz = 576 MHz	1.73 ns	8.8 kHz
100	16	144 x 2MHz = 288 MHz	3.47 ns	4.4 kHz
101	32	144 MHz	6.95 ns	2.2 kHz
110	64	144/2 MHz = 72 MHz	13.88 ns	1.1 kHz
111	128	144/4 MHz = 36 MHz	27.7 ns	550 Hz

The High-resolution is available for edge positioning, PWM period adjustment and externally triggered pulse duration.

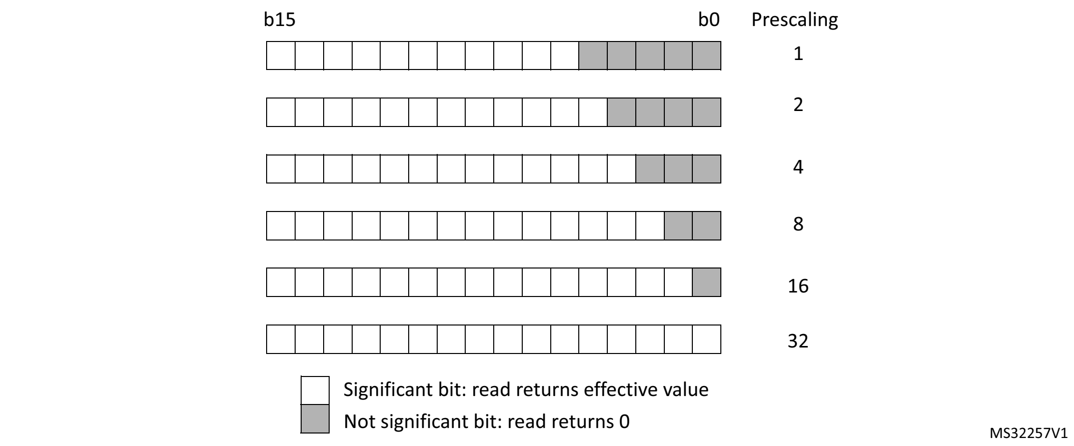
The high-resolution is not available for the following features

- Timer counter read and write accesses
- Capture unit

For clock prescaling ratios below 32 (CKPSC[2:0] <5), the least significant bits of the counter and capture registers are not significant. The least significant bits cannot be written (counter register only) and return 0 when read.

For instance, if CKPSC[2:0] = 2 (prescaling by 4), writing 0xFFFF into the counter register will yield an effective value of 0xFFF0. Conversely, any counter value between 0xFFFF and 0xFFF0 will be read as 0xFFF0.

Figure 245. Counter and capture register format vs clock prescaling factor


b15
b0
Prescaling
1
2
4
8
16
32
Significant bit: read returns effective value
Not significant bit: read returns 0
MS32257V1

Initialization

At start-up, it is mandatory to initialize first the prescaler bitfields before writing the compare and period registers. Once the timer is enabled (MCEN or TxCEN bit set in the HRTIM_MCR register), the prescaler cannot be modified.

When multiple timers are enabled, the prescalers are synchronized with the prescaler of the timer that was started first.

Warning: **It is possible to have different prescaling ratios in the master and TIMA..E timers only if the counter and output behavior does not depend on other timers' information and signals. It is mandatory to configure identical prescaling ratios in these timers when one of the following events is propagated from one timing unit (or master timer) to another: output set/reset event, counter reset event, update event, external event filter or capture triggers. Prescaler factors not equal will yield to unpredictable results.**

Deadtime generator clock

The deadtime prescaler is supplied by f_{HRTIM} / 8 / $2^{(DTPRSC[2:0])}$, programmed with DTPRSC[2:0] bits in the HRTIM_DTxR register.

t_{DTG} ranges from 868 ps to 6.94 ns for f_{HRTIM} = 144 MHz.

Chopper stage clock

The chopper stage clock source f_{CHPFRQ} is derived from f_{HRTIM} with a division factor ranging from 16 to 31, so that 562.5 kHz <= f_{CHPFRQ} <= 9 MHz for f_{HRTIM} = 144 MHz.

t_{1STPW} is the length of the initial pulse in chopper mode, programmed with the STRPW[3:0] bits in the HRTIM_CHPxR register, as follows:

t_{1STPW} = (STRPW[3:0]+1) x 16 x t_{HRTIM}.

It uses f_{HRTIM} / 16 as clock source (9 MHz for f_{HRTIM}= 144 MHz).

Burst Mode Prescaler

The burst mode controller counter clock f_{BRST} can be supplied by several sources, among which one is derived from f_{HRTIM}.

In this case, f_{BRST} ranges from f_{HRTIM} to f_{HRTIM} / 32768 (4.4 kHz for f_{HRTIM} = 144 MHz).

Fault input sampling clock

The fault input noise rejection filter has a time constant defined with $f_{SAMPLING}$ which can be either f_{HRTIM} or f_{FLTS}.

f_{FLTS} is derived from f_{HRTIM} and ranges from 144 MHz to 18 MHz for f_{HRTIM} = 144 MHz.

External Event input sampling clock

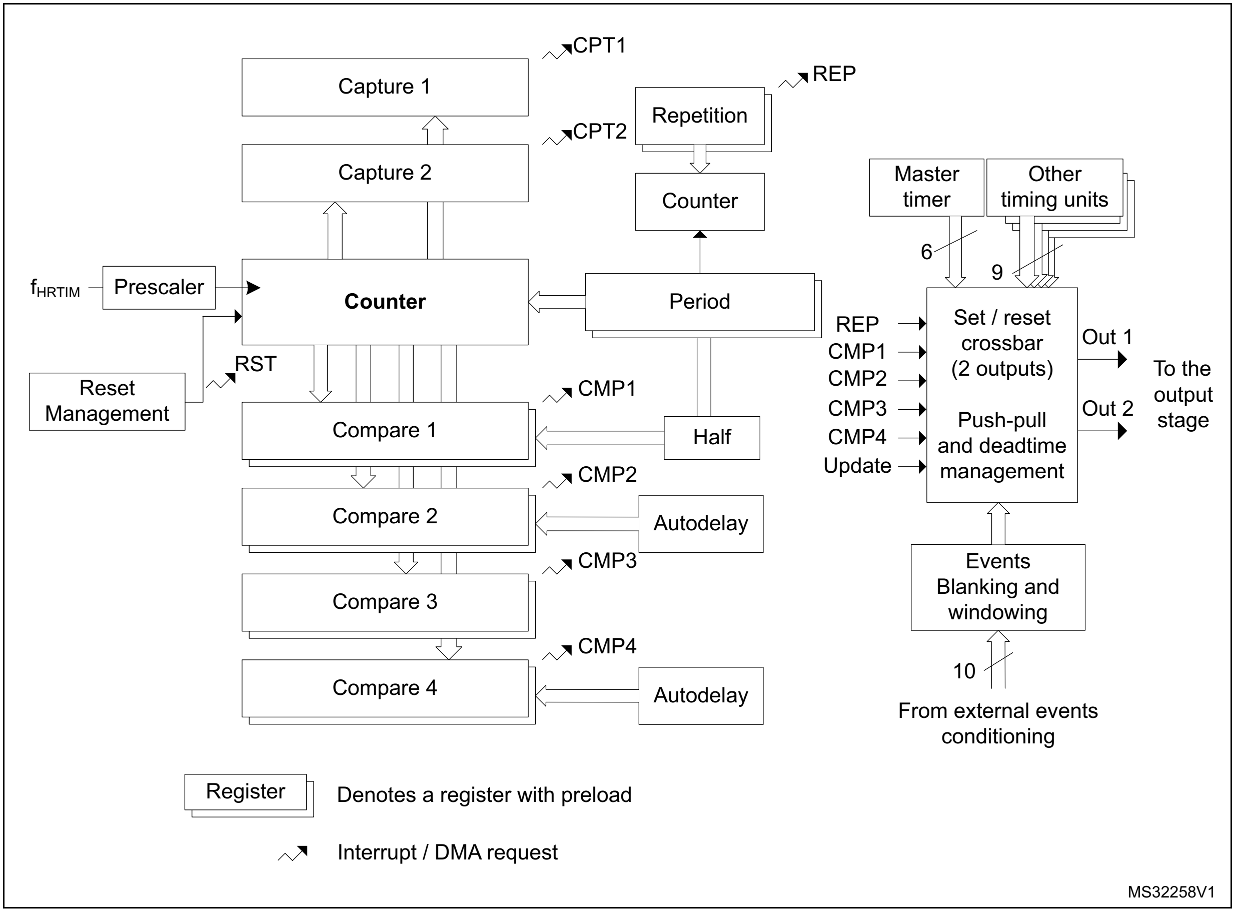
The fault input noise rejection filter has a time constant defined with $f_{SAMPLING}$ which can be either f_{HRTIM} or f_{EEVS}.

f_{EEVS} is derived from f_{HRTIM} and ranges from 144 MHz to 18 MHz for f_{HRTIM} = 144 MHz.

21.3.4 Timer A..E timing units

The HRTIM embeds 5 identical timing units made of a 16-bit up-counter with an auto-reload mechanism to define the counting period, 4 compare and 2 capture units, as per *Figure 246*. Each unit includes all control features for 2 outputs, so that it can operate as a standalone timer.

Figure 246. Timer A..E overview



The period and compare values must be within a lower and an upper limit related to the high-resolution implementation and listed in *Table 82*:

- The minimum value must be greater than or equal to 3 periods of the f_{HRTIM} clock
- The maximum value must be less than or equal to 0xFFFF - 1 periods of the f_{HRTIM} clock

Table 82. Period and Compare registers min and max values

CKPSC[2:0] value	Min	Max
0	0x0060	0xFFDF
1	0x0030	0xFFEF
2	0x0018	0xFFF7
3	0x000C	0xFFFB

Table 82. Period and Compare registers min and max values (continued)

CKPSC[2:0] value	Min	Max
4	0x0006	0xFFFD
≥ 5	0x0003	0xFFFD

Note: *A compare value greater than the period register value will not generate a compare match event.*

Counter operating mode

Timer A..E can operate in continuous (free-running) mode or in single-shot manner where counting is started by a reset event, using the CONT bit in the HRTIM_TIMxCR control register. An additional RETRIG bit allows you to select whether the single-shot operation is retriggerable or non-retriggerable. Details of operation are summarized on *Table 83* and on *Figure 247* and *Figure 248*.

Table 83. Timer operating modes

CONT	RETRIG	Operating mode	Start / Stop conditions Clocking and event generation
0	0	Single-shot Non-retriggerable	Setting the TxEN bit enables the timer but does not start the counter. A first reset event starts the counting and any subsequent reset is ignored until the counter reaches the PER value. The PER event is then generated and the counter is stopped. A reset event re-starts the counting operation from 0x0000.
0	1	Single-shot Retriggerable	Setting the TxEN bit enables the timer but does not start the counter. A reset event starts the counting if the counter is stopped, otherwise it clears the counter. When the counter reaches the PER value, the PER event is generated and the counter is stopped. A reset event re-starts the counting operation from 0x0000.
1	X	Continuous mode	Setting the TxEN bit enables the timer and starts the counter simultaneously. When the counter reaches the PER value, it rolls-over to 0x0000 and resumes counting. The counter can be reset at any time.

The TxEN bit can be cleared at any time to disable the timer and stop the counting.

Figure 247. Continuous timer operation

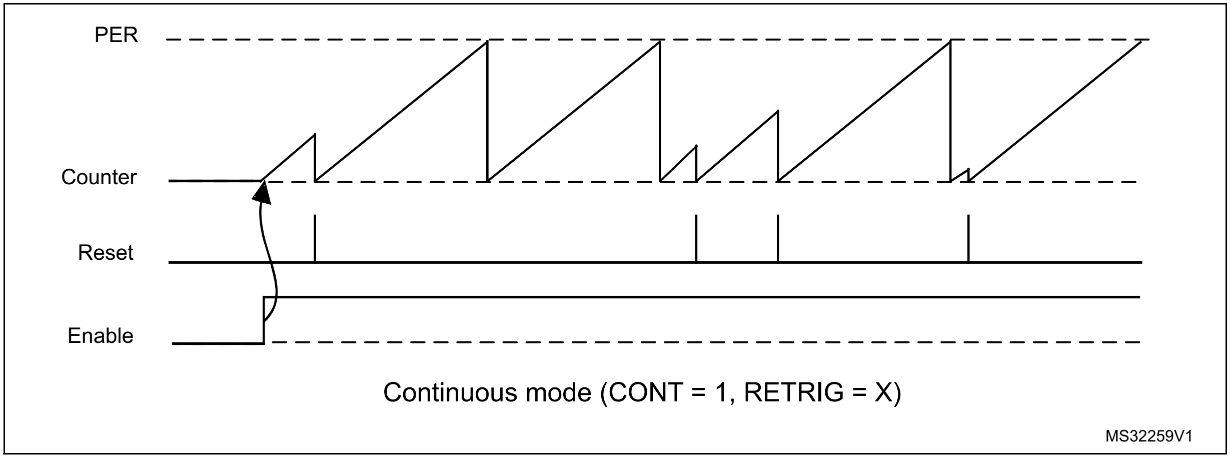
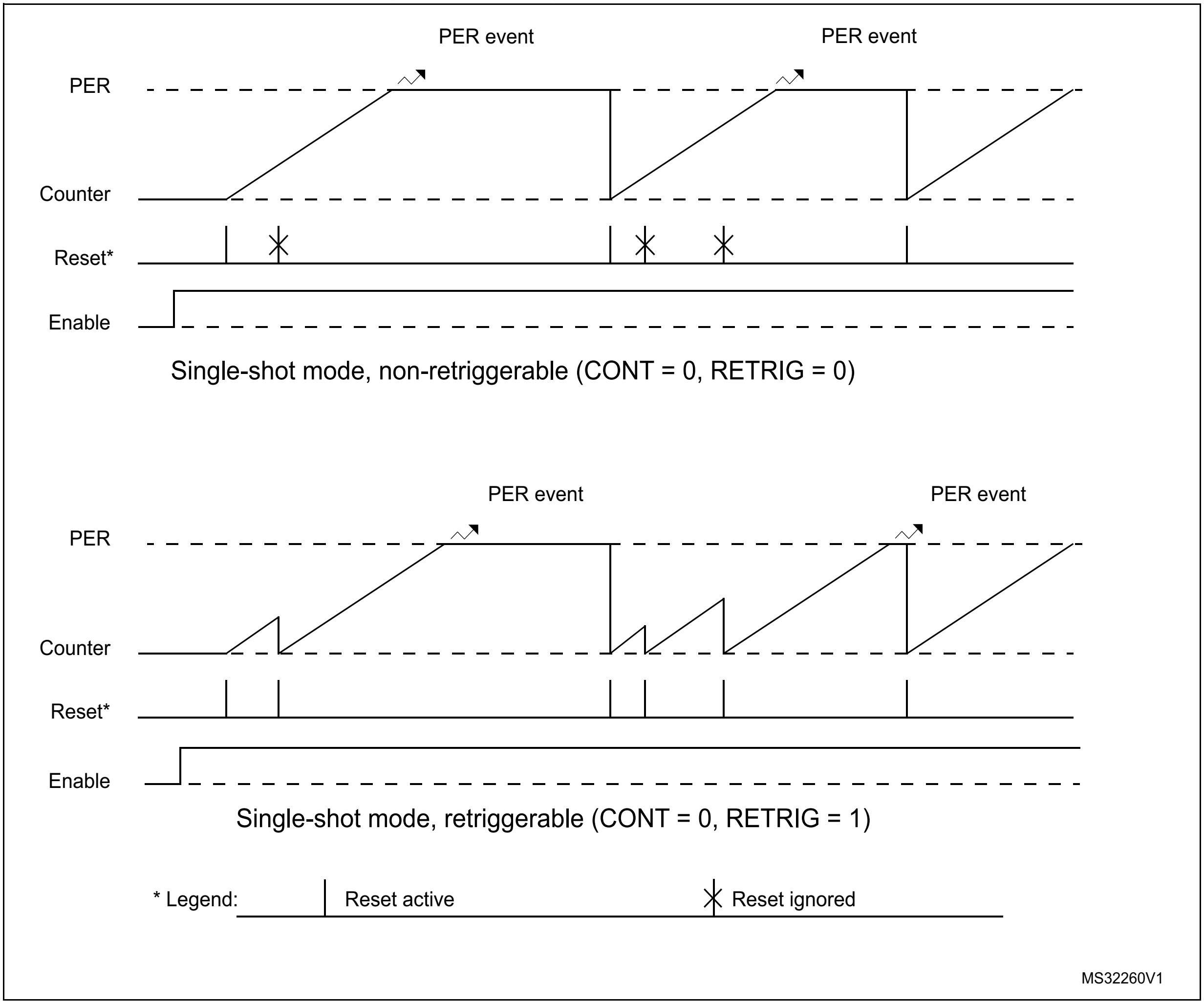


Figure 248. Single-shot timer operation



Roll-over event

A counter roll-over event is generated when the counter goes back to 0 after having reached the period value set in the HRTIM_PERxR register in continuous mode.

This event is used for multiple purposes in the HRTIM:

- To set/reset the outputs
- To trigger the register content update (transfer from preload to active)
- To trigger an IRQ or a DMA request
- To serve as a burst mode clock source or a burst start trigger
- as an ADC trigger
- To decrement the repetition counter

If the initial counter value is above the period value when the timer is started, or if a new period is set while the counter is already above this value, the counter is not reset: it will overflow at the maximum period value and the repetition counter will not decrement.

Timer reset

The reset of the timing unit counter can be triggered by up to 30 events that can be selected simultaneously in the HRTIM_RSTxR register, among the following sources:

- The timing unit: Compare 2, Compare 4 and Update (3 events)
- The master timer: Reset and Compare 1..4 (5 events)
- The external events EXTEVNT1..10 (10 events)
- All other timing units (e.g. Timer B..E for timer A): Compare 1, 2 and 4 (12 events)

Several events can be selected simultaneously to handle multiple reset sources. In this case, the multiple reset requests are ORed. When 2 counter reset events are generated within the same f_{HRTIM} clock cycle, the last counter reset is taken into account.

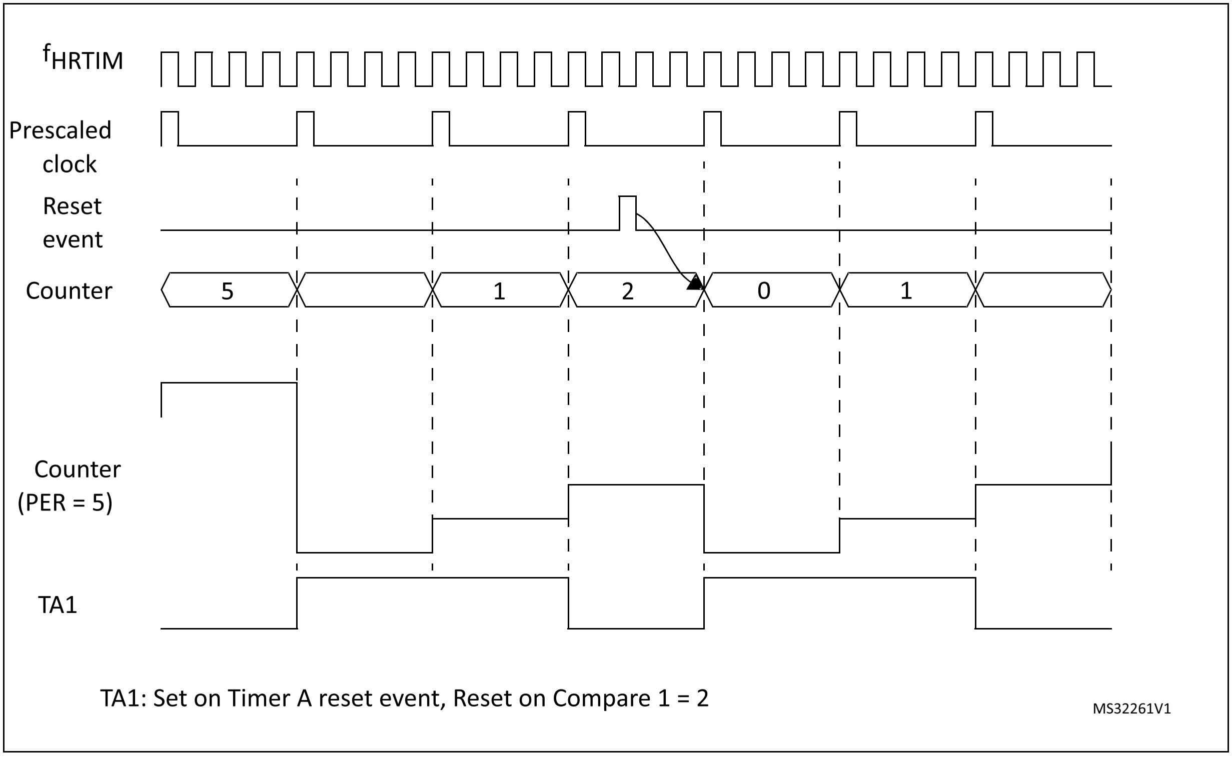
Additionally, it is possible to do a software reset of the counter using the TxRST bits in the HRTIM_CR2 register. These control bits are grouped into a single register to allow the simultaneous reset of several counters.

The reset requests are taken into account only once the related counters are enabled (TxCEN bit set).

When the f_{HRTIM} clock prescaling ratio is above 32 (counting period above f_{HRTIM}), the counter reset event is delayed to the next active edge of the prescaled clock. This allows to maintain a jitterless waveform generation when an output transition is synchronized to the reset event (typically a constant Ton time converter).

Figure 249 shows how the reset is handled for a clock prescaling ratio of 128 (f_{HRTIM} divided by 4).

Figure 249. Timer reset resynchronization (prescaling ratio above 32)



TA1: Set on Timer A reset event, Reset on Compare 1 = 2

Repetition counter

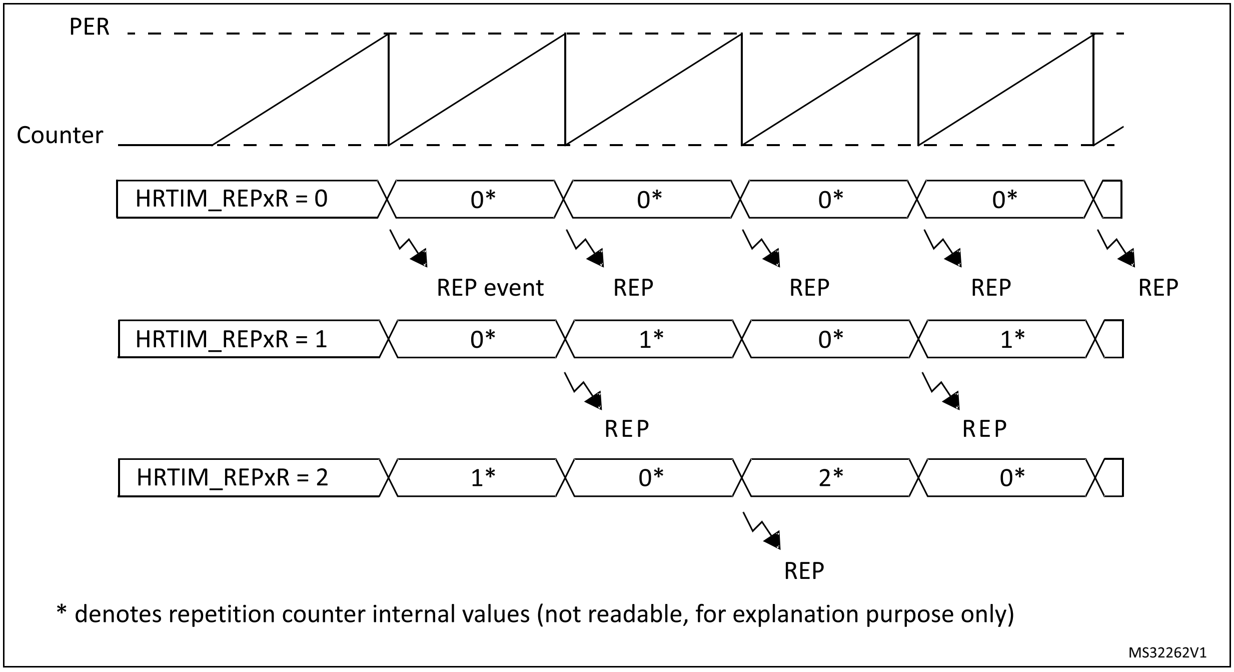
A common software practice is to have an interrupt generated when the period value is reached, so that the maximum amount of time is left for processing before the next period begins. The main purpose of the repetition counter is to adjust the period interrupt rate and off-load the CPU by decoupling the switching frequency and the interrupt frequency.

The timing units have a repetition counter. This counter cannot be read, but solely programmed with an auto-reload value in the HRTIM_REPxR register.

The repetition counter is initialized with the content of the HRTIM_REPxR register when the timer is enabled (TXCEN bit set). Once the timer has been enabled, any time the counter is cleared, either due to a reset event or due to a counter roll-over, the repetition counter is decreased. When it reaches zero, a REP interrupt or a DMA request is issued if enabled (REPIE and REPDE bits in the HRTIM_DIER register).

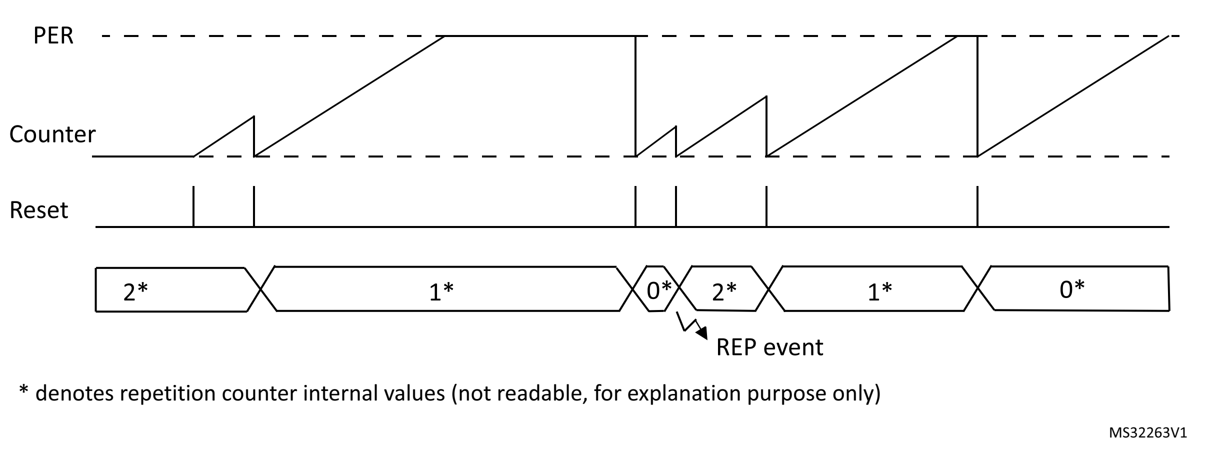
If the HRTIM_REPxR register is set to 0, an interrupt is generated for each and every period. For any value above 0, a REP interrupt is generated after (HRTIM_REPxR + 1) periods. *Figure 250* presents the repetition counter operation for various values, in continuous mode.

Figure 250. Repetition rate vs HRTIM_REPxR content in continuous mode



The repetition counter can also be used when the counter is reset before reaching the period value (variable frequency operation) either in continuous or in single-shot mode (*Figure 251* here-below). The reset causes the repetition counter to be decremented, at the exception of the very first start following counter enable (TxCEN bit set).

Figure 251. Repetition counter behavior in single-shot mode


PER
Counter
Reset
2*
1*
0*
2*
1*
0*
REP event
* denotes repetition counter internal values (not readable, for explanation purpose only)
MS32263V1

A reset or start event from the HRTIM_SCIN[3:1] source causes the repetition to be decremented as any other reset. However, in SYNCIN-started single-shot mode (SYNCSTRTx bit set in the HRTIM_TIMxCR register), the repetition counter will be decremented only on the 1st reset event following the period. Any subsequent reset will not alter the repetition counter until the counter is re-started by a new request on HRTIM_SCIN[3:1] inputs.

Set / reset crossbar

A “set” event correspond to a transition to the output active state, while a “reset” event corresponds to a transition to the output inactive state.
The polarity of the waveform is defined in the output stage to accommodate positive or negative logic external components: an active level corresponds to a logic level 1 for a positive polarity (POLx = 0), and to a logic level 0 for a negative polarity (POLx = 1).

Each of the timing units handles the set/reset crossbar for two outputs. These 2 outputs can be set, reset or toggled by up to 32 events that can be selected among the following sources:

- The timing unit: Period, Compare 1..4, register update (6 events)
- The master timer: Period, Compare 1..4, HRTIM synchronization (6 events)
- All other timing units (e.g. Timer B..E for timer A): TIMEVNT1..9 (9 events described in *Table 84*)
- The external events EXTEVNT1..10 (10 events)
- A software forcing (1 event)

The event sources are ORed and multiple events can be simultaneously selected.

Each output is controlled by two 32-bit registers, one coding for the set (HRTIM_SETxyR) and another one for the reset (HRTIM_RSTxyR), where x stands for the timing unit: A..E and y stands for the output 1or 2 (e.g. HRTIM_SETA1R, HRTIM_RSTC2R,...).

If the same event is selected for both set and reset, it will toggle the output. It is not possible to toggle the output state more than one time per t_{HRTIM} period: in case of two consecutive toggling events within the same cycle, only the first one is considered.

The set and reset requests are taken into account only once the counter is enabled (TxCEN bit set), except if the software is forcing a request to allow the prepositioning of the outputs at timer start-up.

Table 84 summarizes the events from other timing units that can be used to set and reset the outputs. The number corresponds to the timer events (such as TIMEVNTx) listed in the register, and empty locations are indicating non-available events.

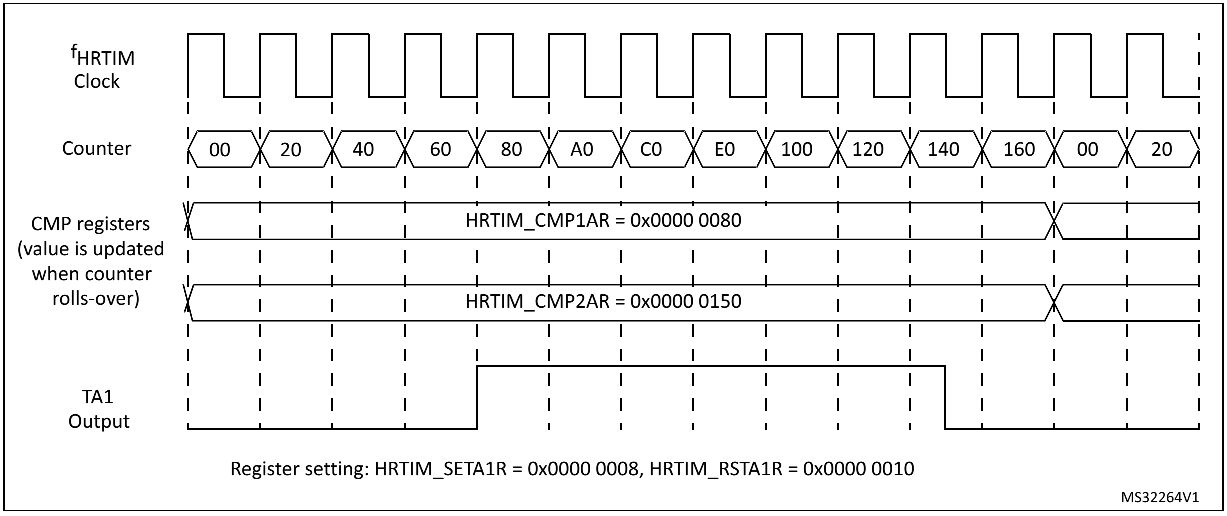
For instance, Timer A outputs can be set or reset by the following events: Timer B Compare1, 2 and 4, Timer C Compare 2 and 3,... and Timer E Compare 3 will be listed as TIMEVNT8 in HRTIM_SETA1R.

Table 84. Events mapping across Timer A to E

Source		Timer A				Timer B				Timer C				Timer D				Timer E			
		CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4
Destination	Timer A	-	-	-	-	1	2	-	3	-	4	5	-	6	7	-	-	-	-	8	9
	Timer B	1	2	-	3	-	-	-	-	-	-	4	5	-	-	6	7	8	9	-	-
	Timer C	-	1	2	-	-	3	4	-	-	-	-	-	-	5	-	6	-	7	8	9
	Timer D	1	-	-	2	-	3	-	4	5	-	6	7	-	-	-	-	8	-	-	9
	Timer E	-	-	1	2	-	-	3	4	5	6	-	-	7	8	-	9	-	-	-	-

Figure 252 represents how a PWM signal is generated using two compare events.

Figure 252. Compare events action on outputs: set on compare 1, reset on compare 2


f_{HRTIM} Clock
Counter
00
20
40
60
80
A0
C0
E0
100
120
140
160
00
20
CMP registers (value is updated when counter rolls-over)
HRTIM_CMP1AR = 0x0000 0080
HRTIM_CMP2AR = 0x0000 0150
TA1 Output
Register setting: HRTIM_SETA1R = 0x0000 0008, HRTIM_RSTA1R = 0x0000 0010
MS32264V1

Set/Reset on Update events

A set or reset event on update is done at low resolution. When CKPSC[2:0] < 5, the high-resolution delay is set to its maximum value so that a set/reset event on update will always lag as compared to other compare set/reset events, with a jitter varying between 0 and 31/32 of a f_{HRTIM} clock period.

Half mode

This mode aims at generating square signal with fixed 50% duty cycle and variable frequency (typically for converters using resonant topologies). It allows to have the duty cycle automatically forced to half of the period value when a new period is programmed.

This mode is enabled by writing HALF bit to 1 in the HRTIM_TIMxCR register. When the HRTIM_PERxR register is written, it causes an automatic update of the Compare 1 value with HRTIM_PERxR/2 value.

The output on which a square wave is generated must be programmed to have one transition on CMP1 event, and one transition on the period event, as follows:

- HRTIM_SETxyR = 0x0000 0008, HRTIM_RSTxyR = 0x0000 0004, or
- HRTIM_SETxyR = 0x0000 0004, HRTIM_RSTxyR = 0x0000 0008

The HALF mode overrides the content of the HRTIM_CMP1xR register. The access to the HRTIM_PERxR register only causes Compare 1 internal register to be updated. The user-accessible HRTIM_CMP1xR register is not updated with the HRTIM_PERxR / 2 value.

When the preload is enabled (PREEN = 1, MUDIS, TxUDIS), Compare 1 active register is refreshed on the Update event. If the preload is disabled (PREEN= 0), Compare 1 active register is updated as soon as HRTIM_PERxR is written.

The period must be greater than or equal to 6 periods of the f_{HRTIM} clock (0xC0 if CKPSC[2:0] = 0, 0x60 if CKPSC[2:0] = 1, 0x30 if CKPSC[2:0] = 2,...) when the HALF mode is enabled.

Capture

The timing unit has the capability to capture the counter value, triggered by internal and external events. The purpose is to:

- measure events arrival timings or occurrence intervals
- update Compare 2 and Compare 4 values in auto-delayed mode (see *Auto-delayed mode*).

The capture is done with f_{HRTIM} resolution: for a clock prescaling ratio below 32 (CKPSC[2:0] < 5), the least significant bits of the register are not significant (read as 0).

The timer has 2 capture registers: HRTIM_CPT1xR and HRTIM_CPT2xR. The capture triggers are programmed in the HRTIM_CPT1xCR and HRTIM_CPT2xCR registers.

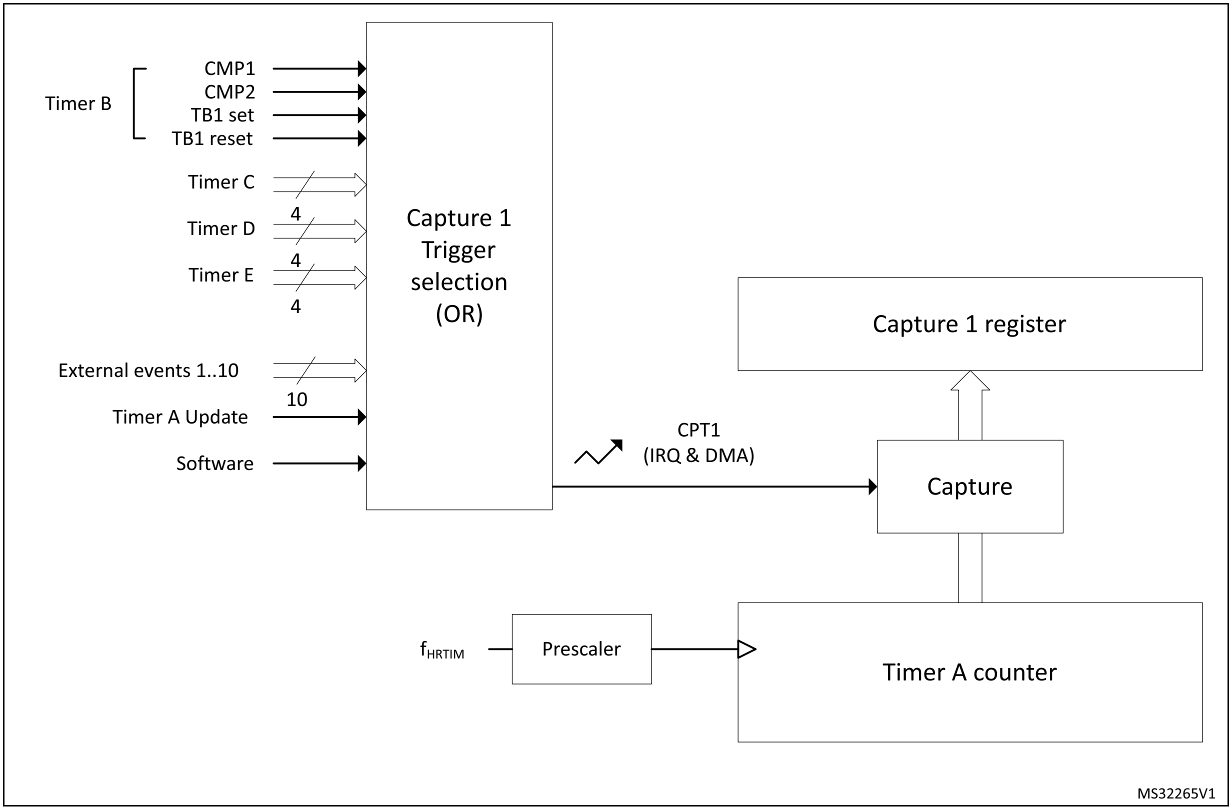
The capture of the timing unit counter can be triggered by up to 28 events that can be selected simultaneously in the HRTIM_CPT1xCR and HRTIM_CPT2xCR registers, among the following sources:

- The external events, EXTEVNT1..10 (10 events)
- All other timing units (e.g. Timer B..E for timer A): Compare 1, 2 and output 1 set/reset events (16 events)
- The timing unit: Update (1 event)
- A software capture (1 event)

Several events can be selected simultaneously to handle multiple capture triggers. In this case, the concurrent trigger requests are ORed. The capture can generate an interrupt or a DMA request when CPTxIE and CPTxDE bits are set in the HRTIM_TIMxDIER register.

Over-capture is not prevented by the circuitry: a new capture is triggered even if the previous value was not read, or if the capture flag was not cleared.

Figure 253. Timing unit capture circuitry



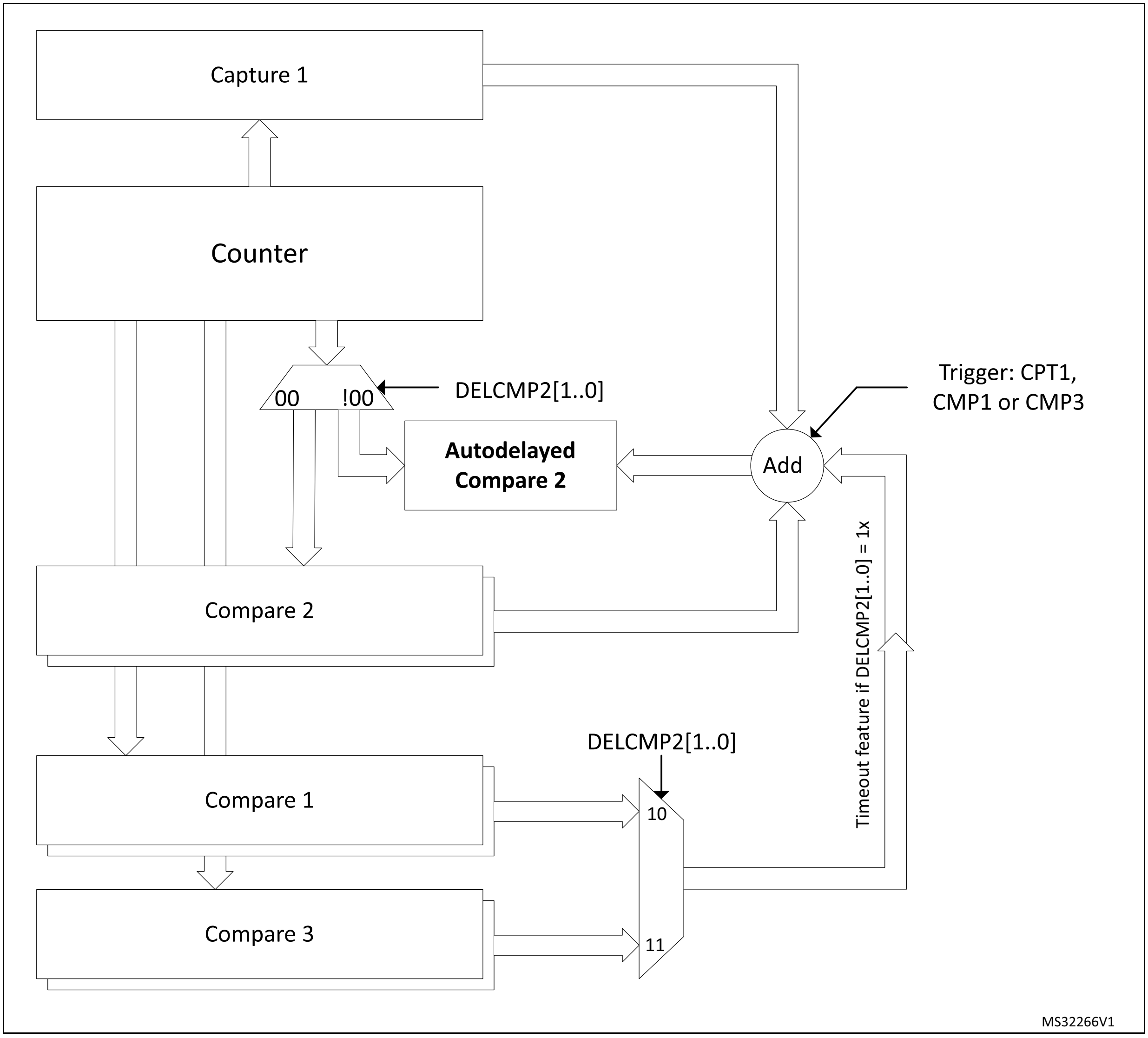
Auto-delayed mode

This mode allows to have compare events generated relatively to capture events, so that for instance an output change can happen with a programmed timing following a capture. In this case, the compare match occurs independently from the timer counter value. It enables the generation of waveforms with timings synchronized to external events without the need of software computation and interrupt servicing.

As long as no capture is triggered, the content of the HRTIM_CMPxR register is ignored (no compare event is generated when the counter value matches the Compare value. Once the capture is triggered, the compare value programmed in HRTIM_CMPxR is summed with the captured counter value in HRTIM_CPTxyR, and it updates the internal auto-delayed compare register, as seen on *Figure 254*. The auto-delayed compare register is internal to the timing unit and cannot be read. The HRTIM_CMPxR preload register is not modified after the calculation.

This feature is available only for Compare 2 and Compare 4 registers. Compare 2 is associated with capture 1, while Compare 4 is associated with capture 2. HRTIM_CMP2xR and HRTIM_CMP4xR Compares cannot be programmed with a value below 3 f_{HRTIM} clock periods, as in the regular mode.

Figure 254. Auto-delayed overview (Compare 2 only)



The auto-delayed Compare is only valid from the capture up to the period event: once the counter has reached the period value, the system is re-armed with Compare disabled until a capture occurs.

DELCMP2[1:0] and DELCMP4[1:0] bits in HRTIM_TIMxCR register allow to configure the auto-delayed mode as follows:

- 00
  Regular compare mode: HRTIM_CMP2xR and HRTIM_CMP4xR register contents are directly compared with the counter value.
- 01
  Auto-delayed mode: Compare 2 and Compare 4 values are recomputed and used for comparison with the counter after a capture 1/2 event.

- 1X
  Auto-delayed mode with timeout: Compare 2 and Compare 4 values are recomputed and used for comparison with the counter after a capture 1/2 event or after a Compare 1 match (DELCMPx[1:0]= 10) or a Compare 3 match (DELCMPx[1:0]= 11) to have a timeout function if capture 1/2 event is missing.

When the capture occurs, the comparison is done with the (HRTIM_CMP2/4xR + HRTIM_CPT1/2xR) value. If no capture is triggered within the period, the behavior depends on the DELCMPx[1:0] value:

- DELCMPx[1:0] = 01: the compare event is not generated
- DELCMPx[1:0] = 10 or 11: the comparison is done with the sum of the 2 compares (for instance HRTIM_CMP2xR + HRTIM_CMP1xR). The captures are not taken into account if they are triggered after CMPx + CMP1 (resp. CMPx + CMP3).

The captures are enabled again at the beginning of the next PWM period.

If the result of the auto-delayed summation is above 0xFFFF (overflow), the value is ignored and no compare event will be generated until a new period is started.

Note: DELCMPx[1:0] bitfield must be reset when reprogrammed from one value to the other to re-initialize properly the auto-delayed mechanism, for instance:

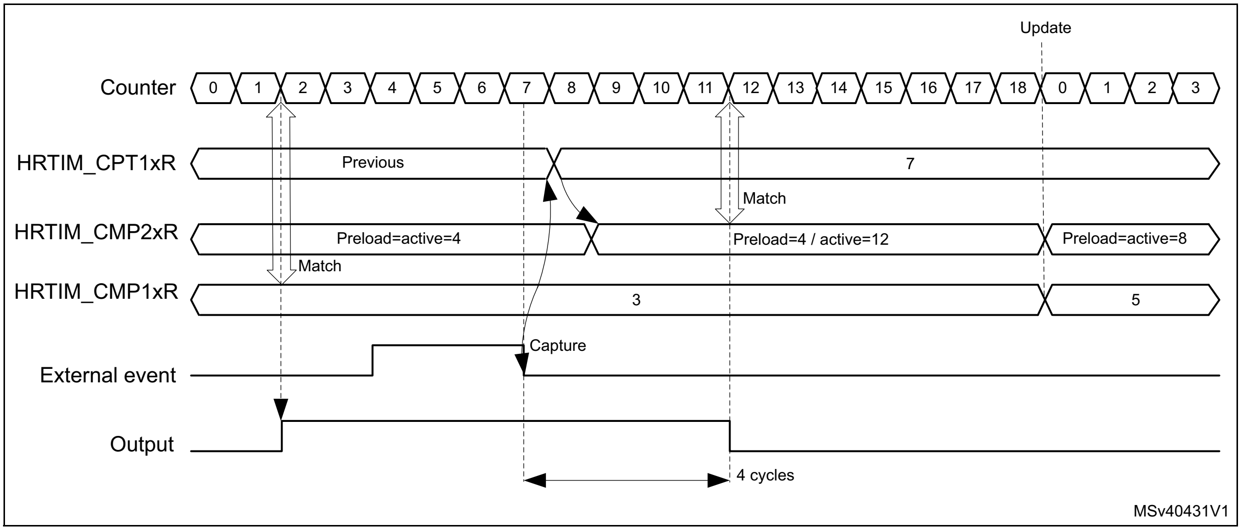
- DELCMPx[1:0] = 10
- DELCMPx[1:0] = 00
- DELCMPx[1:0] = 11

As an example, *Figure 255* shows how the following signal can be generated:

- Output set when the counter is equal to Compare 1 value
- Output reset 4 cycles after a falling edge on a given external event

Note: To simplify the figure, the high-resolution is not used in this example (CKPSC[2:0] = 101), thus the counter is incremented at the f_{HRTIM} rate. Similarly, the external event signal is shown without any resynchronization delay: practically, there is a delay of 1 to 2 f_{HRTIM} clock periods between the falling edge and the capture event due to an internal resynchronization stage which is necessary to process external input signals.

Figure 255. Auto-delayed compare



A regular compare channel (e.g. Compare 1) is used for the output set: as soon as the counter matches the content of the compare register, the output goes to its active state.

A delayed compare is used for the output reset: the compare event can be generated only if a capture event has occurred. No event is generated when the counter matches the delayed compare value (counter = 4). Once the capture event has been triggered by the external event, the content of the capture register is summed to the delayed compare value to have the new compare value. In the example, the auto-delayed value 4 is summed to the capture equal to 7 to give a value of 12 in the auto-delayed compare register. From this time on, the compare event can be generated and will happen when the counter is equal to 12, causing the output to be reset.

Overcapture management in auto-delayed mode

Overcapture is prevented when the auto-delayed mode is enabled (DELCMPx[1:0] = 01, 10, 11).

When multiple capture requests occur within the same counting period, only the first capture is taken into account to compute the auto-delayed compare value. A new capture is possible only:

- Once the auto-delayed compare has matched the counter value (compare event)
- Once the counter has rolled over (period)
- Once the timer has been reset

Changing auto-delayed compare values

When the auto-delayed compare value is preloaded (PREEN bit set), the new compare value is taken into account on the next coming update event (for instance on the period event), regardless of when the compare register was written and if the capture occurred (see *Figure 255*, where the delay is changed when the counter rolls over).

When the preload is disabled (PREEN bit reset), the new compare value is taken into account immediately, even if it is modified after the capture event has occurred, as per the example below:

1. At t1, DELCMP2 = 1.
2. At t2, CMP2_act = 0x40 => comparison disabled
3. At t3, a capture event occurs capturing the value CPTR1 = 0x20. => comparison enabled, compare value = 0x60
4. At t4, CMP2_act = 0x100 (before the counter reached value CPTR1 + 0x40) => comparison still enabled, new compare value = 0x120
5. At t5, the counter reaches the period value => comparison disabled, cmp2_act = 0x100

Similarly, if the CMP1(CMP3) value changes while DELCMPx = 10 or 11, and preload is disabled:

1. At t1, DELCMP2 = 2.
2. At t2, CMP2_act = 0x40 => comparison disabled
3. At t3, CMP3 event occurs - CMP3_act = 0x50 before capture 1 event occurs => comparison enabled, compare value = 0x90
4. At t4, CMP3_act = 0x100 (before the counter reached value 0x90) => comparison still enabled, Compare 2 event will occur at = 0x140

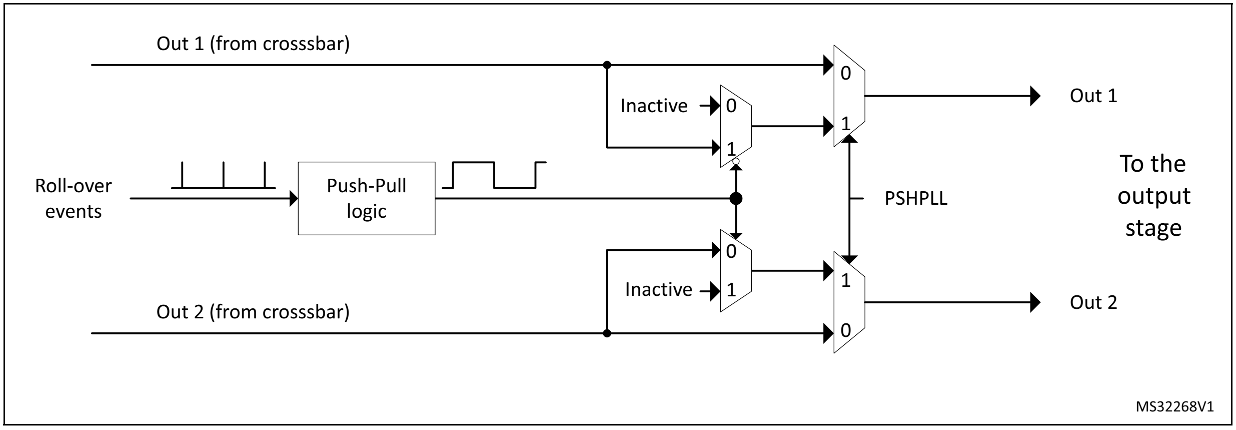
Push-pull mode

This mode primarily aims at driving converters using push-pull topologies. It also needs to be enabled when the delayed idle protection is required, typically for resonant converters (refer to *Section 21.3.9: Delayed Protection*).

The push-pull mode is enabled by setting PSHPLL bit in the HRTIM_TIMxCR register.

It applies the signals generated by the crossbar to output 1 and output 2 alternatively, on the period basis, maintaining the other output to its inactive state. The redirection rate (push-pull frequency) is defined by the timer's period event, as shown on *Figure 256*. The push-pull period is twice the timer counting period.

Figure 256. Push-pull mode block diagram



The push-pull mode is only available when the timer operates in continuous mode: the counter must not be reset once it has been enabled (TxCEN bit set). It is necessary to disable the timer to stop a push-pull operation and to reset the counter before re-enabling it.

The signal shape is defined using HRTIM_SETxyR and HRTIM_RSTxyR for both outputs. It is necessary to have HRTIM_SETx1R = HRTIM_SETx2R and HRTIM_RSTx1R = HRTIM_RSTx2R to have both outputs with identical waveforms and to achieve a balanced operation. Still, it is possible to have different programming on both outputs for other uses.

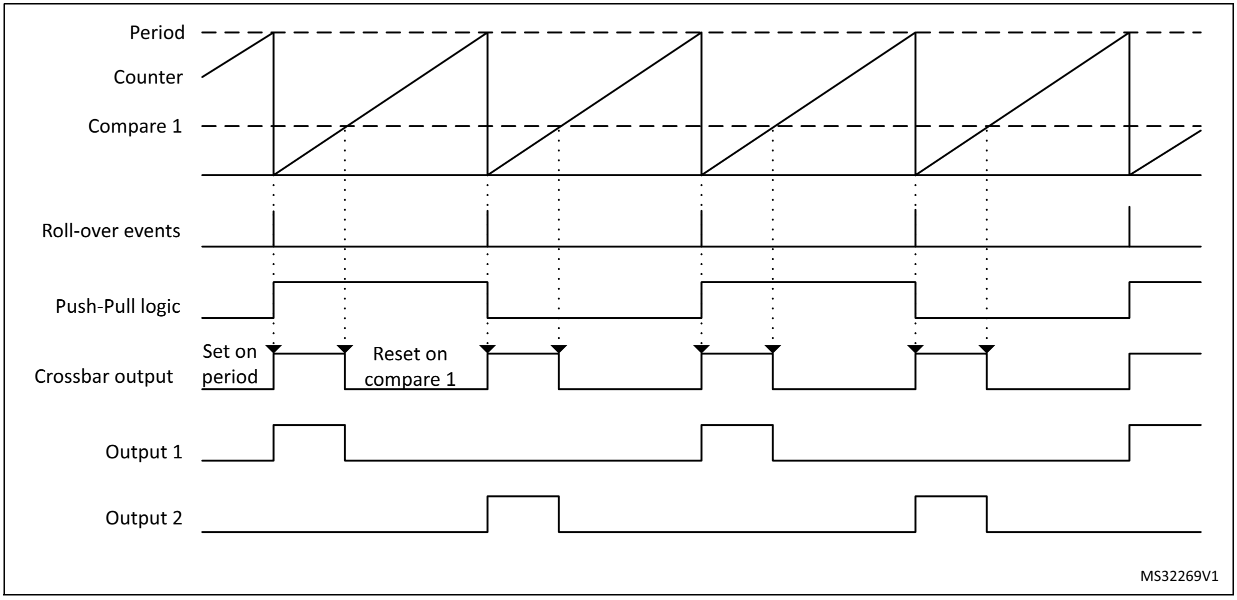
Note: The push-pull operation cannot be used when a deadtime is enabled (mutually exclusive functions).

The CPPSAT status bit in HRTIM_TIMxISR indicates on which output the signal is currently active. CPPSTAT is reset when the push-pull mode is disabled.

In the example given on *Figure 257*, the timer internal waveform is defined as follows:

- Output set on period event
- Output reset on Compare 1 match event

Figure 257. Push-pull mode example



Deadtime

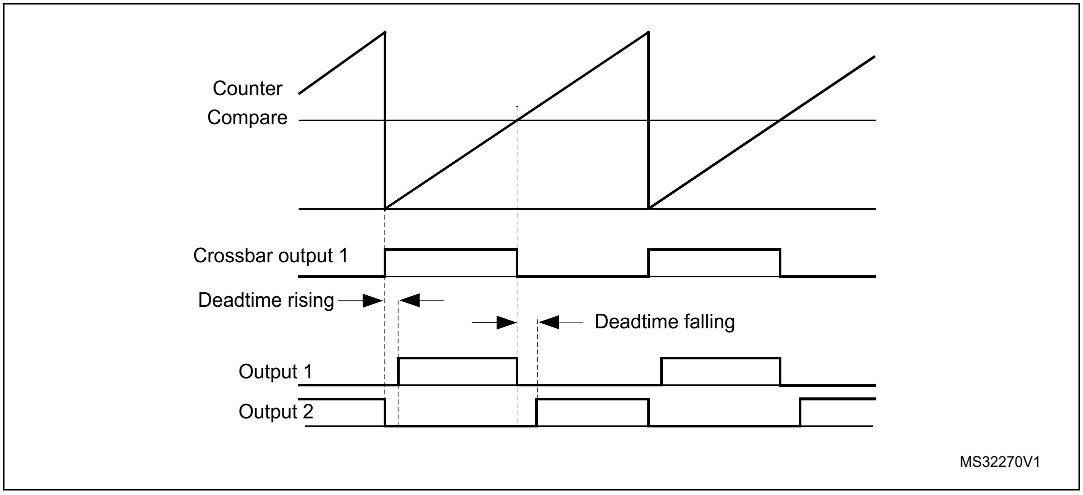
A deadtime insertion unit allows to generate a couple of complementary signals from a single reference waveform, with programmable delays between active state transitions. This is commonly used for topologies using half-bridges or full bridges. It simplifies the software: only 1 waveform is programmed and controlled to drive two outputs.

The Dead time insertion is enabled by setting DTEN bit in HRTIM_OUTxR register. The complementary signals are built based on the reference waveform defined for output 1, using HRTIM_SETx1R and HRTIM_RSTx1R registers: HRTIM_SETx2R and HRTIM_RSTx2R registers are not significant when DTEN bit is set.

Note: The deadtime cannot be used simultaneously with the push-pull mode.

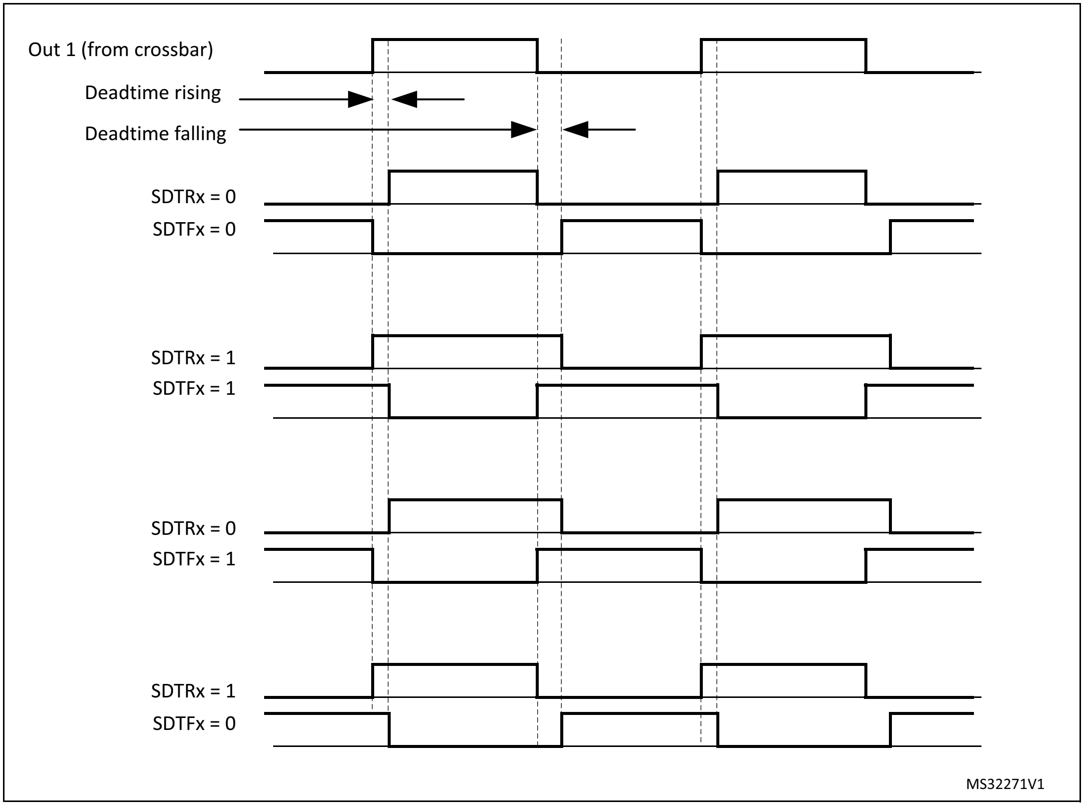
Two deadtimes can be defined in relationship with the rising edge and the falling edge of the reference waveform, as in *Figure 258*.

Figure 258. Complementary outputs with deadtime insertion



Negative deadtime values can be defined when some control overlap is required. This is done using the deadtime sign bits (SDTFx and SDTRx bits in HRTIM_DTxR register). *Figure 259* shows complementary signal waveforms depending on respective signs.

Figure 259. Deadtime insertion vs deadtime sign (1 indicates negative deadtime)



The deadtime values are defined with DTFx[8:0] and DTRx[8:0] bitfields and based on a specific clock prescaled according to DTPRSC[2:0] bits, as follows:

t_{DTx} = +/- DTx[8:0] x t_{DTG}

where x is either R or F and $t_{DTG} = (2^{(DTPRSC[2:0])})$ x $(t_{HRTIM} / 8)$.

Table 85 gives the resolution and maximum absolute values depending on the prescaler value.

Table 85. Deadtime resolution and max absolute values

DTPRSC[2:0]	t_{DTG}	t_{DTx} max	f_{HRTIM}= 144MHz	
			t_{DTG} (ns)	$\|t_{DTx}\|$ max (µs)
000	t_{HRTIM} / 8	511 * t_{DTG}	0.87	0.44
001	t_{HRTIM} / 4		1.74	0.89
010	t_{HRTIM} / 2		3.47	1.77
011	t_{HRTIM}		6.94	3.54
100	2 * t_{HRTIM}		13.89	7.10
101	4 * t_{HRTIM}		27.78	14.19
110	8 * t_{HRTIM}		55.55	28.39
111	16 * t_{HRTIM}		111.10	56.77

Figure 260 to *Figure 263* present how the deadtime generator behaves for reference waveforms with pulsewidth below the deadtime values, for all deadtime configurations.

Figure 260. Complementary outputs for low pulse width (SDTRx = SDTFx = 0)

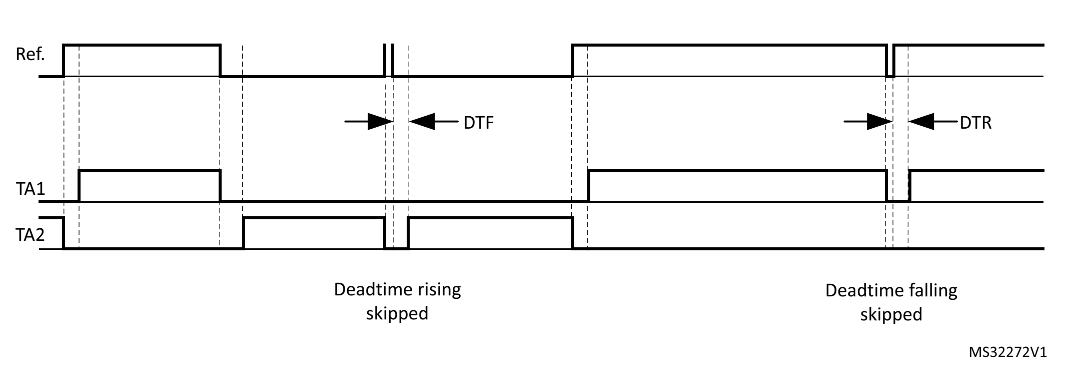


Figure 261. Complementary outputs for low pulse width (SDTRx = SDTFx = 1)

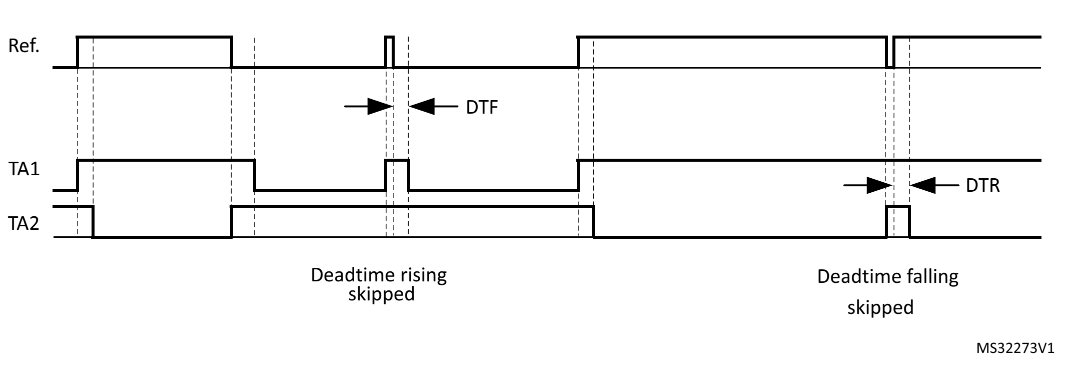


Figure 262. Complementary outputs for low pulse width (SDTRx = 0, SDTFx = 1)

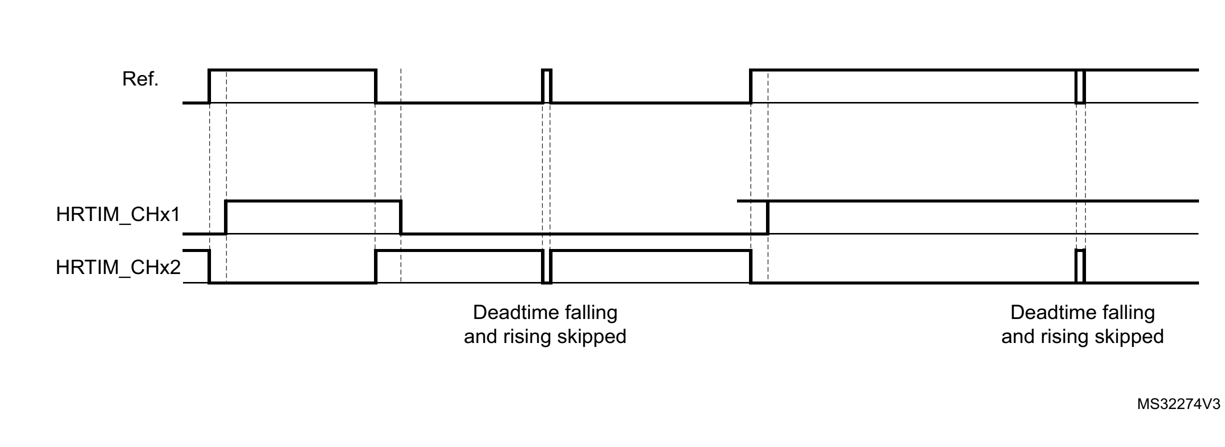
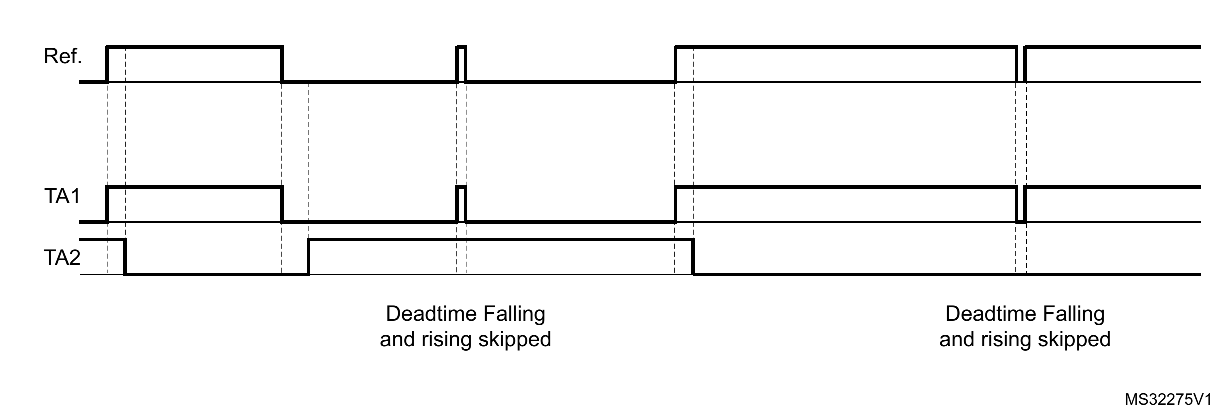

Ref.
HRTIM_CHx1
HRTIM_CHx2
Deadtime falling and rising skipped
Deadtime falling and rising skipped
MS32274V3

Figure 263. Complementary outputs for low pulse width (SDTRx = 1, SDTFx=0)


Ref.
TA1
TA2
Deadtime Falling and rising skipped
Deadtime Falling and rising skipped
MS32275V1

For safety purposes, it is possible to prevent any spurious write into the deadtime registers by locking the sign and/or the value of the deadtime using DTFLKx, DTRLKx, DTFSLKx and DTRSLKx. Once these bits are set, the related bits and bitfields are becoming read only until the next system reset.

Caution: DTEN bit must not be changed in the following cases:
- When the timer is enabled (TxEN bit set)
- When the timer outputs are set/reset by another timer (while TxEN is reset)
Otherwise, an unpredictable behavior would result.
It is therefore necessary to disable the timer (TxCEN bit reset) and have the corresponding outputs disabled.

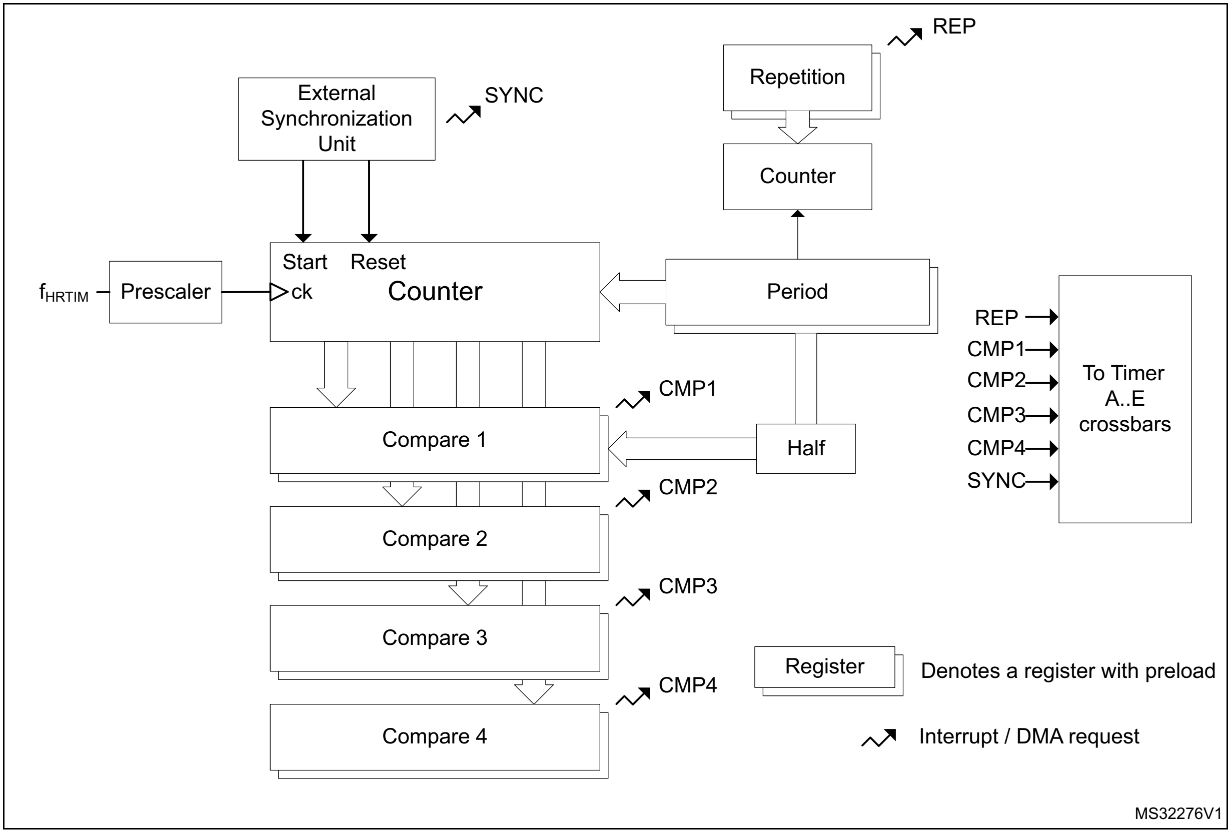
For the particular case where DTEN must be set while the burst mode is enabled with a deadtime upon entry (BME = 1, DIDL = 1, IDLEM = 1), it is necessary to force the two outputs in their IDLES state by software commands (SST, RST bits) before setting DTEN bit. This is to avoid any side effect resulting from a burst mode entry that would happen immediately before a deadtime enable.

21.3.5 Master timer

The main purpose of the master timer is to provide common signals to the 5 timing units, either for synchronization purpose or to set/reset outputs. It does not have direct control over any outputs, but still can be used indirectly by the set/reset crossbars.

Figure 264 provides an overview of the master timer.

Figure 264. Master timer overview



The master timer is based on the very same architecture as the timing units, with the following differences:

- It does not have outputs associated with, nor output related control
- It does not have its own crossbar unit, nor push-pull or deadtime mode
- It can only be reset by the external synchronization circuitry
- It does not have a capture unit, nor the auto-delayed mode
- It does not include external event blanking and windowing circuitry
- It has a limited set of interrupt / DMA requests: Compare 1..4, repetition, register update and external synchronization event.

The master timer control register includes all the timer enable bits, for the master and Timer A..E timing units. This allows to have all timer synchronously started with a single write access.

It also handles the external synchronization for the whole HRTIM timer (see *Section 21.3.17: Synchronizing the HRTIM with other timers or HRTIM instances*), with both MCU internal and external (inputs/outputs) resources.

Master timer control registers are mapped with the same offset as the timing units' registers.

21.3.6 Set/reset events priorities and narrow pulses management

This section describes how the output waveform is generated when several set and/or reset requests are occurring within 3 consecutive t_{HRTIM} periods.

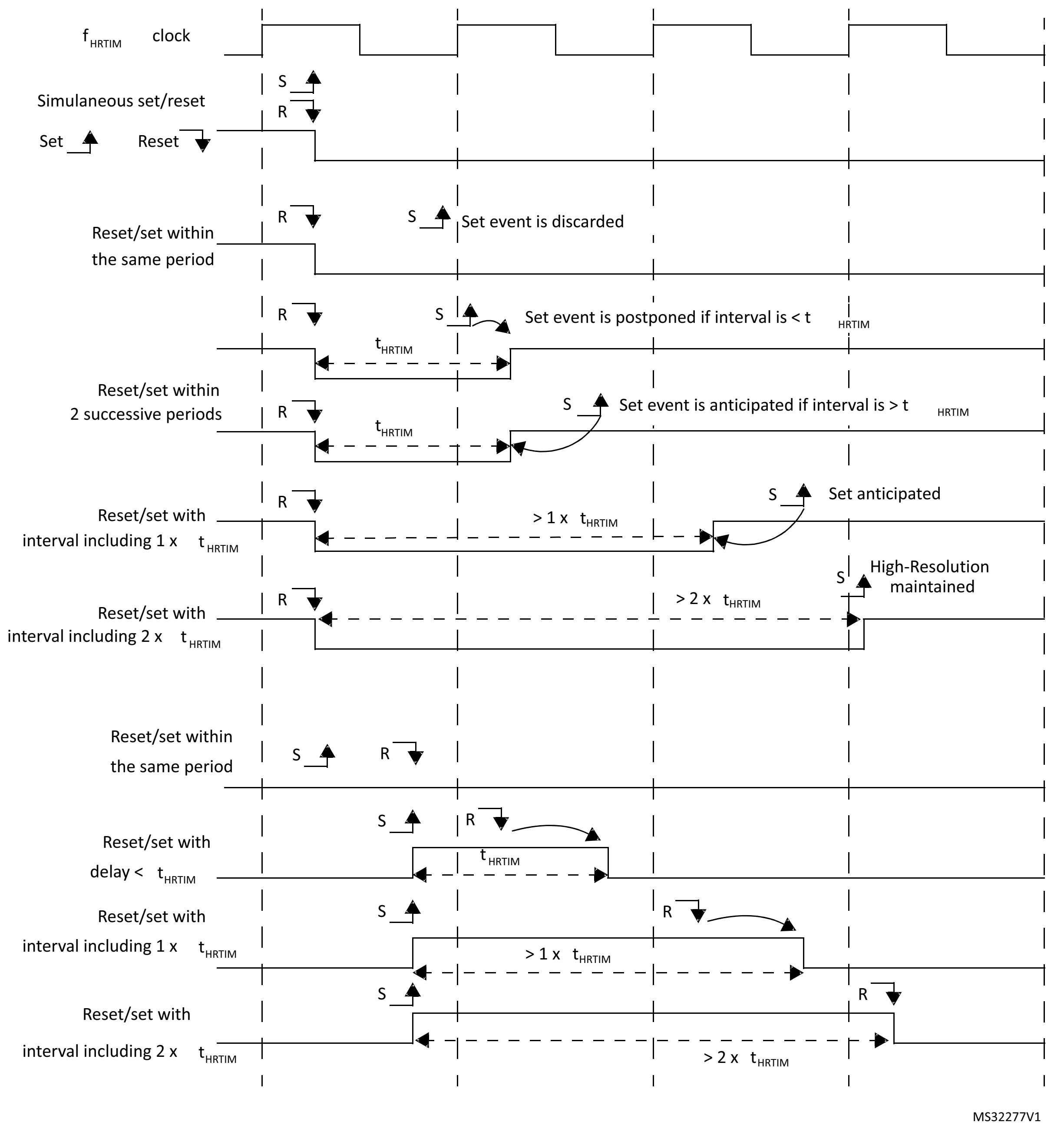
Case 1: clock prescaler CKPSC[2:0] < 5

An arbitration is performed during each t_{HRTIM} period, in 3 steps:

1. For each active event, the desired output transition is determined (set, reset or toggle).
2. A predefined arbitration is performed among the active events (from highest to lowest priority CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1 ⟶ PER, see *Concurrent set request / Concurrent reset requests*.
3. A high-resolution delay-based arbitration is performed with reset having the highest priority, among the low-resolution events and events having won the predefined arbitration.

When set and reset requests from two different sources are simultaneous, the reset action has the highest priority. If the interval between set and reset requests is below 2 f_{HRTIM} period, the behavior depends on the time interval and on the alignment with the f_{HRTIM} clock, as shown on *Figure 265*.

Figure 265. Short distance set/reset management for narrow pulse generation


f HRTIM clock
Simulaneous set/reset
Set
Reset
S
R
Reset/set within the same period
Set event is discarded
Set event is postponed if interval is < t HRTIM
t HRTIM
Reset/set within 2 successive periods
Set event is anticipated if interval is > t HRTIM
Reset/set with interval including 1 x t HRTIM
> 1 x t HRTIM
Set anticipated
Reset/set with interval including 2 x t HRTIM
> 2 x t HRTIM
High-Resolution maintained
Reset/set within the same period
Reset/set with delay < t HRTIM
Reset/set with interval including 1 x t HRTIM
Reset/set with interval including 2 x t HRTIM
MS32277V1

If the set and reset events are generated within the same t_{HRTIM} period, the reset event has the highest priority and the set event is ignored.

If the set and reset events are generated with an interval below t_{HRTIM} period, across 2 periods, a pulse of 1 t_{HRTIM} period is generated.

If the set and reset events are generated with an interval below 2 t_{HRTIM} periods, a pulse of 2 t_{HRTIM} periods is generated.

If the set and reset events are generated with an interval between 2 and 3 t_{HRTIM} periods, the high-resolution is available if the interval is over 2 complete t_{HRTIM} periods.

If the set and reset events are generated with an interval above 3 t_{HRTIM} periods, the high-resolution is always available.

Concurrent set request / Concurrent reset requests

When multiple sources are selected for a set event, an arbitration is performed when the set requests occur within the same f_{HRTIM} clock period.

In case of multiple requests from adjacent timers (TIMEVNT1..9), the request which occurs first is taken into account. The arbitration is done in 2 steps, depending on:

1. the source (CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1),
2. the delay.

If multiple requests from the master timer occur within the same f_{HRTIM} clock period, a predefined arbitration is applied and a single request will be taken into account, whatever the effective high-resolution setting (from the highest to the lowest priority):

MSTCMP4 → MSTCMP3 → MSTCMP2 → MSTCMP1 → MSTCMPER

Note: It is advised to avoid generating multiple set (reset) requests from the master timer to a given timer with an interval below 3x t_{HRTIM} *to maintain the high-resolution.*

When multiple requests internal to the timer occur within the same f_{HRTIM} clock period, a predefined arbitration is applied and the requests are taken with the following priority, whatever the effective timing (from highest to lowest):

CMP4 → CMP3 → CMP2 → CMP1 → PER

Note: Practically, this is of a primary importance only when using auto-delayed Compare 2 and Compare 4 simultaneously (i.e. when the effective set/reset cannot be determined a priori because it is related to an external event). In this case, the highest priority signal must be affected to the CMP4 event.

Last, the highest priority is given to low-resolution events: EXTEVNT1..10, RESYNC (coming from SYNC event if SYNCRSTx or SYNCSTRTx is set or from a software reset), update and software set (SST). The update event is considered as having the largest delay (0x1F if PSC = 0).

As a summary, in case of a close vicinity (events occurring within the same f_{HRTIM} clock period), the effective set (reset) event will be arbitrated between:

- Any TIMEVNT1..9 event
- A single source from the master (as per the fixed arbitration given above)
- A single source from the timer
- The “low-resolution events”.

The same arbitration principle applies for concurrent reset requests. In this case, the reset request has the highest priority.

Case 2: clock prescaler CKPSC[2:0] ≥ 5

The narrow pulse management is simplified when the high-resolution is not effective.

A set or reset event occurring within the prescaler clock cycle is delayed to the next active edge of the prescaled clock (as for a counter reset), even if the arbitration is still performed every t_{HRTIM} cycle.

If a reset event is followed by a set event within the same prescaler clock cycle, the latest event will be considered.

21.3.7 External events global conditioning

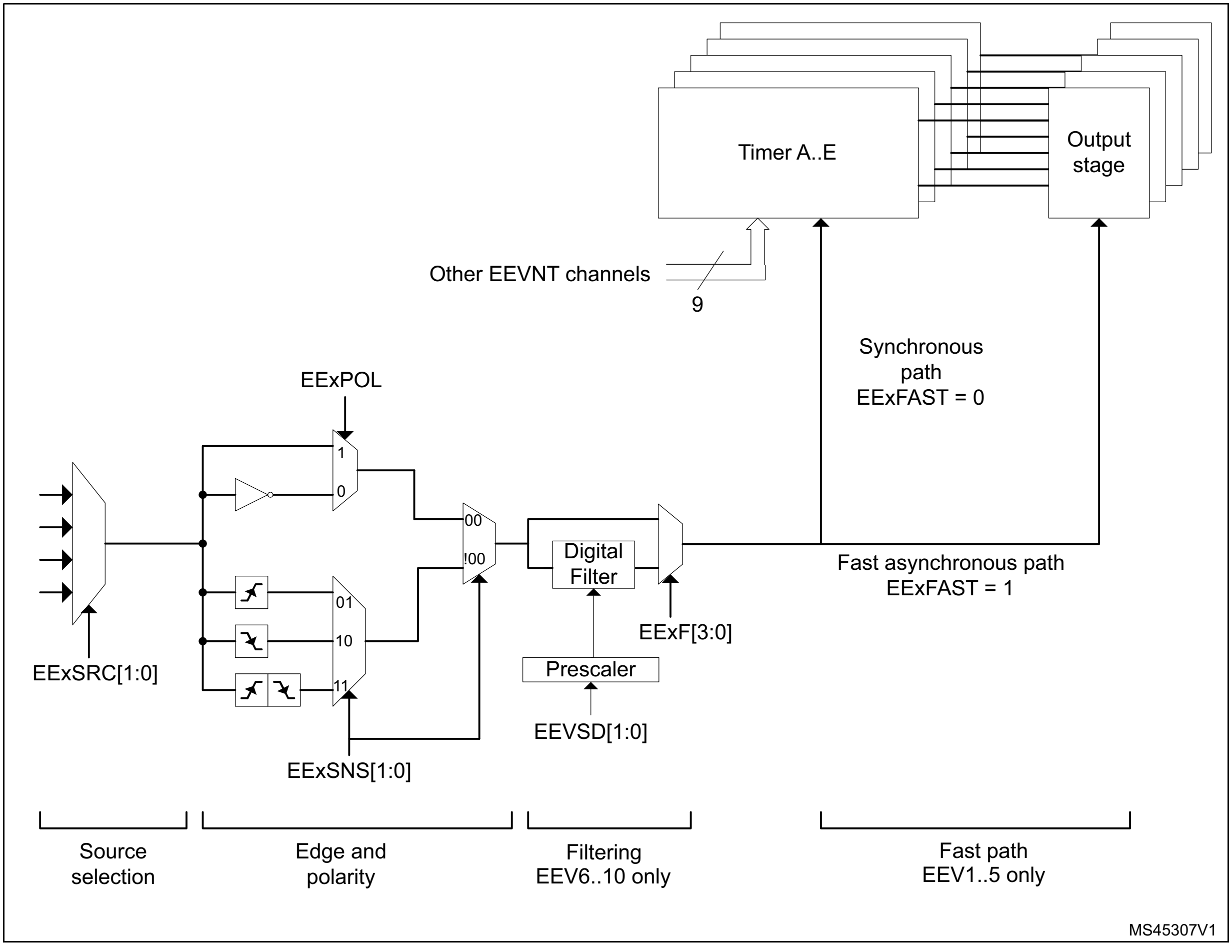
The HRTIM timer can handle events not generated within the timer, referred to as “external event”. These external events come from multiple sources, either on-chip or off-chip:

- built-in comparators,
- digital input pins (typically connected to off-chip comparators and zero-crossing detectors),
- on-chip events for other peripheral (ADC’s analog watchdogs and general purpose timer trigger outputs).

The external events conditioning circuitry allows to select the signal source for a given channel (with a 4:1 multiplexer) and to convert it into an information that can be processed by the crossbar unit (for instance, to have an output reset triggered by a falling edge detection on an external event channel).

Up to 10 external event channels can be conditioned and are available simultaneously for any of the 5 timers. This conditioning is common to all timers, since this is usually dictated by external components (such as a zero-crossing detector) and environmental conditions (typically the filter set-up will be related to the applications noise level and signature). *Figure 266* presents an overview of the conditioning logic for a single channel.

Figure 266. External event conditioning overview (1 channel represented)



The 10 external events are initialized using the HRTIM_EECR1 and HRTIM EECR2 registers:

- to select up to 4 sources with the EExSRC[1:0] bits,
- to select the sensitivity with EExSNS[1:0] bits, to be either level-sensitive or edge-sensitive (rising, falling or both),
- to select the polarity, in case of a level sensitivity, with EExPOL bit,
- to have a low latency mode, with EExFAST bits (see *Latency to external events*), for external events 1 to 5.

Note: The external events used as triggers for reset, capture, burst mode, ADC triggers and delayed protection are edge-sensitive even if EESNS bit is reset (level-sensitive selection): if POL = 0 the trigger is active on external event rising edge, while if POL = 1 the trigger is active on external event falling edge.

The external events are discarded as long as the counters are disabled (TxCEN bit reset) to prevent any output state change and counter reset, except if they are used as ADC triggers.

Additionally, it is possible to enable digital noise filters, for external events 6 to 10, using EExF[3:0] bits in the HRTIM_EECR3 register.

A digital filter is made of a counter in which a number N of valid samples is needed to validate a transition on the output. If the input value changes before the counter has reached the value N, the counter is reset and the transition is discarded (considered as a spurious event). If the counter reaches N, the transition is considered as valid and transmitted as a correct external event. Consequently, the digital filter adds a latency to the external events being filtered, depending on the sampling clock and on the filter length (number of valid samples expected).

The sampling clock is either the f_{HRTIM} clock or a specific prescaled clock f_{EEVS} derived from f_{HRTIM}, defined with EEVSD[1:0] bits in HRTIM_EECR3 register.

Table 86 summarizes the available sources and features associated with each of the 10 external events channels.

Table 86. External events mapping and associated features

External event channel	Fast mode	Digital filter	Balanced fault timer A,B,C	Balanced fault timer D,E	Src1	Src 2	Src3	Src4	Comparator and input sources available per package		
									32-pin	48-pin	64-pin
1	Yes	-	-	-	PC12	COMP2	TIM1_TRGO	ADC1_AWD1	Comp	Comp	Comp & Input
2	Yes	-	-	-	PC11	COMP4	TIM2_TRGO	ADC1_AWD2	Comp	Comp	Comp & Input
3	Yes	-	-	-	PB7	COMP6	TIM3_TRGO	ADC1_AWD3	Input	Comp & Input	Comp & Input
4	Yes	-	-	-	PB6	OPAMP2[1]	-	ADC2_AWD1	OPAMP & Input	OPAMP & Input	OPAMP & Input
5	Yes	-	-	-	PB9	-	-	ADC2_AWD2	-	Input	Input
6	-	Yes	Yes	-	PB5	COMP2	TIM6_TRGO	ADC2_AWD3	Comp & Input	Comp & Input	Comp & Input

Table 86. External events mapping and associated features (continued)

External event channel	Fast mode	Digital filter	Balanc-ed fault timer A,B,C	Balanc-ed fault timer D,E	Src1	Src 2	Src3	Src4	Comparator and input sources available per package		
									32-pin	48-pin	64-pin
7	-	Yes	Yes	-	PB4	COMP4	TIM7_TRGO	-	Comp & Input	Comp & Input	Comp & Input
8	-	Yes	-	Yes	PB8	COMP6	-	-	-	Comp & Input	Comp & Input
9	-	Yes	-	Yes	PB3	OPAMP2 (1)	TIM15_TRGO	-	OPAMP & Input	OPAMP & Input	OPAMP & Input
10	-	Yes	-	-	PC6	-	-	-	-	-	Input

1. OPAMP2_OUT can be used as High-resolution timer internal event source. In this case, the software must set OPAMP2_DIG as of PA6 alternate function (AF13) to redirect OPAMP2_VOUT signal to the HRTIM external events through the Schmitt trigger.

Latency to external events

The external event conditioning gives the possibility to adjust the external event processing time (and associated latency) depending on performance expectations:

- A regular operating mode, in which the external event is resampled with the clock before acting on the output crossbar. This adds some latency but gives access to all crossbar functionalities. It enables the generation of an externally triggered high-resolution pulse.
- A fast operating mode, in which the latency between the external event and the action on the output is minimized. This mode is convenient for ultra-fast over-current protections, for instance.

EExFAST bits in the HRTIM_EECR1 register allow to define the operating for channels 1 to 5. This influences the latency and the jitter present on the output pulses, as summarized in the table below.

Table 87. Output set/reset latency and jitter vs external event operating mode

EExFAST	Response time latency	Response time jitter	Jitter on output pulse (counter reset by ext. event)
0	5 to 6 cycles of f_{HRTIM} clock	1 cycles of f_{HRTIM} clock	No jitter, pulse width maintained with high-resolution
1	Minimal latency (depends whether the comparator or digital input is used)	Minimal jitter	1 cycle of f_{HRTIM} clock jitter pulse width resolution down to t_{HRTIM}

The EExFAST mode is only available with level-sensitive programming (EExSNS[1:0] = 00); the edge-sensitivity cannot be programmed.

It is possible to apply event filtering to external events (both blanking and windowing with EExFLTR[3:0] != 0000, see *Section 21.3.8*). In this case, EExLTCHx bit must be reset: the postponed mode is not supported, neither the windowing timeout feature.

Note: The external event configuration (source and polarity) must not be modified once the related EExFAST bit is set.

A fast external event cannot be used to toggle an output: if must be enabled either in HRTIM_SETxyR or HRTIM_RSTxyR registers, not in both.

When a set and a reset event - from 2 independent fast external events - occur simultaneously, the reset has the highest priority in the crossbar and the output becomes inactive.

When EExFAST bit is set, the output cannot be changed during the 11 f_{HRTIM} clock periods following the external event.

Figure 267 and *Figure 268* give practical examples of the reaction time to external events, for output set/reset and counter reset.

Figure 267. Latency to external events falling edge (counter reset and output set)

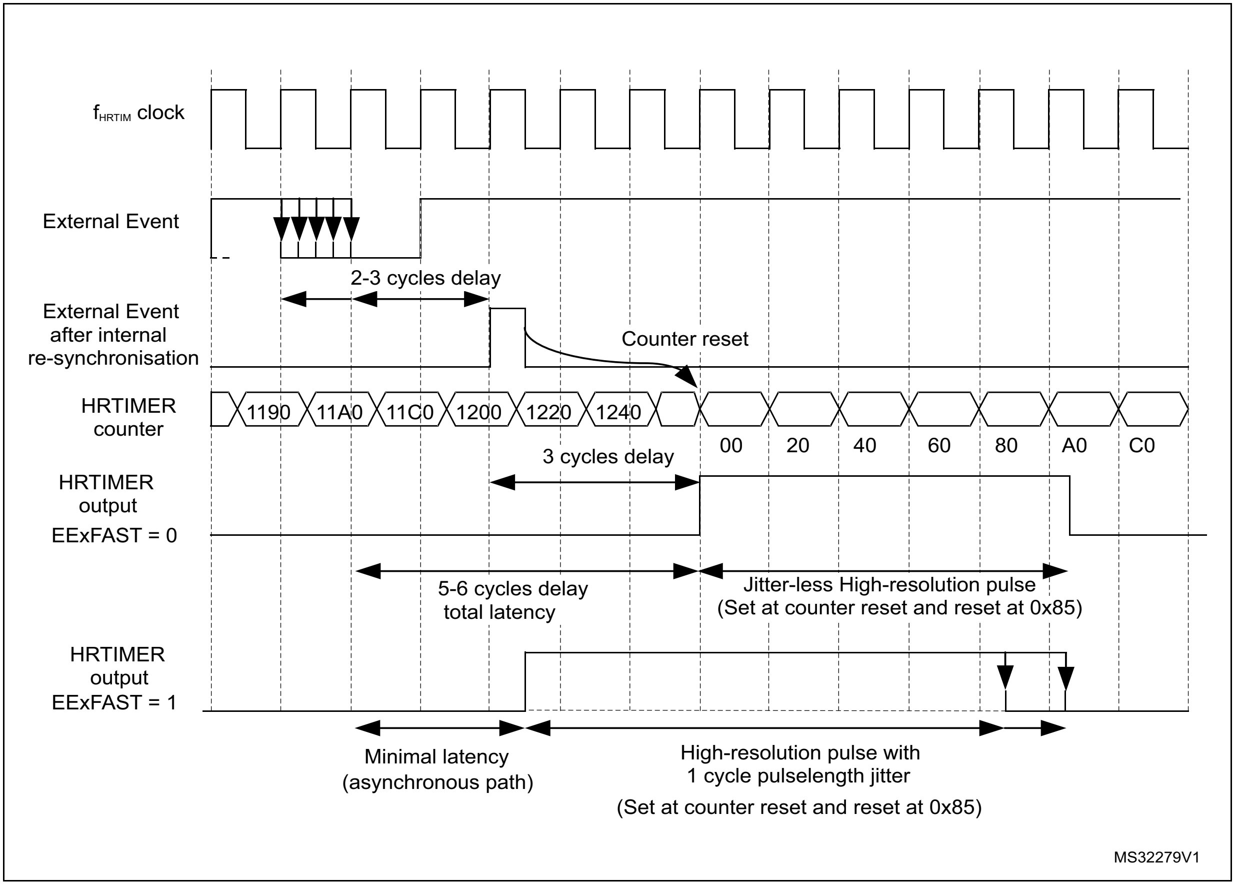
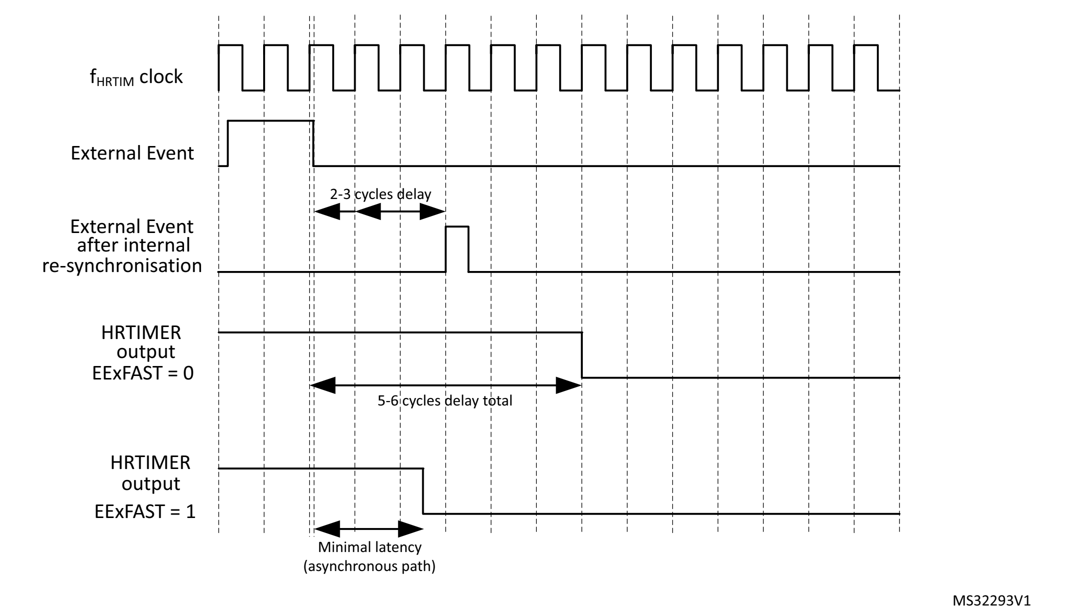

f_HRTIM clock
External Event
2-3 cycles delay
External Event after internal re-synchronisation
Counter reset
HRTIMER counter
1190
11A0
11C0
1200
1220
1240
00
20
40
60
80
A0
C0
3 cycles delay
HRTIMER output EExFAST = 0
5-6 cycles delay total latency
Jitter-less High-resolution pulse (Set at counter reset and reset at 0x85)
HRTIMER output EExFAST = 1
Minimal latency (asynchronous path)
High-resolution pulse with 1 cycle pulselength jitter
(Set at counter reset and reset at 0x85)
MS32279V1

Figure 268. Latency to external events (output reset on external event)


f_{HRTIM} clock
External Event
2-3 cycles delay
External Event after internal re-synchronisation
HRTIMER output EExFAST = 0
5-6 cycles delay total
HRTIMER output EExFAST = 1
Minimal latency (asynchronous path)
MS32293V1

21.3.8 External event filtering in timing units

Once conditioned, the 10 external events are available for all timing units.

They can be used directly and are active as soon as the timing unit counter is enabled (TxCEN bit set).

They can also be filtered to have an action limited in time, usually related to the counting period. Two operations can be performed:

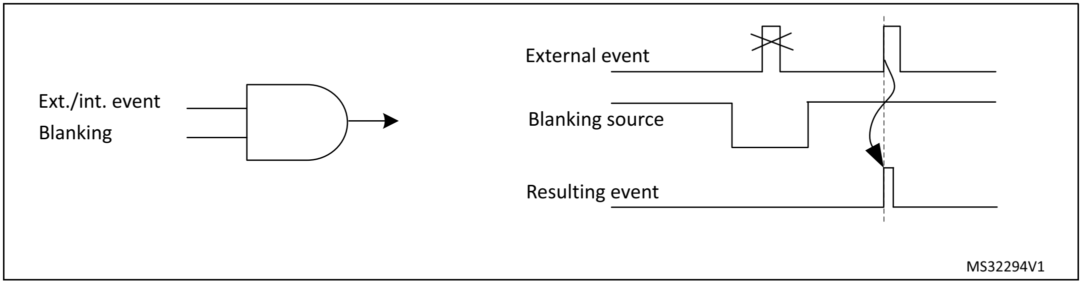
- blanking, to mask external events during a defined time period,
- windowing, to enable external events only during a defined time period.

These modes are enabled using HRTIM_EExFLTR[3:0] bits in the HRTIM_EEFxR1 and HRTIM_EEFxR2 registers. Each of the 5 TimerA..E timing units has its own programmable filter settings for the 10 external events.

Blanking mode

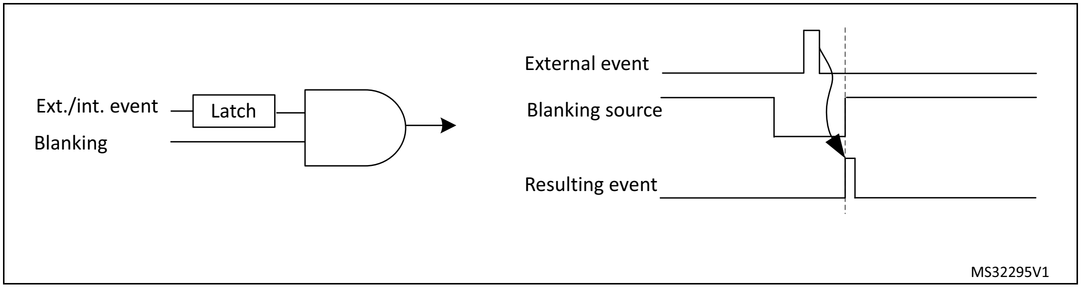
In event blanking mode (see *Figure 269*), the external event is ignored if it happens during a given blanking period. This is convenient, for instance, to avoid a current limit to trip on switching noise at the beginning of a PWM period. This mode is active for EExFLTR[3:0] bitfield values ranging from 0001 to 1100.

Figure 269. Event blanking mode



In event postpone mode, the external event is not taken into account immediately but is memorized (latched) and generated as soon as the blanking period is completed, as shown on *Figure 270*. This mode is enabled by setting EExLTCH bit in HRTIM_EEFxR1 and HRTIM_EEFxR2 registers.

Figure 270. Event postpone mode



The blanking signal comes from several sources:

- the timer itself: the blanking lasts from the counter reset to the compare match (EExFLTR[3:0] = 0001 to 0100 for Compare 1 to Compare 4)
- from other timing units (EExFLTR[3:0] = 0101 to 1100): the blanking lasts from the selected timing unit counter reset to one of its compare match, or can be fully programmed as a waveform on Tx2 output. In this case, events are masked as long as the Tx2 signal is inactive (it is not necessary to have the output enabled, the signal is taken prior to the output stage).

The EEXFLTR[3:0] configurations from 0101 to 1100 are referred to as TIMFLTR1 to TIMFLTR8 in the bit description, and differ from one timing unit to the other. *Table 88* gives the 8 available options per timer: CMPx refers to blanking from counter reset to compare match, Tx2 refers to the timing unit TIMx output 2 waveform defined with HRTIM_SETx2 and HRTIM_RSTx2 registers. For instance, Timer B (TIMFLTR6) is Timer C output 2 waveform.

Table 88. Filtering signals mapping per time

	Source	Timer A				Timer B				Timer C				Timer D				Timer E			
		CMP 1	CMP 2	CMP 4	TA2	CMP 1	CMP 2	CMP 4	TB2	CMP 1	CMP 2	CMP 4	TC2	CMP 1	CMP 2	CMP 4	TD2	CMP 1	CMP 2	CMP 4	TE2
Destination	Timer A	-	-	-	-	1	-	2	3	4	-	5	6	7	-	-	-	-	8	-	-
	Timer B	1	-	2	3	-	-	-	-	4	5	-	6	-	7	-	-	8	-	-	-
	Timer C	-	1	-	-	2	-	3	4	-	-	-	-	5	-	6	7	-	-	8	-
	Timer D	1	-	-	-	-	2	-	-	3	4	-	5	-	-	-	-	6	-	7	8
	Timer E	-	1	-	-	2	-	-	-	3	-	4	5	6	-	7	8	-	-	-	-

Figure 271 and *Figure 272* give an example of external event blanking for all edge and level sensitivities, in regular and postponed modes.

Figure 271. External trigger blanking with edge-sensitive trigger

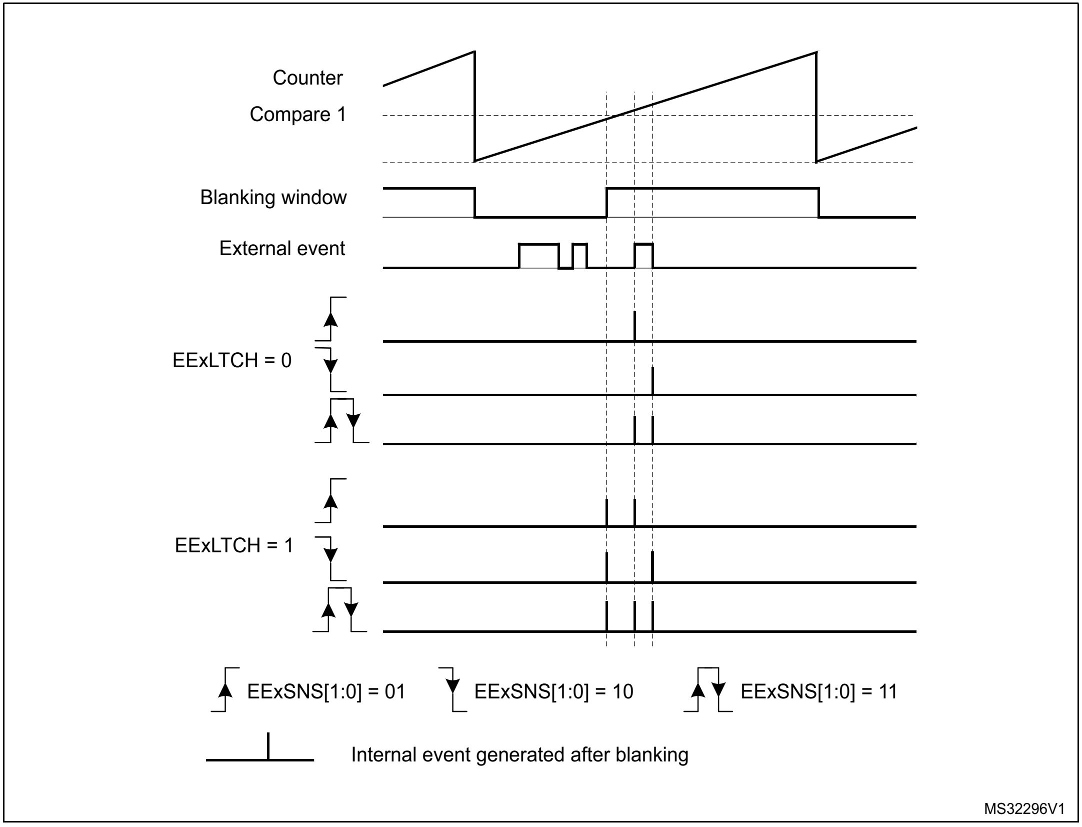
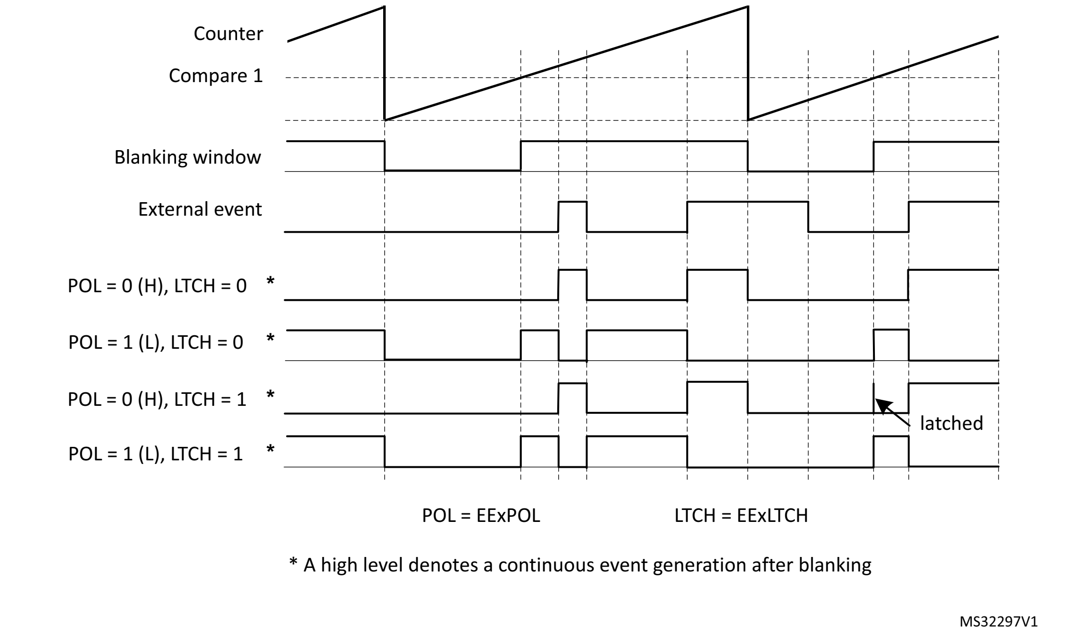


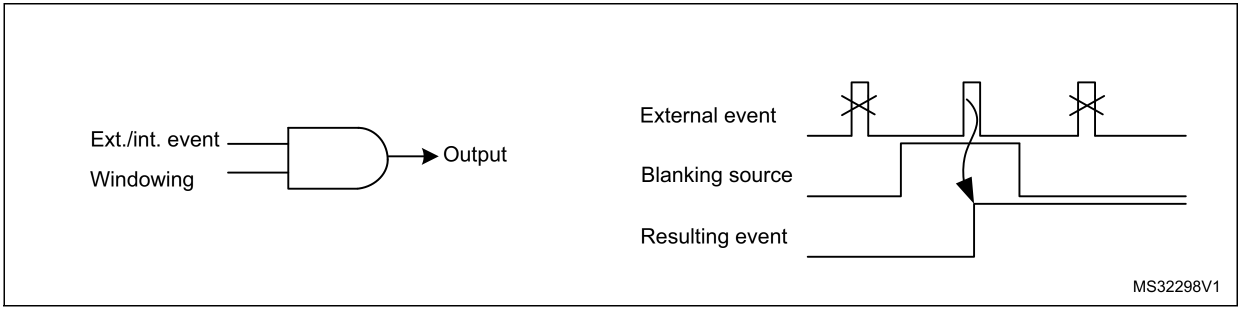
Figure 272. External trigger blanking, level sensitive triggering



Windowing mode

In event windowing mode, the event is taken into account only if it occurs within a given time window, otherwise it is ignored. This mode is active for EExFLTR[3:0] ranging from 1101 to 1111.

Figure 273. Event windowing mode



EExLTCH bit in EEFxR1 and EEFxR2 registers allows to latch the signal, if set to 1: in this case, an event is accepted if it occurs during the window but is delayed at the end of it.

- If EExLTCH bit is reset and the signal occurs during the window, it is passed through directly.
- If EExLTCH bit is reset and no signal occurs, a timeout event is generated at the end of the window.

A use case of the windowing mode is to filter synchronization signals. The timeout generation allows to force a default synchronization event, when the expected synchronization event is lacking (for instance during a converter start-up).

There are 3 sources for each external event windowing, coded as follows:

- 1101 and 1110: the windowing lasts from the counter reset to the compare match (respectively Compare 2 and Compare 3)
- 1111: the windowing is related to another timing unit and lasts from its counter reset to its Compare 2 match. The source is described as TIMWIN in the bit description and is given in *Table 89*. As an example, the external events in timer B can be filtered by a window starting from timer A counter reset to timer A Compare 2.

Table 89. Windowing signals mapping per timer (EEFLTR[3:0] = 1111)

Destination	Timer A	Timer B	Timer C	Timer D	Timer E
TIMWIN (source)	Timer B CMP2	Timer A CMP2	Timer D CMP2	Timer C CMP2	Timer D CMP2

Note: The timeout event generation is not supported if the external event is programmed in fast mode.

Figure 274 and *Figure 275* present how the events are generated for the various edge and level sensitivities, as well as depending on EExLTCH bit setting. Timeout events are specifically mentioned for clarity reasons.

Figure 274. External trigger windowing with edge-sensitive trigger

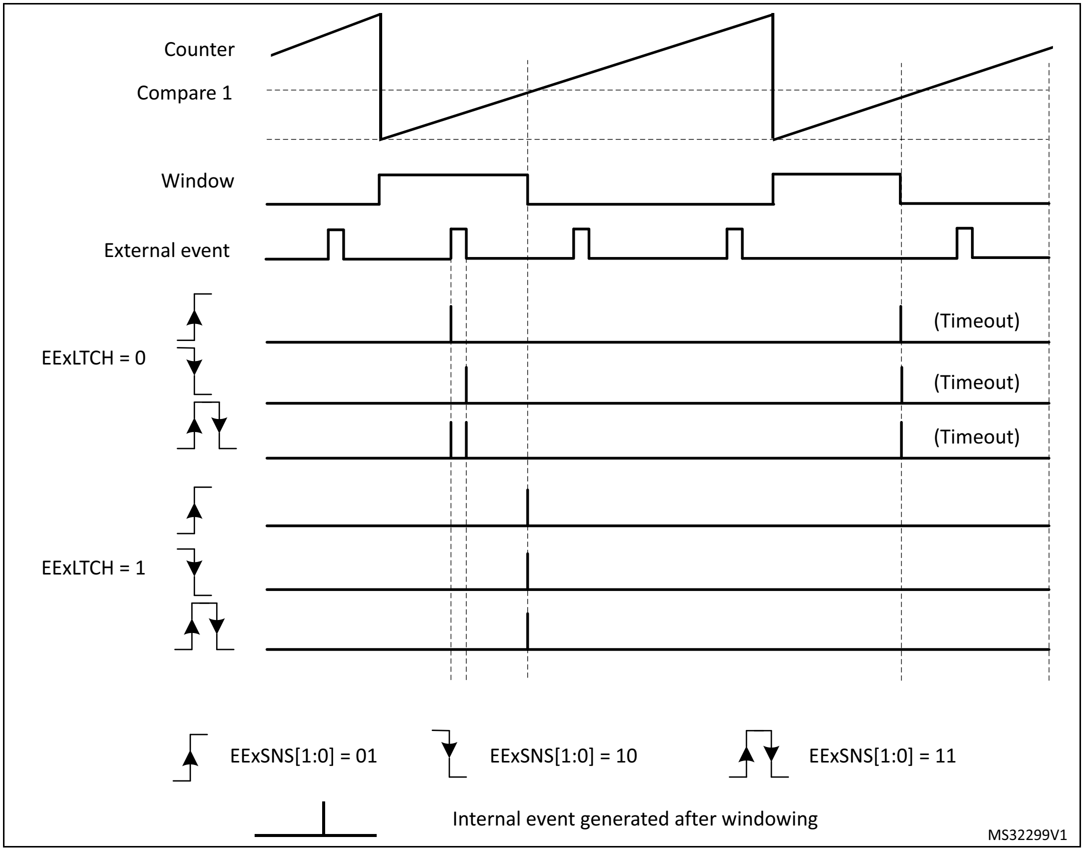
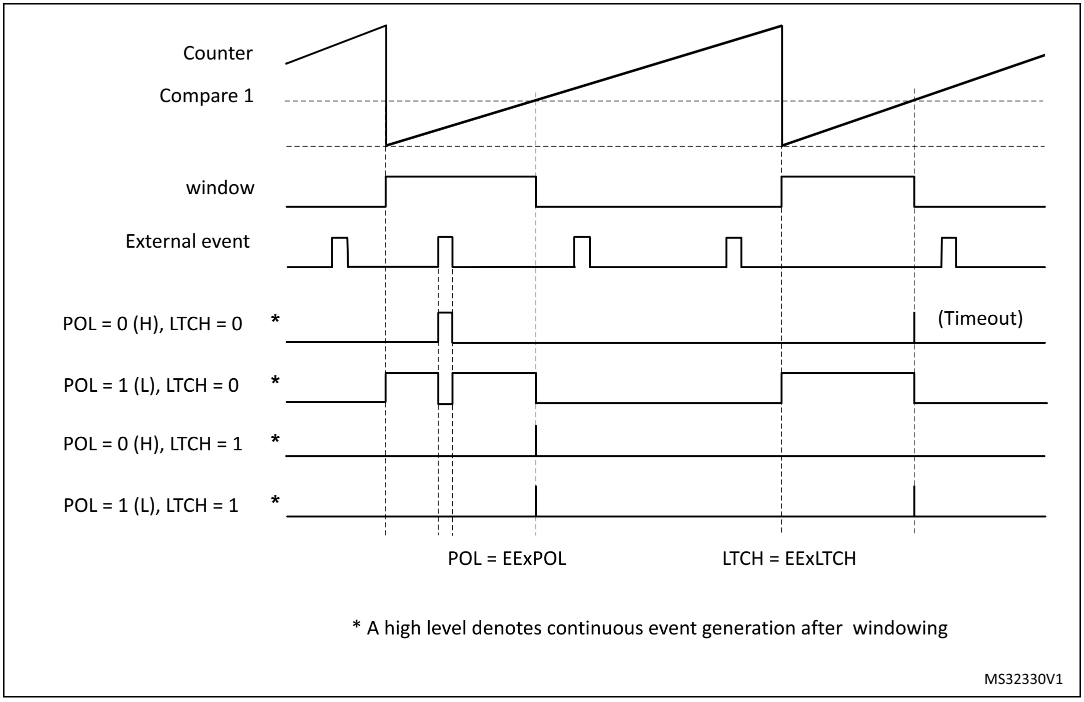


Figure 275. External trigger windowing, level sensitive triggering



21.3.9 Delayed Protection

The HRTIM features specific protection schemes, typically for resonant converters when it is necessary to shut down the PWM outputs in a delayed manner, either once the active pulse is completed or once a push-pull period is completed. These features are enabled with DLYPRTEN bit in the HRTIM_OUTxR register, and are using specific external event channels.

Delayed Idle

In this mode, the active pulse is completed before the protection is activated. The selected external event causes the output to enter in idle mode at the end of the active pulse (defined by an output reset event in HRTIM_RSTx1R or HRTIM_RSTx2R).

Once the protection is triggered, the idle mode is permanently maintained but the counter continues to run, until the output is re-enabled. Tx1OEN and Tx2OEN bits are not affected by the delayed idle entry. To exit from delayed idle and resume operation, it is necessary to overwrite Tx1OEN and Tx2OEN bits to 1. The output state will change on the first transition to an active state following the output enable command.

Note: The delayed idle mode cannot be exited immediately after having been entered, before the active pulse is completed: it is mandatory to make sure that the outputs are in idle state before resuming the run mode. This can be done by waiting up to the next period, for instance, or by polling the O1CPY and/or O2CPY status bits in the TIMxISR register.

The delayed idle mode can be applied to a single output (DLYPRT[2:0] = x00 or x01) or to both outputs (DLYPRT[2:0] = x10).

An interrupt or a DMA request can be generated in response to a Delayed Idle mode entry. The DLYPRT flag in HRTIM_TIMxISR is set as soon as the external event arrives, independently from the end of the active pulse on output.

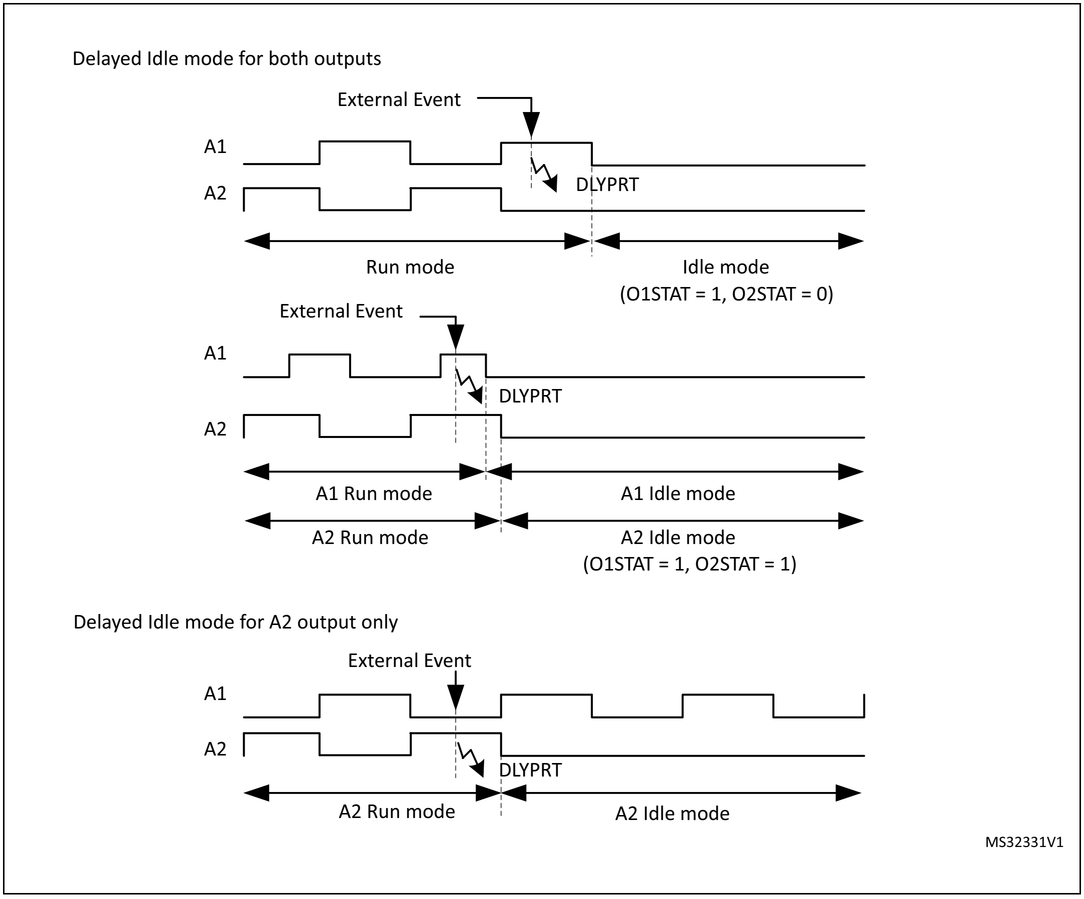
When the Delayed Idle mode is triggered, the output states can be determined using O1STAT and O2STAT in HRTIM_TIMxISR. Both status bits are updated even if the delayed idle is applied to a single output. When the push-pull mode is enabled, the IPPSTAT flag in HRTIM_TIMxISR indicates during which period the delayed protection request occurred.

This mode is available whatever the timer operating mode (regular, push-pull, deadtime). It is available with 2 external events only:

- EEV6 and EEV7 for Timer A, B and C
- EEV8 and EEV9 for Timer D and E

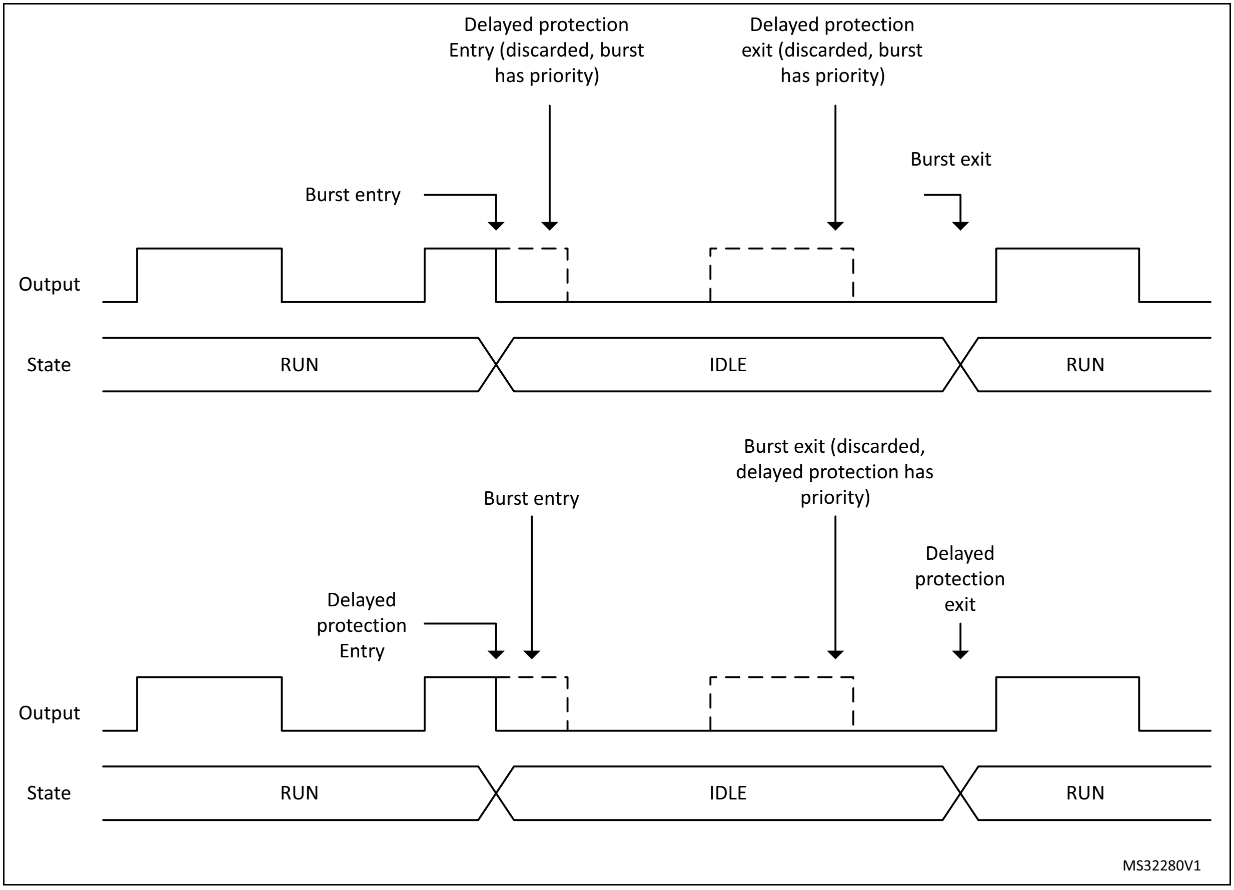
The delayed protection mode can be triggered only when the counter is enabled (TxCEN bit set). It remains active even if the TxEN bit is reset, until the TxyOEN bits are set.

Figure 276. Delayed Idle mode entry



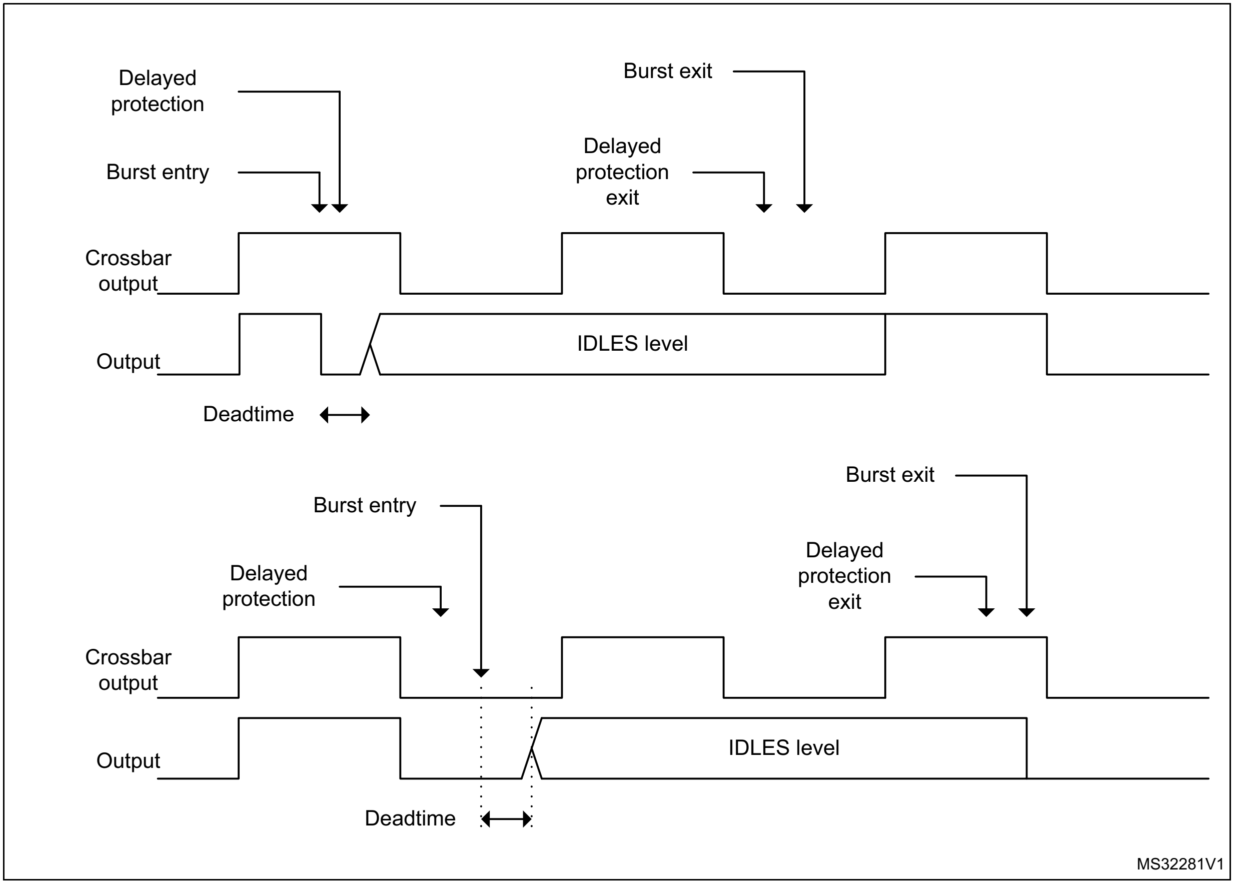
The delayed idle mode has a higher priority than the burst mode: any burst mode exit request is discarded once the delayed idle protection has been triggered. On the contrary, If the delayed protection is exited while the burst mode is active, the burst mode will be resumed normally and the output will be maintained in the idle state until the burst mode exits. *Figure 277* gives an overview of these different scenarios.

Figure 277. Burst mode and delayed protection priorities (DIDL = 0)


Delayed protection Entry (discarded, burst has priority)
Delayed protection exit (discarded, burst has priority)
Burst exit
Burst entry
Output
State
RUN
IDLE
RUN
Burst exit (discarded, delayed protection has priority)
Burst entry
Delayed protection exit
Delayed protection Entry
Output
State
RUN
IDLE
RUN
MS32280V1

The same priorities are applied when the delayed burst mode entry is enabled (DIDL bit set), as shown on *Figure 278* below.

Figure 278. Burst mode and delayed protection priorities (DIDL = 1)



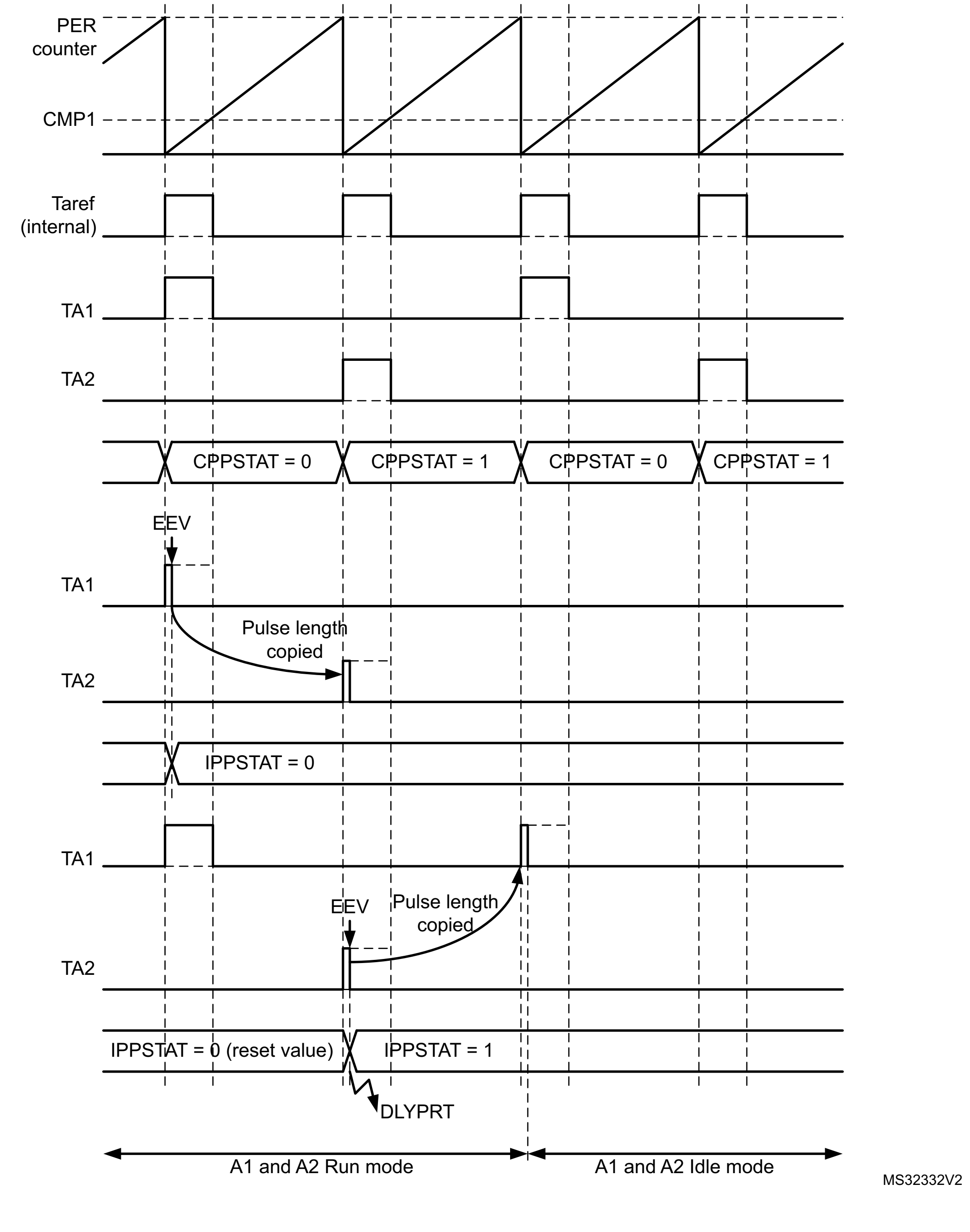
Balanced Idle

Only available in push-pull mode, it allows to have balanced pulsewidth on the two outputs when one of the active pulse is shortened due to a protection. The pulsewidth, which was terminated earlier than programmed, is copied on the alternate output and the two outputs are then put in idle state, until the normal operation is resumed by software. This mode is enabled by writing x11 in DLYPRT[2:0] bitfield in HRTIM_OUTxR.

This mode is available with 2 external events only:

- EEV6 and EEV7 for Timer A, B and C
- EEV8 and EEV9 for Timer D and E

Figure 279. Balanced Idle protection example



When the balanced Idle mode is enabled, the selected external event triggers a capture of the counter value into the Compare 4 active register (this value is not user-accessible). The push-pull is maintained for one additional period so that the shorten pulse can be repeated: a new output reset event is generated while the regular output set event is maintained.

The Idle mode is then entered and the output takes the level defined by IDLESx bits in the HRTIM_OUTxR register. The balanced idle mode entry is indicated by the DLYPRT flag, while the IPPSTAT flag indicates during which period the external event occurred, to determine the sequence of shorten pulses (HRTIM_CHA1 then HRTIM_CHA2 or vice versa).

The timer operation is not interrupted (the counter continues to run).

To enable the balanced idle mode, it is necessary to have the following initialization:

- timer operating in continuous mode (CONT = 1)
- Push-pull mode enabled
- HRTIM_CMP4xR must be set to 0 and the content transferred into the active register (for instance by forcing a software update)
- DELCMP4[1:0] bit field must be set to 00 (auto-delayed mode disabled)
- DLYPRT[2:0] = x11 (delayed protection enable)

Note: The HRTIM_CMP4xR register must not be written during a balanced idle operation. The CMP4 event is reserved and cannot be used for another purpose.

In balanced idle mode, it is recommended to avoid multiple external events or software-based reset events causing an output reset. If such an event arrives before a balanced idle request within the same period, it will cause the output pulses to be unbalanced (1st pulse length defined by the external event or software reset, while the 2nd pulse is defined by the balanced idle mode entry).

The minimum pulsewidth that can be handled in balanced idle mode is 4 f_{HRTIM} clock periods (0x80 when CKPSC[2:0] = 0, 0x40 if CKPSC[2:0] = 1, 0x20 if CKPSC[2:0] = 2,...).

If the capture occurs before the counter has reached this minimum value, the current pulse is extended up to 4 f_{HRTIM} clock periods before being copied into the secondary output. In any case, the pulsewidths are always balanced.

Tx1OEN and Tx2OEN bits are not affected by the balanced idle entry. To exit from balanced idle and resume the operation, it is necessary to overwrite Tx1OEN and Tx2OEN bits to 1 simultaneously. The output state will change on the first active transition following the output enable.

It is possible to resume operation similarly to the delayed idle entry. For instance, if the external event arrives while output 1 is active (delayed idle effective after output 2 pulse), the re-start sequence can be initiated for output 1 first. To do so, it is necessary to poll CPPSTAT bit in the HRTIM_TIMxISR register. Using the above example (IPPSTAT flag equal to 0), the operation will be resumed when CPPSTAT bit is 0.

In order to have a specific re-start sequence, it is possible to poll the CPPSTAT to know which output will be active first. This allows, for instance, to re-start with the same sequence as the idle entry sequence: if EEV arrives during output 1 active, the re-start sequence will be initiated when the output 1 is active (CPPSTAT = 0).

Note: The balanced idle mode must not be disabled while a pulse balancing sequence is on-going. It is necessary to wait until the CMP4 flag is set, thus indicating that the sequence is completed, to reset the DLYPRTEN bit.

The balanced idle protection mode can be triggered only when the counter is enabled (TxCEN bit set). It remains active even if the TxCEN bit is reset, until TxyOEN bits are set.

Balanced idle can be used together with the burst mode under the following conditions:

- TxBM bit must be reset (counter clock maintained during the burst, see *Section 21.3.13*),
- No balanced idle protection must be triggered while the outputs are in a burst idle state.

The balanced idle mode has a higher priority than the burst mode: any burst mode exit request is discarded once the balanced idle protection has been triggered. On the contrary, if the delayed protection is exited while the burst mode is active, the burst mode will be resumed normally.

Note: Although the output state is frozen in idle mode, a number of events are still generated on the auxiliary outputs (see Section 21.3.16) during the idle period following the delayed protection:
- Output set/reset interrupt or DMA requests
- External event filtering based on output signal
- Capture events triggered by set/reset

21.3.10 Register preload and update management

Most of HRTIM registers are buffered and can be preloaded if needed. Typically, this allows to prevent the waveforms from being altered by a register update not synchronized with the active events (set/reset).

When the preload mode is enabled, accessed registers are shadow registers. Their content is transferred into the active register after an update request, either software or synchronized with an event.

By default, PREEN bits in HRTIM_MCR and HRTIM_TIMxCR registers are reset and the registers are not preloaded: any write directly updates the active registers. If PREEN bit is reset while the timer is running and preload was enabled, the content of the preload registers is directly transferred into the active registers.

Each timing unit and the master timer have their own PREEN bit. If PRREN is set, the preload registers are enabled and transferred to the active register only upon an update event.

There are two options to initialize the timer when the preload feature is needed:

- Enable PREEN bit at the very end of the timer initialization to have the preload registers transferred into the active registers before the timer is enabled (by setting MCEN and TxCEN bits).
- enable PREEN bit at any time during the initialization and force a software update immediately before starting.

Table 90 lists the registers which can be preloaded, together with a summary of available update events.

Table 90. HRTIM preloadable control registers and associated update sources

Timer	Preloadable registers	Preload enable	Update sources
Master Timer	HRTIM_DIER HRTIM_MPER HRTIM_MREP HRTIM_MCMP1R HRTIM_MCMP2R HRTIM_MCMP3R HRTIM_MCMP4R	PREEN bit in HRTIM_MCR	Software Repetition event Burst DMA event Repetition event following a burst DMA event
Timer x x = A..E	HRTIM_TIMxDIER HRTIM_TIMxPER HRTIM_TIMxREP HRTIM_TIMxCMP1R HRTIM_TIMxCMP1CR HRTIM_TIMxCMP2R HRTIM_TIMxCMP3R HRTIM_TIMxCMP4R HRTIM_DTxR HRTIM_SETx1R HRTIM_RSTx1R HRTIM_SETx2R HRTIM_RSTx2R HRTIM_RSTxR	PREEN bit in HRTIM_TIMxCR	Software TIMx Repetition event TIMx Reset Event Burst DMA event Update event from other timers (TIMy, Master) Update event following a burst DMA event Update enable input 1..3 Update event following an update enable input 1..3
HRTIM Common	HRTIM_ADC1R HRTIM_ADC2R HRTIM_ADC3R HRTIM_ADC4R	TIMx or Master timer Update, depending on ADxUSRC[2:0] bits in HRTIM_CR1, if PREEN = 1 in the selected timer	

The master timer has 4 update options:

1. Software: writing 1 into MSWU bit in HRTIM_CR2 forces an immediate update of the registers. In this case, any pending hardware update request is cancelled.
2. Update done when the master counter rolls over and the master repetition counter is equal to 0. This is enabled when MREPU bit is set in HRTIM_MCR.
3. Update done once Burst DMA is completed (see Section 21.3.21 for details). This is enabled when BRSTDMA[1:0] = 01 in HRTIM_MCR. It is possible to have both MREPU=1 and BRSTDMA=01.
   Note: The update can take place immediately after the end of the burst sequence if SWU bit is set (i.e. forced update mode). If SWU bit is reset, the update will be done on the next update event following the end of the burst sequence.
4. Update done when the master counter rolls over following a Burst DMA completion. This is enabled when BRSTDMA[1:0] = 10 in HRTIM_MCR.

An interrupt or a DMA request can be generated by the master update event.

Each timer (TIMA..E) can also have the update done as follows:

- By software: writing 1 into TxSWU bit in HRTIM_CR2 forces an immediate update of the registers. In this case, any pending hardware update request is canceled.
- Update done when the counter rolls over and the repetition counter is equal to 0. This is enabled when TxREPU bit is set in HRTIM_TIMxCR.
- Update done when the counter is reset or rolls over in continuous mode. This is enabled when TxRSTU bit is set in HRTIM_TIMxCR. This is used for a timer operating in single-shot mode, for instance.
- Update done once a Burst DMA is completed. This is enabled when UPDGAT[3:0] = 0001 in HRTIM_TIMxCR.
- Update done on the update event following a Burst DMA completion (the event can be enabled with TxREPU, MSTU or TxU). This is enabled when UPDGAT[3:0] = 0010 in HRTIM_TIMxCR.
- Update done when receiving a request on the update enable input 1..3. This is enabled when UPDGAT[3:0] = 0011, 0100, 0101 in HRTIM_TIMxCR.
- Update done on the update event following a request on the update enable input 1..3 (the event can be enabled with TxREPU, MSTU or TxU). This is enabled when UPDGAT[3:0] = 0110, 0111, 1000 in HRTIM_TIMxCR
- Update done synchronously with any other timer or master update (for instance TIMA can be updated simultaneously with TIMB). This is used for converters requiring several timers, and is enabled by setting bits MSTU and TxU in HRTIM_TIMxCR register.

The update enable inputs 1..3 allow to have an update event synchronized with on-chip events coming from the general-purpose timers. These inputs are rising-edge sensitive.

Table 91 lists the connections between update enable inputs and the on-chip sources.

Table 91. Update enable inputs and sources

Update enable input	Update source
Update enable input 1	TIM16_OC
Update enable input 2	TIM17_OC
Update enable input 3	TIM6_TRGO

This allows to synchronize low frequency update requests with high-frequency signals (for instance an update on the counter roll-over of a 100 kHz PWM that has to be done at a 100 Hz rate).

Note: The update events are synchronized to the prescaler clock when CKPSC[2:0] > 5.

An interrupt or a DMA request can be generated by the Timx update event.

MUDIS and TxUDIS bits in the HRTIM_CR1 register allow to temporarily disable the transfer from preload to active registers, whatever the selected update event. This allows to modify several registers in multiple timers. The regular update event takes place once these bits are reset.

MUDIS and TxUDIS bits are all grouped in the same register. This allows the update of multiple timers (not necessarily synchronized) to be disabled and resumed simultaneously.

The following example is a practical use case. A first power converter is controlled with the master, TIMB and TIMC. TIMB and TIMC must be updated simultaneously with the master timer repetition event. A second converter works in parallel with TIMA, TIMD and TIME, and TIMD, TIME must be updated with TIMA repetition event.

First converter

In HRTIM_MCR, MREPU bit is set: the update will occur at the end of the master timer counter repetition period. In HRTIM_TIMBCR and HRTIM_TIMCCR, MSTU bits are set to have TIMB and TIMC timers updated simultaneously with the master timer.

When the power converter set-point has to be adjusted by software, MUDIS, TBUDIS and TCUDIS bits of the HRTIM_CR register must be set prior to write accessing registers to update the values (for instance the compare values). From this time on, any hardware update request is ignored and the preload registers can be accessed without any risk to have them transferred into the active registers. Once the software processing is over, MUDIS, TBUDIS and TCUDIS bits must be reset. The transfer from preload to active registers will be done as soon as the master repetition event occurs.

Second converter

In HRTIM_TIMACR, TAREPU bit is set: the update will occur at the end of the Timer A counter repetition period. In HRTIM_TIMDCR and HRTIM_TIMECR, TAU bits are set to have TIMD and TIME timers updated simultaneously with Timer A.

When the power converter set-point has to be adjusted by software, TAUDIS, TDUDIS and TEUDIS bits of the HRTIM_CR register must be set prior to write accessing the registers to update the values (for instance the compare values). From this time on, any hardware update request is ignored and the preload registers can be accessed without any risk to have them transferred into the active registers. Once the software processing is over, TAUDIS, TDUDIS and TEUDIS bits can be reset: the transfer from preload to active registers will be done as soon as the Timer A repetition event occurs.

21.3.11 Events propagation within or across multiple timers

The HRTIM offers many possibilities for cascading events or sharing them across multiple timing units, including the master timer, to get full benefits from its modular architecture. These are key features for converters requiring multiple synchronized outputs.

This section summarizes the various options and specifies whether and how an event is propagated within the HRTIM.

TIMx update triggered by the Master timer update

The sources listed in *Table 92* are generating a master timer update. The table indicates if the source event can be used to trigger a simultaneous update in any of TIMx timing units.

Operating condition: MSTU bit is set in HRTIM_TIMxCR register.

Table 92. Master timer update event propagation

Source	Condition	Propagation	Comment
Burst DMA end	BRSTDMA[1:0] = 01	No	Must be done in TIMxCR (UPDGAT[3:0] = 0001)
Roll-over event following a Burst DMA end	BRSTDMA[1:0] = 10	Yes	-
Repetition event caused by a counter roll-over	MREPU = 1	Yes	-
Repetition event caused by a counter reset (from HRTIM_SCIN or software)		No	-
Software update	MSWU = 1	No	All software update bits (TxSWU) are grouped in the HRTIM_CR2 register and can be used for a simultaneous update

TIMx update triggered by the TIMy update

The sources listed in *Table 93* are generating a TIMy update. The table indicates if the given event can be used to trigger a simultaneous update in another or multiple TIMx timers.

Operating condition: TyU bit set in HRTIM_TIMxCR register (source = TIMy and destination = TIMx).

Table 93. TIMx update event propagation

Source	Condition	Propagation	Comment
Burst DMA end	UPDGAT[3:0] = 0001	No	Must be done directly in HRTIM_TIMxCR (UPDGAT[3:0] = 0001)
Update caused by the update enable input	UPDGAT[3:0] = 0011, 0100, 0101	No	Must be done directly in HRTIM_TIMxCR (UPDGAT[3:0] = 0011, 0100, 0101
Master update	MSTU = 1 in HRTIM_TIMyCR	No	Must be done with MSTU = 1 in HRTIM_TIMxCR

Another TIMx update (TIMz>TIMy>TIMx)	TzU=1 in HRTIM_TIMyCR
	TyU=1 in TIMxCR
		No	Must be done with TzU=1 in HRTIM_TIMxCR
			TzU=1 in HRTIM_TIMyCR

Repetition event caused by a counter roll-over	TyREPU = 1	Yes	-
Repetition event caused by a counter reset	TyREPU = 1	-	Refer to counter reset cases below
Counter roll-over	TyRSTU = 1	Yes	-
Counter software reset	TyRST=1 in HRTIM_CR2	No	Can be done simultaneously with update in HRTIM_CR2 register
Counter reset caused by a TIMz compare	TIMzCMPn in HRTIM_RSTyR	No	Must be done using TIMzCMPn in HRTIM_RSTxR
Counter reset caused by external events	EXTEVNTn in HRTIM_RSTyR	Yes	-

Table 93. TIMx update event propagation (continued)

Source	Condition	Propagation	Comment
Counter reset caused by a master compare or a master period	MSTCMPn or MSTPER in HRTIM_RSTyR	No	-
Counter reset caused by a TIMy compare	CMPn in HRTIM_RSTyR	Yes	-
Counter reset caused by an update	UPDT in HRTIM_RSTyR	No	Propagation would result in a lock-up situation (update causing reset causing update)
Counter reset caused by HRTIM_SCIN	SYNCRSTy in HRTIM_TIMyCR	No	-
Software update	TySWU = 1	No	All software update bits (TxSWU) are grouped in the HRTIM_CR2 register and can be used for a simultaneous update

TIMx Counter reset causing a TIMx update

Table 94 lists the counter reset sources and indicates whether they can be used to generate an update.

Operating condition: TxRSTU bit in HRTIM_TIMxCR register.

Table 94. Reset events able to generate an update

Source	Condition	Propagation	Comment
Counter roll-over		Yes	
Update event	UPDT in HRTIM_RSTxR	No	Propagation would result in a lock-up situation (update causing a reset causing an update)
External Event	EXTEVNTn in HRTIM_RSTxR	Yes	-
TIMy compare	TIMyCMPn in HRTIM_RSTxR	Yes	-
Master compare	MSTCMPn in HRTIM_RSTxR	Yes	-
Master period	MSTPER in HRTIM_RSTxR	Yes	-
Compare 2 and 4	CMPn in HRTIM_RSTxR	Yes	-
Software	TxRST=1 in HRTIM_CR2	Yes	-
HRTIM_SCIN	SYNCRSTx in HRTIM_TIMxCR	Yes	-

TIMx update causing a TIMx counter reset

Table 95 lists the update event sources and indicates whether they can be used to generate a counter reset.

Operating condition: UPDT bit set in HRTIM_RSTxR.

Table 95. Update event propagation for a timer reset

Source	Condition	Propagation	Comment
Burst DMA end	UPDGAT[3:0] = 0001	Yes	-
Update caused by the update enable input	UPDGAT[3:0] = 0011, 0100, 0101	Yes	-
Master update caused by a roll-over after a Burst DMA	MSTU = 1 in HRTIM_TIMxCR BRSTDMA[1:0] = 10 in HRTIM_MCR	Yes	-

Master update caused by a repetition event following a roll-over	MSTU = 1 in HRTIM_TIMxCR
	MREPU = 1 in HRTIM_MCR
		Yes	-

Master update caused by a repetition event following a counter reset (software or due to HRTIM_SCIN)		No	-

Software triggered master timer update	MSTU = 1 in HRTIM_TIMxCR
	MSWU = 1
	in HRTIM_CR2
		No	All software update bits (TxSWU) are grouped in the HRTIM_CR2 register and can be used for a simultaneous update

TIMy update caused by a TIMy counter roll-over	TyU = 1 in HRTIM_TIMxCR
	TyRSTU = 1 in HRTIM_TIMyCR
		Yes	-

TIMy update caused by a TIMy repetition event	TyU = 1 in HRTIM_TIMxCR
	TyREPU = 1 in HRTIM_TIMyCR
		Yes	-

TIMy update caused by an external event or a TIMy compare (through a TIMy reset)	TyU = 1 in HRTIM_TIMxCR
	TyRSTU = 1 in HRTIM_TIMyCR
	EXTEVNTn or CMP4/2
	in HRTIM_RSTyCR
		Yes	-

TIMy update caused by sources other than those listed above	TyU = 1 in HRTIM_TIMxCR	No	-

Table 95. Update event propagation for a timer reset (continued)

Source	Condition	Propagation	Comment
Repetition event following a roll-over	TxREPU = 1 in HRTIM_TIMxCR	Yes	-
Repetition event following a counter reset		No	-
Timer reset	TxRSTU = 1 in HRTIM_TIMxCR	No	Propagation would result in a lock-up situation (reset causing an update causing a reset)
Software	TxSWU in HRTIM_CR2	No	-

21.3.12 Output management

Each timing unit controls a pair of outputs. The outputs have three operating states:

- RUN: this is the main operating mode, where the output can take the active or inactive level as programmed in the crossbar unit.
- IDLE: this state is the default operating state after an HRTIM reset, when the outputs are disabled by software or during a burst mode operation (where outputs are temporary disabled during a normal operating mode; refer to *Section 21.3.13* for more details). It is either permanently active or inactive.
- FAULT: this is the safety state, entered in case of a shut-down request on FAULTx inputs. It can be permanently active, inactive or Hi-Z.

The output status is indicated by TxyOEN bit in HRTIM_OENR register and TxyODS bit in HRTIM_ODSR register, as in *Table 96*.

Table 96. Output state programming, x= A..E, y = 1 or 2

TxyOEN (control/status) (set by software, cleared by hardware)	TxyODS (status)	Output operating state
1	x	RUN
0	0	IDLE
0	1	FAULT

TxyOEN bit is both a control and a status bit: it must be set by software to have the output in RUN mode. It is cleared by hardware when the output goes back in IDLE or FAULT mode. When TxyOEN bit is cleared, TxyODS bit indicates whether the output is in the IDLE or FAULT state. A third bit in the HRTIM_ODISR register allows to disable the output by software.

Figure 280. Output management overview

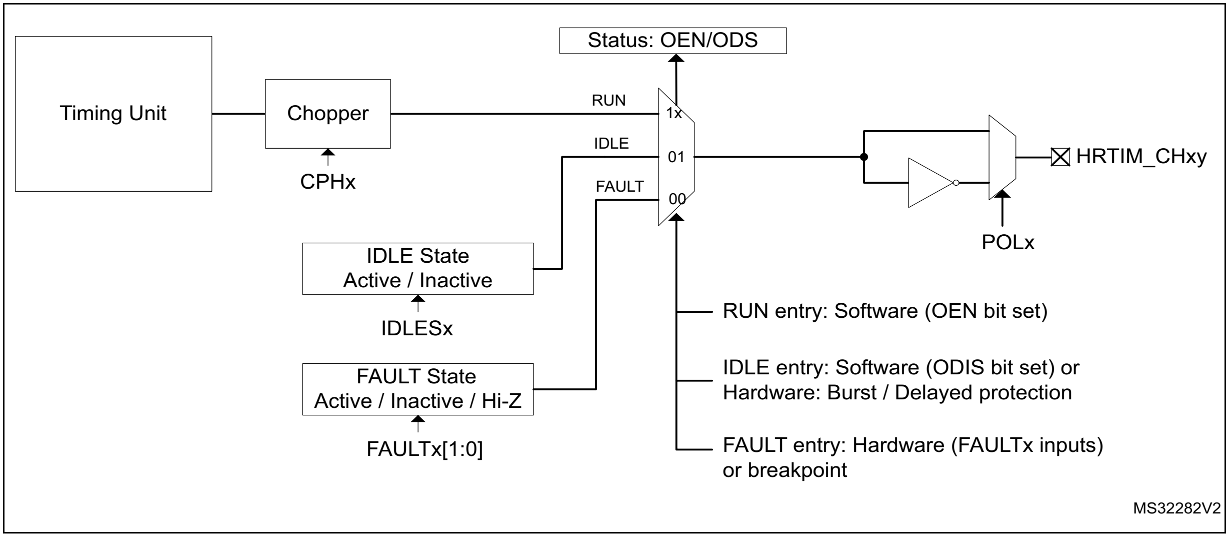
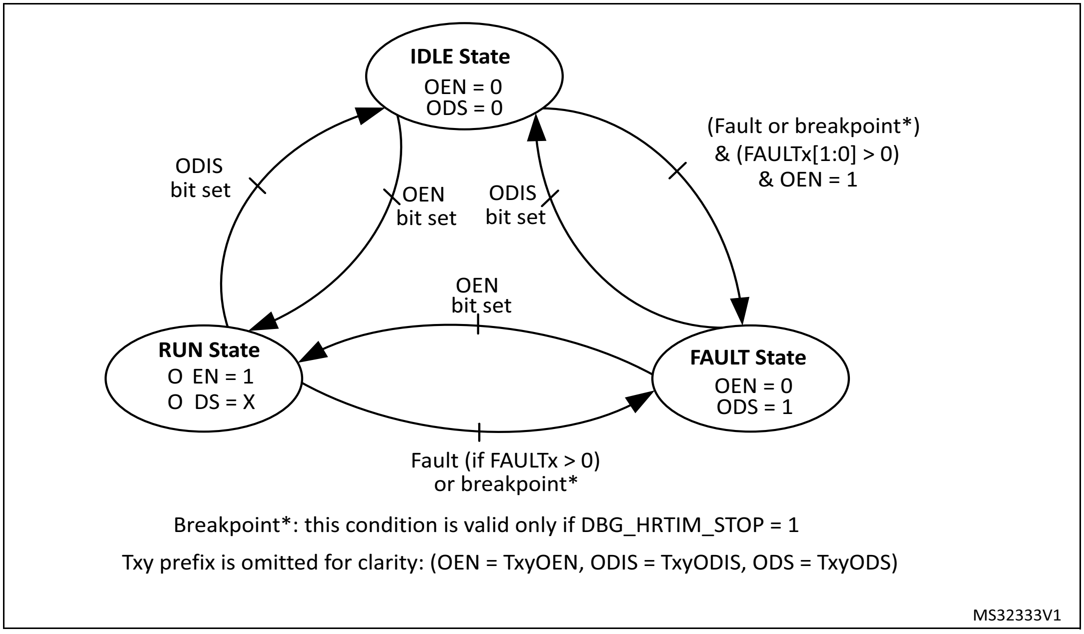


Figure 281 summarizes the bit values for the three states and how the transitions are triggered. Faults can be triggered by any external or internal fault source, as listed in *Section 21.3.15*, while the Idle state can be entered when the burst mode or delayed protections are active.

Figure 281. HRTIM output states and transitions



The FAULT and IDLE levels are defined as active or inactive. Active (or inactive) refers to the level on the timer output that causes a power switch to be closed (or opened for an inactive state).

The IDLE state has the highest priority: the transition FAULT → IDLE is possible even if the FAULT condition is still valid, triggered by ODIS bit set.

The FAULT state has priority over the RUN state: if TxyOEN bit is set simultaneously with a Fault event, the FAULT state will be entered. The condition is given on the transition IDLE →

FAULT, as in *Figure 281*: fault protection needs to be enabled (FAULTx[1:0] bits = 01, 10, 11) and the Txy OEN bit set with a fault active (or during a breakpoint if DBG_HRTIM_STOP = 1).

The output polarity is programmed using POLx bits in HRTIM_OUTxR. When POLx = 0, the polarity is positive (output active high), while it is active low in case of a negative polarity (POLx = 1). Practically, the polarity is defined depending on the power switch to be driven (PMOS vs. NMOS) or on a gate driver polarity.

The output level in the FAULT state is configured using FAULTx[1:0] bits in HRTIM_OUTxR, for each output, as follows:

- 00: output never enters the fault state and stays in RUN or IDLE state
- 01: output at active level when in FAULT
- 10: output at inactive level when in FAULT
- 11: output is tri-stated when in FAULT. The safe state must be forced externally with pull-up or pull-down resistors, for instance.

Note: FAULTx[1:0] bits must not be changed as long as the outputs are in FAULT state.

The level of the output in IDLE state is configured using IDLESx bit in HRTIM_OUTxR, as follows:

- 0: output at inactive level when in IDLE
- 1: output at active level when in IDLE

When TxyOEN bit is set to enter the RUN state, the output is immediately connected to the crossbar output. If the timer clock is stopped, the level will either be inactive (after an HRTIM reset) or correspond to the RUN level (when the timer was stopped and the output disabled).

During the HRTIM initialization, the output level can be prepositioned prior to have it in RUN mode, using the software forced output set and reset in the HRTIM_SETx1R and HRTIM_RSTx1R registers.

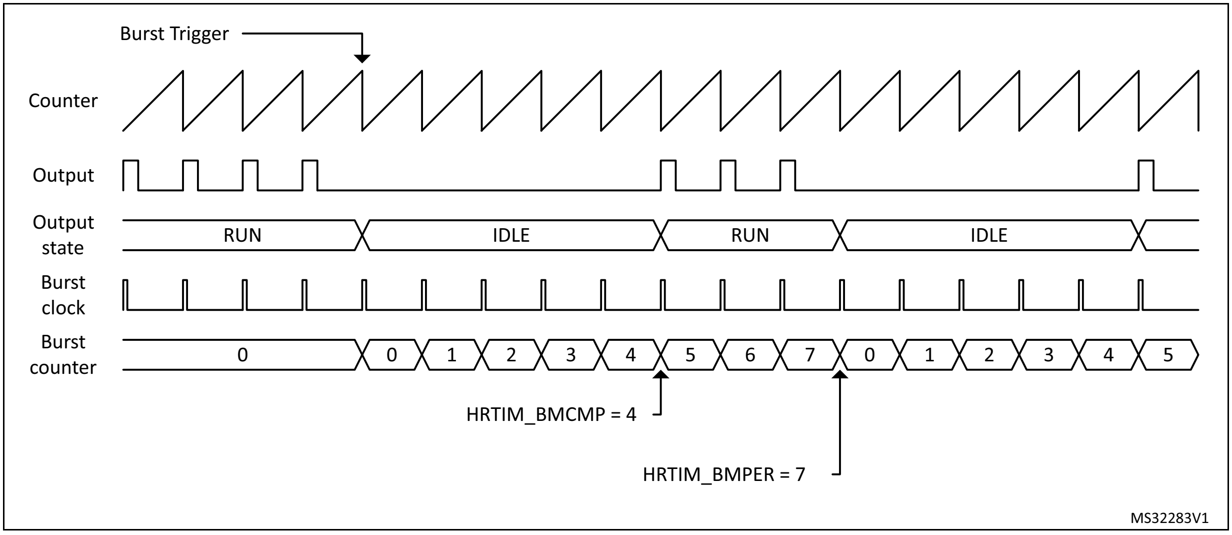
21.3.13 Burst mode controller

The burst mode controller allows to have the outputs alternatively in IDLE and RUN state, by hardware, so as to skip some switching periods with a programmable periodicity and duty cycle.

Burst mode operation is of common use in power converters when operating under light loads. It can significantly increase the efficiency of the converter by reducing the number of transitions on the outputs and the associated switching losses.

When operating in burst mode, one or a few pulses are outputs followed by an idle period equal to several counting periods, typically, where no output pulses are produced, as shown in the example on *Figure 282*.

Figure 282. Burst mode operation example



The burst mode controller consists of:

- A counter that can be clocked by various sources, either within or outside the HRTIM (typically the end of a PWM period).
- A compare register to define the number of idle periods: HRTIM_BMCMP.
- A period register to define the burst repetition rate (corresponding to the sum of the idle and run periods): HRTIM_BMPER.

The burst mode controller is able to take over the control of any of the 10 PWM outputs. The state of each output during a burst mode operation is programmed using IDLESx and IDLEMx bits in the HRTIM_OUTxR register, as in *Table 97*.

Table 97. Timer output programming for burst mode

IDLEMx	IDLESx	**Output state during burst mode**
0	X	No action: the output is not affected by the burst mode operation.
1	0	Output inactive during the burst
1	1	Output active during the burst

Note: IDLEMx bit must not be changed while the burst mode is active.

The burst mode controller only acts on the output stage. A number of events are still generated during the idle period:

- Output set/reset interrupt or DMA requests
- External event filtering based on Tx2 output signal
- Capture events triggered by output set/reset

During the burst mode, neither start not reset events are generated on the HRTIM_SCOUT output, even if TxBM bit is set.

Operating mode

It is necessary to have the counter enabled (TxCEN bit set) before using the burst mode on a given timing unit.The burst mode is enabled with BME bit in the HRTIM_BMCR register.

It can operate in continuous or single-shot mode, using BMOM bit in the HRTIM_BMCR register. The continuous mode is enabled when BMOM = 1. The Burst operation is maintained until BMSTAT bit in HRTIM_BMCR is reset to terminate it.

In single-shot mode (BMOM = 0), the idle sequence is executed once, following the burst mode trigger, and the normal timer operation is resumed immediately after.

The duration of the idle and run periods is defined with a burst mode counter and 2 registers. The HRTIM_BMCMPR register defines the number of counts during which the selected timer(s) are in an idle state (idle period). HRTIM_BMPER defines the overall burst mode period (sum of the idle and run periods). Once the initial burst mode trigger has occurred, the idle period length is HRTIM_BMCMPR+1, the overall burst period is HRTIM_BMPER+1.

Note: The burst mode period must not be less than or equal to the deadtime duration defined with DTRx[8:0] and DTFx[8:0] bitfields.

The counters of the timing units and the master timer can be stopped and reset during the burst mode operation. HRTIM_BMCR holds 6 control bits for this purpose: MTBM (master) and TABM..TEBM for Timer A..E.

When MTBM or TxBM bit is reset, the counter clock is maintained. This allows to keep a phase relationship with other timers in multiphase systems, for instance.

When MTBM or TxBM bit is set, the corresponding counter is stopped and maintained in reset state during the burst idle period. This allows to have the timer restarting a full period when exiting from idle. If SYNCSRC[1:0] = 00 or 10 (synchronization output on the master start or timer A start), a pulse is sent on the HRTIM_SCOUT output when exiting the burst mode.

Note: TxBM bit must not be set when the balanced idle mode is active (DLYPRT[1:0] = 0x11).

Burst mode clock

The burst mode controller counter can be clocked by several sources, selected with BMCLK[3:0] bits in the HRTIM_BMCR register:

- BMCLK[3:0] = 0000 to 0101: Master timer and TIMA..E reset/roll-over events. This allows to have burst mode idle and run periods aligned with the timing unit counting period (both in free-running and counter reset mode).
- BMCLK[3:0] = 0110 to 1001: The clocking is provided by the general purpose timers, as in *Table 98*. In this case, the burst mode idle and run periods are not necessarily aligned with timing unit counting period (a pulse on the output may be interrupted, resulting a waveform with modified duty cycle for instance.
- BMCLK[3:0] = 1010: The f_{HRTIM} clock prescaled by a factor defined with BMPRSC[3:0] bits in HRTIM_BMCR register. In this case, the burst mode idle and run periods are not necessarily aligned with the timing unit counting period (a pulse on the output may be interrupted, resulting in a waveform with a modified duty cycle, for instance.

Table 98. Burst mode clock sources from general purpose timer

BMCLK[3:0]	Clock source
0110	TIM16 OC
0111	TIM17 OC
1000	TIM7 TRGO
1001	Reserved

The pulsewidth on TIM16/17 OC output must be at least N f_{HRTIM} clock cycles long to be detected by the HRTIM burst mode controller.

Burst mode triggers

To trigger the burst operation, 32 sources are available and are selected using the HRTIM_BMTRGR register:

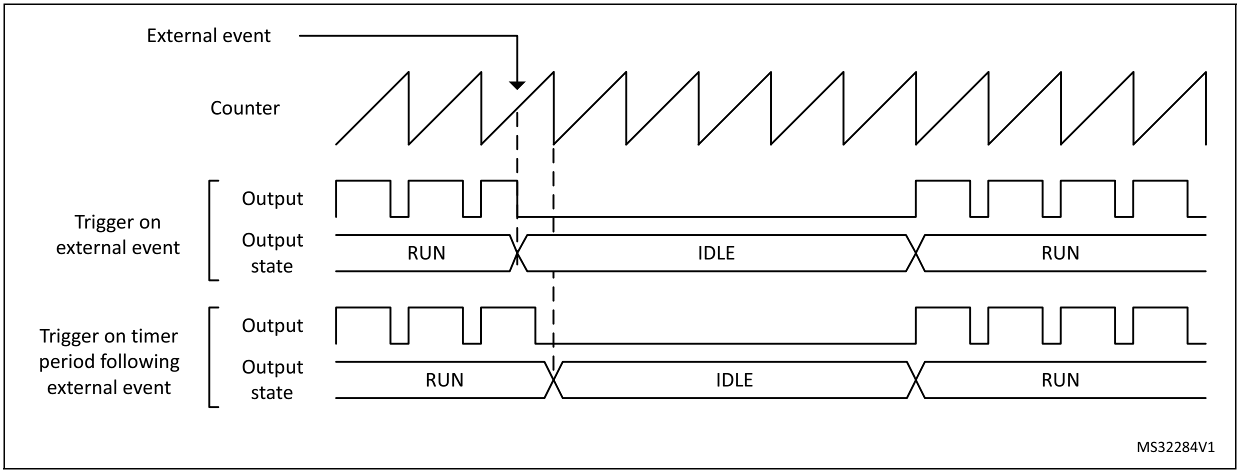
- Software trigger (set by software and reset by hardware)
- 6 Master timer events: repetition, reset/roll-over, Compare 1 to 4
- 5 x 4 events from timers A..E: repetition, reset/roll-over, Compare 1 and 2
- External Event 7 (including TIMA event filtering) and External Event 8 (including TIMD event filtering)
- Timer A period following External Event 7 (including TIMA event filtering)
- Timer D period following External Event 8 (including TIMD event filtering)
- On-chip events coming from other general purpose timer (TIM7_TRGO output)

These sources can be combined to have multiple concurrent triggers.

Burst mode is not re-triggerable. In continuous mode, new triggers are ignored until the burst mode is terminated, while in single-shot mode, the triggers are ignored until the current burst completion including run periods (HRTIM_BMPER+1 cycles). This is also valid for software trigger (the software bit is reset by hardware even if it is discarded).

Figure 283 shows how the burst mode is started in response to an external event, either immediately or on the timer period following the event.

Figure 283. Burst mode trigger on external event



For TAEEV7 and TDEEV8 combined triggers (trigger on a Timer period following an external event), the external event detection is always active, regardless of the burst mode programming and the on-going burst operation:

- When the burst mode is enabled (BME=1) or the trigger is enabled (TAEEV7 or TDEEV8 bit set in the BMTRG register) in between the external event and the timer period event, the burst is triggered.
- The single-shot burst mode is re-triggered even if the external event occurs before the burst end (as long as the corresponding period happens after the burst).

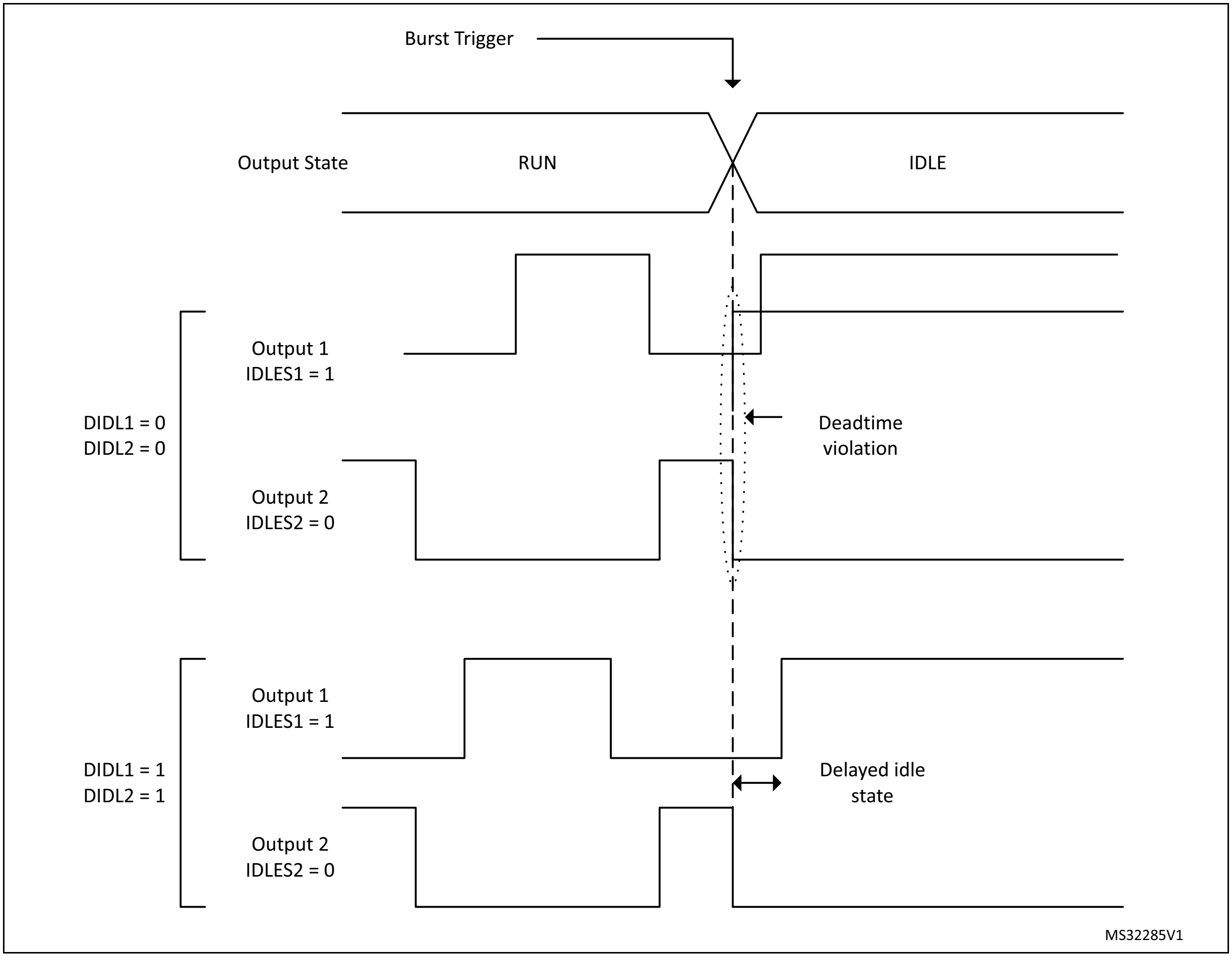
Note: *TAEEV7 and TDEEV8 triggers are valid only after a period event. If the counter is reset before the period event, the pending EEV7/8 event is discarded.*

Burst mode delayed entry

By default, the outputs are taking their idle level (as per IDLES1 and IDLES2 setting) immediately after the burst mode trigger.

It is also possible to delay the burst mode entry and force the output to an inactive state during a programmable period before the output takes its idle state. This is useful when driving two complementary outputs, one of them having an active idle state, to avoid a deadtime violation as shown on *Figure 284*. This prevents any risk of shoot through current in half-bridges, but causes a delayed response to the burst mode entry.

Figure 284. Delayed burst mode entry with deadtime enabled and IDLESx = 1



The delayed burst entry mode is enabled with DIDLx bit in the HRTIM_OUTxR register (one enable bit per output). It forces a deadtime insertion before the output takes its idle state. Each TIMx output has its own deadtime value:

- DTRx[8:0] on output 1 when DIDL1 = 1
- DTFx[8:0] on output 2 when DIDL2 = 1

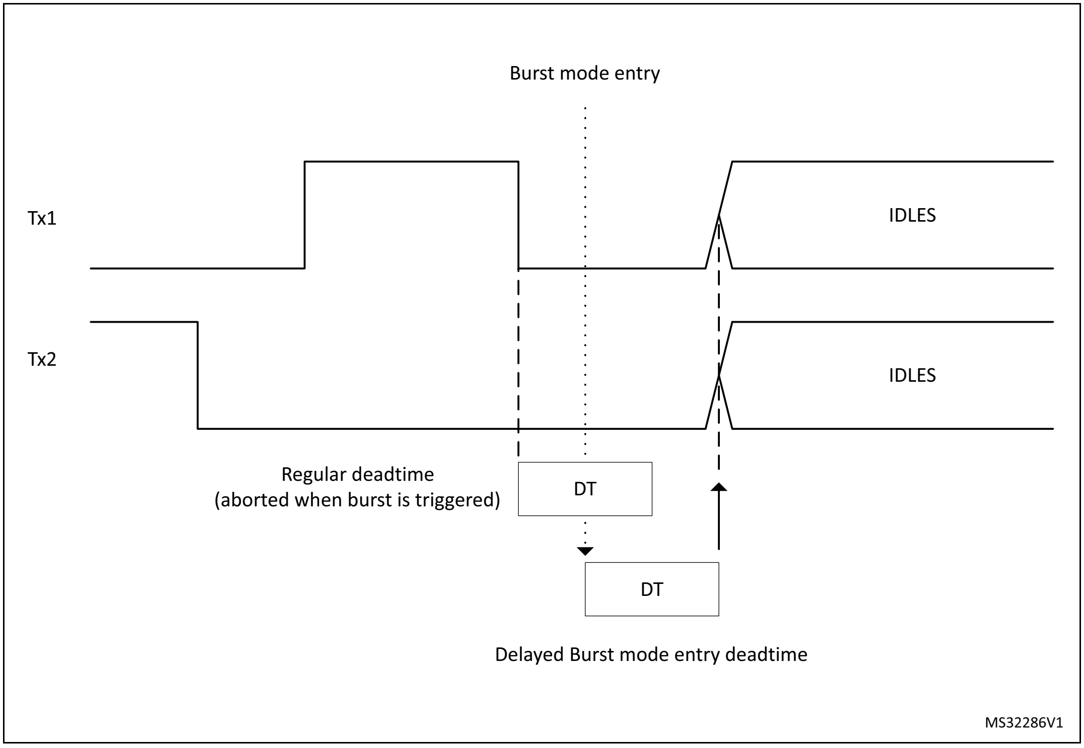
DIDLx bits can be set only if one of the outputs has an active idle level during the burst mode (IDLES = 1) and only when positive deadtimes are used (SDTR/SDTF set to 0).

Note: The delayed burst entry mode uses deadtime generator resources. Consequently, when any of the 2 DIDLx bits is set and the corresponding timing unit uses the deadtime insertion (DTEN bit set in HRTIM_OUTxR), it is not possible to use the timerx output 2 as a filter for external events (Tx2 filtering signal is not available).

When durations defined by DTRx[8:0] and DTFx[8:0] are lower than 3 f_{HRTIM} clock cycle periods, the limitations related to the narrow pulse management listed in *Section 21.3.6* must be applied.

When the burst mode entry arrives during the regular deadtime, it is aborted and a new deadtime is re-started corresponding to the inactive period, as on *Figure 285*.

Figure 285. Delayed Burst mode entry during deadtime


Burst mode entry
Tx1
IDLES
Tx2
IDLES
Regular deadtime
(aborted when burst is triggered)
DT
DT
Delayed Burst mode entry deadtime
MS32286V1

Burst mode exit

The burst mode exit is either forced by software (in continuous mode) or once the idle period is elapsed (in single-shot mode). In both cases, the counter is re-started immediately (if it was hold in a reset state with MTBM or TxBM bit = 1), but the effective output state transition from the idle to active mode only happens after the programmed set/reset event.

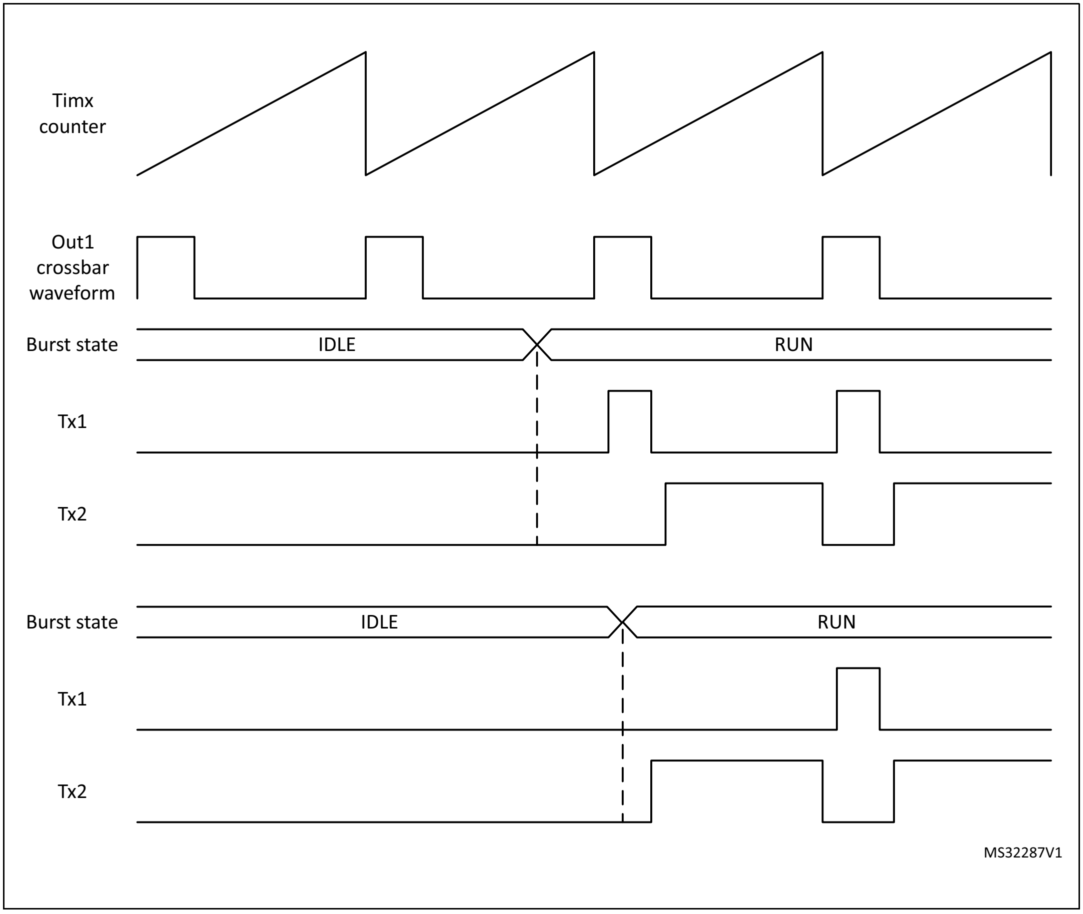
A burst period interrupt is generated in single-shot and continuous modes when BMPERIE enable bit is set in the HRTIM_IER register. This interrupt can be used to synchronize the burst mode exit with a burst period in continuous burst mode.

Figure 286 shows how a normal operation is resumed when the deadtime is enabled. Although the burst mode exit is immediate, this is only effective on the first set event on any of the complementary outputs.

Two different cases are presented:

1. The burst mode ends while the signal is inactive on the crossbar output waveform. The active state is resumed on Tx1 and Tx2 on the set event for the Tx1 output, and the Tx2 output does not take the complementary level on burst exit.
2. The burst mode ends while the crossbar output waveform is active: the activity is resumed on the set event of Tx2 output, and Tx1 does not take the active level immediately on burst exit.

Figure 286. Burst mode exit when the deadtime generator is enabled



The behavior described above is slightly different when the push-pull mode is enabled. The push-pull mode forces an output reset at the beginning of the period if the output is inactive, or symmetrically forces an active level if the output was high during the preceding period.

Consequently, an output with an active idle state can be reset at the time the burst mode is exited even if no transition is explicitly programmed. For symmetrical reasons, an output can be set at the time the burst mode is exited even if no transition is explicitly programmed, in case it was active when it entered in idle state.

Burst mode registers preloading and update

BMPREN bit (Burst mode Preload Enable) allows to have the burst mode compare and period registers preloaded (HRTIM_BMCMP and HRTIM_BMPER).

When BMPREN is set, the transfer from preload to active register happens:

- when the burst mode is enabled (BME = 1),
- at the end of the burst mode period.

A write into the HRTIM_BMPER period register disables the update temporarily, until the HRTIM_BMCMP compare register is written, to ensure the consistency of the two registers when they are modified.

If the compare register only needs to be changed, a single write is necessary. If the period only needs to be changed, it is also necessary to re-write the compare to have the new values taken into account.

When BMPREN bits is reset, the write access into BMCMPR and BMPER directly updates the active register. In this case, it is necessary to consider when the update is done during the overall burst period, for the 2 cases below:

a) Compare register update

If the new compare value is above the current burst mode counter value, the new compare is taken into account in the current period.

If the new compare value is below the current burst mode counter value, the new compare is taken into account in the next burst period in continuous mode, and ignored in single-shot mode (no compare match will occur and the idle state will last until the end of the idle period).

b) Period register update

If the new period value is above the current burst mode counter value, the change is taken into account in the current period.

Note: If the new period value is below the current burst mode counter value, the new period will not be taken into account, the burst mode counter will overflow (at 0xFFFF) and the change will be effective in the next period. In single-shot mode, the counter will roll over at 0xFFFF and the burst mode will re-start for another period up to the new programmed value.

Burst mode emulation using a compound register

The burst mode controller only controls one or a set of timers for a single converter. When the burst mode is necessary for multiple independent timers, it is possible to emulate a simple burst mode controller using the DMA and the HRTIM_CMP1CxR compound register, which holds aliases of both the repetition and the Compare 1 registers.

This is applicable to a converter which only requires a simple PWM (typically a buck converter), where the duty cycle only needs to be updated. In this case, the CMP1 register is used to reset the output (and define the duty cycle), while it is set on the period event.

In this case, a single 32-bit write access in CMP1CxR is sufficient to define the duty cycle (with the CMP1 value) and the number of periods during which this duty cycle is maintained (with the repetition value). To implement a burst mode, it is then only necessary to transfer by DMA (upon repetition event) two 32-bit data in continuous mode, organized as follows:

CMPC1xR = {REP_Run; CMP1 = Duty_Cycle}, {REP_Idle; CMP1 = 0}

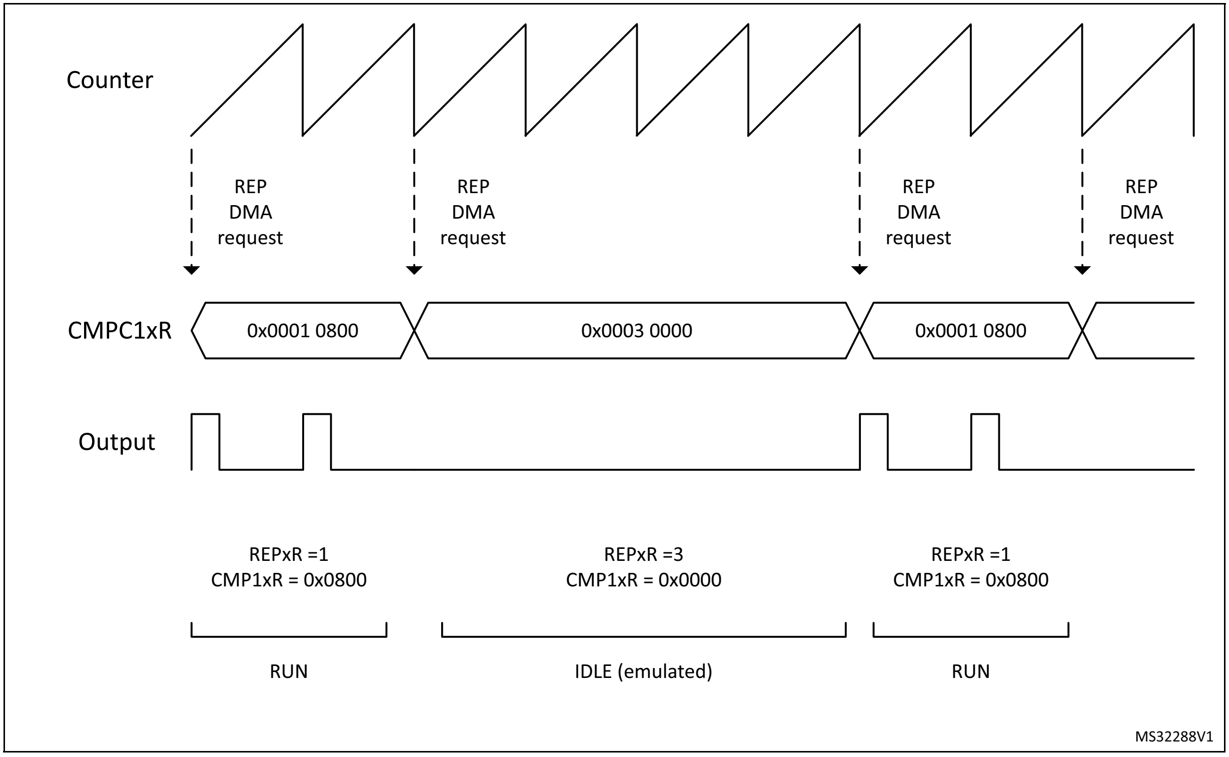
For instance, the values:

{0x0003 0000}: CMP1 = 0 for 3 periods

{0x0001 0800}: CMP1 = 0x0800 for 1 period

will provide a burst mode with 2 periods active every 6 PWM periods, as shown on *Figure 287*.

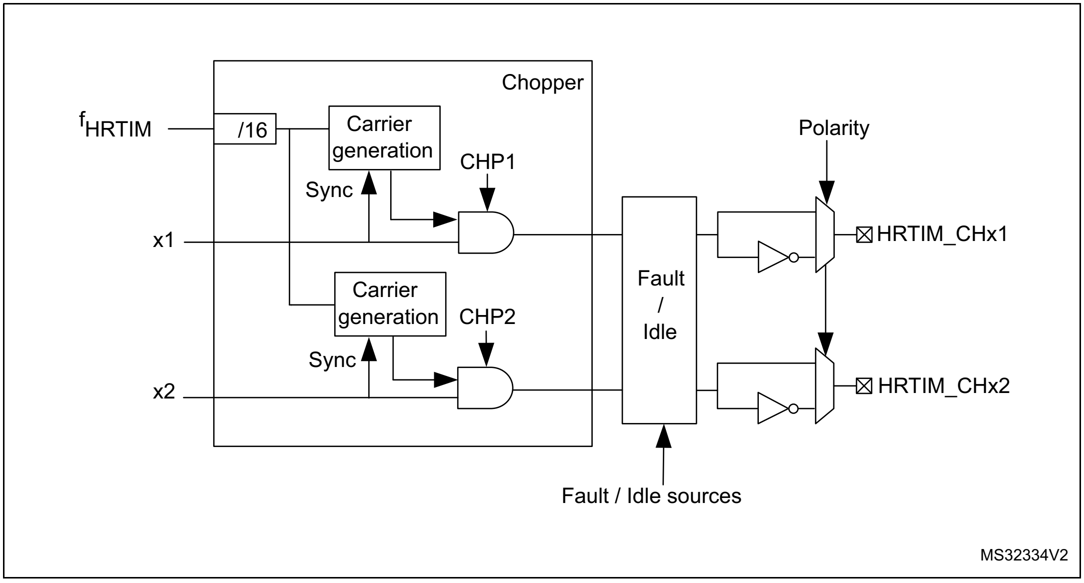
Figure 287. Burst mode emulation example



21.3.14 Chopper

A high-frequency carrier can be added on top of the timing unit output signals to drive isolation transformers. This is done in the output stage before the polarity insertion, as shown on *Figure 288*, using CHP1 and CHP2 bits in the HRTIM_OUTxR register, to enable chopper on outputs 1 and 2, respectively.

Figure 288. Carrier frequency signal insertion



The chopper parameters can be adjusted using the HRIM_CHPxR register, with the possibility to define a specific pulsewidth at the beginning of the pulse, to be followed by a carrier frequency with programmable frequency and duty cycle, as in *Figure 289*.

CARFRQ[3:0] bits define the frequency, ranging from 562.5 kHz to 9 MHz (for f_{HRTIM} = 144 MHz) following the formula $F_{CHPFRQ} = f_{HRTIM}$ / (16 x (CARFRQ[3:0]+1)).

The duty cycle can be adjusted by 1/8 step with CARDTY[2:0], from 0/8 up to 7/8 duty cycle. When CARDTY[2:0] = 000 (duty cycle = 0/8), the output waveform only contains the starting pulse following the rising edge of the reference waveform, without any added carrier.

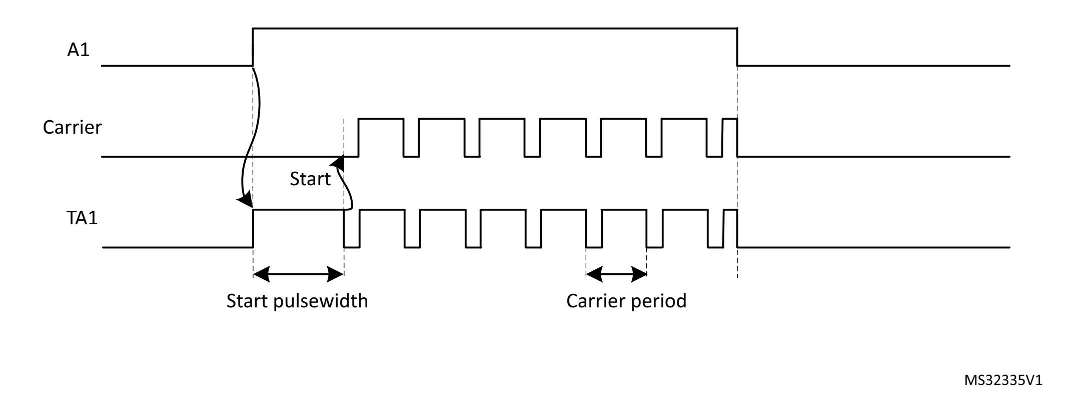
The pulsewidth of the initial pulse is defined using the STRPW[3:0] bitfield as follows: t1STPW = (STRPW[3:0]+1) x 16 x t_{HRTIM} and ranges from 111 ns to 1.77 µs (for f_{HRTIM}=144 MHz).

The carrier frequency parameters are defined based on the f_{HRTIM} frequency, and are not dependent from the CKPSC[2:0] setting.

In chopper mode, the carrier frequency and the initial pulsewidth are combined with the reference waveform using an AND function. A synchronization is performed at the end of the initial pulse to have a repetitive signal shape.

The chopping signal is stopped at the end of the output waveform active state, without waiting for the current carrier period to be completed. It can thus contain shorter pulses than programmed.

Figure 289. HRTIM outputs with Chopper mode enabled



Note: CHP1 and CHP2 bits must be set prior to the output enable done with TxyOEN bits in the HRTIM_OENR register.

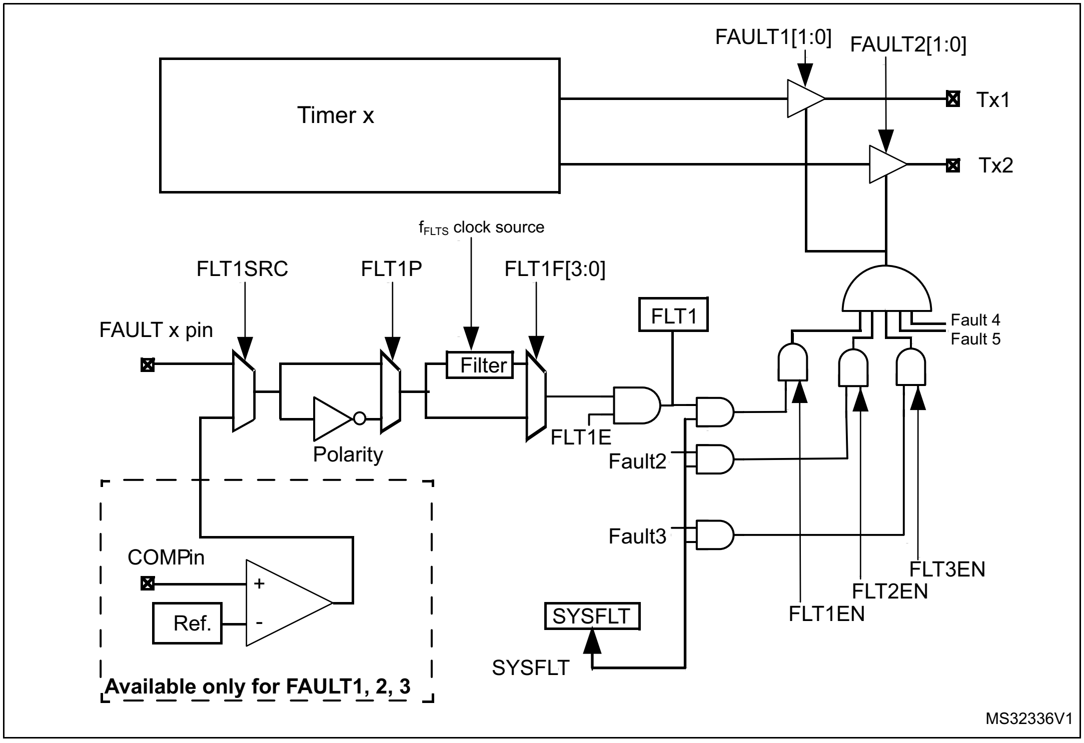
CARFRQ[2:0], CARDTY[2:0] and STRPW[3:0] bitfields cannot be modified while the chopper mode is active (at least one of the two CHPx bits is set).

21.3.15 Fault protection

The HRTIMER has a versatile fault protection circuitry to disable the outputs in case of an abnormal operation. Once a fault has been triggered, the outputs take a predefined safe state. This state is maintained until the output is re-enabled by software. In case of a permanent fault request, the output will remain in its fault state, even if the software attempts to re-enable them, until the fault source disappears.

The HRTIM has 5 FAULT input channels; all of them are available and can be combined for each of the 5 timing units, as shown on *Figure 290*.

Figure 290. Fault protection circuitry (FAULT1 fully represented, FAULT2..5 partially)



Each fault channel is fully configurable using HRTIM_FLTINR1 and HRTIM_FLTINR2 registers before being routed to the timing units. FLTxSRC bit selects the source of the Fault signal, that can be either a digital input or an internal event (built-in comparator output).

Table 99 summarizes the available sources for each of the 10 faults channels:

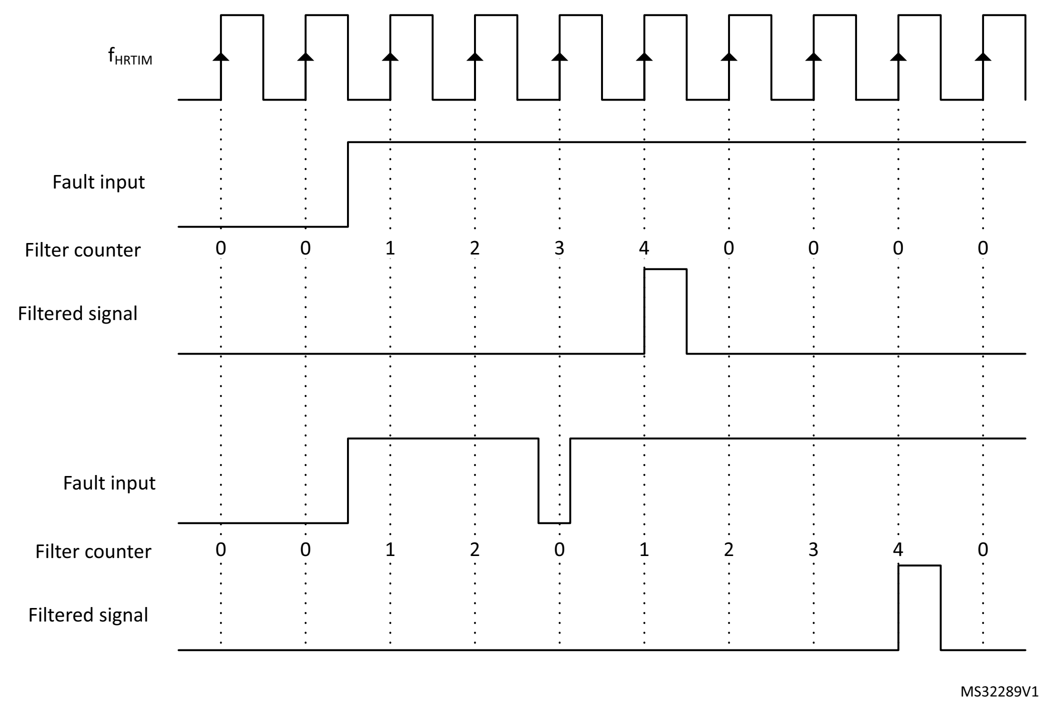
Table 99. Fault inputs

Fault channel	External Input (FLTxSRC = 0)	On-chip source (FLTxSRC = 1)
FAULT 1	PA12	COMP2
FAULT 2	PA15	COMP4
FAULT 3	PB10	COMP6
FAULT 4	PB11	NC
FAULT 5	PC7	NC

The polarity of the signal can be selected to define the active level, using the FLTxP polarity bit in HRTIM_FLTINRx registers. If FLTxP = 0, the signal is active at low level; if FLTxP = 1, it is active when high.

The fault information can be filtered after the polarity setting. If FLTxF[3:0] bitfield is set to 0000, the signal is not filtered and will act asynchronously, independently from the f_{HRTIM} clock. For all other FLTxF[3:0] bitfield values, the signal is digitally filtered. The digital filter is made of a counter in which a number N of valid samples is needed to validate a transition on the output. If the input value changes before the counter has reached the value N, the counter is reset and the transition is discarded (considered as a spurious event). If the counter reaches N, the transition is considered as valid and transmitted as a correct external event. Consequently, the digital filter adds a latency to the external events being filtered, depending on the sampling clock and on the filter length (number of valid samples expected). *Figure 291* shows how a spurious fault signal is filtered.

Figure 291. Fault signal filtering (FLTxF[3:0]= 0010: $f_{SAMPLING} = f_{HRTIM}$, N = 4)



The filtering period ranges from 2 cycles of the f_{HRTIM} clock up to 8 cycles of the f_{FLTS} clock divided by 32. f_{FLTS} is defined using FLTSD[1:0] bits in the HRTIM_FLTINR2 register. *Table 100* summarizes the sampling rate and the filter length. A jitter of 1 sampling clock period must be subtracted from the filter length to take into account the uncertainty due to the sampling and have the effective filtering.

Table 100. Sampling rate and filter length vs FLTFxF[3:0] and clock setting

	f_{FLTS} vs FLTSD[1:0]				Filter length for f_{HRTIM} = 144 MHz	
FLTFxF[3:0]	**00**	**01**	**10**	**11**	**Min**	**Max**
0001,0010,0011	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}, N =2 13.9 ns	f_{HRTIM}, N =8 55.5 ns
0100, 0101	f_{HRTIM} /2	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /2, N = 6 83.3 ns	f_{HRTIM} /16, N = 8 888.9 ns
0110, 0111	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /4, N = 6 166.7 ns	f_{HRTIM} /32, N = 8 1.777 µs
1000, 1001	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /8, N = 6 333.3 ns	f_{HRTIM} /64, N = 8 3.55 µs

Table 100. Sampling rate and filter length vs FLTFxF[3:0] and clock setting (continued)

	f_{FLTS} vs FLTSD[1:0]				Filter length for f_{HRTIM} = 144 MHz	
FLTFxF[3:0]	00	01	10	11	Min	Max
1010, 1011, 1100	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /16, N = 5 555.5ns	f_{HRTIM} /128, N = 8 7.11 µs
1101, 1110, 1111	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /256	f_{HRTIM} /32, N = 5 1.11 µs	f_{HRTIM} /256, N = 8 14.22 µs

System fault input (SYSFLT)

This fault is provided by the MCU Class B circuitry (see the System configuration controller (SYSCFG) section for details) and corresponds to a system fault coming from:

- the Clock Security System
- the SRAM parity checker
- the Cortex®-M4-lockup signal
- the PVD detector

This input overrides the FAULT inputs and disables all outputs having FAULTy[1:0] = 01, 10, 11.

For each FAULT channel, a write-once FLTxLCK bit in the HRTIM_FLTxR register allows to lock FLTxE, FLTxP, FLTxSRC, FLTxF[3:0] bits (it renders them read-only), for functional safety purpose. If enabled, the fault conditioning set-up is frozen until the next HRTIM or system reset.

Once the fault signal is conditioned as explained above, it is routed to the timing units. For any of them, the 5 fault channels are enabled using bits FLT1EN to FLT5EN in the HRTIM_FLTxR register, and they can be selected simultaneously (the sysfault is automatically enabled as long as the output is protected by the fault mechanism). This allows to have, for instance:

- One fault channel simultaneously disabling several timing units
- Multiple fault channels being ORed to disable a single timing unit

A write-once FLTLCK bit in the HRTIM_FLTxR register allows to lock FLTxEN bits (it renders them read-only) until the next reset, for functional safety purpose. If enabled, the timing unit fault-related set-up is frozen until the next HRTIM or system reset.

For each of the timers, the output state during a fault is defined with FAULT1[1:0] and FAULT2[1:0] bits in the HRTIM_OUTxR register (see *Section 21.3.12*).

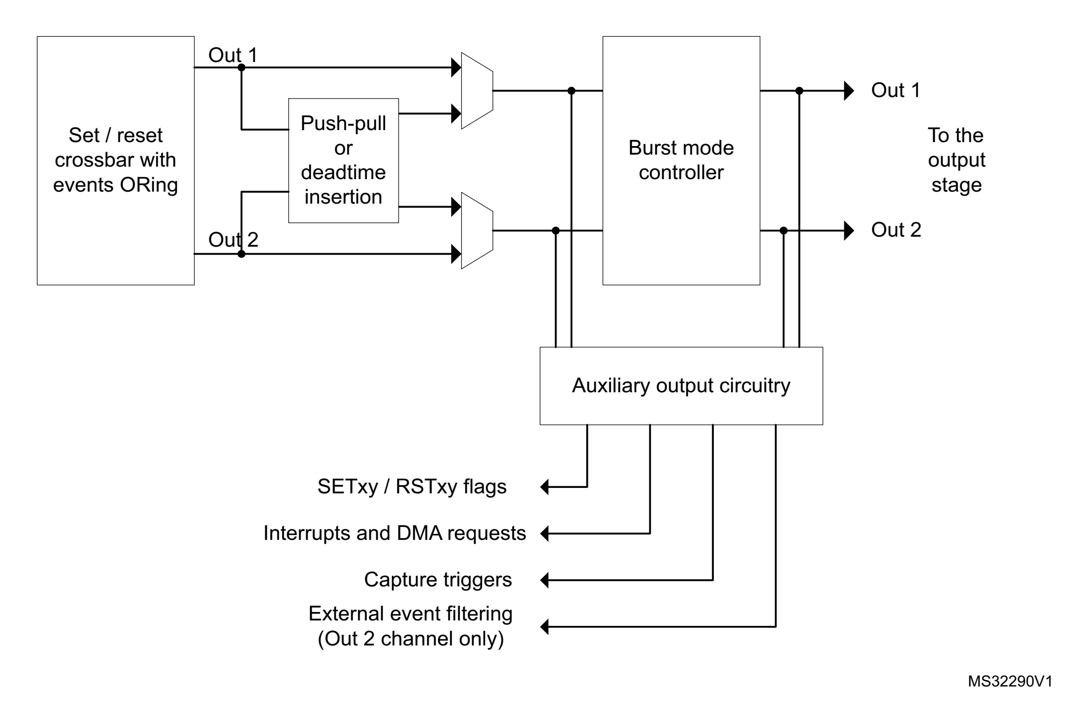
21.3.16 Auxiliary outputs

Timer A to E have auxiliary outputs in parallel with the regular outputs going to the output stage. They provide the following internal status, events and signals:

- SETxy and RSTxy status flags, together with the corresponding interrupts and DMA requests
- Capture triggers upon output set/reset
- External event filters following a Tx2 output copy (see details in *Section 21.3.8*)

The auxiliary outputs are taken either before or after the burst mode controller, depending on the HRTIM operating mode. An overview is given on *Figure 292*.

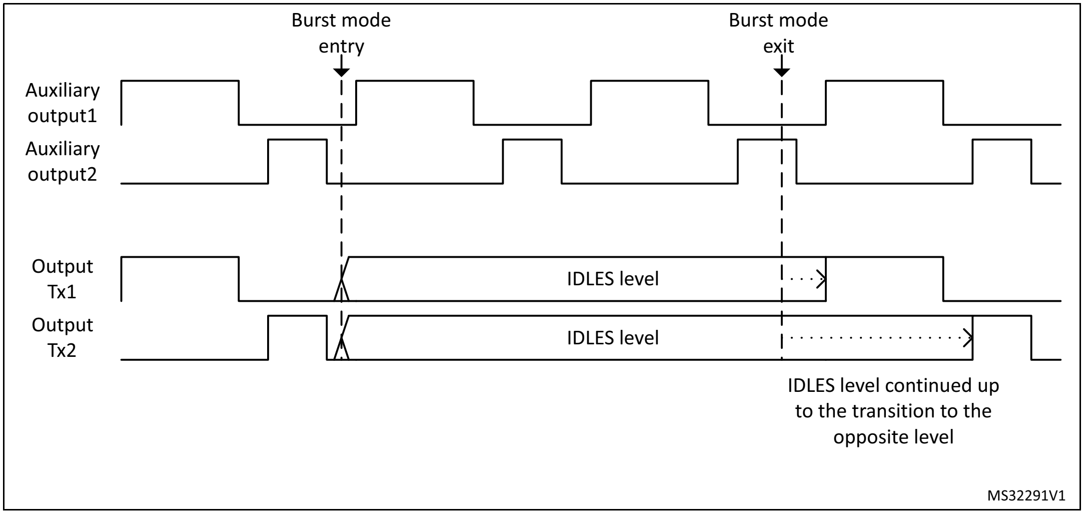
Figure 292. Auxiliary outputs


Set / reset crossbar with events ORing
Out 1
Out 2
Push-pull or deadtime insertion
Burst mode controller
Out 1
Out 2
To the output stage
Auxiliary output circuitry
SETxy / RSTxy flags
Interrupts and DMA requests
Capture triggers
External event filtering (Out 2 channel only)
MS32290V1

By default, the auxiliary outputs are copies of outputs Tx1 and Tx2. The exceptions are:

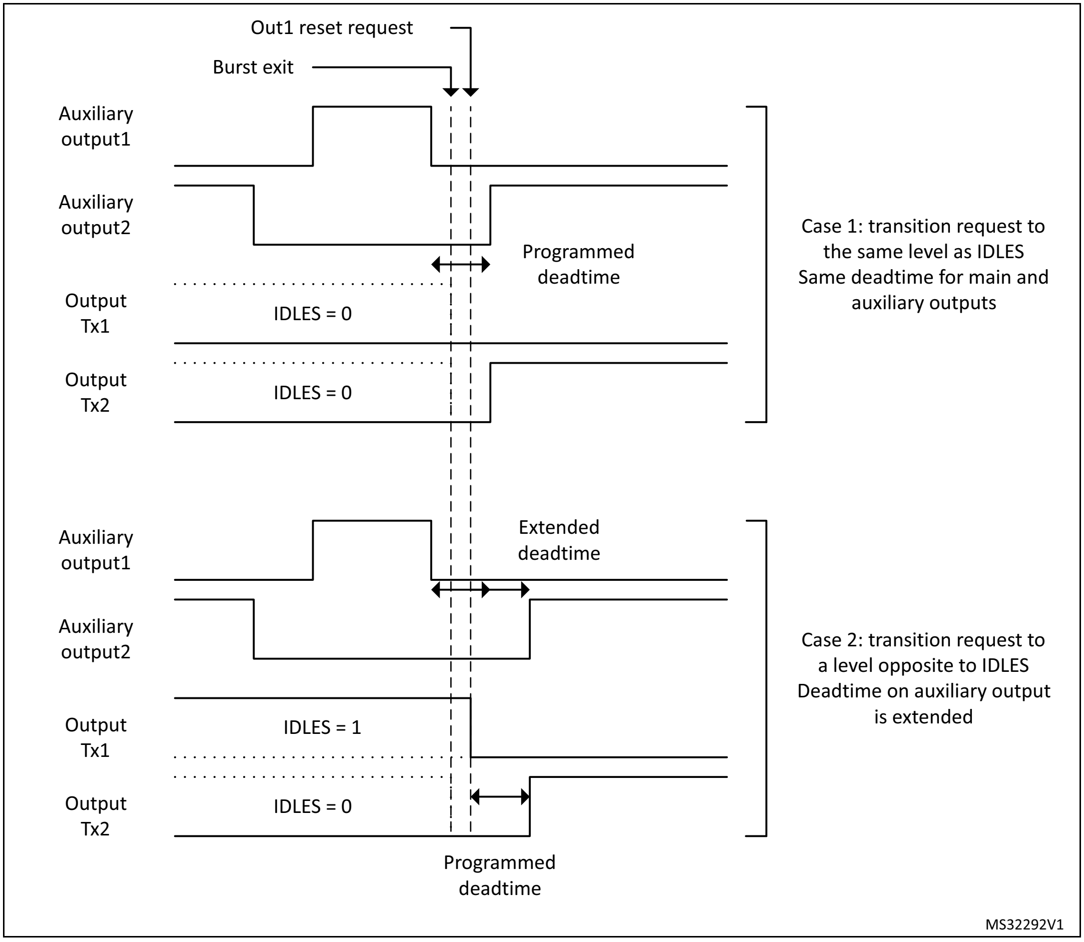
- The delayed idle and the balanced idle protections, when the deadtime is disabled (DTEN = 0). When the protection is triggered, the auxiliary outputs are maintained and follow the signal coming out of the crossbar. On the contrary, if the deadtime is enabled (DTEN = 1), both main and auxiliary outputs are forced to an inactive level.
- The burst mode (TCEN=1, IDLEMx=1); there are 2 cases:
  a) If DTEN=0 or DIDLx=0, the auxiliary outputs are not affected by the burst mode entry and continue to follow the reference signal coming out of the crossbar (see *Figure 293*).
  b) If the deadtime is enabled (DTEN=1) together with the delayed burst mode entry (DIDLx=1), the auxiliary outputs have the same behavior as the main outputs. They are forced to the IDLES level after a deadtime duration, then they keep this level during all the burst period. When the burst mode is terminated, the IDLES level is maintained until a transition occurs to the opposite level, similarly to the main output.

Figure 293. Auxiliary and main outputs during burst mode (DIDLx = 0)



The signal on the auxiliary output can be slightly distorted when exiting from the burst mode or when re-enabling the outputs after a delayed protection, if this happens during a deadtime. In this case, the deadtime applied to the auxiliary outputs is extended so that the deadtime on the main outputs is respected. *Figure 294* gives some examples.

Figure 294. Deadtime distortion on auxiliary output when exiting burst mode



21.3.17 Synchronizing the HRTIM with other timers or HRTIM instances

The HRTIM provides options for synchronizing multiple HRTIM instances, as a master unit (generating a synchronization signal) or as a slave (waiting for a trigger to be synchronized). This feature can also be used to synchronize the HRTIM with other timers, either external or on-chip. The synchronization circuitry is controlled inside the master timer.

Synchronization output

This section explains how the HRTIM must be configured to synchronize external resources and act as a master unit.

Four events can be selected as the source to be sent to the synchronization output. This is done using SYNCSRC[1:0] bits in the HRTIM_MCR register, as follows:

- 00: Master timer Start
  This event is generated when MCEN bit is set or when the timer is re-started after having reached the period value in single-shot mode. It is also generated on a reset which occurs during the counting (when CONT or RETRIG bits are set).
- 01: Master timer Compare 1 event
- 10: Timer A start
  This event is generated when TACEN bit is set or when the counter is reset and re-starts counting in response to this reset. The following counter reset events are not propagated to the synchronization output: counter roll-over in continuous mode, and discarded reset request in single-shot non-retriggerable mode. The reset is only taken into account when it occurs during the counting (CONT or RETRIG bits are set).
- 11: Timer A Compare 1 event

SYNCOUT[1:0] bits in the HRTIM_MCR register specify how the synchronization event is generated.

The synchronization pulses are generated on the HRTIM_SCOUT output pin, with SYNCOUT[1:0] = 1x. SYNCOUT[0] bit specifies the polarity of the synchronization signal. If SYNCOUT[0] = 0, the HRTIM_SCOUT pin has a low idle level and issues a positive pulse of 16 f_{HRTIM} clock cycles length for the synchronization). If SYNCOUT[0] = 1, the idle level is high and a negative pulse is generated.

Note: The synchronization pulse is followed by an idle level of 16 f_{HRTIM} clock cycles during which any new synchronization request is discarded. Consequently, the maximum synchronization frequency is f_{HRTIM}/32.

The idle level on the HRTIM_SCOUT pin is applied as soon as the SYNCOUT[1:0] bits are enabled (i.e. the bitfield value is different from 00).

The synchronization output initialization procedure must be done prior to the configuration of the MCU outputs and counter enable, in the following order:

1. SYNCOUT[1:0] and SYNCSRC[1:0] bitfield configuration in HRTIM_MCR
2. HRTIM_SCOUT pin configuration (see the General-purpose I/Os section)
3. Master or Timer A counter enable (MCEN or TACEN bit set)

When the synchronization input mode is enabled and starts the counter (using SYNCSTRTM/SYNCSTRTx bits) simultaneously with the synchronization output mode (SYNCSRC[1:0] = 00 or 10), the output pulse is generated only when the counter is starting or is reset while running. Any reset request clearing the counter without causing it to start will not affect the synchronization output.

Synchronization input

The HRTIM can be synchronized by external sources, as per the programming of the SYNCIN[1:0] bits in the HRTIM_MCR register:

- 00: synchronization input is disabled
- 01: reserved configuration
- 10: the on-chip TIM1 general purpose timer (TIM1 TRGO output)
- 11: a positive pulse on the HRTIM_SCIN input pin

This bitfield cannot be changed once the destination timer (master timer or timing unit) is enabled (MCEN and/or TxCEN bit set).

The HRTIM_SCIN input is rising-edge sensitive. The timer behavior is defined with the following bits present in HRTIM_MCR and HRTIM_TIMxCR registers (see *Table 101* for details):

- Synchronous start: the incoming signal starts the timer's counter (SYNCSTRTM and/or SYNCSTRTx bits set). TxCEN (MCEN) bits must be set to have the timer enabled and the counter ready to start. In continuous mode, the counter will not start until the synchronization signal is received.
- Synchronous reset: the incoming signal resets the counter (SYNCRSTM and/or SYNCRSTx bits set). This event decrements the repetition counter as any other reset event.

The synchronization events are taken into account only once the related counters are enabled (MCEN or TxCEN bit set). A synchronization request triggers a SYNC interrupt.

Note: A synchronized start event resets the counter if the current counter value is above the active period value.

The effect of the synchronization event depends on the timer operating mode, as summarized in *Table 101*.

Table 101. Effect of sync event vs timer operating modes

Operating mode	SYNC RSTx	SYNC STRTx	Behavior following a SYNC reset or start event
Single-shot non-retriggerable	0	1	Start events are taken into account when the counter is stopped and: – once the MCEN or TxCEN bits are set – once the period has been reached. A start occurring when the counter is stopped at the period value resets the counter. A reset request clears the counter but does not start it (the counter can solely be re-started with the synchronization). Any reset occurring during the counting is ignored (as during regular non-retriggerable mode).
	1	X	Reset events are starting the timer counting. They are taken into account only if the counter is stopped and: – once the MCEN or TxCEN bits are set – once the period has been reached. When multiple reset requests are selected (from HRTIM_SCIN and from internal events), only the first arriving request is taken into account.

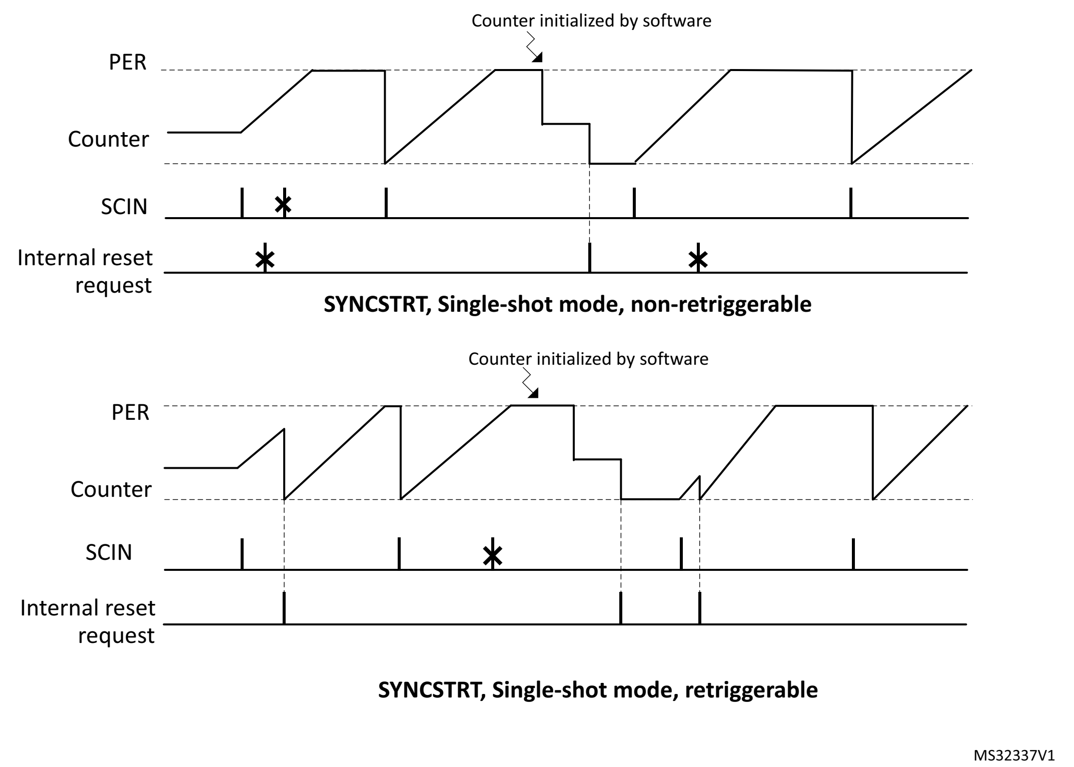
Table 101. Effect of sync event vs timer operating modes (continued)

Operating mode	SYNC RSTx	SYNC STRTx	Behavior following a SYNC reset or start event
Single-shot retriggerable	0	1	The counter start is effective only if the counter is not started or period is elapsed. Any synchronization event occurring after counter start has no effect. A start occurring when the counter is stopped at the period value resets the counter. A reset request clears the counter but does not start it (the counter can solely be started by the synchronization). A reset occurring during counting is taken into account (as during regular retriggerable mode).
	1	X	The reset from HRTIM_SCIN is taken into account as any HRTIM counter reset from internal events and is starting or re-starting the timer counting. When multiple reset requests are selected, the first arriving request is taken into account.
Continuous mode	0	1	The timer is enabled (MCEN or TxCEN bit set) and is waiting for the synchronization event to start the counter. Any synchronization event occurring after the counter start has no effect (the counter can solely be started by the synchronization). A reset request clears the counter but does not start it.
	1	X	The reset from HRTIM_SCIN is taken into account as any HRTIM counter reset from internal events and is starting or re-starting the timer counting. When multiple reset requests are selected, the first arriving request is taken into account.

When a synchronization reset event occurs within the same f_{HRTIM} *clock cycle as the period event, this reset is postponed to a programmed period event (since both events are causing a counter roll-over). This applies only when the high-resolution is active* (CKPSC[2:0] < 5).

Figure 295 presents how the synchronized start is done in single-shot mode.

Figure 295. Counter behavior in synchronized start mode


Counter initialized by software
PER
Counter
SCIN
Internal reset request
SYNCSTRT, Single-shot mode, non-retriggerable
Counter initialized by software
PER
Counter
SCIN
Internal reset request
SYNCSTRT, Single-shot mode, retriggerable
MS32337V1

21.3.18 ADC triggers

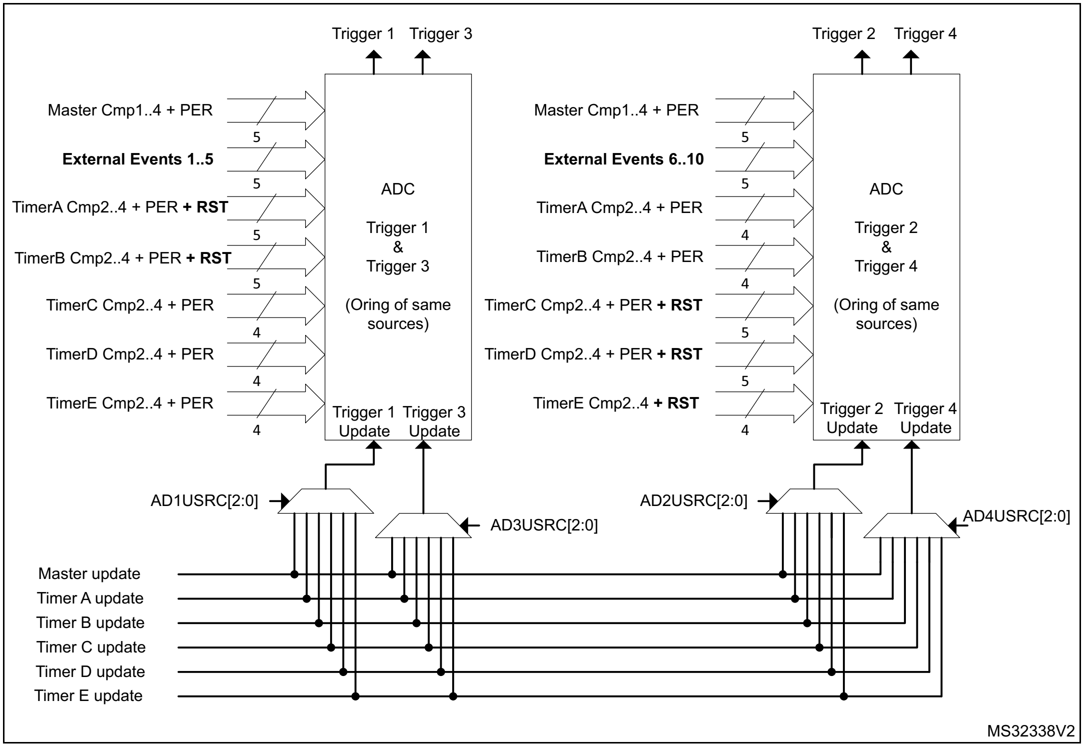
The ADCs can be triggered by the master and the 5 timing units.

4 independent triggers are available to start both the regular and the injected sequencers of the 2 ADCs. Up to 32 events can be combined (ORed) for each trigger output, in registers HRTIM_ADC1R to HRTIM_ADC4R, as shown on *Figure 296*. Triggers 1/3 and 2/4 are using the same source set.

The external events can be used as a trigger. They are taken right after the conditioning defined in HRTIM_EECRx registers, and are not depending on EEFxR1 and EEFxR2 register settings.

Multiple triggering is possible within a single switching period by selecting several sources simultaneously. A typical use case is for a non-overlapping multiphase converter, where all phases can be sampled in a row using a single ADC trigger output.

Figure 296. ADC trigger selection overview



HRTIM_ADC1R to HRTIM_ADC4R registers are preloaded and can be updated synchronously with the timer they are related to. The update source is defined with ADxUSRC[2:0] bits in the HRTIM_CR1 register.

For instance, if ADC trigger 1 outputs Timer A CMP2 events (HRTIM_ADC1R = 0x0000 0400), HRTIM_ADC1R will be typically updated simultaneously with Timer A (AD1USRC[2:0] = 001).

When the preload is disabled (PREEN bit reset) in the source timer, the HRTIM_ADCxR registers are not preloaded either: a write access will result in an immediate update of the trigger source.

21.3.19 DAC triggers

The HRTIMER allows to have the embedded DACs updated synchronously with the timer updates.

The update events from the master timer and the timer units can generate DAC update triggers on any of the 3 DACtrigOutx outputs.

Note: Each timer has its own DAC-related control register.

DACSYNC[1:0] bits of the HRTIM_MCR and HRTIM_TIMxCR registers are programmed as follows:

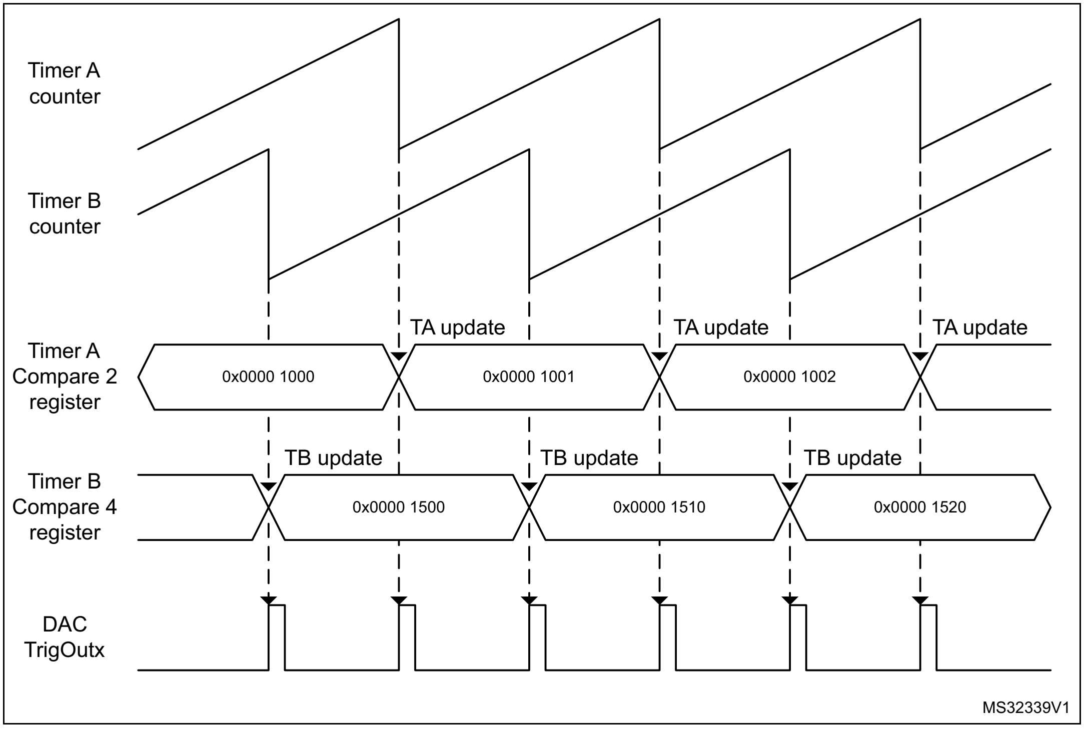
- 00: No update generated
- 01: Update generated on DACtrigOut1
- 10: Update generated on DACtrigOut2
- 11: Update generated on DACtrigOut3

An output pulse of 32 f_{HRTIM} clock periods is generated on the DACtrigOutx output.

Note: The synchronization pulse is followed by an idle level of 32 APB clock cycles during which any new DAC update request is ignored. Consequently, the maximum synchronization frequency is f_{apb}/64.

When DACSYNC[1:0] bits are enabled in multiple timers, the DACtrigOutx output will consist of an OR of all timers' update events. For instance, if DACSYNC = 1 in timer A and in timer B, the update event in timer A will be ORed with the update event in timer B to generate a DAC update trigger on the corresponding DACtrigOutx output, as shown on *Figure 297*.

Figure 297. Combining several updates on a single DACtrigOutx output



DACtrigOutx pins are connected to the DACs as follows:

- DACtrigOut1: DAC1_CH1 trigger input 3 (TSEL1[2:0] = 011 in DAC_CR of DAC1 peripheral)
- DACtrigOut2: DAC1_CH2 trigger input 5 (TSEL1[2:0] = 101 in DAC_CR of DAC1 peripheral and DAC1_TRIG3_RMP bit set in SYSCFG_CFGR2)
- DACTrigOut3: DAC2_CH1 trigger input 5 (TSEL1[2:0] = 101 in DAC_CR of DAC2 peripheral)

21.3.20 HRTIM Interrupts

7 interrupts can be generated by the master timer:

- Master timer registers update
- Synchronization event received
- Master timer repetition event
- Master Compare 1 to 4 event

14 interrupts can be generated by each timing unit:

- Delayed protection triggered
- Counter reset or roll-over event
- Output 1 and output 2 reset (transition active to inactive)
- Output 1 and output 2 set (transition inactive to active)
- Capture 1 and 2 events
- Timing unit registers update
- Repetition event
- Compare 1 to 4 event

8 global interrupts are generated for the whole HRTIM:

- System fault and Fault 1 to 5 (regardless of the timing unit attribution)
- DLL calibration done
- Burst mode period completed

The interrupt requests are grouped in 7 vectors as follows:

- IRQ1: Master timer interrupts (Master Update, Sync Input, Repetition, MCMP1..4) and global interrupt except faults (Burst mode period and DLL ready interrupts)
- IRQ2: TIMA interrupts
- IRQ3: TIMB interrupts
- IRQ4: TIMC interrupts
- IRQ5: TIMD interrupts
- IRQ6: TIME interrupts
- IRQ7: Dedicated vector all fault interrupts to allow high-priority interrupt handling

Table 102 is a summary of the interrupt requests, their mapping and associated control, and status bits.

Table 102. HRTIM interrupt summary

Interrupt vector	Interrupt event	Event flag	Enable control bit	Flag clearing bit
IRQ1	Burst mode period completed	BMPER	BMPERIE	BMPERC
	DLL calibration done	DLLRDY	DLLRDYIE	DLLRDYC
	Master timer registers update	MUPD	MUPDIE	MUPDC
	Synchronization event received	SYNC	SYNCIE	SYNCC
	Master timer repetition event	MREP	MREPIE	MREPC
	Master Compare 1 to 4 event	MCMP1	MCMP1IE	MCP1C
		MCMP2	MCMP2IE	MCP2C
		MCMP3	MCMP3IE	MCP3C
		MCMP4	MCMP4IE	MCP4C
IRQ2 IRQ3 IRQ4 IRQ5 IRQ6	Delayed protection triggered	DLYPRT	DLYPRTIE	DLYPRTC
	Counter reset or roll-over event	RST	RSTIE	RSTC
	Output 1 and output 2 reset (transition active to inactive)	RSTx1	RSTx1IE	RSTx1C
		RSTx2	RSTx2IE	RSTx2C
	Output 1 and output 2 set (transition inactive to active)	SETx1	SETx1IE	SETx1C
		SETx2	SETx2IE	SETx2C
	Capture 1 and 2 events	CPT1	CPT1IE	CPT1C
		CPT2	CPT2IE	CPT2C
	Timing unit registers update	UPD	UPDIE	UPDC
	Repetition event	REP	REPIE	REPC
	Compare 1 to 4 event	CMP1	CMP1IE	CMP1C
		CMP2	CMP2IE	CMP2C
		CMP3	CMP3IE	CMP3C
		CMP4	CMP4IE	CMP4C
IRQ7	System fault	SYSFLT	SYSFLTIE	SYSFLTC
	Fault 1 to 5	FLT1	FLT1IE	FLT1C
		FLT2	FLT2IE	FLT2C
		FLT3	FLT3IE	FLT3C
		FLT4	FLT4IE	FLT4C
		FLT5	FLT5IE	FLT5C

21.3.21 DMA

Most of the events able to generate an interrupt can also generate a DMA request, even both simultaneously. Each timer (master, TIMA...E) has its own DMA enable register.

The individual DMA requests are ORed into 6 channels as follows:

- 1 channel for the master timer
- 1 channel per timing unit

Note: Before disabling a DMA channel (DMA enable bit reset in TIMxDIER), it is necessary to disable first the DMA controller.

Table 103 is a summary of the events with their associated DMA enable bits.

Table 103. HRTIM DMA request summary

DMA Channel	Event	DMA capable	DMA enable bit
Master timer: Channel 2	Burst mode period completed	No	N/A
	DLL calibration done	No	N/A
	Master timer registers update	Yes	MUPDDE
	Synchronization event received	Yes	SYNCDE
	Master timer repetition event	Yes	MREPDE
	Master Compare 1 to 4 event	Yes	MCMP1DE
		Yes	MCMP2DE
		Yes	MCMP3DE
		Yes	MCMP4DE
Timer A: Channel 3 Timer B: Channel 4 Timer C: Channel 5 Timer D: Channel 6 Timer E: Channel 7	Delayed protection triggered	Yes	DLYPRTDE
	Counter reset or roll-over event	Yes	RSTDE
	Output 1 and output 2 reset (transition active to inactive)	Yes	RSTx1DE
		Yes	RSTx2DE
	Output 1 and output 2 set (transition inactive to active)	Yes	SETx1DE
		Yes	SETx2DE
	Capture 1 and 2 events	Yes	CPT1DE
		Yes	CPT2DE
	Timing unit registers update	Yes	UPDDE
	Repetition event	Yes	REPDE
	Compare 1 to 4 event	Yes	CMP1DE
		Yes	CMP2DE
		Yes	CMP3DE
		Yes	CMP4DE

Table 103. HRTIM DMA request summary (continued)

DMA Channel	Event	DMA capable	DMA enable bit
N/A	System fault	No	N/A
	Fault 1 to 5	No	N/A
	Burst mode period completed	No	N/A
	DLL calibration done	No	N/A

Burst DMA transfers

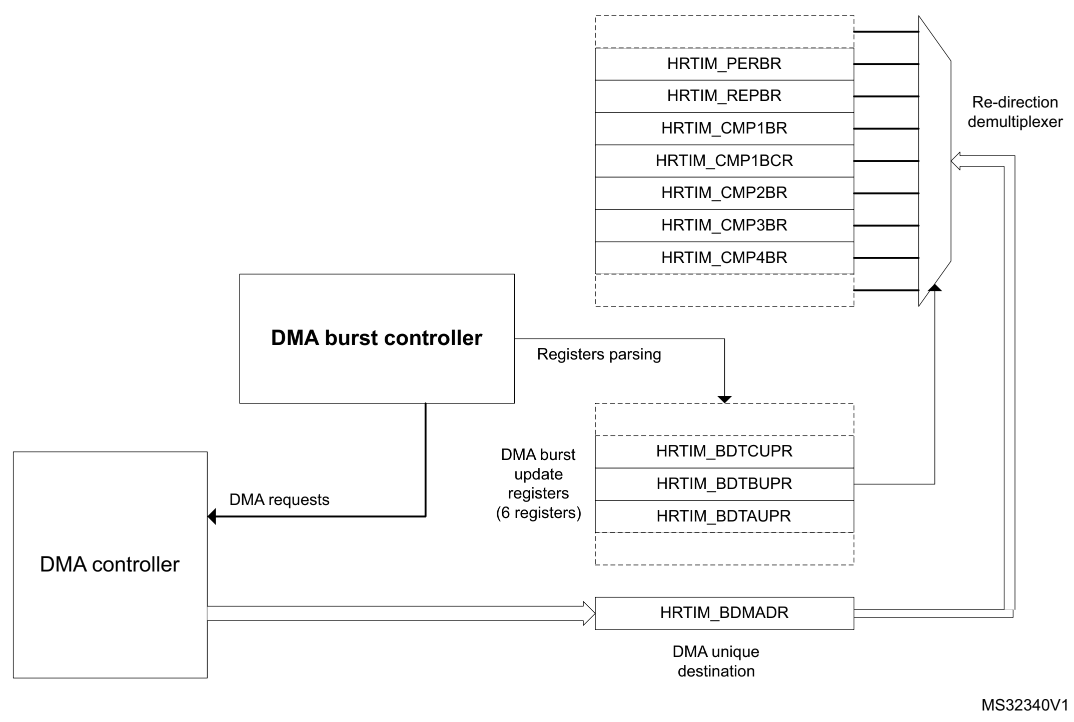
In addition to the standard DMA requests, the HRTIM features a DMA burst controller to have multiple registers updated with a single DMA request. This allows to:

- update multiple data registers with one DMA channel only,
- reprogram dynamically one or several timing units, for converters using multiple timer outputs.

The burst DMA feature is only available for one DMA channel, but any of the 6 channels can be selected for burst DMA transfers.

The principle is to program which registers are to be written by DMA. The master timer and TIMA..E have the burst DMA update register, where most of their control and data registers are associated with a selection bit: HRTIM_BDMUPR, HRTIM_BDTAUPR to HRTIM_BDTEUPR (this is applicable only for registers with write accesses). A redirection mechanism allows to forward the DMA write accesses to the HRTIM registers automatically, as shown on *Figure 298*.

Figure 298. DMA burst overview



When the DMA trigger occurs, the HRTIM generates multiple 32-bit DMA requests and parses the update register. If the control bit is set, the write access is redirected to the associated register. If the bit is reset, the register update is skipped and the register parsing is resumed until a new bit set is detected, to trigger a new request. Once the 6 update registers (HRTIM_BDMUPR, 5x HRTIM_BDTxUPR) are parsed, the burst is completed and the system is ready for another DMA trigger (see the flowchart on *Figure 299*).

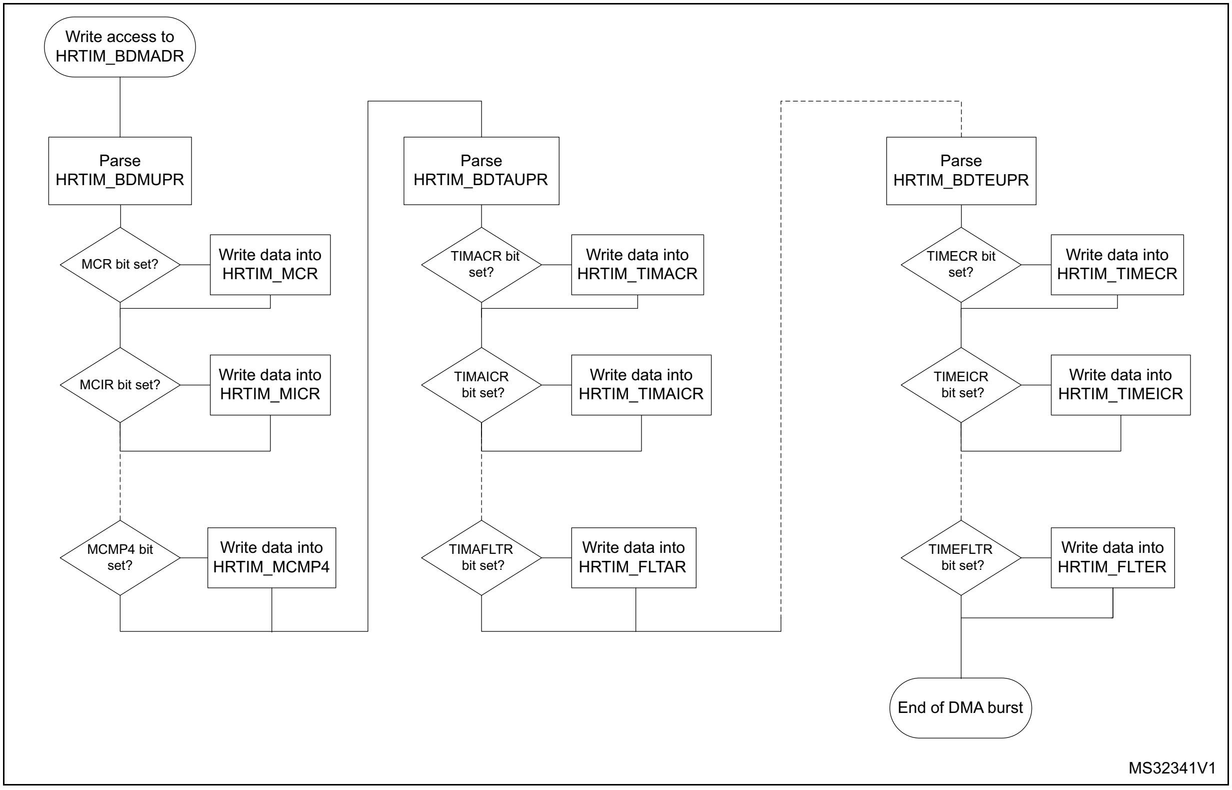
Note: Any trigger occurring while the burst is on-going is discarded, except if it occurs during the very last data transfer.

The burst DMA mode is permanently enabled (there is no enable bit). A burst DMA operation is started by the first write access into the HRTIM_BDMADR register.

It is only necessary to have the DMA controller pointing to the HRTIM_BDMADR register as the destination, in the memory, to the peripheral configuration with the peripheral increment mode disabled (the HRTIM handles internally the data re-routing to the final destination register).

To re-initialize the burst DMA mode if it was interrupted during a transaction, it is necessary to write at least to one of the 6 update registers.

Figure 299. Burst DMA operation flowchart


Write access to HRTIM_BDMADR
Parse HRTIM_BDMUPR
MCR bit set?
Write data into HRTIM_MCR
MCIR bit set?
Write data into HRTIM_MICR
MCMP4 bit set?
Write data into HRTIM_MCMP4
Parse HRTIM_BDTAUPR
TIMACR bit set?
Write data into HRTIM_TIMACR
TIMAICR bit set?
Write data into HRTIM_TIMAICR
TIMAFLTR bit set?
Write data into HRTIM_FLTAR
Parse HRTIM_BDTEUPR
TIMECR bit set?
Write data into HRTIM_TIMECR
TIMEICR bit set?
Write data into HRTIM_TIMEICR
TIMEFLTR bit set?
Write data into HRTIM_FLTER
End of DMA burst
MS32341V1

Several options are available once the DMA burst is completed, depending on the register update strategy.

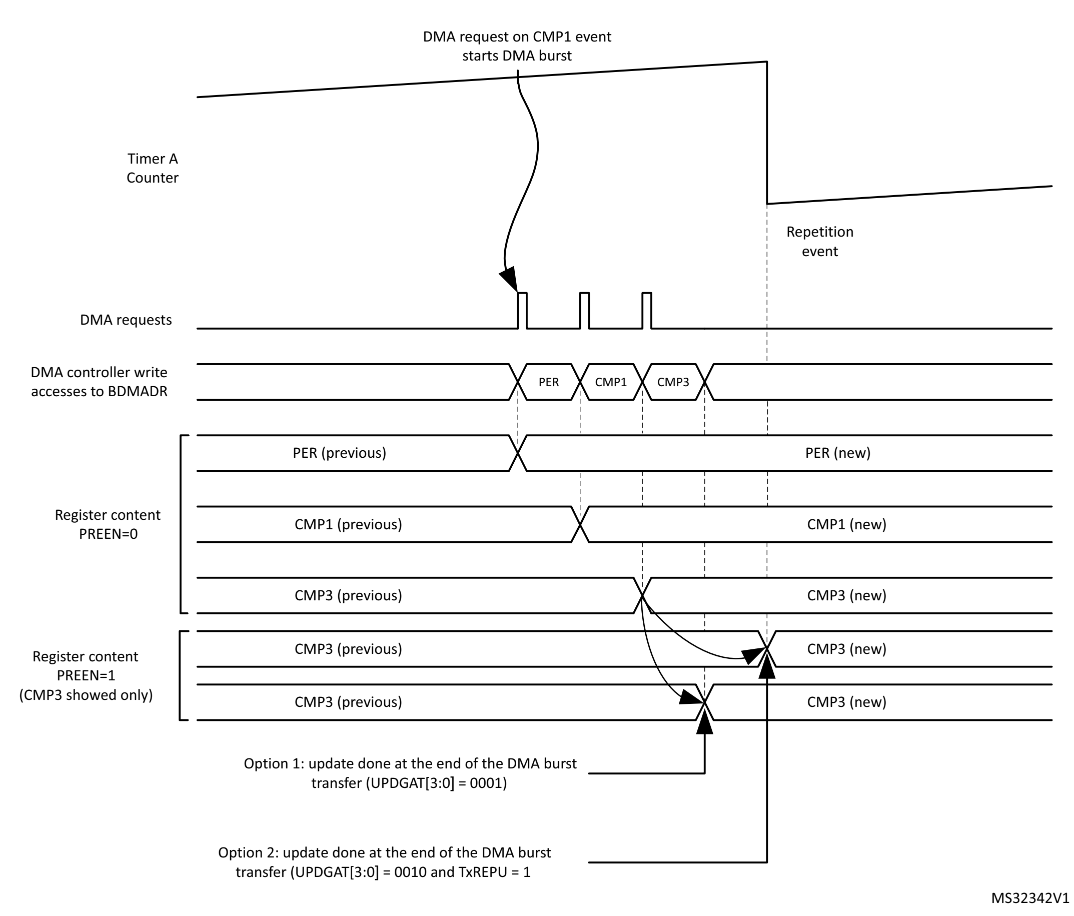
If the PREEN bit is reset (preload disabled), the value written by the DMA is immediately transferred into the active register and the registers are updated sequentially, following the DMA transaction pace.

When the preload is enabled (PREEN bit set), there are 3 use cases:

1. The update is done independently from DMA burst transfers (UPDGAT[3:0] = 0000 in HRTIM_TIMxCR and BRSTDMA[1:0] = 00 in HRTIM_MCR). In this case, and if it is necessary to have all transferred data taken into account simultaneously, the user must check that the DMA burst is completed before the update event takes place. On the contrary, if the update event happens while the DMA transfer is on-going, only part of the registers will be loaded and the complete register update will require 2 consecutive update events.
2. The update is done when the DMA burst transfer is completed (UPDGAT[3:0] = 0000 in HRTIM_TIMxCR and BRSTDMA[1:0] = 01 in HRTIM_MCR). This mode guarantees that all new register values are transferred simultaneously. This is done independently from the counter value and can be combined with regular update events, if necessary (for instance, an update on a counter reset when TxRSTU is set).
3. The update is done on the update event following the DMA burst transfer completion (UPDGAT[3:0] = 0010 in HRTIM_TIMxCR and BRSTDMA[1:0] = 10 in HRTIM_MCR). This mode guarantees both a coherent update of all transferred data and the synchronization with regular update events, with the timer counter. In this case, if a regular update request occurs while the transfer is on-going, it will be discarded and the effective update will happen on the next coming update request.

The chronogram on *Figure 300* presents the active register content for 3 cases: PREEN=0, UPDGAT[3:0] = 0001 and UPDGAT[3:0] = 0001 (when PREEN = 1).

Figure 300. Registers update following DMA burst transfer



21.3.22 HRTIM initialization

This section describes the recommended HRTIM initialization procedure, including other related MCU peripherals.

The HRTIM clock source must be enabled in the Reset and Clock control unit (RCC), while respecting the f_{HRTIM} range for the DLL lock.

The DLL calibration must be started by setting CAL bit in HRTIM_DLLCR register.

The HRTIM master and timing units can be started only once the high-resolution unit is ready. This is indicated by the DLLRDY flag set. The DLLRDY flag can be polled before resuming the initialization or the calibration can run in background while other registers of the HRTIM or other MCU peripherals are initialized. In this case, the DLLRDY flag must be checked before starting the counters (an end-of-calibration interrupt can be issued if necessary, enabled with DLLRDYIE flag in HRTIM_IER). Once the DLL calibration is done, CALEN bit must be set to have it done periodically and compensate for potential voltage and temperature drifts. The calibration periodicity is defined using the CALRTE[1:0] bitfield in the HRTIM_DLLCR register.

The HRTIM control registers can be initialized as per the power converter topology and the timing units use case. All inputs have to be configured (source, polarity, edge-sensitivity).

The HRTIM outputs must be set up eventually, with the following sequence:

- the polarity must be defined using POLx bits in HRTIM_OUTxR
- the FAULT and IDLE states must be configured using FAULTx[1:0] and IDLESx bits in HRTIM_OUTxR

The HRTIM outputs are ready to be connected to the MCU I/Os. In the GPIO controller, the selected HRTIM I/Os have to be configured as per the alternate function mapping table in the product datasheet.

From this point on, the HRTIM controls the outputs, which are in the IDLE state.

The outputs are configured in RUN mode by setting TxyOEN bits in the HRTIM_OENR register. The 2 outputs are in the inactive state until the first valid set/reset event in RUN mode. Any output set/reset event (except software requests using SST, SRT) are ignored as long as TxCEN bit is reset, as well as burst mode requests (IDLEM bit value is ignored). Similarly, any counter reset request coming from the burst mode controller is ignored (if TxBM bit is set).

Note: When the deadtime insertion is enabled (DTEN bit set), it is necessary to force the output state by software, using SST and RST bits, to have the outputs in a complementary state as soon as the RUN mode is entered.

The HRTIM operation can eventually be started by setting TxCEN or MCEN bits in HRTIM_MCR.

If the HRTIM peripheral is reset with the Reset and Clock Controller, the output control is released to the GPIO controller and the outputs are tri stated.

21.3.23 Debug

When a microcontroller enters the debug mode (Cortex®-M4 core halted), the TIMx counter either continues to work normally or stops, depending on DBG_HRTIM_STOP configuration bit in DBG module:

- DBG_HRTIM_STOP = 0: no behavior change, the HRTIM continues to operate.
- DBG_HRTIM_STOP = 1: all HRTIM timers, including the master, are stopped. The outputs in RUN mode enter the FAULT state if FAULTx[1:0] = 01,10,11, or keep their current state if FAULTx[1:0] = 00. The outputs in idle state are maintained in this state. This is permanently maintained even if the MCU exits the halt mode. This allows to maintain a safe state during the execution stepping. The outputs can be enabled again by settings TxyOEN bit (requires the use of the debugger).

Timer behavior during MCU halt when DBG_HRTIM_STOP = 1

The set/reset crossbar, the dead-time and push-pull unit, the idle/balanced fault detection and all the logic driving the normal output in RUN mode are not affected by debug. The output will keep on toggling internally, so as to retrieve regular signals of the outputs when TxyOEN will be set again (during or after the MCU halt). Associated triggers and filters are also following internal waveforms when the outputs are disabled.

FAULT inputs and events (any source) are enabled during the MCU halt.

Fault status bits can be set and TxyOEN bits reset during the MCU halt if a fault occurs at that time (TxyOEN and TxyODS are not affected by DBG_HRTIM_STOP bit state).

Synchronization, counter reset, start and reset-start events are discarded in debug mode, as well as capture events. This is to keep all related registers stable as long as the MCU is halted.

The counter stops counting when a breakpoint is reached. However, the counter enable signal is not reset; consequently no start event will be emitted when exiting from debug. All counter reset and capture triggers are disabled, as well as external events (ignored as long as the MCU is halted). The outputs SET and RST flags are frozen, except in case of forced software set/reset. A level-sensitive event is masked during the debug but will be active again as soon as the debug will be exited. For edge-sensitive events, if the signal is maintained active during the MCU halt, a new edge is not generated when exiting from debug.

The update events are discarded. This prevents any update trigger on UPD_EN[3:1] inputs. DMA triggers are disabled. The burst mode circuit is frozen: the triggers are ignored and the burst mode counter stopped.

DLL calibration is not blocked while the MCU is halted (the DLLRDY flag can be set).

21.4 Application use cases

21.4.1 Buck converter

Buck converters are of common use as step-down converters. The HRTIM can control up to 10 buck converters with 6 independent switching frequencies.

The converter usually operates at a fixed frequency and the Vin/Vout ratio depends on the duty cycle D applied to the power switch:.

$$V_{out} = D \times V_{in}$$

The topology is given on *Figure 301* with the connection to the ADC for voltage reading.

Figure 301. Buck converter topology

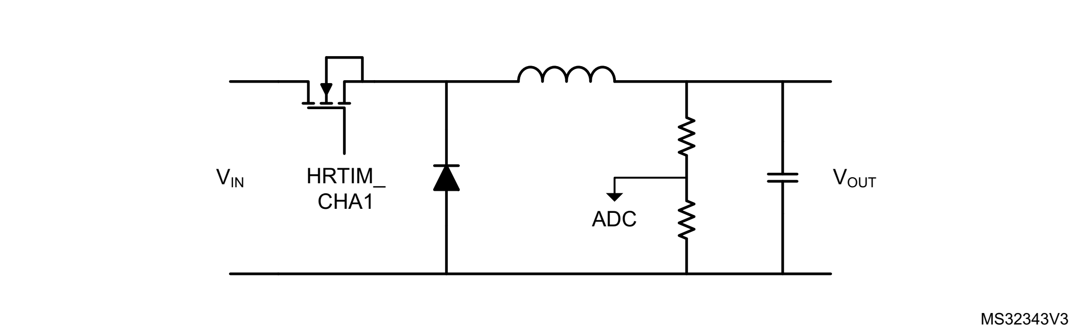
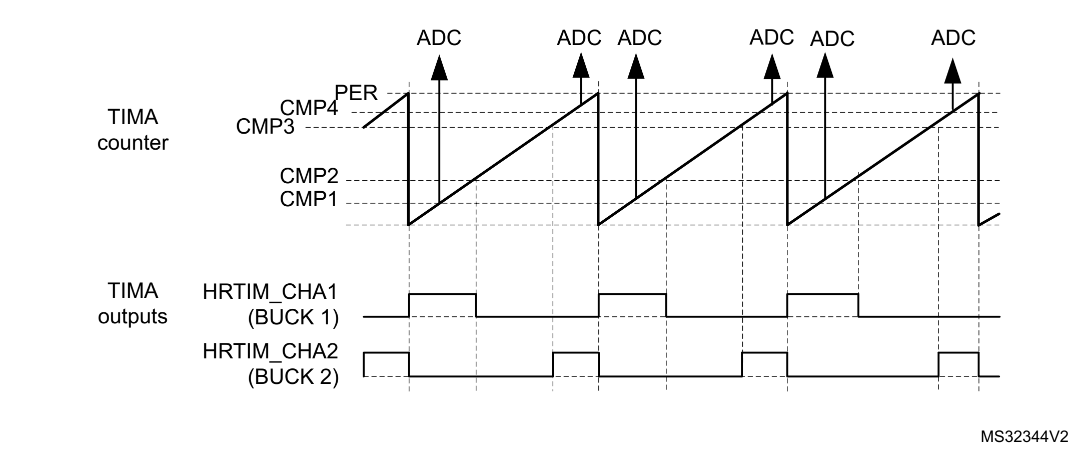


Figure 302 presents the management of two converters with identical frequency PWM signals. The outputs are defined as follows:

- HRTIM_CHA1 set on period, reset on CMP1
- HRTIM_CHA2 set on CMP3, reset on PER

The ADC is triggered twice per period, precisely in the middle of the ON time, using CMP2 and CMP4 events.

Figure 302. Dual Buck converter management

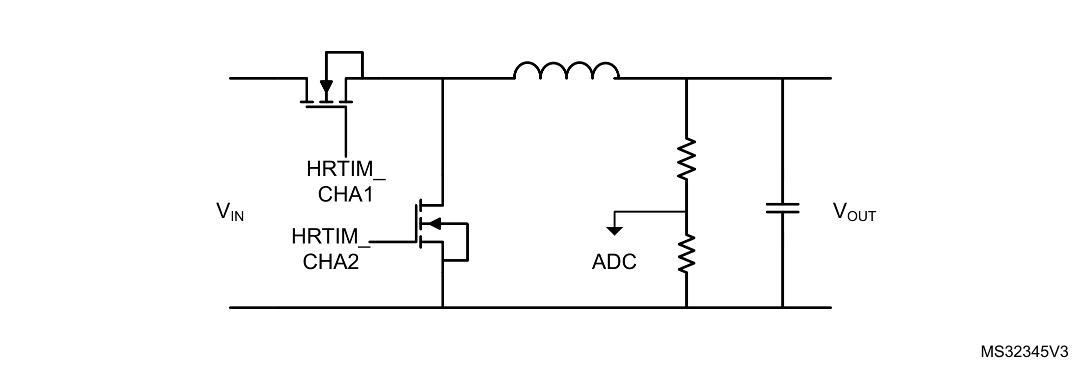


Timers A..E provide either 10 buck converters coupled by pairs (both with identical switching frequencies) or 6 completely independent converters (each of them having a different switching frequency), using the master timer as the 6th time base.

21.4.2 Buck converter with synchronous rectification

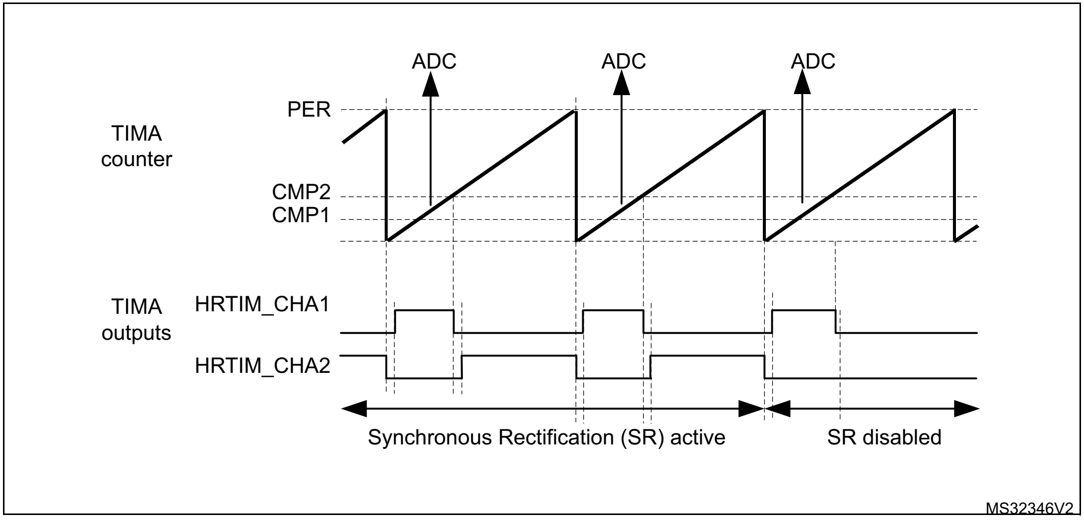
Synchronous rectification allows to minimize losses in buck converters, by means of a FET replacing the freewheeling diode. Synchronous rectification can be turned on or off on the fly depending on the output current level, as shown on *Figure 303*.

Figure 303. Synchronous rectification depending on output current



The main difference vs. a single-switch buck converter is the addition of a deadtime for an almost complementary waveform generation on HRTIM_CHA2, based on the reference waveform on HRTIM_CHA1 (see *Figure 304*).

Figure 304. Buck with synchronous rectification



21.4.3 Multiphase converters

Multiphase techniques can be applied to multiple power conversion topologies (buck, flyback). Their main benefits are:

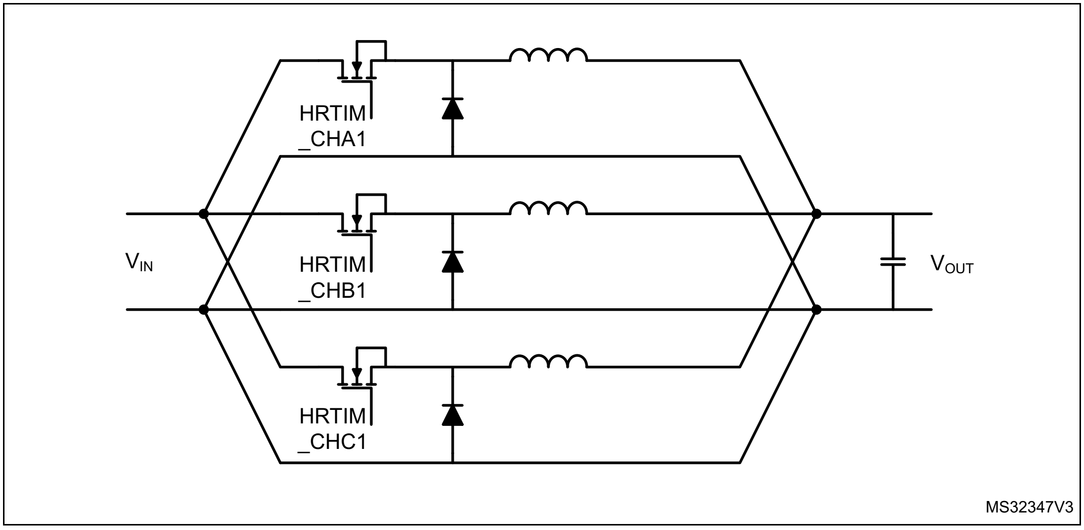
- Reduction of the current ripple on the input and output capacitors
- Reduced EMI
- Higher efficiency at light load by dynamically changing the number of phases (phase shedding)

The HRTIM is able to manage multiple converters. The number of converters that can be controlled depends on the topologies and resources used (including the ADC triggers):

- 5 buck converters with synchronous rectification (SR), using the master timer and the 5 timers
- 4 buck converters (without SR), using the master timer and 2 timers
- ...

Figure 306 presents the topology of a 3-phase interleaved buck converter.

Figure 305. 3-phase interleaved buck converter



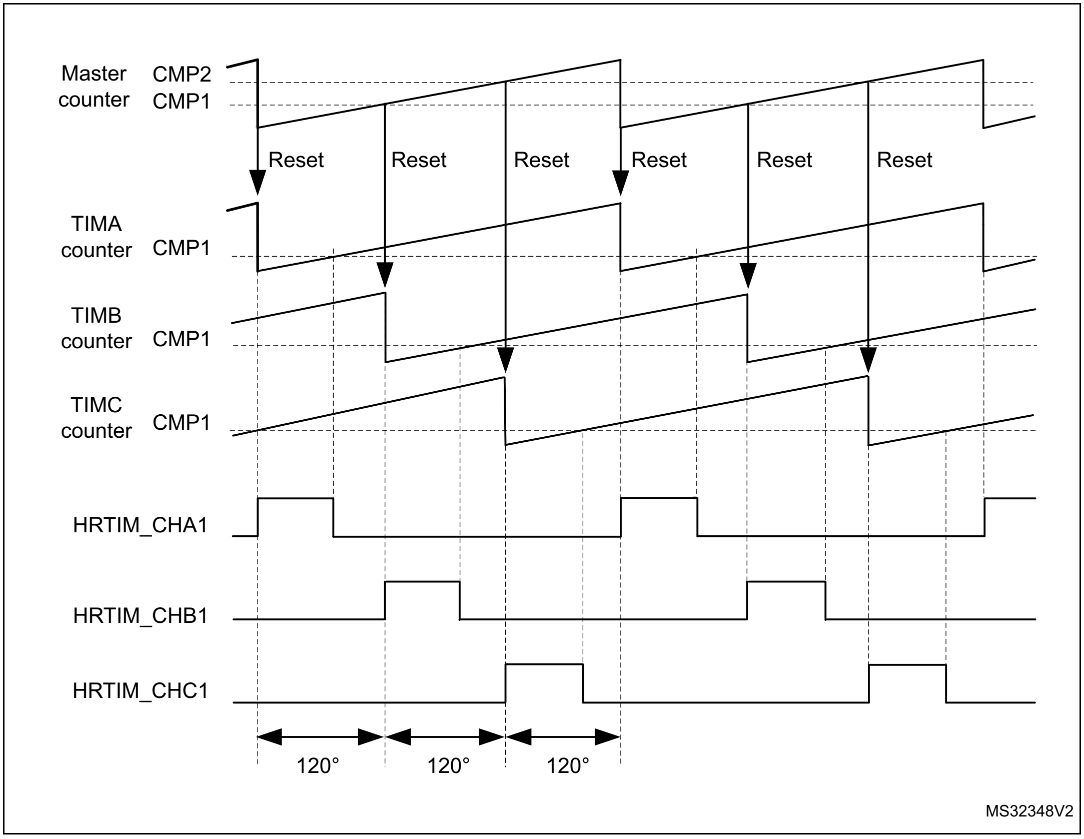
The master timer is responsible for the phase management: it defines the phase relationship between the converters by resetting the timers periodically. The phase-shift is 360° divided by the number of phases, 120° in the given example.

The duty cycle is then programmed into each of the timers. The outputs are defined as follows:

- HRTIM_CHA1 set on master timer period, reset on TACMP1
- HRTIM_CHB1 set on master timer MCMP1, reset on TBCMP1
- HRTIM_CHC1 set on master timer MCMP2, reset on TCCMP1

The ADC trigger can be generated on TxCMP2 compare event. Since all ADC trigger sources are phase-shifted because of the converter topology, it is possible to have all of them combined into a single ADC trigger to save ADC resources (for instance 1 ADC regular channel for the full multi-phase converter).

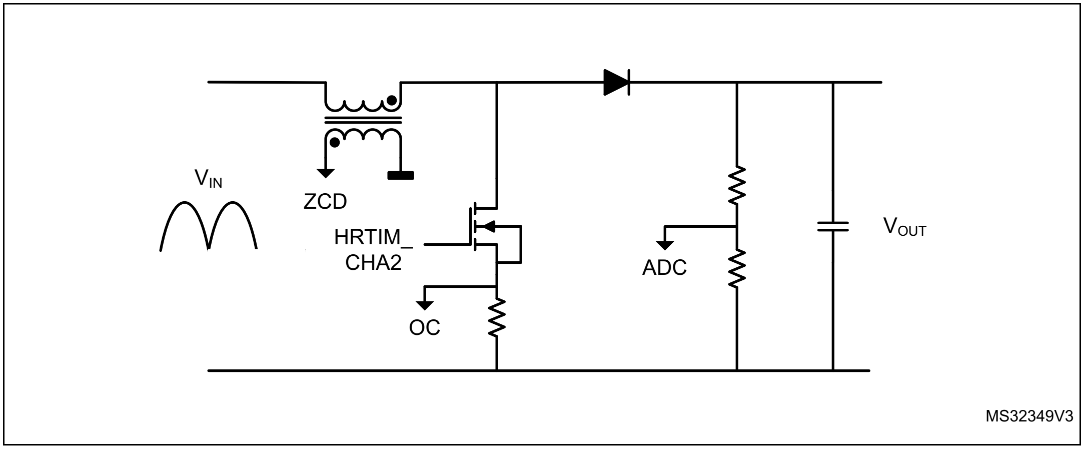
Figure 306. 3-phase interleaved buck converter control



21.4.4 Transition mode Power Factor Correction

The basic operating principle is to build up current into an inductor during a fixed Ton time. This current will then decay during the Toff time, and the period will be re-started when it becomes null. This is detected using a Zero Crossing Detection circuitry (ZCD), as shown on *Figure 307*. With a constant Ton time, the peak current value in the inductor is proportional to the rectified AC input voltage, which provides the power factor correction.

Figure 307. Transition mode PFC



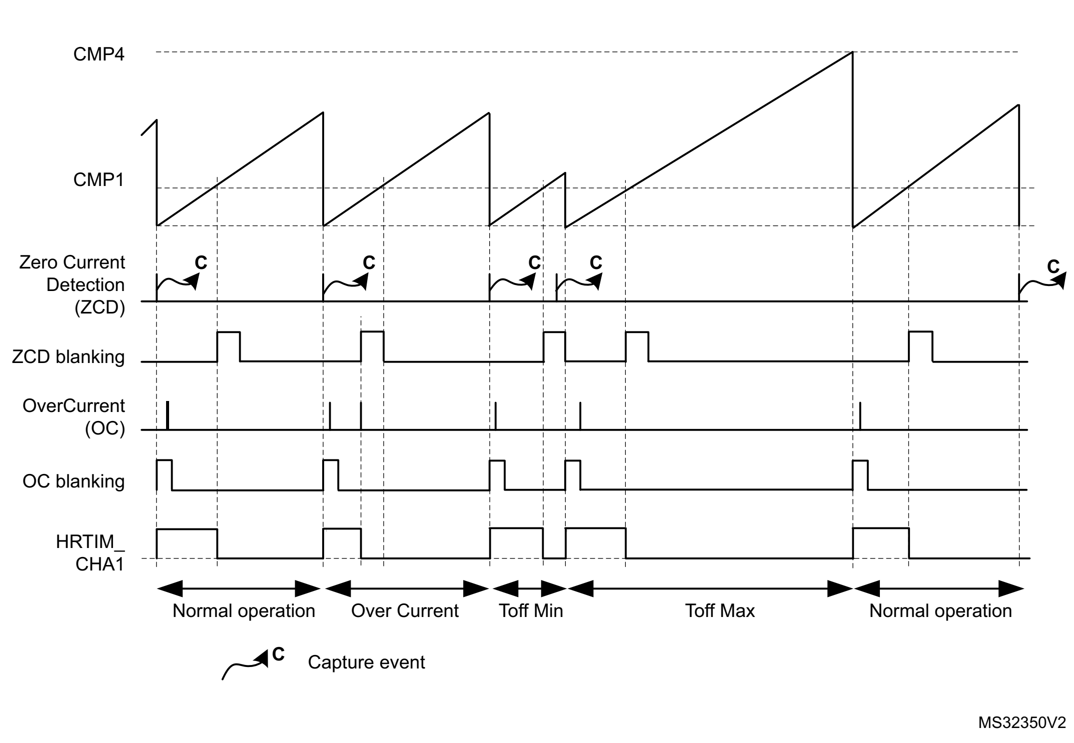
This converter is operating with a constant Ton time and a variable frequency due the Toff time variation (depending on the input voltage). It must also include some features to operate when no zero-crossing is detected, or to limit the Ton time in case of over-current (OC). The OC feedback is usually conditioned with the built-in comparator and routed onto an external event input.

Figure 308 presents the waveform during the various operating modes, with the following parameters defined:

- Ton Min: masks spurious overcurrent (freewheeling diode recovery current), represented as OC blanking
- Ton Max: practically, the converter set-point. It is defined by CMP1
- Toff Min: limits the frequency when the current limit is close to zero (demagnetization is very fast). It is defined with CMP2.
- Toff Max: prevents the system to be stuck if no ZCD occurs. It is defined with CMP4 in auto-delayed mode.

Both Toff values are auto-delayed since the value must be relative to the output falling edge.

Figure 308. Transition mode PFC waveforms



21.5 HRTIM registers

21.5.1 HRTIM Master Timer Control Register (HRTIM_MCR)

Address offset: 0x0000h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	Res.	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN
rw	rw	rw		rw	rw	rw				rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		Res.	Res.	HALF	RETRIG	CONT	CKPSC[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

Bits 31:30 **BRSTDMA[1:0]**: *Burst DMA Update*

These bits define how the update occurs relatively to a burst DMA transaction.

00: Update done independently from the DMA burst transfer completion

01: Update done when the DMA burst transfer is completed

10: Update done on master timer roll-over following a DMA burst transfer completion. This mode only works in continuous mode.

11: reserved

Bit 29 **MREPU**: Master *Timer Repetition update*

This bit defines whether an update occurs when the master timer repetition period is completed (either due to roll-over or reset events). MREPU can be set only if BRSTDMA[1:0] = 00 or 01.

0: Update on repetition disabled

1: Update on repetition enabled

Bit 28 Reserved, must be kept at reset value.

Bit 27 **PREEN**: *Preload enable*

This bit enables the registers preload mechanism and defines whether the write accesses to the memory mapped registers are done into HRTIM active or preload registers.

0: Preload disabled: the write access is directly done into the active register

1: Preload enabled: the write access is done into the preload register

Bits 26:25 **DACSYNC[1:0]** *DAC Synchronization*

A DAC synchronization event can be enabled and generated when the master timer update occurs. These bits are defining on which output the DAC synchronization is sent (refer to *Section 21.3.19: DAC triggers* for connections details).

00: No DAC trigger generated

01: Trigger generated on DACtrigOut1

10: Trigger generated on DACtrigOut2

11: Trigger generated on DACtrigOut3

Bits 24:22 Reserved, must be kept at reset value.

Bit 21 **TECEN**: *Timer E counter enable*

This bit starts the Timer E counter.

0: Timer E counter disabled

1: Timer E counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bit 20 **TDCEN**: *Timer D counter enable*

This bit starts the Timer D counter.
0: Timer D counter disabled
1: Timer D counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bit 19 **TCCEN**: *Timer C counter enable*

This bit starts the Timer C counter.
0: Timer C counter disabled
1: Timer C counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bit 18 **TBCEN**: *Timer B counter enable*

This bit starts the Timer B counter.
0: Timer B counter disabled
1: Timer B counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bit 17 **TACEN**: *Timer A counter enable*

This bit starts the Timer A counter.
0: Timer A counter disabled
1: Timer A counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bit 16 **MCEN**: *Master timer counter enable*

This bit starts the Master timer counter.
0: Master counter disabled
1: Master counter enabled

Note: This bit must not be changed within a minimum of 8 cycles of f_{HRTIM} clock.

Bits 15:14 **SYNCSRC[1:0]**: *Synchronization source*

These bits define the source and event to be sent on the synchronization outputs SYNCOUT[2:1]
00: Master timer Start
01: Master timer Compare 1 event
10: Timer A start/reset
11: Timer A Compare 1 event

Bits 13:12 **SYNCOUT[1:0]**: *Synchronization output*

These bits define the routing and conditioning of the synchronization output event.
00: disabled
01: Reserved.
10: Positive pulse on HRTIM_SCOUT output (16x f_{HRTIM} clock cycles)
11: Negative pulse on HRTIM_SCOUT output (16x f_{HRTIM} clock cycles)

Note: This bitfield must not be modified once the counter is enabled (TxCEN bit set)

Bit 11 **SYNCSTRTM**: *Synchronization Starts Master*

This bit enables the Master timer start when receiving a synchronization input event:
0: No effect on the Master timer
1: A synchronization input event starts the Master timer

Bit 10 **SYNCRSTM**: *Synchronization Resets Master*

This bit enables the Master timer reset when receiving a synchronization input event:
0: No effect on the Master timer
1: A synchronization input event resets the Master timer

Bits 9:8 **SYNCIN[1:0]** *Synchronization input*

These bits are defining the synchronization input source.
00: disabled. HRTIM is not synchronized and runs in standalone mode.
01: Reserved.
10: Internal event: the HRTIM is synchronized with the on-chip timer (see *Synchronization input*).
11: External event (input pin). A positive pulse on HRTIM_SCIN input triggers the HRTIM.

Note: This parameter cannot be changed once the impacted timers are enabled.

Bits 7:6 Reserved, must be kept at reset value.

Bit 5 **HALF**: *Half mode*

This bit enables the half duty-cycle mode: the HRTIM_MCMP1xR active register is automatically updated with HRTIM_MPER/2 value when HRTIM_MPER register is written.
0: Half mode disabled
1: Half mode enabled

Bit 4 **RETRIG**: *Re-triggerable mode*

This bit defines the behavior of the master timer counter in single-shot mode.
0: The timer is not re-triggerable: a counter reset can be done only if the counter is stopped (period elapsed)
1: The timer is re-triggerable: a counter reset is done whatever the counter state (running or stopped)

Bit 3 **CONT**: *Continuous mode*

0: The timer operates in single-shot mode and stops when it reaches the MPER value
1: The timer operates in continuous (free-running) mode and rolls over to zero when it reaches the MPER value

Bits 2:0 **CKPSC[2:0]**: *Clock prescaler*

These bits define the master timer high-resolution clock prescaler ratio.
The counter clock equivalent frequency ($f_{COUNTER}$) is equal to f_{HRCK} / $2^{CKPSC[2:0]}$.
The prescaling ratio cannot be modified once the timer is enabled.

21.5.2 HRTIM Master Timer Interrupt Status Register (HRTIM_MISR)

Address offset: 0x0004h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
									r	r	r	r	r	r	r

Bits 31:7 Reserved, must be kept at reset value.

Bit 6 **MUPD**: Master Update Interrupt Flag

This bit is set by hardware when the Master timer registers are updated.
0: No Master Update interrupt occurred
1: Master Update interrupt occurred

Bit 5 **SYNC**: Sync Input Interrupt Flag

This bit is set by hardware when a synchronization input event is received.
0: No Sync input interrupt occurred
1: Sync input interrupt occurred

Bit 4 **MREP**: Master Repetition Interrupt Flag

This bit is set by hardware when the Master timer repetition period has elapsed.
0: No Master Repetition interrupt occurred
1: Master Repetition interrupt occurred

Bit 3 **MCMP4**: Master Compare 4 Interrupt Flag

Refer to MCMP1 description

Bit 2 **MCMP3**: Master Compare 3 Interrupt Flag

Refer to MCMP1 description

Bit 1 **MCMP2**: Master Compare 2 Interrupt Flag

Refer to MCMP1 description

Bit 0 **MCMP1**: Master Compare 1 Interrupt Flag

This bit is set by hardware when the Master timer counter matches the value programmed in the master Compare 1 register.
0: No Master Compare 1 interrupt occurred
1: Master Compare 1 interrupt occurred

21.5.3 HRTIM Master Timer Interrupt Clear Register (HRTIM_MICR)

Address offset: 0x0008h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
									w	w	w	w	w	w	w

Bits 31:7 Reserved, must be kept at reset value.

Bit 6 **MUPDC**: Master update Interrupt flag clear

Writing 1 to this bit clears the MUPDC flag in HRTIM_MISR register

Bit 5 **SYNCC**: Sync Input Interrupt flag clear

Writing 1 to this bit clears the SYNC flag in HRTIM_MISR register

Bit 4 **MREPC**: Repetition Interrupt flag clear

Writing 1 to this bit clears the MREP flag in HRTIM_MISR register

Bit 3 **MCMP4C**: Master Compare 4 Interrupt flag clear

Writing 1 to this bit clears the MCMP4 flag in HRTIM_MISR register

Bit 2 **MCMP3C**: Master Compare 3 Interrupt flag clear

Writing 1 to this bit clears the MCMP3 flag in HRTIM_MISR register

Bit 1 **MCMP2C**: Master Compare 2 Interrupt flag clear

Writing 1 to this bit clears the MCMP2 flag in HRTIM_MISR register

Bit 0 **MCMP1C**: Master Compare 1 Interrupt flag clear

Writing 1 to this bit clears the MCMP1 flag in HRTIM_MISR register

21.5.4 HRTIM Master Timer DMA / Interrupt Enable Register (HRTIM_MDIER)

Address offset: 0x000Ch

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
									rw	rw	rw	rw	rw	rw	rw

Bits 31:23 Reserved, must be kept at reset value.

Bit 22 **MUPDDE**: Master Update DMA request Enable

This bit is set and cleared by software to enable/disable the Master update DMA requests.
0: Master update DMA request disabled
1: Master update DMA request enabled

Bit 21 **SYNCDE**: Sync Input DMA request Enable

This bit is set and cleared by software to enable/disable the Sync input DMA requests.
0: Sync input DMA request disabled
1: Sync input DMA request enabled

Bit 20 **MREPDE**: Master Repetition DMA request Enable

This bit is set and cleared by software to enable/disable the Master timer repetition DMA requests.
0: Repetition DMA request disabled
1: Repetition DMA request enabled

Bit 19 **MCMP4DE**: Master Compare 4 DMA request Enable

Refer to MCMP1DE description

Bit 18 **MCMP3DE**: Master Compare 3 DMA request Enable

Refer to MCMP1DE description

Bit 17 **MCMP2DE**: Master Compare 2 DMA request Enable

Refer to MCMP1DE description

Bit 16 **MCMP1DE**: Master Compare 1 DMA request Enable

This bit is set and cleared by software to enable/disable the Master timer Compare 1 DMA requests.
0: Compare 1 DMA request disabled
1: Compare 1 DMA request enabled

Bits 15:6 Reserved, must be kept at reset value.

Bit 6 **MUPDIE**: Master Update Interrupt Enable

This bit is set and cleared by software to enable/disable the Master timer registers update interrupts
0: Master update interrupts disabled
1: Master update interrupts enabled

Bit 5 **SYNCIE**: Sync Input Interrupt Enable

This bit is set and cleared by software to enable/disable the Sync input interrupts

0: Sync input interrupts disabled

1: Sync input interrupts enabled

Bit 4 **MREPIE**: Master Repetition Interrupt Enable

This bit is set and cleared by software to enable/disable the Master timer repetition interrupts

0: Master repetition interrupt disabled

1: Master repetition interrupt enabled

Bit 3 **MCMP4IE**: Master Compare 4 Interrupt Enable

Refer to MCMP1IE description

Bit 2 **MCMP3IE**: Master Compare 3 Interrupt Enable

Refer to MCMP1IE description

Bit 1 **MCMP2IE**: MAster Compare 2 Interrupt Enable

Refer to MCMP1IE description

Bit 0 **MCMP1IE**: Master Compare 1 Interrupt Enable

This bit is set and cleared by software to enable/disable the Master timer Compare 1 interrupt

0: Compare 1 interrupt disabled

1: Compare 1 interrupt enabled

21.5.5 HRTIM Master Timer Counter Register (HRTIM_MCNTR)

Address offset: 0x0010h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MCNT[15:0]**: *Counter value*

Holds the master timer counter value. This register can only be written when the master timer is stopped (MCEN = 0 in HRTIM_MCR).

Note: For HR clock prescaling ratio below 32 (CKPSCCKPSC[2:0] < 5), the least significant bits of the counter are not significant. They cannot be written and return 0 when read.

Note: The timer behavior is not guaranteed if the counter value is set above the HRTIM_MPER register value.

21.5.6 HRTIM Master Timer Period Register (HRTIM_MPER)

Address offset: 0x0014h

Reset value: 0x0000 FFDF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MPER[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MPER[15:0]**: *Master Timer Period value*

This register defines the counter overflow value.
The period value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...
The maximum value is 0x0000 FFDF.

21.5.7 HRTIM Master Timer Repetition Register (HRTIM_MREP)

Address offset: 0x0018h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MREP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:8 Reserved, must be kept at reset value.

Bits 7:0 **MREP[7:0]**: *Master Timer Repetition period value*

This register holds the repetition period value for the master counter. It is either the preload register or the active register if preload is disabled.

21.5.8 HRTIM Master Timer Compare 1 Register (HRTIM_MCMP1R)

Address offset: 0x001Ch

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MCMP1[15:0]**: *Master Timer Compare 1 value*

This register holds the master timer Compare 1 value. It is either the preload register or the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.9 HRTIM Master Timer Compare 2 Register (HRTIM_MCMP2R)

Address offset: 0x0024h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MCMP2[15:0]**: *Master Timer Compare 2 value*

This register holds the master timer Compare 2 value. It is either the preload register or the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.10 HRTIM Master Timer Compare 3 Register (HRTIM_MCMP3R)

Address offset: 0x0028h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MCMP3[15:0]**: *Master Timer Compare 3 value*

This register holds the master timer Compare 3 value. It is either the preload register or the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.11 HRTIM Master Timer Compare 4 Register (HRTIM_MCMP4R)

Address offset: 0x002Ch

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **MCMP4[15:0]**: *Master Timer Compare 4 value*

This register holds the master timer Compare 4 value. It is either the preload register or the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.12 HRTIM Timerx Control Register (HRTIM_TIMxCR)

Address offset: 0x0000h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDGAT[3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	Res.	TxRSTU	TxREPU	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DELCMP4[1:0]		DELCMP2[1:0]		SYNCSTRTx	SYNCRSTx	Res.	Res.	Res.	PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw	rw	rw

Bits 31:28 **UPDGAT[3:0]**: *Update Gating*

These bits define how the update occurs relatively to the burst DMA transaction and the external update request on update enable inputs 1 to 3 (see *Table 91: Update enable inputs and sources*)

The update events, as mentioned below, can be: MSTU, TEU, TDU, TCU, TBU, TAU, TxRSTU, TxREPU.

0000: the update occurs independently from the DMA burst transfer
0001: the update occurs when the DMA burst transfer is completed
0010: the update occurs on the update event following the DMA burst transfer completion
0011: the update occurs on a rising edge of HRTIM update enable input 1
0100: the update occurs on a rising edge of HRTIM update enable input 2
0101: the update occurs on a rising edge of HRTIM update enable input 3
0110: the update occurs on the update event following a rising edge of HRTIM update enable input 1
0111: the update occurs on the update event following a rising edge of HRTIM update enable input 2
1000: the update occurs on the update event following a rising edge of HRTIM update enable input 3
Other codes: reserved

Note: This bitfield must be reset before programming a new value.

For UPDGAT[3:0] values equal to 0001, 0011, 0100, 0101, it is possible to have multiple concurrent update source (for instance RSTU and DMA burst).

Bit 27 **PREEN**: *Preload enable*

This bit enables the registers preload mechanism and defines whether a write access into a preload-able register is done into the active or the preload register.

0: Preload disabled: the write access is directly done into the active register
1: Preload enabled: the write access is done into the preload register

Bits 26:25 **DACSYNC[1:0]** *DAC Synchronization*

A DAC synchronization event is generated when the timer update occurs. These bits are defining on which output the DAC synchronization is sent (refer to *Section 21.3.19: DAC triggers* for connections details).

00: No DAC trigger generated
01: Trigger generated on DACtrigOut1
10: Trigger generated on DACtrigOut2
11: Trigger generated on DACtrigOut3

Bit 24 **MSTU**: *Master Timer update*

Register update is triggered by the master timer update.

0: Update by master timer disabled
1: Update by master timer enabled

Bit 23 In HRTIM_TIMACR, HRTIM_TIMBCR, HRTIM_TIMCCR, HRTIM_TIMDCR:

TEU: *Timer E update*

Register update is triggered by the timer E update
0: Update by timer E disabled
1: Update by timer E enabled

In HRTIM_TIMECR:

Reserved, must be kept at reset value

Bit 22 In HRTIM_TIMACR, HRTIM_TIMBCR, HRTIM_TIMCCR, HRTIM_TIMECR:

TDU: *Timer D update*

Register update is triggered by the timer D update
0: Update by timer D disabled
1: Update by timer D enabled

In HRTIM_TIMDCR:

Reserved, must be kept at reset value

Bit 21 In HRTIM_TIMACR, HRTIM_TIMBCR, HRTIM_TIMDCR, HRTIM_TIMECR:

TCU: *Timer C update*

Register update is triggered by the timer C update
0: Update by timer C disabled
1: Update by timer C enabled

In HRTIM_TIMCCR:

Reserved, must be kept at reset value

Bit 20 In HRTIM_TIMACR, HRTIM_TIMCCR, HRTIM_TIMDCR, HRTIM_TIMECR:

TBU: *Timer B update*

Register update is triggered by the timer B update
0: Update by timer B disabled
1: Update by timer B enabled

In HRTIM_TIMBCR:

Reserved, must be kept at reset value

Bit 19 In HRTIM_TIMBCR, HRTIM_TIMCCR, HRTIM_TIMDCR, HRTIM_TIMECR:

TAU: *Timer A update*

Register update is triggered by the timer A update
0: Update by timer A disabled
1: Update by timer A enabled

In HRTIM_TIMACR:

Reserved, must be kept at reset value

Bit 18 **TxRSTU**: *Timerx reset update*

Register update is triggered by Timerx counter reset or roll-over to 0 after reaching the period value in continuous mode.
0: Update by timer x reset / roll-over disabled
1: Update by timer x reset / roll-over enabled

Bit 17 **TxREPU**: *Timer x Repetition update*

Register update is triggered when the counter rolls over and HRTIM_REPx = 0

0: Update on repetition disabled

1: Update on repetition enabled

Bit 16 Reserved, must be kept at reset value.

Bits 15:14 **DELCMP4[1:0]**: *CMP4 auto-delayed mode*

This bitfield defines whether the compare register is behaving in standard mode (compare match issued as soon as counter equal compare), or in auto-delayed mode (see *Auto-delayed mode*).

00: CMP4 register is always active (standard compare mode)

01: CMP4 value is recomputed and is active following a capture 2 event

10: CMP4 value is recomputed and is active following a capture 2 event, or is recomputed and active after Compare 1 match (timeout function if capture 2 event is missing)

11: CMP4 value is recomputed and is active following a capture event, or is recomputed and active after Compare 3 match (timeout function if capture event is missing)

Note: This bitfield must not be modified once the counter is enabled (TxCEN bit set)

Bits 13:12 **DELCMP2[1:0]**: *CMP2 auto-delayed mode*

This bitfield defines whether the compare register is behaving in standard mode (compare match issued as soon as counter equal compare), or in auto-delayed mode (see *Auto-delayed mode*).

00: CMP2 register is always active (standard compare mode)

01: CMP2 value is recomputed and is active following a capture 1 event

10: CMP2 value is recomputed and is active following a capture 1 event, or is recomputed and active after Compare 1 match (timeout function if capture event is missing)

11: CMP2 value is recomputed and is active following a capture 1 event, or is recomputed and active after Compare 3 match (timeout function if capture event is missing)

Note: This bitfield must not be modified once the counter is enabled (TxCEN bit set)

Bit 11 **SYNCSTRTx**: *Synchronization Starts Timer x*

This bit defines the Timer x behavior following the synchronization event:

0: No effect on Timer x

1: A synchronization input event starts the Timer x

Bit 10 **SYNCRSTx**: *Synchronization Resets Timer x*

This bit defines the Timer x behavior following the synchronization event:

0: No effect on Timer x

1: A synchronization input event resets the Timer x

Bits 9:7 Reserved, must be kept at reset value.

Bit 6 **PSHPLL**: *Push-Pull mode enable*

This bit enables the push-pull mode.

0: Push-Pull mode disabled

1: Push-Pull mode enabled

Note: This bitfield must not be modified once the counter is enabled (TxCEN bit set)

Bit 5 **HALF**: *Half mode enable*

This bit enables the half duty-cycle mode: the HRTIM_CMP1xR active register is automatically updated with HRTIM_PERxR/2 value when HRTIM_PERxR register is written.

0: Half mode disabled

1: Half mode enabled

Bit 4 **RETRIG**: *Re-triggerable mode*

This bit defines the counter behavior in single shot mode.

0: The timer is not re-triggerable: a counter reset is done if the counter is stopped (period elapsed in single-shot mode or counter stopped in continuous mode)

1: The timer is re-triggerable: a counter reset is done whatever the counter state.

Bit 3 **CONT**: *Continuous mode*

This bit defines the timer operating mode.

0: The timer operates in single-shot mode and stops when it reaches TIMxPER value

1: The timer operates in continuous mode and rolls over to zero when it reaches TIMxPER value

Bits 2:0 **CKPSCx[2:0]**: *HRTIM Timer x Clock prescaler*

These bits define the master timer high-resolution clock prescaler ratio.

The counter clock equivalent frequency ($f_{COUNTER}$) is equal to f_{HRCK} / $2^{CKPSC[2:0]}$.

The prescaling ratio cannot be modified once the timer is enabled.

21.5.13 HRTIM Timerx Interrupt Status Register (HRTIM_TIMxISR)

Address offset: 0x0004h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2STAT	O1STAT	IPPSTAT	CPPSTAT
										r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	r	r	r	r	r	r	r	r	r		r	r	r	r	r

Bits 31:22 Reserved, must be kept at reset value.

Bit 21 **O2CPY**: Output 2 Copy

This status bit is a raw copy of the output 2 state, before the output stage (chopper, polarity). It allows to check the current output state before re-enabling the output after a delayed protection.

0: Output 2 is inactive

1: Output 2 is active

Bit 20 **O1CPY**: Output 1 Copy

This status bit is a raw copy of the output 1 state, before the output stage (chopper, polarity). It allows to check the current output state before re-enabling the output after a delayed protection.

0: Output 1 is inactive

1: Output 1 is active

Bit 19 **O2STAT**: Output 2 Status

This status bit indicates the output 2 state when the delayed idle protection was triggered. This bit is updated upon any new delayed protection entry. This bit is not updated in balanced idle.

0: Output 2 was inactive

1: Output 2 was active

Bit 18 **O1STAT**: Output 1 Status

This status bit indicates the output 1 state when the delayed idle protection was triggered. This bit is updated upon any new delayed protection entry. This bit is not updated in balanced idle.

0: Output 1 was inactive

1: Output 1 was active

Bit 17 **IPPSTAT**: Idle Push Pull Status

This status bit indicates on which output the signal was applied, in push-pull mode balanced fault mode or delayed idle mode, when the protection was triggered (whatever the output state, active or inactive).

0: Protection occurred when the output 1 was active and output 2 forced inactive

1: Protection occurred when the output 2 was active and output 1 forced inactive

Bit 16 **CPPSTAT**: Current Push Pull Status

This status bit indicates on which output the signal is currently applied, in push-pull mode. It is only significant in this configuration.

0: Signal applied on output 1 and output 2 forced inactive

1: Signal applied on output 2 and output 1 forced inactive

Bit 15 Reserved

Bit 14 **DLYPRT**: Delayed Protection Flag

This bit indicates delayed idle or the balanced idle mode entry.

Bit 13 **RST**: Reset and/or roll-over Interrupt Flag

This bit is set by hardware when the timer x counter is reset or rolls over in continuous mode.
0: No TIMx counter reset/roll-over interrupt occurred
1: TIMX counter reset/roll-over interrupt occurred

Bit 12 **RSTx2**: Output 2 Reset Interrupt Flag

Refer to RSTx1 description

Bit 11 **SETx2**: Output 2 Set Interrupt Flag

Refer to SETx1 description

Bit 10 **RSTx1**: Output 1 Reset Interrupt Flag

This bit is set by hardware when the Tx1 output is reset (goes from active to inactive mode).
0: No Tx1 output reset interrupt occurred
1: Tx1 output reset interrupt occurred

Bit 9 **SETx1**: Output 1 Set Interrupt Flag

This bit is set by hardware when the Tx1 output is set (goes from inactive to active mode).
0: No Tx1 output set interrupt occurred
1: Tx1 output set interrupt occurred

Bit 8 **CPT2**: Capture2 Interrupt Flag

Refer to CPT1 description

Bit 7 **CPT1**: Capture1 Interrupt Flag

This bit is set by hardware when the timer x capture 1 event occurs.
0: No timer x Capture 1 reset interrupt occurred
1: Timer x output 1 reset interrupt occurred

Bit 6 **UPD**: Update Interrupt Flag

This bit is set by hardware when the timer x update event occurs.
0: No timer x update interrupt occurred
1: Timer x update interrupt occurred

Bit 5 Reserved, must be kept at reset value.

Bit 4 **REP**: Repetition Interrupt Flag

This bit is set by hardware when the timer x repetition period has elapsed.
0: No timer x repetition interrupt occurred
1: Timer x repetition interrupt occurred

Bit 3 **CMP4**: Compare 4 Interrupt Flag

Refer to CMP1 description

Bit 2 **CMP3**: Compare 3 Interrupt Flag

Refer to CMP1 description

Bit 1 **CMP2**: Compare 2 Interrupt Flag

Refer to CMP1 description

Bit 0 **CMP1**: Compare 1 Interrupt Flag

This bit is set by hardware when the timer x counter matches the value programmed in the Compare 1 register.
0: No Compare 1 interrupt occurred
1: Compare 1 interrupt occurred

21.5.14 HRTIM Timerx Interrupt Clear Register (HRTIM_TIMxICR)

Address offset: 0x0008h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

Bits 31:15 Reserved, must be kept at reset value.

Bit 14 **DLYPRTC**: Delayed Protection Flag Clear

Writing 1 to this bit clears the DLYPRT flag in HRTIM_TIMxISR register

Bit 13 **RSTC**: Reset Interrupt flag Clear

Writing 1 to this bit clears the RST flag in HRTIM_TIMxISR register

Bit 12 **RSTx2C**: Output 2 Reset flag Clear

Writing 1 to this bit clears the RSTx2 flag in HRTIM_TIMxISR register

Bit 11 **SETx2C**: Output 2 Set flag Clear

Writing 1 to this bit clears the SETx2 flag in HRTIM_TIMxISR register

Bit 10 **RSTx1C**: Output 1 Reset flag Clear

Writing 1 to this bit clears the RSTx1 flag in HRTIM_TIMxISR register

Bit 9 **SETx1C**: Output 1 Set flag Clear

Writing 1 to this bit clears the SETx1 flag in HRTIM_TIMxISR register

Bit 8 **CPT2C**: Capture2 Interrupt flag Clear

Writing 1 to this bit clears the CPT2 flag in HRTIM_TIMxISR register

Bit 7 **CPT1C**: Capture1 Interrupt flag Clear

Writing 1 to this bit clears the CPT1 flag in HRTIM_TIMxISR register

Bit 6 **UPDC**: Update Interrupt flag Clear

Writing 1 to this bit clears the UPD flag in HRTIM_TIMxISR register

Bit 5 Reserved, must be kept at reset value.

Bit 4 **REPC**: Repetition Interrupt flag Clear

Writing 1 to this bit clears the REP flag in HRTIM_TIMxISR register

Bit 3 **CMP4C**: Compare 4 Interrupt flag Clear

Writing 1 to this bit clears the CMP4 flag in HRTIM_TIMxISR register

Bit 2 **CMP3C**: Compare 3 Interrupt flag Clear

Writing 1 to this bit clears the CMP3 flag in HRTIM_TIMxISR register

Bit 1 **CMP2C**: Compare 2 Interrupt flag Clear

Writing 1 to this bit clears the CMP2 flag in HRTIM_TIMxISR register

Bit 0 **CMP1C**: Compare 1 Interrupt flag Clear

Writing 1 to this bit clears the CMP1 flag in HRTIM_TIMxISR register

21.5.15 HRTIM Timerx DMA / Interrupt Enable Register (HRTIM_TIMxDIER)

Address offset: 0x000Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SETx1DE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

Bit 31 Reserved

Bit 30 **DLYPRTDE**: Delayed Protection DMA request Enable

This bit is set and cleared by software to enable/disable DMA requests on delayed protection.
0: Delayed protection DMA request disabled
1: Delayed protection DMA request enabled

Bit 29 **RSTDE**: Reset/roll-over DMA request Enable

This bit is set and cleared by software to enable/disable DMA requests on timer x counter reset or roll-over in continuous mode.
0: Timer x counter reset/roll-over DMA request disabled
1: Timer x counter reset/roll-over DMA request enabled

Bit 28 **RSTx2DE**: Output 2 Reset DMA request Enable

Refer to RSTx1DE description

Bit 27 **SETx2DE**: Output 2 Set DMA request Enable

Refer to SETx1DE description

Bit 26 **RSTx1DE**: Output 1 Reset DMA request Enable

This bit is set and cleared by software to enable/disable Tx1 output reset DMA requests.
0: Tx1 output reset DMA request disabled
1: Tx1 output reset DMA request enabled

Bit 25 **SETx1DE**: Output 1 Set DMA request Enable

This bit is set and cleared by software to enable/disable Tx1 output set DMA requests.
0: Tx1 output set DMA request disabled
1: Tx1 output set DMA request enabled

Bit 24 **CPT2DE**: Capture 2 DMA request Enable

Refer to CPT1DE description

Bit 23 **CPT1DE**: Capture 1 DMA request Enable

This bit is set and cleared by software to enable/disable Capture 1 DMA requests.
0: Capture 1 DMA request disabled
1: Capture 1 DMA request enabled

Bit 22 **UPDDE**: Update DMA request Enable

This bit is set and cleared by software to enable/disable DMA requests on update event.
0: Update DMA request disabled
1: Update DMA request enabled

Bit 21 Reserved, must be kept at reset value.

Bit 20 **REPDE**: Repetition DMA request Enable
This bit is set and cleared by software to enable/disable DMA requests on repetition event.
0: Repetition DMA request disabled
1: Repetition DMA request enabled

Bit 19 **CMP4DE**: Compare 4 DMA request Enable
Refer to CMP1DE description

Bit 18 **CMP3DE**: Compare 3 DMA request Enable
Refer to CMP1DE description

Bit 17 **CMP2DE**: Compare 2 DMA request Enable
Refer to CMP1DE description

Bit 16 **CMP1DE**: Compare 1 DMA request Enable
This bit is set and cleared by software to enable/disable the Compare 1 DMA requests.
0: Compare 1 DMA request disabled
1: Compare 1 DMA request enabled

Bit 15 Reserved

Bit 14 **DLYPRTIE**: Delayed Protection Interrupt Enable
This bit is set and cleared by software to enable/disable interrupts on delayed protection.
0: Delayed protection interrupts disabled
1: Delayed protection interrupts enabled

Bit 13 **RSTIE**: Reset/roll-over Interrupt Enable
This bit is set and cleared by software to enable/disable interrupts on timer x counter reset or roll-over in continuous mode.
0: Timer x counter reset/roll-over interrupt disabled
1: Timer x counter reset/roll-over interrupt enabled

Bit 12 **RSTx2IE**: Output 2 Reset Interrupt Enable
Refer to RSTx1IE description

Bit 11 **SETx2IE**: Output 2 Set Interrupt Enable
Refer to SETx1IE description

Bit 10 **RSTx1IE**: Output 1 Reset Interrupt Enable
This bit is set and cleared by software to enable/disable Tx1 output reset interrupts.
0: Tx1 output reset interrupts disabled
1: Tx1 output reset interrupts enabled

Bit 9 **SETx1IE**: Output 1 Set Interrupt Enable
This bit is set and cleared by software to enable/disable Tx1 output set interrupts.
0: Tx1 output set interrupts disabled
1: Tx1 output set interrupts enabled

Bit 8 **CPT2IE**: Capture Interrupt Enable
Refer to CPT1IE description

Bit 7 **CPT1IE**: Capture Interrupt Enable
This bit is set and cleared by software to enable/disable Capture 1 interrupts.
0: Capture 1 interrupts disabled
1: Capture 1 interrupts enabled

Bit 6 **UPDIE**: Update Interrupt Enable

This bit is set and cleared by software to enable/disable update event interrupts.
0: Update interrupts disabled
1: Update interrupts enabled

Bit 5 Reserved, must be kept at reset value.

Bit 4 **REPIE**: Repetition Interrupt Enable

This bit is set and cleared by software to enable/disable repetition event interrupts.
0: Repetition interrupts disabled
1: Repetition interrupts enabled

Bit 3 **CMP4IE**: Compare 4 Interrupt Enable

Refer to CMP1IE description

Bit 2 **CMP3IE**: Compare 3 Interrupt Enable

Refer to CMP1IE description

Bit 1 **CMP2IE**: Compare 2 Interrupt Enable

Refer to CMP1IE description

Bit 0 **CMP1IE**: Compare 1 Interrupt Enable

This bit is set and cleared by software to enable/disable the Compare 1 interrupts.
0: Compare 1 interrupt disabled
1: Compare 1 interrupt enabled

21.5.16 HRTIM Timerx Counter Register (HRTIM_CNTxR)

Address offset: 0x0010h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNTx[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CNTx[15:0]**: *Timerx Counter value*

This register holds the Timerx counter value. It can only be written when the timer is stopped (TxCEN = 0 in HRTIM_TIMxCR).

Note: For HR clock prescaling ratio below 32 (CKPSC[2:0] < 5), the least significant bits of the counter are not significant. They cannot be written and return 0 when read.

Note: The timer behavior is not guaranteed if the counter value is above the HRTIM_PERxR register value.

21.5.17 HRTIM Timerx Period Register (HRTIM_PERxR)

Address offset: 0x14h (this offset address is relative to timer x base address)

Reset value: 0x0000 FFDF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERx[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **PERx[15:0]**: *Timerx Period value*

This register holds timer x period value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

The period value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

The maximum value is 0x0000 FFDF.

21.5.18 HRTIM Timerx Repetition Register (HRTIM_REPxR)

Address offset: 0x18h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

Bits31:8 Reserved, must be kept at reset value.

Bits 7:0 **REPx[7:0]**: *Timerx Repetition period value*

This register holds the repetition period value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

21.5.19 HRTIM Timerx Compare 1 Register (HRTIM_CMP1xR)

Address offset: 0x1Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
CMP1x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CMP1x[15:0]**: *Timerx Compare 1 value*

This register holds the compare 1 value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.20 HRTIM Timerx Compare 1 Compound Register (HRTIM_CMP1CxR)

Address offset: 0x20h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP1x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:24 Reserved, must be kept at reset value.

Bits 23:16 **REPx[7:0]**: *Timerx Repetition value (aliased from HRTIM_REPx register)*

This bitfield is an alias from the REPx[7:0] bitfield in the HRTIMx_REPxR register.

Bits 15:0 **CMP1x[15:0]**: *Timerx Compare 1 value*

This bitfield is an alias from the CMP1x[15:0] bitfield in the HRTIMx_CMP1xR register.

21.5.21 HRTIM Timerx Compare 2 Register (HRTIM_CMP2xR)

Address offset: 0x24h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP2x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CMP2x[15:0]**: *Timerx Compare 2 value*

This register holds the Compare 2 value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

This register can behave as an auto-delayed compare register, if enabled with DELCMP2[1:0] bits in HRTIM_TIMxCR.

21.5.22 HRTIM Timerx Compare 3 Register (HRTIM_CMP3xR)

Address offset: 0x28h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP3x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CMP3x[15:0]**: *Timerx Compare 3 value*

This register holds the Compare 3 value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

21.5.23 HRTIM Timerx Compare 4 Register (HRTIM_CMP4xR)

Address offset: 0x2Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP4x[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CMP4x[15:0]**: *Timerx Compare 4 value*

This register holds the Compare 4 value.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

The compare value must be above or equal to 3 periods of the f_{HRTIM} clock, that is 0x60 if CKPSC[2:0] = 0, 0x30 if CKPSC[2:0] = 1, 0x18 if CKPSC[2:0] = 2,...

This register can behave as an auto-delayed compare register, if enabled with DELCMP4[1:0] bits in HRTIM_TIMxCR.

21.5.24 HRTIM Timerx Capture 1 Register (HRTIM_CPT1xR)

Address offset: 0x30h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT1x[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CPT1x[15:0]**: *Timerx Capture 1 value*

This register holds the counter value when the capture 1 event occurred.

Note: This is a regular resolution register: for HR clock prescaling ratio below 32 (CKPSC[2:0] < 5), the least significant bits of the counter are not significant. They cannot be written and return 0 when read.

21.5.25 HRTIM Timerx Capture 2 Register (HRTIM_CPT2xR)

Address offset: 0x34h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT2x[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **CPT2x[15:0]**: *Timerx Capture 2 value*

This register holds the counter value when the capture 2 event occurred.

Note: This is a regular resolution register: for HR clock prescaling ratio below 32 (CKPSC[2:0] < 5), the least significant bits of the counter are not significant. They cannot be written and return 0 when read.

21.5.26 HRTIM Timerx Deadtime Register (HRTIM_DTxR)

Address offset: 0x38h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]								
rwo	rwo					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTRLKx	DTRSLKx	Res.	DTPRSC[1:0]			SDTRx	DTRx[8:0]								
rwo	rwo		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **DTFLKx**: *Deadtime Falling Lock*

This write-once bit prevents the deadtime (sign and value) to be modified, if enabled.
0: Deadtime falling value and sign is writable
1: Deadtime falling value and sign is read-only

Note: This bit is not preloaded

Bit 30 **DTFSLKx**: *Deadtime Falling Sign Lock*

This write-once bit prevents the sign of falling deadtime to be modified, if enabled.
0: Deadtime falling sign is writable
1: Deadtime falling sign is read-only

Note: This bit is not preloaded

Bits 29:26 Reserved, must be kept at reset value.

Bit 25 **SDTFx**: *Sign Deadtime Falling value*

This register determines whether the deadtime is positive (signals not overlapping) or negative (signals overlapping).
0: Positive deadtime on falling edge
1: Negative deadtime on falling edge

Bits 24:16 **DTFx[8:0]**: *Deadtime Falling value*

This register holds the value of the deadtime following a falling edge of reference PWM signal.
t_{DTF} = DTFx[8:0] x t_{DTG}

Bit 15 **DTRLKx**: *Deadtime Rising Lock*

This write-once bit prevents the deadtime (sign and value) to be modified, if enabled
0: Deadtime rising value and sign is writable
1: Deadtime rising value and sign is read-only

Note: This bit is not preloaded

Bit 14 **DTRSLKx**: *Deadtime Rising Sign Lock*

This write-once bit prevents the sign of deadtime to be modified, if enabled
0: Deadtime rising sign is writable
1: Deadtime rising sign is read-only

Note: This bit is not preloaded

Bit 13 Reserved, must be kept at reset value.

Bits 12:10 **DTPRSC[2:0]**: *Deadtime Prescaler*

This register holds the value of the deadtime clock prescaler.

t_{DTG} = $(2^{(DTPRSC[2:0])})$ x (t_{HRTIM} / 8)

(i.e. 000: 868 ps, 001= 1.736ns,...)

This bitfield is read-only as soon as any of the lock bit is enabled (DTFLKs, DTFSLKx, DTRLKx, DTRSLKx).

Bit 9 **SDTRx**: *Sign Deadtime Rising value*

This register determines whether the deadtime is positive or negative (overlapping signals)

0: Positive deadtime on rising edge

1: Negative deadtime on rising edge

Bits 8:0 **DTRx[8:0]**: *Deadtime Rising value*

This register holds the value of the deadtime following a rising edge of reference PWM signal.

t_{DTR} = DTRx[8:0] x t_{DTG}

21.5.27 HRTIM Timerx Output1 Set Register (HRTIM_SETx1R)

Address offset: 0x3Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXT EVNT10	EXT EVNT9	EXT EVNT8	EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	TIM EVNT9	TIM EVNT8	TIM EVNT7	TIM EVNT6	TIM EVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIM EVNT4	TIM EVNT3	TIM EVNT2	TIM EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **UPDATE**: *Registers update (transfer preload to active)*
Register update event forces the output to its active state.

Bit 30 **EXTEVNT10**: *External Event 10*
Refer to EXTEVNT1 description

Bit 29 **EXTEVNT9**: *External Event 9*
Refer to EXTEVNT1 description

Bit 28 **EXTEVNT8**: *External Event 8*
Refer to EXTEVNT1 description

Bit 27 **EXTEVNT7**: *External Event 7*
Refer to EXTEVNT1 description

Bit 26 **EXTEVNT6**: *External Event 6*
Refer to EXTEVNT1 description

Bit 25 **EXTEVNT5**: *External Event 5*
Refer to EXTEVNT1 description

Bit 24 **EXTEVNT4**: *External Event 4*
Refer to EXTEVNT1 description

Bit 23 **EXTEVNT3**: *External Event 3*
Refer to EXTEVNT1 description

Bit 22 **EXTEVNT2**: *External Event 2*
Refer to EXTEVNT1 description

Bit 21 **EXTEVNT1**: *External Event 1*
External event 1 forces the output to its active state.

Bit 20 **TIMEVNT9**: *Timer Event 9*
Refer to TIMEVNT1 description

Bit 19 **TIMEVNT8**: *Timer Event 8*
Refer to TIMEVNT1 description

Bit 18 **TIMEVNT7**: *Timer Event 7*
Refer to TIMEVNT1 description

Bit 17 **TIMEVNT6**: *Timer Event 6*
Refer to TIMEVNT1 description

Bit 16 **TIMEVNT5**: *Timer Event 5*

Refer to TIMEVNT1 description

Bit 15 **TIMEVNT4**: *Timer Event 4*

Refer to TIMEVNT1 description

Bit 14 **TIMEVNT3**: *Timer Event 3*

Refer to TIMEVNT1 description

Bit 13 **TIMEVNT2**: *Timer Event 2*

Refer to TIMEVNT1 description

Bit 12 **TIMEVNT1**:*Timer Event 1*

Timers event 1 forces the output to its active state (refer to *Table 84* for Timer Events assignments)

Bit 11 **MSTCMP4**: *Master Compare 4*

Master Timer Compare 4 event forces the output to its active state.

Bit 10 **MSTCMP3**: *Master Compare 3*

Master Timer Compare 3 event forces the output to its active state.

Bit 9 **MSTCMP2**: *Master Compare 2*

Master Timer Compare 2 event forces the output to its active state.

Bit 8 **MSTCMP1**: *Master Compare 1*

Master Timer compare 1 event forces the output to its active state.

Bit 7 **MSTPER**: *Master Period*

The master timer counter roll-over in continuous mode, or to the master timer reset in single-shot mode forces the output to its active state.

Bit 6 **CMP4**: *Timer x Compare 4*

Timer A compare 4 event forces the output to its active state.

Bit 5 **CMP3**: *Timer x Compare 3*

Timer A compare 3 event forces the output to its active state.

Bit 4 **CMP2**: *Timer x Compare 2*

Timer A compare 2 event forces the output to its active state.

Bit 3 **CMP1**: *Timer x Compare 1*

Timer A compare 1 event forces the output to its active state.

Bit 2 **PER**: *Timer x Period*

Timer A Period event forces the output to its active state.

Bit 1 **RESYNC:** *Timer A resynchronization*

Timer A reset event coming solely from software or SYNC input forces the output to its active state.

Note: Other timer reset are not affecting the output when RESYNC=1

Bit 0 **SST**: *Software Set trigger*

This bit forces the output to its active state. This bit can only be set by software and is reset by hardware.

Note: This bit is not preloaded

21.5.28 HRTIM Timerx Output1 Reset Register (HRTIM_RSTx1R)

Address offset: 0x40h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXT EVNT10	EXT EVNT9	EXT EVNT8	EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	TIM EVNT9	TIM EVNT8	TIM EVNT7	TIM EVNT6	TIM EVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIM EVNT4	TIM EVNT3	TIM EVNT2	TIM EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 Refer to HRTIM_SETx1R bits description.
These bits are defining the source which can force the Tx1 output to its inactive state.

21.5.29 HRTIM Timerx Output2 Set Register (HRTIM_SETx2R)

Address offset: 0x44h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXT EVNT10	EXT EVNT9	EXT EVNT8	EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	TIM EVNT9	TIM EVNT8	TIM EVNT7	TIM EVNT6	TIM EVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIM EVNT4	TIM EVNT3	TIM EVNT2	TIM EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 Refer to HRTIM_SETx1R bits description.
These bits are defining the source which can force the Tx2 output to its active state.

21.5.30 HRTIM Timerx Output2 Reset Register (HRTIM_RSTx2R)

Address offset: 0x48h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXT EVNT10	EXT EVNT9	EXT EVNT8	EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	TIM EVNT9	TIM EVNT8	TIM EVNT7	TIM EVNT6	TIM EVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIM EVNT4	TIM EVNT3	TIM EVNT2	TIM EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 Refer to HRTIM_SETx1R bits description.

These bits are defining the source which can force the Tx2 output to its inactive state.

21.5.31 HRTIM Timerx External Event Filtering Register 1 (HRTIM_EEFxR1)

Address offset: 0x4Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE5FLTR[3:0]				EE5LTCH	Res.	EE4FLTR[3:0]				EE4LTCH	Res.	EE3FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EE3FLTR[2:0]			EE3LTCH	Res.	EE2FLTR[3:0]				EE2LTCH	Res.	EE1FLTR[3:0]				EE1LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

Bits 31:29 Reserved, must be kept at reset value.

Bits 28:25 **EE5FLTR[3:0]**: *External Event 5 filter*
Refer to EE1FLTR[3:0] description

Bit 24 **EE5LTCH**: *External Event 5 latch*
Refer to EE1LTCH description

Bit 23 Reserved, must be kept at reset value.

Bits 22:19 **EE4FLTR[3:0]**: *External Event 4 filter*
Refer to EE1FLTR[3:0] description

Bit 18 **EE4LTCH**: *External Event 4 latch*
Refer to EE1LTCH description

Bit 17 Reserved, must be kept at reset value.

Bits 16:13 **EE3FLTR[3:0]**: *External Event 3 filter*
Refer to EE1FLTR[3:0] description

Bit 12 **EE3LTCH**: *External Event 3 latch*
Refer to EE1LTCH description

Bit 11 Reserved, must be kept at reset value.

Bits 10:7 **EE2FLTR[3:0]**: *External Event 2 filter*
Refer to EE1FLTR[3:0] description

Bit 6 **EE2LTCH**: *External Event 2 latch*
Refer to EE1LTCH description

Bit 5 Reserved, must be kept at reset value.

Bits 4:1 **EE1FLTR[3:0]**: *External Event 1 filter*

0000: No filtering
0001: Blanking from counter reset/roll-over to Compare 1
0010: Blanking from counter reset/roll-over to Compare 2
0011: Blanking from counter reset/roll-over to Compare 3
0100: Blanking from counter reset/roll-over to Compare 4
0101: Blanking from another timing unit: TIMFLTR1 source (see *Table 88* for details)
0110: Blanking from another timing unit: TIMFLTR2 source (see *Table 88* for details)
0111: Blanking from another timing unit: TIMFLTR3 source (see *Table 88* for details)
1000: Blanking from another timing unit: TIMFLTR4 source (see *Table 88* for details)
1001: Blanking from another timing unit: TIMFLTR5 source (see *Table 88* for details)
1010: Blanking from another timing unit: TIMFLTR6 source (see *Table 88* for details)
1011: Blanking from another timing unit: TIMFLTR7 source (see *Table 88* for details)
1100: Blanking from another timing unit: TIMFLTR8 source (see *Table 88* for details)
1101: Windowing from counter reset/roll-over to Compare 2
1110: Windowing from counter reset/roll-over to Compare 3
1111: Windowing from another timing unit: TIMWIN source (see *Table 89* for details)

Note: Whenever a compare register is used for filtering, the value must be strictly above 0.
This bitfield must not be modified once the counter is enabled (TxCEN bit set)

Bit 0 **EE1LTCH**: *External Event 1 latch*

0: Event 1 is ignored if it happens during a blank, or passed through during a window.
1: Event 1 is latched and delayed till the end of the blanking or windowing period.

Note: A timeout event is generated in window mode (EE1FLTR[3:0]=1101, 1110, 1111) if EE1LTCH = 0, except if the External event is programmed in fast mode (EExFAST = 1).
This bitfield must not be modified once the counter is enabled (TxCEN bit set)

21.5.32 HRTIM Timerx External Event Filtering Register 2 (HRTIM_EEFxR2)

Address offset: 0x50h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10FLTR[3:0]				EE10LTCH	Res.	EE9FLTR[3:0]				EE9LTCH	Res.	EE8FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8FLTR[2:0]			EE8LTCH	Res.	EE7FLTR[3:0]				EE7LTCH	Res.	EE6FLTR[3:0]				EE6LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

Bits 31:29 Reserved, must be kept at reset value.

Bits 28:25 **EE10FLTR[3:0]**: *External Event 10 filter*
Refer to EE1FLTR[3:0] description

Bit 24 **EE10LTCH**: *External Event 10 latch*
Refer to EE1LTCH description

Bit 23 Reserved, must be kept at reset value.

Bits 22:19 **EE9FLTR[3:0]**: *External Event 9 filter*
Refer to EE1FLTR[3:0] description

Bit 18 **EE9LTCH**: *External Event 9 latch*
Refer to EE1LTCH description

Bit 17 Reserved, must be kept at reset value.

Bits 16:13 **EE8FLTR[3:0]**: *External Event 8 filter*
Refer to EE1FLTR[3:0] description

Bit 12 **EE8LTCH**: *External Event 8 latch*
Refer to EE1LTCH description

Bit 11 Reserved, must be kept at reset value.

Bits 10:7 **EE7FLTR[3:0]**: *External Event 7 filter*
Refer to EE1FLTR[3:0] description

Bit 6 **EE7LTCH**: *External Event 7 latch*
Refer to EE1LTCH description

Bit 5 Reserved, must be kept at reset value.

Bits 4:1 **EE6FLTR[3:0]**: *External Event 6 filter*
Refer to EE1FLTR[3:0] description

Bit 0 **EE6LTCH**: *External Event 6 latch*
Refer to EE1LTCH description

21.5.33 HRTIM Timerx Reset Register (HRTIM_RSTxR)

HRTIM TimerA Reset Register (HRTIM_RSTAR)

Address offset: 0xD4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIME CMP4	TIME CMP2	TIME CMP1	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	EXTEV NT10	EXTEV NT9	EXTEV NT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEV NT7	EXTEV NT6	EXTEV NT5	EXTEV NT4	EXTEV NT3	EXTEV NT2	EXTEV NT1	MSTC MP4	MSTC MP3	MSTC MP2	MSTC MP1	MSTPE R	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bit 31 Reserved, must be kept at reset value.

Bit 30 **TECPM4**: *Timer E Compare 4*

The timer A counter is reset upon timer E Compare 4 event.

Bit 29 **TECMP2**: *Timer E Compare 2*

The timer A counter is reset upon timer E Compare 2 event.

Bit 28 **TECMP1**: *Timer E Compare 1*

The timer A counter is reset upon timer E Compare 1 event.

Bit 27 **TDCMP4**: *Timer D Compare 4*

The timer A counter is reset upon timer D Compare 4 event.

Bit 26 **TDCMP2**: *Timer D Compare 2*

The timer A counter is reset upon timer D Compare 2 event.

Bit 25 **TDCMP1**: *Timer D Compare 1*

The timer A counter is reset upon timer D Compare 1 event.

Bit 24 **TCCMP4**: *Timer C Compare 4*

The timer A counter is reset upon timer C Compare 4 event.

Bit 23 **TCCMP2**: *Timer C Compare 2*

The timer A counter is reset upon timer C Compare 2 event.

Bit 22 **TCCMP1**: *Timer C Compare 1*

The timer A counter is reset upon timer C Compare 1 event.

Bit 21 **TBCMP4**: *Timer B Compare 4*

The timer A counter is reset upon timer B Compare 4 event.

Bit 20 **TBCMP2**: *Timer B Compare 2*

The timer A counter is reset upon timer B Compare 2 event.

Bit 19 **TBCMP1**: *Timer B Compare 1*

The timer A counter is reset upon timer B Compare 1 event.

Bit 18 **EXTEVNT10**: *External Event*

The timer A counter is reset upon external event 10.

Bit 17 **EXTEVNT9**: *External Event 9*

The timer A counter is reset upon external event 9.

Bit 16 **EXTEVNT8**: *External Event 8*
The timer A counter is reset upon external event 8.

Bit 15 **EXTEVNT7**: *External Event 7*
The timer A counter is reset upon external event 7.

Bit 14 **EXTEVNT6**: *External Event 6*
The timer A counter is reset upon external event 6.

Bit 13 **EXTEVNT5**: *External Event 5*
The timer A counter is reset upon external event 5.

Bit 12 **EXTEVNT4**: *External Event 4*
The timer A counter is reset upon external event 4.

Bit 11 **EXTEVNT3**: *External Event 3*
The timer A counter is reset upon external event 3.

Bit 10 **EXTEVNT2**: *External Event 2*
The timer A counter is reset upon external event 2.

Bit 9 **EXTEVNT1**: *External Event 1*
The timer A counter is reset upon external event 1.

Bit 8 **MSTCMP4**: *Master compare 4*
The timer A counter is reset upon master timer Compare 4 event.

Bit 7 **MSTCMP3**: *Master compare 3*
The timer A counter is reset upon master timer Compare 3 event.

Bit 6 **MSTCMP2**: *Master compare 2*
The timer A counter is reset upon master timer Compare 2 event.

Bit 5 **MSTCMP1**: *Master compare 1*
The timer A counter is reset upon master timer Compare 1 event.

Bit 4 **MSTPER** *Master timer Period*
The timer A counter is reset upon master timer period event.

Bit 3 **CMP4**: *Timer A compare 4 reset*
The timer A counter is reset upon Timer A Compare 4 event.

Bit 2 **CMP2**: *Timer A compare 2 reset*
The timer A counter is reset upon Timer A Compare 2 event.

Bit 1 **UPDT**: *Timer A Update reset*
The timer A counter is reset upon update event.

Bit 0 Reserved, must be kept at reset value.

HRTIM TimerB Reset Register (HRTIM_RSTBR)

Address offset: 0x154h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 30:1 Refer to HRTIM_RSTAR bits description.

Bits 30:19 differ (reset signals come from TIMA, TIMC, TIMD and TIME)

HRTIM TimerC Reset Register (HRTIM_RSTCR)

Address offset: 0x1D4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 30:1 Refer to HRTIM_RSTAR bits description.

Bits 30:19 differ (reset signals come from TIMA, TIMB, TIMD and TIME)

HRTIM TimerD Reset Register (HRTIM_RSTDR)

Address offset: 0x254h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 30:1 Refer to HRTIM_RSTAR bits description.

Bits 30:19 differ (reset signals come from TIMA, TIMB, TIMC and TIME)

HRTIM Timerx Reset Register (HRTIM_RSTER)

Address offset: 0x2D4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXTEV NT10	EXTEV NT9	EXTEV NT8
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEV NT7	EXTEV NT6	EXTEV NT5	EXTEV NT4	EXTEV NT3	EXTEV NT2	EXTEV NT1	MSTC MP4	MSTC MP3	MSTC MP2	MSTC MP1	MSTPE R	CMP4	CMP2	UPDT	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 30:1 Refer to HRTIM_RSTAR bits description.
Bits 30:19 differ (reset signals come from TIMA, TIMB, TIMC and TIMD)

21.5.34 HRTIM Timerx Chopper Register (HRTIM_CHPxR)

Address offset: 0x58h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	STRTPW[3:0]				CARDTY[2:0)			CARFRQ[3:0]			
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:11 Reserved, must be kept at reset value.

Bits 10:7 **STRPW[3:0]**: *Timerx start pulsewidth*

This register defines the initial pulsewidth following a rising edge on output signal.
This bitfield cannot be modified when one of the CHPx bits is set.
t_{1STPW} = (STRPW[3:0]+1) x 16 x t_{HRTIM}.

0000: 111 ns (1/9 MHz)
...
1111: 1.77 µs (16/9 MHz)

Bits 6:4 **CARDTY[2:0]**: *Timerx chopper duty cycle value*

This register defines the duty cycle of the carrier signal. This bitfield cannot be modified when one of the CHPx bits is set.
000: 0/8 (i.e. only 1st pulse is present)
...
111: 7/8

Bits 3:0 **CARFRQ[3:0]**: *Timerx carrier frequency value*

This register defines the carrier frequency F_{CHPFRQ} = f_{HRTIM} / (16 x (CARFRQ[3:0]+1)).
This bitfield cannot be modified when one of the CHPx bits is set.
0000: 9 MHz (f_{HRTIM}/ 16)
...
1111: 562.5 kHz (f_{HRTIM} / 256)

21.5.35 HRTIM Timerx Capture 1 Control Register (HRTIM_CPT1xCR)

Address offset: 0x5Ch (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved (for TIME only)				Reserved (for TIMD only)				Reserved (for TIMC only)				Reserved (for TIMB only)			
TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved (for TIMA only)				EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
TACMP2	TACMP1	TA1RST	TA1SET												
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 Refer to HRTIM_CPT2xCR bit description

21.5.36 HRTIM Timerx Capture 2 Control Register (HRTIM_CPT2xCR)

Address offset: 0x60h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved (for TIME only)				Reserved (for TIMD only)				Reserved (for TIMC only)				Reserved (for TIMB only)			
TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved (for TIMA only)				EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
TACMP2	TACMP1	TA1RST	TA1SET												
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **TECMP2**: Timer E Compare 2

Refer to TACMP1 description

Note: This bit is reserved for Timer E

Bit 30 **TECMP1**: Timer E Compare 1

Refer to TACMP1 description

Note: This bit is reserved for Timer E

Bit 29 **TE1RST**: Timer E output 1 Reset

Refer to TA1RST description

Note: This bit is reserved for Timer E

Bit 28 **TE1SET**: Timer E output 1 Set

Refer to TA1SET description

Note: This bit is reserved for Timer E

Bit 27 **TDCMP2**: Timer D Compare 2

Refer to TACMP1 description

Note: This bit is reserved for Timer D

Bit 26 **TDCMP1**:Timer D Compare 1

Refer to TACMP1 description

Note: This bit is reserved for Timer D

Bit 25 **TD1RST**: Timer D output 1 Reset

Refer to TA1RST description

Note: This bit is reserved for Timer D

Bit 24 **TD1SET**: Timer D output 1 Set

Refer to TA1SET description

Note: This bit is reserved for Timer D

Bit 23 **TCCMP2**: Timer C Compare 2

Refer to TACMP1 description

Note: This bit is reserved for Timer C

Bit 22 **TCCMP1**:Timer C Compare 1

Refer to TACMP1 description

Note: This bit is reserved for Timer C

Bit 21 **TC1RST**: Timer C output 1 Reset

Refer to TA1RST description

Note: This bit is reserved for Timer C

Bit 20 **TC1SET**: Timer C output 1 Set

Refer to TA1SET description

Note: This bit is reserved for Timer C

Bit 19 **TBCMP2**: Timer B Compare 2

Refer to TACMP1 description

Note: This bit is reserved for Timer B

Bit 18 **TBCMP1**: Timer B Compare 1

Refer to TACMP1 description

Note: This bit is reserved for Timer B

Bit 17 **TB1RST**: Timer B output 1 Reset

Refer to TA1RST description

Note: This bit is reserved for Timer B

Bit 16 **TB1SET**: Timer B output 1 Set

Refer to TA1SET description

Note: This bit is reserved for Timer B

Bit 15 **TACMP2**: Timer A Compare 2

Timer A Compare 2 triggers Capture 2.

Note: This bit is reserved for Timer A

Bit 14 **TACMP1**: Timer A Compare 1

Timer A Compare 1 triggers Capture 2.

Note: This bit is reserved for Timer A

Bit 13 **TA1RST**: Timer B output 1 Reset

Capture 2 is triggered by HRTIM_CHA1 output active to inactive transition.

Note: This bit is reserved for Timer A

Bit 12 **TA1SET**: Timer B output 1 Set

Capture 2 is triggered by HRTIM_CHA1 output inactive to active transition.

Note: This bit is reserved for Timer A

Bit 11 **EXEV10CPT**: *External Event 10 Capture*

Refer to EXEV1CPT description

Bit 10 **EXEV9CPT**: *External Event 9 Capture*

Refer to EXEV1CPT description

Bit 9 **EXEV8CPT**: *External Event 8 Capture*

Refer to EXEV1CPT description

Bit 8 **EXEV7CPT**: *External Event 7 Capture*

Refer to EXEV1CPT description

Bit 7 **EXEV6CPT**: *External Event 6 Capture*

Refer to EXEV1CPT description

Bit 6 **EXEV5CPT**: *External Event 5 Capture*

Refer to EXEV1CPT description

Bit 5 **EXEV4CPT**: *External Event 4 Capture*

Refer to EXEV1CPT description

Bit 4 **EXEV3CPT**: *External Event 3 Capture*

Refer to EXEV1CPT description

Bit 3 **EXEV2CPT**: *External Event 2 Capture*

Refer to EXEV1CPT description

Bit 2 **EXEV1CPT**: *External Event 1 Capture*

The External event 1 triggers the Capture 2.

Bit 1 **UPDCPT**: *Update Capture*

The update event triggers the Capture 2.

Bit 0 **SWCPT**: *Software Capture*

This bit forces the Capture 2 by software. This bit is set only, reset by hardware

21.5.37 HRTIM Timerx Output Register (HRTIM_OUTxR)

Address offset: 0x64h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM 2	POL2	Res.
								rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DLYPRT[2:0]			DLYPR TEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM 1	POL1	Res.
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bits 31:24 Reserved, must be kept at reset value.

Bit 23 **DIDL2**: *Output 2 Deadtime upon burst mode Idle entry*

This bit can delay the idle mode entry by forcing a deadtime insertion before switching the outputs to their idle state. This setting only applies when entering in idle state during a burst mode operation.

0: The programmed Idle state is applied immediately to the Output 2

1: Deadtime (inactive level) is inserted on output 2 before entering the idle mode. The deadtime value is set by DTFx[8:0].

Note: This parameter cannot be changed once the timer x is enabled.

DIDL=1 can be set only if one of the outputs is active during the burst mode (IDLES=1), and with positive deadtimes (SDTR/SDTF set to 0).

Bit 22 **CHP2**: *Output 2 Chopper enable*

This bit enables the chopper on output 2

0: Output signal is not altered

1: Output signal is chopped by a carrier signal

Note: This parameter cannot be changed once the timer x is enabled.

Bits 21:20 **FAULT2[1:0]**: *Output 2 Fault state*

These bits select the output 2 state after a fault event

00: No action: the output is not affected by the fault input and stays in run mode.

01: Active

10: Inactive

11: High-Z

Note: This parameter cannot be changed once the timer x is enabled (TxCEN bit set), if FLTENx bit is set or if the output is in FAULT state.

Bit 19 **IDLES2**: *Output 2 Idle State*

This bit selects the output 2 idle state

0: Inactive

1: Active

Note: This parameter must be set prior to have the HRTIM controlling the outputs.

Bit 18 **IDLEM2**: *Output 2 Idle mode*

This bit selects the output 2 idle mode

0: No action: the output is not affected by the burst mode operation

1: The output is in idle state when requested by the burst mode controller.

Note: This bit is preloaded and can be changed during run-time, but must not be changed while the burst mode is active.

Bit 17 **POL2**: *Output 2 polarity*

This bit selects the output 2 polarity

0: positive polarity (output active high)

1: negative polarity (output active low)

Note: This parameter cannot be changed once the timer x is enabled.

Bits 16:12 Reserved, must be kept at reset value.

Bits 12:10 **DLYPRT[2:0]**: *Delayed Protection*

These bits define the source and outputs on which the delayed protection schemes are applied.

In HRTIM_OUTAR, HRTIM_OUTBR, HRTIM_OUTCR:

000: Output 1 delayed Idle on external Event 6

001: Output 2 delayed Idle on external Event 6

010: Output 1 and output 2 delayed Idle on external Event 6

011: Balanced Idle on external Event 6

100: Output 1 delayed Idle on external Event 7

101: Output 2 delayed Idle on external Event 7

110: Output 1 and output 2 delayed Idle on external Event 7

111: Balanced Idle on external Event 7

In HRTIM_OUTDR, HRTIM_OUTER:

000: Output 1 delayed Idle on external Event 8

001: Output 2 delayed Idle on external Event 8

010: Output 1 and output 2 delayed Idle on external Event 8

011: Balanced Idle on external Event 8

100: Output 1 delayed Idle on external Event 9

101: Output 2 delayed Idle on external Event 9

110: Output 1 and output 2 delayed Idle on external Event 9

111: Balanced Idle on external Event 9

Note: This bitfield must not be modified once the delayed protection is enabled (DLYPRTEN bit set)

Bit 9 **DLYPRTEN**: *Delayed Protection Enable*

This bit enables the delayed protection scheme

0: No action

1: Delayed protection is enabled, as per DLYPRT[2:0] bits

Note: This parameter cannot be changed once the timer x is enabled (TxEN bit set).

Bit 8 **DTEN**: *Deadtime enable*

This bit enables the deadtime insertion on output 1 and output 2

0: Output 1 and output 2 signals are independent.

1: Deadtime is inserted between output 1 and output 2 (reference signal is output 1 signal generator)

Note: This parameter cannot be changed once the timer is operating (TxEN bit set) or if its outputs are enabled and set/reset by another timer.

Bit 7 **DIDL1**: *Output 1 Deadtime upon burst mode Idle entry*

This bit can delay the idle mode entry by forcing a deadtime insertion before switching the outputs to their idle state. This setting only applies when entering the idle state during a burst mode operation.

0: The programmed Idle state is applied immediately to the Output 1

1: Deadtime (inactive level) is inserted on output 1 before entering the idle mode. The deadtime value is set by DTRx[8:0].

Note: This parameter cannot be changed once the timer x is enabled.

DIDL=1 can be set only if one of the outputs is active during the burst mode (IDLES=1), and with positive deadtimes (SDTR/SDTF set to 0).

Bit 6 **CHP1**: *Output 1 Chopper enable*

This bit enables the chopper on output 1

0: Output signal is not altered

1: Output signal is chopped by a carrier signal

Note: This parameter cannot be changed once the timer x is enabled.

Bits 5:4 **FAULT1[1:0]**: *Output 1 Fault state*

These bits select the output 1 state after a fault event

00: No action: the output is not affected by the fault input and stays in run mode.

01: Active

10: Inactive

11: High-Z

Note: This parameter cannot be changed once the timer x is enabled (TxCEN bit set), if FLTENx bit is set or if the output is in FAULT state.

Bit 3 **IDLES1**: *Output 1 Idle State*

This bit selects the output 1 idle state

0: Inactive

1: Active

Note: This parameter must be set prior to HRTIM controlling the outputs.

Bit 2 **IDLEM1**: *Output 1 Idle mode*

This bit selects the output 1 idle mode

0: No action: the output is not affected by the burst mode operation

1: The output is in idle state when requested by the burst mode controller.

Note: This bit is preloaded and can be changed during runtime, but must not be changed while burst mode is active.

Bit 1 **POL1**: *Output 1 polarity*

This bit selects the output 1 polarity

0: positive polarity (output active high)

1: negative polarity (output active low)

Note: This parameter cannot be changed once the timer x is enabled.

Bit 0 Reserved

21.5.38 HRTIM Timerx Fault Register (HRTIM_FLTxR)

Address offset: 0x68h (this offset address is relative to timer x base address)

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLTLCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rwo															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5EN	FLT4EN	FLT3EN	FLT2EN	FLT1EN
											rw	rw	rw	rw	rw

Bit 31 **FLTLCK**: *Fault sources Lock*

0: FLT1EN..FLT5EN bits are read/write

1: FLT1EN..FLT5EN bits are read only

The FLTLCK bit is write-once. Once it has been set, it cannot be modified till the next system reset.

Bits 30:5 Reserved, must be kept at reset value.

Bit 4 **FLT5EN**: *Fault 5 enable*

0: Fault 5 input ignored

1: Fault 5 input is active and can disable HRTIM outputs.

Bit 3 **FLT4EN**: *Fault 4 enable*

0: Fault 4 input ignored

1: Fault 4 input is active and can disable HRTIM outputs.

Bit 2 **FLT3EN**: *Fault 3 enable*

0: Fault 3 input ignored

1: Fault 3 input is active and can disable HRTIM outputs.

Bit 1 **FLT2EN**: *Fault 2 enable*

0: Fault 2 input ignored

1: Fault 2 input is active and can disable HRTIM outputs.

Bit 0 **FLT1EN**: *Fault 1 enable*

0: Fault 1 input ignored

1: Fault 1 input is active and can disable HRTIM outputs.

21.5.39 HRTIM Control Register 1 (HRTIM_CR1)

Address offset: 0x380h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AD4USRC[2:0]			AD3USRC[2:0]			AD2USRC[2:0]			AD1USRC[2:0]		
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEUDIS	TDUDIS	TCUDIS	TBUDIS	TAUDIS	MUDIS
										rw	rw	rw	rw	rw	rw

Bits 31:28 Reserved, must be kept at reset value.

Bits 27:25 **AD4USRC[2:0]**: *ADC Trigger 4* Update Source
Refer to AD1USRC[2:0] description

Bits 24:22 **AD3USRC[2:0]**: *ADC Trigger 3 Update Source*
Refer to AD1USRC[2:0] description

Bits 21:19 **AD2USRC[2:0]**: *ADC Trigger 2 Update Source*
Refer to AD1USRC[2:0] description

Bits 18:16 **AD1USRC[2:0]**: *ADC Trigger 1 Update Source*
These bits define the source which will trigger the update of the HRTIM_ADC1R register (transfer from preload to active register). It only defines the source timer. The precise condition is defined within the timer itself, in HRTIM_MCR or HRTIM_TIMxCR.
000: Master Timer
001: Timer A
010: Timer B
011: Timer C
100: Timer D
101: Timer E
110, 111: Reserved

Bits 15:6 Reserved, must be kept at reset value.

Bit 5 **TEUDIS**: *Timer E Update Disable*
Refer to TAUDIS description

Bit 4 **TDUDIS**: *Timer D Update Disable*
Refer to TAUDIS description

Bit 3 **TCUDIS**: *Timer C Update Disable*
Refer to TAUDIS description

Bit 2 **TBUDIS**: *Timer B Update Disable*

Refer to TAUDIS description

Bit 1 **TAUDIS**: *Timer A Update Disable*

This bit is set and cleared by software to enable/disable an update event generation temporarily on Timer A.

0: update enabled. The update occurs upon generation of the selected source.

1: update disabled. The updates are temporarily disabled to allow the software to write multiple registers that have to be simultaneously taken into account.

Bit 0 **MUDIS**: Master *Update Disable*

This bit is set and cleared by software to enable/disable an update event generation temporarily.

0: update enabled.

1: update disabled. The updates are temporarily disabled to allow the software to write multiple registers that have to be simultaneously taken into account.

21.5.40 HRTIM Control Register 2 (HRTIM_CR2)

Address offset: 0x384h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	Res.	TESWU	TDSWU	TCSWU	TBSWU	TASWU	MSWU
		rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

Bits 31:14 Reserved, must be kept at reset value.

Bit 13 **TERST**: *Timer E counter software reset*
Refer to TARST description

Bit 12 **TDRST**: *Timer D counter software reset*
Refer to TARST description

Bit 11 **TCRST**: *Timer C counter software reset*
Refer to TARST description

Bit 10 **TBRST**: *Timer B counter software reset*
Refer to TARST description

Bit 9 **TARST**: *Timer A counter software reset*
Setting this bit resets the TimerA counter.
The bit is automatically reset by hardware.

Bit 8 **MRST**: Master *Counter software reset*
Setting this bit resets the Master timer counter.
The bit is automatically reset by hardware.

Bits 7:6 Reserved, must be kept at reset value.

Bit 5 **TESWU**: *Timer E Software Update*
Refer to TASWU description

Bit 4 **TDSWU**: *Timer D Software Update*
Refer to TASWU description

Bit 3 **TCSWU**: *Timer C Software Update*
Refer to TASWU description

Bit 2 **TBSWU**: *Timer B Software Update*
Refer to TASWU description

Bit 1 **TASWU**: *Timer A Software update*
This bit is set by software and automatically reset by hardware. It forces an immediate transfer from the preload to the active register and any pending update request is cancelled.

Bit 0 **MSWU**: *Master Timer Software update*
This bit is set by software and automatically reset by hardware. It forces an immediate transfer from the preload to the active register in the master timer and any pending update request is cancelled.

21.5.41 HRTIM Interrupt Status Register (HRTIM_ISR)

Address offset: 0x388h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER	DLLRDY
														r	r
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
											r	r	r	r	r

Bits 31:18 Reserved, must be kept at reset value.

Bit 17 **BMPER**: Burst mode Period Interrupt Flag

This bit is set by hardware when a single-shot burst mode operation is completed or at the end of a burst mode period in continuous mode. It is cleared by software writing it at 1.

0: No Burst mode period interrupt occurred

1: Burst mode period interrupt occurred

Bit 16 **DLLRDY**: DLL Ready Interrupt Flag

This bit is set by hardware when the DLL calibration is completed. It is cleared by software writing it at 1.

0: No DLL calibration ready interrupt occurred

1: DLL calibration ready interrupt occurred

Bits 15:6 Reserved, must be kept at reset value.

Bit 5 **SYSFLT**: System Fault Interrupt Flag

Refer to FLT1 description

Bit 4 **FLT5**: Fault 5 Interrupt Flag

Refer to FLT1 description

Bit 3 **FLT4**: Fault 4 Interrupt Flag

Refer to FLT1 description

Bit 2 **FLT3**: Fault 3 Interrupt Flag

Refer to FLT1 description

Bit 1 **FLT2**: Fault 2 Interrupt Flag

Refer to FLT1 description

Bit 0 **FLT1**: Fault 1 Interrupt Flag

This bit is set by hardware when Fault 1 event occurs. It is cleared by software writing it at 1.

0: No Fault 1 interrupt occurred

1: Fault 1 interrupt occurred

21.5.42 HRTIM Interrupt Clear Register (HRTIM_ICR)

Address offset: 0x38Ch

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	DLLRDYC
														w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
											w	w	w	w	w

Bits 31:18 Reserved, must be kept at reset value.

Bit 17 **BMPERC**: Burst mode period flag Clear

Writing 1 to this bit clears the BMPER flag in HRTIM_ISR register.

Bit 16 **DLLRDYC**: DLL Ready Interrupt flag Clear

Writing 1 to this bit clears the DLLRDY flag in HRTIM_ISR register.

Bits 15:6 Reserved, must be kept at reset value.

Bit 5 **SYSFLTC**: System Fault Interrupt Flag Clear

Writing 1 to this bit clears the SYSFLT flag in HRTIM_ISR register.

Bit 4 **FLT5C**: Fault 5 Interrupt Flag Clear

Writing 1 to this bit clears the FLT5 flag in HRTIM_ISR register.

Bit 3 **FLT4C**: Fault 4 Interrupt Flag Clear

Writing 1 to this bit clears the FLT4 flag in HRTIM_ISR register.

Bit 2 **FLT3C**: Fault 3 Interrupt Flag Clear

Writing 1 to this bit clears the FLT3 flag in HRTIM_ISR register.

Bit 1 **FLT2C**: Fault 2 Interrupt Flag Clear

Writing 1 to this bit clears the FLT2 flag in HRTIM_ISR register.

Bit 0 **FLT1C**: Fault 1 Interrupt Flag Clear

Writing 1 to this bit clears the FLT1 flag in HRTIM_ISR register.

21.5.43 HRTIM Interrupt Enable Register (HRTIM_IER)

Address offset: 0x390h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERIE	DLLRDYIE
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
											rw	rw	rw	rw	rw

Bits 31:18 Reserved, must be kept at reset value.

Bit 17 **BMPERIE**: Burst mode period Interrupt Enable

This bit is set and cleared by software to enable/disable the Burst mode period interrupt.
0: Burst mode period interrupt disabled
1: Burst mode period interrupt enabled

Bit 16 **DLLRDYIE**: DLL Ready Interrupt Enable

This bit is set and cleared by software to enable/disable the DLL ready interrupt.
0: DLL ready interrupt disabled
1: DLL ready interrupt enabled

Bits 15:6 Reserved, must be kept at reset value.

Bit 5 **SYSFLTIE**: System Fault Interrupt Enable

Refer to FLT1IE description

Bit 4 **FLT5IE**: Fault 5 Interrupt Enable

Refer to FLT1IE description

Bit 3 **FLT4IE**: Fault 4 Interrupt Enable

Refer to FLT1IE description

Bit 2 **FLT3IE**: Fault 3 Interrupt Enable

Refer to FLT1IE description

Bit 1 **FLT2IE**: Fault 2 Interrupt Enable

Refer to FLT1IE description

Bit 0 **FLT1IE**: Fault 1 Interrupt Enable

This bit is set and cleared by software to enable/disable the Fault 1 interrupt.
0: Fault 1 interrupt disabled
1: Fault 1 interrupt enabled

21.5.44 HRTIM Output Enable Register (HRTIM_OENR)

Address offset: 0x394h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
						rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

Bits 31:10 Reserved, must be kept at reset value.

Bit 9 **TE2OEN**: Timer E Output 2 Enable
Refer to TA1OEN description

Bit 8 **TE1OEN**: Timer E Output 1 Enable
Refer to TA1OEN description

Bit 7 **TD2OEN**: Timer D Output 2 Enable
Refer to TA1OEN description

Bit 6 **TD1OEN**: Timer D Output 1 Enable
Refer to TA1OEN description

Bit 5 **TC2OEN**: Timer C Output 2 Enable
Refer to TA1OEN description

Bit 4 **TC1OEN**: Timer C Output 1 Enable
Refer to TA1OEN description

Bit 3 **TB2OEN**: Timer B Output 2 Enable
Refer to TA1OEN description

Bit 2 **TB1OEN**: Timer B Output 1 Enable
Refer to TA1OEN description

Bit 1 **TA2OEN**: Timer A Output 2 Enable
Refer to TA1OEN description

Bit 0 **TA1OEN**: Timer A Output 1 (HRTIM_CHA1) Enable
Setting this bit enables the Timer A output 1. Writing “0” has no effect.
Reading the bit returns the output enable/disable status.
This bit is cleared asynchronously by hardware as soon as the timer-related fault input(s) is (are) active.
0: output HRTIM_CHA1 disabled. The output is either in Fault or Idle state.
1: output HRTIM_CHA1 enabled

Note: The disable status corresponds to both idle and fault states. The output disable status is given by TA1ODS bit in the HRTIM_ODSR register.

21.5.45 HRTIM Output Disable Register (HRTIM_ODISR)

Address offset: 0x398h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2ODIS	TE1ODIS	TD2ODIS	TD1ODIS	TC2ODIS	TC1ODIS	TB2ODIS	TB1ODIS	TA2ODIS	TA1ODIS
						w	w	w	w	w	w	w	w	w	w

Bits 31:10 Reserved, must be kept at reset value.

Bit 9 **TE2ODIS**: Timer E Output 2 disable
Refer to TA1ODIS description

Bit 8 **TE1ODIS**: Timer E Output 1 disable
Refer to TA1ODIS description

Bit 7 **TD2ODIS**: Timer D Output 2 disable
Refer to TA1ODIS description

Bit 6 **TD1ODIS**: Timer D Output 1 disable
Refer to TA1ODIS description

Bit 5 **TC2ODIS**: Timer C Output 2 disable
Refer to TA1ODIS description

Bit 4 **TC1ODIS**: Timer C Output 1 disable
Refer to TA1ODIS description

Bit 3 **TB2ODIS**: Timer B Output 2 disable
Refer to TA1ODIS description

Bit 2 **TB1ODIS**: Timer B Output 1 disable
Refer to TA1ODIS description

Bit 1 **TA2ODIS**: Timer A Output 2 disable
Refer to TA1ODIS description

Bit 0 **TA1ODIS**: Timer A Output 1 (HRTIM_CHA1) disable
Setting this bit disables the Timer A output 1. The output enters the idle state, either from the run state or from the fault state.
Writing “0” has no effect.

21.5.46 HRTIM Output Disable Status Register (HRTIM_ODSR)

Address offset: 0x39Ch

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
						r	r	r	r	r	r	r	r	r	r

Bits 31:10 Reserved, must be kept at reset value.

Bit 9 **TE2ODS**: Timer E Output 2 disable status
Refer to TA1ODS description

Bit 8 **TE1ODS**: Timer E Output 1 disable status
Refer to TA1ODS description

Bit 7 **TD2ODS**: Timer D Output 2 disable status
Refer to TA1ODS description

Bit 6 **TD1ODS**: Timer D Output 1 disable status
Refer to TA1ODS description

Bit 5 **TC2ODS**: Timer C Output 2 disable status
Refer to TA1ODS description

Bit 4 **TC1ODS**: Timer C Output 1 disable status
Refer to TA1ODS description

Bit 3 **TB2ODS**: Timer B Output 2 disable status
Refer to TA1ODS description

Bit 2 **TB1ODS**: Timer B Output 1 disable status
Refer to TA1ODS description

Bit 1 **TA2ODS**: Timer A Output 2 disable status
Refer to TA1ODS description

Bit 0 **TA1ODS**: Timer A Output 1 disable status
Reading the bit returns the output disable status. It is not significant when the output is active (Tx1OEN or Tx2OEN = 1).
0: output HRTIM_CHA1 disabled, in Idle state.
1: output HRTIM_CHA1 disabled, in Fault state.

21.5.47 HRTIM Burst Mode Control Register (HRTIM_BMCR)

Address offset: 0x3A0h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEBM	TDBM	TCBM	TBBM	TABM	MTBM
rc_w0										rw	rw	rw	rw	rw	rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **BMSTAT**: *Burst Mode Status*

This bit gives the current operating state.

0: Normal operation

1: Burst operation on-going. Writing this bit to 0 causes a burst mode early termination.

Bits 30:22 Reserved, must be kept at reset value.

Bit 21 **TEBM**: *Timer E Burst Mode*

Refer to TABM description

Bit 20 **TDBM**: *Timer D Burst Mode*

Refer to TABM description

Bit 19 **TCBM**: *Timer C Burst Mode*

Refer to TABM description

Bit 18 **TBBM**: *Timer B Burst Mode*

Refer to TABM description

Bit 17 **TABM**: *Timer A Burst Mode*

This bit defines how the timer behaves during a burst mode operation. This bitfield cannot be changed while the burst mode is enabled.

0: Timer A counter clock is maintained and the timer operates normally

1: Timer A counter clock is stopped and the counter is reset

Note: This bit must not be set when the balanced idle mode is active (DLYPRT[2:0] = 0x11)

Bit 16 **MTBM**: *Master Timer Burst Mode*

This bit defines how the timer behaves during a burst mode operation. This bitfield cannot be changed while the burst mode is enabled.

0: Master Timer counter clock is maintained and the timer operates normally

1: Master Timer counter clock is stopped and the counter is reset

Bits 15:11 Reserved, must be kept at reset value.

Bit 10 **BMPREN**: *Burst Mode Preload Enable*

This bit enables the registers preload mechanism and defines whether a write access into a preloadable register (HRTIM_BMCMPR, HRTIM_BMPER) is done into the active or the preload register.

0: Preload disabled: the write access is directly done into active registers

1: Preload enabled: the write access is done into preload registers

Bits 9:6 **BMPRSC[3:0]**: *Burst Mode Prescaler*

Defines the prescaling ratio of the f_{HRTIM} clock for the burst mode controller. This bitfield cannot be changed while the burst mode is enabled.

0000: Clock not divided
0001: Division by 2
0010: Division by 4
0011: Division by 8
0100: Division by 16
0101: Division by 32
0110: Division by 64
0111: Division by 128
1000: Division by 256
1001: Division by 512
1010: Division by 1024
1011: Division by 2048
1100: Division by 4096
1101:Division by 8192
1110: Division by 16384
1111: Division by 32768

Bits 5:2 **BMCLK[3:0]**: *Burst Mode Clock source*

This bitfield defines the clock source for the burst mode counter. It cannot be changed while the burst mode is enabled (refer to *Table 98* for on-chip events 1..4 connections details).

0000: Master timer counter reset/roll-over
0001: Timer A counter reset/roll-over
0010: Timer B counter reset/roll-over
0011: Timer C counter reset/roll-over
0100: Timer D counter reset/roll-over
0101: Timer E counter reset/roll-over
0110: On-chip Event 1 (BMClk[1]), acting as a burst mode counter clock
0111: On-chip Event 2 (BMClk[2]) acting as a burst mode counter clock
1000: On-chip Event 3 (BMClk[3]) acting as a burst mode counter clock
1001: On-chip Event 4 (BMClk[4]) acting as a burst mode counter clock
1010: Prescaled f_{HRTIM} clock (as per BMPRSC[3:0] setting)
Other codes reserved

Bit 1 **BMOM**: *Burst Mode operating mode*

This bit defines if the burst mode is entered once or if it is continuously operating.

0: Single-shot mode
1: Continuous operation

Bit 0 **BME**: *Burst Mode enable*

This bit starts the burst mode controller which becomes ready to receive the start trigger.

Writing this bit to 0 causes a burst mode early termination.

0: Burst mode disabled
1: Burst mode enabled

21.5.48 HRTIM Burst Mode Trigger Register (HRTIM_BMTRGR)

Address offset: 0x3A4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OCHPEV	EEV8	EEV7	TDEEV8	TAEEV7	TECMP2	TECMP1	TEREP	TERST	TDCMP2	TDCMP1	TDREP	TDRST	TCCMP2	TCCMP1	TCREP
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **OCHPEV**: *On-chip Event*

A rising edge on an on-chip Event (see *Section : Burst mode triggers*) triggers a burst mode entry.

Bit 30 **EEV8**: *External Event 8 (TIMD filters applied)*

The external event 8 conditioned by TIMD filters is starting the burst mode operation.

Bit 29 **EEV7**: *External Event 7 (TIMA filters applied)*

The external event 7 conditioned by TIMA filters is starting the burst mode operation.

Bit 28 **TDEEV8**: *Timer D period following External Event 8*

The timer D period following an external event 8 (conditioned by TIMD filters) is starting the burst mode operation.

Bit 27 **TAEEV7**: *Timer A period following External Event 7*

The timer A period following an external event 7 (conditioned by TIMA filters) is starting the burst mode operation.

Bit 26 **TECMP2**: *Timer E Compare 2 event*

Refer to TACMP1 description

Bit 25 **TECMP1**: *Timer E Compare 1 event*

Refer to TACMP1 description

Bit 24 **TEREP**: *Timer E repetition*

Refer to TAREP description

Bit 23 **TERST**: *Timer E counter reset or roll-over*

Refer to TARST description

Bit 22 **TDCMP2**: *Timer D Compare 2 event*

Refer to TACMP1 description

Bit 21 **TDCMP1**: *Timer D Compare 1 event*

Refer to TACMP1 description

Bit 20 **TDREP**: *Timer D repetition*

Refer to TAREP description

Bit 19 **TDRST**: *Timer D reset or roll-over*

Refer to TARST description

Bit 18 **TCCMP2**: *Timer C Compare 2 event*

Refer to TACMP1 description

Bit 17 **TCCMP1**: *Timer C Compare 1 event*
Refer to TACMP1 description

Bit 16 **TCREP**: *Timer C repetition*
Refer to TAREP description

Bit 15 **TCRST**: *Timer C reset or roll-over*
Refer to TARST description

Bit 14 **TBCMP2**: *Timer B Compare 2 event*
Refer to TACMP1 description

Bit 13 **TBCMP1**: *Timer B Compare 1 event*
Refer to TACMP1 description

Bit 12 **TBREP**: *Timer B repetition*
Refer to TAREP description

Bit 11 **TBRST**: *Timer B reset or roll-over*
Refer to TARST description

Bit 10 **TACMP2**: *Timer A Compare 2 event*
Refer to TACMP1 description

Bit 9 **TACMP1**: *Timer A Compare 1 event*
The timer A compare 1 event is starting the burst mode operation.

Bit 8 **TAREP**: *Timer A repetition*
The Timer A repetition event is starting the burst mode operation.

Bit 7 **TARST**: *Timer A reset or roll-over*
The Timer A reset or roll-over event is starting the burst mode operation.

Bit 6 **MSTCMP4**: *Master Compare 4*
Refer to MSTCMP1 description

Bit 5 **MSTCMP3**: *Master Compare 3*
Refer to MSTCMP1 description

Bit 4 **MSTCMP2**: *Master Compare 2*
Refer to MSTCMP1 description

Bit 3 **MSTCMP1**: *Master Compare 1*
The master timer Compare 1 event is starting the burst mode operation.

Bit 2 **MSTREP**: *Master repetition*
The master timer repetition event is starting the burst mode operation.

Bit 1 **MSTRST**: *Master reset or roll-over*
The master timer reset and roll-over event is starting the burst mode operation.

Bit 0 **SW**: *Software start*
This bit is set by software and automatically reset by hardware.
When set, It starts the burst mode operation immediately.
This bit is not active if the burst mode is not enabled (BME bit is reset).

21.5.49 HRTIM Burst Mode Compare Register (HRTIM_BMCMPR)

Address offset: 0x3A8h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BMCMP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **BMCMP[15:0]**: *Burst mode compare value*

Defines the number of periods during which the selected timers are in idle state.

This register holds either the content of the preload register or the content of the active register if the preload is disabled.

Note: BMCMP[15:0] cannot be set to 0x0000 when using the f_{HRTIM} clock without a prescaler as the burst mode clock source (BMCLK[3:0] = 1010 and BMPRESC[3:0] = 0000).

21.5.50 HRTIM Burst Mode Period Register (HRTIM_BMPER)

Address offset: 0x3ACh

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BMPER[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 Reserved, must be kept at reset value.

Bits 15:0 **BMPER[15:0]**: *Burst mode Period*

Defines the burst mode repetition period.

This register holds either the content of the preload register or the content of the active register if preload is disabled.

Note: The BMPER[15:0] must not be null when the burst mode is enabled.

21.5.51 HRTIM Timer External Event Control Register 1 (HRTIM_EECR1)

Address offset: 0x3B0h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1]
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE3SNS[0]	EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:30 Reserved, must be kept at reset value.

Bit 29 **EE5FAST**: *External Event 5 Fast mode*

Refer to EE1FAST description

Bits 28:27 **EE5SNS[1:0]**: *External Event 5 Sensitivity*

Refer to EE1SNS[1:0] description

Bit 26 **EE5POL**: *External Event 5 Polarity*

Refer to EE1POL description

Bits 25:24 **EE5SRC[1:0]**: *External Event 5 Source*

Refer to EE1SRC[1:0] description

Bit 23 **EE4FAST**: *External Event 4 Fast mode*

Refer to EE1FAST description

Bits 22:21 **EE4SNS[1:0]**: *External Event 4 Sensitivity*

Refer to EE1SNS[1:0] description

Bit 20 **EE4POL**: *External Event 4 Polarity*

Refer to EE1POL description

Bits 19:18 **EE4SRC[1:0]**: *External Event 4 Source*

Refer to EE1SRC[1:0] description

Bit 17 **EE3FAST**: *External Event 3 Fast mode*

Refer to EE1FAST description

Bits 16:15 **EE3SNS[1:0]**: *External Event 3 Sensitivity*

Refer to EE1SNS[1:0] description

Bit 14 **EE3POL**: *External Event 3 Polarity*

Refer to EE1POL description

Bits 13:12 **EE3SRC[1:0]**: *External Event 3 Source*

Refer to EE1SRC[1:0] description

Bit 11 **EE2FAST**: *External Event 2 Fast mode*

Refer to EE1FAST description

Bits 10:9 **EE2SNS[1:0]**: *External Event 2 Sensitivity*

Refer to EE1SNS[1:0] description

Bit 8 **EE2POL**: *External Event 2 Polarity*

Refer to EE1POL description

Bits 7:6 **EE2SRC[1:0]**: *External Event 2 Source*

Refer to EE1SRC[1:0] description

Bit 5 **EE1FAST**: *External Event 1 Fast mode*

0: External Event 1 is re-synchronized by the HRTIM logic before acting on outputs, which adds a f_{HRTIM} clock-related latency

1: External Event 1 is acting asynchronously on outputs (low latency mode)

Note: This bit must not be modified once the counter in which the event is used is enabled (TxCEN bit set)

Bits 4:3 **EE1SNS[1:0]**: *External Event 1 Sensitivity*

00: On active level defined by EE1POL bit

01: Rising edge, whatever EE1POL bit value

10: Falling edge, whatever EE1POL bit value

11: Both edges, whatever EE1POL bit value

Bit 2 **EE1POL**: *External Event 1 Polarity*

This bit is only significant if EE1SNS[1:0] = 00.

0: External event is active high

1: External event is active low

Note: This parameter cannot be changed once the timer x is enabled. It must be configured prior to setting EE1FAST bit.

Bits 1:0 **EE1SRC[1:0]**: *External Event 1 Source*

00: EE1Src1

01: EE1Src2

10: EE1Src3

11: EE1Src4

Note: This parameter cannot be changed once the timer x is enabled. It must be configured prior to setting EE1FAST bit.

21.5.52 HRTIM Timer External Event Control Register 2 (HRTIM_EECR2)

Address offset: 0x3B4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS[1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS[1]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
EE8SNS[0]	EE8POL	EE8SRC[1:0]		Res.	EE7SNS[1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS[1:0]		EE6POL	EE6SRC[1:0]	
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

Bits 31:29 Reserved, must be kept at reset value.

Bits 28:27 **EE10SNS[1:0]**: *External Event 10 Sensitivity*
Refer to EE1SNS[1:0] description

Bit 26 **EE10POL**: *External Event 10 Polarity*
Refer to EE1POL description

Bits 25:24 **EE10SRC[1:0]**: *External Event 10 Source*
Refer to EE1SRC[1:0] description

Bit 23 Reserved, must be kept at reset value.

Bits 22:21 **EE9SNS[1:0]**: *External Event 9 Sensitivity*
Refer to EE1SNS[1:0] description

Bit 20 **EE9POL**: *External Event 9 Polarity*
Refer to EE1POL description

Bits 19:18 **EE9SRC[1:0]**: *External Event 9 Source*
Refer to EE1SRC[1:0] description

Bit 17 Reserved, must be kept at reset value.

Bits 16:15 **EE8SNS[1:0]**: *External Event 8 Sensitivity*
Refer to EE1SNS[1:0] description

Bit 14 **EE8POL**: *External Event 8 Polarity*
Refer to EE1POL description

Bits 13:12 **EE8SRC[1:0]**: *External Event 8 Source*
Refer to EE1SRC[1:0] description

Bit 11 Reserved, must be kept at reset value.

Bits 10:9 **EE7SNS[1:0]**: *External Event 7 Sensitivity*
Refer to EE1SNS[1:0] description

Bit 8 **EE7POL**: *External Event 7 Polarity*
Refer to EE1POL description

Bits 7:6 **EE7SRC[1:0]**: *External Event 7 Source*
Refer to EE1SRC[1:0] description

Bit 5 Reserved, must be kept at reset value.

Bits 4:3 **EE6SNS[1:0]**: *External Event 6 Sensitivity*

Refer to EE1SNS[1:0] description

Bit 2 **EE6POL**: *External Event 6 Polarity*

Refer to EE1POL description

Bits 1:0 **EE6SRC[1:0]**: *External Event 6 Source*

Refer to EE1SRC[1:0] description

21.5.53 HRTIM Timer External Event Control Register 3 (HRTIM_EECR3)

Address offset: 0x3B8h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EEVSD[1:0]		Res.	Res.	EE10F[3:0]				Res.	Res.	EE9F[3:0]				Res.	Res.
rw	rw			rw	rw	rw	rw			rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8F[3:0]				Res.	Res.	EE7F[3:0]				Res.	Res.	EE6F[3:0]			
rw	rw	rw	rw			rw	rw	rw	rw			rw	rw	rw	rw

Bits 31:30 **EEVSD[1:0]**: External Event Sampling clock division

This bitfield indicates the division ratio between the timer clock frequency (f_{HRTIM}) and the External Event signal sampling clock (f_{EEVS}) used by the digital filters.

00: f_{EEVS}=f_{HRTIM}
01: f_{EEVS}=f_{HRTIM} / 2
10: f_{EEVS}=f_{HRTIM} / 4
11: f_{EEVS}=f_{HRTIM} / 8

Bits 29:28 Reserved, must be kept at reset value.

Bits 27:24 **EE10F[3:0]**: External Event 10 filter

Refer to EE6F[3:0] description

Bits 23:22 Reserved, must be kept at reset value.

Bits 21:18 **EE9F[3:0]**: External Event 9 filter

Refer to EE6F[3:0] description

Bits 17:16 Reserved, must be kept at reset value.

Bits 15:12 **EE8F[3:0]**: External Event 8 filter

Refer to EE6F[3:0] description

Bits 11:10 Reserved, must be kept at reset value.

Bits 9:6 **EE7F[3:0]**: External Event 7 filter

Refer to EE6F[3:0] description

Bits 4:5 Reserved, must be kept at reset value.

Bits 3:0 **EE6F[3:0]**: External Event 6 filter

This bitfield defines the frequency used to sample External Event 6 input and the length of the digital filter applied to EEV6. The digital filter is made of a counter in which N valid samples are needed to validate a transition on the output.

0000: Filter disabled
0001: $f_{SAMPLING}$= f_{HRTIM}, N=2
0010: $f_{SAMPLING}$= f_{HRTIM}, N=4
0011: $f_{SAMPLING}$= f_{HRTIM}, N=8
0100: $f_{SAMPLING}$= f_{EEVS}/2, N=6
0101: $f_{SAMPLING}$= f_{EEVS}/2, N=8
0110: $f_{SAMPLING}$= f_{EEVS}/4, N=6
0111: $f_{SAMPLING}$= f_{EEVS}/4, N=8
1000: $f_{SAMPLING}$= f_{EEVS}/8, N=6
1001: $f_{SAMPLING}$= f_{EEVS}/8, N=8
1010: $f_{SAMPLING}$= f_{EEVS}/16, N=5
1011: $f_{SAMPLING}$= f_{EEVS}/16, N=6
1100: $f_{SAMPLING}$= f_{EEVS}/16, N=8
1101: $f_{SAMPLING}$= f_{EEVS}/32, N=5
1110: $f_{SAMPLING}$= f_{EEVS}/32, N=6
1111: $f_{SAMPLING}$= f_{EEVS}/32, N=8

21.5.54 HRTIM ADC Trigger 1 Register (HRTIM_ADC1R)

Address offset: 0x3BCh

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AD1TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 These bits select the trigger source for th ADC Trigger 1 output . Refer to HRTIM_ADC3R bits description for details

21.5.55 HRTIM ADC Trigger 2 Register (HRTIM_ADC2R)

Address offset: 0x3C0h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 These bits select the trigger source for th ADC Trigger 2 output . Refer to HRTIM_ADC4R bits description for details

21.5.56 HRTIM ADC Trigger 3 Register (HRTIM_ADC3R)

Address offset: 0x3C4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC3TEPER	ADC3TEC4	ADC3TEC3	ADC3TEC2	ADC3TDPER	ADC3TDC4	ADC3TDC3	ADC3TDC2	ADC3TCPER	ADC3TCC4	ADC3TCC3	ADC3TCC2	ADC3TBRST	ADC3TBPER	ADC3TBC4	ADC3TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3TBC2	ADC3TARST	ADC3TAPER	ADC3TAC4	ADC3TAC3	ADC3TAC2	ADC3EEV5	ADC3EEV4	ADC3EEV3	ADC3EEV2	ADC3EEV1	ADC3MPER	ADC3MC4	ADC3MC3	ADC3MC2	ADC3MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **ADC3TEPER**: *ADC trigger 3 on Timer E Period*
Refer to ADC3TAPER description

Bit 30 **ADC3TEC4**: *ADC trigger 3 on Timer E Compare 4*
Refer to ADC3TAC2 description

Bit 29 **ADC3TEC3**: *ADC trigger 3 on Timer E Compare 3*
Refer to ADC3TAC2 description

Bit 28 **ADC3TEC2**: *ADC trigger 3 on Timer E Compare 2*
Refer to ADC3TAC2 description

Bit 27 **ADC3TDPER**: *ADC trigger 3 on Timer D Period*
Refer to ADC3TAPER description

Bit 26 **ADC3TDC4**: *ADC trigger 3 on Timer D Compare 4*
Refer to ADC3TAC2 description

Bit 25 **ADC3TDC3**: *ADC trigger 3 on Timer D Compare 3*
Refer to ADC3TAC2 description

Bit 24 **ADC3TDC2**: *ADC trigger 3 on Timer D Compare 2*
Refer to ADC3TAC2 description

Bit 23 **ADC3TCPER**: *ADC trigger 3 on Timer C Period*
Refer to ADC3TAPER description

Bit 22 **ADC3TCC4**: *ADC trigger 3 on Timer C Compare 4*
Refer to ADC3TAC2 description

Bit 21 **ADC3TCC3**: *ADC trigger 3 on Timer C Compare 3*
Refer to ADC3TAC2 description

Bit 20 **ADC3TCC2**: *ADC trigger 3 on Timer C Compare 2*
Refer to ADC3TAC2 description

Bit 19 **ADC3TBRST**: *ADC trigger 3 on Timer B Reset and counter roll-over*
Refer to ADC3TBRST description

Bit 18 **ADC3TBPER**: *ADC trigger 3 on Timer B Period*
Refer to ADC3TAPER description

Bit 17 **ADC3TBC4**: *ADC trigger 3 on Timer B Compare 4*
Refer to ADC3TAC2 description

Bit 16 **ADC3TBC3**: *ADC trigger 3 on Timer B Compare 3*
Refer to ADC3TAC2 description

Bit 15 **ADC3TBC2**: *ADC trigger 3 on Timer B Compare 2*
Refer to ADC3TAC2 description

Bit 14 **ADC3TARST**: *ADC trigger 3 on Timer A Reset and counter roll-over*
This bit enables the generation of an ADC Trigger upon Timer A reset and roll-over event, on ADC Trigger 1 output.

Bit 13 **ADC3TAPER**: *ADC trigger 3 on Timer A Period*
This bit enables the generation of an ADC Trigger upon Timer A period event, on ADC Trigger 3 output.

Bit 12 **ADC3TAC4**: *ADC trigger 3 on Timer A Compare 4*
Refer to ADC3TAC2 description

Bit 11 **ADC3TAC3**: *ADC trigger 3 on Timer A Compare 3*
Refer to ADC3TAC2 description

Bit 10 **ADC3TAC2**: *ADC trigger 3 on Timer A Compare 2*
This bit enables the generation of an ADC Trigger upon Timer A Compare 2 event, on ADC Trigger 3 output.

Bit 9 **ADC3EEV5**: *ADC trigger 3 on External Event 5*
Refer to ADC3EEV1 description

Bit 8 **ADC3EEV4**: *ADC trigger 3 on External Event 4*
Refer to ADC3EEV1 description

Bit 7 **ADC3EEV3**: *ADC trigger 3 on External Event 3*
Refer to ADC3EEV1 description

Bit 6 **ADC3EEV2**: *ADC trigger 3 on External Event 2*
Refer to ADC3EEV1 description

Bit 5 **ADC3EEV1**: *ADC trigger 3 on External Event 1*
This bit enables the generation of an ADC Trigger upon External event 1, on ADC Trigger 3 output.

Bit 4 **ADC3MPER**: *ADC trigger 3 on Master Period*
This bit enables the generation of an ADC Trigger upon Master timer period event, on ADC Trigger 3 output.

Bit 3 **ADC3MC4**: *ADC trigger 3 on Master Compare 4*
Refer to ADC3MC1 description

Bit 2 **ADC3MC3**: *ADC trigger 3 on Master Compare 3*
Refer to ADC3MC1 description

Bit 1 **ADC3MC2**: *ADC trigger 3 on Master Compare 2*
Refer to ADC3MC1 description

Bit 0 **ADC3MC1**: *ADC trigger 3 on Master Compare 1*
This bit enables the generation of an ADC Trigger upon Master Compare 1 event, on ADC Trigger 3 output.

21.5.57 HRTIM ADC Trigger 4 Register (HRTIM_ADC4R)

Address offset: 0x3C8h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC4T ERST	ADC4T EC4	ADC4T EC3	ADC4T EC2	ADC4T DRST	ADC4T DPER	ADC4T DC4	ADC4T DC3	ADC4T DC2	ADC4T CRST	ADC4T CPER	ADC4T CC4	ADC4T CC3	ADC4T CC2	ADC4T BPER	ADC4T BC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC4T BC3	ADC4T BC2	ADC4T APER	ADC4T AC4	ADC4T AC3	ADC4T AC2	ADC4E EV10	ADC4E EV9	ADC4E EV8	ADC4E EV7	ADC4E EV6	ADC4M PER	ADC4M C4	ADC4M C3	ADC4M C2	ADC4M C1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **ADC4TERST**: *ADC trigger 4 on Timer E Reset and counter roll-over* [1]

Refer to ADC4TCRST description

Bit 30 **ADC4TEC4**: *ADC trigger 4 on Timer E Compare 4*

Refer to ADC4TAC2 description

Bit 29 **ADC4TEC3**: *ADC trigger 4 on Timer E Compare 3*

Refer to ADC4TAC2 description

Bit 28 **ADC4TEC2**: *ADC trigger 4 on Timer E Compare 2*

Refer to ADC4TAC2 description

Bit 27 **ADC4TDRST**: *ADC trigger 4 on Timer D Reset and counter roll-over* [1]

Refer to ADC4TCRST description

Bit 26 **ADC4TDPER**: *ADC trigger 4 on Timer D Period*

Refer to ADC4TAPER description

Bit 25 **ADC4TDC4**: *ADC trigger 4 on Timer D Compare 4*

Refer to ADC4TAC2 description

Bit 24 **ADC4TDC3**: *ADC trigger 4 on Timer D Compare 3*

Refer to ADC4TAC2 description

Bit 23 **ADC4TDC2**: *ADC trigger 2 on Timer D Compare 2*

Refer to ADC4TAC2 description

Bit 22 **ADC4TCRST**: *ADC trigger 4 on Timer C Reset and counter roll-over* [1]

This bit enables the generation of an ADC Trigger upon Timer C reset and roll-over event, on ADC Trigger 4 output.

Bit 21 **ADC4TCPER**: *ADC trigger 4 on Timer C Period*

Refer to ADC4TAPER description

Bit 20 **ADC4TCC4**: *ADC trigger 4 on Timer C Compare 4*

Refer to ADC4TAC2 description

Bit 19 **ADC4TCC3**: *ADC trigger 4 on Timer C Compare 3*

Refer to ADC4TAC2 description

Bit 18 **ADC4TCC2**: *ADC trigger 4 on Timer C Compare 2*

Refer to ADC4TAC2 description

Bit 17 **ADC4TBPER**: *ADC trigger 4 on Timer B Period*
Refer to ADC4TAPER description

Bit 16 **ADC4TBC4**: *ADC trigger 4 on Timer B Compare 4*
Refer to ADC4TAC2 description

Bit 15 **ADC4TBC3**: *ADC trigger 4 on Timer B Compare 3*
Refer to ADC4TAC2 description

Bit 14 **ADC4TBC2**: *ADC trigger 4 on Timer B Compare 2*
Refer to ADC4TAC2 description

Bit 13 **ADC4TAPER**: *ADC trigger 4 on Timer A Period*
This bit enables the generation of an ADC Trigger upon Timer A event, on ADC Trigger 4 output.

Bit 12 **ADC4TAC4**: *ADC trigger 4 on Timer A Compare 4*
Refer to ADC4TAC2 description

Bit 11 **ADC4TAC3**: *ADC trigger 4 on Timer A Compare 3*
Refer to ADC4TAC2 description

Bit 10 **ADC4TAC2**: *ADC trigger 4 on Timer A Compare 2*
This bit enables the generation of an ADC Trigger upon Timer A Compare 2, on ADC Trigger 4 output.

Bit 9 **ADC4EEV10**: *ADC trigger 4 on External Event 10* [(1)]
Refer to ADC4EEV6 description

Bit 8 **ADC4EEV9**: *ADC trigger 4 on External Event 9* [(1)]
Refer to ADC4EEV6 description

Bit 7 **ADC4EEV8**: *ADC trigger 4 on External Event 8* [(1)]
Refer to ADC4EEV6 description

Bit 6 **ADC4EEV7**: *ADC trigger 4 on External Event 7* [(1)]
Refer to ADC4EEV6 description

Bit 5 **ADC4EEV6**: *ADC trigger 4 on External Event 6* [(1)]
This bit enables the generation of an ADC Trigger upon external event 6, on ADC Trigger 4 output.

Bit 4 **ADC4MPER**: *ADC trigger 4 on Master Period*
This bit enables the generation of an ADC Trigger upon Master period event, on ADC Trigger 4 output.

Bit 3 **ADC4MC4**: *ADC trigger 4 on Master Compare 4*
Refer to ADC4MC1 description

Bit 2 **ADC4MC3**: *ADC trigger 4 on Master Compare 3*
Refer to ADC4MC1 description

Bit 1 **ADC4MC2**: *ADC trigger 4 on Master Compare 2*
Refer to ADC4MC1 description

Bit 0 **ADC4MC1**: *ADC trigger 4 on Master Compare 1*
This bit enables the generation of an ADC Trigger upon Master Compare 1 event, on ADC Trigger 4 output.

1. These triggers are differing from HRTIM_ADC1R/HRTIM_ADC3R to HRTIM_ADC2R/HRTIM_ADC4R.

21.5.58 HRTIM DLL Control Register (HRTIM_DLLCR)

Address offset: 0x3CCh

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALRTE[1:0]		CALEN	CAL
												rw	rw	rw	wo

Bits 31:4 Reserved, must be kept at reset value

Bits 3:2 **CALRTE[1:0]**: DLL Calibration rate

This defines the DLL calibration periodicity.
00: 1048576 * t_{HRTIM} (7.3 ms)
01: 131072 * t_{HRTIM} (910 µs)
10: 16384 * t_{HRTIM} (114 µs)
11: 2048 * t_{HRTIM} (14 µs)

Bit 1 **CALEN**: DLL Calibration Enable

This bit enables the periodic DLL calibration, as per CALRTE[1:0] bit setting. When CALEN bit is reset, the calibration can be started in single-shot mode with CAL bit.
0: Periodic calibration disabled
1: Calibration is performed periodically, as per CALRTE[1:0] setting

Note: CALEN must not be set simultaneously with CAL bit

Bit 0 **CAL**: DLL Calibration Start

This bit starts the DLL calibration process. It is write-only.
0: No calibration request
1: Calibration start

Note: CAL must not be set simultaneously with CALEN bit

21.5.59 HRTIM Fault Input Register 1 (HRTIM_FLTINR1)

Address offset: 0x3D0h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLT4LCK	FLT4F[3:0]				FLT4SRC	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC	FLT3P	FLT3E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT2LCK	FLT2F[3:0]				FLT2SRC	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC	FLT1P	FLT1E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit 31 **FLT4LCK**: Fault 4 Lock
Refer to FLT5LCK description in HRTIM_FLTINR2 register

Bits 30:27 **FLT4F[3:0]**: Fault 4 filter
Refer to FLT5F[3:0] description in HRTIM_FLTINR2 register

Bit 26 **FLT4SRC**: Fault 4 source
Refer to FLT5SRC description in HRTIM_FLTINR2 register

Bit 25 **FLT4P**: Fault 4 polarity
Refer to FLT5P description in HRTIM_FLTINR2 register

Bit 24 **FLT4E**: Fault 4 enable
Refer to FLT5E description in HRTIM_FLTINR2 register

Bit 23 **FLT3LCK**: Fault 3 Lock
Refer to FLT5LCK description in HRTIM_FLTINR2 register

Bits 22:19 **FLT3F[3:0]**: Fault 3 filter
Refer to FLT5F[3:0] description in HRTIM_FLTINR2 register

Bit 18 **FLT3SRC**: Fault 3 source
Refer to FLT5SRC description in HRTIM_FLTINR2 register

Bit 17 **FLT3P**: Fault 3 polarity
Refer to FLT5P description in HRTIM_FLTINR2 register

Bit 16 **FLT3E**: Fault 3 enable
Refer to FLT5E description in HRTIM_FLTINR2 register

Bit 15 **FLT2LCK**: Fault 2 Lock
Refer to FLT5LCK description in HRTIM_FLTINR2 register

Bits 14:11 **FLT2F[3:0]**: Fault 2 filter
Refer to FLT5F[3:0] description in HRTIM_FLTINR2 register

Bit 10 **FLT2SRC**: Fault 2 source
Refer to FLT5SRC description in HRTIM_FLTINR2 register

Bit 9 **FLT2P**: Fault 2 polarity
Refer to FLT2P description in HRTIM_FLTINR2 register

Bit 8 **FLT2E**: Fault 2 enable
Refer to FLT5E description in HRTIM_FLTINR2 register

Bit 7 **FLT1LCK**: Fault 1 Lock
Refer to FLT5LCK description in HRTIM_FLTINR2 register

Bits 6:3 **FLT1F[3:0]**: Fault 1 filter
Refer to FLT5F[3:0] description in HRTIM_FLTINR2 register

Bit 2 **FLT1SRC**: Fault 1 source
Refer to FLT5SRC description in HRTIM_FLTINR2 register

Bit 1 **FLT1P**: Fault 1 polarity
Refer to FLT5P description in HRTIM_FLTINR2 register

Bit 0 **FLT1E**: Fault 1 enable
Refer to FLT5E description in HRTIM_FLTINR2 register

21.5.60 HRTIM Fault Input Register 2 (HRTIM_FLTINR2)

Address offset: 0x3D4h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5LCK	FLT5F[3:0]				FLT5SRC	FLT5P	FLT5E
								rwo	rw	rw	rw	rw	rw	rw	rw

Bits 31:26 Reserved, must be kept at reset value.

Bits 25:24 **FLTSD[1:0]**: Fault Sampling clock division

This bitfield indicates the division ratio between the timer clock frequency (f_{HRTIM}) and the fault signal sampling clock (f_{FLTS}) used by the digital filters.

00: $f_{FLTS}=f_{HRTIM}$

01: $f_{FLTS}=f_{HRTIM} / 2$

10: $f_{FLTS}=f_{HRTIM} / 4$

11: $f_{FLTS}=f_{HRTIM} / 8$

Note: This bitfield must be written prior to any of the FLTxE enable bits.

Bits 23:8 Reserved, must be kept at reset value.

Bit 7 **FLT5LCK**: Fault 5 Lock

The FLT5LCK bit modifies the write attributes of the fault programming bit, so that they can be protected against spurious write accesses.

This bit is write-once. Once it has been set, it cannot be modified till the next system reset.

0: FLT5E, FLT5P, FLT5SRC, FLT5F[3:0] bits are read/write.

1: FLT5E, FLT5P, FLT5SRC, FLT5F[3:0] bits can no longer be written (read-only mode)

Bits 6:3 **FLT5F[3:0]**: Fault 5 filter

This bitfield defines the frequency used to sample FLT5 input and the length of the digital filter applied to FLT5. The digital filter is made of an event counter in which N events are needed to validate a transition on the output:

0000: No filter, FLT5 acts asynchronously
0001: $f_{SAMPLING} = f_{HRTIM}$, N = 2
0010: $f_{SAMPLING} = f_{HRTIM}$, N = 4
0011: $f_{SAMPLING} = f_{HRTIM}$, N = 8
0100: $f_{SAMPLING} = f_{FLTS}/2$, N = 6
0101: $f_{SAMPLING} = f_{FLTS}/2$, N = 8
0110: $f_{SAMPLING} = f_{FLTS}/4$, N = 6
0111: $f_{SAMPLING} = f_{FLTS}/4$, N = 8
1000: $f_{SAMPLING} = f_{FLTS}/8$, N = 6
1001: $f_{SAMPLING} = f_{FLTS}/8$, N = 8
1010: $f_{SAMPLING} = f_{FLTS}/16$, N = 5
1011: $f_{SAMPLING} = f_{FLTS}/16$, N = 6
1100: $f_{SAMPLING} = f_{FLTS}/16$, N = 8
1101: $f_{SAMPLING} = f_{FLTS}/32$, N = 5
1110: $f_{SAMPLING} = f_{FLTS}/32$, N = 6
1111: $f_{SAMPLING} = f_{FLTS}/32$, N = 8

Note: This bitfield can be written only when FLT5E enable bit is reset.
This bitfield cannot be modified when FLT5LOCK has been programmed.

Bit 2 **FLT5SRC**: Fault 5 source

This bit selects the FAULT5 input source (refer to *Table 99* for connection details).

0: Fault 1 input is HRTIM_FLT5 input pin

1: Fault 1 input is FLT5_Int signal

Note: This bitfield can be written only when FLT5E enable bit is reset

Bit 1 **FLT5P**: Fault 5 polarity

This bit selects the FAULT5 input polarity.

0: Fault 5 input is active low

1: Fault 5 input is active high

Note: This bitfield can be written only when FLT5E enable bit is reset

Bit 0 **FLT5E**: Fault 5 enable

This bit enables the global FAULT5 input circuitry.

0: Fault 5 input disabled
1: Fault 5 input enabled

21.5.61 HRTIM Burst DMA Master timer update Register (HRTIM_BDMUPR)

Address offset: 0x3D8h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:10 Reserved, must be kept at reset value.

Bit 9 **MCMP4**: *MCMP4R register update enable*
Refer to MCR description

Bit 8 **MCMP3**: *MCMP3R register update enable*
Refer to MCR description

Bit 7 **MCMP2**: *MCMP2R register update enable*
Refer to MCR description

Bit 6 **MCMP1**: *MCMP1R register update enable*
Refer to MCR description

Bit 5 **MREP**: *MREP register update enable*
Refer to MCR description

Bit 4 **MPER**: *MPER register update enable*
Refer to MCR description

Bit 3 **MCNT**: *MCNTR register update enable*
Refer to MCR description

Bit 2 **MDIER**: *MDIER register update enable*
Refer to MCR description

Bit 1 **MICR**: *MICR register update enable*
Refer to MCR description

Bit 0 **MCR**: *MCR register update enable*
This bit defines if the master timer MCR register is part of the list of registers to be updated by the Burst DMA.
0: MCR register is not updated by Burst DMA accesses
1: MCR register is updated by Burst DMA accesses

21.5.62 HRTIM Burst DMA Timerx update Register (HRTIM_BDTxUPR)

Address offset: 0x3DCh-0x3ECh

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMxFLTR	TIMxOUTR	TIMxCHPR	TIMxRSTR	TIMxEEFR2
											rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMxEEFR1	TIMxRST2R	TIMxSET2R	TIMxRST1R	TIMxSET1R	TIMxDTxR	TIMxCMP4	TIMxCMP3	TIMxCMP2	TIMxCMP1	TIMxREP	TIMxPER	TIMxCNT	TIMxDIER	TIMxICR	TIMxCR
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:21 Reserved, must be kept at reset value.

Bit 20 **TIMxFLTR**: HRTIM_FLTxR register update enable
Refer to TIMxCR description

Bit 19 **TIMxOUTR**: HRTIM_OUTxR register update enable
Refer to TIMxCR description

Bit 18 **TIMxCHPR**: HRTIM_CHPxR register update enable
Refer to TIMxCR description

Bit 17 **TIMxRSTR**: HRTIM_RSTxR register update enable
Refer to TIMxCR description

Bit 16 **TIMxEEFR2**: HRTIM_EEFxR2 register update enable
Refer to TIMxCR description

Bit 15 **TIMxEEFR1**: HRTIM_EEFxR1 register update enable
Refer to TIMxCR description

Bit 14 **TIMxRST2R**: HRTIM_RST2xR register update enable
Refer to TIMxCR description

Bit 13 **TIMxSET2R**: HRTIM_SET2xR register update enable
Refer to TIMxCR description

Bit 12 **TIMxRST1R**: HRTIM_RST1xR register update enable
Refer to TIMxCR description

Bit 11 **TIMxSET1R**: HRTIM_SET1xR register update enable
Refer to TIMxCR description

Bit 10 **TIMxDTR**: HRTIM_DTxR register update enable
Refer to TIMxCR description

Bit 9 **TIMxCMP4**: HRTIM_CMP4xR register update enable
Refer to TIMxCR description

Bit 8 **TIMxCMP3**: HRTIM_CMP3xR register update enable
Refer to TIMxCR description

Bit 7 **TIMxCMP2**: HRTIM_CMP2xR register update enable
Refer to TIMxCR description

Bit 6 **TIMxCMP1**: HRTIM_CMP1xR register update enable
Refer to TIMxCR description

Bit 5 **TIMxREP**: HRTIM_REPxR register update enable
Refer to TIMxCR description

Bit 4 **TIMxPER**: HRTIM_PERxR register update enable
Refer to TIMxCR description

Bit 3 **TIMxCNT**: HRTIM_CNTxR register update enable
Refer to TIMxCR description

Bit 2 **TIMxDIER**: HRTIM_TIMxDIER register update enable
Refer to TIMxCR description

Bit 1 **TIMxICR**: HRTIM_TIMxICR register update enable
Refer to TIMxCR description

Bit 0 **TIMxCR**: HRTIM_TIMxCR register update enable
This bit defines if the master timer MCR register is part of the list of registers to be updated by the Burst DMA.
0: HRTIM_TIMxCR register is not updated by Burst DMA accesses
1: HRTIM_TIMxCR register is updated by Burst DMA accesses

21.5.63 HRTIM Burst DMA Data Register (HRTIM_BDMADR)

Address offset: 0x3F0h

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BDMADR[31:16]															
wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDMADR[15:0]															
wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo	wo

Bits 31:0 **BDMADR[31:0]**: Burst DMA Data register
Write accesses to this register triggers:

- the copy of the data value into the registers enabled in BDTxUPR and BDMUPR register bits
- the increment of the register pointer to the next location to be filled

21.5.64 HRTIM register map

The tables below summarize the HRTIM registers mapping. The address offsets in *Table 105* and *Table 106* are referred to in the base address offsets given in *Table 104*.

Table 104. RTIM global register map

Base address offset	Register
0x000 - 0x07F	Master timer
0x080 - 0x0FF	Timer A
0x100 - 0x17F	Timer B
0x180 - 0x1FF	Timer C
0x200 - 0x27F	Timer D
0x280 - 0x2FF	Timer E
0x300 - 0x37F	Reserved
0x380 - 0x3FF	Common registers

Table 105. HRTIM Register map and reset values: Master timer

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	**HRTIM_MCR**	BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	Res.	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN	SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		Res.	Res.	HALF	RETRIG	CONT	CKPSC[2:0]		
	Reset value	0	0	0		0	0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x0004	**HRTIM_MISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
	Reset value																										0	0	0	0	0	0	0
0x0008	**HRTIM_MICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
	Reset value																										0	0	0	0	0	0	0
0x000C	**HRTIM_ MDIER**[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
	Reset value										0	0	0	0	0	0	0										0	0	0	0	0	0	0
0x0010	**HRTIM_MCNT R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCNT[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 105. HRTIM Register map and reset values: Master timer (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0014	HRTIM_MPER[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPER[15:0]															
	Reset value																	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0x0018	HRTIM_MREP[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MREP[7:0]							
	Reset value																									0	0	0	0	0	0	0	0
0x001C	HRTIM_MCMP1R[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP1[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	Reserved	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	Reset value																																
0x0024	HRTIM_MCMP2R[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP2[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_MCMP3R[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP3[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	HRTIM_MCMP4R[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

[1] This register can be preloaded (see *Table 90 on page 673*).

Table 106. HRTIM Register map and reset values: TIMx (x= A..E)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	HRTIM_TIMxCR	UPDGAT [3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	Res.	TxRSTU	TxREPU	Res.	DELCMP4[1:0]		DELCMP2[1:0]		SYNCSTRTx	SYNCRSTx	Res.	Res.	Res.	PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0		0	0		0	0	0	0	0	0				0	0	0	0	0	0	0
0x0004	HRTIM_TIMxISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2STAT	O1STAT	IPPSTAT	CPPSTAT	Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	Reset value											0	0	0	0	0	0		0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x0008	HRTIM_TIMxICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	Reset value																	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x000C	HRTIM_TIMxDIER(1)	Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SET1xDE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE	Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	Reset value		0	0	0	0	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0		0	0	0	0	0
0x0010	HRTIM_CNTxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTx[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	HRTIM_PERxR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERx[15:0]															
	Reset value																	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0x0018	HRTIM_REPxR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
	Reset value																									0	0	0	0	0	0	0	0
0x001C	HRTIM_CMP1xR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP1x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	HRTIM_CMP1CxR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]								CMP1x[15:0]															
	Reset value									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	HRTIM_CMP2xR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP2x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_CMP3xR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP3x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	HRTIM_CMP4xR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP4x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	HRTIM_CPT1xR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPT1x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 106. HRTIM Register map and reset values: TIMx (x= A..E) (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0034	HRTIM_CPT2xR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CPT2x[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0038	HRTIM_DTxR(1)	DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]									DTRLKx	DTRSLKx	Res.	DTPRSC[2:0]			SDTRx	DTRx[8:0]								
	Reset value	0	0					0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x003C	HRTIM_SETx1R(1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	HRTIM_RSTx1R(1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	HRTIM_SETx2R(1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	HRTIM_RSTx2R(1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	HRTIM_EEFxR1	Res.	Res.	Res.	EE5FLTR[3:0]				EE5LTCH	Res.	EE4FLTR[3:0]				EE4LTCH	Res.	EE3FLTR[3:0]				EE3LTCH	Res.	EE2FLTR[3:0]				EE2LTCH	Res.	EE1FLTR[3:0]				EE1LTCH
	Reset value				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x0050	HRTIM_EEFxR2	Res.	Res.	Res.	EE10FLTR[3:0]				EE10LTCH	Res.	EE9FLTR[3:0]				EE9LTCH	Res.	EE8FLTR[3:0]				EE8LTCH	Res.	EE7FLTR[3:0]				EE7LTCH	Res.	EE6FLTR[3:0]				EE6LTCH
	Reset value				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x0054	HRTIM_RSTAR(1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	Reset value		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

Table 106. HRTIM Register map and reset values: TIMx (x= A..E) (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0054	HRTIM_ RSTBR(1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	Reset value		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_ RSTCR(1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	Reset value		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_ RSTDR(1)	Res.	TIMECMP4	TIMECMP2	TIMECMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	Reset value		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0054	HRTIM_ RSTER(1)	Res.	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	Res.
	Reset value		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x0058	HRTIM_CHPxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STRTPW [3:0]				CARDTY [2:0]			CARFRQ [3:0]			
	Reset value																						0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_ CPT1ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	Res.	Res.	Res.	Res.	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_ CPT1BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	Res.	Res.	Res.	Res.	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_ CPT1CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	Res.	Res.	Res.	Res.	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_ CPT1DCR	TECMP2	TECMP1	TE1RST	TE1SET	Res.	Res.	Res.	Res.	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 106. HRTIM Register map and reset values: TIMx (x= A..E) (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x005C	HRTIM_ CPT1ECR	Res.	Res.	Res.	Res.	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ CPT2ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	Res.	Res.	Res.	Res.	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ CPT2BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	Res.	Res.	Res.	Res.	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ CPT2CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	Res.	Res.	Res.	Res.	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ CPT2DCR	TECMP2	TECMP1	TE1RST	TE1SET	Res.	Res.	Res.	Res.	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ CPT2ECR	Res.	Res.	Res.	Res.	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	Reset value					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	HRTIM_OUTxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM2	POL2	Res.	Res.	Res.	Res.	DLYPRT[2:0]			DLYPRTEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM1	POL1	Res.
	Reset value									0	0	0	0	0	0	0									0	0	0	0	0	0	0	0	
0x0068	HRTIM_FLTxR	FLTLCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5EN	FLT4EN	FLT3EN	FLT2EN	FLT1EN
	Reset value	0																											0	0	0	0	0

1. This register can be preloaded (see *Table 90 on page 673*).

Table 107. HRTIM Register map and reset values: Common functions

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	HRTIM_CR1	Res.	Res.	Res.	Res.	AD4USRC[2:0]			AD3USRC[2:0]			AD2USRC[2:0]			AD1USRC[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEUDIS	TDUDIS	TCUDIS	TBUDIS	TAUDIS	MUDIS
	Reset value					0	0	0	0	0	0	0	0	0	0	0	0											0	0	0	0	0	0
0x0004	HRTIM_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	Res.	TESWU	TDSWU	TCSWU	TBSWU	TASWU	MSWU
	Reset value																			0	0	0	0	0	0			0	0	0	0	0	0
0x008	HRTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER	DLLRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
	Reset value															0	0											0	0	0	0	0	0
0x000C	HRTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	DLLRDYC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
	Reset value															0	0											0	0	0	0	0	0
0x0010	HRTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERIE	DLLRDYIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
	Reset value															0	0											0	0	0	0	0	0
0x0014	HRTIM_OENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
	Reset value																							0	0	0	0	0	0	0	0	0	0
0x0018	HRTIM_DISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2ODIS	TE1ODIS	TD2ODIS	TD1ODIS	TC2ODIS	TC1ODIS	TB2ODIS	TB1ODIS	TA2ODIS	TA1ODIS
	Reset value																							0	0	0	0	0	0	0	0	0	0
0x001C	HRTIM_ODSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
	Reset value																							0	0	0	0	0	0	0	0	0	0
0x0020	HRTIM_BMCR	BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TEBM	TDBM	TCBM	TBBM	TABM	MTBM	Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
	Reset value	0										0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x0024	HRTIM_BMTRG	OCHPEV	Res.	Res.	Res.	Res.	TECMP2	TECMP1	TEREP	TERST	TDCMP2	TDCMP1	TDREP	TDRST	TCCMP2	TCCMP1	TCREP	TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
	Reset value	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	HRTIM_ BMCMPR[1]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMCMP[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 107. HRTIM Register map and reset values: Common functions (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x002C	HRTIM_BMPER(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	HRTIM_EECR1	Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1:0]		EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
	Reset value			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0034	HRTIM_EECR2	Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS[1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS[1:0]		EE8POL	EE8SRC[1:0]		Res.	EE7SNS[1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS[1:0]		EE6POL	EE6SRC[1:0]	
	Reset value				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x0038	HRTIM_EECR3	Res.	Res.	Res.	EE10SNS[1:0]		EE10POL	EE10SRC[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EE6SRC[1:0]	
	Reset value				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x003C	HRTIM_ADC1R(1)	AD1TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3	AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0040	HRTIM_ADC2R(1)	AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4	AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	HRTIM_ADC3R(1)	ADC3TEPER	AD1TEC4	AD1TEC3	AD1TEC2	AD1TDPER	AD1TDC4	AD1TDC3	AD1TDC2	AD1TCPER	AD1TCC4	AD1TCC3	AD1TCC2	AD1TBRST	AD1TBPER	AD1TBC4	AD1TBC3	AD1TBC2	AD1TARST	AD1TAPER	AD1TAC4	AD1TAC3	AD1TAC2	AD1EEV5	AD1EEV4	AD1EEV3	AD1EEV2	AD1EEV1	AD1MPER	AD1MC4	AD1MC3	AD1MC2	AD1MC1
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0048	HRTIM_ADC4R(1)	AD2TERST	AD2TEC4	AD2TEC3	AD2TEC2	AD2TDRST	AD2TDPER	AD2TDC4	AD2TDC3	AD2TDC2	AD2TCRST	AD2TCPER	AD2TCC4	AD2TCC3	AD2TCC2	AD2TBPER	AD2TBC4	AD2TBC3	AD2TBC2	AD2TAPER	AD2TAC4	AD2TAC3	AD2TAC2	AD2EEV10	AD2EEV9	AD2EEV8	AD2EEV7	AD2EEV6	AD2MPER	AD2MC4	AD2MC3	AD2MC2	AD2MC1
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004C	HRTIM_DLLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALRTE [1:0]		CALEN	CAL
	Reset value																													0	0	0	0
0x0050	HRTIM_FLTINxR1	FLT4LCK	FLT4F[3:0]				FLT4SRC	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC	FLT3P	FLT3E	FLT2LCK	FLT2F[3:0]				FLT2SRC	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC	FLT1P	FLT1E
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Table 107. HRTIM Register map and reset values: Common functions (continued)

Offset	Register name	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0054	HRTIM_FLTINxR2	Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT5LCK	FLT5F[3:0]				FLT5SRC	FLT5P	FLT5E
	Reset value							0	0																	0	0	0	0	0	0	0	0
0x0058	HRTIM_ BDMUPDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
	Reset value																							0	0	0	0	0	0	0	0	0	0
0x005C	HRTIM_BDTAUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMAFLTR	TIMAOUTR	TIMACHPR	TIMARSTR	TIMAEEFR2	TIMAEEFR1	TIMARST2R	TIMASET2R	TIMARST1R	TIMASET1R	TIMADTxR	TIMACMP4	TIMACMP3	TIMACMP2	TIMACMP1	TIMAREP	TIMAPER	TIMACNT	TIMADIER	TIMAICR	TIMACR
	Reset value												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0060	HRTIM_ BDTBUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMBFLTR	TIMBOUTR	TIMBCHPR	TIMBRSTR	TIMBEEFR2	TIMBEEFR1	TIMBRST2R	TIMBSET2R	TIMBRST1R	TIMBSET1R	TIMBDTxR	TIMBCMP4	TIMBCMP3	TIMBCMP2	TIMBCMP1	TIMBREP	TIMBPER	TIMBCNT	TIMBDIER	TIMBICR	TIMBCR
	Reset value												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0064	HRTIM_ BDTCUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMCFLTR	TIMCOUTR	TIMCCHPR	TIMCRSTR	TIMCEEFR2	TIMCEEFR1	TIMCRST2R	TIMCSET2R	TIMCRST1R	TIMCSET1R	TIMCDTxR	TIMCCMP4	TIMCCMP3	TIMCCMP2	TIMCCMP1	TIMCREP	TIMCPER	TIMCCNT	TIMCDIER	TIMCICR	TIMCCR
	Reset value												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0068	HRTIM_ BDTDUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMDFLTR	TIMDOUTR	TIMDCHPR	TIMDRSTR	TIMDEEFR2	TIMDEEFR1	TIMDRST2R	TIMDSET2R	TIMDRST1R	TIMDSET1R	TIMDDTxR	TIMDCMP4	TIMDCMP3	TIMDCMP2	TIMDCMP1	TIMDREP	TIMDPER	TIMDCNT	TIMDDIER	TIMDICR	TIMDCR
	Reset value												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x006C	HRTIM_ BDTEUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEFLTR	TIMEOUTR	TIMECHPR	TIMERSTR	TIMEEEFR2	TIMEEEFR1	TIMERST2R	TIMESET2R	TIMERST1R	TIMESET1R	TIMEDTxR	TIMECMP4	TIMECMP3	TIMECMP2	TIMECMP1	TIMEREP	TIMEPER	TIMECNT	TIMEDIER	TIMEICR	TIMECR
	Reset value												0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0070	HRTIM_BDMADR	BDMADR[31:0]																															
	Reset value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. This register can be preloaded (see *Table 90 on page 673*).

Refer to *Section 2.2 on page 47* for the register boundary addresses.