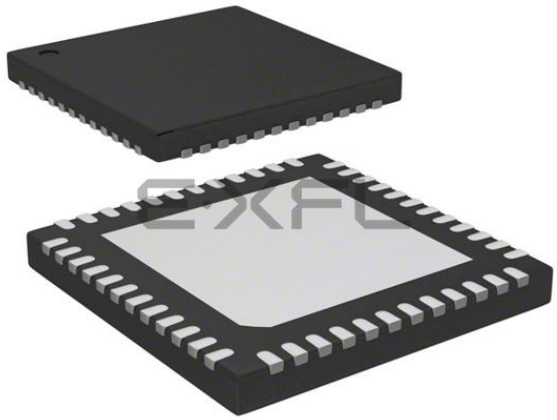


Welcome to E-XFL.COM

Embedded - Microcontrollers - Application Specific: Tailored Solutions for Precision and Performance

Embedded - Microcontrollers - Application Specific represents a category of microcontrollers designed with unique features and capabilities tailored to specific application needs. Unlike general-purpose microcontrollers, application-specific microcontrollers are optimized for particular tasks, offering enhanced performance, efficiency, and functionality to meet the demands of specialized applications.

What Are Embedded - Microcontrollers - Application Specific?

Application-specific microcontrollers are engineered to

Details	
Product Status	Obsolete
Applications	RF4CE, Remote Control
Core Processor	ARM® Cortex®-M3
Program Memory Type	FLASH (256kB)
Controller Series	STM32W
RAM Size	16K x 8
Interface	I²C, SPI, UART/USART
Number of I/O	24
Voltage - Supply	1.18V ~ 3.6V
Operating Temperature	-40°C ~ 105°C
Mounting Type	Surface Mount
Package / Case	48-UFQFN Exposed Pad
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/stmicroelectronics/stm32w108ccu74tr

The STM32W108xx has an ultra low power, deep sleep state with a choice of clocking modes. The sleep timer can be clocked with either the external 32.768 kHz crystal oscillator or with a 1 kHz clock derived from the internal 10 kHz LSI RC oscillator. Alternatively, all clocks can be disabled for the lowest power mode. In the lowest power mode, only external events on GPIO pins will wake up the chip. The STM32W108xx has a fast startup time (typically 100 µs) from deep sleep to the execution of the first ARM® Cortex®-M3 instruction.

The STM32W108xx contains three power domains. The always-on high voltage supply powers the GPIO pads and critical chip functions. Regulated low voltage supplies power the rest of the chip. The low voltage supplies are be disabled during deep sleep to reduce power consumption. Integrated voltage regulators generate regulated 1.25 V and 1.8 V voltages from an unregulated supply voltage. The 1.8 V regulator output is decoupled and routed externally to supply analog blocks, RAM, and Flash memories. The 1.25 V regulator output is decoupled externally and supplies the core logic.

The digital section of the receiver uses a coherent demodulator to generate symbols for the hardware-based MAC. The digital receiver also contains the analog radio calibration routines and controls the gain within the receiver path.

In addition to 2 general-purpose timers, the STM32W108xx also contains a watchdog timer to ensure protection against software crashes and CPU lockup, a 32-bit sleep timer dedicated to system timing and waking from sleep at specific times and an ARM® standard system event timer in the NVIC.

The STM32W108xx integrates hardware support for a Packet Trace module, which allows robust packet-based debug.

Note: The STM32W108xx is not pin-compatible with the previous generation chip, the SN250, except for the RF section of the chip. Pins 1-11 and 45-48 are compatible, to ease migration to the STM32W108xx.

1.2.2 ARM® Cortex®-M3 core

The STM32W108xx integrates the ARM® Cortex®-M3 microprocessor, revision r1p1, developed by ARM Ltd, making the STM32W108xx a true system-on-a-chip solution. The ARM® Cortex®-M3 is an advanced 32-bit modified Harvard architecture processor that has separate internal program and data buses, but presents a unified program and data address space to software. The word width is 32 bits for both the program and data sides. The ARM® Cortex®-M3 allows unaligned word and half-word data accesses to support efficiently-packed data structures.

The ARM® Cortex®-M3 clock speed is configurable to 6 MHz, 12 MHz, or 24 MHz. For normal operation 12 MHz is preferred over 24 MHz due to its lower power consumption. The 6 MHz operation can only be used when radio operations are not required since the radio requires an accurate 12 MHz clock.

The ARM® Cortex®-M3 in the STM32W108xx has also been enhanced to support two separate memory protection levels. Basic protection is available without using the MPU, but the usual operation uses the MPU. The MPU protects unimplemented areas of the memory map to prevent common software bugs from interfering with software operation. The architecture could also separate the networking stack from the application code using a fine granularity RAM protection module. Errant writes are captured and details are reported to the developer to assist in tracking down and fixing issues.

6.3.6 Clock switching registers

Clock sleep mode control register (CLK_SLEEPCR)

The sleep timer controls the low power clock gated modes.

Clearing the LSI10KEN bit in the CLK_SLEEPCR register before executing WFI with the SLEEPDEEP bit set to '1' in the SCB_SCR register (for more details refer to the Cortex-M3 Programming manual PM0056) causes deep sleep 2 mode to be entered. Setting the LSI10KEN bit in the CLK_SLEEPCR register causes deep sleep 1 mode to be entered.

Address: 0x4000 0008
Reset value: 0x0000 0002

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														LSI10KEN	LSEEN
														rw	rw

Bits 31:2 Reserved, must be kept at reset value

Bit 1 LSI10KEN:
1: Enables 10 kHz internal RC during deep sleep mode.
2: Disables 10 kHz internal RC during deep sleep mode

Bit 0 LSEEN:
1: Enables 32 kHz external oscillator during deep sleep mode.
2: Disables 32 kHz external oscillator during deep sleep mode.

Low-speed internal 10 KHz clock (LSI10K) control register (CLK_LSI10KCR)

Address: 0x4000 000C
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												TUNE[3:0]			
												rw	rw	rw	rw

Bits 1:4 Reserved, must be kept at reset value

Bits 3:0 TUNE[3:0]:
Tunes the value for the HSI clock.

Low-speed internal 1 KHz clock control register (CLK_LSI1KCR)

Address: 0x4000 0010
Reset value: 0x0000 5000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALINT[4:0]					CLKFRAC[10:0]										
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 15:11 CALINT[4:0]:
Divider value integer portion.

Bits 10:0 CALINT[10:0]:
Divider value fractional portion.

High-speed external clock control register 1 (CLK_HSECR1)

Address: 0x4000 4004
Reset value: 0x0000 000F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												BIASTRIM[3:0]			
												rw	rw	rw	rw

Bits 31:4 Reserved, must be kept at reset value

Bits 3:0 BIASTRIM[3:0]:
Bias trim setting for 24-MHz oscillator. Reset to full bias power up. May be overwritten in software.

Clock switching (CLK) register map

Table 9 gives the CLK register map and reset values.

Table 9. CLK register map and reset values

Offset	Register	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
0x0008	CLK_SLEEPCR	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.	LSI10KEN	LSEEN
	Reset value																															1	0
0x000C	CLK_LSI10KCR	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.	TUNE[3:0]			
	Reset value																													0	0	0	0
0x0010	CLK_LSI1KCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALINT[4:0]					CLKFRAC[10:0]										
	Reset value																	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
0x0014-0x4000	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.	Res.
0x4004	CLK_HSECR1	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.	BIASTRIM[3:0]			
	Reset value																													1	1	1	1
0x4008	CLK_HSICR	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.	TUNE[4:0]			
	Reset value																												0	1	1	1	1
0x400C	CLK_HSECOMPR	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.	HLEVEL	LLEVEL
	Reset value																															0	0
0x4010	CLK_PERIODCR	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.	MODE[1:0]	
	Reset value																															0	0
0x4014	CLK_PERIODSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERIOD[15:0]															
	Reset value																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4018	CLK_DITHERCR	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.	DIS
	Reset value																																0
0x401C	CLK_HSECR2	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.	EN	SW1
	Reset value																															0	0
0x4020	CLK_CPUCR	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.	SW2
	Reset value																																0

Refer to *Figure 4: STM32W108xB memory mapping*, *Figure 5: STM32W108CC and STM32W108CZ memory mapping*, and *Table 3: STM32W108xx peripheral register boundary addresses* for the register boundary addresses of the peripherals available in all STM32W108xx devices.

The SPI master controller uses the three signals:

- MOSI (Master Out, Slave In) - outputs serial data from the master
- MISO (Master In, Slave Out) - inputs serial data from a slave
- SCLK (Serial Clock) - outputs the serial clock used by MOSI and MISO

The GPIO pins used for these signals are shown in *Table 21*. Additional outputs may be needed to drive the nSSEL signals on slave devices.

Table 21. SPI master GPIO usage

Parameter	MOSI	MISO	SCLK
Direction	Output	Input	Output
GPIO configuration	Alternate Output (push-pull)	Input	Alternate Output (push-pull) Special SCLK mode
SC1 pin	PB1	PB2	PB3
SC2 pin	PA0	PA1	PA2

9.3.1 Setup and configuration

Both serial controllers, SC1 and SC2, support SPI master mode. SPI master mode is enabled by the following register settings:

- The serial controller mode register (SCx_CR) is ‘2’.
- The MSTR bit in the SPI configuration register (SCx_SPICR) is ‘1’.
- The ACK bit in the I^2C control register (SCx_I2CCR2) is ‘1’.

The SPI serial clock (SCLK) is produced by a programmable clock generator. The serial clock is produced by dividing down 12 MHz according to this equation:

$$Rate = \frac{12MHz}{(LIN + 1)x2^{EXP}}$$

EXP is the value written to the SCx_CRR2 register and LIN is the value written to the SCx_CRR1 register. The SPI master mode clock may not exceed 6 Mbps, so EXP and LIN cannot both be zero.

The SPI master controller supports various frame formats depending upon the clock polarity (CPOL), clock phase (CPHA), and direction of data (LSBFIRST) (see *SPI master mode formats on page 114*). The bits CPOL, CPHA, and LSBFIRST are defined within the SCx_SPICR register.

9.10 Serial controller: Inter-integrated circuit (I^2C) registers

9.10.1 Serial controller I^2C status register (SCx_I2CSR)

Address offset: 0xC844 (SC1_I2CSR) and 0xC044 (SC2_I2CSR)
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CMD FIN	BRF	BTF	NACK
												r	r	r	r

Bits 31:4 Reserved, must be kept at reset value

Bit 3 CMDFIN: Command finished flag

This bit is set when a START or STOP command completes. It is cleared on the next I^2C bus activity.

0: START/ STOP command transmission not done

1: START/ STOP command transmission succeeded

Bit 2 BRF: Byte receive finished flag

This bit is set when a byte is received. It clears on the next I^2C bus activity.

0: Data byte reception not done

1: Data byte reception succeeded

Bit 1 BTF: Byte transfer finished flag

This bit is set when a byte is transmitted. It clears on the next I^2C bus activity.

0: Data byte transmission not done

1: Data byte transmission succeeded

Bit 0 NACK: Not acknowledge flag

This bit is set when a NACK is received from the slave. It clears on the next I^2C bus activity.

0: No NACK received

1: NACK receive succeeded

9.12.5 Serial controller transmit DMA begin address channel A register (SCx_DMATXBEGADDAR)

Address offset: 0xC810 (SC1_DMATXBEGADDAR) and 0xC010 (SC2_DMATXBEGADDAR)
Reset value: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			ADD[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:13 Reserved, must be kept at reset value

Bits 12:0 ADD[12:0]: DMA transmit buffer A start address

9.12.6 Serial controller transmit DMA end address channel A register (SCx_DMATXENDADDAR)

Address offset: 0xC814 (SC1_DMATXENDADDAR) and 0xC014 (SC2_DMATXENDADDAR)
Reset value: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			ADD[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:13 Reserved, must be kept at reset value

Bits 12:0 ADD[12:0]: Address of the last byte that is read from the DMA transmit buffer A

9.12.7 Serial controller transmit DMA begin address channel B register (SCx_DMATXBEGADDBR)

Address offset: 0xC818 (SC1_DMATXBEGADDBR) and 0xC018 (SC2_DMATXBEGADDBR)
Reset value: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			ADD[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:13 Reserved, must be kept at reset value

Bits 12:0 ADD[12:0]: DMA transmit buffer B start address

9.12.8 Serial controller transmit DMA end address channel B register (SCx_DMATXENDADDBR)

Address offset: 0xC81C (SC1_DMATXENDADDBR) and 0xC01C (SC2_DMATXENDADDBR)
Reset value: 0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			ADD[12:0]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:13 Reserved, must be kept at reset value

Bits 12:0 ADD[12:0]: Address of the last byte that is read from the DMA transmit buffer B

Table 31. SC1/SC2 register map and reset values (continued)

Offset	Register	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
0xC80C	SC1_DMARX ENDADDBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC810	SC1_DMATX BEGADDAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC814	SC1_DMATX ENDADDAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC818	SC1_DMATX BEGADDBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC81C	SC1_DMATX ENDADDBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC820	SC1_DMARX CNTAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC824	SC1_DMARX CNTBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC828	SC1_DMATX CNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC82C	SC1_DMASR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NSSS [2:0]			FEB	FEA	PEB	PEA	OVRB	OVRA	TXBACK	TXAACK	RXBACK	RXAACK
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC830	SC1_DMACR	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.	TXRST	RXRST	TXLODB	TXLODA	RXLODB	RXLODA
	Reset value																											0	0	0	0	0	0
0xC834	SC1_DMARX ERRAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC838	SC1_DMARX ERRBR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADD[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xC870	SC1_DMARX CNTSAVEDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[12:0]												
	Reset value																				0	0	0	0	0	0	0	0	0	0	0	0	0
0xA80C	SC2_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PE	FE	TXULODB	TXULODA	RXULODB	RXULODA	NACK	CMDFIN	BTF	BRF	UDR	OVR	IDLE	TXE	RXNE
	Reset value																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA84C	SC2_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEIE	FEIE	TXULODRIE	TXULODAIE	RXULODBIE	RXULODAIE	NACKIE	CMDFINIE	BTFIE	BRFIE	UDRIE	OVRIE	IDLEIE	TXEIE	RXNEIE
	Reset value																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

10.1.2 Counter modes

Up-counting mode

In up-counting mode, the counter counts from 0 to the auto-reload value (contents 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, by setting the UG bit in the TIMx_EGR register, or by using the slave mode controller.

Software can disable the update event by setting the UDIS bit in the TIMx_CR1 register, to avoid updating the shadow registers while writing new values in the buffer registers. No update event will occur until the UDIS bit is written to 0. Both the counter and the prescalar counter restart from 0, but the prescale rate does not change. In addition, if the URS bit in the TIMx_CR1 register is set, setting the UG bit generates an update event but without setting the UIF flag. Thus no interrupt request is sent. This avoids generating both update and capture interrupts when clearing the counter on the capture event.

When an update event occurs, the update flag (the UIF bit in the TIMx_SR register) is set (depending on the URS bit in the TIMx_CR1 register) and the following registers are updated:

- The buffer of the prescaler is reloaded with the buffer value (contents of the TIMx_PSC register).
- The auto-reload shadow register is updated with the buffer value (TIMx_ARR).

Figure 17, *Figure 18*, *Figure 19*, and *Figure 20* show some examples of the counter behavior for different clock frequencies when TIMx_ARR = 0x36.

Figure 17. Counter timing diagram, internal clock divided by 1

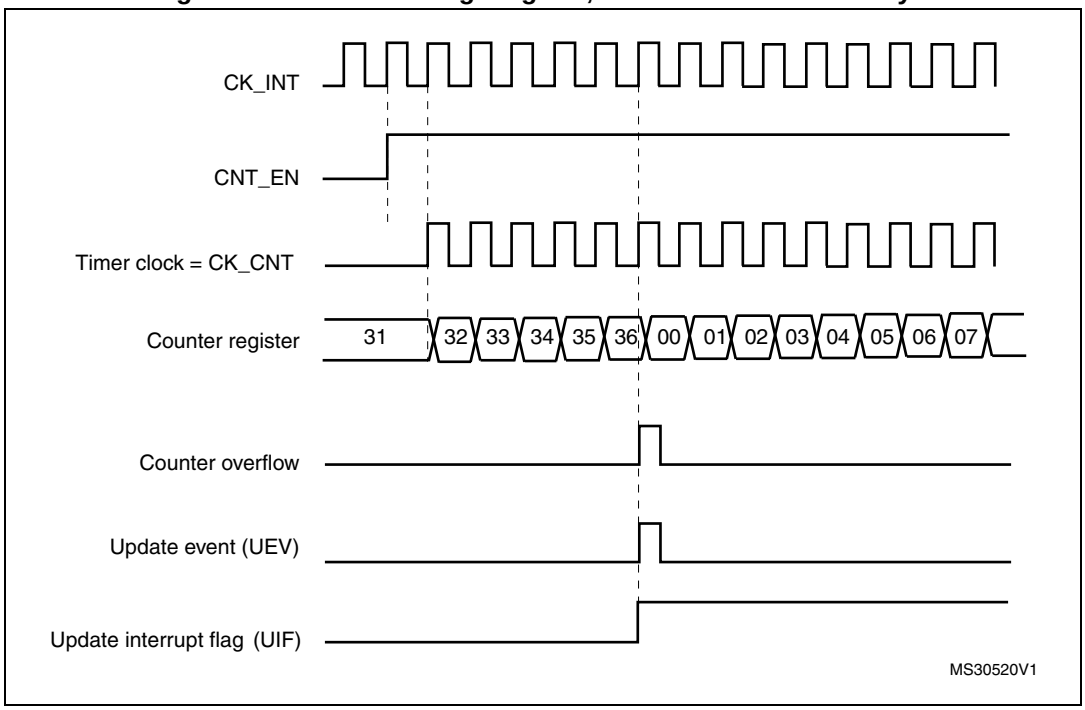


Figure 24. Counter timing diagram, update event with ARPE = 1 (counter underflow)

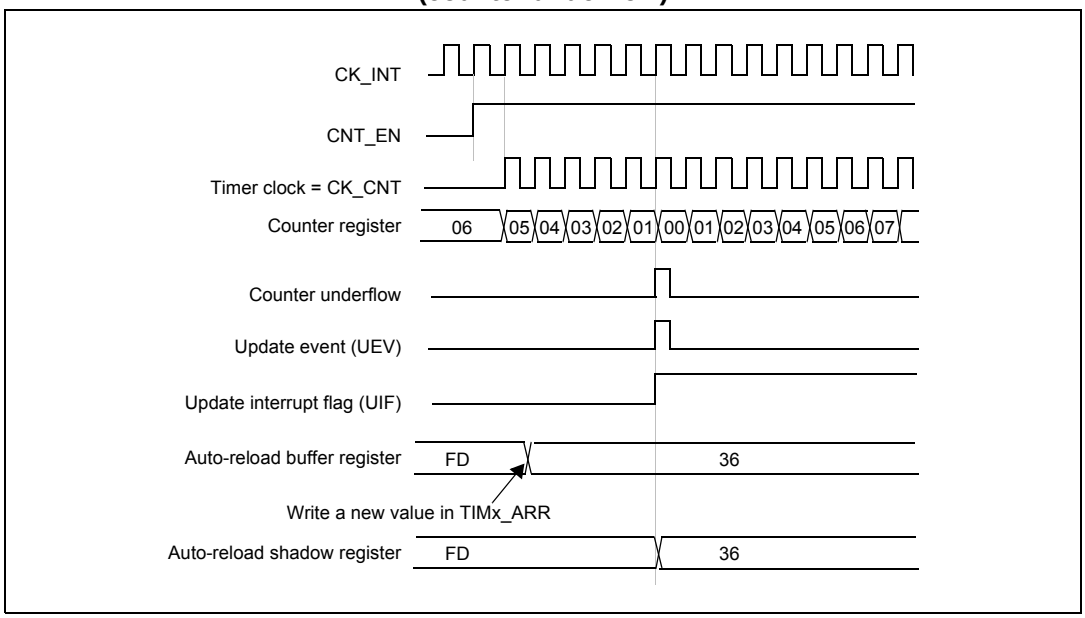
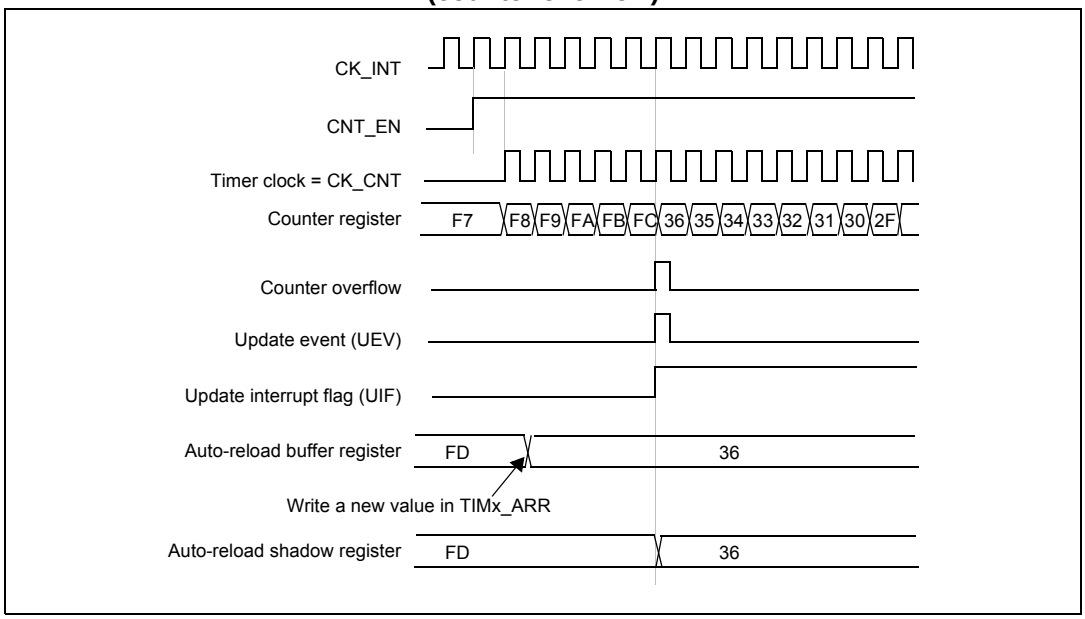


Figure 25. Counter timing diagram, update event with ARPE = 1 (counter overflow)



Hints on using center-aligned mode:

- When starting in center-aligned mode, the current up-down configuration is used. This means that the counter counts up or down depending on the value written in the DIR bit in the TIMx_CR1 register. The DIR and CMS bits must not be changed at the same time by the software.
- Writing to the counter while running in center-aligned mode is not recommended as it can lead to unexpected results. In particular:
- The direction is not updated the value written to the counter that is greater than the auto-reload value (TIMx_CNT > TIMx_ARR). For example, if the counter was counting up, it continues to count up.
- The direction is updated if when 0 or the TIMx_ARR value is written to the counter, but no update event is generated.
- The safest way to use center-aligned mode is to generate an update by software (setting the UG bit in the TIMx_EGR register) just before starting the counter, and not to write the counter while it is running.

10.1.10 One-pulse mode

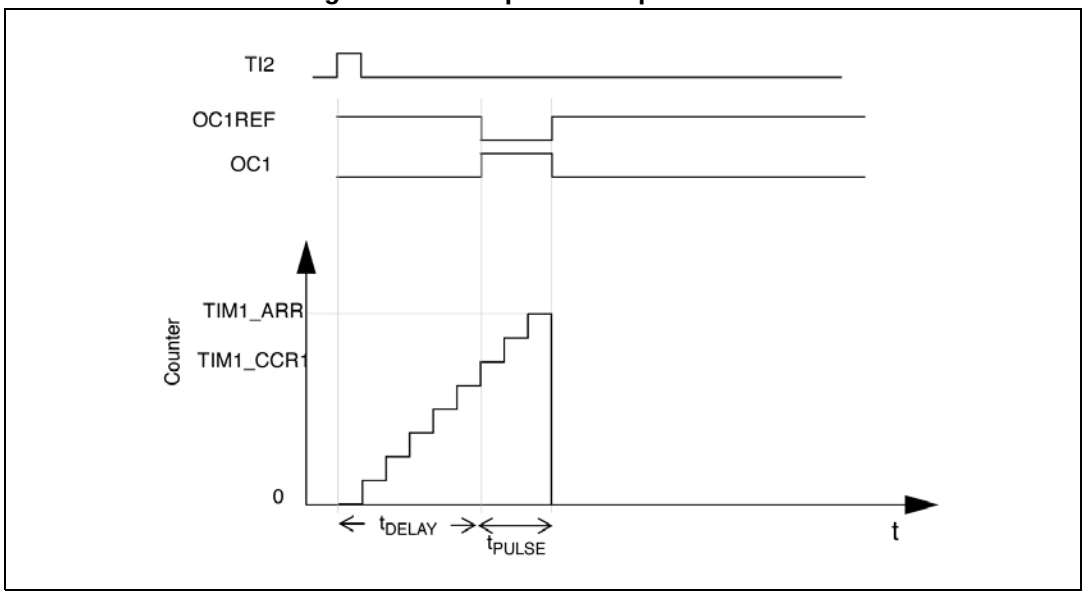
One-pulse mode (OPM) is a special case of the previous modes. It allows the counter to be started in response to a stimulus and to generate a pulse with a programmable length after a programmable delay.

Starting the counter can be controlled through the slave mode controller. Generating the waveform can be done in output compare mode or PWM mode. Select OPM by setting the OPM bit in the TIMx_CR1 register. This makes the counter stop automatically at the next update event.

A pulse can be correctly generated only if the compare value is different from the counter initial value. Before starting (when the timer is waiting for the trigger), the configuration must be:

- In up-counting: TIMx_CNT < TIMx_CCRy ≤TIMx_ARR (in particular, 0 < TIMx_CCRy),
- In down-counting: TIMx_CNT > TIMx_CCRy.

Figure 38. Example of one pulse mode



For example, to generate a positive pulse on OC1 with a length of tPULSE and after a delay of tDELAY as soon as a rising edge is detected on the TI2 input pin, using TI2FP2 as trigger 1:

- Map TI2FP2 on TI2 by writing IC2S = 01 in the TIMx_CCMR1 register.
- TI2FP2 must detect a rising edge. Write CC2P = 0 in the TIMx_CCER register.
- Configure TI2FP2 as trigger for the slave mode controller (TRGI) by writing TS = 110 in the TIMx_SMCR register.
- TI2FP2 is used to start the counter by writing SMS to 110 in the TIMx_SMCR register (trigger mode).
- The OPM waveform is defined: Write the compare registers, taking into account the clock frequency and the counter prescaler.

The t_{DELAY} is defined by the value written in the TIMx_CCR1 register.

The tPULSE is defined by the difference between the auto-reload value and the compare value (TIMx_ARR - TIMx_CCR1).

To build a waveform with a transition from 0 to 1 when a compare match occurs and a transition from 1 to 0 when the counter reaches the auto-reload value, enable PWM mode 2 by writing OC1M = 111 in the TIMx_CCMR1 register. Optionally, enable the buffer registers by writing OC1PE = 1 in the TIMx_CCMR1 register and ARPE in the TIMx_CR1 register. In this case, also write the compare value in the TIMx_CCR1 register, the auto-reload value in the TIMx_ARR register, generate an update by setting the UG bit, and wait for external trigger event on TI2. CC1P is written to 0 in this example.

In the example, the DIR and CMS bits in the TIMx_CR1 register should be low.

Since only one pulse is desired, software should set the OPM bit in the TIMx_CR1 register to stop the counter at the next update event (when the counter rolls over from the auto-reload value back to 0).

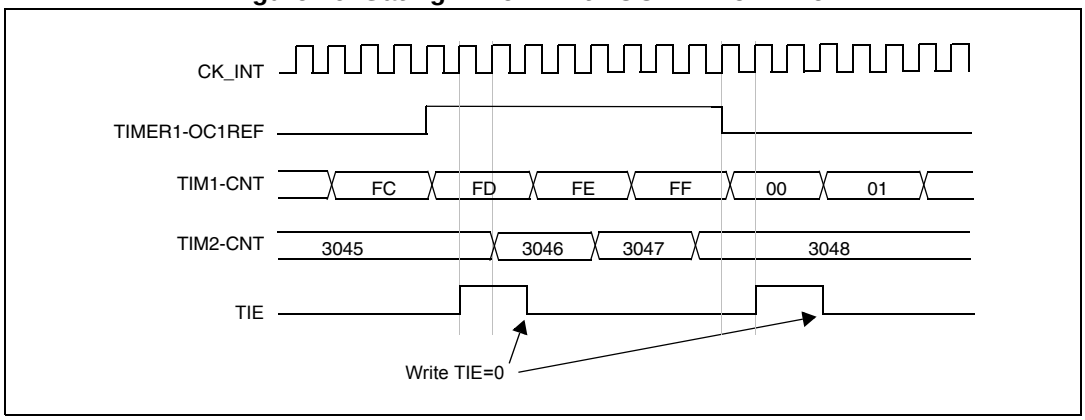
Using one timer to enable the other timer

In this example, the enable of Timer 2 is controlled with the output compare 1 of Timer 1. Refer to *Figure 45* for connections. Timer 2 counts on the divided internal clock only when OC1REF of Timer 1 is high. Both counter clock frequencies are divided by 3 by the prescaler compared to CK_INT (fCK_CNT = fCK_INT /3).

- Configure Timer 1 in master mode to send its Output Compare Reference (OC1REF) signal as trigger output (MMS = 100 in the TIM1_CR2 register).
- Configure the Timer 1 OC1REF waveform (TIM1_CCMR1 register).
- Configure Timer 2 to get the input trigger from Timer 1 (TS = 000 in the TIM2_SMCR register).
- Configure Timer 2 in Gated mode (SMS = 101 in the TIM2_SMCR register).
- Enable Timer 2 by writing 1 in the CEN bit (TIM2_CR1 register).
- Start Timer 1 by writing 1 in the CEN bit (TIM1_CR1 register).

Note: The counter 2 clock is not synchronized with counter 1, this mode only affects the Timer 2 counter enable signal.

Figure 46. Gating Timer 2 with OC1REF of Timer 1



In the example in *Figure 46*, the Timer 2 counter and prescaler are not initialized before being started. So they start counting from their current value. It is possible to start from a given value by resetting both timers before starting Timer 1, then writing the desired value in the timer counters. The timers can easily be reset by software using the UG bit in the TIMx_EGR registers.

10.2 Interrupts

Several kinds of timer events can generate a timer interrupt, and each has a status flag in the TIMx_SR register to identify the reason(s) for the interrupt:

- TIE - set by a rising edge on an external trigger, either edge in gated mode
- CCyIF - set by a channel y input capture or output compare event
- UIF - set by an update event

Clear bits in TIMx_SR by writing a 1 to their bit position. When a channel is in capture mode, reading the TIMx_CCRy register will also clear the CCyIF bit.

The TIMx_IER register controls whether or not the TIMx_SR bits actually request an ARM® Cortex®-M3 timer interrupt. Only the events whose bits are set to 1 in TIMx_IER can do so.

If an input capture or output compare event occurs and its CCyIM is already set, the corresponding capture/compare missed flag is set in the TIMx_MISSR register. Clear a bit in the TIMx_MISSR register by writing a 1 to it.

10.3 General-purpose timers 1 and 2 registers

10.3.1 Timer *x* interrupt and status register (TIMx_ISR)

Address offset: 0xA800 (TIM1) and 0xA804 (TIM2)
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			RSVD[3:0]				Reserved		TIF	Reserved	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			r	r	r	r			rw		rw	rw	rw	rw	rw

Bits 31:13] Reserved, must be kept at reset value

Bits 12:9] RSVD[3:0]: May change during normal operation

Bits 8:7] Reserved, must be kept at reset value

Bit 6 TIF: Trigger interrupt

Bit 5 Reserved, must be kept at reset value

Bit 4 CC4IF: Capture or compare 4 interrupt pending

Bit 3 CC3IF: Capture or compare 3 interrupt pending

Bit 2 CC2IF: Capture or compare 2 interrupt pending

Bit 1 CC1IF: Capture or compare 1 interrupt pending

Bit 0 UIF: Update interrupt pending

10.3.10 Timer *x* capture/compare enable register (TIMx_CCER)

Address offset: 0xE020 (TIM1) and 0xF020 (TIM2)
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		CC4P	CC4E	Reserved		CC3P	CC3E	Reserved		CC2P	CC2E	Reserved		CC1P	CC1E
		rw	rw			rw	rw			rw	rw			rw	rw

Bits 31:14 Reserved, must be kept at reset value

Bit 13 CC4P: Capture/Compare 4 output Polarity
If CC4 is configured as an output channel:
0: OC4 is active high
1: OC4 is active low.

If CC4 configured as an input channel:
0: IC4 is not inverted. Capture occurs on a rising edge of IC4. When used as an external trigger, IC4 is not inverted.
0: IC4 is inverted. Capture occurs on a falling edge of IC4. When used as an external trigger, IC4 is inverted.
1: Capture is enabled

Bit 12 CC4E: Capture/Compare 4 output Enable
If CC4 is configured as an output channel:
0: OC4 is disabled
1: OC4 is enabled

If CC4 configured as an input channel:
0: Capture is disabled
1: Capture is enabled

Bits 11:10 Reserved, must be kept at reset value

Bit 9 CC3P: Refer to the CC4P description above

Bit 8 CC3E: Refer to the CC4E description above

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

Bit 5 CC2P: Refer to the CC4P description above

Bit 4 CC2E: Refer to the CC4E description above

Bits 3:2 Reserved, must be kept at reset value

Bit 1 CC1P: Refer to the CC4P description above

Bit 0 CC1E: Refer to the CC4E description above

10.3.19 Timer 2 option register (TIM2_OR)

Address offset: 0xF050
Reset value: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								REMAPC4	REMAPC3	REMAPC2	REMAPC1	ORRSVD	CLKMSKEN	EXTRIGSEL[1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

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

Bit 7 REMAPC4
Selects the GPIO used for TIM2_CH4: 0 = PA2, 1 = PB4

Bit 6 REMAPC3
Selects the GPIO used for TIM2_CH3: 0 = PA1, 1 = PB3

Bit 5 REMAPC2
Selects the GPIO used for TIM2_CH2: 0 = PA3, 1 = PB2

Bit 4 REMAPC1
Selects the GPIO used for TIM2_CH1: 0 = PA0, 1 = PB1

Bit 3 ORRSVD
Reserved: this bit must always be set to 0

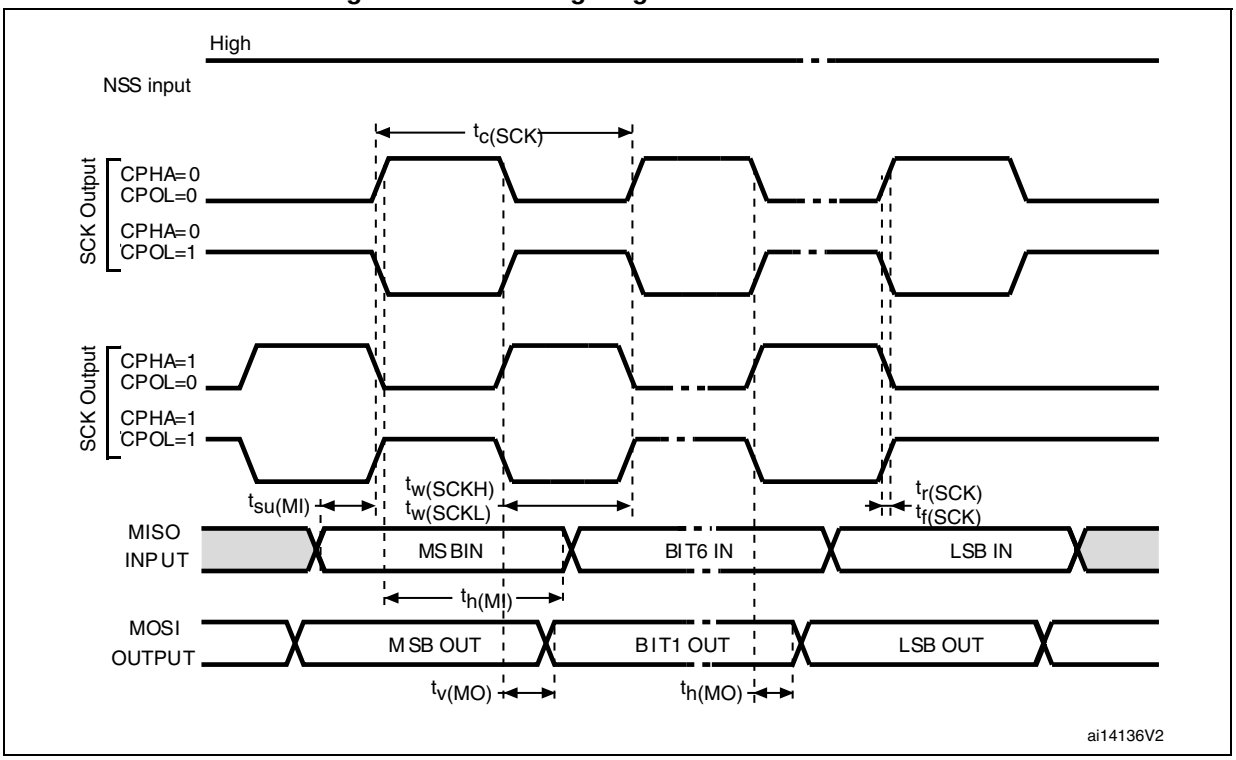
Bit 2 CLKMSKEN
Enables TIM2MSK when TIM2CLK is selected as the external trigger: 0 = TIM2MSK not used, 1 = TIM2CLK is ANDed with the TIM2MSK input.

Bits 1:0 EXTRIGSEL[1:0]:
Selects the external trigger used in external clock mode 2: 0 = PCLK, 1 = calibrated 1 kHz clock, 2 = 32 kHz reference clock (if available), 3 = TIM2CLK pin.

Table 43. NVIC exception table (continued)

Exception	Position	Description
Memory Fault	4	MPU mismatch, including access violation and no match. Synchronous.
Bus Fault	5	Pre-fetch, memory access, and other address/memory-related faults. Synchronous when precise and asynchronous when imprecise.
Usage Fault	6	Usage fault, such as 'undefined instruction executed' or 'illegal state transition attempt'. Synchronous.
-	7-10	Reserved, must be kept at reset value
SVCall	11	System service call with SVC instruction. Synchronous.
Debug Monitor	12	Debug monitor, when not halting. Synchronous, but only active when enabled. It does not activate if lower priority than the current activation.
-	13	Reserved, must be kept at reset value
PendSV	14	Pendable request for system service. Asynchronous and only pended by software.
SysTick	15	System tick timer has fired. Asynchronous.
Timer 1	16	Timer 1 peripheral interrupt.
Timer 2	17	Timer 2 peripheral interrupt.
Management	18	Management peripheral interrupt.
Baseband	19	Baseband peripheral interrupt.
Sleep Timer	20	Sleep Timer peripheral interrupt.
Serial Controller 1	21	Serial Controller 1 peripheral interrupt.
Serial Controller 2	22	Serial Controller 2 peripheral interrupt.
Security	23	Security peripheral interrupt.
MAC Timer	24	MAC Timer peripheral interrupt.
MAC Transmit	25	MAC Transmit peripheral interrupt.
MAC Receive	26	MAC Receive peripheral interrupt.
ADC	27	ADC peripheral interrupt.
IRQA	28	IRQA peripheral interrupt.
IRQB	29	IRQB peripheral interrupt.
IRQC	30	IRQC peripheral interrupt.
IRQD	31	IRQD peripheral interrupt.
Debug	32	Debug peripheral interrupt.

Figure 57. SPI timing diagram - master mode(1)



1. Measurement points are done at CMOS levels: $0.3V_{DD}$ and $0.7V_{DD}$.

14.9 Non-RF system electrical characteristics

Table 66 lists the non-RF system level characteristics for the STM32W.

Table 66. Non-RF system electrical characteristics

Parameter	Conditions	Min.	Typ.	Max.	Unit
System wakeup time from deep sleep	From wakeup event to first ARM® Cortex®-M3 instruction running from 6MHz internal RC clock Includes supply ramp time and oscillator startup time	–	110	–	µs
Shutdown time going into deep sleep	From last ARM® Cortex®-M3 instruction to deep sleep mode	–	5	–	µs

14.10 RF electrical characteristics

14.10.1 Receive

Table 67 lists the key parameters of the integrated IEEE 802.15.4 receiver on the STM32W.

Note: Receive measurements were collected with ST's STM32W Ceramic Balun Reference Design (Version A0) at 2440 MHz. The Typical number indicates one standard deviation above the mean, measured at room temperature (25° C). The Min and Max numbers were measured over process corners at room temperature

Table 67. Receive characteristics

Parameter	Conditions	Min.	Typ.	Max.	Unit
Frequency range	-	2400	–	2500	MHz
Sensitivity (boost mode)	1% PER, 20 byte packet defined by IEEE 802.15.4-2003	–	-102	-96	dBm
Sensitivity	1% PER, 20 byte packet defined by IEEE 802.15.4-2003	–	-100	-94	dBm
High-side adjacent channel rejection	IEEE 802.15.4 signal at -82 dBm	–	35	–	dB
Low-side adjacent channel rejection	IEEE 802.15.4 signal at -82 dBm	–	35	–	dB
2nd high-side adjacent channel rejection	IEEE 802.15.4 signal at -82 dBm	–	46	–	dB
2nd low-side adjacent channel rejection	IEEE 802.15.4 signal at -82 dBm	–	46	–	dB
Channel rejection for all other channels	IEEE 802.15.4 signal at -82 dBm	–	40	–	dB
802.11g rejection centered at +12 MHz or -13 MHz	IEEE 802.15.4 signal at -82 dBm	–	36	–	dB