

3.3 Memory map

See the datasheet corresponding to your device for a comprehensive diagram of the memory map. [Table 3](#) gives the boundary addresses of the peripherals available in all STM32F10xxx devices.

Table 3. Register boundary addresses

Boundary address	Peripheral	Bus	Register map
0xA000 0000 - 0xA000 0FFF	FSMC	AHB	Section 21.6.9 on page 564
0x5000 0000 - 0x5003 FFFF	USB OTG FS		Section 28.16.6 on page 913
0x4003 0000 - 0x4FFF FFFF	Reserved		-
0x4002 8000 - 0x4002 9FFF	Ethernet		Section 29.8.5 on page 1071
0x4002 3400 - 0x4002 7FFF	Reserved		-
0x4002 3000 - 0x4002 33FF	CRC		Section 4.4.4 on page 65
0x4002 2000 - 0x4002 23FF	Flash memory interface		-
0x4002 1400 - 0x4002 1FFF	Reserved		-
0x4002 1000 - 0x4002 13FF	Reset and clock control RCC		Section 7.3.11 on page 121
0x4002 0800 - 0x4002 0FFF	Reserved		-
0x4002 0400 - 0x4002 07FF	DMA2		Section 13.4.7 on page 289
0x4002 0000 - 0x4002 03FF	DMA1		-
0x4001 8400 - 0x4001 FFFF	Reserved		-
0x4001 8000 - 0x4001 83FF	SDIO		Section 22.9.16 on page 621

Table 3. Register boundary addresses (continued)

Boundary address	Peripheral	Bus	Register map
0x4001 5800 - 0x4001 7FFF	Reserved	APB2	-
0x4001 5400 - 0x4001 57FF	TIM11 timer		Section 16.5.11 on page 468
0x4001 5000 - 0x4001 53FF	TIM10 timer		Section 16.5.11 on page 468
0x4001 4C00 - 0x4001 4FFF	TIM9 timer		Section 16.4.13 on page 458
0x4001 4000 - 0x4001 4BFF	Reserved		-
0x4001 3C00 - 0x4001 3FFF	ADC3		Section 11.12.15 on page 252
0x4001 3800 - 0x4001 3BFF	USART1		Section 27.6.8 on page 827
0x4001 3400 - 0x4001 37FF	TIM8 timer		Section 14.4.21 on page 363
0x4001 3000 - 0x4001 33FF	SPI1		Section 25.5 on page 742
0x4001 2C00 - 0x4001 2FFF	TIM1 timer		Section 14.4.21 on page 363
0x4001 2800 - 0x4001 2BFF	ADC2		Section 11.12.15 on page 252
0x4001 2400 - 0x4001 27FF	ADC1		
0x4001 2000 - 0x4001 23FF	GPIO Port G		
0x4001 1C00 - 0x4001 1FFF	GPIO Port F		
0x4001 1800 - 0x4001 1BFF	GPIO Port E		
0x4001 1400 - 0x4001 17FF	GPIO Port D		Section 9.5 on page 194
0x4001 1000 - 0x4001 13FF	GPIO Port C		
0x4001 0C00 - 0x4001 0FFF	GPIO Port B		
0x4001 0800 - 0x4001 0BFF	GPIO Port A		
0x4001 0400 - 0x4001 07FF	EXTI		Section 10.3.7 on page 214
0x4001 0000 - 0x4001 03FF	AFIO		Section 9.5 on page 194

Table 3. Register boundary addresses (continued)

Boundary address	Peripheral	Bus	Register map
0x4000 7800 - 0x4000 FFFF	Reserved	APB1	-
0x4000 7400 - 0x4000 77FF	DAC		Section 12.5.14 on page 273
0x4000 7000 - 0x4000 73FF	Power control PWR		Section 5.4.3 on page 80
0x4000 6C00 - 0x4000 6FFF	Backup registers (BKP)		Section 6.4.5 on page 85
0x4000 6400 - 0x4000 67FF	bxCAN1		Section 24.9.5 on page 695
0x4000 6800 - 0x4000 6BFF	bxCAN2		-
0x4000 6000 ⁽¹⁾ - 0x4000 63FF	Shared USB/CAN SRAM 512 bytes		Section 23.5.4 on page 651
0x4000 5C00 - 0x4000 5FFF	USB device FS registers		Section 26.6.10 on page 784
0x4000 5800 - 0x4000 5BFF	I2C2		Section 27.6.8 on page 827
0x4000 5400 - 0x4000 57FF	I2C1		-
0x4000 5000 - 0x4000 53FF	UART5		Section 25.5 on page 742
0x4000 4C00 - 0x4000 4FFF	UART4		Section 25.5 on page 742
0x4000 4800 - 0x4000 4BFF	USART3		-
0x4000 4400 - 0x4000 47FF	USART2		Section 19.4.5 on page 499
0x4000 4000 - 0x4000 43FF	Reserved		Section 20.6.4 on page 506
0x4000 3C00 - 0x4000 3FFF	SPI3/I2S		Section 18.4.7 on page 493
0x4000 3800 - 0x4000 3BFF	SPI2/I2S		-
0x4000 3400 - 0x4000 37FF	Reserved		Section 16.5.11 on page 468
0x4000 3000 - 0x4000 33FF	Independent watchdog (IWDG)		Section 16.4.13 on page 458
0x4000 2C00 - 0x4000 2FFF	Window watchdog (WWDG)		Section 17.4.9 on page 481
0x4000 2800 - 0x4000 2BFF	RTC		Section 15.4.19 on page 423
0x4000 2400 - 0x4000 27FF	Reserved		
0x4000 2000 - 0x4000 23FF	TIM14 timer		
0x4000 1C00 - 0x4000 1FFF	TIM13 timer		
0x4000 1800 - 0x4000 1BFF	TIM12 timer		
0x4000 1400 - 0x4000 17FF	TIM7 timer		
0x4000 1000 - 0x4000 13FF	TIM6 timer		
0x4000 0C00 - 0x4000 0FFF	TIM5 timer		
0x4000 0800 - 0x4000 0BFF	TIM4 timer		
0x4000 0400 - 0x4000 07FF	TIM3 timer		
0x4000 0000 - 0x4000 03FF	TIM2 timer		

1. This shared SRAM can be fully accessed only in low-, medium-, high- and XL-density devices, not in connectivity line devices.

3.3.1 Embedded SRAM

The STM32F10xxx features up to 96 Kbytes of static SRAM. It can be accessed as bytes, half-words (16 bits) or full words (32 bits). The SRAM start address is 0x2000 0000.

3.3.2 Bit banding

The Cortex®-M3 memory map includes two bit-band regions. These regions map each word in an alias region of memory to a bit in a bit-band region of memory. Writing to a word in the alias region has the same effect as a read-modify-write operation on the targeted bit in the bit-band region.

In the STM32F10xxx both peripheral registers and SRAM are mapped in a bit-band region. This allows single bit-band write and read operations to be performed. The operations are only available for Cortex®-M3 accesses, not from other bus masters (e.g. DMA).

A mapping formula shows how to reference each word in the alias region to a corresponding bit in the bit-band region. The mapping formula is:

$$\text{bit_word_addr} = \text{bit_band_base} + (\text{byte_offset} \times 32) + (\text{bit_number} \times 4)$$

where:

bit_word_addr is the address of the word in the alias memory region that maps to the targeted bit.

bit_band_base is the starting address of the alias region

byte_offset is the number of the byte in the bit-band region that contains the targeted bit

bit_number is the bit position (0-7) of the targeted bit.

Example:

The following example shows how to map bit 2 of the byte located at SRAM address 0x20000300 in the alias region:

$$0x22006008 = 0x22000000 + (0x300*32) + (2*4).$$

Writing to address 0x22006008 has the same effect as a read-modify-write operation on bit 2 of the byte at SRAM address 0x20000300.

Reading address 0x22006008 returns the value (0x01 or 0x00) of bit 2 of the byte at SRAM address 0x20000300 (0x01: bit set; 0x00: bit reset).

For more information on Bit-Banding refer to the *Cortex®-M3 Technical Reference Manual*.

3.3.3 Embedded Flash memory

The high-performance Flash memory module has the following key features:

- For XL-density devices: density of up to 1 Mbyte with dual bank architecture for read-while-write (RWW) capability:
 - bank 1: fixed size of 512 Kbytes
 - bank 2: up to 512 Kbytes
- For other devices: density of up to 512 Kbytes
- Memory organization: the Flash memory is organized as a main block and an information block:
 - Main memory block of size:
 - up to 128 Kbytes × 64 bits divided into 512 pages of 2 Kbytes each (see [Table 8](#)) for XL-density devices
 - up to 4 Kb × 64 bits divided into 32 pages of 1 Kbyte each for low-density devices (see [Table 4](#))
 - up to 16 Kb × 64 bits divided into 128 pages of 1 Kbyte each for medium-density devices (see [Table 5](#))
 - up to 64 Kb × 64 bits divided into 256 pages of 2 Kbytes each (see [Table 6](#)) for high-density devices
 - up to 32 Kbit × 64 bits divided into 128 pages of 2 Kbytes each (see [Table 7](#)) for connectivity line devices
 - Information block of size:
 - 770 × 64 bits for XL-density devices (see [Table 8](#))
 - 2360 × 64 bits for connectivity line devices (see [Table 7](#))
 - 258 × 64 bits for other devices (see [Table 4](#), [Table 5](#) and [Table 6](#))

The Flash memory interface (FLITF) features:

- Read interface with prefetch buffer (2x64-bit words)
- Option byte Loader
- Flash Program / Erase operation
- Read / Write protection

Table 4. Flash module organization (low-density devices)

Block	Name	Base addresses	Size (bytes)
Main memory	Page 0	0x0800 0000 - 0x0800 03FF	1 K
	Page 1	0x0800 0400 - 0x0800 07FF	1 K
	Page 2	0x0800 0800 - 0x0800 0BFF	1 K
	Page 3	0x0800 0C00 - 0x0800 0FFF	1 K
	Page 4	0x0800 1000 - 0x0800 13FF	1 K
	.	.	.
	.	.	.
	Page 31	0x0800 7C00 - 0x0800 7FFF	1 K

Table 4. Flash module organization (low-density devices) (continued)

Block	Name	Base addresses	Size (bytes)
Information block	System memory	0x1FFF F000 - 0x1FFF F7FF	2 K
	Option bytes	0x1FFF F800 - 0x1FFF F80F	16
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WRPTR	0x4002 2020 - 0x4002 2023	4

Table 5. Flash module organization (medium-density devices)

Block	Name	Base addresses	Size (bytes)
Main memory	Page 0	0x0800 0000 - 0x0800 03FF	1 K
	Page 1	0x0800 0400 - 0x0800 07FF	1 K
	Page 2	0x0800 0800 - 0x0800 0BFF	1 K
	Page 3	0x0800 0C00 - 0x0800 0FFF	1 K
	Page 4	0x0800 1000 - 0x0800 13FF	1 K
	.	.	.
	.	.	.
	Page 127	0x0801 FC00 - 0x0801 FFFF	1 K
Information block	System memory	0x1FFF F000 - 0x1FFF F7FF	2 K
	Option bytes	0x1FFF F800 - 0x1FFF F80F	16
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WRPTR	0x4002 2020 - 0x4002 2023	4

Table 6. Flash module organization (high-density devices)

Block	Name	Base addresses	Size (bytes)
Main memory	Page 0	0x0800 0000 - 0x0800 07FF	2 K
	Page 1	0x0800 0800 - 0x0800 0FFF	2 K
	Page 2	0x0800 1000 - 0x0800 17FF	2 K
	Page 3	0x0800 1800 - 0x0800 1FFF	2 K
	.	.	.
	Page 255	0x0807 F800 - 0x0807 FFFF	2 K
Information block	System memory	0x1FFF F000 - 0x1FFF F7FF	2 K
	Option bytes	0x1FFF F800 - 0x1FFF F80F	16
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WRPR	0x4002 2020 - 0x4002 2023	4

Table 7. Flash module organization (connectivity line devices)

Block	Name	Base addresses	Size (bytes)
Main memory	Page 0	0x0800 0000 - 0x0800 07FF	2 K
	Page 1	0x0800 0800 - 0x0800 0FFF	2 K
	Page 2	0x0800 1000 - 0x0800 17FF	2 K
	Page 3	0x0800 1800 - 0x0800 1FFF	2 K
	.	.	.
	Page 127	0x0803 F800 - 0x0803 FFFF	2 K
Information block	System memory	0x1FFF B000 - 0x1FFF F7FF	18 K
	Option bytes	0x1FFF F800 - 0x1FFF F80F	16

Table 7. Flash module organization (connectivity line devices) (continued)

Block	Name	Base addresses	Size (bytes)
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WRPR	0x4002 2020 - 0x4002 2023	4

Table 8. XL-density Flash module organization

Block	Name	Base addresses	Size (bytes)		
Main memory	Bank 1	Page 0	0x0800 0000 - 0x0800 07FF	2 K	
		Page 1	0x0800 0800 - 0x0800 0FFF	2 K	
		
		Page 255	0x0807 F800 - 0x0807 FFFF	2 K	
	Bank 2	Page 256	0x0808 0000 - 0x0808 07FF	2 K	
		Page 257	0x0808 0800 - 0x0808 0FFF	2 K	
		
		Page 511	0x080F F800 - 0x080F FFFF	2 K	
Information block		System memory	0x1FFF E000 - 0x1FFF F7FF	6 K	
		Option bytes	0x1FFF F800 - 0x1FFF F80F	16	

Table 8. XL-density Flash module organization (continued)

Block	Name	Base addresses	Size (bytes)
Flash memory interface registers	FLASH_ACR	0x4002 2000 - 0x4002 2003	4
	FLASH_KEYR	0x4002 2004 - 0x4002 2007	4
	FLASH_OPTKEYR	0x4002 2008 - 0x4002 200B	4
	FLASH_SR	0x4002 200C - 0x4002 200F	4
	FLASH_CR	0x4002 2010 - 0x4002 2013	4
	FLASH_AR	0x4002 2014 - 0x4002 2017	4
	Reserved	0x4002 2018 - 0x4002 201B	4
	FLASH_OBR	0x4002 201C - 0x4002 201F	4
	FLASH_WPR	0x4002 2020 - 0x4002 2023	4
	Reserved	0x4002 2024 - 0x4002 2043	32
	FLASH_KEYR2	0x4002 2044 - 0x4002 2047	4
	Reserved	0x4002 2048 - 0x4002 204B	4
	FLASH_SR2	0x4002 204C - 0x4002 204F	4
	FLASH_CR2	0x4002 2050 - 0x4002 2053	4
	FLASH_AR2	0x4002 2054 - 0x4002 2057	4

Note: For further information on the Flash memory interface registers, refer to the “STM32F10xxx XL-density Flash programming manual” (PM0068) for XL-density devices, “STM32F10xxx Flash programming manual” (PM0075) for other devices.

Reading the Flash memory

Flash memory instructions and data access are performed through the AHB bus. The prefetch block is used for instruction fetches through the ICode bus. Arbitration is performed in the Flash memory interface, and priority is given to data access on the DCode bus.

Read accesses can be performed with the following configuration options:

- Latency: number of wait states for a read operation programmed on-the-fly
- Prefetch buffer (2 x 64-bit blocks): it is enabled after reset; a whole block can be replaced with a single read from the Flash memory as the size of the block matches the bandwidth of the Flash memory. Thanks to the prefetch buffer, faster CPU execution is possible as the CPU fetches one word at a time with the next word readily available in the prefetch buffer
- Half cycle: for power optimization

Note: These options have to be used in accordance with the Flash memory access time. The wait states represent the ratio of the SYSCLK (system clock) period to the Flash memory access time:

- 0 wait states, if $0 < \text{SYSCLK} \leq 24 \text{ MHz}$
- 1 wait state, if $24 \text{ MHz} < \text{SYSCLK} \leq 48 \text{ MHz}$
- 2 wait states, if $48 \text{ MHz} < \text{SYSCLK} \leq 72 \text{ MHz}$

Half cycle configuration is not available in combination with a prescaler on the AHB. The system clock (SYSCLK) should be equal to the HCLK clock. This feature can therefore be

used only with a low-frequency clock of 8 MHz or less. It can be generated from the HSI or the HSE but not from the PLL.

The prefetch buffer must be kept on when using a prescaler different from 1 on the AHB clock.

The prefetch buffer must be switched on/off only when SY SCLK is lower than 24 MHz and no prescaler is applied on the AHB clock (SY SCLK must be equal to HCLK). The prefetch buffer is usually switched on/off during the initialization routine, while the microcontroller is running on the internal 8 MHz RC (HSI) oscillator.

Using DMA: DMA accesses Flash memory on the DCode bus and has priority over ICode instructions. The DMA provides one free cycle after each transfer. Some instructions can be performed together with DMA transfer.

Programming and erasing the Flash memory

The Flash memory can be programmed 16 bits (half words) at a time.

For write and erase operations on the Flash memory (write/erase), the internal RC oscillator (HSI) must be ON.

The Flash memory erase operation can be performed at page level or on the whole Flash area (mass-erase). The mass-erase does not affect the information blocks.

To ensure that there is no over-programming, the Flash Programming and Erase Controller blocks are clocked by a fixed clock.

The End of write operation (programming or erasing) can trigger an interrupt. This interrupt can be used to exit from WFI mode, only if the FLITF clock is enabled. Otherwise, the interrupt is served only after an exit from WFI.

The FLASH_ACR register is used to enable/disable prefetch and half cycle access, and to control the Flash memory access time according to the CPU frequency. The tables below provide the bit map and bit descriptions for this register.

For complete information on Flash memory operations and register configurations, refer to the STM32F10xxx Flash programming manual (PM0075) or to the XL STM32F10xxx Flash programming manual (PM0068).

Flash access control register (FLASH_ACR)

Address offset: 0x00

Reset value: 0x0000 0030

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															
								PRFTBS	PRFTBE	HLFCYA	LATENCY				
								r	rw	rw	rw	rw	rw	rw	rw

Bits 31:6 Reserved, must be kept at reset value.

Bit 5 **PRFTBS**: Prefetch buffer status

This bit provides the status of the prefetch buffer.

- 0: Prefetch buffer is disabled
- 1: Prefetch buffer is enabled

Bit 4 **PRFTBE**: Prefetch buffer enable

- 0: Prefetch is disabled
- 1: Prefetch is enabled

Bit 3 **HLFCYA**: Flash half cycle access enable

- 0: Half cycle is disabled
- 1: Half cycle is enabled

Bits 2:0 **LATENCY**: Latency

These bits represent the ratio of the SYSCLK (system clock) period to the Flash access time.

000 Zero wait state, if $0 < \text{SYSCLK} \leq 24 \text{ MHz}$

001 One wait state, if $24 \text{ MHz} < \text{SYSCLK} \leq 48 \text{ MHz}$

010 Two wait states, if $48 \text{ MHz} < \text{SYSCLK} \leq 72 \text{ MHz}$

3.4 Boot configuration

In the STM32F10xxx, 3 different boot modes can be selected through BOOT[1:0] pins as shown in [Table 9](#).

Table 9. Boot modes

Boot mode selection pins		Boot mode	Aliasing
BOOT1	BOOT0		
x	0	Main Flash memory	Main Flash memory is selected as boot space
0	1	System memory	System memory is selected as boot space
1	1	Embedded SRAM	Embedded SRAM is selected as boot space

The values on the BOOT pins are latched on the 4th rising edge of SYSCLK after a reset. It is up to the user to set the BOOT1 and BOOT0 pins after Reset to select the required boot mode.

The BOOT pins are also re-sampled when exiting from Standby mode. Consequently they must be kept in the required Boot mode configuration in Standby mode. After this startup delay has elapsed, the CPU fetches the top-of-stack value from address 0x0000 0000, then starts code execution from the boot memory starting from 0x0000 0004.

Due to its fixed memory map, the code area starts from address 0x0000 0000 (accessed through the ICode/DCode buses) while the data area (SRAM) starts from address 0x2000 0000 (accessed through the system bus). The Cortex®-M3 CPU always fetches the reset vector on the ICode bus, which implies to have the boot space available only in the code area (typically, Flash memory). STM32F10xxx microcontrollers implement a special mechanism to be able to boot also from SRAM and not only from main Flash memory and System memory.

Depending on the selected boot mode, main Flash memory, system memory or SRAM is accessible as follows:

- Boot from main Flash memory: the main Flash memory is aliased in the boot memory space (0x0000 0000), but still accessible from its original memory space (0x800 0000). In other words, the Flash memory contents can be accessed starting from address 0x0000 0000 or 0x800 0000.
- Boot from system memory: the system memory is aliased in the boot memory space (0x0000 0000), but still accessible from its original memory space (0x1FFF B000 in connectivity line devices, 0x1FFF F000 in other devices).
- Boot from the embedded SRAM: SRAM is accessible only at address 0x2000 0000.

Note:

When booting from SRAM, in the application initialization code, you have to relocate the vector table in SRAM using the NVIC exception table and offset register.

For XL-density devices, when booting from the main Flash memory, you have an option to boot from any of two memory banks. By default, boot from Flash memory bank 1 is selected. You can choose to boot from Flash memory bank 2 by clearing the BFB2 bit in the user option bytes. When this bit is cleared and the boot pins are in the boot from main Flash memory configuration, the device boots from system memory, and the boot loader jumps to execute the user application programmed in Flash memory bank 2. For further details refer to AN2606.

Note:

When booting from Bank2 in the applications initialization code, relocate the vector table to the Bank2 base address. (0x0808 0000) using the NVIC exception table and offset register.

Embedded boot loader

The embedded boot loader is located in the System memory, programmed by ST during production. It is used to reprogram the Flash memory with one of the available serial interfaces:

- In low-, medium- and high-density devices the bootloader is activated through the USART1 interface.
- In XL-density devices the boot loader is activated through the following interfaces: USART1 or USART2 (remapped).
- In connectivity line devices the boot loader can be activated through one of the following interfaces: USART1, USART2 (remapped), CAN2 (remapped) or USB OTG FS in Device mode (DFU: device firmware upgrade).

The USART peripheral operates with the internal 8 MHz oscillator (HSI). The CAN and USB OTG FS, however, can only function if an external 8 MHz, 14.7456 MHz or 25 MHz clock (HSE) is present.

Note: *For further details refer to AN2606.*

4 CRC calculation unit

Low-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 16 and 32 Kbytes.

Medium-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 64 and 128 Kbytes.

High-density devices are *STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 256 and 512 Kbytes*.

XL-density devices are *STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 768 Kbytes and 1 Mbyte*.

Connectivity line devices are STM32F105xx and STM32F107xx microcontrollers.

4.1 CRC introduction

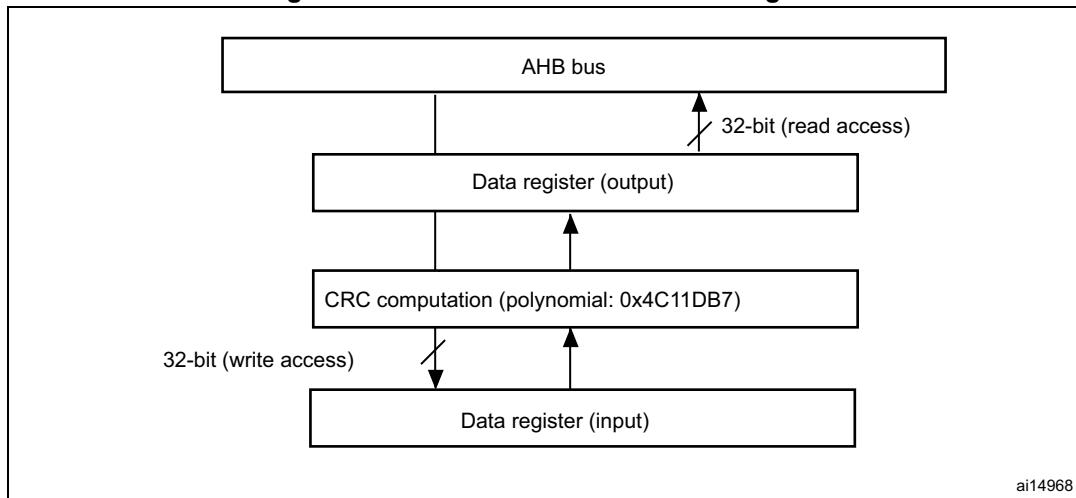
The CRC (cyclic redundancy check) calculation unit is used to get a CRC code from a 32-bit data word and a fixed generator polynomial.

Among other applications, CRC-based techniques are used to verify data transmission or storage integrity. In the scope of the EN/IEC 60335-1 standard, they offer a means of verifying the Flash memory integrity. The CRC calculation unit helps compute a signature of the software during runtime, to be compared with a reference signature generated at link-time and stored at a given memory location.

4.2 CRC main features

- Uses CRC-32 (Ethernet) polynomial: 0x4C11DB7
 - $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- Single input/output 32-bit data register
- CRC computation done in 4 AHB clock cycles (HCLK)
- General-purpose 8-bit register (can be used for temporary storage)

The block diagram is shown in [Figure 3](#).

Figure 3. CRC calculation unit block diagram

4.3 CRC functional description

The CRC calculation unit mainly consists of a single 32-bit data register, which:

- is used as an input register to enter new data in the CRC calculator (when writing into the register)
- holds the result of the previous CRC calculation (when reading the register)

Each write operation into the data register creates a combination of the previous CRC value and the new one (CRC computation is done on the whole 32-bit data word, and not byte per byte).

The write operation is stalled until the end of the CRC computation, thus allowing back-to-back write accesses or consecutive write and read accesses.

The CRC calculator can be reset to 0xFFFF FFFF with the RESET control bit in the CRC_CR register. This operation does not affect the contents of the CRC_IDR register.

4.4 CRC registers

The CRC calculation unit contains two data registers and a control register. The peripheral The CRC registers have to be accessed by words (32 bits).

4.4.1 Data register (CRC_DR)

Address offset: 0x00

Reset value: 0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DR [31:16]															
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
DR [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 31:0 Data register bits

Used as an input register when writing new data into the CRC calculator.
Holds the previous CRC calculation result when it is read.

4.4.2 Independent data register (CRC_IDR)

Address offset: 0x04

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								IDR[7:0]							
		rw		rw		rw		rw		rw		rw		rw	

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

Bits 7:0 General-purpose 8-bit data register bits

Can be used as a temporary storage location for one byte.

This register is not affected by CRC resets generated by the RESET bit in the CRC_CR register.

4.4.3 Control register (CRC_CR)

Address offset: 0x08

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														RESET		
																w

Bits 31:1 Reserved, must be kept at reset value.

Bit 0 RESET bit

Resets the CRC calculation unit and sets the data register to 0xFFFF FFFF.
This bit can only be set, it is automatically cleared by hardware.

4.4.4 CRC register map

The following table provides the CRC register map and reset values.

Table 10. CRC calculation unit register map and reset values

Offset	Register	31-24	23-16	15-8	7	6	5	4	3	2	1	0	
0x00	CRC_DR	Data register											
	Reset value	0xFFFF FFFF											

Table 10. CRC calculation unit register map and reset values (continued)

Offset	Register	31-24	23-16	15-8	7	6	5	4	3	2	1	0		
0x04	CRC_IDR	Reserved				Independent data register								
	Reset value					0x00								
0x08	CRC_CR	Reserved									RESET	0		
	Reset value													

5 Power control (PWR)

Low-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 16 and 32 Kbytes.

Medium-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 64 and 128 Kbytes.

High-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 256 and 512 Kbytes.

XL-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 768 Kbytes and 1 Mbyte.

Connectivity line devices are STM32F105xx and STM32F107xx microcontrollers.

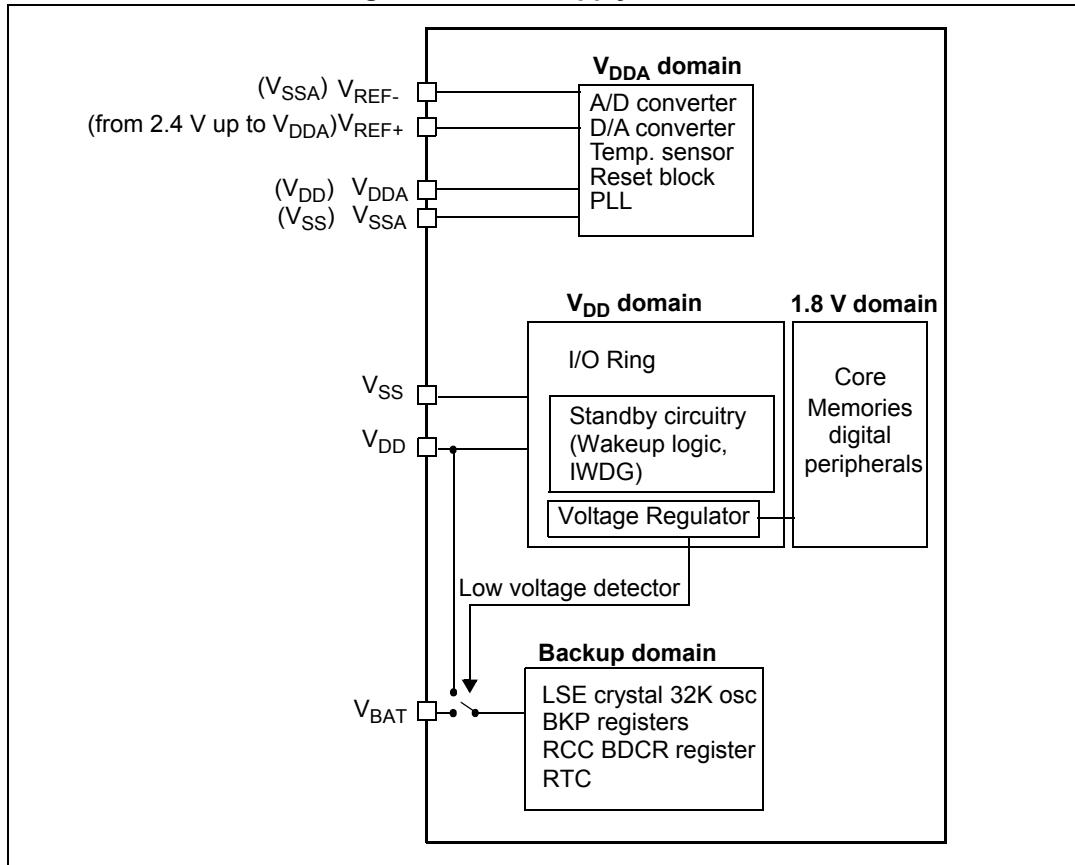
This section applies to the whole STM32F101xx family, unless otherwise specified.

5.1 Power supplies

The device requires a 2.0-to-3.6 V operating voltage supply (V_{DD}). An embedded regulator is used to supply the internal 1.8 V digital power.

The real-time clock (RTC) and backup registers can be powered from the V_{BAT} voltage when the main V_{DD} supply is powered off.

Figure 4. Power supply overview



1. V_{DDA} and V_{SSA} must be connected to V_{DD} and V_{SS} , respectively.

5.1.1 Independent A/D and D/A converter supply and reference voltage

To improve conversion accuracy, the ADC and the DAC have an independent power supply which can be separately filtered and shielded from noise on the PCB.

- The ADC and DAC voltage supply input is available on a separate V_{DDA} pin.
 - An isolated supply ground connection is provided on pin V_{SSA} .

When available (according to package), V_{REF-} must be tied to V_{SSA} .

On 100-pin and 144-pin packages

To ensure a better accuracy on low-voltage inputs and outputs, the user can connect a separate external reference voltage on V_{REF+} . V_{REF+} is the highest voltage, represented by the full scale value, for an analog input (ADC) or output (DAC) signal. The voltage on V_{REF+} can range from 2.4 V to V_{DDA} :

On 64-pin packages and packages with less pins

The V_{REF+} and V_{REF-} pins are not available, they are internally connected to the ADC voltage supply (V_{DDA}) and ground (V_{SSA}).

5.1.2 Battery backup domain

To retain the content of the Backup registers and supply the RTC function when V_{DD} is turned off, V_{BAT} pin can be connected to an optional standby voltage supplied by a battery or by another source.

The V_{BAT} pin powers the RTC unit, the LSE oscillator and the PC13 to PC15 IOs, allowing the RTC to operate even when the main digital supply (V_{DD}) is turned off. The switch to the V_{BAT} supply is controlled by the Power Down Reset embedded in the Reset block.

Warning: During $t_{RSTTEMPO}$ (temporization at V_{DD} startup) or after a PDR is detected, the power switch between V_{BAT} and V_{DD} remains connected to V_{BAT} .
 During the startup phase, if V_{DD} is established in less than $t_{RSTTEMPO}$ (Refer to the datasheet for the value of $t_{RSTTEMPO}$) and $V_{DD} > V_{BAT} + 0.6$ V, a current may be injected into V_{BAT} through an internal diode connected between V_{DD} and the power switch (V_{BAT}).
If the power supply/battery connected to the V_{BAT} pin cannot support this current injection, it is strongly recommended to connect an external low-drop diode between this power supply and the V_{BAT} pin.

If no external battery is used in the application, it is recommended to connect V_{BAT} externally to V_{DD} with a 100 nF external ceramic decoupling capacitor (for more details refer to AN2586).

When the backup domain is supplied by V_{DD} (analog switch connected to V_{DD}), the following functions are available:

- PC14 and PC15 can be used as either GPIO or LSE pins
- PC13 can be used as GPIO, TAMPER pin, RTC Calibration Clock, RTC Alarm or second output (refer to [Section 6: Backup registers \(BKP\)](#))

Note: Due to the fact that the switch only sinks a limited amount of current (3 mA), the use of GPIOs PC13 to PC15 in output mode is restricted: the speed has to be limited to 2 MHz with a maximum load of 30 pF and these IOs must not be used as a current source (e.g. to drive a LED).

When the backup domain is supplied by V_{BAT} (analog switch connected to V_{BAT} because V_{DD} is not present), the following functions are available:

- PC14 and PC15 can be used as LSE pins only
- PC13 can be used as TAMPER pin, RTC Alarm or Second output (refer to [Section 6.4.2: RTC clock calibration register \(BKP_RTCCR\)](#)).

5.1.3 Voltage regulator

The voltage regulator is always enabled after Reset. It works in three different modes depending on the application modes.

- In Run mode, the regulator supplies full power to the 1.8 V domain (core, memories and digital peripherals).
- In Stop mode the regulator supplies low-power to the 1.8 V domain, preserving contents of registers and SRAM
- In Standby Mode, the regulator is powered off. The contents of the registers and SRAM are lost except for the Standby circuitry and the Backup Domain.

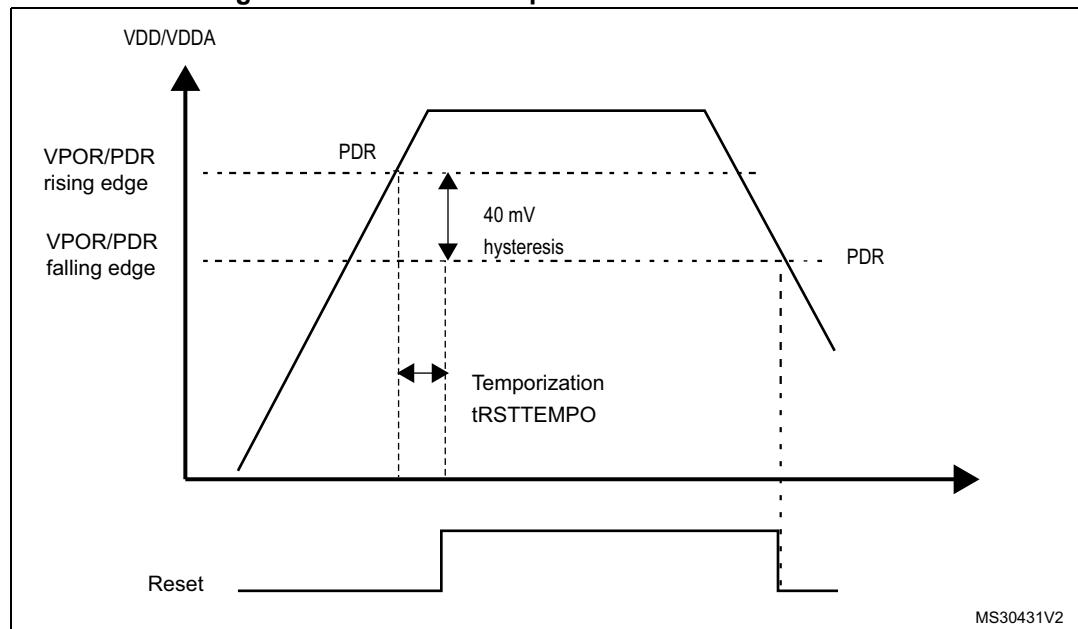
5.2 Power supply supervisor

5.2.1 Power on reset (POR)/power down reset (PDR)

The device has an integrated POR/PDR circuitry that allows proper operation starting from/down to 2 V.

The device remains in Reset mode when V_{DD}/V_{DDA} is below a specified threshold, $V_{POR/PDR}$, without the need for an external reset circuit. For more details concerning the power on/power down reset threshold, refer to the electrical characteristics of the datasheet.

Figure 5. Power on reset/power down reset waveform



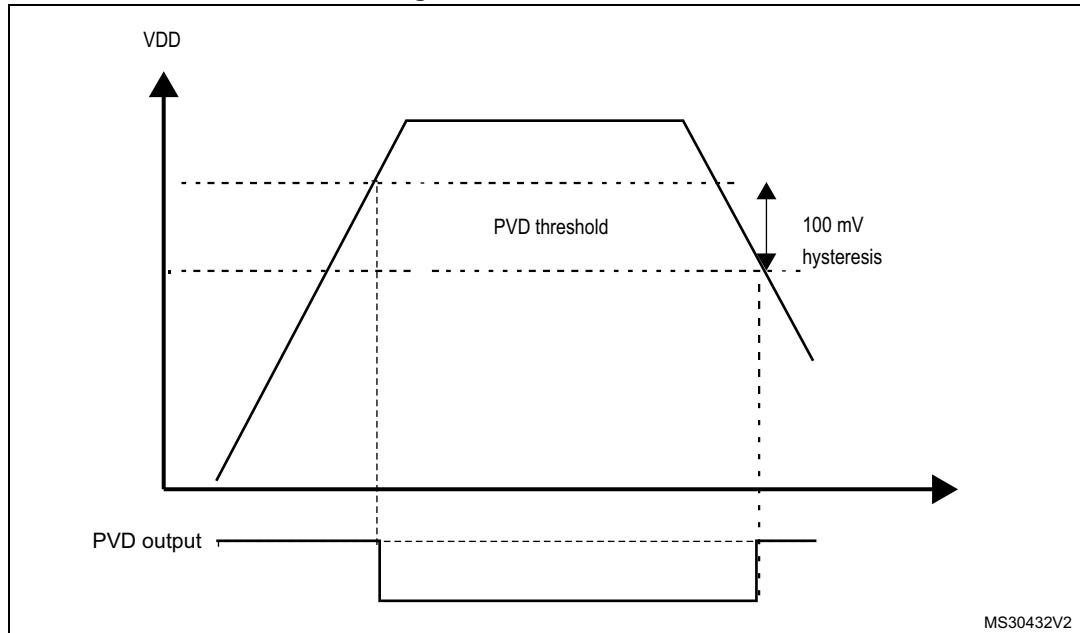
5.2.2 Programmable voltage detector (PVD)

The PVD can be used to monitor the V_{DD}/V_{DDA} power supply by comparing it to a threshold selected by the PLS[2:0] bits in the [Power control register \(PWR_CR\)](#).

The PVD is enabled by setting the PVDE bit.

A PVDO flag is available, in the *Power control/status register (PWR_CSR)*, to indicate if V_{DD}/V_{DDA} is higher or lower than the PVD threshold. This event is internally connected to the EXTI line16 and can generate an interrupt if enabled through the EXTI registers. The PVD output interrupt can be generated when V_{DD}/V_{DDA} drops below the PVD threshold and/or when V_{DD}/V_{DDA} rises above the PVD threshold depending on EXTI line16 rising/falling edge configuration. As an example the service routine could perform emergency shutdown tasks.

Figure 6. PVD thresholds



5.3 Low-power modes

By default, the microcontroller is in Run mode after a system or a power Reset. Several low-power modes are available to save power when the CPU does not need to be kept running, for example when waiting for an external event. It is up to the user to select the mode that gives the best compromise between low-power consumption, short startup time and available wakeup sources.

The STM32F10xxx devices feature three low-power modes:

- Sleep mode (CPU clock off, all peripherals including Cortex®-M3 core peripherals like NVIC, SysTick, etc. are kept running)
- Stop mode (all clocks are stopped)
- Standby mode (1.8V domain powered-off)

In addition, the power consumption in Run mode can be reduced by one of the following means:

- Slowing down the system clocks
- Gating the clocks to the APB and AHB peripherals when they are unused.

Table 11. Low-power mode summary

Mode name	Entry	Wakeup	Effect on 1.8V domain clocks	Effect on V _{DD} domain clocks	Voltage regulator
Sleep (Sleep now or Sleep-on -exit)	WFI	Any interrupt	CPU clock OFF no effect on other clocks or analog clock sources	None	ON
	WFE	Wakeup event			
Stop	PDDS and LPDS bits + SLEEPDEEP bit + WFI or WFE	Any EXTI line (configured in the EXTI registers)	All 1.8V domain clocks OFF	HSI and HSE oscillators OFF	ON or in low-power mode (depends on <i>Power control register (PWR_CR)</i>)
Standby	PDDS bit + SLEEPDEEP bit + WFI or WFE	WKUP pin rising edge, RTC alarm, external reset in NRST pin, IWDG reset			OFF

5.3.1 Slowing down system clocks

In Run mode the speed of the system clocks (SYSCLK, HCLK, PCLK1, PCLK2) can be reduced by programming the prescaler registers. These prescalers can also be used to slow down peripherals before entering Sleep mode.

For more details refer to [Section 7.3.2: Clock configuration register \(RCC_CFGR\)](#).

5.3.2 Peripheral clock gating

In Run mode, the HCLK and PCLKx for individual peripherals and memories can be stopped at any time to reduce power consumption.

To further reduce power consumption in Sleep mode the peripheral clocks can be disabled prior to executing the WFI or WFE instructions.

Peripheral clock gating is controlled by the *AHB peripheral clock enable register (RCC_AHBENR)*, *APB1 peripheral clock enable register (RCC_APB1ENR)* and *APB2 peripheral clock enable register (RCC_APB2ENR)*.

5.3.3 Sleep mode

Entering Sleep mode

The Sleep mode is entered by executing the WFI (Wait For Interrupt) or WFE (Wait for Event) instructions. Two options are available to select the Sleep mode entry mechanism, depending on the SLEEPONEXIT bit in the Cortex®-M3 System Control register:

- Sleep-now: if the SLEEPONEXIT bit is cleared, the MCU enters Sleep mode as soon as WFI or WFE instruction is executed.
- Sleep-on-exit: if the SLEEPONEXIT bit is set, the MCU enters Sleep mode as soon as it exits the lowest priority ISR.

In the Sleep mode, all I/O pins keep the same state as in the Run mode.

Refer to [Table 12](#) and [Table 13](#) for details on how to enter Sleep mode.

Exiting Sleep mode

If the WFI instruction is used to enter Sleep mode, any peripheral interrupt acknowledged by the nested vectored interrupt controller (NVIC) can wake up the device from Sleep mode.

If the WFE instruction is used to enter Sleep mode, the MCU exits Sleep mode as soon as an event occurs. The wakeup event can be generated either by:

- enabling an interrupt in the peripheral control register but not in the NVIC, and enabling the SEVONPEND bit in the Cortex®-M3 System Control register. When the MCU resumes from WFE, the peripheral interrupt pending bit and the peripheral NVIC IRQ channel pending bit (in the NVIC interrupt clear pending register) have to be cleared.
- or configuring an external or internal EXTI line in event mode. When the CPU resumes from WFE, it is not necessary to clear the peripheral interrupt pending bit or the NVIC IRQ channel pending bit as the pending bit corresponding to the event line is not set.

This mode offers the lowest wakeup time as no time is wasted in interrupt entry/exit.

Refer to [Table 12](#) and [Table 13](#) for more details on how to exit Sleep mode.

Table 12. Sleep-now

Sleep-now mode	Description
Mode entry	WFI (Wait for Interrupt) or WFE (Wait for Event) while: – SLEEPDEEP = 0 and – SLEEPONEXIT = 0 Refer to the Cortex®-M3 System Control register.
Mode exit	If WFI was used for entry: Interrupt: Refer to Section 10.1.2: Interrupt and exception vectors If WFE was used for entry Wakeup event: Refer to Section 10.2.3: Wakeup event management
Wakeup latency	None

Table 13. Sleep-on-exit

Sleep-on-exit	Description
Mode entry	WFI (wait for interrupt) while: – SLEEPDEEP = 0 and – SLEEPONEXIT = 1 Refer to the Cortex®-M3 System Control register.
Mode exit	Interrupt: refer to Section 10.1.2: Interrupt and exception vectors .
Wakeup latency	None

5.3.4 Stop mode

The Stop mode is based on the Cortex®-M3 deepsleep mode combined with peripheral clock gating. The voltage regulator can be configured either in normal or low-power mode. In Stop mode, all clocks in the 1.8 V domain are stopped, the PLL, the HSI and the HSE RC oscillators are disabled. SRAM and register contents are preserved.

In the Stop mode, all I/O pins keep the same state as in the Run mode.

Entering Stop mode

Refer to [Table 14](#) for details on how to enter the Stop mode.

To further reduce power consumption in Stop mode, the internal voltage regulator can be put in low-power mode. This is configured by the LPDS bit of the [Power control register \(PWR_CR\)](#).

If Flash memory programming is ongoing, the Stop mode entry is delayed until the memory access is finished.

If an access to the APB domain is ongoing, The Stop mode entry is delayed until the APB access is finished.

In Stop mode, the following features can be selected by programming individual control bits:

- Independent watchdog (IWDG): the IWDG is started by writing to its Key register or by hardware option. Once started it cannot be stopped except by a Reset. See [Section 19.3: IWDG functional description](#).
- Real-time clock (RTC): this is configured by the RTCEN bit in the [Backup domain control register \(RCC_BDCR\)](#)
- Internal RC oscillator (LSI RC): this is configured by the LSION bit in the [Control/status register \(RCC_CSR\)](#).
- External 32.768 kHz oscillator (LSE OSC): this is configured by the LSEON bit in the [Backup domain control register \(RCC_BDCR\)](#).

The ADC or DAC can also consume power during the Stop mode, unless they are disabled before entering it. To disable them, the ADON bit in the ADC_CR2 register and the ENx bit in the DAC_CR register must both be written to 0.

Note: *If the application needs to disable the external clock before entering Stop mode, the HSEON bit must first be disabled and the system clock switched to HSI. Otherwise, if the HSEON bit remains enabled and the external clock (external oscillator) is removed when entering Stop mode, the clock security system (CSS) feature must be enabled to detect any external oscillator failure and avoid a malfunction behavior when entering stop mode.*

Exiting Stop mode

Refer to [Table 14](#) for more details on how to exit Stop mode.

When exiting Stop mode by issuing an interrupt or a wakeup event, the HSI RC oscillator is selected as system clock.

When the voltage regulator operates in low-power mode, an additional startup delay is incurred when waking up from Stop mode. By keeping the internal regulator ON during Stop mode, the consumption is higher although the startup time is reduced.

Table 14. Stop mode

Stop mode	Description
Mode entry	<p>WFI (Wait for Interrupt) or WFE (Wait for Event) while:</p> <ul style="list-style-type: none"> – Set SLEEPDEEP bit in Cortex®-M3 System Control register – Clear PDDS bit in Power Control register (PWR_CR) – Select the voltage regulator mode by configuring LPDS bit in PWR_CR <p>Note: To enter Stop mode, all EXTI Line pending bits (in Pending register (EXTI_PR)), all peripheral interrupt pending bits, and RTC Alarm flag must be reset. Otherwise, the Stop mode entry procedure is ignored and program execution continues.</p>
Mode exit	<p>If WFI was used for entry:</p> <p>Any EXTI Line configured in Interrupt mode (the corresponding EXTI Interrupt vector must be enabled in the NVIC). Refer to Section 10.1.2: Interrupt and exception vectors.</p> <p>If WFE was used for entry:</p> <p>Any EXTI Line configured in event mode. Refer to Section 10.2.3: Wakeup event management</p>
Wakeup latency	HSI RC wakeup time + regulator wakeup time from Low-power mode

5.3.5 Standby mode

The Standby mode allows to achieve the lowest power consumption. It is based on the Cortex®-M3 deepsleep mode, with the voltage regulator disabled. The 1.8 V domain is consequently powered off. The PLL, the HSI oscillator and the HSE oscillator are also switched off. SRAM and register contents are lost except for registers in the Backup domain and Standby circuitry (see [Figure 4](#)).

Entering Standby mode

Refer to [Table 15](#) for more details on how to enter Standby mode.

In Standby mode, the following features can be selected by programming individual control bits:

- Independent watchdog (IWDG): the IWDG is started by writing to its Key register or by hardware option. Once started it cannot be stopped except by a reset. See [Section 19.3: IWDG functional description](#).
- Real-time clock (RTC): this is configured by the RTCEN bit in the Backup domain control register (RCC_BDCR)
- Internal RC oscillator (LSI RC): this is configured by the LSION bit in the Control/status register (RCC_CSR).
- External 32.768 kHz oscillator (LSE OSC): this is configured by the LSEON bit in the Backup domain control register (RCC_BDCR)

Exiting Standby mode

The microcontroller exits the Standby mode when an external reset (NRST pin), an IWDG reset, a rising edge on the WKUP pin or the rising edge of an RTC alarm occurs (see [Figure 179: RTC simplified block diagram](#)). All registers are reset after wakeup from Standby except for [Power control/status register \(PWR_CSR\)](#).

After waking up from Standby mode, program execution restarts in the same way as after a Reset (boot pins sampling, vector reset is fetched, etc.). The SBF status flag in the [Power control/status register \(PWR_CSR\)](#) indicates that the MCU was in Standby mode.

Refer to [Table 15](#) for more details on how to exit Standby mode.

Table 15. Standby mode

Standby mode	Description
Mode entry	WFI (Wait for Interrupt) or WFE (Wait for Event) while: – Set SLEEPDEEP in Cortex®-M3 System Control register – Set PDDS bit in Power Control register (PWR_CR) – Clear WUF bit in Power Control/Status register (PWR_CSR) – No interrupt (for WFI) or event (for WFI) is pending
Mode exit	WKUP pin rising edge, RTC alarm event's rising edge, external Reset in NRST pin, IWDG Reset.
Wakeup latency	Reset phase

I/O states in Standby mode

In Standby mode, all I/O pins are high impedance except:

- Reset pad (still available)
- TAMPER pin if configured for tamper or calibration out
- WKUP pin, if enabled

Debug mode

By default, the debug connection is lost if the application puts the MCU in Stop or Standby mode while the debug features are used. This is due to the fact that the Cortex®-M3 core is no longer clocked.

However, by setting some configuration bits in the DBGMCU_CR register, the software can be debugged even when using the low-power modes extensively. For more details, refer to [Section 31.16.1: Debug support for low-power modes](#).

5.3.6

Auto-wakeup (AWU) from low-power mode

The RTC can be used to wakeup the MCU from low-power mode without depending on an external interrupt (Auto-wakeup mode). The RTC provides a programmable time base for waking up from Stop or Standby mode at regular intervals. For this purpose, two of the three alternative RTC clock sources can be selected by programming the RTCSEL[1:0] bits in the [Backup domain control register \(RCC_BDCR\)](#):

- Low-power 32.768 kHz external crystal oscillator (LSE OSC).
This clock source provides a precise time base with very low-power consumption (less than 1µA added consumption in typical conditions)
- Low-power internal RC Oscillator (LSI RC)
This clock source has the advantage of saving the cost of the 32.768 kHz crystal. This internal RC Oscillator is designed to add minimum power consumption.

To wakeup from Stop mode with an RTC alarm event, it is necessary to:

- Configure the EXTI Line 17 to be sensitive to rising edge
- Configure the RTC to generate the RTC alarm

To wakeup from Standby mode, there is no need to configure the EXTI Line 17.

5.4

Power control registers

The peripheral registers can be accessed by half-words (16-bit) or words (32-bit).

5.4.1

Power control register (PWR_CR)

Address offset: 0x000

Reset value: 0x0000 0000 (reset by wakeup from Standby mode)

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							DBP	PLS[2:0]			PVDE	CSBF	CWUF	PDDS	LPDS
							rw	rw	rw	rw	rw	rc_w1	rc_w1	rw	rw

Bits 31:9 Reserved, must be kept at reset value..

Bit 8 **DBP**: Disable backup domain write protection.

In reset state, the RTC and backup registers are protected against parasitic write access.

This bit must be set to enable write access to these registers.

0: Access to RTC and Backup registers disabled

1: Access to RTC and Backup registers enabled

Note: If the HSE divided by 128 is used as the RTC clock, this bit must remain set to 1.

Bits 7:5 **PLS[2:0]**: PVD level selection.

These bits are written by software to select the voltage threshold detected by the programmable voltage detector

000: 2.2V

001: 2.3V

010: 2.4V

011: 2.5V

100: 2.6V

101: 2.7V

110: 2.8V

111: 2.9V

Note: Refer to the electrical characteristics of the datasheet for more details.

Bit 4 **PVDE**: programmable voltage detector enable.

This bit is set and cleared by software.

0: PVD disabled

1: PVD enabled

Bit 3 **CSBF**: Clear standby flag.

This bit is always read as 0.

0: No effect

1: Clear the SBF Standby Flag (write).

Bit 2 **CWUF**: Clear wakeup flag.

This bit is always read as 0.

0: No effect

1: Clear the WUF Wakeup Flag **after 2 System clock cycles**. (write)

Bit 1 **PDSS**: Power down deepsleep.

This bit is set and cleared by software. It works together with the LPDS bit.

0: Enter Stop mode when the CPU enters Deepsleep. The regulator status depends on the LPDS bit.

1: Enter Standby mode when the CPU enters Deepsleep.

Bit 0 **LPDS**: Low-power deepsleep.

This bit is set and cleared by software. It works together with the PDSS bit.

0: Voltage regulator on during Stop mode

1: Voltage regulator in low-power mode during Stop mode

5.4.2 Power control/status register (PWR_CSR)

Address offset: 0x04

Reset value: 0x0000 0000 (not reset by wakeup from Standby mode)

Additional APB cycles are needed to read this register versus a standard APB read.

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								EWUP	Reserved					PVDO	SBF	WUF
								rw						r	r	r

Bits 31:9 Reserved, must be kept at reset value.

Bit 8 **EWUP**: Enable WKUP pin

This bit is set and cleared by software.

0: WKUP pin is used for general purpose I/O. An event on the WKUP pin does not wakeup the device from Standby mode.

1: WKUP pin is used for wakeup from Standby mode and forced in input pull down configuration (rising edge on WKUP pin wakes-up the system from Standby mode).

Note: This bit is reset by a system Reset.

Bits 7:3 Reserved, must be kept at reset value.

Bit 2 **PVDO**: PVD output

This bit is set and cleared by hardware. It is valid only if PVD is enabled by the PVDE bit.

0: V_{DD}/V_{DDA} is higher than the PVD threshold selected with the PLS[2:0] bits.

1: V_{DD}/V_{DDA} is lower than the PVD threshold selected with the PLS[2:0] bits.

Note: The PVD is stopped by Standby mode. For this reason, this bit is equal to 0 after Standby or reset until the PVDE bit is set.

Bit 1 **SBF**: Standby flag

This bit is set by hardware and cleared only by a POR/PDR (power on reset/power down reset) or by setting the CSBF bit in the [Power control register \(PWR_CR\)](#)

0: Device has not been in Standby mode

1: Device has been in Standby mode

Bit 0 **WUF**: Wakeup flag

This bit is set by hardware and cleared by hardware, by a system reset or by setting the CWUF bit in the [Power control register \(PWR_CR\)](#)

0: No wakeup event occurred

1: A wakeup event was received from the WKUP pin or from the RTC alarm

Note: An additional wakeup event is detected if the WKUP pin is enabled (by setting the EWUP bit) when the WKUP pin level is already high.

5.4.3 PWR register map

The following table summarizes the PWR registers.

Table 16. PWR 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
0x000	PWR_CR	Reserved																		DBP	PLS [2:0]		PVDE	CSBF	CWUF	0	0	0	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				
0x004	PWR_CSR	Reserved																		C_EWUP	Reserved		PVDO	SBF	WUF	0	0	0	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				

Refer to [Table 3 on page 50](#) for the register boundary addresses.

6 Backup registers (BKP)

Low-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 16 and 32 Kbytes.

Medium-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 64 and 128 Kbytes.

High-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 256 and 512 Kbytes.

XL-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 768 Kbytes and 1 Mbyte.

Connectivity line devices are STM32F105xx and STM32F107xx microcontrollers.

This section applies to the whole STM32F101xx family, unless otherwise specified.

6.1 BKP introduction

The backup registers are forty two 16-bit registers for storing 84 bytes of user application data.

They are implemented in the backup domain that remains powered on by V_{BAT} when the V_{DD} power is switched off. They are not reset when the device wakes up from Standby mode or by a system reset or power reset.

In addition, the BKP control registers are used to manage the Tamper detection feature and RTC calibration.

After reset, access to the Backup registers and RTC is disabled and the Backup domain (BKP) is protected against possible parasitic write access. To enable access to the Backup registers and the RTC, proceed as follows:

- enable the power and backup interface clocks by setting the PWREN and BKPN bits in the RCC_APB1ENR register
- set the DBP bit in the Power control register (PWR_CR) to enable access to the Backup registers and RTC.

6.2 BKP main features

- 20-byte data registers (in medium-density and low-density devices) or 84-byte data registers (in high-density, XL-density and connectivity line devices)
- Status/control register for managing tamper detection with interrupt capability
- Calibration register for storing the RTC calibration value
- Possibility to output the RTC Calibration Clock, RTC Alarm pulse or Second pulse on TAMPER pin PC13 (when this pin is not used for tamper detection)

6.3 BKP functional description

6.3.1 Tamper detection

The TAMPER pin generates a Tamper detection event when the pin changes from 0 to 1 or from 1 to 0 depending on the TPAL bit in the [Backup control register \(BKP_CR\)](#). A tamper detection event resets all data backup registers.

However to avoid losing Tamper events, the signal used for edge detection is logically ANDed with the Tamper enable in order to detect a Tamper event in case it occurs before the TAMPER pin is enabled.

- **When TPAL=0:** If the TAMPER pin is already high before it is enabled (by setting TPE bit), an extra Tamper event is detected as soon as the TAMPER pin is enabled (while there was no rising edge on the TAMPER pin after TPE was set)
- **When TPAL=1:** If the TAMPER pin is already low before it is enabled (by setting the TPE bit), an extra Tamper event is detected as soon as the TAMPER pin is enabled (while there was no falling edge on the TAMPER pin after TPE was set)

By setting the TPIE bit in the BKP_CSR register, an interrupt is generated when a Tamper detection event occurs.

After a Tamper event has been detected and cleared, the TAMPER pin should be disabled and then re-enabled with TPE before writing to the backup data registers (BKP_DRx) again. This prevents software from writing to the backup data registers (BKP_DRx), while the TAMPER pin value still indicates a Tamper detection. This is equivalent to a level detection on the TAMPER pin.

Note: *Tamper detection is still active when V_{DD} power is switched off. To avoid unwanted resetting of the data backup registers, the TAMPER pin should be externally tied to the correct level.*

6.3.2 RTC calibration

For measurement purposes, the RTC clock with a frequency divided by 64 can be output on the TAMPER pin. This is enabled by setting the CCO bit in the [RTC clock calibration register \(BKP_RTCCR\)](#).

The clock can be slowed down by up to 121 ppm by configuring CAL[6:0] bits.

For more details about RTC calibration and how to use it to improve timekeeping accuracy, refer to AN2604 "STM32F101xx and STM32F103xx RTC calibration".

6.4 BKP registers

Refer to [Section 2.2 on page 45](#) for a list of abbreviations used in register descriptions.

The peripheral registers can be accessed by half-words (16-bit) or words (32-bit).

6.4.1 Backup data register x (BKP_DRx) (x = 1 ..42)

Address offset: 0x04 to 0x28, 0x40 to 0xBC

Reset value: 0x0000 0000

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

Bits 15:0 **D[15:0]** Backup data

These bits can be written with user data.

Note: The BKP_DRx registers are not reset by a System reset or Power reset or when the device wakes up from Standby mode.

They are reset by a Backup Domain reset or by a TAMPER pin event (if the TAMPER pin function is activated).

6.4.2 RTC clock calibration register (BKP_RTCCR)

Address offset: 0x2C

Reset value: 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					ASOS	ASOE	CCO	CAL[6:0]							
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bits 15:10 Reserved, must be kept at reset value.

Bit 9 **ASOS:** Alarm or second output selection

When the ASOE bit is set, the ASOS bit can be used to select whether the signal output on the TAMPER pin is the RTC Second pulse signal or the Alarm pulse signal:

0: RTC Alarm pulse output selected

1: RTC Second pulse output selected

Note: This bit is reset only by a Backup domain reset.

Bit 8 **ASOE**: Alarm or second output enable

Setting this bit outputs either the RTC Alarm pulse signal or the Second pulse signal on the TAMPER pin depending on the ASOS bit.

The output pulse duration is one RTC clock period. The TAMPER pin must not be enabled while the ASOE bit is set.

Note: This bit is reset only by a Backup domain reset.

Bit 7 **CCO**: Calibration clock output

0: No effect

1: Setting this bit outputs the RTC clock with a frequency divided by 64 on the TAMPER pin. The TAMPER pin must not be enabled while the CCO bit is set in order to avoid unwanted Tamper detection.

Note: This bit is reset when the V_{DD} supply is powered off.

Bit 6:0 **CAL[6:0]**: Calibration value

This value indicates the number of clock pulses that will be ignored every 2²⁰ clock pulses. This allows the calibration of the RTC, slowing down the clock by steps of 1000000/2²⁰ PPM.

The clock of the RTC can be slowed down from 0 to 121PPM.

6.4.3 Backup control register (BKP_CR)

Address offset: 0x30

Reset value: 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														TPAL	TPE
														rw	rw

Bits 15:2 Reserved, must be kept at reset value.

Bit 1 **TPAL**: TAMPER pin active level

0: A high level on the TAMPER pin resets all data backup registers (if TPE bit is set).

1: A low level on the TAMPER pin resets all data backup registers (if TPE bit is set).

Bit 0 **TPE**: TAMPER pin enable

0: The TAMPER pin is free for general purpose I/O

1: Tamper alternate I/O function is activated.

Note: Setting the TPAL and TPE bits at the same time is always safe, however resetting both at the same time can generate a spurious Tamper event. For this reason it is recommended to change the TPAL bit only when the TPE bit is reset.

6.4.4 Backup control/status register (BKP_CSR)

Address offset: 0x34

Reset value: 0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
Reserved						TIF	TEF	Reserved						TPIE	CTI	CTE
						r	r							rw	w	w

Bits 15:10 Reserved, must be kept at reset value.

Bit 9 TIF: Tamper interrupt flag

This bit is set by hardware when a Tamper event is detected and the TPIE bit is set. It is cleared by writing 1 to the CTI bit (also clears the interrupt). It is also cleared if the TPIE bit is reset.

- 0: No Tamper interrupt
- 1: A Tamper interrupt occurred

Note: This bit is reset only by a system reset and wakeup from Standby mode.

Bit 8 TEF: Tamper event flag

This bit is set by hardware when a Tamper event is detected. It is cleared by writing 1 to the CTE bit.

- 0: No Tamper event
- 1: A Tamper event occurred

Note: A Tamper event resets all the BKP_DRx registers. They are held in reset as long as the TEF bit is set. If a write to the BKP_DRx registers is performed while this bit is set, the value will not be stored.

Bits 7:3 Reserved, must be kept at reset value.

Bit 2 TPIE: TAMPER pin interrupt enable

- 0: Tamper interrupt disabled
- 1: Tamper interrupt enabled (the TPE bit must also be set in the BKP_CR register)

Note: A Tamper interrupt does not wake up the core from low-power modes.

This bit is reset only by a system reset and wakeup from Standby mode.

Bit 1 CTI: Clear tamper interrupt

This bit is write only, and is always read as 0.

- 0: No effect
- 1: Clear the Tamper interrupt and the TIF Tamper interrupt flag.

Bit 0 CTE: Clear tamper event

This bit is write only, and is always read as 0.

- 0: No effect
- 1: Reset the TEF Tamper event flag (and the Tamper detector)

6.4.5 BKP register map

BKP registers are mapped as 16-bit addressable registers as described in the table below:

Table 17. BKP 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
0x00	Reserved																																
0x04	BKP_DR1	Reserved																		D[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	0	0	0	0	0			

Table 17. BKP 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															
0x08	BKP_DR2	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x0C	BKP_DR3	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x10	BKP_DR4	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x14	BKP_DR5	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x18	BKP_DR6	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x1C	BKP_DR7	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x20	BKP_DR8	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x24	BKP_DR9	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x28	BKP_DR10	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x2	BKP_RTCCR	Reserved												ASOS	ASOE	CCO	CAL[6:0]																															
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	
0x30	BKP_CR	Reserved												TTF	TEF	Reserved												TPAL	TPE																			
	Reset value													0	0																																	
0x34	BKP_CSR	Reserved												TIE	CTI	Reserved												CTE	TPE																			
	Reset value													0	0																																	
0x38		Reserved																																														
0x3C		Reserved																																														
0x40	BKP_DR11	Reserved												D[15:0]																																		
	Reset value													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																	

Table 17. BKP 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
0x44	BKP_DR12	Reserved															D[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	0	0	0	0	0	0		
0x48	BKP_DR13	Reserved															D[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	0	0	0	0	0	0		
0x4C	BKP_DR14	Reserved															D[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	0	0	0	0	0	0		
0x50	BKP_DR15	Reserved															D[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	0	0	0	0	0	0		
0x54	BKP_DR16	Reserved															D[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	0	0	0	0	0	0		
0x58	BKP_DR17	Reserved															D[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	0	0	0	0	0	0		
0x5C	BKP_DR18	Reserved															D[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	0	0	0	0	0	0		
0x60	BKP_DR19	Reserved															D[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	0	0	0	0	0	0		
0x64	BKP_DR20	Reserved															D[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	0	0	0	0	0	0		
0x68	BKP_DR21	Reserved															D[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	0	0	0	0	0	0		
0x6C	BKP_DR22	Reserved															D[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	0	0	0	0	0	0		
0x70	BKP_DR23	Reserved															D[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	0	0	0	0	0	0		
0x74	BKP_DR24	Reserved															D[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	0	0	0	0	0	0		
0x78	BKP_DR25	Reserved															D[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	0	0	0	0	0	0		

Table 17. BKP 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
0x7C	BKP_DR26	Reserved															D[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	0	0	0	0	0	0		
0x80	BKP_DR27	Reserved															D[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	0	0	0	0	0	0		
0x84	BKP_DR28	Reserved															D[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	0	0	0	0	0	0		
0x88	BKP_DR29	Reserved															D[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	0	0	0	0	0	0		
0x8C	BKP_DR30	Reserved															D[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	0	0	0	0	0	0		
0x90	BKP_DR31	Reserved															D[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	0	0	0	0	0	0		
0x94	BKP_DR32	Reserved															D[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	0	0	0	0	0	0		
0x98	BKP_DR33	Reserved															D[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	0	0	0	0	0	0		
0x9C	BKP_DR34	Reserved															D[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	0	0	0	0	0	0		
0xA0	BKP_DR35	Reserved															D[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	0	0	0	0	0	0		
0xA4	BKP_DR36	Reserved															D[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	0	0	0	0	0	0		
0xA8	BKP_DR37	Reserved															D[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	0	0	0	0	0	0		
0xAC	BKP_DR38	Reserved															D[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	0	0	0	0	0	0		
0xB0	BKP_DR39	Reserved															D[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	0	0	0	0	0	0		

- Bit 20 **UART5RST:** USART5 reset
Set and cleared by software.
0: No effect
1: Reset USART5
- Bit 19 **UART4RST:** USART4 reset
Set and cleared by software.
0: No effect
1: Reset USART4
- Bit 18 **USART3RST:** USART3 reset
Set and cleared by software.
0: No effect
1: Reset USART3
- Bit 17 **USART2RST:** USART2 reset
Set and cleared by software.
0: No effect
1: Reset USART2
- Bit 16 Reserved, must be kept at reset value.
- Bit 15 **SPI3RST:** SPI3 reset
Set and cleared by software.
0: No effect
1: Reset SPI3
- Bit 14 **SPI2RST:** SPI2 reset
Set and cleared by software.
0: No effect
1: Reset SPI2
- Bits 13:12 Reserved, must be kept at reset value.
- Bit 11 **WWDGRST:** Window watchdog reset
Set and cleared by software.
0: No effect
1: Reset window watchdog
- Bits 10:9 Reserved, must be kept at reset value.
- Bit 8 **TIM14RST:** TIM14 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM14
- Bit 7 **TIM13RST:** TIM13 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM13
- Bit 6 **TIM12RST:** TIM12 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM12

- Bit 5 **TIM7RST:** TIM7 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM7
- Bit 4 **TIM6RST:** TIM6 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM6
- Bit 3 **TIM5RST:** TIM5 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM5
- Bit 2 **TIM4RST:** TIM4 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM4
- Bit 1 **TIM3RST:** TIM3 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM3
- Bit 0 **TIM2RST:** TIM2 timer reset
Set and cleared by software.
0: No effect
1: Reset TIM2

7.3.6 AHB peripheral clock enable register (RCC_AHBENR)

Address offset: 0x14

Reset value: 0x0000 0014

Access: no wait state, word, half-word and byte access

Note: When the peripheral clock is not active, the peripheral register values may not be readable by software and the returned value is always 0x0.

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					SDIO EN	Res.	FSMC EN	Res.	CRCE N	Res.	FLITF EN	Res.	SRAM EN	DMA2 EN	DMA1 EN
					rw		rw		rw		rw		rw	rw	rw

Bits 31:11 Reserved, must be kept at reset value.

- Bit 10 **SDIOEN:** SDIO clock enable
Set and cleared by software.
0: SDIO clock disabled
1: SDIO clock enabled

Bits 9 Reserved, always read as 0.

- Bit 8 **FSMCEN:** FSMC clock enable
Set and cleared by software.
0: FSMC clock disabled
1: FSMC clock enabled
- Bit 7 Reserved, always read as 0.
- Bit 6 **CRCEN:** CRC clock enable
Set and cleared by software.
0: CRC clock disabled
1: CRC clock enabled
- Bit 5 Reserved, must be kept at reset value.
- Bit 4 **FLITFEN:** FLITF clock enable
Set and cleared by software to disable/enable FLITF clock during Sleep mode.
0: FLITF clock disabled during Sleep mode
1: FLITF clock enabled during Sleep mode
- Bit 3 Reserved, must be kept at reset value.
- Bit 2 **SRAMEN:** SRAM interface clock enable
Set and cleared by software to disable/enable SRAM interface clock during Sleep mode.
0: SRAM interface clock disabled during Sleep mode.
1: SRAM interface clock enabled during Sleep mode
- Bit 1 **DMA2EN:** DMA2 clock enable
Set and cleared by software.
0: DMA2 clock disabled
1: DMA2 clock enabled
- Bit 0 **DMA1EN:** DMA1 clock enable
Set and cleared by software.
0: DMA1 clock disabled
1: DMA1 clock enabled

7.3.7 APB2 peripheral clock enable register (RCC_APB2ENR)

Address: 0x18

Reset value: 0x0000 0000

Access: word, half-word and byte access

No wait states, except if the access occurs while an access to a peripheral in the APB2 domain is on going. In this case, wait states are inserted until the access to APB2 peripheral is finished.

Note: When the peripheral clock is not active, the peripheral register values may not be readable by software and the returned value is always 0x0.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
Reserved										TIM11 EN	TIM10 EN	TIM9 EN	Reserved			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
ADC3 EN	USART 1EN	TIM8 EN	SPI1 EN	TIM1 EN	ADC2 EN	ADC1 EN	IOPG EN	IOPF EN	IOPE EN	IOPD EN	IOPC EN	IOPB EN	IOPA EN	Res.	AFIO EN	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	

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

Bit 21 **TIM11EN:** TIM11 timer clock enable

Set and cleared by software.

0: TIM11 timer clock disabled

1: TIM11 timer clock enabled

Bit 20 **TIM10EN:** TIM10 timer clock enable

Set and cleared by software.

0: TIM10 timer clock disabled

1: TIM10 timer clock enabled

Bit 19 **TIM9EN:** TIM9 timer clock enable

Set and cleared by software.

0: TIM9 timer clock disabled

1: TIM9 timer clock enabled

Bits 18:16 Reserved, always read as 0.

Bit 15 **ADC3EN:** ADC3 interface clock enable

Set and cleared by software.

0: ADC3 interface clock disabled

1: ADC3 interface clock enabled

Bit 14 **USART1EN:** USART1 clock enable

Set and cleared by software.

0: USART1 clock disabled

1: USART1 clock enabled

Bit 13 **TIM8EN:** TIM8 Timer clock enable

Set and cleared by software.

0: TIM8 timer clock disabled

1: TIM8 timer clock enabled

Bit 12 **SPI1EN:** SPI1 clock enable

Set and cleared by software.

0: SPI1 clock disabled

1: SPI1 clock enabled

Bit 11 **TIM1EN:** TIM1 timer clock enable

Set and cleared by software.

0: TIM1 timer clock disabled

1: TIM1 timer clock enabled

Bit 10 **ADC2EN:** ADC 2 interface clock enable

Set and cleared by software.

0: ADC 2 interface clock disabled

1: ADC 2 interface clock enabled

Bit 9 **ADC1EN**: ADC 1 interface clock enable

Set and cleared by software.

0: ADC 1 interface disabled

1: ADC 1 interface clock enabled

Bit 8 **IOPGEN**: IO port G clock enable

Set and cleared by software.

0: IO port G clock disabled

1: IO port G clock enabled

Bit 7 **IOPFEN**: IO port F clock enable

Set and cleared by software.

0: IO port F clock disabled

1: IO port F clock enabled

Bit 6 **IOPEEN**: IO port E clock enable

Set and cleared by software.

0: IO port E clock disabled

1: IO port E clock enabled

Bit 5 **IOPDEN**: IO port D clock enable

Set and cleared by software.

0: IO port D clock disabled

1: IO port D clock enabled

Bit 4 **IOPCEN**: IO port C clock enable

Set and cleared by software.

0: IO port C clock disabled

1: IO port C clock enabled

Bit 3 **IOPBEN**: IO port B clock enable

Set and cleared by software.

0: IO port B clock disabled

1: IO port B clock enabled

Bit 2 **IOPAEN**: IO port A clock enable

Set and cleared by software.

0: IO port A clock disabled

1: IO port A clock enabled

Bit 1 Reserved, must be kept at reset value.

Bit 0 **AFIOEN**: Alternate function IO clock enable

Set and cleared by software.

0: Alternate Function IO clock disabled

1: Alternate Function IO clock enabled

7.3.8 APB1 peripheral clock enable register (RCC_APB1ENR)

Address: 0x1C

Reset value: 0x0000 0000

Access: word, half-word and byte access

No wait state, except if the access occurs while an access to a peripheral on APB1 domain is on going. In this case, wait states are inserted until this access to APB1 peripheral is finished.

Note: *When the peripheral clock is not active, the peripheral register values may not be readable by software and the returned value is always 0x0.*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		DAC EN	PWR EN	BKP EN	Res.	CAN EN	Res.	USB EN	I2C2 EN	I2C1 EN	UART5 EN	UART4 EN	USART3 EN	USART2 EN	Res.
		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
SPI3 EN	SPI2 EN	Reserved	WWD GEN	Reserved	TIM14 EN	TIM13 EN	TIM12 EN	TIM7 EN	TIM6 EN	TIM5 EN	TIM4 EN	TIM3 EN	TIM2 EN		
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 **DACEN:** DAC interface clock enable

Set and cleared by software.

0: DAC interface clock disabled

1: DAC interface clock enable

Bit 28 **PWREN:** Power interface clock enable

Set and cleared by software.

0: Power interface clock disabled

1: Power interface clock enable

Bit 27 **BKOPEN:** Backup interface clock enable

Set and cleared by software.

0: Backup interface clock disabled

1: Backup interface clock enabled

Bit 26 Reserved, must be kept at reset value.

Bit 25 **CANEN:** CAN clock enable

Set and cleared by software.

0: CAN clock disabled

1: CAN clock enabled

Bit 24 Reserved, always read as 0.

Bit 23 **USBEN:** USB clock enable

Set and cleared by software.

0: USB clock disabled

1: USB clock enabled

- Bit 22 **I2C2EN:** I2C2 clock enable
Set and cleared by software.
0: I2C2 clock disabled
1: I2C2 clock enabled
- Bit 21 **I2C1EN:** I2C1 clock enable
Set and cleared by software.
0: I2C1 clock disabled
1: I2C1 clock enabled
- Bit 20 **UART5EN:** USART5 clock enable
Set and cleared by software.
0: USART5 clock disabled
1: USART5 clock enabled
- Bit 19 **UART4EN:** USART4 clock enable
Set and cleared by software.
0: USART4 clock disabled
1: USART4 clock enabled
- Bit 18 **USART3EN:** USART3 clock enable
Set and cleared by software.
0: USART3 clock disabled
1: USART3 clock enabled
- Bit 17 **USART2EN:** USART2 clock enable
Set and cleared by software.
0: USART2 clock disabled
1: USART2 clock enabled
- Bits 16 Reserved, always read as 0.
- Bit 15 **SPI3EN:** SPI 3 clock enable
Set and cleared by software.
0: SPI 3 clock disabled
1: SPI 3 clock enabled
- Bit 14 **SPI2EN:** SPI2 clock enable
Set and cleared by software.
0: SPI2 clock disabled
1: SPI2 clock enabled
- Bits 13:12 Reserved, must be kept at reset value.
- Bit 11 **WWDGEN:** Window watchdog clock enable
Set and cleared by software.
0: Window watchdog clock disabled
1: Window watchdog clock enabled
- Bits 10:9 Reserved, must be kept at reset value.
- Bit 8 **TIM14EN:** TIM14 timer clock enable
Set and cleared by software.
0: TIM14 clock disabled
1: TIM14 clock enabled

- Bit 7 **TIM13EN:** TIM13 timer clock enable
Set and cleared by software.
0: TIM13 clock disabled
1: TIM13 clock enabled
- Bit 6 **TIM12EN:** TIM12 timer clock enable
Set and cleared by software.
0: TIM12 clock disabled
1: TIM12 clock enabled
- Bit 5 **TIM7EN:** TIM7 timer clock enable
Set and cleared by software.
0: TIM7 clock disabled
1: TIM7 clock enabled
- Bit 4 **TIM6EN:** TIM6 timer clock enable
Set and cleared by software.
0: TIM6 clock disabled
1: TIM6 clock enabled
- Bit 3 **TIM5EN:** TIM5 timer clock enable
Set and cleared by software.
0: TIM5 clock disabled
1: TIM5 clock enabled
- Bit 2 **TIM4EN:** TIM4 timer clock enable
Set and cleared by software.
0: TIM4 clock disabled
1: TIM4 clock enabled
- Bit 1 **TIM3EN:** TIM3 timer clock enable
Set and cleared by software.
0: TIM3 clock disabled
1: TIM3 clock enabled
- Bit 0 **TIM2EN:** TIM2 timer clock enable
Set and cleared by software.
0: TIM2 clock disabled
1: TIM2 clock enabled

7.3.9 Backup domain control register (RCC_BDCR)

Address offset: 0x20

Reset value: 0x0000 0000, reset by Backup domain Reset.

Access: $0 \leq$ wait state ≤ 3 , word, half-word and byte access

Wait states are inserted in case of successive accesses to this register.

Note:

The LSEON, LSEBYP, RTCSEL and RTCEN bits of the Backup domain control register (RCC_BDCR) are in the Backup domain. As a result, after Reset, these bits are write-protected and the DBP bit in the Power control register (PWR_CR) has to be set before these can be modified. Refer to Section 5: Power control (PWR) for further information. These bits are only reset after a Backup domain Reset (see Section 7.1.3: Backup domain reset). Any internal or external Reset will not have any effect on these bits.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														BDRST	
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTC EN	Reserved				RTCSEL[1:0]		Reserved				LSE BYP	LSE RDY	LSEON		
					rw	rw					rw	r	rw		

Bits 31:17 Reserved, must be kept at reset value.

Bit 16 **BDRST**: Backup domain software reset

Set and cleared by software.

0: Reset not activated

1: Resets the entire Backup domain

Bit 15 **RTCEN**: RTC clock enable

Set and cleared by software.

0: RTC clock disabled

1: RTC clock enabled

Bits 14:10 Reserved, must be kept at reset value.

Bits 9:8 **RTCSEL[1:0]**: RTC clock source selection

Set by software to select the clock source for the RTC. Once the RTC clock source has been selected, it cannot be changed anymore unless the Backup domain is reset. The BDRST bit can be used to reset them.

00: No clock

01: LSE oscillator clock used as RTC clock

10: LSI oscillator clock used as RTC clock

11: HSE oscillator clock divided by 128 used as RTC clock

Bits 7:3 Reserved, must be kept at reset value.

Bit 2 **LSEBYP:** External low-speed oscillator bypass

Set and cleared by software to bypass oscillator in debug mode. This bit can be written only when the external 32 kHz oscillator is disabled.

- 0: LSE oscillator not bypassed
- 1: LSE oscillator bypassed

Bit 1 **LSERDY:** External low-speed oscillator ready

Set and cleared by hardware to indicate when the external 32 kHz oscillator is stable. After the LSEON bit is cleared, LSERDY goes low after 6 external low-speed oscillator clock cycles.

- 0: External 32 kHz oscillator not ready
- 1: External 32 kHz oscillator ready

Bit 0 **LSEON:** External low-speed oscillator enable

Set and cleared by software.

- 0: External 32 kHz oscillator OFF
- 1: External 32 kHz oscillator ON

7.3.10 Control/status register (RCC_CSR)

Address: 0x24

Reset value: 0x0C00 0000, reset by system Reset, except reset flags by power Reset only.

Access: $0 \leq$ wait state ≤ 3 , word, half-word and byte access

Wait states are inserted in case of successive accesses to this register.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	POR RSTF	PIN RSTF	Res.	RMVF	Reserved								
rw	rw	rw	rw	rw	rw	Res.	rw	15	14	13	12	11	10	9	8	
15	14	13	12	11	10		rw	7	6	5	4	3	2	1	0	
Reserved												LSI RDY		LSION		
												r	rw			

Bit 31 LPWRRSTF: Low-power reset flag

Set by hardware when a Low-power management reset occurs.
Cleared by writing to the RMVF bit.

0: No Low-power management reset occurred
1: Low-power management reset occurred

For further information on Low-power management reset, refer to [Low-power management reset](#).

Bit 30 WWDGRSTF: Window watchdog reset flag

Set by hardware when a window watchdog reset occurs.
Cleared by writing to the RMVF bit.

0: No window watchdog reset occurred
1: Window watchdog reset occurred

Bit 29 IWDGRSTF: Independent watchdog reset flag

Set by hardware when an independent watchdog reset from V_{DD} domain occurs.
Cleared by writing to the RMVF bit.

0: No watchdog reset occurred
1: Watchdog reset occurred

Bit 28 SFRSTF: Software reset flag

Set by hardware when a software reset occurs.
Cleared by writing to the RMVF bit.
0: No software reset occurred
1: Software reset occurred

Bit 27 PORRSTF: POR/PDR reset flag

Set by hardware when a POR/PDR reset occurs.
Cleared by writing to the RMVF bit.
0: No POR/PDR reset occurred
1: POR/PDR reset occurred

Bit 26 PINRSTF: PIN reset flag

Set by hardware when a reset from the NRST pin occurs.
Cleared by writing to the RMVF bit.
0: No reset from NRST pin occurred
1: Reset from NRST pin occurred

Bit 25 Reserved, must be kept at reset value.

Bit 24 RMVF: Remove reset flag

Set by software to clear the reset flags.
0: No effect
1: Clear the reset flags

Bits 23:2 Reserved, must be kept at reset value.

Bit 1 LSIRDY: Internal low-speed oscillator ready

Set and cleared by hardware to indicate when the internal RC 40 kHz oscillator is stable.
After the LSION bit is cleared, LSIRDY goes low after 3 internal RC 40 kHz oscillator clock cycles.
0: Internal RC 40 kHz oscillator not ready
1: Internal RC 40 kHz oscillator ready

Bit 0 LSION: Internal low-speed oscillator enable

Set and cleared by software.
0: Internal RC 40 kHz oscillator OFF
1: Internal RC 40 kHz oscillator ON

7.3.11 RCC register map

The following table gives the RCC register map and the reset values.

Table 18. RCC 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											
0x00	RCC_CR																																											
	Reset value																																											
0x04	RCC_CFGR																																											
	Reset value																																											
0x08	RCC_CIR																																											
	Reset value																																											
0x0C	RCC_APB2RSTR																																											
	Reset value																																											
0x0E	RCC_APB1RSTR																																											
	Reset value																																											
0x14	RCC_AHBENR																																											
	Reset value																																											
0x18	RCC_APB2ENR																																											
	Reset value																																											
0x1C	RCC_APB1ENR																																											
	Reset value																																											
0x20	RCC_BDCR																																											
	Reset value																																											

Table 18. RCC 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
0x24	RCC_CSR	LPWWRSTF	WWDGRSTF	IWDGRSTF	SFRSTF	PORRSTF	PINRSTF	Reserved	RMVF																						LSIRDY	LSION	
	Reset value	0	0	0	0	0	1	1	0																					0	0		

Refer to [Table 3 on page 50](#) for the register boundary addresses.

8 Connectivity line devices: reset and clock control (RCC)

Low-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 16 and 32 Kbytes.

Medium-density devices are STM32F101xx, STM32F102xx and STM32F103xx microcontrollers where the Flash memory density ranges between 64 and 128 Kbytes.

High-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 256 and 512 Kbytes.

XL-density devices are STM32F101xx and STM32F103xx microcontrollers where the Flash memory density ranges between 768 Kbytes and 1 Mbyte.

Connectivity line devices are STM32F105xx and STM32F107xx microcontrollers.

This section applies to all connectivity line devices, unless otherwise specified.

8.1 Reset

There are three types of reset, defined as system reset, power reset and backup domain reset.

8.1.1 System reset

A system reset sets all registers to their reset values except the reset flags in the clock controller CSR register and the registers in the Backup domain (see [Figure 4](#)).

A system reset is generated when one of the following events occurs:

1. A low level on the NRST pin (external reset)
2. Window watchdog end of count condition (WWDG reset)
3. Independent watchdog end of count condition (IWDG reset)
4. A software reset (SW reset) (see [Software reset](#))
5. Low-power management reset (see [Low-power management reset](#))

The reset source can be identified by checking the reset flags in the Control/Status register, RCC_CSR (see [Section 8.3.10: Control/status register \(RCC_CSR\)](#)).

Software reset

The SYSRESETREQ bit in Cortex®-M3 Application Interrupt and Reset Control register must be set to force a software reset on the device. Refer to the *STM32F10xxx Cortex®-M3 programming manual* (see [Related documents](#)) for more details.

Low-power management reset

There are two ways to generate a low-power management reset:

1. Reset generated when entering Standby mode:

This type of reset is enabled by resetting nRST_STDBY bit in User Option Bytes. In this case, whenever a Standby mode entry sequence is successfully executed, the device is reset instead of entering Standby mode.

2. Reset when entering Stop mode:

This type of reset is enabled by resetting nRST_STOP bit in User Option Bytes. In this case, whenever a Stop mode entry sequence is successfully executed, the device is reset instead of entering Stop mode.

For further information on the User Option Bytes, refer to the STM32F10xxx Flash programming manual.

8.1.2 Power reset

A power reset is generated when one of the following events occurs:

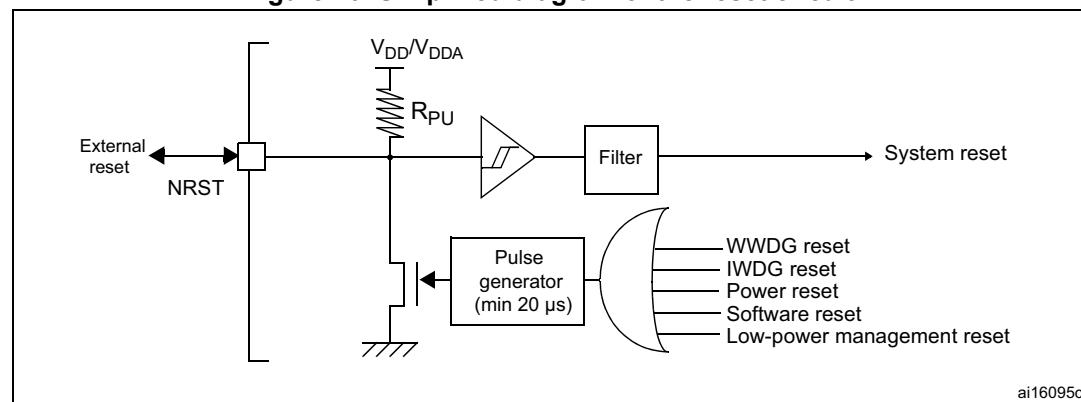
1. Power-on/power-down reset (POR/PDR reset)
2. When exiting Standby mode

A power reset sets all registers to their reset values except the Backup domain (see [Figure 4](#))

These sources act on the NRST pin and it is always kept low during the delay phase. The RESET service routine vector is fixed at address 0x0000_0004 in the memory map. For more details, refer to [Table 63: Vector table for other STM32F10xxx devices](#).

The system reset signal provided to the device is output on the NRST pin. The pulse generator guarantees a minimum reset pulse duration of 20 µs for each reset source (external or internal reset). In case of an external reset, the reset pulse is generated while the NRST pin is asserted low.

Figure 10. Simplified diagram of the reset circuit



8.1.3 Backup domain reset

The backup domain has two specific resets that affect only the backup domain (see [Figure 4](#)).

A backup domain reset is generated when one of the following events occurs:

1. Software reset, triggered by setting the BDRST bit in the [*Backup domain control register \(RCC_BDCR\)*](#).
2. V_{DD} or V_{BAT} power on, if both supplies have previously been powered off.

8.2 Clocks

Three different clock sources can be used to drive the system clock (SYSCLK):

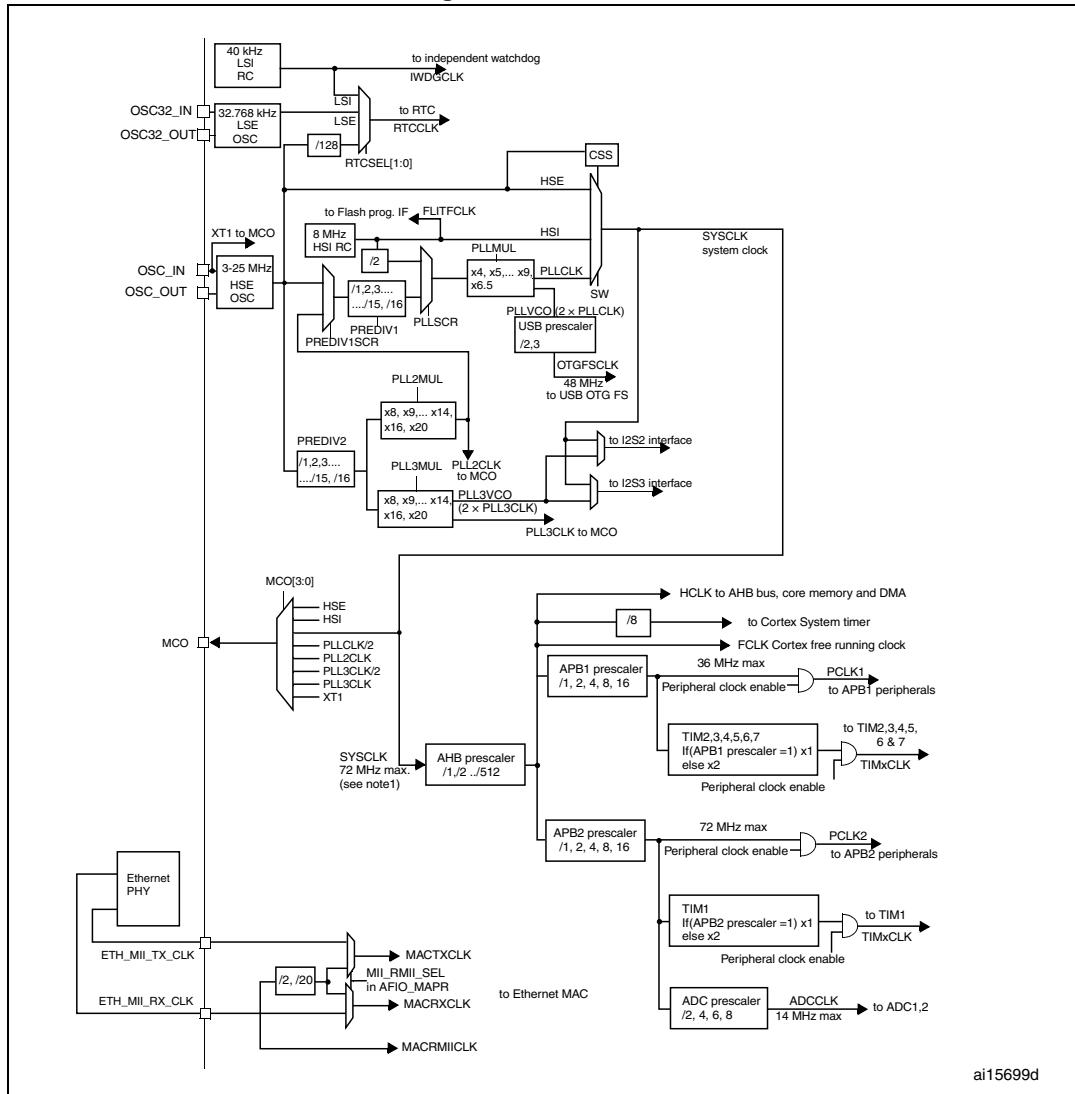
- HSI oscillator clock
- HSE oscillator clock
- PLL clock

The devices have the following two secondary clock sources:

- 40 kHz low speed internal RC (LSI RC) which drives the independent watchdog and optionally the RTC used for Auto-wakeup from Stop/Standy mode.
- 32.768 kHz low speed external crystal (LSE crystal) which optionally drives the real-time clock (RTCCLK)

Each clock source can be switched on or off independently when it is not used, to optimize power consumption.

Figure 11. Clock tree



- When the HSI is used as a PLL clock input, the maximum system clock frequency that can be achieved is 36 MHz.
- For full details about the internal and external clock source characteristics, refer to the "Electrical characteristics" section in your device datasheet.

The advanced clock controller features 3 PLLs to provide a high degree of flexibility to the application in the choice of the external crystal or oscillator to run the core and peripherals at the highest frequency and guarantee the appropriate frequency for the Ethernet and USB OTG FS.

A single 25 MHz crystal can clock the entire system and all peripherals including the Ethernet and USB OTG FS peripherals. In order to achieve high-quality audio performance, an audio crystal can be used. In this case, the I²S master clock can generate all standard sampling frequencies from 8 kHz to 96 kHz with less than 0.5% accuracy.

For more details about clock configuration for applications requiring Ethernet, USB OTG FS and/or I²S (audio), refer to "Appendix A Applicative block diagrams" in your connectivity line device datasheet.

Several prescalers allow the configuration of the AHB frequency, the high speed APB (APB2) and the low speed APB (APB1) domains. The maximum frequency of the AHB and the APB2 domains is 72 MHz. The maximum allowed frequency of the APB1 domain is 36 MHz.

All peripheral clocks are derived from the system clock (SYSCLK) except:

- The Flash memory programming interface clock (FLITFCLK) is always the HSI clock
- The USB OTG FS 48 MHz clock which is derived from the PLL VCO clock ($2 \times$ PLLCLK), followed by a programmable prescaler (divide by 3 or 2). This selection is made through the OTGFSPRE bit in the RCC_CFGR register. For proper USB OTG FS operation, the PLL should be configured to output 72 MHz or 48 MHz.
- The I2S2 and I2S3 clocks which can be derived from the system clock (SYSCLK) or the PLL3 VCO clock ($2 \times$ PLL3CLK). This selection is made through the I2SxSRC bit in the RCC_CFGR2 register. For more information on PLL3 and how to configure the I2S clock to achieve high-quality audio performance, refer to [Section 25.4.3: Clock generator](#).
- The Ethernet MAC clocks (TX, RX and RMII) which are provided from the external PHY. For further information on the Ethernet configuration, refer to [Section 29.4.4: MII/RMII selection](#).

When the Ethernet is used, the AHB clock frequency must be at least 25 MHz.

The RCC feeds the Cortex[®] System Timer (SysTick) external clock with the AHB clock (HCLK) divided by 8. The SysTick can work either with this clock or with the Cortex[®] clock (HCLK), configurable in the SysTick Control and Status register. The ADCs are clocked by the clock of the High Speed domain (APB2) divided by 2, 4, 6 or 8.

The timer clock frequencies are automatically fixed by hardware. There are two cases:

1. if the APB prescaler is 1, the timer clock frequencies are set to the same frequency as that of the APB domain to which the timers are connected.
2. otherwise, they are set to twice ($\times 2$) the frequency of the APB domain to which the timers are connected.

FCLK acts as Cortex[®]-M3 free-running clock. For more details refer to Arm[®] *Cortex-M3 r1p1 Technical Reference Manual (TRM)*.

8.2.1 HSE clock

The high speed external clock signal (HSE) can be generated from two possible clock sources:

- HSE external crystal/ceramic resonator
- HSE user external clock

The resonator and the load capacitors have to be placed as close as possible to the oscillator pins in order to minimize output distortion and startup stabilization time. The loading capacitance values must be adjusted according to the selected oscillator.

Figure 12. HSE/ LSE clock sources

Clock source	Hardware configuration
External clock	
Crystal/ceramic resonators	

External source (HSE bypass)

In this mode, an external clock source must be provided. It can have a frequency of up to 50 MHz. Select this mode by setting the HSEBYP and HSEON bits in the [Clock control register \(RCC_CR\)](#). The external clock signal (square, sinus or triangle) with ~50% duty cycle has to drive the OSC_IN pin while the OSC_OUT pin should be left hi-Z. See [Figure 12](#).

External crystal/ceramic resonator (HSE crystal)

The 3 to 25 MHz external oscillator has the advantage of producing a very accurate rate on the main clock.

The associated hardware configuration is shown in [Figure 12](#). Refer to the electrical characteristics section of the *datasheet* for more details.

The HSERDY flag in the [Clock control register \(RCC_CR\)](#) indicates if the high-speed external oscillator is stable or not. At startup, the clock is not released until this bit is set by hardware. An interrupt can be generated if enabled in the [Clock interrupt register \(RCC_CIR\)](#).

The HSE crystal can be switched on and off using the HSEON bit in the [Clock control register \(RCC_CR\)](#).

8.2.2 HSI clock

The HSI clock signal is generated from an internal 8 MHz RC Oscillator and can be used directly as a system clock or divided by 2 to be used as PLL input.

The HSI RC oscillator has the advantage of providing a clock source at low cost (no external components). It also has a faster startup time than the HSE crystal oscillator however, even with calibration the frequency is less accurate than an external crystal oscillator or ceramic resonator.

Calibration

RC oscillator frequencies can vary from one chip to another due to manufacturing process variations, this is why each device is factory calibrated by ST for 1% accuracy at $T_A = 25^\circ\text{C}$.

After reset, the factory calibration value is loaded in the HSICAL[7:0] bits in the [Clock control register \(RCC_CR\)](#).

If the application is subject to voltage or temperature variations this may affect the RC oscillator speed. You can trim the HSI frequency in the application using the HSITRIM[4:0] bits in the [Clock control register \(RCC_CR\)](#).

The HSIRDY flag in the [Clock control register \(RCC_CR\)](#) indicates if the HSI RC is stable or not. At startup, the HSI RC output clock is not released until this bit is set by hardware.

The HSI RC can be switched on and off using the HSION bit in the [Clock control register \(RCC_CR\)](#).

The HSI signal can also be used as a backup source (Auxiliary clock) if the HSE crystal oscillator fails. Refer to [Section 8.2.7: Clock security system \(CSS\)](#).

8.2.3 PLLs

The main PLL provides a frequency multiplier starting from one of the following clock sources:

- HSI clock divided by 2
- HSE or PLL2 clock through a configurable divider

Refer to [Figure 11](#) and [Clock control register \(RCC_CR\)](#).

PLL2 and PLL3 are clocked by HSE through a specific configurable divider. Refer to [Figure 11](#) and [Clock configuration register2 \(RCC_CFGR2\)](#)

The configuration of each PLL (selection of clock source, predivision factor and multiplication factor) must be done before enabling the PLL. Each PLL should be enabled after its input clock becomes stable (ready flag). Once the PLL is enabled, these parameters can not be changed.

When changing the entry clock source of the main PLL, the original clock source must be switched off only after the selection of the new clock source (done through bit PLLSRC in the Clock configuration register (RCC_CFGR)).

An interrupt can be generated when the PLL is ready if enabled in the [Clock interrupt register \(RCC_CIR\)](#).

8.2.4 LSE clock

The LSE crystal is a 32.768 kHz Low Speed External crystal or ceramic resonator. It has the advantage providing a low-power but highly accurate clock source to the real-time clock peripheral (RTC) for clock/calendar or other timing functions.

The LSE crystal is switched on and off using the LSEON bit in [Backup domain control register \(RCC_BDCR\)](#).

The LSERDY flag in the [Backup domain control register \(RCC_BDCR\)](#) indicates if the LSE crystal is stable or not. At startup, the LSE crystal output clock signal is not released until this bit is set by hardware. An interrupt can be generated if enabled in the [Clock interrupt register \(RCC_CIR\)](#).