Module 9

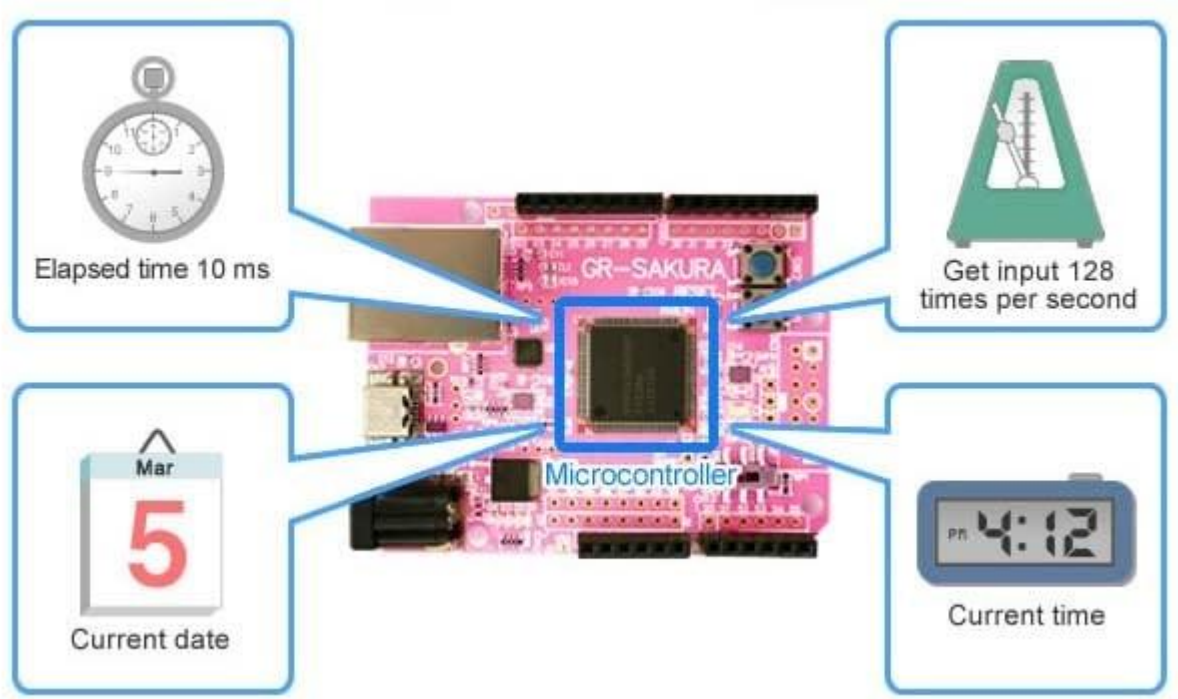
Timers

22nd January 2025

General Timers

Timers

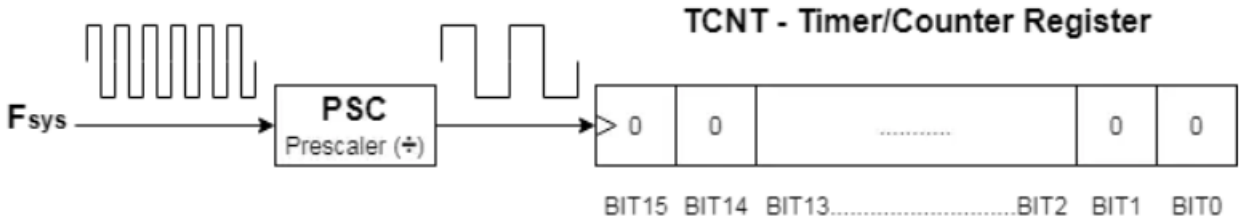
Timers are essential peripheral devices used for generating precise time intervals, generating PWM signals, measuring time durations, and triggering periodic events in embedded systems.



Timers in STM32

Two Major Timers:

1. Advanced-Control Timer (TIM1)
2. General Purpose Timers (TIM2 - TIM 11)



Other Timer Related Elements:

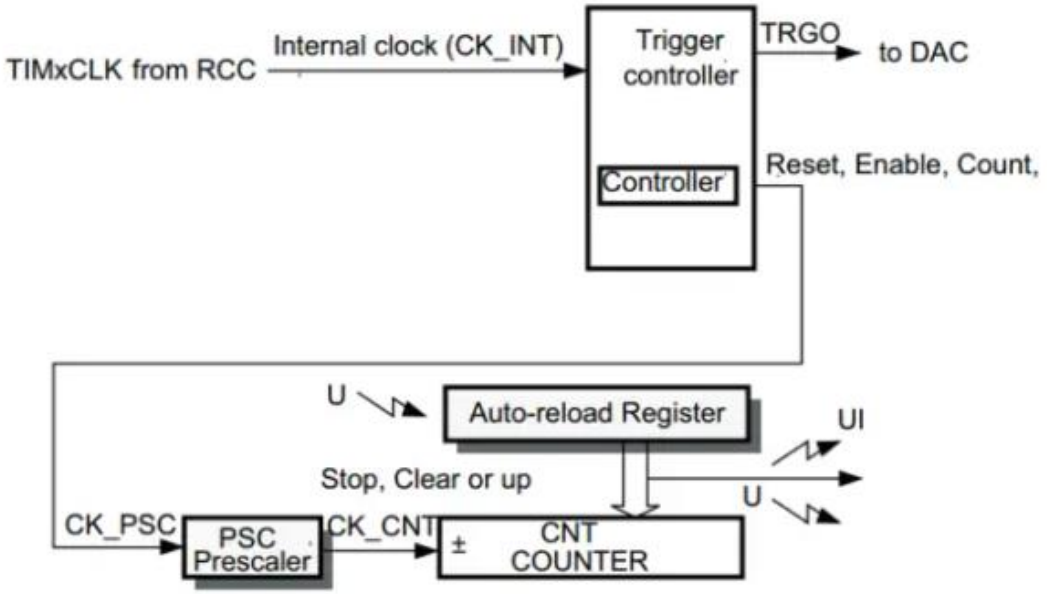
1. Independent Watchdog (IWDG)
2. Window Watchdog (WWDG)
3. Real-Time Clock (RTC)

Timers

Thynk Loop

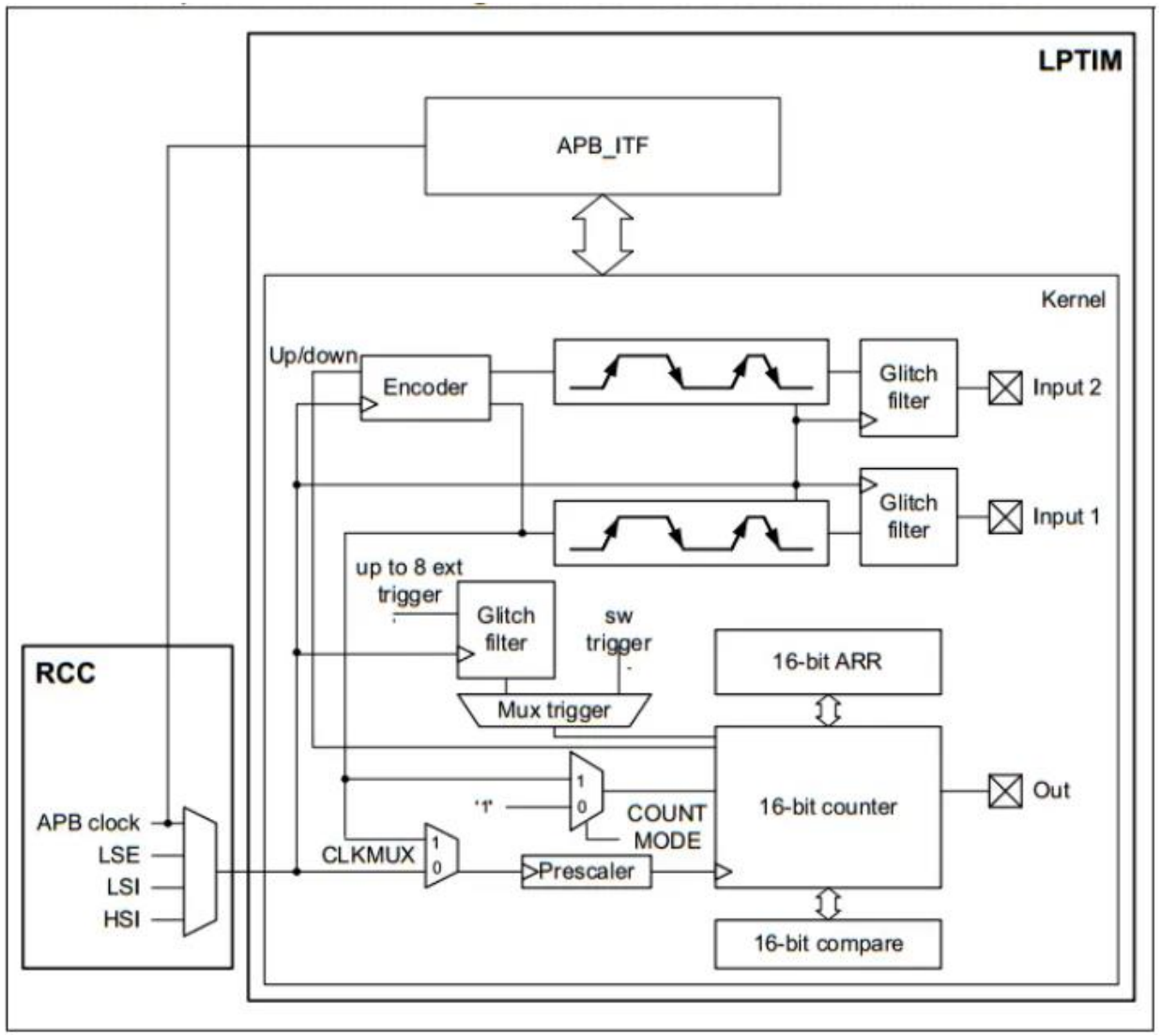
Aspect	Basic Timers	General Purpose	Advanced Timers
Resolution	16-bit	16/32-bit	16-bit
Max Frequency	APB clock	APB clock	APB clock
DMA Support	Limited	Full	Enhanced
Complementary Outputs	No	No	Yes
Dead-time Generation	No	No	Yes
Break Function	No	No	Yes

Basic Timers



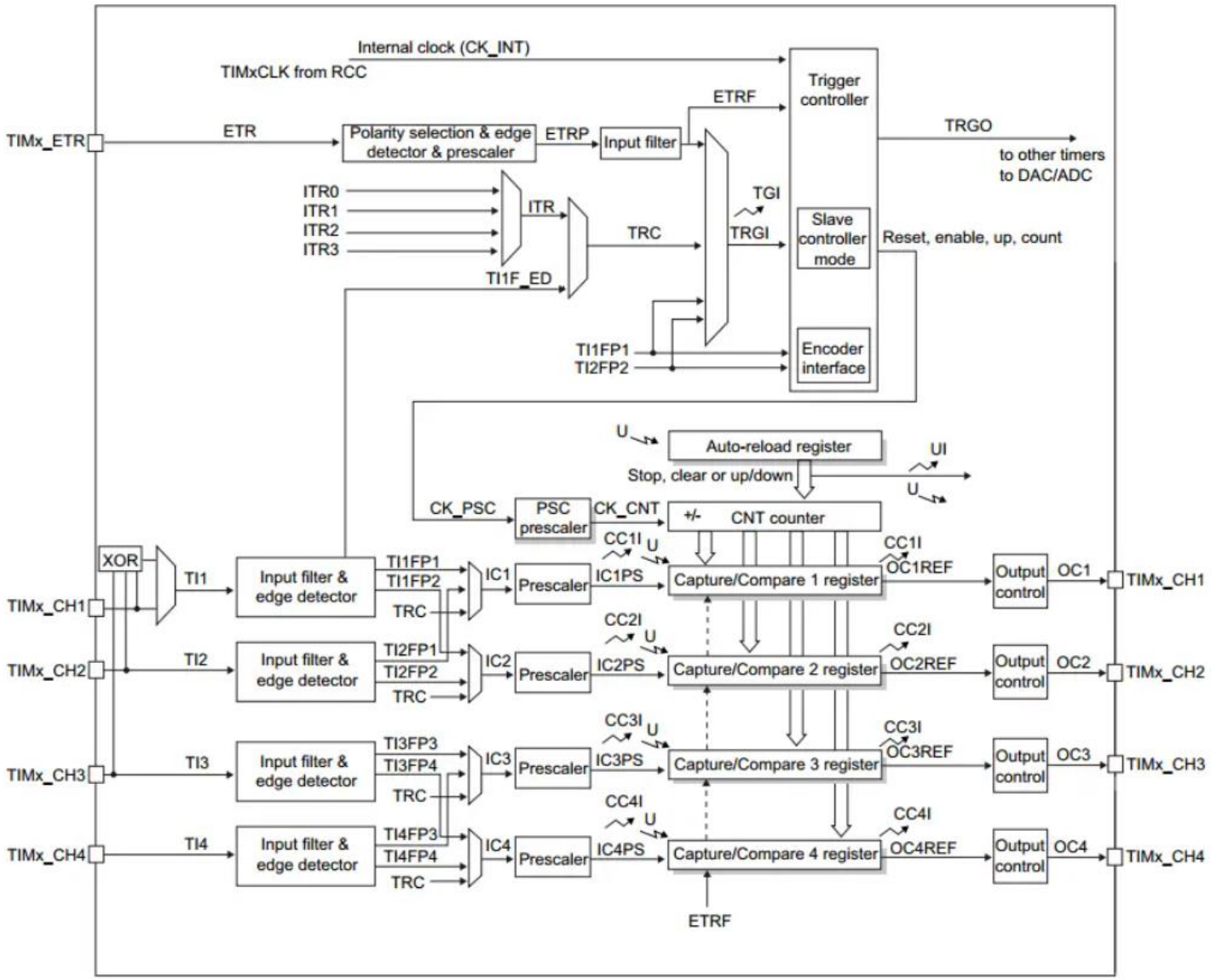
- 16-bit auto-reload up-counter
- 16-bit programmable Prescaler used to divide (also “on the fly”) the counter clock frequency by any factor between 1 and 65536
- Synchronization circuit to trigger the DAC
- Interrupt/DMA generation on the update event: counter overflow

Low Power Timers



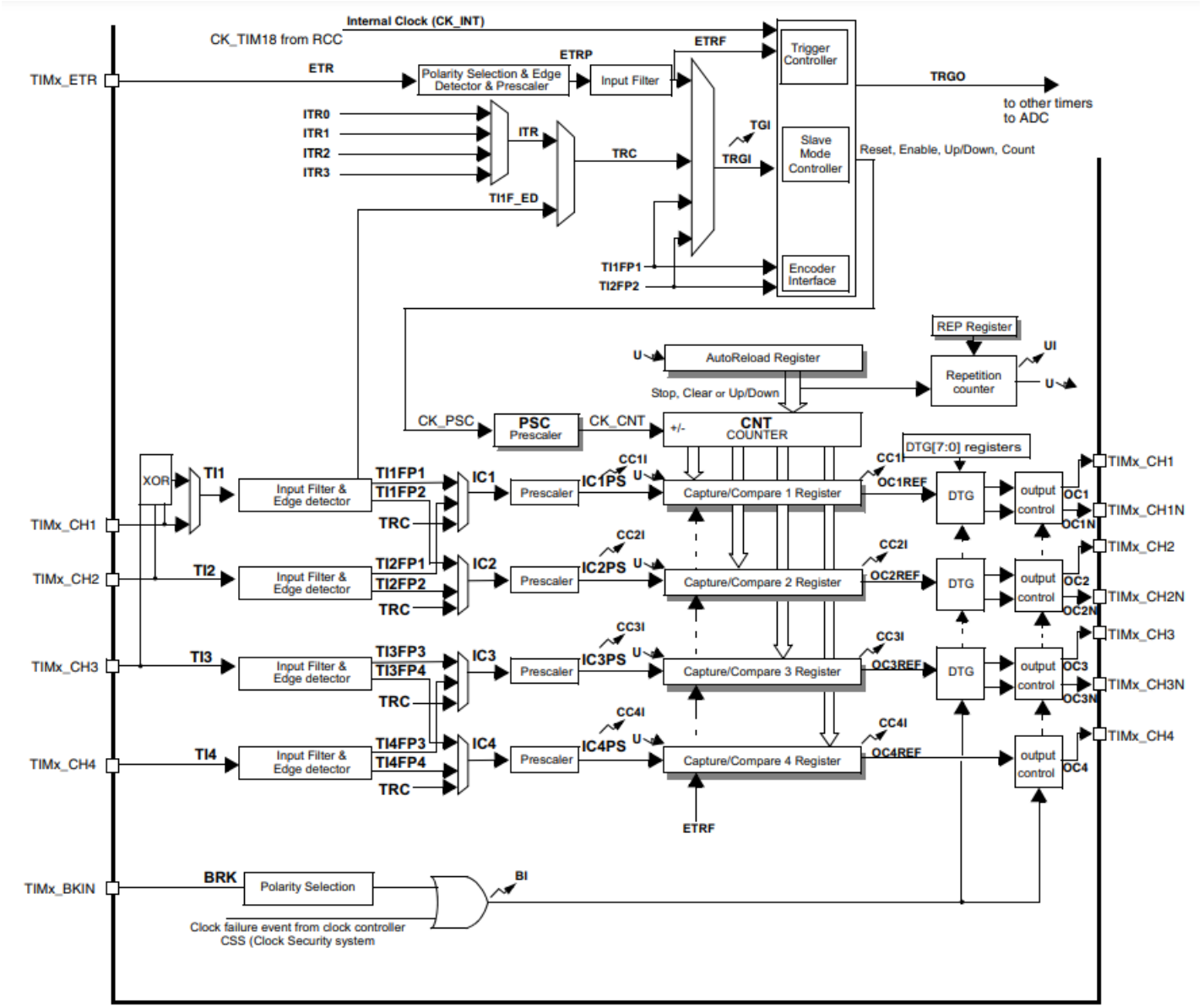
- 16-bit up-counter
- 3-bit Prescaler with 8 possible dividing factors (1,2,4,8,16,32,64,128)
- 16 bit ARR auto-reload register
- 16 bit compare register
- Continuous/One-shot mode
- Selectable software/hardware input trigger
- Programmable Digital Glitch filter
- Configurable output: Pulse, PWM
- Configurable I/O polarity
- Encoder mode
- Repetition counter

General Purpose Timers



- Up to 4 independent channels for:
  - Input capture
  - Output compare
  - PWM generation (Edge- and Center-aligned modes)
  - One-pulse mode output

Advanced Timers

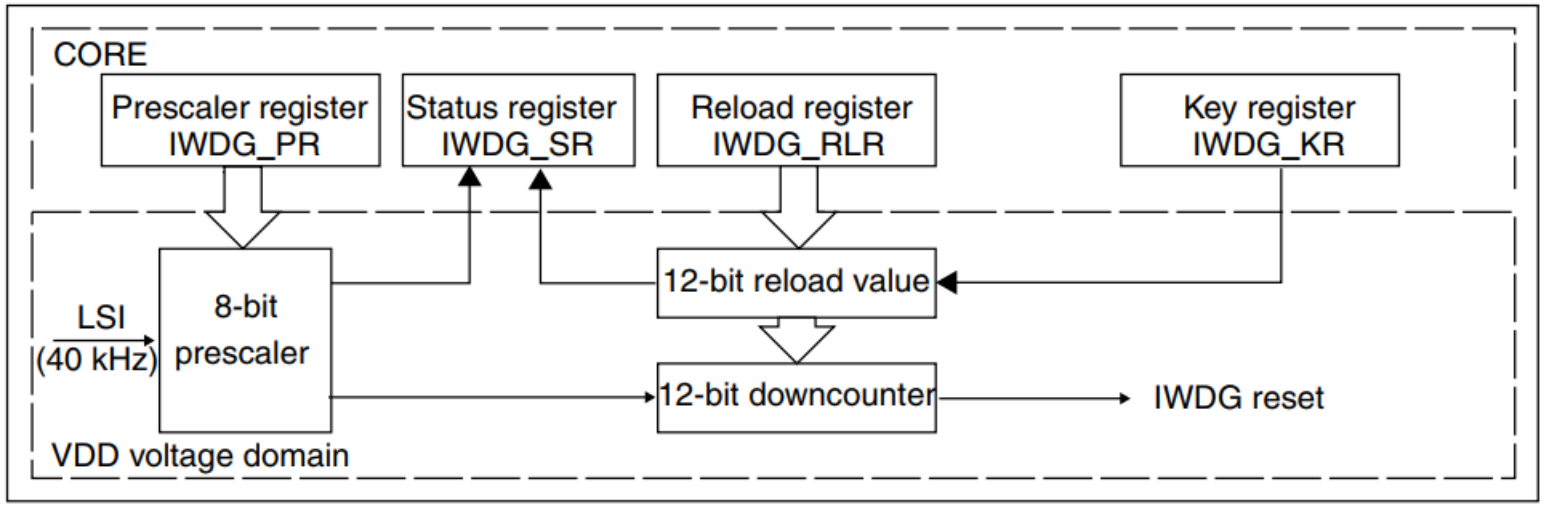


General vs Advanced Timers

Thynk Loop

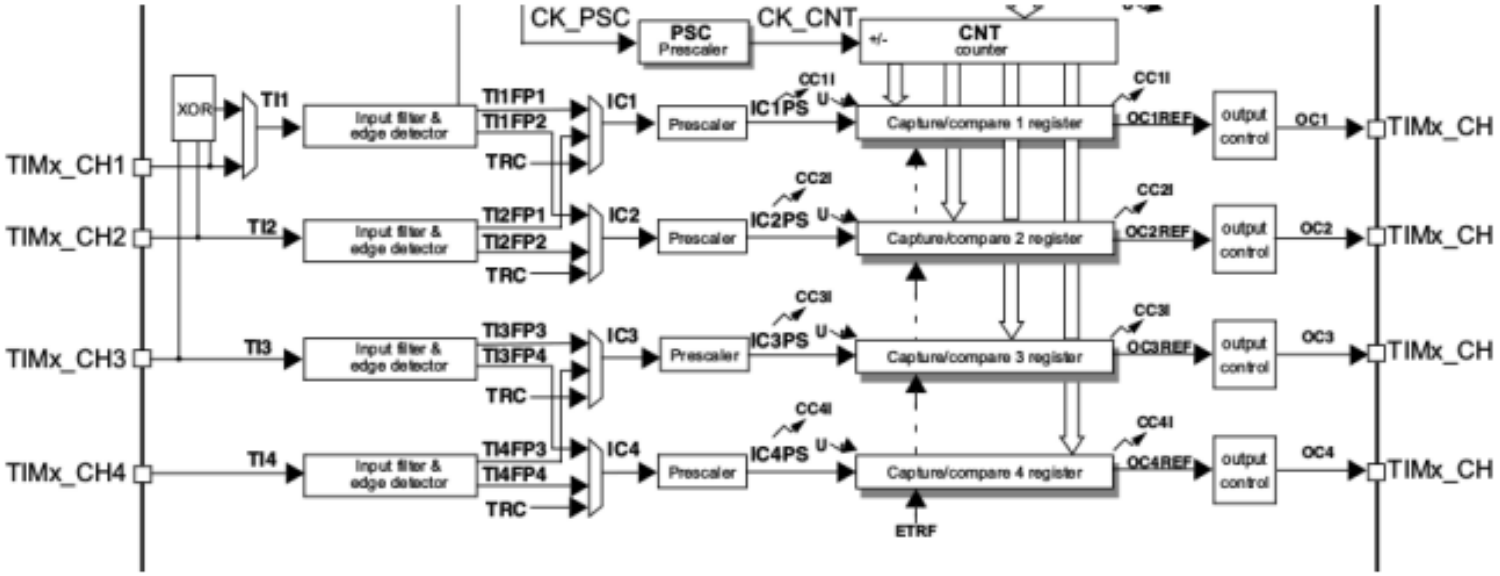
Feature	Advanced-control Timers	General-purpose Timers
PWM Generation	Advanced (e.g., 3-phase)	Basic
Complementary Outputs	Yes	No
Dead-time Insertion	Yes	No
Break Input	Yes	No
Fault Handling	Yes	No
Synchronization	Advanced	Basic
Use Case	Motor control, inverters	Simple timing, PWM, delay
Flexibility	High	Moderate

Independent Watch Dog



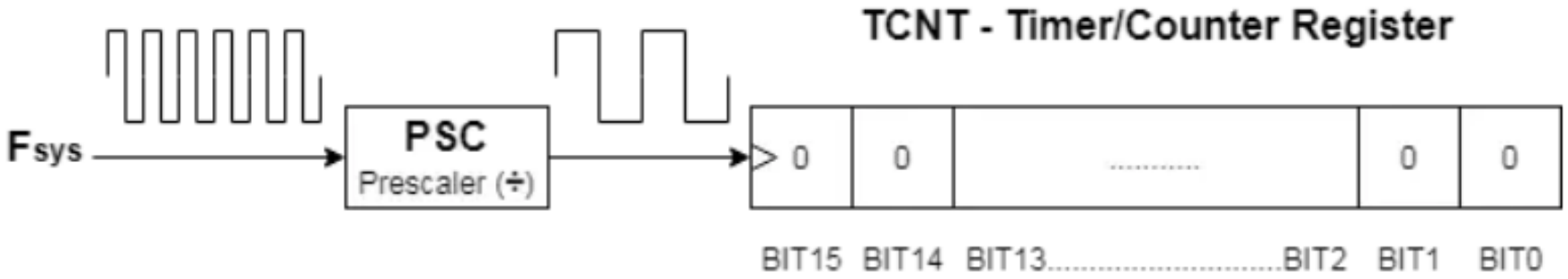
- **8-bit Prescaler Register (IWDG_PR):** Configures the prescaler to adjust the IWDG clock.
- **Reload Register (IWDG_RLR):** Stores the reload value to set the countdown period.
- **Status Register (IWDG_SR):** Indicates the status of key operations.
- **Key Register (IWDG_KR):** Used to reload the watchdog timer or enable the IWDG.

Timers



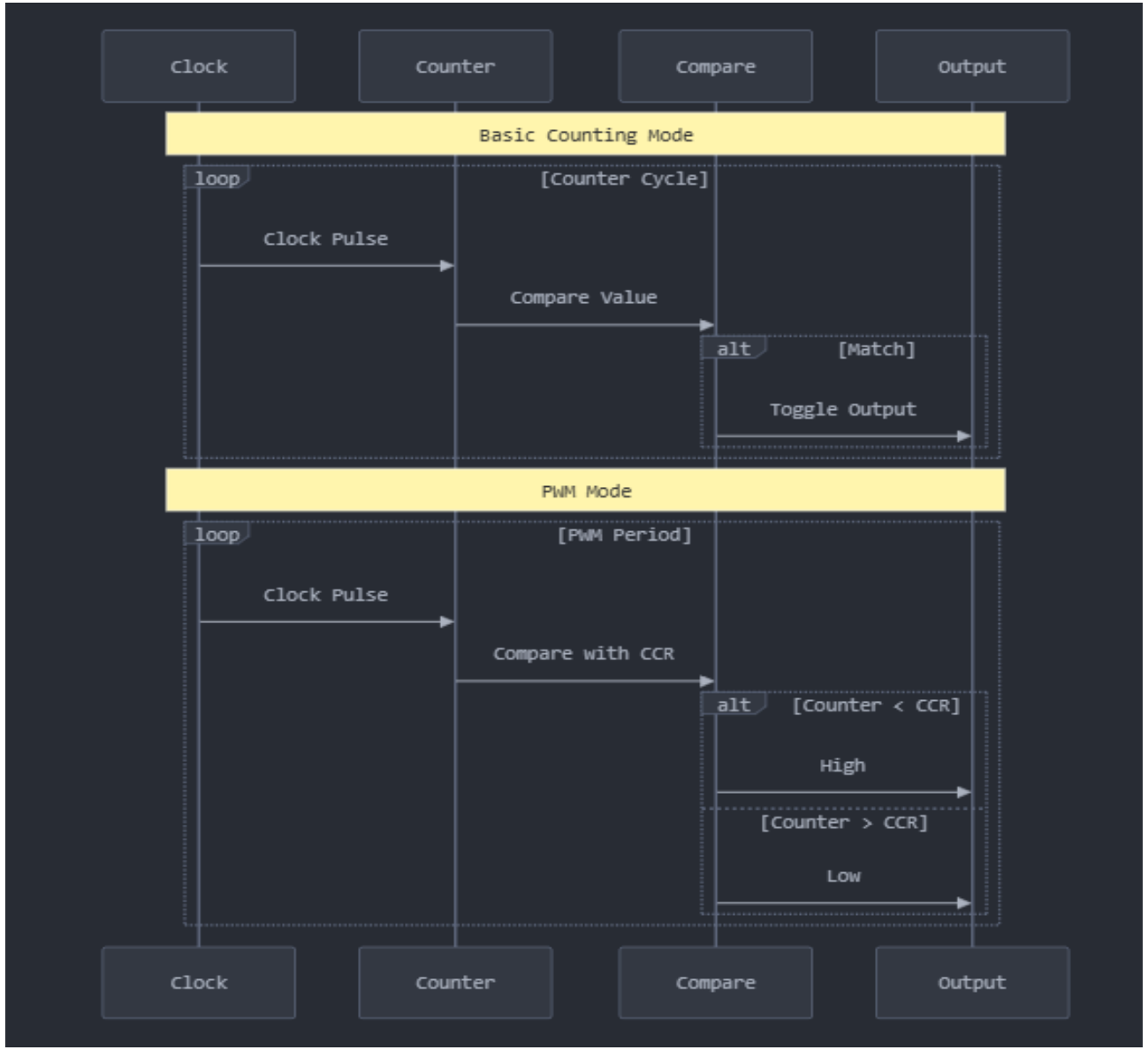
- Each STM32 timer has four slave channels
- Each channel can be connected to a GPIO line and can be programmed for:
  - input capture
  - PWM input measurement
  - output compare
  - PWM output generation
- Each channel as an additional register CCRy (y is the channel number) that has the same size of the counter register

Timer Mode

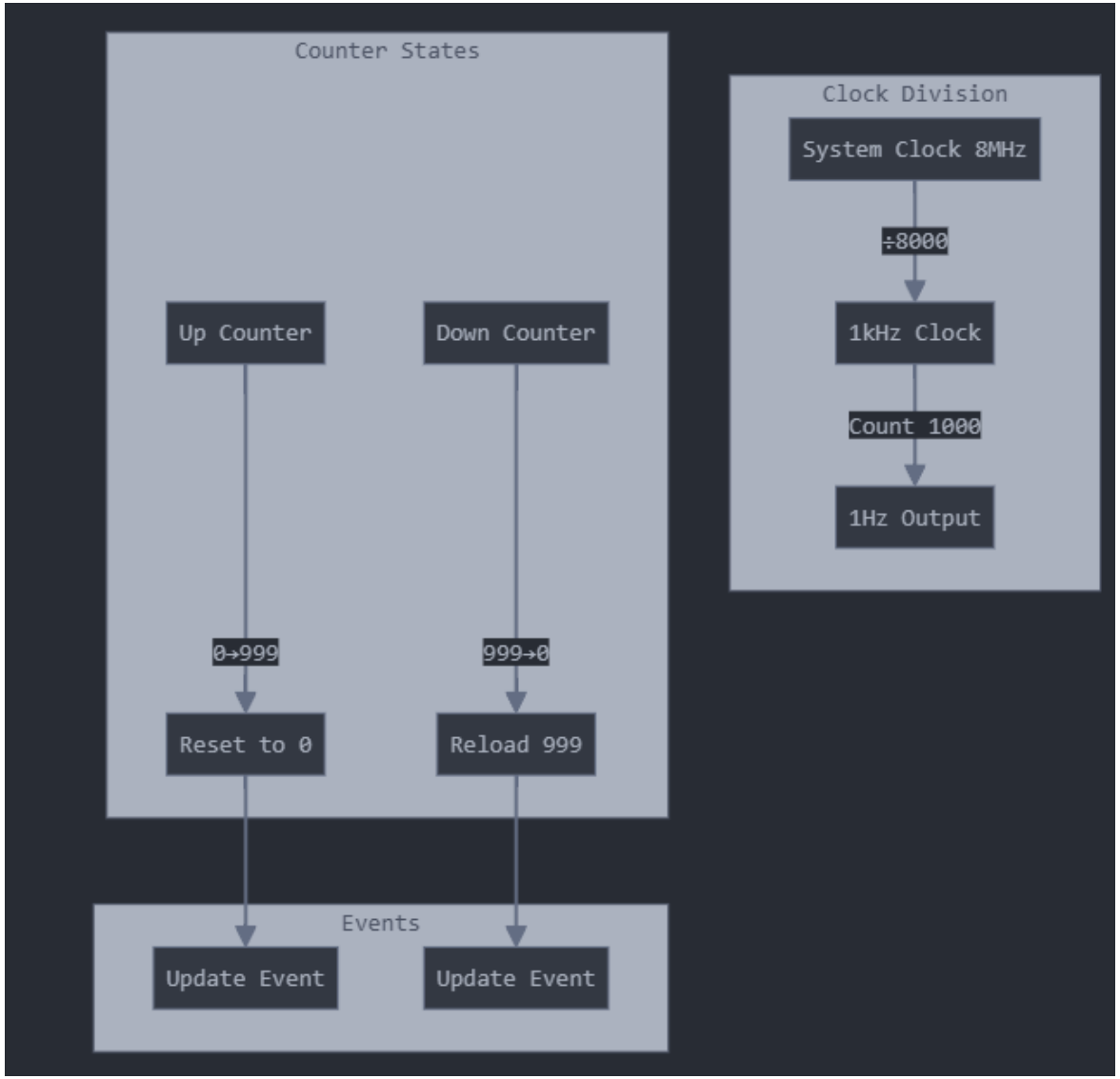


- Basic mode in timers
- Uses internal clock source
- Can be used for delays

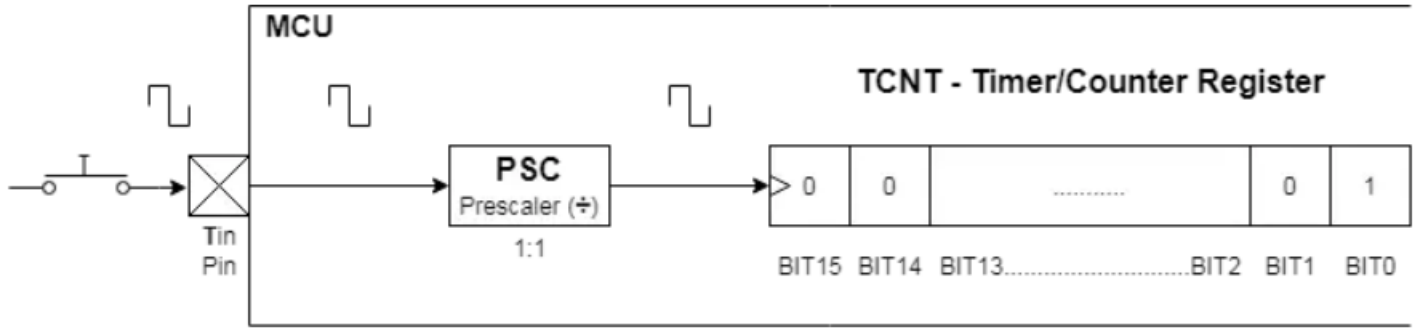
Timer Mode – Flow Diagram



Up & Down Counter

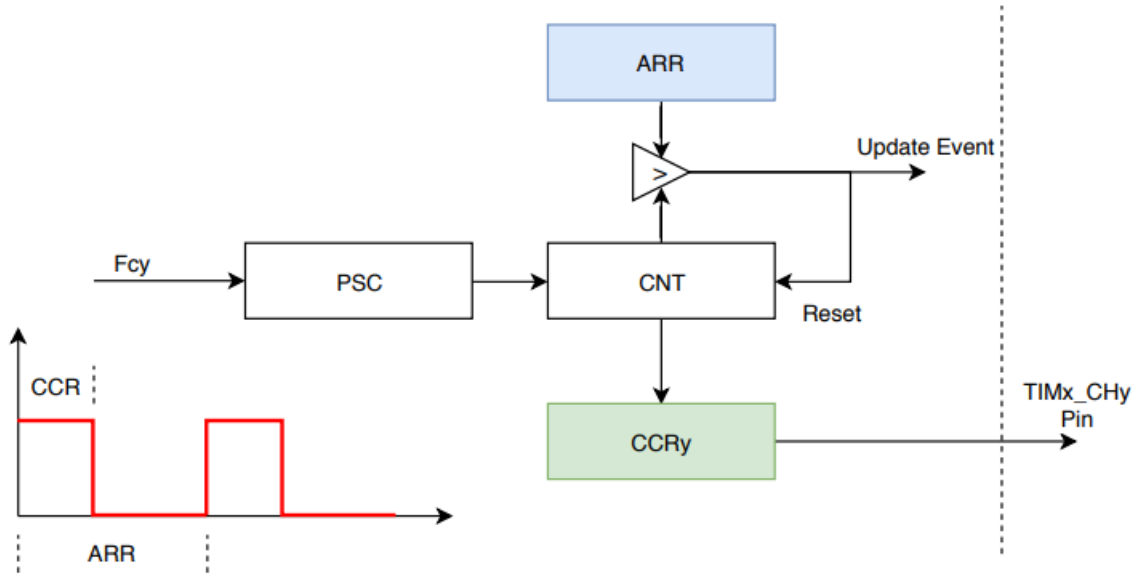


Counter Mode



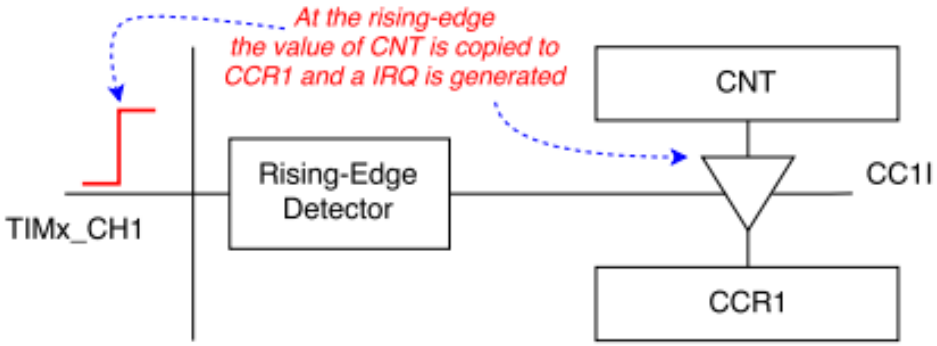
- Clock referenced from external source
- Uses Timer input pin to get in
- the timer counts up or down on each rising or falling edge of the external input

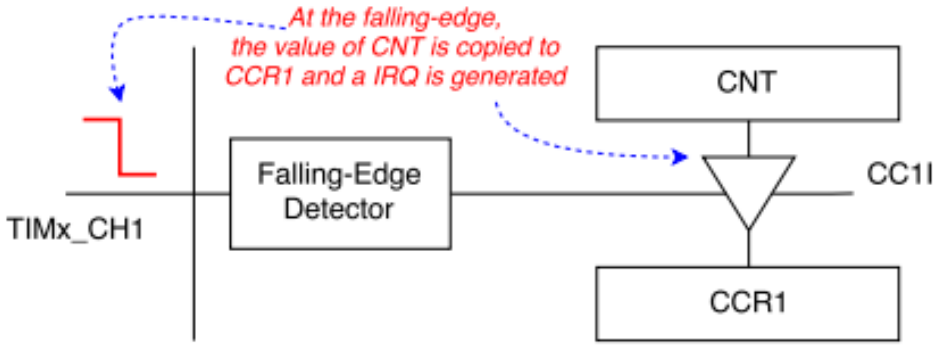
PWM Mode



- **A channel y** can be configured to generate a PWM Signal with specific
- frequency/period and duty cycle
- the **period** is given in terms of count units and the value must be
- assigned to **ARR register**
- the duty cycle is specified as the duration of positive part, it is given in
- terms of count units and the value must be assigned to **CCRy** register

Input Capture

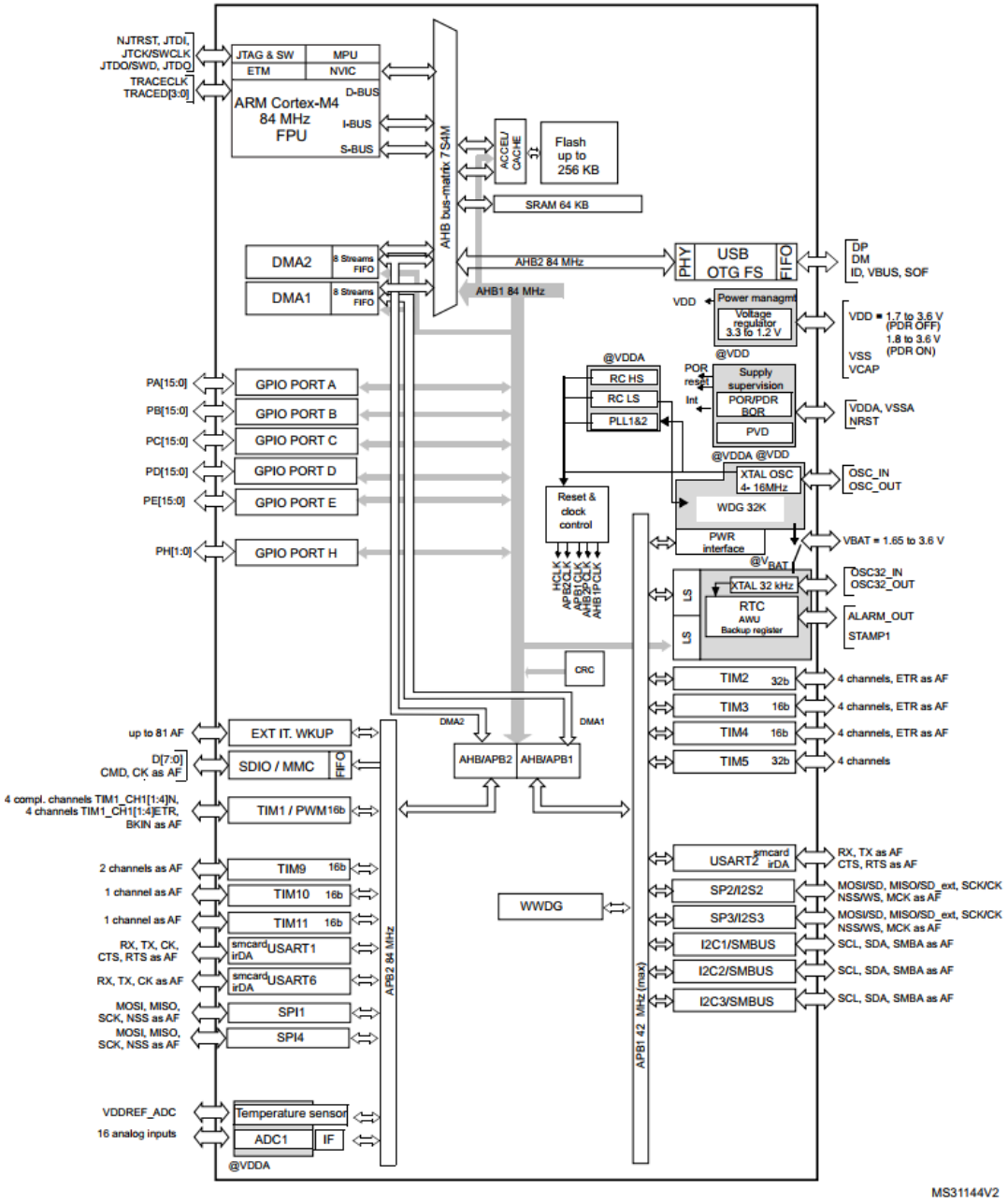




- When a channel y is configured as **input capture** the relevant GPIO line must be configured as input
- The capture circuit is able to detect **an edge (rising or falling**) in the input signal, according to configuration
- Then the edge occurs the value of the **CNT** is copied to **CCRy**
- A **capture-event interrupt** is generated

Timers in STM32F401

STM32F401 Architecture



Datasheet – PG14

Registers of Interest

Thynk Loop

Configuration Registers:

- TIMx_CR1
- TIMx_SR
- TIMx_CCMR1
- TIMx_CCER
- TIMx_CNT
- TIMx_PSC
- TIMx_ARR
- TIMx_CCR1

Source ST Website

TIMx_CR1

Thynk Loop

13.4.1 TIMx control register 1 (TIMx_CR1)

Address offset: 0x00

Reset value: 0x0000

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

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

0: TIMx_ARR register is not buffered

1: TIMx_ARR register is buffered

Bit 0 **CEN**: Counter enable

0: Counter disabled

1: Counter enabled

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)

Ref Manual

TIMX_SR

Thynk Loop

13.4.5 TIMx status register (TIMx_SR)

Address offset: 0x10

Reset value: 0x0000

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

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:

Ref Manual

TIMx_CCMR1

TIMx capture/compare mode register 1 (TIMx_CCMR1)

Address offset: 0x18

Reset value: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
IC2F[3:0]				IC2PSC[1:0]				IC1F[3:0]				IC1PSC[1:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 6:4 **OC1M**: Output compare 1 mode

These bits define the behavior of the output reference signal OC1REF from which OC1 and OC1N are derived. OC1REF is active high whereas OC1 and OC1N active level depends on CC1P and CC1NP bits.

000: Frozen - The comparison between the output compare register TIMx_CCR1 and the counter TIMx_CNT has no effect on the outputs.(this mode is used to generate a timing base).

001: Set channel 1 to active level on match. OC1REF signal is forced high when the counter TIMx_CNT matches the capture/compare register 1 (TIMx_CCR1).

010: Set channel 1 to inactive level on match. OC1REF signal is forced low when the counter TIMx_CNT matches the capture/compare register 1 (TIMx_CCR1).

011: Toggle - OC1REF toggles when TIMx_CNT=TIMx_CCR1.

100: Force inactive level - OC1REF is forced low.

101: Force active level - OC1REF is forced high.

110: PWM mode 1 - In upcounting, channel 1 is active as long as TIMx_CNT<TIMx_CCR1 else inactive. In downcounting, channel 1 is inactive (OC1REF=‘0) as long as TIMx_CNT>TIMx_CCR1 else active (OC1REF=1).

111: PWM mode 2 - In upcounting, channel 1 is inactive as long as TIMx_CNT<TIMx_CCR1 else active. In downcounting, channel 1 is active as long as TIMx_CNT>TIMx_CCR1 else inactive.

Note: In PWM mode 1 or 2, the OCREF level changes only when the result of the comparison changes or when the output compare mode switches from “frozen” mode to “PWM” mode.

Bit 3 **OC1PE**: Output compare 1 preload enable

0: Preload register on TIMx_CCR1 disabled. TIMx_CCR1 can be written at anytime, the new value is taken in account immediately.

1: Preload register on TIMx_CCR1 enabled. Read/Write operations access the preload register. TIMx_CCR1 preload value is loaded in the active register at each update event.

Ref Manual

TIMx_CCER

Thynk Loop

13.4.9 TIMx capture/compare enable register (TIMx_CCER)

Address offset: 0x20

Reset value: 0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

Bit 0 **CC1E**: *Capture/Compare 1 output enable.*

CC1 channel configured as output:

0: Off - OC1 is not active

1: On - OC1 signal is output on the corresponding output pin

CC1 channel configured as input:

This bit determines if a capture of the counter value can actually be done into the input capture/compare register 1 (TIMx_CCR1) or not.

0: Capture disabled

1: Capture enabled

Ref Manual

TIMx_CNT

Thynk Loop

13.4.10 TIMx counter (TIMx_CNT)

Address offset: 0x24

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNT[31:16] (depending on timers)															
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
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:16 **CNT[31:16]**: *High counter value (on TIM2 and TIM5).*

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

Ref Manual

TIMx_PSC

Thynk Loop

13.4.11 TIMx prescaler (TIMx_PSC)

Address offset: 0x28

Reset value: 0x0000

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

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 (including when the counter is cleared through UG bit of TIMx_EGR register or through trigger controller when configured in “reset mode”).

Ref Manual

TIMx_ARR

13.4.12 TIMx auto-reload register (TIMx_ARR)

Address offset: 0x2C

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ARR[31:16] (depending on timers)															
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
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Ref Manual

TIMx_CCR1

13.4.13 TIMx capture/compare register 1 (TIMx_CCR1)

Address offset: 0x34

Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1[31:16] (depending on timers)															
rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1[15:0]															
rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro	rw/ro

Bits 31:16 **CCR1[31:16]**: High Capture/Compare 1 value (on TIM2 and TIM5).

Bits 15:0 **CCR1[15:0]**: Low Capture/Compare 1 value

If channel CC1 is configured as output:
CCR1 is the value to be loaded in the actual capture/compare 1 register (preload value).
It is loaded permanently if the preload feature is not selected in the TIMx_CCMR1 register (bit OC1PE). Else the preload value is copied in the active capture/compare 1 register when an update event occurs.
The active capture/compare register contains the value to be compared to the counter TIMx_CNT and signaled on OC1 output.

If channel CC1is configured as input:
CCR1 is the counter value transferred by the last input capture 1 event (IC1). The TIMx_CCR1 register is read-only and cannot be programmed.

Ref Manual

Timer Delay Sequence

TIMER DELAY CONFIGURATION SEQUENCE

Thynk Loop

→Enable TIM2

→Configure PSC

→Configure CR1 to One pulse mode

→Set the ARR to the delay value

→Reset the counter

→Clear the interrupt flag in SR

→Enable the timer in CR1

→Check for the update interrupt flag in SR register

PWM Sequence

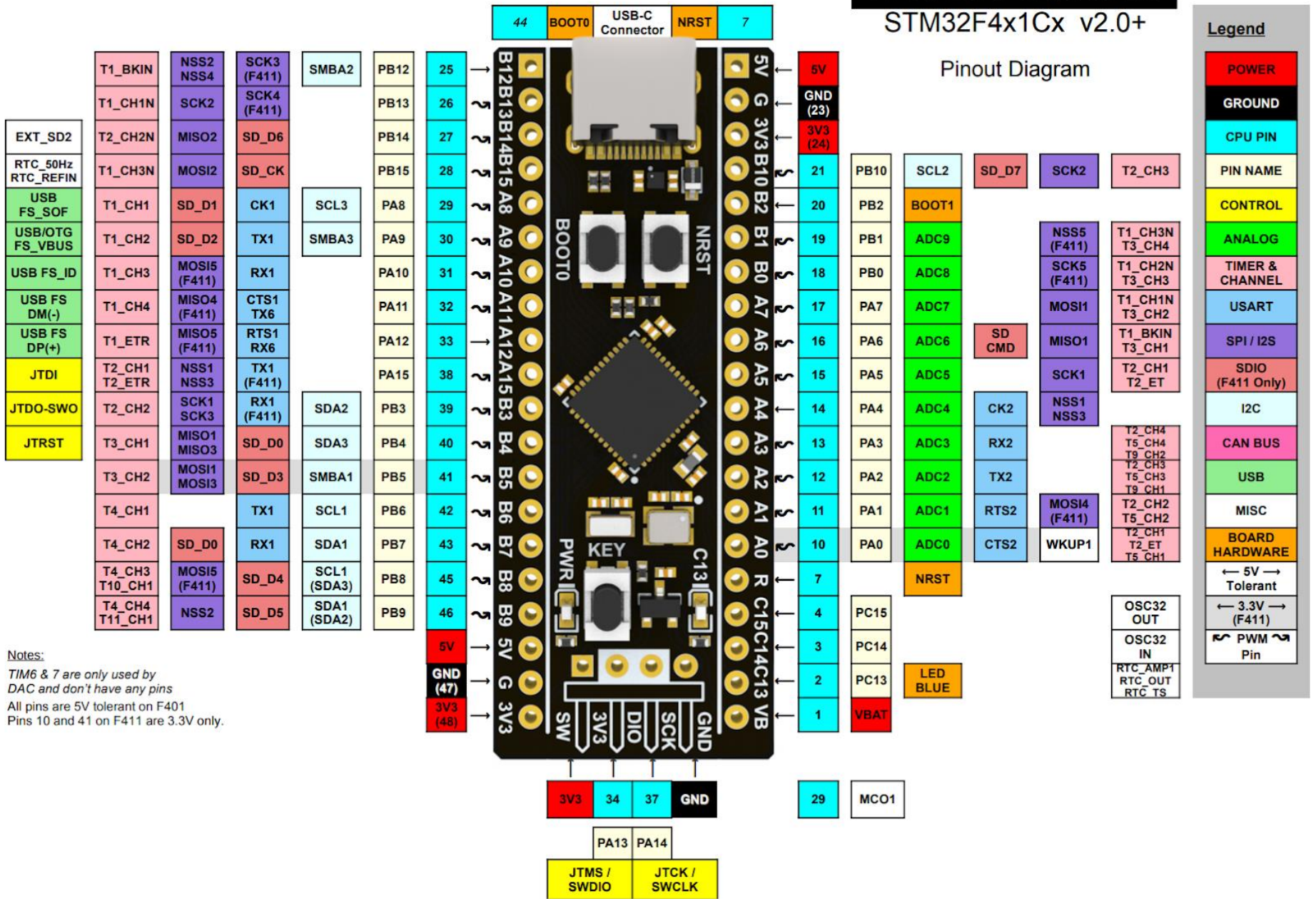
PWM CONFIGURATION SEQUENCE

Thynk Loop

→Enable TIM3
→Enable GPIOA
→Configure GPIOA PIN 6 in ALT FUNC
→Configure PSC
→Configure ARR
→Configure CCR1
→Configure CCMR1 – PWM mode 1, Enable preload
→Configure CCER – Enable output channel 1
→Enable auto-reload preload in CR1
→Enable the timer in CR1

→Change CCR1 for in a loop or something to create the PWM

BlackPill – Pin Configurations



Notes:

TIM6 & 7 are only used by DAC and don't have any pins

All pins are 5V tolerant on F401
Pins 10 and 41 on F411 are 3.3V only.

GitHub Repo

Thynk Loop