CAIRO UNIVERSITY

STM32 MICROCONTROLLER: GENERAL-PURPOSE TIMERS (TIM2-TIM5)

Lecture 5

Prof. Yasser Mostafa Kadah

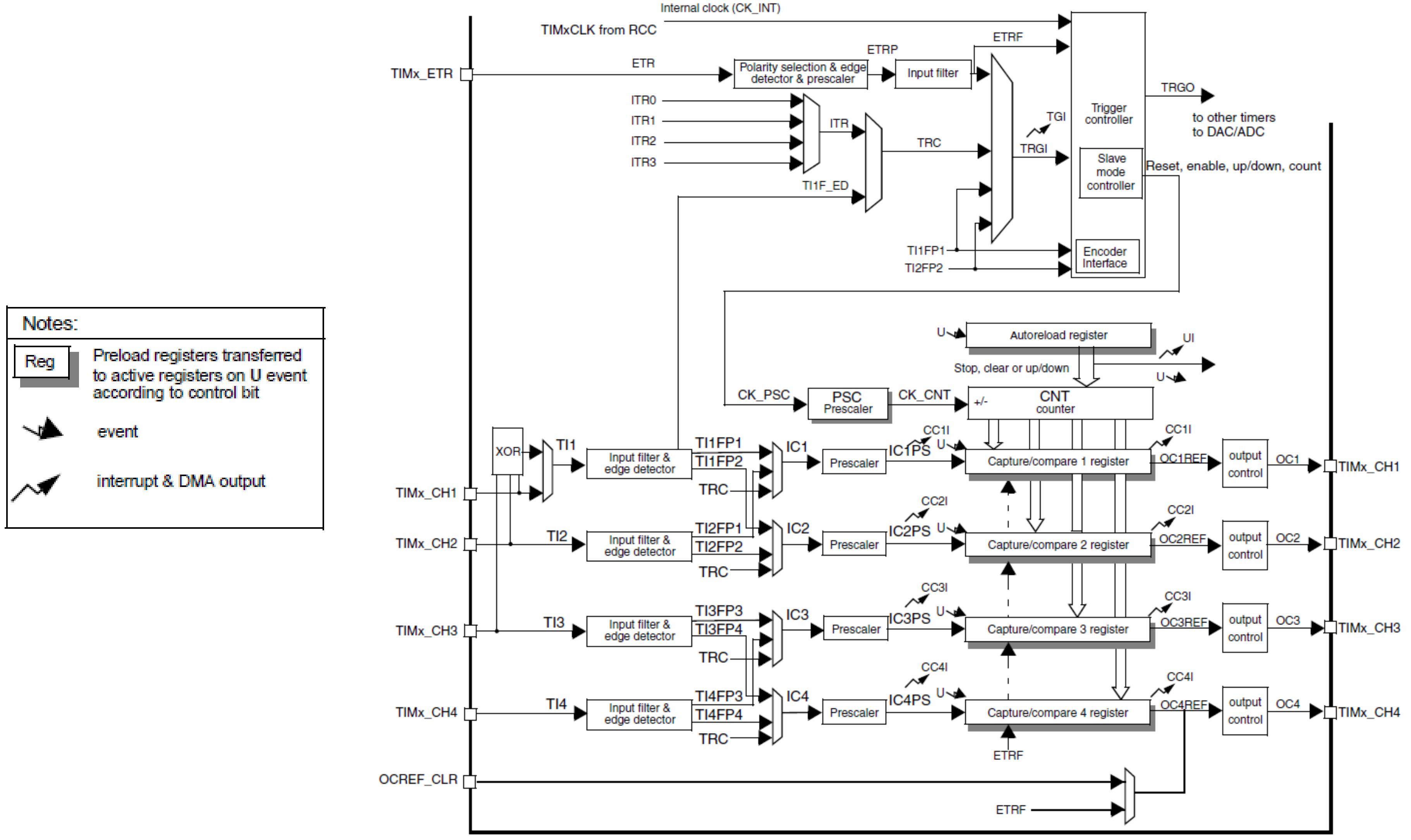
TIM2-TIM5 Introduction

- The general-purpose timers consist of a 16-bit auto-reload counter driven by a programmable prescaler.
  - Measuring the pulse lengths of input signals (input capture)
  - Generating output waveforms (output compare, PWM)
- Pulse lengths and waveform periods can be modulated from a few microseconds to several milliseconds using the timer prescaler and the RCC clock controller prescalers
- General-purpose (TIMx) timers are completely independent, and do not share any resources
  - They can still be synchronized together

TIM2-TIM5 Main Features

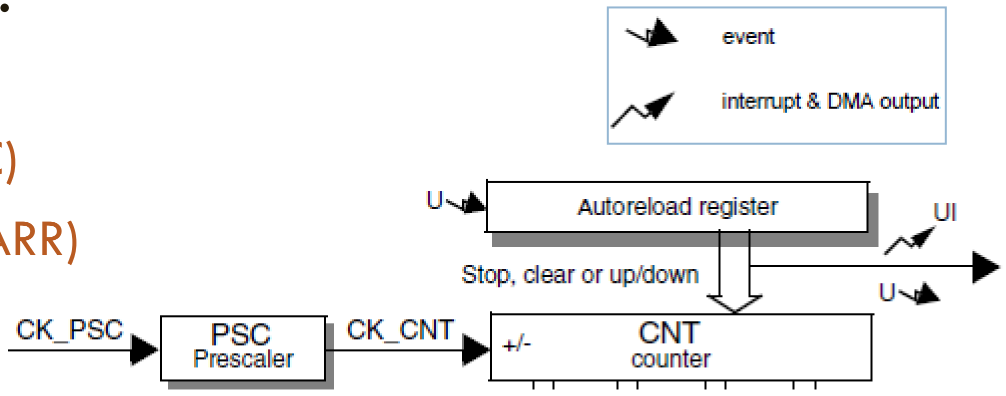
- 16-bit up, down, up/down auto-reload counter
- 16-bit programmable prescaler allowing dividing (also “on the fly”) the counter clock frequency either by any factor between 1 and 65535.
- Up to 4 independent channels for:
  - Input Capture
  - Output Compare
  - PWM generation (Edge and Center-aligned Mode)
  - One-pulse mode output
- Synchronization circuit to control timer with external signals and to interconnect several timers together
- Interrupt/DMA generation based on several events

TIM2-TIM5 Block Diagram



Time-Base Unit

- The main block of the programmable timer is a 16-bit counter with its related auto-reload register
  - The counter can count up, down or both up and down
  - The counter clock can be divided by a prescaler.
- The counter, the auto-reload register and the prescaler register can be written or read by software
  - This is true even when the counter is running
- The time-base unit includes:
  - Counter register (TIMx_CNT)
  - Prescaler register (TIMx_PSC)
  - Auto-reload register (TIMx_ARR)



Time-Base Unit

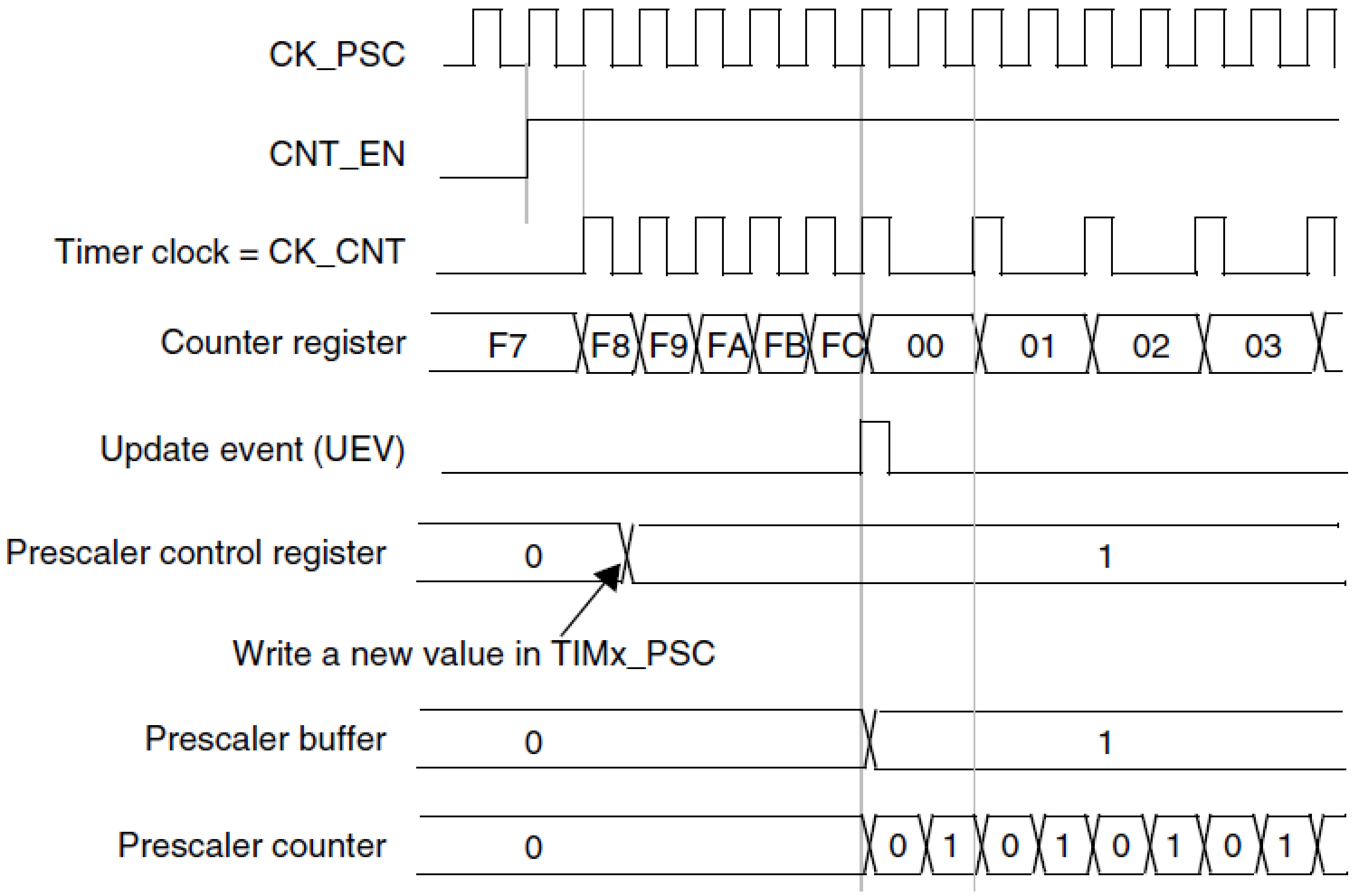
- Auto-reload register is preloaded
- Writing to or reading from the auto-reload register accesses the preload register
- Contents of preload register are transferred into the shadow register permanently or at each update event (UEV), depending on the auto-reload preload enable bit (ARPE) in TIMx_CR1 register
- Update event is sent when counter reaches overflow or underflow and if the UDIS bit equals 0 in TIMx_CR1 register
- Update event can also be generated by software
- Counter is clocked by prescaler output CK_CNT, which is enabled only when counter enable bit (CEN) in TIMx_CR1 register is set
  - actual counter enable signal CNT_EN is set 1 clock cycle after CEN

Prescaler

- Prescaler can divide the counter clock frequency by any factor between 1 and 65536
- Based on a 16-bit counter controlled through a 16-bit register (in the TIMx_PSC register)
- It can be changed on the fly as this control register is buffered
  - New prescaler ratio is taken into account at the next update event

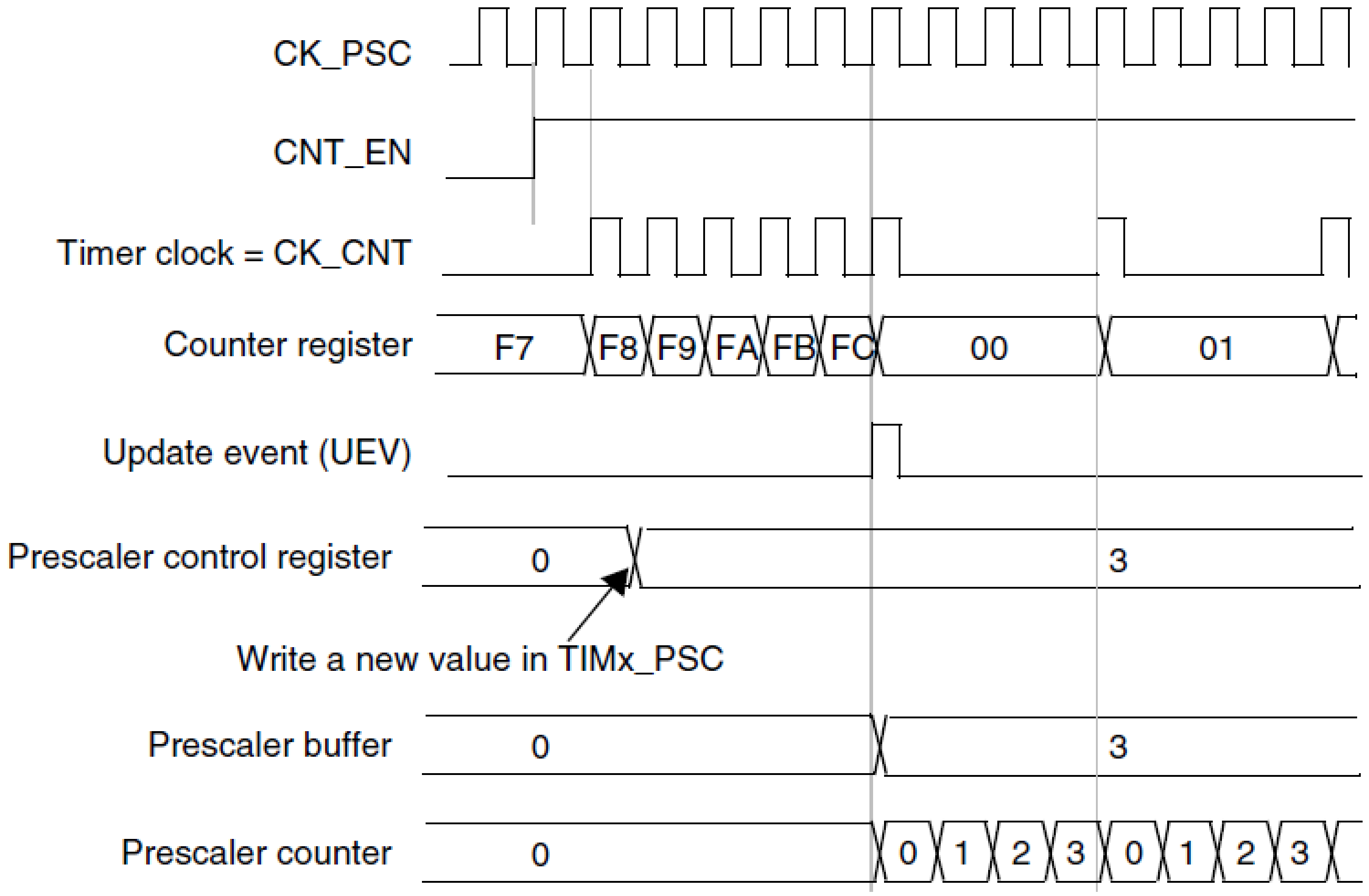
Prescaler

- Counter timing diagram with prescaler division change from 1 to 2



Prescaler

- Counter timing diagram with prescaler division change from 1 to 4

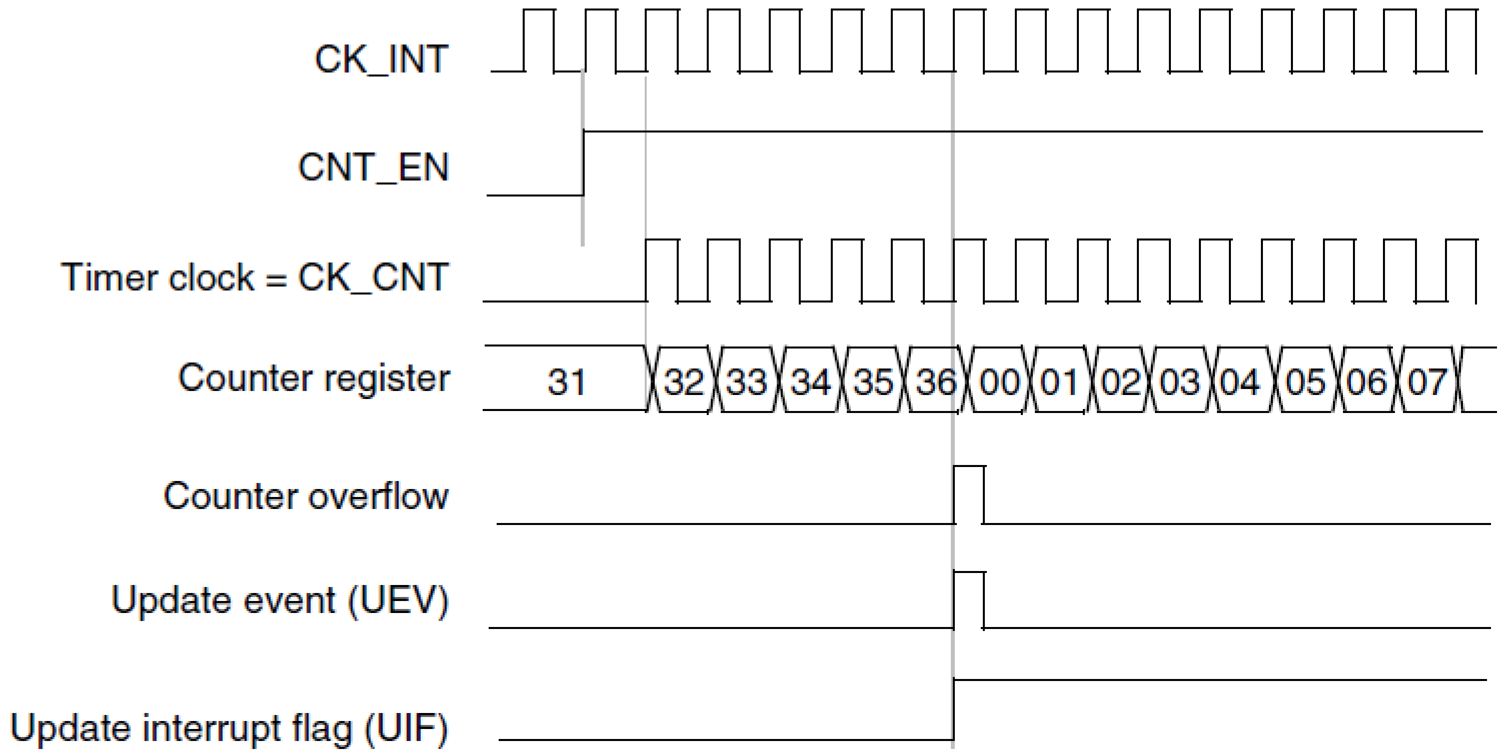


Counter Modes: Upcounting Mode

- Counter counts from 0 to the auto-reload value (content of the TIMx_ARR register), then restarts from 0 and generates a counter overflow event
- An Update event can be generated at each counter overflow or by setting the UG bit in the TIMx_EGR register
- When an update event occurs, all the registers are updated and the update flag (UIF bit in TIMx_SR register) is set (depending on the URS bit):
  - The buffer of the prescaler is reloaded with the preload value (content of the TIMx_PSC register)
  - The auto-reload shadow register is updated with the preload value (TIMx_ARR)

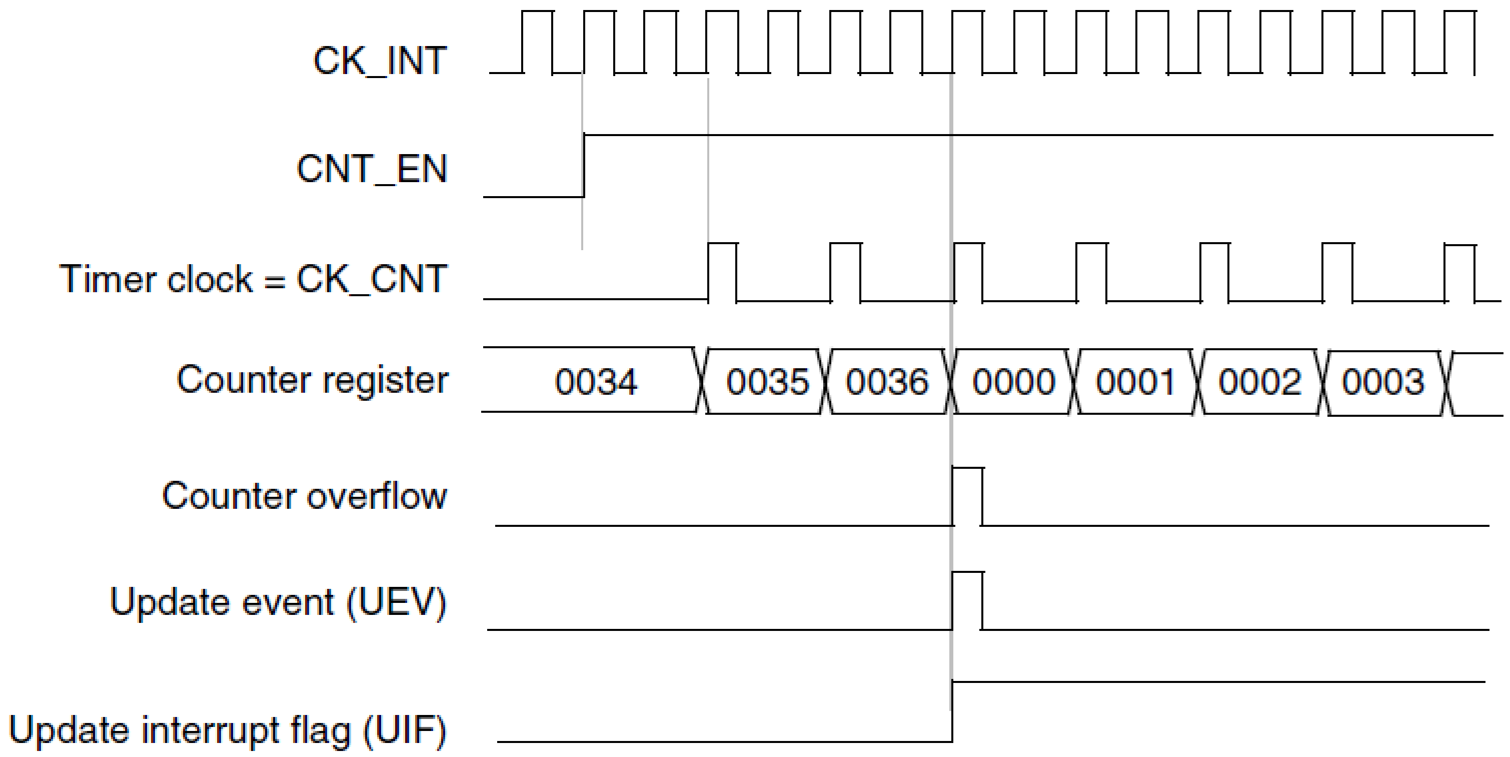
Upcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 1



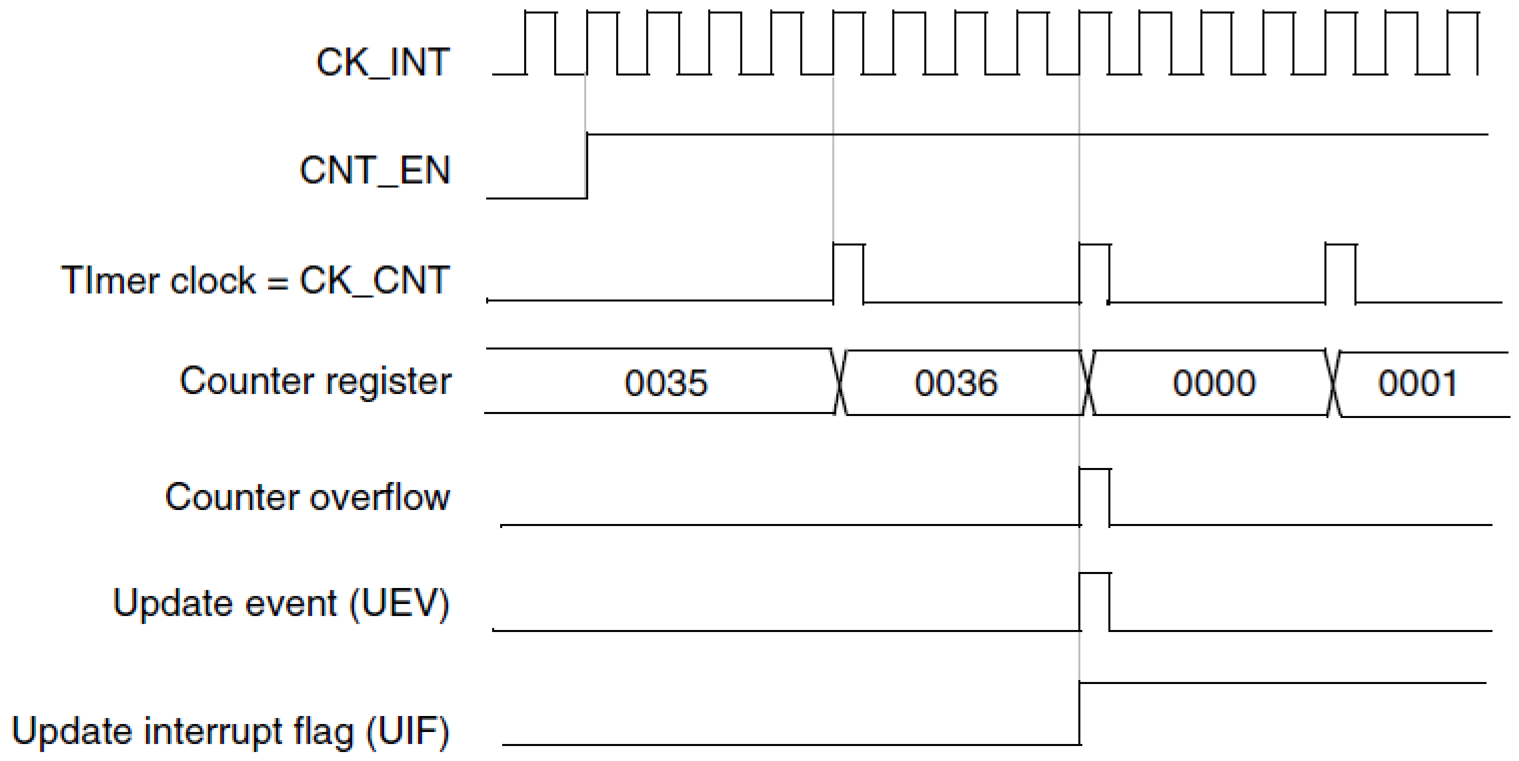
Upcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 2



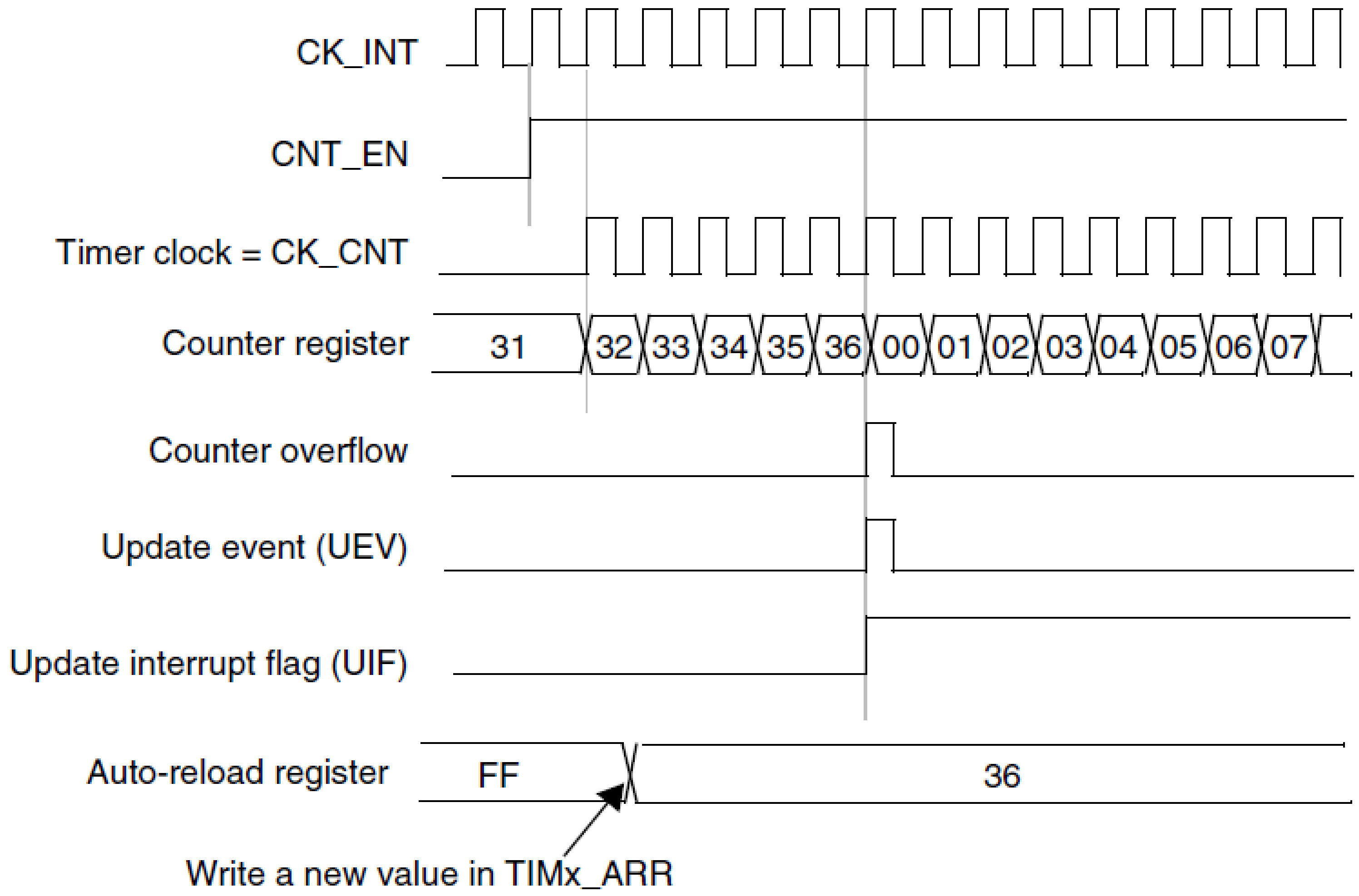
Upcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 4



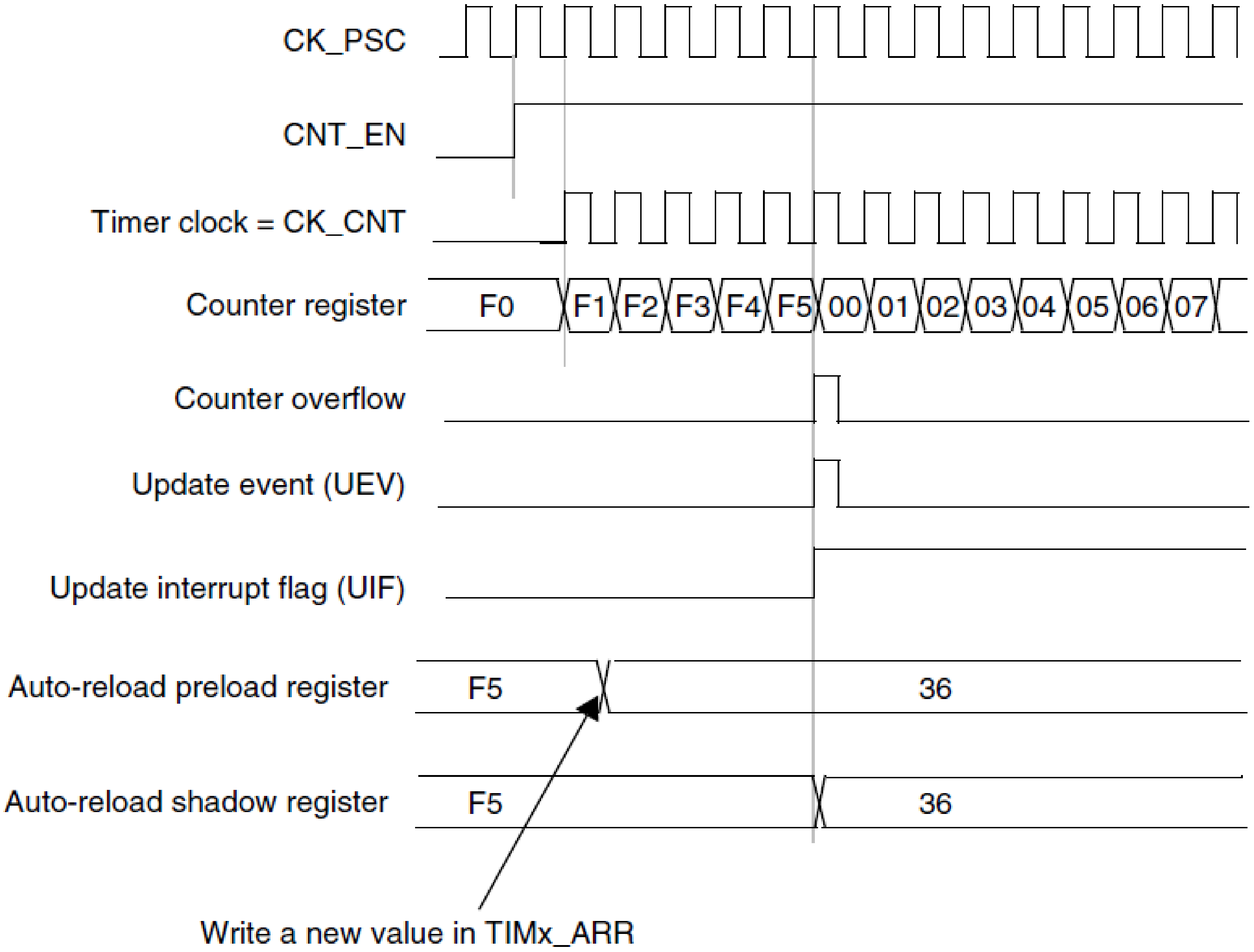
Upcounting Mode Example

- TIMx_ARR=0x36 , Update event when ARPE=0 (TIMx_ARR not preloaded)



Upcounting Mode Example

- TIMx_ARR=0x36 , Update event when ARPE=1 (TIMx_ARR preloaded)

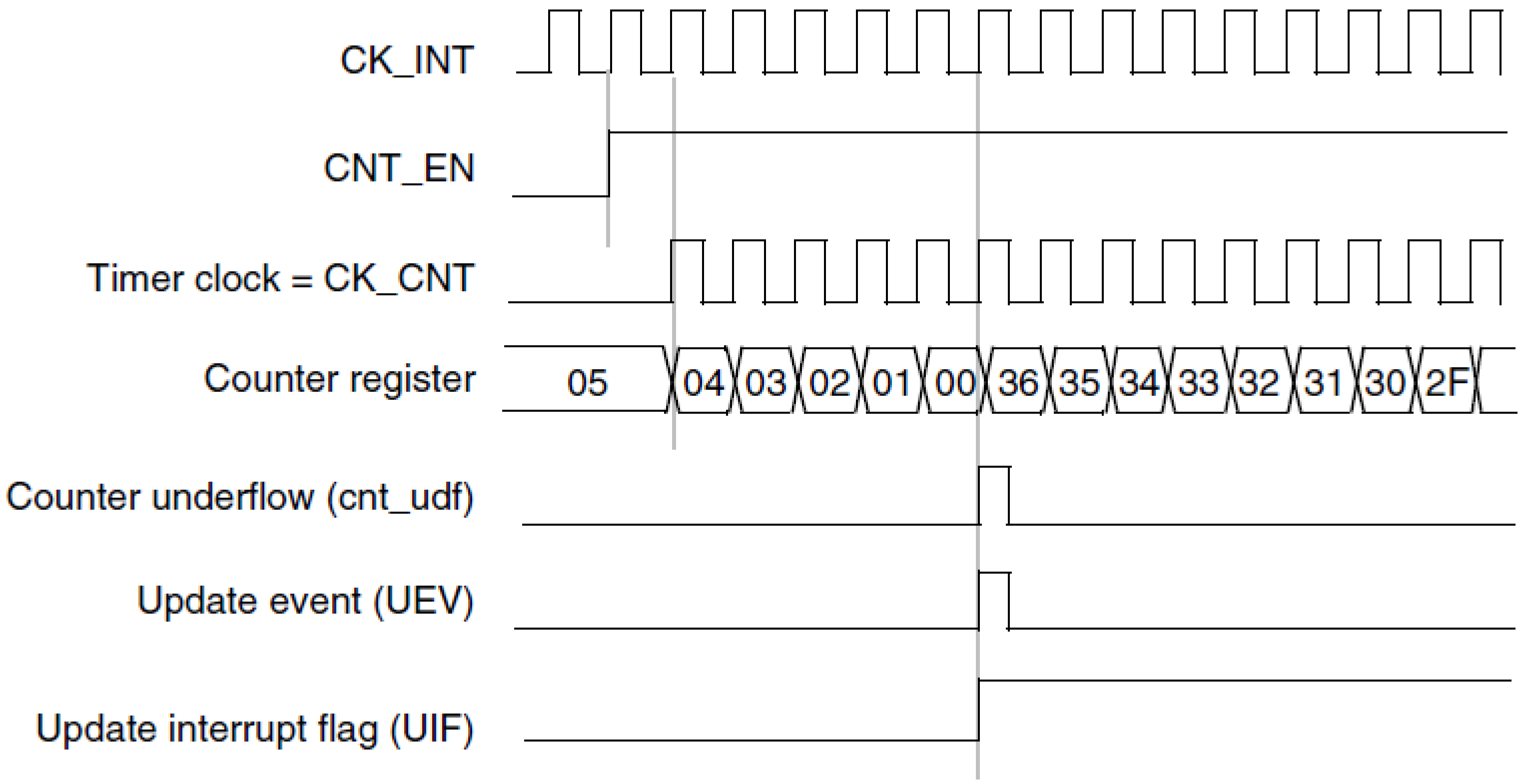


Counter Modes: Downcounting Mode

- Counter counts from the auto-reload value (content of the
- TIMx_ARR register) down to 0, then restarts from the auto-reload value and generates a counter underflow event
- An Update event can be generate at each counter underflow or by setting the UG bit in the TIMx_EGR register
- When an update event occurs, all the registers are updated and the update flag (UIF bit in TIMx_SR register) is set (depending on the URS bit):
  - The buffer of the prescaler is reloaded with the preload value (content of the TIMx_PSC register)
  - The auto-reload shadow register is updated with the preload value (TIMx_ARR)

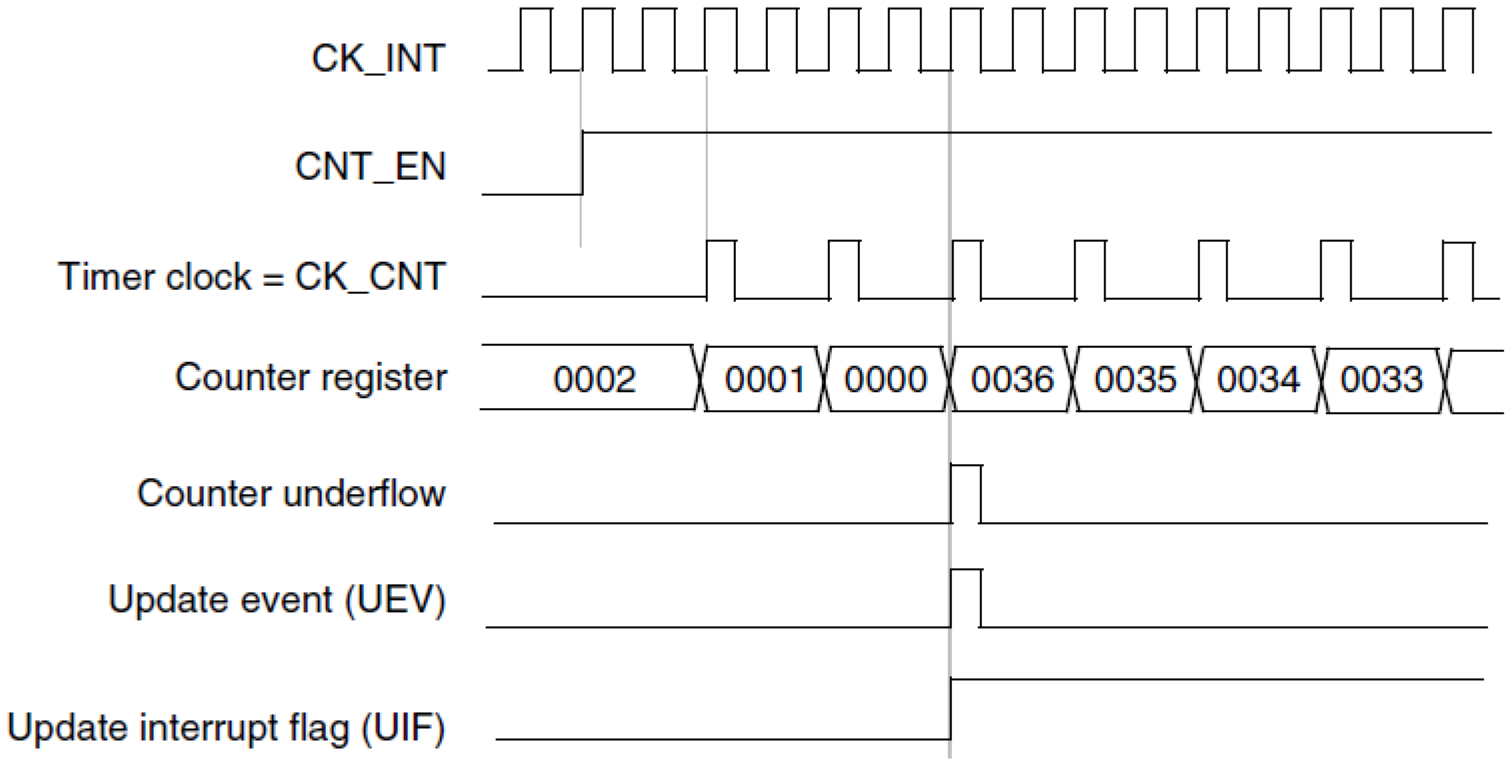
Downcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 1



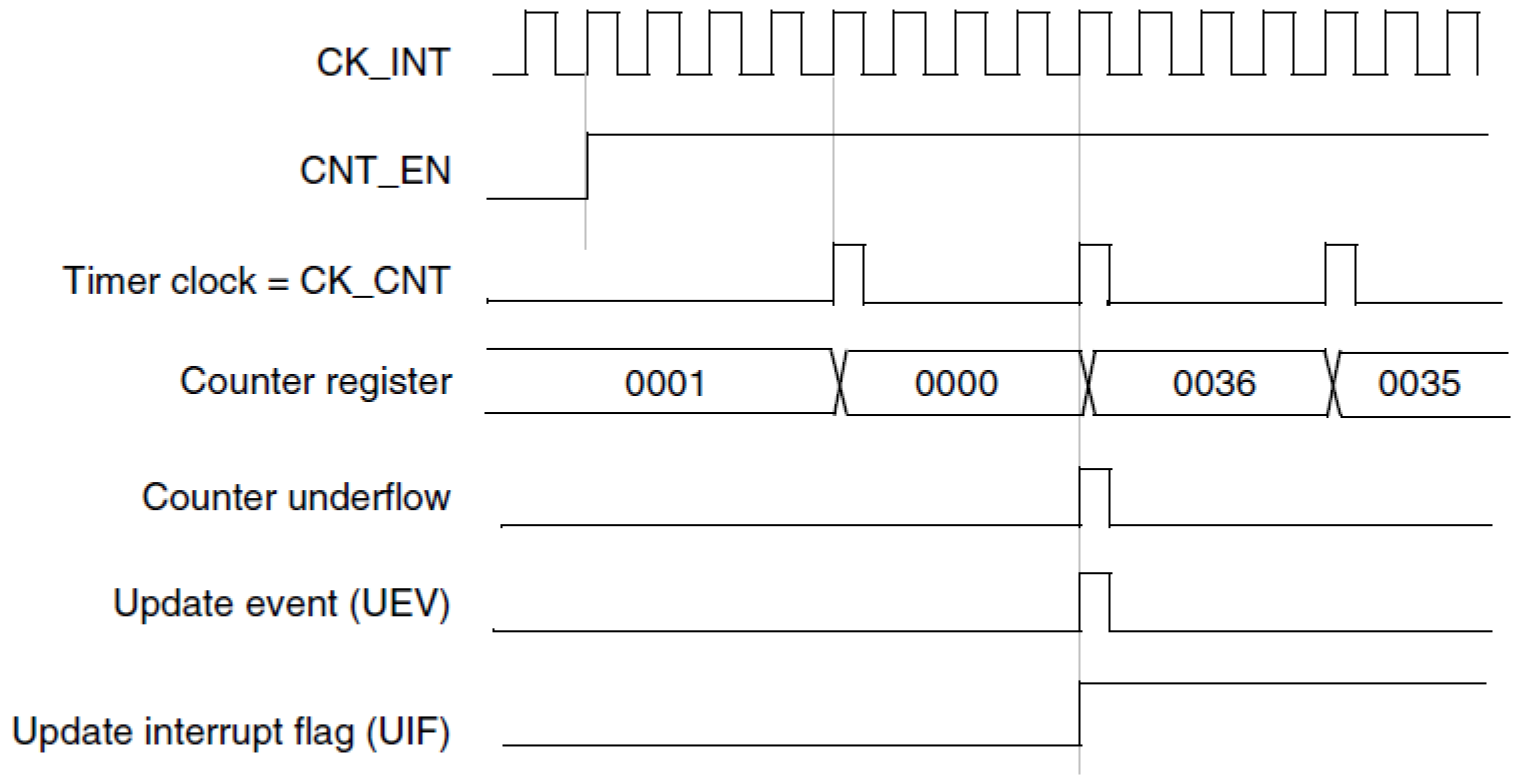
Downcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 2



Downcounting Mode Example

- TIMx_ARR=0x36 , internal clock divided by 4

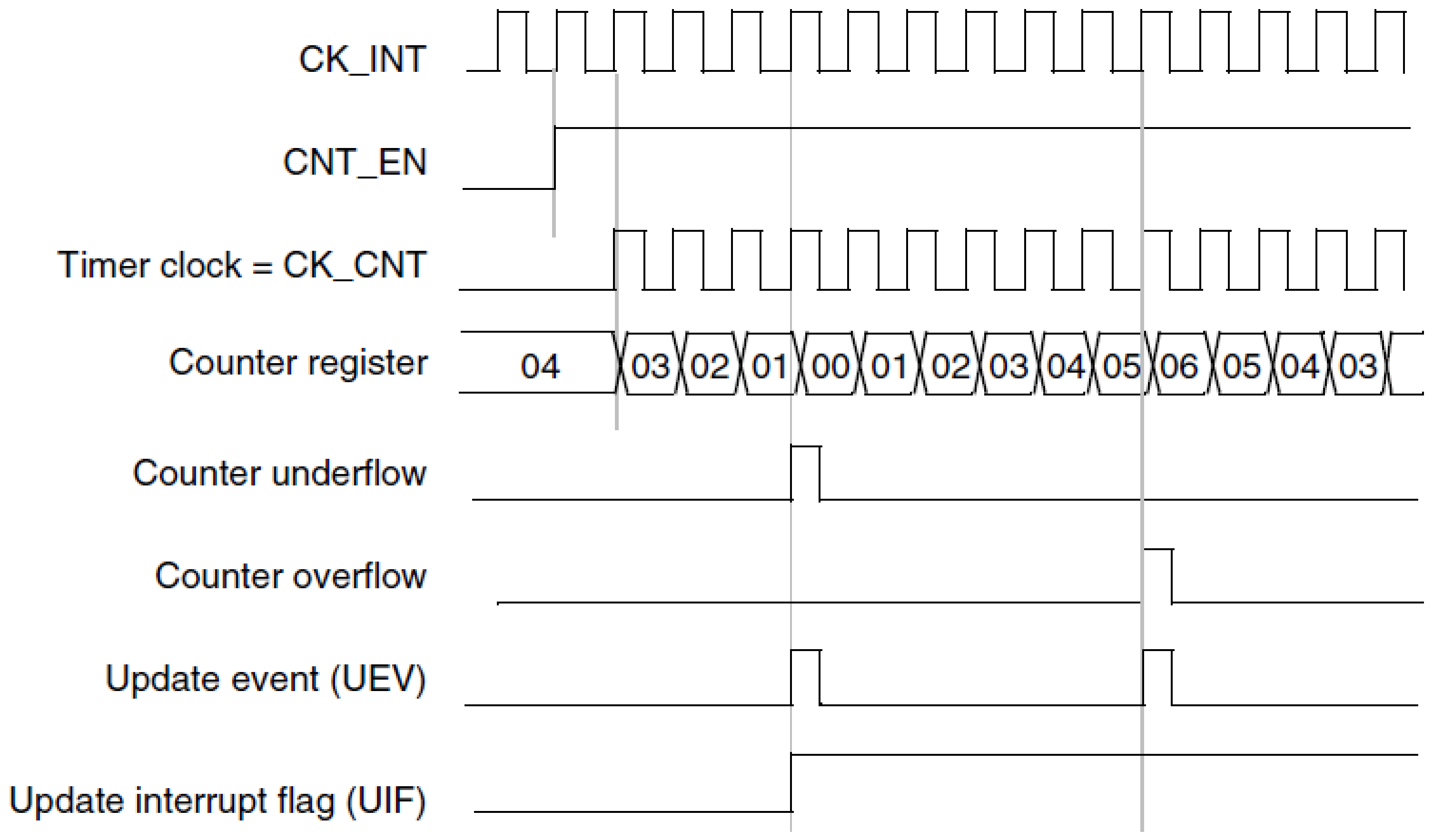


Counter Modes: Center-Aligned Mode (Up/Down Counting)

- Counter counts from 0 to the auto-reload value (content of the TIMx_ARR register) – 1, generates a counter overflow event, then counts from the autoreload value down to 1 and generates a counter underflow event. Then it restarts counting from 0
- Center-aligned mode is active when CMS bits in TIMx_CR1 register are not equal to '00'. The Output compare interrupt flag of channels configured in output is set when: the counter counts down (Center aligned mode 1, CMS = "01"), the counter counts up (Center aligned mode 2, CMS = "10") the counter counts up and down (Center aligned mode 3, CMS = "11")
- In this mode, the direction bit (DIR from TIMx_CR1 register) cannot be written
  - Updated by hardware and gives the current direction of the counter

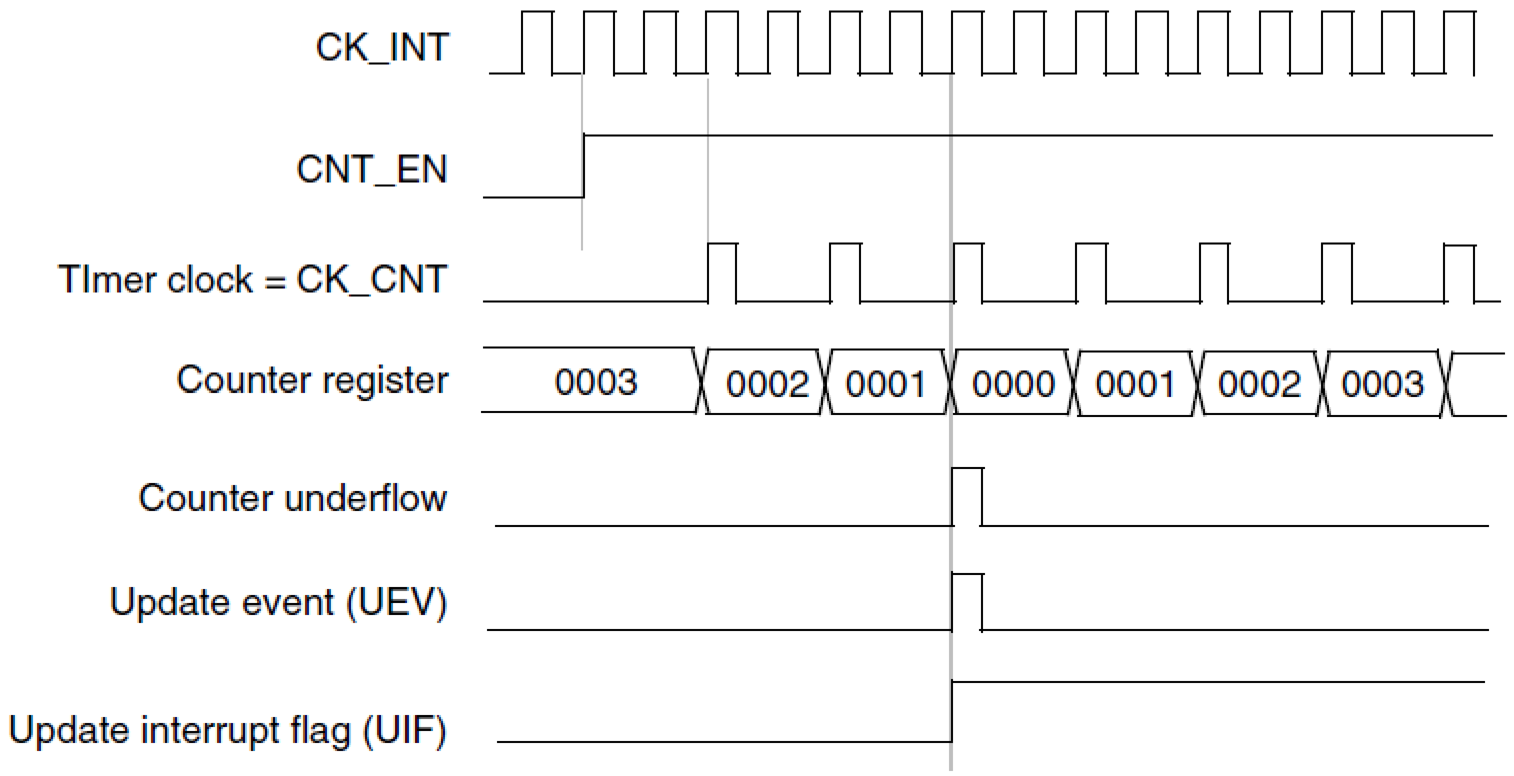
Up/Down Counting Example

- Internal clock divided by 1, TIMx_ARR=0x6



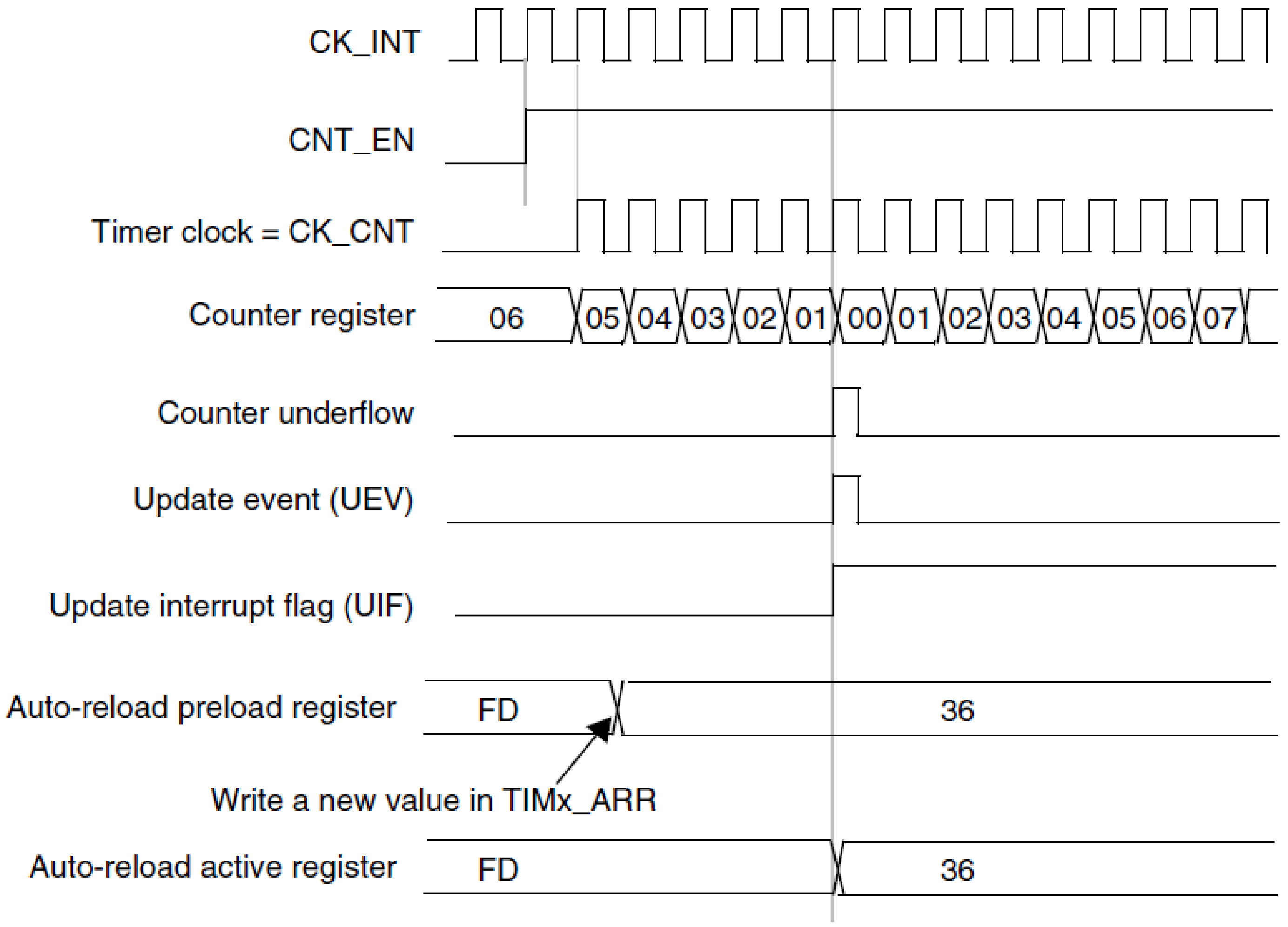
Up/Down Counting Example

- Internal clock divided by 2, TIMx_ARR=0x6



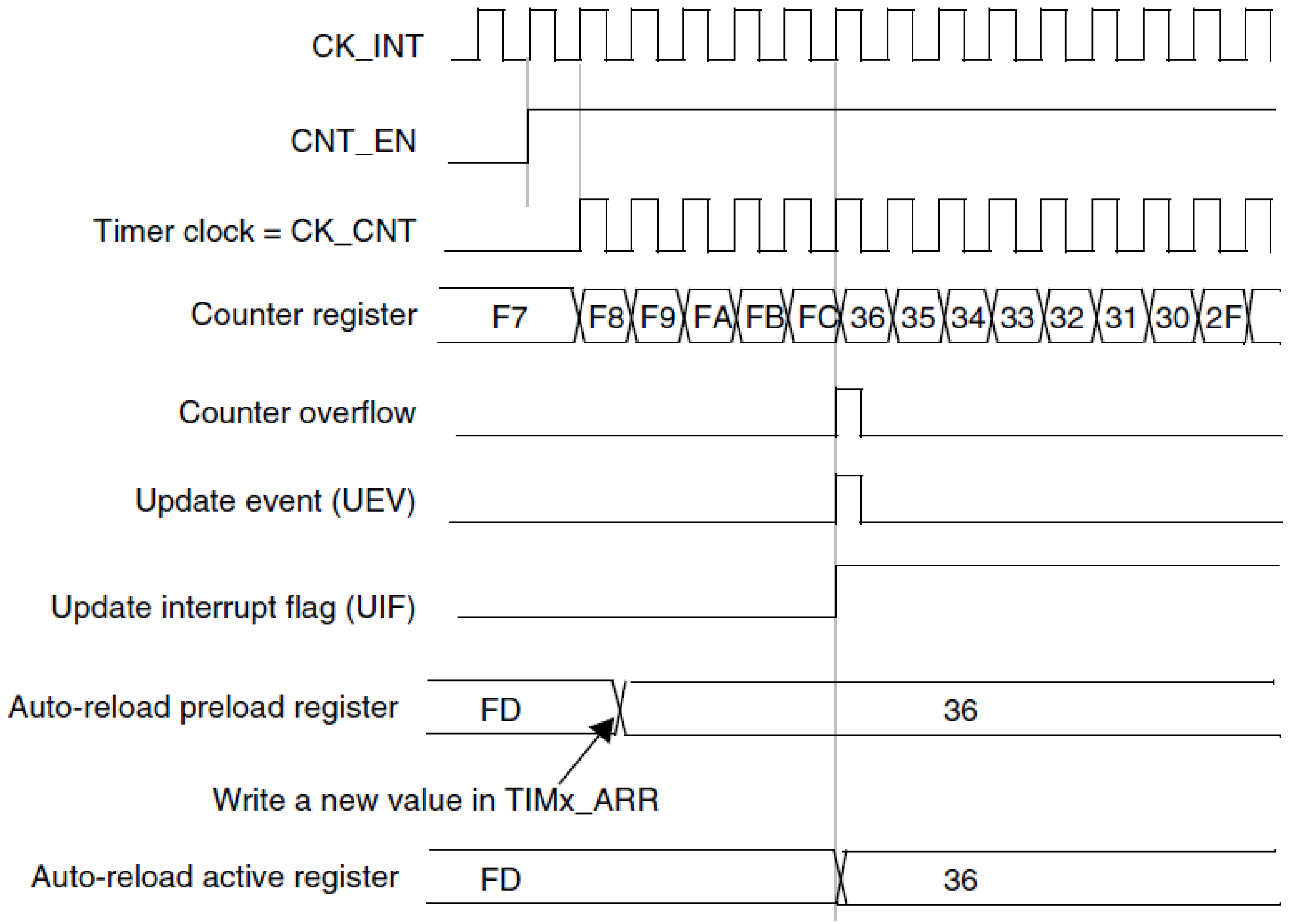
Up/Down Counting Example

- Counter timing diagram, Update event with ARPE=1 (counter underflow)



Up/Down Counting Example

- Counter timing diagram, Update event with ARPE=1 (counter overflow)

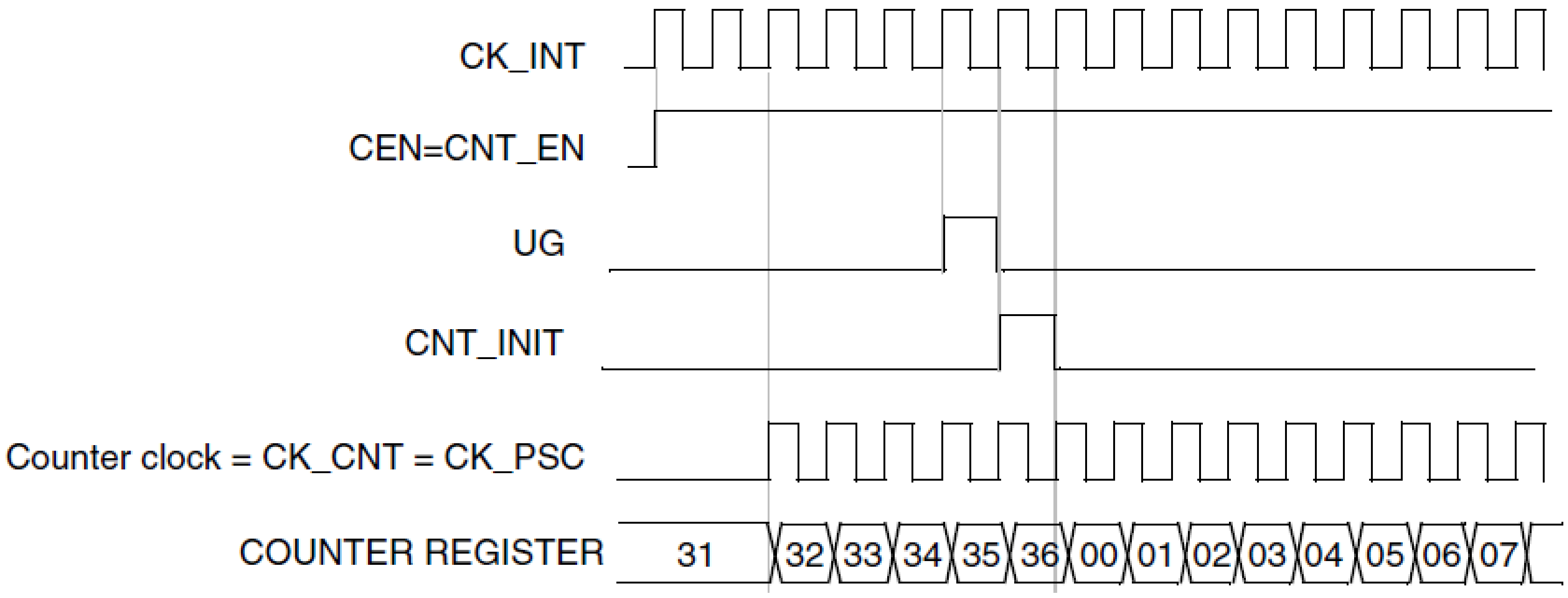


Clock Selection

- The counter clock can be provided by the following clock sources:
  - **Internal clock (CK_INT)** **(Our focus in this part)**
  - External clock mode1: external input pin (TIx)
  - External clock mode2: external trigger input (ETR)
  - Internal trigger inputs (ITRx): using one timer as prescaler for another timer, for example, you can configure Timer 1 to act as a prescaler for Timer 2

Internal Clock Source (CK_INT)

- If the slave mode controller is disabled (SMS=000 in the TIMx_SMCR register), then the CEN, DIR (in the TIMx_CR1 register) and UG bits (in the TIMx_EGR register) are actual control bits and can be changed only by software (except UG which remains cleared automatically)
- As soon as the CEN bit is written to 1, the prescaler is clocked by the internal clock CK_INT.



TIMx Control Register 1 (TIMx_CR1)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						CKD[1:0]		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Reset value: 0x0000

Bits 9:8 **CKD**: Clock division

This bit-field indicates the division ratio between the timer clock (CK_INT) frequency and sampling clock used by the digital filters (ETR, TIx),

00: $t_{DTS} = t_{CK\_INT}$
01: $t_{DTS} = 2 \times t_{CK\_INT}$
10: $t_{DTS} = 4 \times t_{CK\_INT}$
11: Reserved

Bit 7 **ARPE**: Auto-reload preload enable

0: TIMx_ARR register is not buffered
1: TIMx_ARR register is buffered

Bits 6:5 **CMS**: Center-aligned mode selection

00: Edge-aligned mode. The counter counts up or down depending on the direction bit (DIR).
01: Center-aligned mode 1. The counter counts up and down alternatively. Output compare interrupt flags of channels configured in output (CCxS=00 in TIMx_CCMRx register) are set only when the counter is counting down.
10: Center-aligned mode 2. The counter counts up and down alternatively. Output compare interrupt flags of channels configured in output (CCxS=00 in TIMx_CCMRx register) are set only when the counter is counting up.
11: Center-aligned mode 3. The counter counts up and down alternatively. Output compare interrupt flags of channels configured in output (CCxS=00 in TIMx_CCMRx register) are set both when the counter is counting up or down.

Note: It is not allowed to switch from edge-aligned mode to center-aligned mode as long as the counter is enabled (CEN=1)

Bit 4 **DIR**: Direction

0: Counter used as upcounter
1: Counter used as downcounter

Note: This bit is read only when the timer is configured in Center-aligned mode or Encoder mode.

Bit 3 **OPM**: One-pulse mode

0: Counter is not stopped at update event
1: Counter stops counting at the next update event (clearing the bit CEN)

Bit 2 **URS**: Update request source

This bit is set and cleared by software to select the UEV event sources.

0: Any of the following events generate an update interrupt or DMA request if enabled. These events can be:

- Counter overflow/underflow
- Setting the UG bit
- Update generation through the slave mode controller

1: Only counter overflow/underflow generates an update interrupt or DMA request if enabled.

Bit 1 **UDIS**: Update disable

This bit is set and cleared by software to enable/disable UEV event generation.

0: UEV enabled. The Update (UEV) event is generated by one of the following events:

- Counter overflow/underflow
- Setting the UG bit
- Update generation through the slave mode controller

Buffered registers are then loaded with their preload values.

1: UEV disabled. The Update event is not generated, shadow registers keep their value (ARR, PSC, CCRx). However the counter and the prescaler are reinitialized if the UG bit is set or if a hardware reset is received from the slave mode controller.

Bit 0 **CEN**: Counter enable

0: Counter disabled
1: Counter enabled

Note: External clock, gated mode and encoder mode can work only if the CEN bit has been previously set by software. However trigger mode can set the CEN bit automatically by hardware.

CEN is cleared automatically in one-pulse mode, when an update event occurs.

TIMx Slave Mode Control Register (TIMx_SMCR)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Reset value: 0x0000

Bits 2:0 **SMS:** Slave mode selection

When external signals are selected the active edge of the trigger signal (TRGI) is linked to the polarity selected on the external input (see Input Control register and Control Register description.

000: Slave mode disabled - if CEN = ‘1 then the prescaler is clocked directly by the internal clock.

TIMx Event Generation Register (TIMx_EGR)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

Reset value: 0x0000

Bit 6 **TG**: Trigger generation

This bit is set by software in order to generate an event, it is automatically cleared by hardware.
0: No action
1: The TIF flag is set in TIMx_SR register. Related interrupt or DMA transfer can occur if enabled.

Bit 0 **UG**: Update generation

This bit can be set by software, it is automatically cleared by hardware.
0: No action
1: Re-initialize the counter and generates an update of the registers. Note that the prescaler counter is cleared too (anyway the prescaler ratio is not affected). The counter is cleared if the center-aligned mode is selected or if DIR=0 (upcounting), else it takes the auto-reload value (TIMx_ARR) if DIR=1 (downcounting).

TIMx Counter (TIMx_CNT)

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

Reset value: 0x0000

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

TIMx Prescaler (TIMx_PSC)

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

Reset value: 0x0000

Bits 15:0 **PSC[15:0]**: Prescaler value

The counter clock frequency CK_CNT is equal to f_{CK_PSC} / (PSC[15:0] + 1).

PSC contains the value to be loaded in the active prescaler register at each update event.

TIMx Auto-Reload Register (TIMx_ARR)

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

Reset value: 0x0000

Bits 15:0 **ARR[15:0]**: Low Auto-reload value
ARR is the value to be loaded in the actual auto-reload register.

The counter is blocked while the auto-reload value is null.

TIMx DMA/Interrupt Enable Register (TIMx_DIER)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

Reset value: 0x0000

Bit 0 **UIE**: Update interrupt enable

0: Update interrupt disabled
1: Update interrupt enabled

TIMx Status Register (TIMx_SR)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			CC4OF	CC3OF	CC2OF	CC1OF	Reserved		TIF	Res	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

Reset value: 0x0000

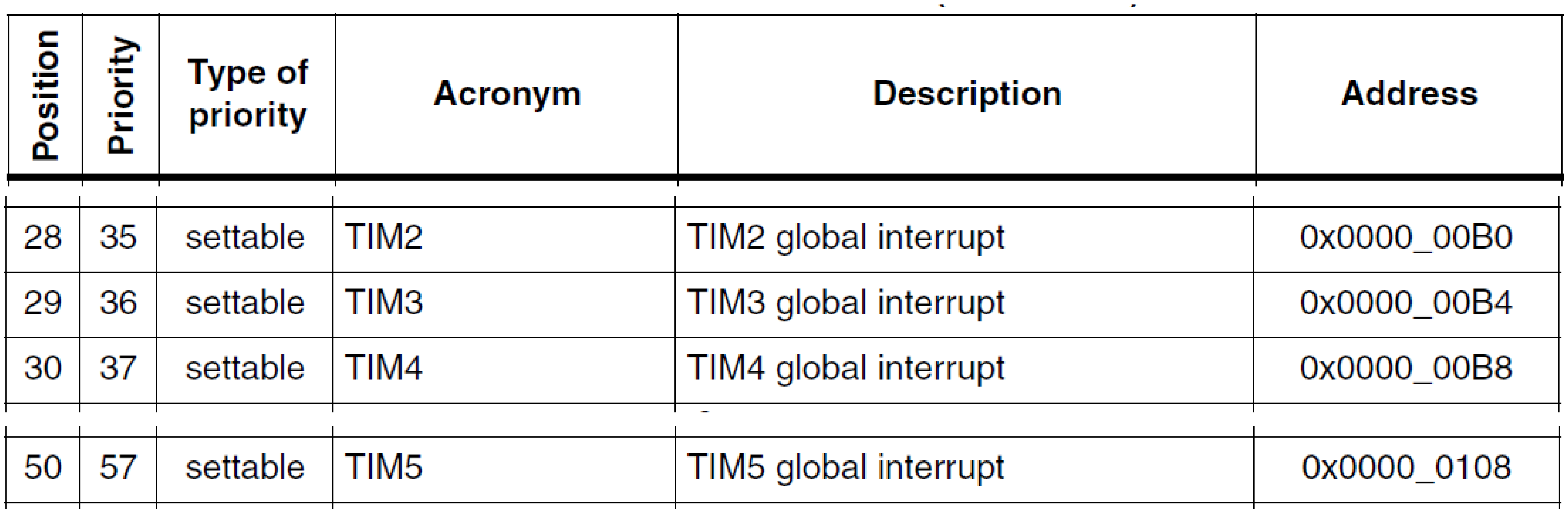
Bit 0 **UIF**: Update interrupt flag

This bit is set by hardware on an update event. It is cleared by software.

0: No update occurred

1: Update interrupt pending. This bit is set by hardware when the registers are updated:

Vector Table for STM32F100xx Devices



Position	Priority	Type of priority	Acronym	Description	Address
28	35	settable	TIM2	TIM2 global interrupt	0x0000_00B0
29	36	settable	TIM3	TIM3 global interrupt	0x0000_00B4
30	37	settable	TIM4	TIM4 global interrupt	0x0000_00B8
50	57	settable	TIM5	TIM5 global interrupt	0x0000_0108

Timer Standard Driver

- Standard interface to all STM32 timers

- `TIM_CounterModeConfig`
- `TIM_SetCounter`
- `TIM_SetAutoreload`
- `TIM_PrescalerConfig`
- `TIM_ITConfig`
- `TIM_Cmd`
- `TIM_ClearITPendingBit`

Note: *Modify only files in the "User" group of your project and never change the standard peripheral drivers since they will affect other programs not just the one you are working on at the time. Modification of such files will result in deducting points off your project grade.*

Assignments

- ARM Project #4