AN4612
Application note

Migrating from STM32L1 Series to STM32L4 Series and STM32L4+ Series microcontrollers

Introduction

For designers of the STM32 microcontroller applications, being able to easily replace one microcontroller type by another in the same product family is an important asset. Migrating an application to a different microcontroller is often needed when product requirements grow, putting extra demands on memory size, or increasing the number of I/Os. The cost reduction objectives may also be an argument to switch to smaller components and shrink the PCB area.

This application note analyzes the steps required to migrate an existing design from STM32L1 Series to STM32L4 Series and STM32L4+ Series. Three aspects need to be considered for the migration: the hardware, the peripheral and the firmware.

This document lists the full set of features available for STM32L1 Series and the equivalent features on STM32L4 Series and STM32L4+ Series (some products may have less features depending on their part number).

To fully benefit from this application note, the user must be familiar with the STM32 microcontrollers documentation available on *www.st.com* with a particular focus on:

- STM32L1 Series reference manual (RM0038)
- STM32L1 Series datasheets
- STM32L4 Series reference manuals:
  - RM0351 (STM32L4x5xx, STM32L4x6xx)
  - RM0394 (STM32L41xxx, STM32L42xxx, STM32L43xxx, STM32L44xxx, STM32L45xxx, STM32L46xxx)
  - RM0392 (STM32L471xx)
- STM32L4 Series datasheets
- STM32L4+ Series reference manual (RM0432)
- STM32L4+ Series datasheets.

Contents

List of tables

List of figures

1 STM32L4 Series and STM32L4+ Series overview

The STM32L4 Series and STM32L4+ Series have a perfect fit in terms of ultra-low power, performance, memory size and peripherals at a cost-effective price.

In particular, the STM32L4 Series and STM32L4+ Series enable a higher frequency and a higher performance operation than the STM32L1 Series devices. STM32L4 Series feature an Arm®(a) Cortex®-M4 @80 MHz or @120 MHz in STM32L4+ Series versus a Cortex®-M3 @32 MHz featured on STM32L1 Series. STM32L4 Series and STM32L4+ Series also feature an optimized Flash memory access through the adaptive real-time memory accelerator (ART Accelerator™).

The STM32L4 Series and STM32L4+ Series MCUs increase the low-power efficiency in Dynamic mode (µA/MHz), and reach a very low level of static power-consumption on the various available low-power modes.

The detailed list of available features and packages for each product is available in the respective datasheet.

STM32L4 Series and STM32L4+ Series include a larger set of peripherals with advanced features compared to STM32L1 Series, such as:

- Touch sensing controller (TSC)
- Controller area network (bxCAN)
- Single-wire protocol interface (SWPMI)
- Serial audio interface (SAI)
- Low-power UART (LPUART)
- Infrared interface (IRTIM)
- Low-power timer (LPTIM)
- Digital filter for sigma delta modulators (DFSDM) (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L47xxx/48xxx and STM32L45xxx/46xxx)
- Voltage reference buffer (VREFBUF)
- Quad-SPI interface (QUADSPI)
- OCTO-SPI (OCTOPSI) (for STM32L4+ Series)
- OCTOSPI IO Manager (OCTOSPIM) (for STM32L4+ Series)
- Display serial interface (DSI) (for STM32L4R9xx/4S9xx)
- LCD-TFT display controller (LTDC) (for STM32L4R7xx/4S7xx/4R9xx/4S9xx)
- DMA request multiplexer (DMAMUX) (for STM32L4+ Series)
- Graphic MMU (GFXMMU) (for STM32L4+ Series)
- Firewall (FW)
- Clock recovery system (CRS) for USB (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)
- Hash processor (HASH) (for STM32L4Sxxx and STM32L49xxx/4Axxx)
- Digital camera interface (DCMI) (for STM32L4+ Series and STM32L49xxx/4Axxx)

arm

a. Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

- Chrom-ART Accelerator™ controller (DMA2D) (for STM32L4+ Series and STM32L49xxx/4Axxx)
- SRAM1 size is different on the various STM32L4 Series and STM32L4+ Series devices:
  - 192 Kbytes for STM32L4+ Series
  - 256 Kbytes for STM32L49xxx/4Axxx
  - 96 Kbytes for STM32L47xxx/48xxx
  - 128 Kbytes for STM32L45xxx/46xxx
  - 48 Kbytes for STM32L43xxx/44xxx
  - 32 Kbytes for STM32L41xxx/44xxx
- Additional SRAM2 with data preservation in Standby mode:
  - 64 Kbytes for STM32L4+ Series and STM32L49xxx/4Axxx
  - 32 Kbytes for STM32L47xxx/48xxx and STM32L45xxx/46xxx
  - 16 Kbytes for STM32L43xxx/44xxx
  - 8 Kbytes for STM32L41xxx/42xxx
- Additional SRAM3 for STM32L4+ Series:
  - 384 Kbytes
- Optimized power consumption and enriched set of low-power modes.

STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx implement an USB FS device only instead of an USB OTG FS. They also implement reduced Flash size (512 Kbytes for STM32L45xxx/46xxx, 256 Kbytes for STM32L43xxx/44xxx and 128 Kbytes for STM32L41xxx/42xxx).

This migration guide is only covering the migration from STM32L1 Series to STM32L4 Series and STM32L4+ Series, and as a consequence any new features present on STM32L4 Series and STM32L4+ Series but not already present on STM32L1 Series are not covered in this document. Refer to the STM32L4 Series and STM32L4+ Series reference manuals and datasheets for an exhaustive picture.

Table 1. STM32L4 Series / STM32L4+ Series memory availability

Part number	Flash size		RAM size			Feature level
	Size	Bank	SRAM1	SRAM2	SRAM3	
STM32L4S9xx	2 Mbytes	Dual	192 Kbytes	64 Kbytes	384 Kbytes	9+crypto
STM32L4R9xx						9
STM32L4S7xx						7+crypto
STM32L4R7xx						7
STM32L4S5xx						5+crypto
STM32L4R5xx						5
STM32L496xx	1 Mbyte		256 Kbytes	64 Kbytes	NA	6
STM32L4A6xx						6+crypto
STM32L471xx			96 Kbytes	32 Kbytes		1
STM32L475xx						5
STM32L476xx						6
STM32L486xx						6+crypto
STM32L451xx	512 Kbytes	Single	128 Kbytes	32 Kbytes		1
STM32L452xx						2
STM32L462xx						2+crypto
STM32L431xx	256 Kbytes		48 Kbytes	16 Kbytes		1
STM32L432xx						2
STM32L442xx						2+crypto
STM32L433xx						3
STM32L443xx						3+crypto
STM32L412xx	128 Kbytes		32 Kbytes	8 Kbytes		2
STM32L422xx						2+crypto

2 Hardware migration

Some packages are available in both STM32L4 Series / STM32L4+ Series and STM32L1 Series such as: LQFP48, LQFP64, LQFP100, LQFP144, BGA64, BGA100 and BGA132. The other packages available on STM32L1 Series are not available on STM32L4 Series / STM32L4+ Series.

Note that the WLCSP packages in STM32L1 Series and the ones in STM32L4 Series / STM32L4+ Series are not equivalent. They have different die sizes for each product.

The list of the available packages in STM32L4 Series / STM32L4+ Series is given in *Table 2*.

Table 2. Packages available on STM32L4 Series and STM32L4+ Series

Package[1]	STM32L4+ Series	STM32L4 Series					Size (mm x mm)	Applicable part numbers
		STM32L 49xxx/ 4Axxx	STM32L 47xxx/ 48xxx	STM32L 45xxx/ 46xxx	STM32L 43xxx/ 44xxx	STM32L 41xxx/ 42xxx		
UFQFPN32	-	-	-	-	X	X	(5 x 5)	STM32L412xx, STM32L422xx, STM32L431xx, STM32L432xx, STM32L442xx
LQFP32	-	-	-	-	-	X	(5 x 5)	STM32L412xx, STM32L422xx
LQFP48	-	-	-	-	X	X	(7 x 7)	STM32L412xx, STM32L422xx, STM32L431xx, STM32L433xx, STM32L443xx
UFQFPN48	-	-	-	X	X	X	(7 x 7)	STM32L412xx, STM32L422xx, STM32L431xx, STM32L433xx, STM32L443xx, STM32L451xx, STM32L452xx, STM32L462xx
WLCSP36	-	-	-	-	-	X	(2.85 x 3.07)	STM32L412xx, STM32L422xx
WLCSP49	-	-	-	-	X	-	(3.141 x 3.127)	STM32L431xx, STM32L433xx, STM32L443xx
WLCSP64	-	-	-	-	X	-	(3.141 x 3.127)	STM32L431xx, STM32L433xx, STM32L443xx

Table 2. Packages available on STM32L4 Series and STM32L4+ Series (continued)

Package[1]	STM32L4+ Series	STM32L4 Series					Size (mm x mm)	Applicable part numbers
		STM32L 49xxx/ 4Axxx	STM32L 47xxx/ 48xxx	STM32L 45xxx/ 46xxx	STM32L 43xxx/ 44xxx	STM32L 41xxx/ 42xxx		
LQFP64	-	X	X	X	X	X	(10 x 10)	STM32L412xx, STM32L422xx, STM32L431xx, STM32L433xx, STM32L443xx, STM32L451xx, STM32L452xx, STM32L462xx, STM32L471xx, STM32L475xx, STM32L476xx, STM32L486xx, STM32L496xx, STM32L4A6xx
UFBGA64	-	-	-	X	X	X	(5 x 5)	STM32L412xx, STM32L422xx, STM32L431xx, STM32L433xx, STM32L443xx, STM32L451xx, STM32L452xx, STM32L462xx
WLCSP64	-	-	-	X	-	-	(3.357 x 3.657)	STM32L451xx, STM32L452xx, STM32L462xx
WLCSP72	-	-	X	-	-	-	(4.4084 x 3.7594)	STM32L471xx, STM32L475xx, STM32L476xx, STM32L486xx
WLCSP81	-	-	X	-	-	-	(4.4084 x 3.7594)	STM32L476xx
WLCSP100	-	X	-	-	-	-	(4.618 x 4.142)	STM32L496xx, STM32L4A6xx

Table 2. Packages available on STM32L4 Series and STM32L4+ Series (continued)

Package[1]	STM32L4+ Series	STM32L4 Series					Size (mm x mm)	Applicable part numbers
		STM32L 49xxx/ 4Axxx	STM32L 47xxx/ 48xxx	STM32L 45xxx/ 46xxx	STM32L 43xxx/ 44xxx	STM32L 41xxx/ 42xxx		
LQFP100	X	X	X	X	X	-	(14 x 14)	STM32L431xx, STM32L433xx, STM32L443xx, STM32L451xx, STM32L452xx, STM32L462xx, STM32L471xx, STM32L475xx, STM32L476xx, STM32L486xx, STM32L496xx, STM32L4A6xx, STM32L4R5xx, STM32L4R9xx, STM32L4S5xx, STM32L4S9xx
UFBGA100	-	-	X	X	X	-	(7 x 7)	STM32L431xx, STM32L433xx, STM32L443xx
UFBGA132	X	X	X	-	-	-	(7 x 7)	STM32L471xx, STM32L475xx, STM32L476xx, STM32L486xx, STM32L496xx, STM32L4A6xx, STM32L4R5xx, STM32L4S5xx
UFBGA144	X	-	-	-	-	-	(10 x 10)	STM32L4R9xx, STM32L4S9xx
LQFP144	X	X	X	-	-	-	(20 x 20)	STM32L471xx, STM32L475xx, STM32L476xx, STM32L486xx, STM32L496xx, STM32L4A6xx, STM32L4R5xx, STM32L4R9xx, STM32L4S5xx, STM32L4S9xx

Table 2. Packages available on STM32L4 Series and STM32L4+ Series (continued)

Package[1]	STM32L4+ Series	STM32L4 Series					Size (mm x mm)	Applicable part numbers
		STM32L 49xxx/ 4Axxx	STM32L 47xxx/ 48xxx	STM32L 45xxx/ 46xxx	STM32L 43xxx/ 44xxx	STM32L 41xxx/ 42xxx		
WLCSP144	X	-	-	-	-	-	(5.24 x 5.24)	STM32L4R5xx, STM32L4R7xx, STM32L4R9xx, STM32L4S5xx, STM32L4S7xx, STM32L4S9xx
UFBGA169	X	X	-	-	-	-	(7 x 7)	STM32L496xx, STM32L4A6xx, STM32L4R5xx, STM32L4R9xx, STM32L4S5xx, STM32L4S9xx

1. X = supported.

Table 3 and *Table 4* show the differences for packages available in both families. The other packages in STM32L1 Series are not available for STM32L4 Series / STM32L4+ Series.

STM32L4 Series / STM32L4+ Series and STM32L1 Series share a high level of pin compatibility. Most peripherals share the same pins in the two series.

The transition from STM32L1 Series to STM32L4 Series and STM32L4+ Series for the QFP and BGA packages is simple since only a few pins are different.

Table 3. Pinout differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (QFP)

STM32L1 Series					STM32L4 Series/ STM32L4+ Series				
QFP 48	QFP 64	QFP 100	QFP 144	Pinout	QFP 48	QFP 64	QFP 100	QFP 144	Pinout
1	1	6	6	VLCD	1	1	6	6	VBAT
-	-	-	95	VDD	-	-	-	95	VDDIO2(1)
-	-	-	131	VDD	-	-	-	131	VDDIO2(1)
-	-	73	106	PH2	-	-	73	106	VDDUSB(1)
36	48	-	-	VDD	36	48	-	-	VDDUSB(1)
44	60	94	138	BOOT0	44	60	94	138	PH3-BOOT0(2)

1. VDDIO2 and VDDUSB pins can be connected externally to VDD.
2. Only for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx.

Note: STM32L4R9xx/4S9xx devices are not compatible with STM32L4 Series, for more details refer to application note Migration between STM32L476xx/486xx and STM32L4+ Series (AN5017).

Table 4. Pinout differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (BGA)

STM32L1 Series				STM32L4 Series / STM32L4+ Series			
BGA64	BGA100	BGA132	Pinout	BGA64	BGA100	BGA132	Pinout
B2	E2	E2	VLCD	B2	E2	E2	VBAT
-	-	G7	VDD	-	-	G7	VDDIO2(1)
-	C11	C11	PH2	-	C11	C11	VDDUSB(1)
-	-	G3	PF6	-	-	G3	PG11
-	-	G4	PF7	-	-	G4	PG6
-	-	H4	PF8	-	-	H4	PG7
-	-	J6	PF9	-	-	J6	PG8
-	-	K1	NC	-	-	K1	PG15
G1	-	-	VREF+	G1	-	-	PC3
E5	-	-	VDD	E5	-	-	VDDUSB(1)
H1	-	-	VDDA	H1	-	-	VDDA / VREF+
B4	A4	A4	BOOT0	B4	A4	A4	PH3-BOOT0(2)

1. VDDIO2 and VDDUSB pins can be connected externally to VDD.
2. Only for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices.

Recommendations to migrate from the STM32L1 Series board to the STM32L4 Series and STM32L4+ Series board

The VLCD pin in STM32L4 Series / STM32L4+ Series is now multiplexed on the PC3 GPIO through alternate function programming. The PC3 GPIO is at pin 29 on QFP144, pin 18 on QFP100, pin 11 on QFP64 and pin K2 on BGA132/BGA100. The VLCD pin is not available in STM32L4+ Series.

This implies that any other function in the STM32L4 Series PC3 pins cannot be used on PC3 when the LCD is used by the application.

This also implies that the related STM32L1 Series PC3 alternate functions, if they are used by the application, must be mapped to other STM32L4 Series pins.

The V_{BAT} or V_{DD} supply (if no specific V_{BAT} power is used), must now be connected to:

- pin 6 (QFP144 and QFP100)
- pin 1 (QFP64/QFP48)
- pin E2 (BGA132/BGA100).
- pin B2 (BGA64)

The PH2 GPIO is now used for dedicated V_{DDUSB} power supply:

- pin 106 (QFP144) and pin 108 for STM32L49xx/4R9xx devices
- pin 73 (QFP100) and pin 75 for STM32L4R9xx/4S9xx devices
- pin C11 (BGA132)

The PH2 GPIO cannot be used as a regular GPIO any more (except on STM32L4+ Series and STM32L49xxx/4Axxx devices with BGA169 package). There is no PH2 GPIO on other STM32L4 Series / STM32L4+ Series devices.

Regarding the BGA132 package, several GPIOs from the STM32L1 Series are mapped on different GPIOs in the STM32L4 Series and STM32L4+ Series:

- PF6 (pin G3) mapped to PG11 on same pin
- PF7 (pin G4) mapped to PG6 on same pin
- PF8 (pin H4) mapped to PG7 on same pin
- PF9 (pin J6) mapped to PG8 on same pin

On the BGA64 package, the G1 ball used for VREF+ signal in the STM32L1 Series is used as PC3 GPIO (multiplexed with VLCD) in STM32L4 Series / STM32L4+ Series, and the VREF+ signal is multiplexed with VDDA on the H1 ball.

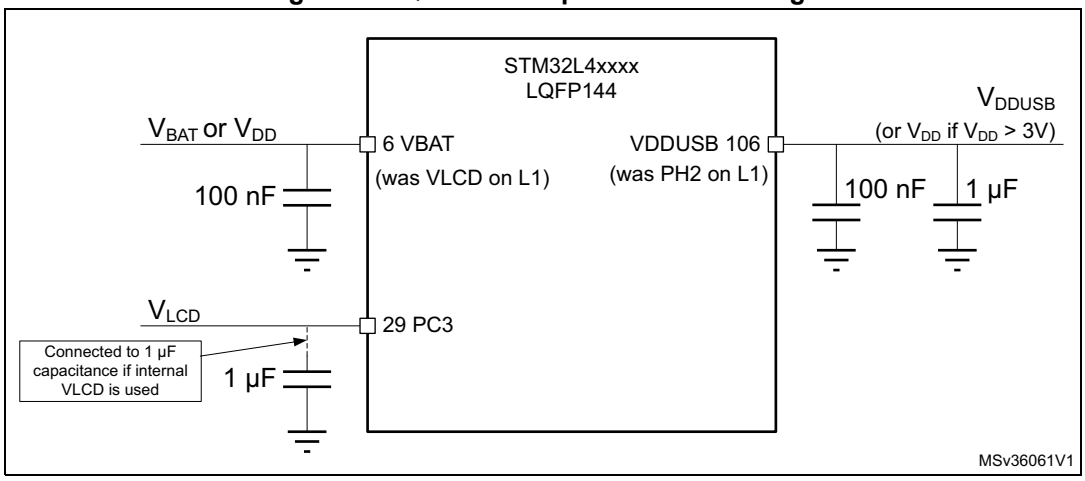
The not connected (NC) K1 pin in STM32L1 Series can be used as PG15 GPIO in STM32L4 Series / STM32L4+ Series.

The boot pins are different in both families. The BOOT0 is multiplexed with the PH3 GPIO on STM32L4 Series (for STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx) and STM32L4+ Series. Refer to *Section 3: Boot mode selection* for details. Those changes do not impact the board design.

Figure 1 to *Figure 7* show some examples of board designs migrating from the STM32L1 Series to the STM32L4 Series and STM32L4+ Series.

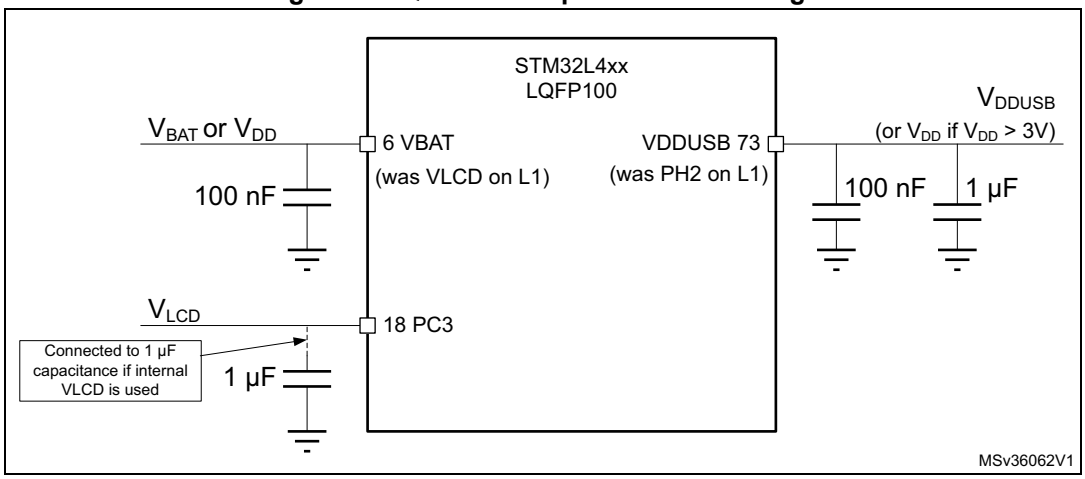
Note: VLCD is not available in STM32L4+ Series.

Figure 1. LQFP144 compatible board design



1. For SM32L4R9xx/4S9xx, the VDDUSB pin = 108.

Figure 2. LQFP100 compatible board design



1. For SM32L4R9xx/4S9xx, the VDUSB pin = 75.

Figure 3. LQFP64 compatible board design

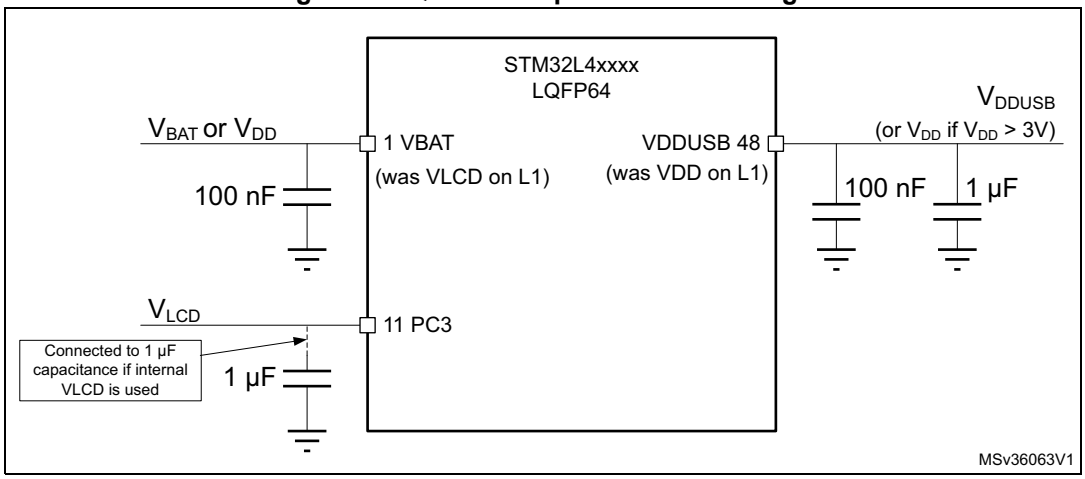


Figure 4. LQFP48 compatible board design

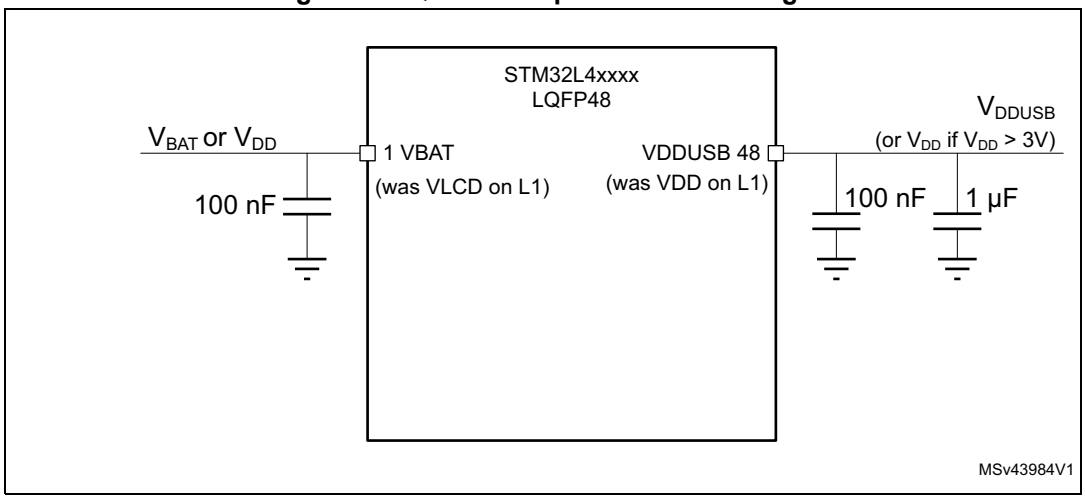


Figure 5. BGA132 compatible board design

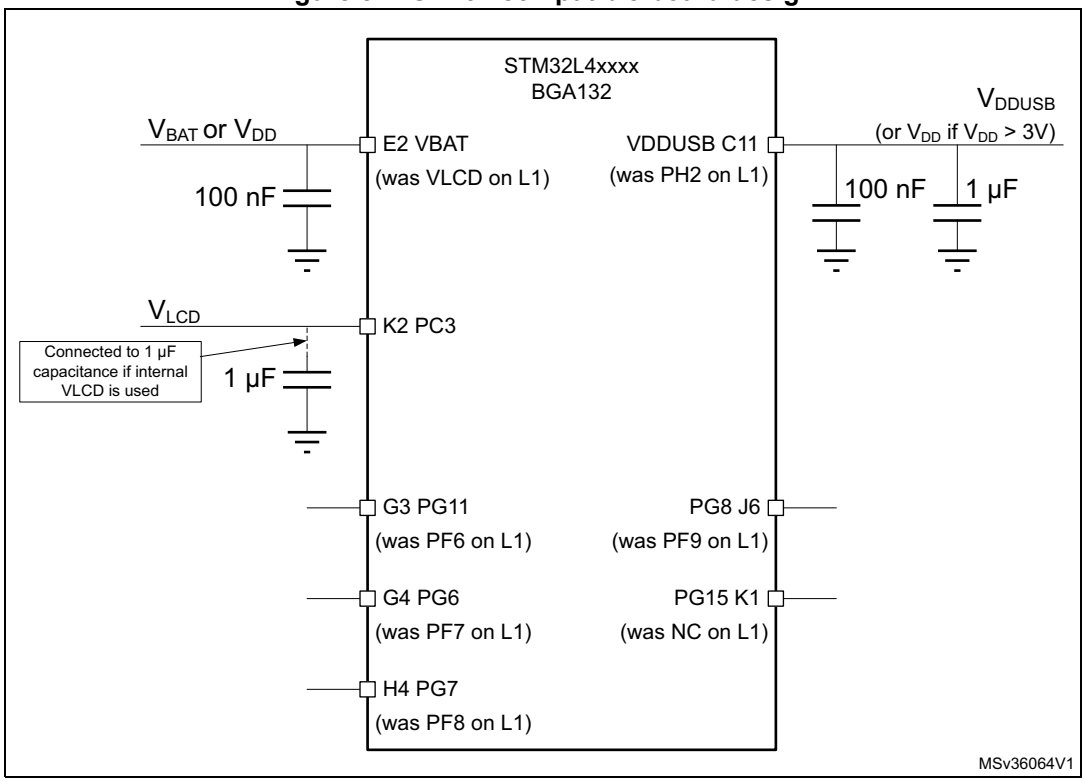


Figure 6. BGA100 compatible board design

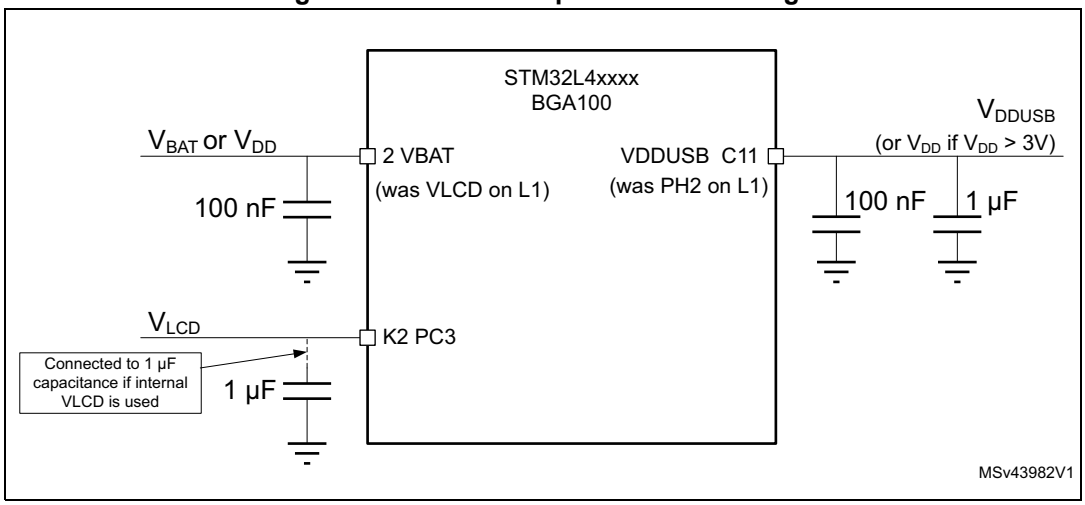
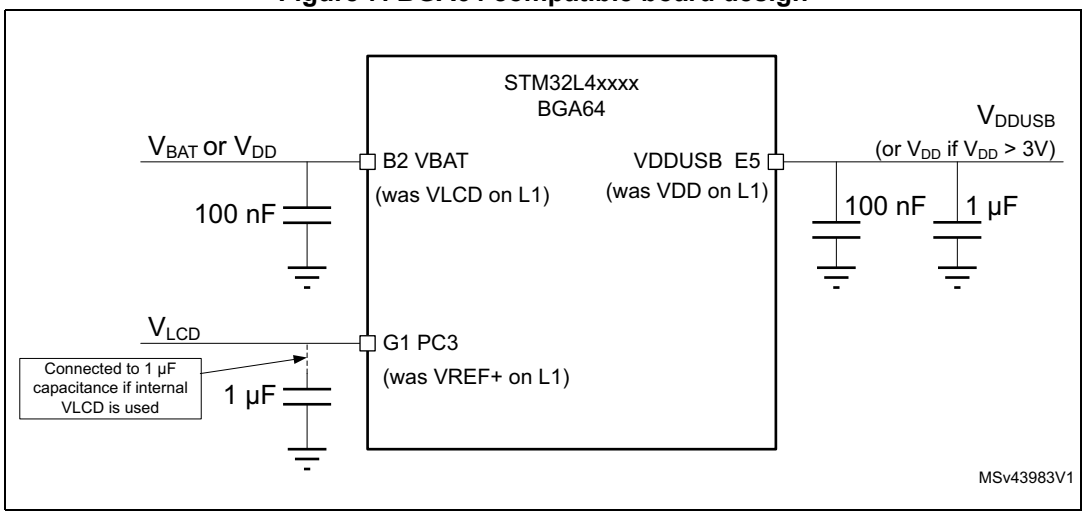


Figure 7. BGA64 compatible board design



SMPS packages

Some devices of STM32L4 Series and STM32L4+ Series offer a package option allowing the connection of an external SMPS.

This is done through two VDD12 pins that are replacing two existing pins in the baseline package.

Compatibility is kept between derivatives of STM32L4 Series and STM32L4+ Series regarding those two VDD12 pins (the pins replaced are different across package types but are the same for all derivatives on similar packages).

Refer to the product datasheets for more details.

3 Boot mode selection

Both STM32L1 Series and STM32L4 Series / STM32L4+ Series can select boot modes between three options: boot from main Flash memory, boot from SRAM or boot from system memory. However, the way to select the boot mode differs between the products.

In STM32L1 Series, the boot mode is selected with two pins: BOOT0 and BOOT1.

In STM32L47xxx/48xxx devices, the boot mode is selected with 1 pin (BOOT0) and with the nBOOT1 option bit located in the user option bytes at memory address 0x1FFF7800.

In STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices, the boot mode is selected with the nBOOT1 option bit and with the BOOT0 pin or the nBOOT0 option bit depending on the value of the nSWBOOT0 option bit in the FLASH_OPTR register.

Table 5 and *Table 6* summarize the different configurations available for selecting the boot mode for STM32L4 Series / STM32L4+ Series and STM32L1 Series.

Table 5. Boot modes for STM32L47xxx/48xxx devices and STM32L1 Series

Boot mode selection(1)		Boot mode	Aliasing
BOOT1(2)	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.

1. X = equivalent to 0 or 1.
2. The BOOT1 value is the opposite of the nBOOT1 option bit for STM32L47xxx/48xxx devices.

Table 6. Boot modes for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx,, STM32L43xxx/44xxx and STM43L41xxx/42xxx devices(1)

nBOOT1 FLASH_OPTR [23]	nBOOT0 FLASH_OPTR [27]	BOOT0 pin PH3	nSWBOOT0 FLASH_OPTR [26]	Main Flash empty(2)	Boot Memory Space Alias
X	X	0	1	0	Main Flash memory is selected as boot area
X	X	0	1	1	System memory is selected as boot area
X	1	X	0	X	Main Flash memory is selected as boot area
0	X	1	1	X	Embedded SRAM1 is selected as boot area
0	0	X	0	X	Embedded SRAM1 is selected as boot area
1	X	1	1	X	System memory is selected as boot area
1	0	X	0	X	System memory is selected as boot area

1. X = equivalent to 0 or 1.
2. For STM32L45xxx/46xxx and STM32L43xxx/44xxx devices, a Flash empty check mechanism is implemented to force the boot from system Flash if the first Flash memory location is not programmed (0xFFFF FFFF) and if the boot selection was configured to boot from the main Flash.

Embedded bootloader

The embedded bootloader is located in the system memory, programmed by ST during production. This bootloader is used to reprogram the Flash memory using one of the serial interfaces listed in *Table 7*.

Table 7. Bootloader interfaces on STM32L1 and STM32L4 Series / STM32L4+ Series

Peripheral[1]	Pin	STM32L1 Series	STM32L4 Series / STM32L4+ Series
DFU	USB_DM (PA11) USB_DP (PA12)	X	X
USART1	USART1_TX (PA9) USART1_RX (PA10)	X	X
USART2	USART2_TX (PD5) USART2_RX (PD6)	X	-
USART2	USART2_TX (PA2) USART2_RX (PA3)	-	X
USART3	USART3_TX (PC10) USART3_RX (PC11)	-	X
I2C1	I2C1_SCL (PB6) I2C1_SDA (PB7)	-	X
I2C2	I2C2_SCL (PB10) I2C2_SDA (PB11)	-	X
I2C3	I2C3_SCL (PC0) I2C3_SDA (PC1)	-	X
I2C4	I2C4_SCL (PD12) I2C4_SDA (PD13)	-	X[2]
SPI1	SPI1_NSS (PA4) SPI1_SCK (PA5) SPI1_MISO (PA6) SPI1_MOSI (PA7)	-	X
SPI2	SPI2_NSS (PB12) SPI2_SCK (PB13) SPI2_MISO (PB14) SPI2_MOSI (PB15)	-	X
CAN1	CAN1_RX (PB8) CAN1_TX (PB9)	-	X[3]
CAN2	CAN2_RX (PB5) CAN2_TX (PB6)	-	X[4]

1. X = supported.
2. Only for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L45xxx/46xxx devices.
3. Not available on STM32L41xxx/42xxx devices.
4. Only for STM32L49xxx/4Axxx devices.

Refer to application note *STM32 microcontroller system memory boot mode* (AN2606) for more details on the bootloader.

For smaller packages, it is important to check the pin and peripheral availability.

4 Peripheral migration

4.1 STM32 product cross-compatibility

The STM32 MCUs embed a set of peripherals that are classified in three groups:

- The first group is for the peripherals that are common to all products. Those peripherals have the same structure, registers and control bits. There is no need to perform any firmware change to keep the same functionality at the application level after migration. All the features and behavior remain the same.
- The second group is for the peripherals that present minor differences from one product to another (usually differences due to the support of new features). Migrating from one product to another is very easy and does not require any significant new development effort.
- The third group is for peripherals that have been considerably modified from one product to another (new architecture, new features...). For this group of peripherals, the migration requires a new development at application level.

Table 8 gives a general overview of this classification.

The software compatibility mentioned in *Table 8* refers only to the register description for low level drivers.

The STMCube™ hardware abstraction layer (HAL) is compatible between the STM32L1 Series and the STM32L4 Series / STM32L4+ Series.

Table 8. Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series

<table>
<tr><th rowspan="2">Peripheral</th><th colspan="7">Number of instances in STM32</th><th colspan="3">Compatibility with STM32L4 Series / STM32L4+ Series</th></tr>
<tr><th>L1 Series</th><th>L4+ Series</th><th>L49xxx /4Axxx</th><th>L47xxx /48xxx</th><th>L45xxx /46xxx</th><th>L43xxx /44xxx</th><th>L41xxx /42xxx</th><th>Software</th><th>Pinout</th><th>Comments</th></tr>
<tr><td>SPI</td><td>3</td><td colspan="4">3</td><td colspan="2">2</td><td rowspan="2">Partial</td><td rowspan="2">Partial</td><td rowspan="2">– I2S is no longer supported by SPI and is instead replaced by a dedicated serial audio interface (SAI) in STM32L4 Series / STM32L4+ Series
– Some alternate functions are not mapped on the same GPIO for SPI1</td></tr>
<tr><td>I2S (full duplex)</td><td>2</td><td colspan="6">0</td></tr>
<tr><td>WWDG</td><td>1</td><td colspan="6">1</td><td>Full</td><td rowspan="3">NA</td><td rowspan="3">-</td></tr>
<tr><td>IWDG</td><td>1</td><td colspan="6">1</td><td>Full</td></tr>
<tr><td>DBGMCU</td><td>1</td><td colspan="6">1</td><td>Full</td></tr>
</table>

Table 8. Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Peripheral	Number of instances in STM32							Compatibility with STM32L4 Series / STM32L4+ Series		
	L1 Series	L4+ Series	L49xxx /4Axxx	L47xxx /48xxx	L45xxx /46xxx	L43xxx /44xxx	L41xxx /42xxx	Software	Pinout	Comments
CRC	1	1						Partial	NA	Additional features in STM32L4 Series / STM32L4+ Series
EXTI	1	1						Partial	Full	– PH2 GPIO available only on STM32L4+ Series and STM32L49xxx/4Axxx for BGA169 package
USB FS	1	0			1					– Additional features on STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx
USB OTG FS	0	1			0			None		– New peripheral (OTG FS) on STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx
DMA	1	2						Full	NA	– Same features – DMA mapping request may differ (see *Section 4.3: Direct memory access controller (DMA)*)
TIM Basic General P. Advanced Low-power IRTIM	 2 7 0 0 0	 2 7 2 2 1	 2 7 2 7 1	 2 7 2 2 1	 2 4 1 2 1	 2 3 1 2 1	 1 3 1 2 1		Partial	– Some pins are not mapped on the same GPIO – Timer instances names may differ – Internal connections may differ
SDIO	1	1								Some pins not mapped on the same GPIO

Table 8. Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

<table>
<tr><th rowspan="2">Peripheral</th><th colspan="7">Number of instances in STM32</th><th colspan="3">Compatibility with STM32L4 Series / STM32L4+ Series</th></tr>
<tr><th>L1 Series</th><th>L4+ Series</th><th>L49xxx /4Axxx</th><th>L47xxx /48xxx</th><th>L45xxx /46xxx</th><th>L43xxx /44xxx</th><th>L41xxx /42xxx</th><th>Software</th><th>Pinout</th><th>Comments</th></tr>
<tr><td>FSMC/ FMC</td><td>1</td><td colspan="3">1</td><td colspan="3">0</td><td>Full</td><td>Full</td><td>– Only SRAM/NOR supported in STM32L1 Series
– NAND supported in STM32L4 Series</td></tr>
<tr><td>PWR</td><td>1</td><td colspan="6">1</td><td rowspan="4">Partial</td><td>NA</td><td>-</td></tr>
<tr><td>RCC</td><td>1</td><td colspan="6">1</td><td>NA</td><td>-</td></tr>
<tr><td>USART
UART
LPUART</td><td>3
2
0</td><td colspan="3">3
2
1</td><td>3
1
1</td><td colspan="2">3
0
1</td><td rowspan="3">Full</td><td>– Additional features in STM32L4 Series / STM32L4+ Series
– Additional LPUART in STM32L4 Series / STM32L4+ Series</td></tr>
<tr><td>I2C</td><td>2</td><td colspan="2">4</td><td>3</td><td>4</td><td colspan="2">3</td><td>None</td><td>Additional features in STM32L4 Series / STM32L4+ Series</td></tr>
<tr><td>DAC channels</td><td>2</td><td colspan="3">2</td><td>1</td><td>2</td><td>0</td><td>Partial</td><td>– Additional features in STM32L4 Series / STM32L4+ Series
– SW compatible except for output buffer management</td></tr>
<tr><td>ADC</td><td>1</td><td>1</td><td colspan="2">3</td><td colspan="2">1</td><td>2</td><td>None</td><td>Partial</td><td>– Additional features in STM32L4 Series / STM32L4+ Series
– Some pins mapped on different GPIOs</td></tr>
<tr><td>RTC</td><td>1</td><td colspan="6">1</td><td>Partial</td><td>Full</td><td>– Additional features in STM32L4 Series / STM32L4+ Series
– Can be powered by VBAT in STM32L4 Series / STM32L4+ Series</td></tr>
<tr><td>FLASH</td><td>1</td><td>1</td><td colspan="2">2</td><td colspan="3">1</td><td>None</td><td>NA</td><td>New peripheral</td></tr>
</table>

Table 8. Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

<table>
<tr><th rowspan="2">Peripheral</th><th colspan="7">Number of instances in STM32</th><th colspan="3">Compatibility with STM32L4 Series / STM32L4+ Series</th></tr>
<tr><th>L1 Series</th><th>L4+ Series</th><th>L49xxx /4Axxx</th><th>L47xxx /48xxx</th><th>L45xxx /46xxx</th><th>L43xxx /44xxx</th><th>L41xxx /42xxx</th><th>Software</th><th>Pinout</th><th>Comments</th></tr>
<tr><td>GPIO</td><td>Up to 115 IOs</td><td>Up to 140</td><td>Up to 136</td><td>Up to 114</td><td>Up to 83</td><td>Up to 83</td><td>Up to 52</td><td rowspan="3">Full</td><td rowspan="4">Partial</td><td>– At Reset, STM32L1 Series configured in Input-floating mode while STM32L4 Series / STM32L4+ Series are configured in Analog mode
– A few changes mentioned in Section 2: Hardware migration</td></tr>
<tr><td>LCD glass</td><td>1</td><td>0</td><td colspan="2">1</td><td>0</td><td>1</td><td>0</td><td>– VLCD muxed on PC3 GPIO
– SEG21 not mapped on same GPIO
– Additional features in STM32L4 Series / STM32L4+ Series
– LCD is not available in STM32L4+ Series</td></tr>
<tr><td>COMP</td><td>2</td><td colspan="5">2</td><td>1</td><td>None</td><td>Some pins mapped on different GPIOs</td></tr>
<tr><td>SYSCFG</td><td>1</td><td colspan="6">1</td><td>Partial</td><td rowspan="2">NA</td><td>-</td></tr>
<tr><td>AES</td><td>1</td><td colspan="6">1</td><td>Full</td><td>Additional features on STM32L4Axxx, STM32L48xxx, STM32L46xxx, STM32L44xxx, STM32L42xxx devices only</td></tr>
</table>

Table 8. Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Peripheral	Number of instances in STM32							Compatibility with STM32L4 Series / STM32L4+ Series		
	L1 Series	L4+ Series	L49xxx /4Axxx	L47xxx /48xxx	L45xxx /46xxx	L43xxx /44xxx	L41xxx /42xxx	Software	Pinout	Comments
OPAMP	3	2			1 None				Partial	– Some pins not mapped on same GPIO – One less OPAMP in STM32L4 Series / STM32L4+ Series

Color key:

= No compatibility (new feature or new architecture)

= Partial compatibility (minor changes)

= Not applicable

4.2 Memory mapping

The peripheral address mapping has been changed in STM32L4 Series / STM32L4+ Series compared to STM32L1 Series.

Table 9 provides the peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series.

Table 9. Peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

Peripheral	STM32L1 Series		STM32L4 Series / STM32L4+ Series	
	Bus	Base address	Bus	Base address(1)
FSMC/FMC	AHB	0xA0000000	AHB3	0xA0000000
AES		0x50060000	AHB2	0x50060000
DMA2		0x40026400	AHB1	0x40020400
DMA1		0x40026000		0x40020000
Flash memory interface		0x40023C00		0x40022000
RCC		0x40023800		0x40021000
CRC		0x40023000		0x40023000
GPIOG		0x40021C00	AHB2	0x48001800
GPIOF		0x40021800		0x48001400
GPIOH		0x40021400		0x48001C00
GPIOE		0x40021000		0x48001000
GPIOD		0x40020C00		0x48000C00
GPIOC		0x40020800		0x48000800
GPIOB		0x40020400		0x48000400
GPIOA		0x40020000		0x48000000
USART1	APB2	0x40013800	APB2	0x40013800
SP1		0x40013000		0x40013000
SDIO (SDMMC in STM32L4 Series)		0x40012C00		0x40012800 0x50062400 (AHB2) on STM32L4+ Series
ADC1 (ADC123 in STM32L4 Series)		0x40012400	AHB2	0x50040000
TIM11		0x40011000	NA	
TIM10		0x40010C00		
TIM9		0x40010800		
EXTI		0x40010400	APB2	0x40010400
SYSCFG		0x40010000		0x40010000

Table 9. Peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Peripheral	STM32L1 Series		STM32L4 Series / STM32L4+ Series	
	Bus	Base address	Bus	Base address(1)
COMP	APB1	0x40007C00	APB2	0x40010200
RI		0x40007C04	NA	
OPAMP	APB1	0x40007C5C	APB1	0x40007800
DAC		0x40007400		0x40007400
PWR		0x40007000		0x40007000
USB device FS SRAM		0x40006000	APB1	0x40006C00
USB device FS		0x40005C00	APB1	0x40006800
I2C2		0x40005800	APB1	0x40005800
I2C1		0x40005400		0x40005400
USART5 (UART5 in STM32L4 Series)		0x40005000		0x40005000
USART4 (UART4 in STM32L4 Series)		0x40004C00		0x40004C00
USART3		0x40004800		0x40004800
USART2		0x40004400		0x40004400
SPI3		0x40003C00		0x40003C00
SPI2		0x40003800		0x40003800
IWDG		0x40003000		0x40003000
WWDG		0x40002C00		0x40002C00
RTC (inc. BKP registers)		0x40002800		0x40002800
LCD		0x40002400		0x40002400
TIM7		0x40001400		0x40001400
TIM6		0x40001000		0x40001000
TIM5		0x40000C00		0x40000C00
TIM4		0x40000800		0x40000800
TIM3		0x40000400		0x40000400
TIM2		0x40000000		0x40000000

Table 9. Peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Peripheral	STM32L1 Series		STM32L4 Series / STM32L4+ Series	
	Bus	Base address	Bus	Base address(1)
RNG	NA		AHB2	0x50060800
HASH				0x50060400
DCMI				0x50050000
GPIOI				0x48002000
USB OTG FS				0x50000000
DMA2D			AHB1	0x4002B000
TSC				0x40024000
DFSDM			APB2	0x40016000
SAI2				0x40015800
SAI1				0x40015400
TIM17				0x40014800
TIM16				0x40014400
TIM15				0x40014000
TIM8				0x40013400
TIM1				0x40012C00
FIREWALL				0x40011C00
VREF				0x40010030
LPTIM2			APB1	0x40009400
SWPMI1				0x40008800
I2C4				0x40008400
LPUART1				0x40008000
LPTIM1				0x40007C00
CAN2				0x40006800
CAN1				0x40006400
CRS				0x40006000
I2C3				0x40005C00

Table 9. Peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Peripheral	STM32L1 Series		STM32L4 Series / STM32L4+ Series	
	Bus	Base address	Bus	Base address(1)
QUADSPI	NA		AHB3(2) AHB4	0xA0001000
OCTOSPI2			AHB3	0xA000 1400
OCTOSPI1				0xA000 1000
OCTOSPIM			AHB2	0x5006 1C00
GFXMMU			AHB1	0x4002 C000
DMAMUX1				0x4002 0800
DSIHOST			APB2	0x4001 6C00
LCD-TFT				0x4001 6800
Color key: = Base address or bus change; = Not applicable				

1. On STM32L4 Series / STM32L4+ Series devices on which the peripheral is not implemented, the memory address is reserved.
2. AHB3 for STM32L47xxx/48xxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices, AHB4 for STM32L49xxx/4Axxx devices.

The system memory mapping has been updated between STM32L1 Series and STM32L4 Series / STM32L4+ Series. Refer to reference manual or datasheet for more details.

STM32L4 Series and STM32L4+ Series feature an additional SRAM (SRAM2) of the following size:

- 64 Kbytes on STM32L4+ Series and STM32L49xxx/4Axxx devices
- 32 Kbytes on STM32L47xxx/48xxx and STM32L45xxx/46xxx devices
- 16 Kbytes on STM32L43xxx/44xxx devices
- 8 Kbytes on STM32L41xxx/42xxx devices

An additional SRAM (SRAM3) of 384 Kbytes is available only in STM32L4+ Series.

The SRAM2 includes the additional features listed below:

- Maximum performance through ICode bus access without physical remap
- Parity check option (32-bit + 4-bit parity check)
- Write protection with 1 Kbyte granularity
- Read protection (RDP)
- Erase by system reset (option byte) or by software
- Content preserved in Low-power run, Low-power sleep, Stop 0, Stop 1, Stop 2 modes
- Content can be preserved (RRS bit set in PWR_CR3 register) in Standby mode (not the case for SRAM1).

Bit-banding

Both STM32L1 Series and STM32L4 Series / STM32L4+ Series support bit-banding on the lowest 1 Mbyte of the SRAM and on the peripheral memory region. However, the peripherals mapped in this bit-banding region are not the same on each series of products.

Detail of the peripherals that are accessible with bit-banding:

- STM32L1 Series: all peripherals except AES and FSMC
- STM32L4 Series: all peripherals except AHB2, AHB3, AHB4, GPIOx, ADC, AES, RNG, FMC, QUADSPI, OTG_FS, DCMI and HASH.

4.3 Direct memory access controller (DMA)

STM32L1 Series and STM32L4 Series / STM32L4+ Series have the same DMA. Both series of products embed two DMA controllers with up to 7 + 5 channels for STM32L1 Series and with 7 + 7 channels for STM32L4 Series. Each channel is dedicated to manage the memory access requests from one or more peripherals. They have an arbiter for handling the priorities among the DMA requests.

For STM32L4+ Series, each DMA request line is connected in parallel to all the channels of the DMAMUX request line multiplexer. In the rest of STM32L4 Series, the DMA request line is connected directly to the peripherals.

The DMAMUX request multiplexer allows a DMA request line to be routed between the peripherals and the DMA controllers of the product. The routine function is ensured by a programmable multi-channel DMA request line multiplexer. Each channel selects a unique DMA request line, unconditionally or synchronously with events from its DMAMUX synchronization inputs.

Table 10 presents the differences between the DMA requests of the peripherals in STM32L1 Series and the peripherals in STM32L4 Series / STM32L4+ Series.

Table 10. DMA request differences migrating from STM32L1 Series to STM32L4 Series / STM32L4+ Series

Peripheral	DMA request	STM32L1 Series	STM32L4 Series / STM32L4+ Series(1)
ADC	**ADC1**	DMA1_Channel1	DMA1_Channel1 DMA2_Channel3
	ADC2	NA	DMA1_Channel2 DMA2_Channel4
	ADC3		DMA1_Channel3 DMA2_Channel5
DAC	**DAC1_CH1** **DAC1_CH2**	DMA1_Channel2 (2) DMA1_Channel3 (2)	NA
	DAC1_CH1	NA	DMA1_Channel3 DMA2_Channel4
	DAC1_CH2		DMA1_Channel4 DMA2_Channel5

Table 10. DMA request differences migrating from STM32L1 Series to STM32L4 Series / STM32L4+ Series (continued)

Peripheral	DMA request	STM32L1 Series	STM32L4 Series / STM32L4+ Series(1)
DFSDM	DFSDM0	NA	DMA1_Channel4
	DFSDM1		DMA1_Channel5
	DFSDM2		DMA1_Channel6
	DFSDM3		DMA1_Channel7
SPI1	SPI1_Rx	DMA1_Channel2	DMA1_Channel2 DMA2_Channel3
	SPI1_Tx	DMA1_Channel3	DMA1_Channel3 DMA2_Channel4
SPI2	SPI2_Rx SPI2_Tx	DMA1_Channel4 DMA1_Channel5	DMA1_Channel4 DMA1_Channel5
SPI3	SPI3_Rx SPI3_Tx	DMA2_Channel1 DMA2_Channel2	DMA2_Channel1 DMA2_Channel2
QUADSPI	QUADSPI	NA	DMA2_Channel7 DMA1_Channel5
USART1	USART1_Rx	DMA1_Channel5	DMA1_Channel5 DMA2_Channel7
	USART1_Tx	DMA1_Channel4	DMA1_Channel4 DMA2_Channel6
USART2	USART2_Rx USART2_Tx	DMA1_Channel6 DMA1_Channel7	DMA1_Channel6 DMA1_Channel7
USART3	USART3_Rx USART3_Tx	DMA1_Channel3 DMA1_Channel2	DMA1_Channel3 DMA1_Channel2
UART4	UART4_Rx UART4_Tx	DMA2_Channel3 DMA2_Channel5	DMA2_Channel5 DMA2_Channel3
UART5	UART5_Rx UART5_Tx	DMA2_Channel2 DMA2_Channel1	DMA2_Channel2 DMA2_Channel1
LPUART	LPUART_RX LPUART_TX	NA	DMA2_Channel7 DMA2_Channel6
I2C1	I2C1_Rx	DMA1_Channel7	DMA1_Channel7 DMA2_Channel6
	I2C1_Tx	DMA1_Channel6	DMA1_Channel6 DMA2_Channel7
I2C2	I2C2_Rx I2C2_Tx	DMA1_Channel5 DMA1_Channel4	DMA1_Channel5 DMA1_Channel4
I2C3	I2C3_Rx I2C3_Tx	NA	DMA1_Channel3 DMA1_Channel2
I2C4	I2C4_Rx I2C4_Tx		DMA2_Channel1 DMA2_Channel2

Table 10. DMA request differences migrating from STM32L1 Series to STM32L4 Series / STM32L4+ Series (continued)

Peripheral	DMA request	STM32L1 Series	STM32L4 Series / STM32L4+ Series(1)
SDIO SDMMC	**SDIO**	DMA2_Channel4	NA
	SDMMC	NA	DMA2_Channel4 DMA2_Channel5
TIM1	**TIM1_CH1** **TIM1_CH2** **TIM1_CH4** **TIM1_TRIG** **TIM1_COM** **TIM1_UP** **TIM1_CH3**		DMA1_Channel2 DMA1_Channel3 DMA1_Channel4 DMA1_Channel4 DMA1_Channel4 DMA1_Channel6 DMA1_Channel7
TIM2	**TIM2_UP** **TIM2_CH1** **TIM2_CH2** **TIM2_CH3** **TIM2_CH4**	DMA1_Channel2 DMA1_Channel5 DMA1_Channel7 DMA1_Channel1 DMA1_Channel7	DMA1_Channel2 DMA1_Channel5 DMA1_Channel7 DMA1_Channel1 DMA1_Channel7
TIM3	**TIM3_UP** **TIM3_CH1** **TIM3_TRIG** **TIM3_CH3** **TIM3_CH4**	DMA1_Channel3 DMA1_Channel6 DMA1_Channel6 DMA1_Channel2 DMA1_Channel3	DMA1_Channel3 DMA1_Channel6 DMA1_Channel6 DMA1_Channel2 DMA1_Channel3
TIM4	**TIM4_UP** **TIM4_CH1** **TIM4_CH2** **TIM4_CH3**	DMA1_Channel7 DMA1_Channel1 DMA1_Channel4 DMA1_Channel5	DMA1_Channel7 DMA1_Channel1 DMA1_Channel4 DMA1_Channel5
TIM5	**TIM5_UP** **TIM5_CH1** **TIM5_CH2** **TIM5_CH3** **TIM5_CH4** **TIM5_TRIG** **TIM5_COM**	DMA2_Channel2 DMA2_Channel5 DMA2_Channel4 DMA2_Channel2 DMA2_Channel1 DMA2_Channel1 DMA2_Channel1	DMA2_Channel2 DMA2_Channel5 DMA2_Channel4 DMA2_Channel2 DMA2_Channel1 DMA2_Channel1 DMA2_Channel1
TIM6	**TIM6_UP**	DMA1_Channel2	DMA1_Channel3 DMA2_Channel4
TIM7	**TIM7_UP**	DMA1_Channel3	DMA1_Channel4 DMA2_Channel5

Table 10. DMA request differences migrating from STM32L1 Series to STM32L4 Series / STM32L4+ Series (continued)

Peripheral	DMA request	STM32L1 Series	STM32L4 Series / STM32L4+ Series(1)
TIM8	**TIM8_CH1** **TIM8_CH2** **TIM8_CH3** **TIM8_UP** **TIM8_CH4** **TIM8_TRIG** **TIM8_COM**	NA	DMA2_Channel6 DMA2_Channel7 DMA2_Channel1 DMA2_Channel1 DMA2_Channel2 DMA2_Channel2 DMA2_Channel2
TIM15	**TIM15_CH1** **TIM15_UP** **TIM15_TRIG** **TIM15_COM**		DMA1_Channel5 DMA1_Channel5 DMA1_Channel5 DMA1_Channel5
TIM16	**TIM16_CH1** **TIM16_UP** **TIM16_CH1** **TIM16_UP**		DMA1_Channel3 DMA1_Channel3 DMA1_Channel6 DMA1_Channel6
TIM17	**TIM17_CH1** **TIM17_UP** **TIM17_CH1** **TIM17_UP**		DMA1_Channel1 DMA1_Channel1 DMA1_Channel7 DMA1_Channel7
SAI	**SAI1_A**	NA	DMA2_Channel1 DMA2_Channel6
	SAI1_B		DMA2_Channel2 DMA2_Channel7
	SAI2_A		DMA1_Channel6 DMA2_Channel3
	SAI2_B		DMA1_Channel7 DMA2_Channel4
SWPMI	**SWPMI_RX**		DMA2_Channel1
	SWPMI_TX		DMA2_Channel2
AES	**AES_OUT**	DMA2_Channel3	DMA2_Channel3 DMA2_Channel2
	AES_IN	DMA2_Channel5	DMA2_Channel5 DMA2_Channel1
DCMI	**DCMI**	NA	DMA2_Channel7 DMA2_Channel5
HASH	**HASH_IN**		DMA2_Channel7
Color key: = Feature not available (NA) = Differences			

1. On the STM32L4 Series and STM32L4+ Series devices on which the peripheral is not implemented, the DMA request is reserved.
2. For the high-density value line devices, the DAC DMA requests are mapped respectively on DMA1_Channel 3 and DMA1_Channel 4.

4.4 Interrupts

Table 11 presents the interrupt vectors in STM32L4 Series / STM32L4+ Series compared to STM32L1 Series.

Table 11. Interrupt vector differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

Position	STM32L1 Series(1)			STM32L4 Series / STM32L4+ Series(2)
	Cat.1 and Cat.2	Cat.3	Cat.4 and Cat.5	
0	WWDG			WWDG
1	PVD			PVD / PVM
2	TAMPER_ STAMP			TAMPER / CSS
3	RTC_WKUP			RTC_WKUP
4	FLASH			FLASH
5	RCC			RCC
6	EXTI0			EXTI0
7	EXTI1			EXTI1
8	EXTI2			EXTI2
9	EXTI3			EXTI3
10	EXTI4			EXTI4
11	DMA1_Channel1			DMA1_Channel1
12	DMA1_Channel2			DMA1_Channel2
13	DMA1_Channel3			DMA1_Channel3
14	DMA1_Channel4			DMA1_Channel4
15	DMA1_Channel5			DMA1_Channel5
16	DMA1_Channel6			DMA1_Channel6
17	DMA1_Channel7			DMA1_Channel7
18	ADC1			ADC1_2
19	USB_HP			CAN1_TX
20	USB_LP			CAN1_RX0
21	DAC			CAN1_RX1
22	COMP, TSC (3)	COMP/CA		CAN1_SCE
23	EXTI9_5			EXTI9_5
24	LCD			TIM1_BRK / TIM15
25	TIM9			TIM1_UP / TIM16
26	TIM10			TIM1_TRG_COM / TIM17
27	TIM11			TIM1_CC
28	TIM2			TIM2

Table 11. Interrupt vector differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Position	STM32L1 Series[1]			STM32L4 Series / STM32L4+ Series[2]
	Cat.1 and Cat.2	Cat.3	Cat.4 and Cat.5	
29	TIM3			TIM3
30	TIM4			TIM4
31	I2C1_EV			I2C1_EV
32	I2C1_ER			I2C1_ER
33	I2C2_EV			I2C2_EV
34	I2C2_ER			I2C2_ER
35	SPI1			SPI1
36	SPI2			SPI2
37	USART1			USART1
38	USART2			USART2
39	USART3			USART3
40	EXTI15_10			EXTI15_10
41	RTC_Alarm			RTC_Alarm
42	USB_FS_WKUP			DFSDM3
43	TIM6			TIM8_BRK
44	TIM7			TIM8_UP
45	NA	TIM5	SDIO	TIM8_TRG_COM
46		SPI3	TIM5	TIM8_CC
47		DMA2_Channel1	SPI3	ADC3
48		DMA2_Channel2	UART4	FMC
49		DMA2_Channel3	UART5	SDMMC
50		DMA2_Channel4	DMA2_Channel1	TIM5
51		DMA2_Channel5	DMA2_Channel2	SPI3
52		AES	DMA2_Channel3	UART4
53		COMP_ACQ	DMA2_Channel4	UART5
54	NA		DMA2_Channel5	TIM6_DACUNDER
55			AES	TIM7
56			COMP_ACQ	DMA2_Channel1
57	NA			DMA2_Channel2
58				DMA2_Channel3
59				DMA2_Channel4
60				DMA2_Channel5
61				DFSDM0

Table 11. Interrupt vector differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Position	STM32L1 Series(1)			STM32L4 Series / STM32L4+ Series(2)
	Cat.1 and Cat.2	Cat.3	Cat.4 and Cat.5	
62	NA			DFSDM1
63				DFSDM2
64				COMP
65				LPTIM1
66				LPTIM2
67				– OTG_FS (STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx devices) – USB_FS (STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices)
68				DMA2_CH6
69				DMA2_CH7
70				LPUART1
71				– QUADSPI – OCTOSPI 1 for STM32L4+ Series
72				I2C3_EV
73				I2C3_ER
74				SAI1
75				SAI2
76				– SWPMI1 – OCTOSPI2 for STM32L4+ Series
77				TSC
78				– LCD – DSIHOST for STM32L4R9xx/4S9xx
79				AES
80				RNG
81				FPU
82				HASH and CRS
83				I2C4_EV
84				I2C4_ER
85				DCMI

Table 11. Interrupt vector differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Position	STM32L1 Series[1]			STM32L4 Series / STM32L4+ Series[2]
	Cat.1 and Cat.2	Cat.3	Cat.4 and Cat.5	
86	NA			CAN2_TX
87				CAN2_RX0
88				CAN2_RX1
89				CAN2_SCE
90				DMA2D
91				LCD-TFT
92				LCD-TFT_ER
93				GFXMMU
94				DMAMUX1_OVR

Color key:

= Interrupt vector name changed but STM32L4 Series / STM21L4+ Series peripheral still mapped on the same interrupt vector position than in STM32L1 Series

= Feature not available (NA)

= Differences

1. The categories definitions for the STM32L1 Series are available in the reference manual RM0038.
2. On the STM32L4 Series and STM32L4+ Series devices on which the peripheral is not implemented, the DMA request is reserved.
3. Depending on the STM32L1 product line used.

4.5 Reset and clock control (RCC)

The main differences related to RCC (reset and clock controller) between STM32L4 Series / STM32L4+ Series and STM32L1 Series, are presented in *Table 12*.

Table 12. RCC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

RCC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
MSI	Multi speed RC factory and user trimmed (64 kHz, 128 kHz, 256 kHz, 512 kHz, 1.02 MHz, 2.05 MHz, 4.1 MHz)	– MSI is a low-power oscillator with programmable frequency up to 48 MHz. It can replace the PLL as system clock (faster wakeup, lower consumption). It can be used as USB device clock (no need for external high-speed crystal oscillator) – Multi speed RC factory and user trimmed (100 kHz, 200 kHz, 400 kHz, 800 kHz, 1 MHz, 2 MHz, 4 MHz (default value), 8 MHz, 16 MHz, 24 MHz, 32 MHz and 48 MHz) – Auto calibration from LSE
HSI16	16 MHz RC factory and user trimmed	
LSI	37 kHz RC	– 32 kHz RC – Lower consumption, higher accuracy (refer to product datasheet)
HSE	1 to 24 MHz	4 to 48 MHz
LSE	32.768 kHz	– 32.768 kHz – Configurable drive/consumption – Available in backup domain (VBAT)
HSI48	NA	– 48 MHz RC (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx) – Can drive USB Full Speed, SDMMC and RNG
PLL	– Main PLL for system – PLL clock sources: HSI, HSE	– Main PLL for system x2 PLLs for SAI1/2, ADC, RNG, SDMMC and OTG FS clock. (for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx) x1 PLL for SAI1, ADC, RNG, SDMMC, USB FS clock (for STM32L45xxx/46xxx and STM32L43xxx/44xxx) – Each PLL provides up to 3 independent outputs – The PLL sources are MSI, HSI16, HSE

Table 12. RCC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

RCC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
System clock source	MSI, HSI16, HSE or PLL	
System clock frequency	– Up to 32 MHz – 2 MHz after reset using MSI	– Up to 80 MHz / or 120 MHz for STM32L4+ Series – 4 MHz after reset using MSI
AHB frequency	Up to 32 MHz	Up to 80 MHz (or up to 120 MHz for STM32L4+ Series)
APB1 frequency	Up to 32 MHz	Up to 80 MHz (or 120 MHz for STM32L4+ Series)
APB2 frequency	Up to 32 MHz	Up to 80 MHz (or 120 MHz for STM32L4+ Series)
RTC clock source	LSI, LSE or HSE clock divided by 2, 4, 8 or 16	LSI, LSE or HSE/32
MCO clock source	– MCO pin (PA8): SYSCLK, HSI16, HSE, PLLCLK, MSI, LSE, LSI or HSI48 (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L44xxx/43xxx and STM32L41xxx/42xxx) With configurable prescaler, 1, 2, 4, 8 or 16 for each output	
CSS	– CSS (clock security system) – CSS on LSE	
Internal oscillator measurement / calibration	– LSE connected to TIM9 or TIM10 CH1 IC: can measure HSI or MSI with respect to LSE clock high precision – LSI connected to TIM10 CH1 IC: can measure LSI with respect to HSI or HSE clock precision – HSE connected to TIM11 CH1 IC: can measure HSE with respect to LSE/HSI clock – MSI connected to TIM11 CH1 IC: can measure MSI with respect to HSI/HSE clock	– Mainly replacing TIM9/10/11 in STM32L1 Series by TIM15/16/17 in STM32L4 Series – LSE connected to TIM15 or TIM16 CH1 IC: can measure HSI16 or MSI with respect to LSE clock high precision – LSI connected to TIM16 CH1 IC: can measure LSI with respect to HSI16 or HSE clock precision – HSE/32 connected to TIM17 CH1 IC: can measure HSE with respect to LSE/HSI16 clock – MSI connected to TIM17 CH1 IC: can measure MSI with respect to HSI16/HSE clock – On STM32L45xxx/46xxx and STM32L43xxx/44xxx devices: HSE/32 and MSI connected to TIM16 CH1 IC

Table 12. RCC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

RCC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Interrupt	– CSS (linked to NMI IRQ) – LSECSS – LSIRDY, LSERDY, HSIRDY, MSIRDY, HSERDY, PLLRDY (linked to RCC global IRQ)	– CSS (linked to NMI IRQ) – LSECSS – LSIRDY, LSERDY, HSIRDY, MSIRDY, HSERDY, PLLRDY, PLLSAI1RDY and PLLSAI2RDY (only on STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx) (linked to RCC global IRQ)
Color key: = New feature or new architecture = Same feature, but specification change or enhancement		

In addition to the differences described in *Table 12*, the following additional adaptation steps may be needed for the migration:

- Performance versus V_{CORE} ranges
- Peripheral access configuration
- Peripheral clock configuration.

4.5.1 Performance versus V_{CORE} ranges

In STM32L1 Series, the maximum CPU clock frequency and the Flash memory wait-state depend on the selected V_{CORE} voltage range. but also on the selected V_{DD} range.

Table 13 presents the different clock source frequencies depending on different product voltage range.

Table 13. Performance versus V_{CORE} ranges for STM32L1 Series and STM32L4 Series / STM32L4+ Series(1)

CPU performance	Power performance	V_{CORE} range	Typical Value (V)	Max frequency (MHz)						V_{DD} range
				5 WS	4 WS	3 WS	2 WS	1 WS	0 WS	
STM32L1 Series										
High	Low	1	1.8	-	-	-	-	32	16	2.0 to 3.6
Medium	Medium	2	1.5	-	-	-	-	16	8	1.65 to 3.6
Low	High	3	1.2	-	-	-	-	4	2	
STM32L4 Series										
High	Medium	1	1.2	-	80	64	48	32	16	NA
Medium	High	2	1.0	-	26	26	18	12	6	NA
STM32L4+ Series										
High	Medium	1 (boost mode)	1.28	120	100	80	60	40	20	NA
		1 (normal mode)	1.2	-	-	40	60	40	20	NA
Medium	High	2	1.0	-	-	-	26	16	8	NA

1. WS = wait state.

4.5.2 Peripheral access configuration

Since the address mapping of some peripherals has been changed in STM32L4 Series / STM32L4+ Series compared to STM32L1 Series, different registers need to be used to [enable/disable] or [enter/exit] the peripheral [clock] or [from Reset mode]. See *Table 14*.

Table 14. RCC registers used for peripheral access configuration for STM32L1 Series and STM32L4 Series / STM32L4+ Series

Bus	Register STM32L1 Series	Register STM32L4 Series	Comments
AHB	RCC_AHBRSTR	RCC_AHB1RSTR (AHB1) RCC_AHB2RSTR (AHB2) RCC_AHB3RSTR (AHB3) (1)	Used to [enter/exit] the AHB peripheral from reset
	RCC_AHBENR	RCC_AHB1ENR (AHB1) RCC_AHB2ENR (AHB2) RCC_AHB3ENR (AHB3) (1)	Used to [enable/disable] the AHB peripheral clock
	RCC_AHBLPENR	RCC_AHB1SMENR (AHB1) RCC_AHB2SMENR (AHB2) RCC_AHB3SMENR (AHB3) (1)	Used to [enable/disable] the AHB peripheral clock in Sleep mode
APB1	RCC_APB1RSTR	RCC_APB1RSTR1 RCC_APB1RSTR2 (1)	Used to [enter/exit] the APB1 peripheral from reset
	RCC_APB1ENR	RCC_APB1ENR1 RCC_APB1ENR2 (1)	Used to [enable/disable] the APB1 peripheral clock
	RCC_APB1LPENR	RCC_APB1SMENR1 RCC_APB1SMENR2 (1)	Used to [enable/disable] the APB1 peripheral clock in Sleep mode
APB2	RCC_APB2RSTR		Used to [enter/exit] the APB2 peripheral from reset
	RCC_APB2ENR		Used to [enable/disable] the APB2 peripheral clock
	RCC_APB2LPENR	RCC_APB2SMENR	Used to [enable/disable] the APB2 peripheral clock in Sleep mode

1. The register configuring the peripherals is not present in the STM32L1 Series, so it is not needed from a migration-only stand point

The configuration to access a given peripheral involves:

- identifying the bus to which the peripheral is connected (refer to *Table 9*)
- selecting the right register according the needed action (refer to *Table 14*)

For example, USART1 is connected to the APB2 bus. In order to enable the USART1 clock, the RCC_APB2ENR register needs to be configured as follows with the STM32Cube HAL driver RCC API:

```
__HAL_RCC_USART1_CLK_ENABLE();
```

In order to disable the USART1 clock during Sleep mode (to reduce power consumption) the RCC_APB2SMENR register needs to be configured as follows with the STM32Cube HAL driver RCC API:

```
__HAL_RCC_USART1_CLK_SLEEP_ENABLE();
```

4.5.3 Peripheral clock configuration

Some peripherals have a dedicated clock source, independent from the system clock, which is used to generate the clock required for their operation:

- **USB**:

  In STM32L1 Series, the USB 48 MHz clock is derived from the PLL VCO clock which must be at 96 MHz.

  In STM32L4 Series / STM32L4+ Series, the USB 48 MHz clock is derived from one of the following sources:

  – Main PLL VCO (PLLUSB1CLK)
  – PLLSAI1 VCO (PLLUSB2CLK)
  – MSI clock (when the MSI clock is auto-trimmed with the LSE, it can be used by the USB OTG FS device)
  – HSI48 internal oscillator (only on STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices).

- **SDIO/SDMMC**:

  In STM32L1 Series, the SDIO clock (SDIOCLK) is derived from the PLL VCO clock and is equal to PLLVCO/2.

  In STM32L4 Series / STM32l4+ Series, the SDMMC clock is derived from one of the following sources:

  – Main PLL VCO (PLLUSB1CLK)
  – PLLSAI1 VCO (PLLUSB2CLK)
  – MSI clock
  – HSI48 internal oscillator (only on STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices).

- **RTC and LCD**:

  The RTC and the LCD glass clock share the same clock source (RTCCLK).

  In STM32L1 Series, the RTC and the LCD glass clock are derived from one of the three following sources: LSE, LSI or HSE divided by prescaler (/2, 4, 8, 16).

  In STM32L4 Series / STM32L4+ Series, the RTC and the LCD glass clocks are derived from one of the three following sources: LSE clock, LSI clock, or HSE clock divided by 32. The PCLK frequency must always be greater than or equal to the RTC clock frequency.

Note: LCD is not available on STM32L4+ Series.

- **ADC***:*

  In STM32L1 Series, the ADC features two clock schemes:

  – Clock for the analog circuitry: ADCCLK. This clock is always the HSI oscillator clock. A divider by 1, 2 or 4 allows to adapt the clock frequency to the operating conditions of the device. This configuration is done using the ADC_CCR[ADCPRE] bits. The ADC clock depends also on the voltage range V_{CORE}. When the product voltage range 3 is selected (V_{CORE} = 1.2 V), the ADC is in low speed (ADCCLK = 4 MHz, 250 Ksps).
  – Clock for the digital interface (used for the read/write access of the register). This clock is the APB2 clock. The digital interface clock is enabled/disabled through the RCC_APB2ENR register (ADC1EN bit) and there is a bit to reset the ADC through the RCC_APB2RSTR[ADCRST] bit.

In STM32L4 Series / STM32L4+ Series, the input clock of the two ADCs (master and slave) can be selected between two different clock sources:

– Derived (selected by software) from system clock (SYSCLK), PLLSAI1 VCO[a] (PLLADC1CLK) or PLLSAI2 VCO[b] (PLLADC2CLK). In this mode, a programmable divider factor can be selected (1, 2, …, 256 according to bits PREC[3:0]).

– Derived from the AHB clock of the ADC bus interface, divided by a programmable factor (1, 2 or 4). In this mode, a programmable divider factor can be selected (1, 2 or 4 according to bits CKMODE[1:0]). Refer to the STM32L4 Series / STM32L4+ Series reference manual for more details.

- **DAC**:

  In STM32L4 Series / STM32L4+ Series, in addition to the PCLK1 clock, the LSI clock is used for the sample and hold operation.

- **U(S)ARTs**:

  In STM32L1 Series, the U(S)ART clock is the APB1 or APB2 clock, depending on which APB bus is mapped to the U(S)ART.

  In STM32L4 Series / STM32L4+ Series, the U(S)ART clock is derived from one of the four following sources: system clock (SYSCLK), HSI16, LSE, APB1 or APB2 clock (depending on which APB bus is mapped to the U(S)ART)

  Using a source-clock independent from the system clock (like HSI16) allows to change the system clock on-the-fly without need to reconfigure the U(S)ART peripheral baud rate prescalers.

- **I2Cs**:

  In STM32L1 Series, the I2C clock is the APB1 clock (PCLK1).

  In STM32L4 Series / STM32L4+ Series, the I2C clock is derived from one of the three following sources: system clock (SYSCLK), HSI16 or APB1 (PCLK1).

  Using a source-clock independent from the system clock (like HSI16) allows to change the system clock on-the-fly without need to reconfigure I2C peripheral timing register.

- **I2S/SAI:**

  In STM32L1 Series, the I2S clocks are derived from one of the three following sources: HSI16, HSE or PLL clock.

  In STM32L4 Series / STM32L4+ Series, the I2S peripherals are not available and they are replaced by the SAIs.

  In STM32L4 Series / STM32L4+ Series, the I2S peripherals are not available and replaced by SAIs.

  For STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx devices, the SAI clocks are derived from one of the four following sources:

  – An external clock mapped on SAI1_EXTCLK or SAI2_EXTCLK

  – PLLSAI1 VCO (PLLSAI1CLK)

  – PLLSAI2 VCO (PLLSAI2CLK)

  – Amain PLL VCO (PLLSAI3CLK)

a. Not available on STM32L41xxx/42xxx, only SYSCLK could be used on those devices.

b. PLLSAI2VCO (PLLADC2CLK) is a clock source only on STM32L49xxx/4Axxx and STM32L47xxx/48xxx devices.

For STM32L45xxx/46xxx and STM32L43xxx/44xxx devices, the SAI clocks are derived from one of the four following sources:

– An external clock mapped on SAI1_EXTCLK for SAI1
– PLLSAI1 (P) divider output (PLLSAI1CLK)
– Amain PLL (P) divider output (PLLSAI2CLK)
– HSI16 clock

4.6 Power control (PWR)

In STM32L4 Series / STM32L4+ Series, the PWR controller presents some differences versus the one in STM32L1 Series. These differences are summarized in *Table 15*.

Table 15. PWR differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

PWR	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Power supplies	– V_{DD} = 1.8 V at power on or 1.65 V at power down to 3.6 V when the BOR is available – V_{DD} = 1.65 V to 3.6 V when BOR is not available – V_{DD} is the external power supply for I/Os and internal regulator – It is provided externally through V_{DD} pins	– V_{DD} = 1.71 to 3.6 V: external power supply for I/Os and internal regulator. – It is provided externally through V_{DD} pins
	– V_{CORE} = 1.2 to 1.8 V – V_{CORE} is the power supply for digital peripherals, SRAM and Flash memory – It is generated by a internal voltage regulator – Three V_{CORE} ranges can be selected by software depending on V_{DD} and target frequency	– V_{CORE} = 1.0 to 1.28 V – V_{CORE} is the power supply for digital peripherals, SRAM and Flash memory – It is generated by a internal voltage regulator – Two V_{CORE} ranges can be selected by software depending on target frequency
	NA	V_{BAT} = 1.55 to 3.6 V: power supply for RTC, external clock 32 kHz oscillator and backup registers (through power switch) when V_{DD} is not present
	V_{DD} and V_{DDA} must be at the same voltage value	Independent power supplies (V_{DDA}, V_{DDUSB}, V_{DDIO2}) allow to improve power consumption by running MCU at lower supply voltage than analog and USB
	– V_{SSA}, V_{DDA} = 1.8 V at power on or 1.65 V at power down to 3.6 V when BOR is available – V_{SSA}, V_{DDA} = 1.65 to 3.6 V when BOR is not available – V_{DDA} is the external analog power supply for ADC, DAC, reset blocks, RC oscillators and PLL. – The minimum voltage to be applied to V_{DDA} is 1.8 V when the ADC is used – V_{DDA} and V_{SSA} must be connected to V_{DD} and V_{SS} respectively	– V_{SSA}, V_{DDA} = 1.62 V (ADCs/COMPs) to 3.6 V 1.8 V (DAC/OPAMPs) to 3.6 V 2.4 V (VREFBUF) to 3.6 V – V_{DDA} is the external analog power supply for A/D and D/A converters, voltage reference buffer, operational amplifiers and comparators – The V_{DDA} voltage level is independent from the V_{DD} voltage

Table 15. PWR differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

PWR	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Power supplies (continued)	– V_{LCD} = 2.5 to 3.6 V – The LCD controller can be powered either externally through the VLCD pin, or internally from an internal voltage generated by the embedded step-up converter	V_{LCD} = 2.5 to 3.6 V (idem STM32L1 Series)
	NA	– V_{DDUSB} = 3.0 V to 3.6 V – V_{DDUSB} is the external independent power supply for USB transceivers – The V_{DDUSB} voltage level is independent from the V_{DD} voltage
		– V_{DDIO2} = 1.08 V to 3.6 V – V_{DDIO2} is the external power supply for 14 I/Os (Port G[15:2]) – The V_{DDIO2} voltage level is independent from the V_{DD} voltage Not applicable for STM32L45xxx/46xxx, STM32L43xxx/44xxx nor STM32L41xxx/42xxx devices
		– Available only on SM32L4R9xx/4S9xx – VDDDSI is independent DSI power supply dedicated for the DSI regulator and the MIPI D-PHY – This supply must be connected to the global VDD
		– Available only on SM32L4R9xx/4S9xx – VCAPDSI is the output of the DSI regulator (1.2 V) which must be connected externally to VDD12DSI
		– Available only on SM32L4R9xx/4S9xx – VDD12DSI is used to supply the MIPI D-PHY, and to supply the clock and data lanes pins – An external capacitor of 2.2µF must be connected on the VDD12DSI pin
Battery backup domain		– RTC with backup registers (128 bytes) – LSE – PC13 to PC15 I/Os – 3 tamper pins

Table 15. PWR differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

PWR	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Power supply supervisor	– Integrated POR / PDR circuitry – Programmable voltage detector (PVD)	
	– Brownout reset (BOR) – BOR can be disabled after power-on	– Brownout reset (BOR) – BOR is always enabled, except in Shutdown mode
	NA	4 peripheral voltage monitoring (PVM): – PVM1 for V_{DDUSB} – PVM2 for V_{DDIO2} (for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxxonly) – PVM3 / PVM4 for V_{DDA} (~1.65 V/ ~2.2 V)
Low-power modes	Sleep mode	
	– Low-power run mode (up to 128 kHz) – Low-power sleep mode (up to 128 kHz)	– Low-power run mode (up to 2 MHz) – Low-power sleep mode (up to 2 MHz) – System clock is limited to 2 MHz, but I2C and U(S)ART/LPUART can be clocked with HSI16 at 16 MHz
	Stop mode	– Stop 0, Stop 1 and Stop 2 mode – Some additional functional peripherals (cf wakeup source)
	Standby mode (V_{CORE} domain powered off)	Standby mode (V_{CORE} domain powered off) with new features: – BOR is always ON – SRAM2 content can be preserved – Pull-up or pull-down can be applied on each I/O
	NA	Shutdown mode (V_{CORE} domain powered off and power monitoring off)
External SMPS	NA	– Support for external SMPS for high-power efficiency. Refer to AN4978.

Table 15. PWR differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

PWR	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Wake-up sources	Sleep mode Any peripheral interrupt/wakeup event	
	Stop mode – Any EXTI line event/interrupt – BOR, PVD, COMP, RTC, USB, IWDG	Stop mode – Any EXTI line event/interrupt – BOR, PVD, PVM, COMP, RTC, USB, IWDG, U(S)ART, LPUART, I2C, SWP, LPTIM, LCD
	Standby mode – 3 WKUP pins rising edge – RTC event – External reset in NRST pin – IWDG reset	Standby mode – Up to 5 WKUP pins rising or falling edge – RTC event – External reset in NRST pin – IWDG reset
	NA	Shutdown mode – Up to 5 WKUP pins rising or falling edge – RTC event – External reset in NRST pin
Wake-up clocks	Wakeup from Stop mode MSI (all ranges up to 4.1 MHz)	Wakeup from Stop mode HSI16 16 MHz or MSI (all ranges up to 48 MHz) allowing 5 µs wakeup at high speed without waiting for PLL startup time
	Wakeup from Standby mode MSI 2.097 MHz	Wakeup from Standby mode MSI (ranges from 1 to 8 MHz)
	NA	Wakeup from Shutdown mode MSI 4 MHz
Configuration	NA	– In STM32L4 Series / STM32L4+ Series the registers are different – From 2 registers in STM32L1 Series to up to 25 registers in STM32L4 Series / STM32l4+ Series 4 control registers 2 status registers 1 status clear register 2 registers per GPIO port (A,B,I) for controlling pull-up and pull-down – Most configuration bits from STM32L1 Series can be found in STM32L4 Series / STM32L4+ Series (but sometime may have different programming mode)
Color key: = New feature or new architecture = Same feature, but specification change or enhancement = Feature not available (NA) = Differences		

4.7 Real-time clock (RTC)

STM32L4 Series / STM32L4+ Series and STM32L1 Series implement almost the same features on the RTC. *Table 16* shows the differences.

Table 16. RTC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

RTC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Features	– Coarse digital calibration (kept for compatibility only) – New developments must only use smooth calibration	Only smooth calibration available
	1 tamper pin (available in VBAT)	3 tamper pins (available in VBAT
	80-byte backup registers	128-byte backup registers
Configuration	-	– RTC_CR/DCE not available – RTC_CALIB register not available – RTC_TAFCR (L1) -> RTC_TAMPCR (L4) Except bit ALARMOUTTYPE available on RTC_OR/RTC_ALARM_TYPE
Color key: = Same feature, but specification change or enhancement = Feature not available (NA)		

For more information about the RTC features of the STM32L4 Series / STM32L4+ Series, refer to the RTC section of the STM32L4 Series and STM32L4+ Series reference manuals.

4.8 System configuration controller (SYSCFG) and RI

STM32L4 Series / STM32L4+ Series and STM32L1 Series implement almost the same features on the SYSCFG.

Table 17 shows the differences.

Table 17. SYSCFG - RI differences between STM32L1 Series and STM32L4 Series

SYSCFG - RI	STM32L1 Series	STM32L4 Series / STM32L4+ Series
RI features	TIM2/TIM3/TIM4's input captures 1,2,3 and four routing selections from selectable I/Os: – Routing of internal reference voltage VREFINT to selectable I/Os for all packages – Up to 40 external I/Os + 3 internal nodes (internal reference voltage + temperature sensor + V_{DD} and $V_{DD}/2$ measurement by VCOMP) can be used for data acquisition purposes in conjunction with the ADC interface – Input and output routing of COMP1 and COMP2	– The RI IO switches control used for the Touch Sense application has been replaced by a dedicated peripheral (TSC) in STM32L4 Series / STM32L4+ Series – The remaining switches control (for ADC, COMP) and internal interconnects are managed inside each specific peripheral in STM32L4 Series / STM32L4+ Series. The overall functionality is not equivalent
SYSCFG features	– Remapping memory areas – Managing the external interrupt line connection to the GPIOs – Configuring USB pull-up resistor	– Remapping memory areas – Managing the external interrupt line connection to the GPIOs – Managing robustness feature – Setting SRAM2 write protection and software erase – Configuring FPU interrupts – Enabling the firewall – Enabling /disabling I2C Fast-mode Plus driving capability on some I/Os and voltage booster for I/Os analog switches
Configuration	-	– Most registers from STM32L1 Series are identical in STM32L4 Series / STM32L4+ Series – A few bits are different and EXTI configuration may differ (number of GPIOs is different depending on the product)
Color key: = Same feature, but specification change or enhancement = Feature not available (NA)		

4.9 General-purpose I/Os (GPIO) interface

The GPIO peripheral of STM32L4 Series / STM32L4+ Series embeds identical features compared to the one present on the STM32L1 Series.

Minor adaptation of the code written for the STM32L1 Series using the GPIO may be required on STM32L4 Series / STM32L4+ Series due to:

- Mapping of particular function on different GPIOs (see pinout difference in *Section 2: Hardware migration*.
- Alternate function selection differences (AFSELy[3:0] in registers GPIOx_AFRL and GPIOx_AFRH).

The main GPIO features are:

- GPIO mapped on AHB bus for better performance.
- I/O pin multiplexer and mapping: the pins are connected to on-chip peripherals/modules through a multiplexer that allows only one peripheral alternate function (AF) connected to an I/O pin at a time. In this way, no conflict can occur between peripherals sharing the same I/O pin.
- More possibilities and features for I/O configuration.

For more information on the STM32L4 Series / STM32L4+ Series GPIO programming and usage, refer to the “I/O pin multiplexer and mapping” section in the GPIO chapter of the STM32L4 Series and STM32L4+ Series reference manuals. For detailed description of the pinout and alternate function mapping, refer to the product datasheets.

4.10 Extended interrupts and events controller (EXTI) source selection

The external interrupt/event controller (EXTI) is very similar on both the STM32L1 Series and the STM32L4 Series / STM32L4+ Series. *Table 18* shows the main differences.

Table 18. EXTI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

EXTI	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Number of event/interrupt lines	Up to 24 lines	Up to 41 lines: – 12 direct, 26 configurable on STM324Rxxx/4Sxxx – 15 direct, 26 configurable on STM32L49xxx/4Axxx devices – 14 direct, 26 configurable on STM32L47xxx/48xxx devices – 12 direct, 25 configurable on STM32L43xxx/44xxx and STM32L41xxx/42xxx devices

Table 18. EXTI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

EXTI	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Configuration	-	The selection of EXTI line source is performed through EXTIx bits in SYSCFG_EXTICRx registers (in STM32L1 and STM32L4 Series / STM32L4+ Series). However, the mapping of the EXTICRx registers has been changed

Color key:

= Same feature, but specification change or enhancement

4.11 Flash memory

Table 19 presents the differences between the Flash interface of STM32L1 Series and STM32L4 Series / STM32L4+ Series.

The STM32L4 Series and STM32L4+ Series instantiates a different Flash module both in terms of architecture/technology and interface, consequently the STM32L4 Series / STM32L4+ Series Flash memory programming procedures and registers are different from the ones for STM32L1 Series, and any code written for the FLASH interface in STM32L1 Series needs to be rewritten to run in STM32L4 Series and STM32L4+ Series.

For more information on programming, erasing and protection of the STM32L4 Series / STM32L4+ Series Flash memory, refer to STM32L4 Series and STM32L4+ Series reference manuals.

Table 19. Flash memory differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

Flash	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Main / Program memory	0x0800 0000 to (up to) 0x0805 FFFF	– 0x0800 0000 to (up to) 0x080F FFFF – 0x0800 0000 to (up to) 0x081F FFFF (only for STM32L4+ Series)
	– Up to 512 Kbytes – Split in 2 Banks – Each bank: up to 256 Kbytes – Sector size = 4 Kbyte: 16 Pages of 256 bytes – Programming granularity: 32-bit – Read granularity: 64/32-bit	– For STM32L4+ Series: Up to 2 Mbytes Split in 2 banks When dual bank is enabled each bank: 256 pages of 8 Kbytes and each page: 8 rows of 512 bytes When dual bank is disabled memory block contains 256 pages of 4 Kbyte and each page: 8 rows of 1024 bytes – For STM32L49xxx/4Axxx and STM32L47xxx/48xxx: Up to 1 Mbytes Split in 2 banks Each bank: 256 pages of 2 Kbytes Each page: 8 rows of 256 bytes – For STM32L45xxx/46xxx: Up to 512 Kbytes 1 bank 256 pages of 2 Kbytes Each page: 8 rows of 256 bytes – For STM32L43xxx/44xxx: Up to 256 Kbytes 1 bank 128 pages of 2 Kbytes Each page: 8 rows of 256 bytes – For STM32L41xxx/42xxx: Up to 128 Kbytes 1 bank 64 pages of 2 Kbytes Each page: 8 rows of 256 bytes – Programming and read granularity: 72-bit (incl 8 ECC bits)
Features	– Read while write (RWW) – Dual bank boot – ECC (data EEPROM only)	– Read while write (RWW) – Dual bank boot (only for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx) – ECC
	NA	– ECC – Flash empty check (only for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)

Table 19. Flash memory differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Flash	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Wait state	Up to 1 (depending on the supply voltage and frequency)	Up to 5 (depending on the core voltage and frequency)
ART Accelerator™	NA	Allowing 0 wait state when executing from the cache
Data EEPROM memory	– 4 Kbytes 0x0808 0000 to 0x0808 0FFF (Cat.1,2) – 8 Kbytes 0x0808 0000 to 0x0808 1FFF (Cat.3) – 12 Kbytes 0x0808 0000 to 0x0808 2FFF (Cat.4) – 16 Kbytes 0x0808 0000 to 0x0808 3FFF (Cat.5)	N/A (can be emulated by SW)
System memory	– 4 Kbytes 0x1FF0 0000 to 0x1FF0 0FFF (Cat.1,2) – 8 Kbytes 0x1FF0 0000 to 0x1FF0 1FFF (Cat.3,4,5)	– 27 Kbytes 0x1FFF 0000 to 0x1FFF 6FFF (bank1) – 27 Kbytes 0x1FFF 8000 to 0x1FFF EFFF (bank2) – Only bank1 for STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx
One time programmable (OTP)	NA	1 Kbytes 0x1FFF 7000 to 0x1FFF 73FF (bank1)
Option bytes	– 0x1FF8 0000 to 0x1FF8001F (all Cat.x) – 0x1FF8 0080 to 0x1FF8 009F (Cat.4,5)	– 0x1FFF 7800 to 0x1FFF 780F (bank1) – 0x1FFF F800 to 0x1FFF F80F (bank2) – Only bank1 for STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx – For STM32L4+ Series: 0x1FF0 0000 to 0x1FF0 0020 (bank 1) 0x1FF0 1000 to 0x1FF0 1020 (bank 2)
Flash memory interface	0x4002 3C00 to 0x4002 3FFF	0x4002 2000 to 0x4002 23FF
	-	Different from STM32L1 Series
Erase granularity	– Program memory: Mass/Page (256 byte) – DATA EEPROM memory: byte/ halfword/ word / double word	Page erase (2 Kbyte), Bank erase and Mass erase (all banks)
Read protection (RDP)	– Level 0 no protection – RDP = 0xAA	
	– Level 1 memory protection – RDP ≠ (Level 2 & Level 0)	
	Level 2 RDP = 0xCC (1)	

Table 19. Flash memory differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Flash	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Proprietary code readout protection (PCROP)	Granularity: 1 sector (4 Kbyte)	– 1 PCROP area per bank – Granularity: 64-bit – PCROP_RDP option: PCROP area preserved when RDP level decreased – For STM32L4+ Series: Dual bank: 1PCROP area per bank Single bank: 2 PCROP area
Write protection (WRP)	Granularity: 1 sector (4 Kbyte)	– Two write protection areas per bank – Granularity: 2 Kbytes – For STM32L4+ Series: Dual bank: 2 areas per bank Single bank: 4 areas

Table 19. Flash memory differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

Flash	STM32L1 Series	STM32L4 Series / STM32L4+ Series
User option bytes	nRST_STOP	nRST_STOP
	nRST_STDBY	nRST_STDBY
	IWDG_SW	IWDG_SW
	NA	IWDG_STOP, IWDG_STDBY
		WWDG_SW
	BOR_LEV[3:0]	BOR_LEV[2:0]
	nBFB2	BFB2 (except for STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices)
	NA	nBOOT1
		SRAM2_RST, SRAM2_PE
		DUAL BANK (except for STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)
		nBOOT0 (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)
		nSWBOOT0 (for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)
		DBANK (only for STM32L4+ Series)
		DB1M (only for STM32L4+ Series)

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Feature not available (NA)

= Differences

4.12 Universal synchronous asynchronous receiver transmitter (U(S)ART)

STM32L4 Series and STM32L4+ Series implement several new features on U(S)ART compared to STM32L1 Series.

Table 20 shows the differences.

Table 20. U(S)ART differences between STM32L1 Series and STM32L4 Series

U(S)ART	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Instances	x3 USART x2 UART	– x3 USART – x2 UART for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx – x1 UART for STM32L45xxx/46xxx – x1 LPUART
Baud rate	Up to 4 Mbit/s when the clock frequency is 32 MHz and oversampling is by 8	Up to 10 Mbit/s when the clock frequency is 80 MHz and oversampling is by 8
Clock	Single clock domain	Dual clock domain allowing: – UART functionality and wakeup from Stop mode – Convenient baud rate programming independent from the PCLK reprogramming
Data	Word length: programmable (8 or 9 bits)	– Word length: programmable (7, 8 or 9 bits) – Programmable data order with MSB-first or LSB-first shifting
Interrupt	10 interrupt sources with flags	– 14 interrupt sources with flags – 23 interrupt sources with flags for STM32L4+ Series
Features	– RS232 hardware flow control (CTS/RTS) – Continuous communication using DMA – Multiprocessor communication – Single-wire half-duplex communication – IrDA SIR ENDEC block – LIN mode – SPI master	
	– Smartcard mode T = 0 and T = 1 is to be implemented by software – Number of stop bits: 0.5, 1, 1.5, 2	– Smartcard mode T = 0, T=1 are supported Features are added to support T = 1 such as receiver timeout, block length, end of block detection, binary data inversion etc...) – Number of stop bits: 1, 1.5, 2

Table 20. U(S)ART differences between STM32L1 Series and STM32L4 Series

U(S)ART	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Features	NA	– Wakeup from Stop mode (Start bit, Received byte, Address match) – Support for ModBus communication: Timeout feature CR/LF character recognition. – Two internal FIFOs for transmit and receive data (for STM32L4+ Series) – SPI slave (for STM32L4+ Series) – Receiver timeout interrupt (except LPUART) – Auto baud rate detection (except LPUART) – Driver enable – Swappable Tx/Rx pin configuration – LPUART does not support Synchronous mode (SPI master), Smartcard mode, IrDA, LIN, ModBus, Receiver timeout interrupt, Auto baud rate detection
Configuration	-	– L1 registers and associated bits are not identical in STM32L4 Series / STM32L4+ Series – Refer to STM32L4 Series and STM32L4+ Series reference manuals for details

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Feature not available (NA)

= Differences

4.13 Inter-integrated circuit (I2C) interface

STM32L4 Series / STM32L4+ Series implement a different I2C peripheral which allows an easier software management.

Table 21 shows the differences.

Table 21. I2C differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

I2C	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Instances	x2 (I2C1, I2C2)	– x3 for STM32L47xxx/48xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx – x4 for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L45xxx/46xxx
Features	– 7-bit and 10-bit Addressing mode – SMBus – Standard mode (Sm, up to 100 kHz) – Fast mode (Fm, up to 400 kHz)	
	NA	– Fast mode Plus (Fm+, up to 1 MHz) – Independent clock – Wakeup from Stop on address match
Configuration	-	– Register configuration is very different in STM32L1 Series and STM32L4 Series / STM32L4+ Series – Refer to STM32L4 Series and STM32L4+ Series reference manuals for details.

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Differences

4.14 Serial peripheral interface (SPI) /IC to IC sound (I2S) / Serial audio interface (SAI)

STM32L4 Series / STM32L4+ Series and STM32L1 Series implement almost the same features on the SPI (apart from the I2S).

Table 22 shows the differences.

Table 22. SPI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

SPI	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Instances	x3 (SPI1, SPI2, SPI3)	x3
Features	SPI + I2S	– I2S feature is not supported by SPI on STM32L4 Series / STM32L4+ Series – SAI interfaces are available instead: x2 for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx x1 for STM32L45xxx/46xxx and STM32L43xxx/44xxx
Data size	Fixed, configurable to 8 or 16 bits	Programmable from 4 to 16-bit
Data buffer	Tx & Rx 16-bit buffers (single data frame)	32-bit Tx & Rx FIFOs (up to 4 data frames)
Data packing	No (16-bit access only)	Yes (8-bit, 16-bit or 32-bit data access, programmable FIFOs data thresholds)
Mode	– SPI TI mode – SPI Motorola mode	– SPI TI mode – SPI Motorola mode – NSSP mode
Speed	16 MHz (core at 32 MHz)	40 MHz (core at 80 Mhz)
Configuration	-	The data size and Tx/Rx flow handling are different in STM32L1 and STM32L4 Series / STM32L4+ Series hence requiring different SW sequences

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Differences

Migrating from I2S to SAI:

STM32L4 Series / STM32L4+ Series do not include an I2S interface as part of the SPI peripheral but two serial audio interfaces instead.

Table 23 shows the main differences between the I2S and the SAI.

Table 23. I2S/SAI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

I2S/SAI	STM32L1 Series (I2S)	STM32L4 Series / STM32L4+ Series (SAI)
Instances	x2	– x2 (SAI1, SAI2) for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx – x1 (SAI1) for STM32L45xxx/46xxx and STM32L43xxx/44xxx
Features	Full-duplex communication.	Two independent audio sub-blocks (per SAI) which can be transmitters or receivers with their respective FIFOs
	Master or slave operations	– Synchronous or Asynchronous mode between the audio sub-blocks – Possible synchronization between multiple SAIs – Master or slave configuration independent for both audio sub-blocks
	8-bit programmable linear prescaler to reach accurate audio sample frequencies (from 8 kHz to 192 kHz)	Clock generator for each audio block to target independent audio frequency sampling when both audio sub-blocks are configured in Master mode
	– Data format may be 16-bit, 24-bit or 32-bit – Data direction is always MSB first	– Data size configurable: 8-, 10-, 16-, 20-, 24-, 32-bit – First active bit position in the slot is configurable – LSB first or MSB first for data transfer
	Channel length is fixed to 16-bit (16-bit data size) or 32-bit (16-bit, 24-bit, 32-bit data size) by audio channel	– Up to 16 slots available with configurable size – Number of bits by frame can be configurable – Frame synchronization active level configurable (offset, bit length, level) – Stereo/mono audio frame capability
	Programmable clock polarity (steady state)	Communication clock strobing edge configurable (SCK)
	16-bit register for transmission and reception with one data register for both channel sides	8-word integrated FIFOs for each audio sub-block (facilitating Interrupt mode)

Table 23. I2S/SAI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

I2S/SAI	STM32L1 Series (I2S)	STM32L4 Series / STM32L4+ Series (SAI)
Features	Supported I2S protocols: – I2S Philips standard – MSB-justified standard (left-justified) – LSB-justified standard (right-justified) – PCM standard With short and long frame synchronization on 16-bit channel frame or 16-bit data frame extended to 32-bit channel frame	– Audio protocol: I2S, LSB or MSB-justified, PCM/DSP, TDM (up to 16 channels), AC'97 – SPDIF output – Mute mode – PDM interface (for STM32L4Rxxx/L4Sxxx)
	DMA capability for transmission and reception (16-bit wide)	2-channel DMA per SAI
	– Master clock may be output to drive an external audio component. – Ratio is fixed at 256 × Fs (where Fs is the audio sampling frequency)	
	Interruption sources when enabled: – Errors – Tx buffer empty, Rx buffer not empty	Interruption sources when enabled: – Errors – FIFO requests
	Error flags with associated interrupts if enabled respectively: – Overrun and underrun detection – Anticipated frame synchronization signal detection in Slave mode, – Late frame synchronization signal detection in Slave mode	Same than STM32L1 Series + protection against misalignment in case of underrun and overrun
Configuration	-	– There is no compatibility between STM32L1 Series I2S and STM32L4 Series/ STM32L4+ Series SAI – The user has to configure the SAI interface for the target protocol – Refer to reference manuals for details
Color key: = New feature or new architecture = Same feature, but specification change or enhancement = Differences		

The SAI peripheral improves the robustness of the communication in Slave mode compared to the I2S peripheral (in case of data clock glitch for example).

In Master mode, while migrating an application from the STM32L1 Series to the STM32L4 Series / STM32L4+ Series, the user must review the possible master clock (MCLK), the data bit clock (SCK) and the frame synchronization (FS) frequency reachable. The user must use the STM32L4 PLL multiplication factors and the SAI internal clock divider for a

given external oscillator which can be different than the one for the I2S in the STM32L1 Series.

In the STM32L4 Series / STM32L4+ Series, the SAI1 and SAI2 input clocks are derived (selected by software) from one of the following sources:

- For STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx devices:
  - An external clock mapped on SAI1_EXTCLK for SAI1 and SAI2_EXTCLK for SAI2
  - PLLSAI1 (P) divider output (PLLSAI1CLK)
  - PLLSAI2 (P) divider output (PLLSAI2CLK)
  - Main PLL (P) divider output (PLLSAI3CLK)
- For STM32L45xxx/46xxx and STM32L43xxx/44xxx devices:
  - An external clock mapped on SAI1_EXTCLK for SAI1
  - PLLSAI1 (P) divider output (PLLSAI1CLK)
  - Main PLL (P) divider output (PLLSAI2CLK)
  - HSI16 clock

When the clock is derived from one of the internal PLLs, the three PLL inputs are either HSI16, HSE or MSI (between 4 and 48 MHz) divided by a programmable factor PLLM (from 1 to 8 (or from1 to 16 for STM32L4+ Series)).

This input is then multiplied by PLLN (from 8 to 86) to reach PLL VCO frequency (must be between 64 and 344 MHz).

For STM32L4+ Series, when the clock is derived from one of the internal PLLs, the three PLL inputs are either HSI16, HSE or MSI divided by its own programmable factor (PLLM, PLLSAI1M and PLLSAI2M) (from 1 to 16).

It is finally divided by PLLP to provide the input clock of SAI (max. 80 MHz / 120 MHz for STM32L4+ Series)

- 7 or 17 on STM32L47xxx/48xxx
- [2...31] on STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx and STM32L43xxx/44xxx

When the master clock MCLK is used by the external slave audio peripheral, the PLL output is divided by the SAI internal master clock divider factor (1, 2, 4, 6, 8, ..., 30) to provide the master clock (MCLK). The data bit clock is then derived from MCLK the following the formula:

$$SCK = MCLK \times (FRL + 1) / 256$$

Where:

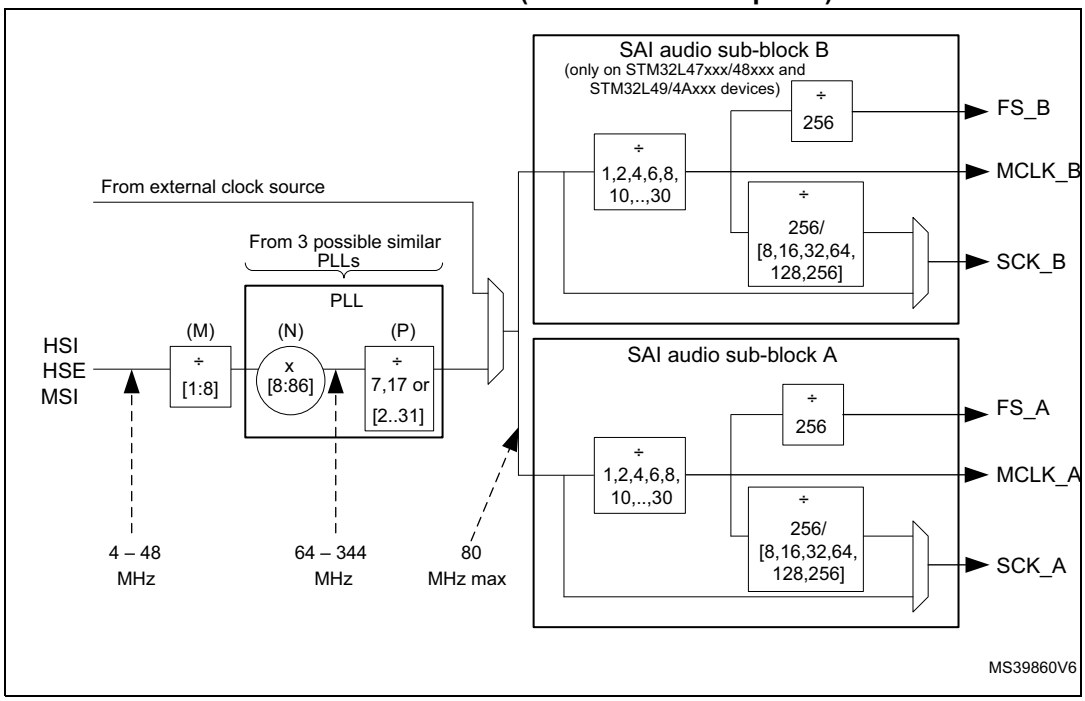
- FRL is the number of bit clock cycles - 1 in the audio frame (0 to 255).
- (FRL+ 1) must be a power of 2 higher or equal to 8.
- (FRL + 1) = 8, 16, 32, 64, 128, 256.

The SCK can also be directly connected to the input clock of the SAI when MCLK output is not needed.

The frame synchronization (FS) frequency is always MCLK / 256.

Figure 8 shows the clock generation scheme in the STM32L4 Series / STM32L4+ Series. Refer to the STM32L4 Series and STM32L4+ Series reference manuals for more details.

Figure 8. STM32L4 Series / STM32L4+ Series clock generation for SAI Master mode (when MCLK is required)



4.15 Cyclic redundancy check calculation unit (CRC)

The CRC calculation unit is very similar in STM32L1 Series and STM32L4 Series / STM32L4+ Series.

Table 24 shows the differences.

Table 24. CRC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

CRC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Features	– Single input/output 32-bit data register – CRC computation done in 4 AHB clock cycles (HCLK) for the 32-bit data size – General-purpose 8-bit register (can be used for temporary storage)	
	– Uses CRC-32 (Ethernet) polynomial: 0x4C11DB7 – Handles 32-bit data size	– Fully programmable polynomial with programmable size (7, 8, 16, 32bits) – Handles 8-,16-, 32-bit data size – Programmable CRC initial value – Input buffer to avoid bus stall during calculation – Reversibility option on I/O data

Table 24. CRC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

CRC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Configuration	-	– Configuration registers in STM32L1 Series are identical in STM32L4 Series / STM32L4+ Series – STM32L4 Series / STM32L4+ Series includes additional registers for new features – Refer to reference manuals for details
Color key: = New feature or new architecture		

4.16 Advanced encryption standard hardware accelerator (AES)

STM32L4 Series/ STM32L4+ Series implement several new features on the AES compared to STM32L1 Series.

Table 25 shows the differences.

Table 25. AES differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

AES	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Features	128-bit register for storing the encryption or derivation key (4x 32-bit registers)	256-bit register for storing the encryption, decryption or derivation key (8x 32-bit registers)
Mode	– Electronic codebook (ECB) – Cipher block chaining (CBC) – Counter mode (CTR)	– Electronic codebook (ECB) – Cipher block chaining (CBC) – Counter mode (CTR) – Galois counter mode (GCM) – Galois message authentication code mode (GMAC) – Cipher message authentication code mode (CMAC)
Key length	128-bit	128-bit, 256-bit
Configuration	-	All registers and programming bits in the STM32L1 Series can be found in the STM32L4 Series / STM32L4+ Series
Color key: = New feature or new architecture = Same feature, but specification change or enhancement		

4.17 Liquid-crystal display controller (LCD)

The LCD in STM32L4 Series / STM32L4+ Series implements the same features than the one in STM32L1 Series except for an additional internal output buffer that allow to improve the contrast. It is possible to use the output buffers instead of the high-drive resistive network.

All programmable registers and associated bits in STM32L1 Series are equivalent to the ones in STM32L4 Series / STM32L4+ Series. However, due to the fact that the VLCD pin is implemented as an alternate function in STM32L4 Series / STM32L4+ Series (contrary to STM32L1 Series), a specific SW sequence is required to configure the LCD when the step-up converter is used as power source.

Refer to the STM32L1 Series and STM32L4 Series / STM32l4+ Series reference manuals for more details.

Note: LCD is not available on STM32L4+ Series nor STM32L4x1x/4x2x devices.

4.18 Universal serial bus interface (USB)

STM32L4 Series / STM32L4+ Series and STM32L1 Series have different USB peripherals.

STM32L1 Series, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices implement a USB FS device interface.

STM32L4+ Series, STM32L49xxx/4Axxx, STM32L47xxx/48xxx devices implement a USB OTG FS

On STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx devices, a clock recovery system (CRS) block is included. It can provide a precise clock to the USB peripheral:

- When using USB device mode, the CRS allows crystal-less USB operation.
- When using USB host mode, the CRS allows low frequency crystal (32.768 kHz) USB operation.

Most features supported by STM32L1 Series are also supported by STM32L4 Series and STM32L4+ Series.

The key differences are listed in *Table 26*.

Table 26. USB differences between STM32L1 Series and STM32L4 Series

USB	STM32L1 Series	STM32L4 Series
Features	Universal Serial Bus Revision 2.0	Universal Serial Bus Revision 2.0, including link power management (LPM) support
	NA	Full support for the USB on-the-go (USB OTG) (only on STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx)
	FS mode: – 1 bidirectional control endpoint – 7 IN endpoints (Bulk, Interrupt, Isochronous) – 7 OUT endpoints (Bulk, Interrupt, Isochronous)	FS mode: – For STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx: 1 bidirectional control endpoint 5 IN endpoints (Bulk, Interrupt, Isochronous) 5 OUT endpoints (Bulk, Interrupt, Isochronous) – For STM32L45xxx/46xxx and STM32L43xxx/44xxx: identical to STM32L1 Series
	USB internal connect/disconnect feature with an internal pull-up resistor on the USB D+ (USB_DP) line	
	NA	– Attach detection protocol (ADP) (only on STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx) – Battery charging detection (BCD)
		Independent V_{DDUSB} power supply allowing lower V_{DDCORE} while using USB
Mapping	APB1	– APB1 for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx – AHB2 for STM32L45xxx/46xxx, STM32L43xxx/44xx and STM32L41xxx/42xxx
Buffer memory	512 bytes (endpoint buffers and buffer descriptors structure)	– For STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx: 1.25 Kbyte data FIFOs Management of up to 6 Tx FIFOs (1 for each IN end point) + 1 Rx FIFO – For STM32L45xxx/46xxx, STM32L43xxx/44xx and STM32L41xxx/42xxx: 1024 bytes of dedicated packet buffer memory SRAM

Table 26. USB differences between STM32L1 Series and STM32L4 Series (continued)

USB	STM32L1 Series	STM32L4 Series
Low-power modes	USB suspend and resume	– USB suspend and resume. – Link power management (LPM) support
Configuration	-	– In STM32L4 Series the registers are different – Refer to reference manuals for details
Color key: = New feature or new architecture = Same feature, but specification change or enhancement = Feature not available (NA) = Differences		

4.19 Analog-to-digital converter (ADC)

Table 27 presents the differences between the ADC peripheral of STM32L1 Series and the one of STM32L4 Series / STM32L4+ Series. The main differences are the a new digital interface and new architecture/features.

Table 27. ADC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

ADC	STM32L1 Series	STM32L4 Series / STM32L4+ Series(1)
ADC type	SAR structure	
Instances	ADC1	– x3 instances for STM32L49xxx/4Axxx and STM32L47xxx/48xxx – 2 instances for STM32L41xxx/42xxx – x1 instance for STM32L4+ Series, STM32L45xxx/46xxx and STM32L43xxx/44xxx
Maximum sampling frequency	1 Msps	– 5.1 Msps (fast channels) – 4.8 Msps (slow channels)
Number of channels	Up to 42 channels	Up to 19 channels per ADC
Resolution	12-bit	12-bit + digital oversampling up to 16-bit
Conversion modes	Single / continuous / scan / discontinuous	Single / Continuous / Scan / Discontinuous / Dual mode
DMA	Yes	

Table 27. ADC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

ADC	STM32L1 Series		STM32L4 Series / STM32L4+ Series(1)	
External trigger	Yes			
	External event for regular group TIM9_CC2 TIM9_TRGO TIM2_CC3 TIM2_CC2 TIM3_TRGO TIM4_CC4 TIM2_TRGO TIM3_CC1 TIM3_CC3 TIM4_TRGO TIM6_TRGO EXTI line11	External event for injected group TIM9_CC1 TIM9_TRGO TIM2_TRGO TIM2_CC1 TIM3_CC4 TIM4_TRGO TIM4_CC1 TIM4_CC2 TIM4_CC3 TIM10_CC1 TIM7_TRGO EXTI line15	External event for regular group: TIM1 CC1 TIM1 CC2 TIM1 CC3 TIM2 CC2 TIM3 TRGO TIM4 CC4 EXTI line 11 TIM8_TRGO TIM8_TRGO2 TIM1_TRGO TIM1_TRGO2 TIM2_TRGO TIM4_TRGO TIM6_TRGO TIM15_TRGO TIM3_CC4	External event for injected group: TIM1 TRGO TIM1 CC4 TIM2 TRGO TIM2 CC1 TIM3 CC4 TIM4 TRGO EXTI line15 TIM8_CC4 TIM1_TRGO2 TIM8_TRGO TIM8_TRGO2 TIM3_CC3 TIM3_TRGO TIM3_CC1 TIM6_TRGO TIM15_TRGO
Supply requirement	1.8 V to 3.6 V		– 1.62 V to 3.6 V – Independent power supply (V_{DDA})	
Reference voltage	External		Reference voltage for STM32L4 Series external (2.0 V to V_{DDA}) or internal (2.048 V or 2.5 V)	
Electrical parameters	1.45 mA (max.), 1.0 mA (Typ.)		Consumption proportional to conversion speed: 200 µA/Msps (Typ.)	
Input range	$V_{REF-} \leq V_{IN} \leq V_{REF+}$			
Color key: = New feature or new architecture = Same feature, but specification change or enhancement				

1. On the STM32L4 Series / STM32L4+ Series devices on which the peripheral is not implemented, the external event is not applicable.

4.20 Digital-to-analog converter (DAC)

STM32L4 Series / STM32L4+ Series implement an enhanced DAC compared to STM32L1 Series. *Table 28* shows the differences.

Table 28. DAC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

DAC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Number of channels	x2	– x2 for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L47xxx/48xxx and STM32L43xxx/44xxx – x1 for STM32L45xxx/46xxx
Resolution	12-bit	
Features	– Left or right data alignment in 12-bit mode – Noise-wave and triangular-wave generation – DAC with 2 channels for independent or simultaneous conversions	
	NA	– Buffer offset calibration – DAC1_OUTx can be disconnected from output pin – Sample and Hold mode for low-power operation in Stop mode
DMA	Yes	
External trigger	Yes	
	– TIM6 TRGO – TIM7 TRGO – TIM9 TRGO – TIM2 TRGO – TIM4 TRGO – EXTI line9 – SW TRIG	– TIM6 TRGO – TIM8 TRGO(1) – TIM7 TRGO – TIM5 TRGO(1) – TIM2 TRGO – TIM4 TRGO(1) – EXTI line9 – SW TRIG Additional trigger for STM32L4+ Series: – TIM1_TRGO – TIM15_TRGO – LPTIM1_OUT – LPTMI2_OUT
Supply requirement	1.8 V to 3.6 V	– 1.8 V to 3.6 V – Independent power supply (V_{DDA})
Reference voltage	External	Reference voltage for STM32L4 Series / STM32L4+ Series external (1.8 V to V_{DDA}) or internal (2.048 V or 2.5 V)

Table 28. DAC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

DAC	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Configuration	-	SW compatible except for output buffer management

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Feature not available (NA)

= Differences

1. On the STM32L4 Series / STM32L4+ Series devices on which the peripheral is not implemented, the external trigger is not applicable.

4.21 Comparators (COMP)

Table 29 presents the differences between the COMP interface of STM32L1 Series and the one of the STM32L4 Series / STM32L4+ Series.

Table 29. COMP differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

COMP	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Type	– COMP1 fixed threshold – COMP2 rail-to-rail	– COMP1 – COMP2 rail-to-rail
Inputs	COMP1: – 25 (Cat.1,2)(24 ext IO + T sensor) – 32 (Cat.3,4,5)(29 ext IO + T sensor + OPAMP1/2)	COMP1: – Non Inverting: 2 (PC5, PB2) – Inverting: 8 (PB1,PC3, DAC1_OUT1/2, V_{REFINT} x 1, 3/4, 1/2, 1/4)
	COMP2: – Non inverting: 2 (Cat.1,2) (PB4, PB5) 4 (Cat3,4,5) (PB4, PB5, PB6, PB7) – Inverting: 7 (PB3, DAC1_OUT1/2, V_{REFINT} x 1, 3/4, 1/2, 1/4)	COMP2: – Non Inverting: 2 (PB4, PB6) – Inverting: 8 (PB3, PB7, DAC1_OUT1/2, V_{REFINT} x 1, 3/4, 1/2, 1/4)
Outputs	– Generation of input capture and OCREF clear signals for TIM2, TIM3, TIM4 and input capture for TIM10 – Generation of wakeup interrupt or events (EXTI line)	– Generation of break input signals for timers through GPIO alternate function – TIM1/TIM8 (STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx) – TIM1 (STM32L45xxx/46xxx and STM32L43xxx/44xxx) – Generation of wakeup interrupt or events (EXTI line)

Table 29. COMP differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

COMP	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Features	Window comparator	
	NA	– Output with blanking source – Programmable hysteresis
	Programmable speed/consumption (COMP2)	Programmable speed/consumption (COMP1/COMP2)
Supply requirement	1.65 V to 3.6 V	1.62 V to 3.6 V
Input range	$V_{REF-} \leq V_{IN} \leq V_{REF+}$	
Color key: = New feature or new architecture = Same feature, but specification change or enhancement = Feature not available (NA)		

4.22 Operational amplifiers (OPAMP)

STM32L4 Series / STM32L4+ Series implement enhanced OPAMPs compared to STM32L1 Series. *Table 30* shows the differences.

Table 30. OPAMP differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series

OPAMP	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Instances	x3	– x2 for STM32L4+ Series, STM32L49xxx/4Axxx and STM32L47xxx/48xxx – x1 for STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx
Features	– Rail-to-rail input and output voltage range – Low input bias current – Low input offset voltage – Low-power mode – Fast wakeup time – Gain bandwidth of 1 MHz	
	NA	Programmable gain amplifier (PGA)

Table 30. OPAMP differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

OPAMP	STM32L1 Series	STM32L4 Series / STM32L4+ Series
Configuration	-	The configuration registers are not organized in the same way in the STM32L4 Series / STM32L4+ Series and in the STM32L1 Series

Color key:

= New feature or new architecture

= Same feature, but specification change or enhancement

= Feature not available (NA)

= Differences

5 Software migration

5.1 References

- The definitive guide to Arm® Cortex®-M3 and Cortex®-M4 processors
- *STM32F10xxx/20xxx/21xxx/L1xxxx Cortex®-M3 programming manual* (PM0056)
- *STM32F3 Series, STM32F4 Series, STM32L4 Series and STM32L4+ Series Cortex®-M4 programming manual* (PM0214)
- Cortex®-M3 Technical Reference Manual, available on http://infocenter.arm.com
- Cortex®-M4 Technical Reference Manual, available from http://infocenter.arm.com

5.2 Cortex®-M3 and Cortex®-M4 overview

5.2.1 STM32 Cortex®-M3 processor and core peripherals

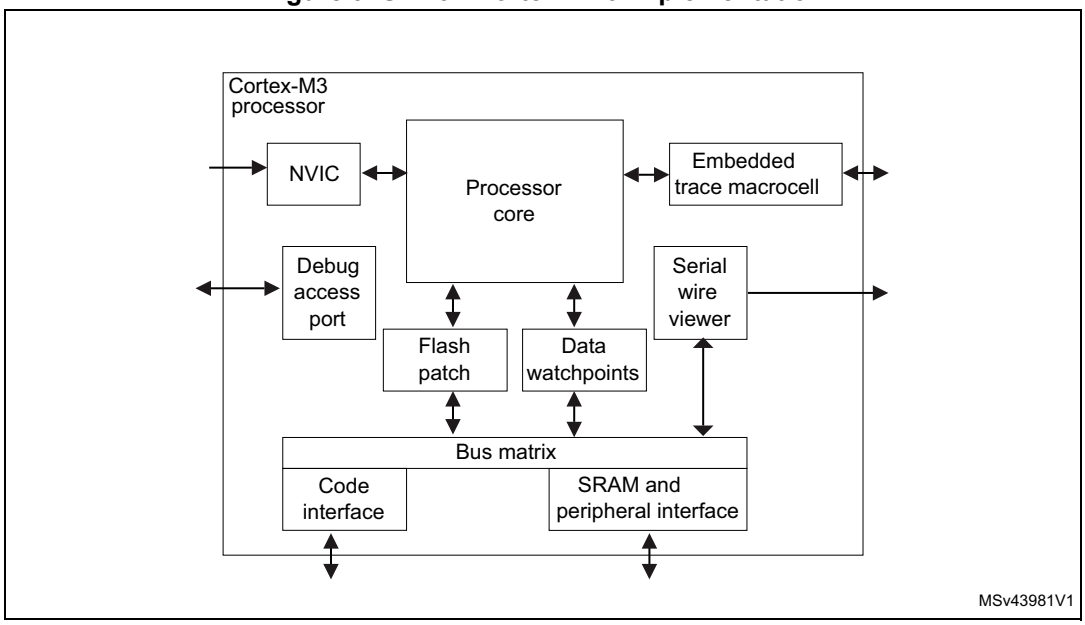
The Cortex®-M3 processor is built on a high-performance processor core, with a 3-stage pipeline Harvard architecture, making it ideal for demanding embedded applications. The processor delivers an exceptional power efficiency through an efficient instruction set and extensively optimized design, providing a high-end processing hardware including single-cycle 32x32 multiplications and a dedicated hardware division.

Cortex-M3 processor features and benefits summary

- Tight integration of system peripherals reducing the area and development costs
- Thumb instruction set combining a high code density with 32-bit performance
- Code-patch ability for ROM system update
- Power control optimization of system components
- Integrated sleep modes for a low-power consumption
- Fast code execution permitting a slower processor clock or increasing the Sleep-mode time
- Hardware division and fast multiplier
- Deterministic, high-performance interrupt handling for time-critical applications
- Extensive debug and trace capabilities.ssor features

Figure 9 presents the STM32 Cortex®-M3 implementation.

Figure 9. STM32 Cortex®-M3 implementation



Cortex®-M3 key features

- Architecture 32 bits RISC ARMv7-M
- 3-stage pipeline with branch speculation
- Instruction set:
  - Thumb, Thumb-2
  - Hardware multiply, hardware divide, saturated arithmetic

5.2.2 STM32 Cortex®-M4 processor and core peripherals

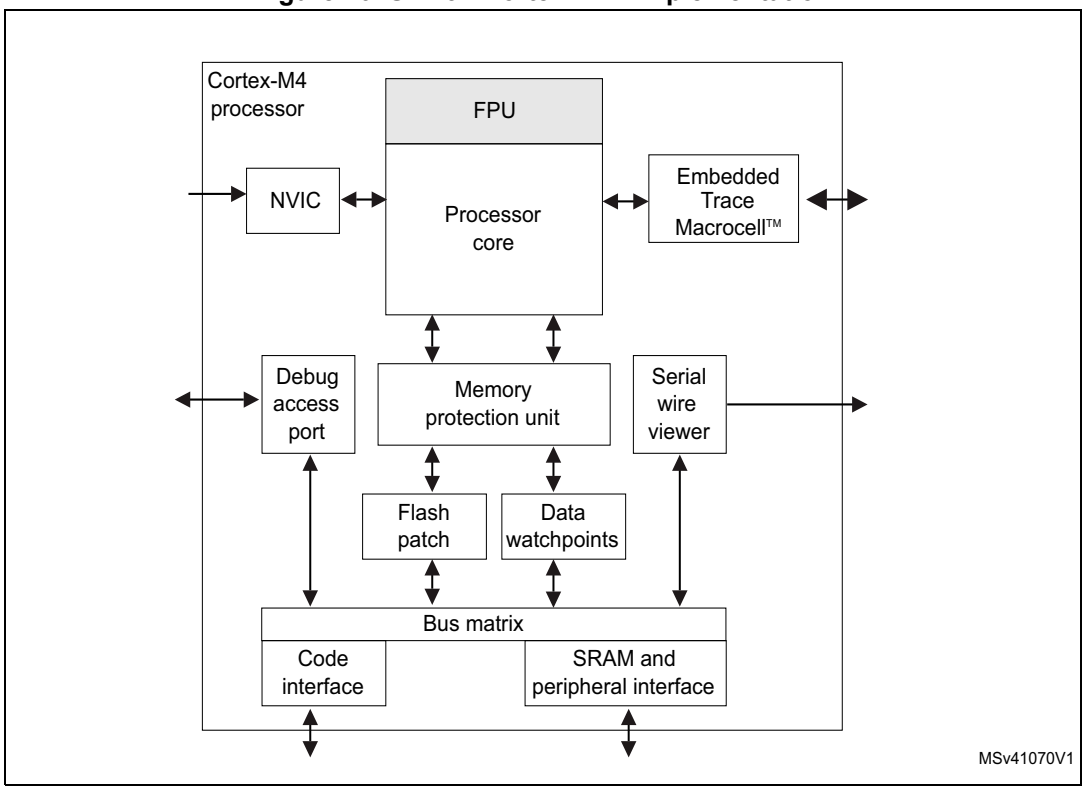
The Cortex®-M4 processor is a high-performance 32-bit processor designed for the microcontroller market. It offers significant benefits to the developers, including:

- Outstanding processing performance combined with fast interrupt handling
- Enhanced system debug with extensive breakpoint and trace capabilities
- Efficient processor core, system and memories
- Ultra-low power consumption with integrated sleep modes
- Platform security robustness, with integrated memory protection unit (MPU).

The Cortex®-M4 processor is built on a high-performance processor core, with a 3-stage pipeline Harvard architecture, making it ideal for demanding embedded applications. The processor delivers an exceptional power efficiency through an efficient instruction set and extensively optimized design, providing high-end processing hardware including IEEE754-compliant single-precision floating-point computation, a range of single-cycle and SIMD multiplication and multiply-with-accumulate capabilities, a saturating arithmetic and dedicated hardware division.

Figure 10 presents the STM32 Cortex®-M4 implementation.

Figure 10. STM32 Cortex®-M4 implementation



Cortex®-M4 key features

- Architecture 32 bits RISC ARMv7E-M
- 3-stage pipeline with branch speculation
- Instruction set:
  - Thumb, Thumb-2
  - Hardware multiply, Hardware divide, saturated arithmetic
  - DSP extensions:
    - Single cycle 16/32-bit MAC
    - Single cycle dual 16-bit MAC
    - 8/16-bit SIMD arithmetic
  - FPU (VFPv4-SP)

5.2.3 Software point of view

In addition to the Cortex®-M3, the Cortex®-M4 provides:

- SIMD, or Single Instruction Multiple Data, operations
- Additional fast MAC and multiply instructions
- Saturating arithmetic instructions
- Single precision FPU, or Floating Point Unit, instructions

This means on software point of view, the Cortex®-M3 software can be run on the Cortex®-M4.

To improve and speed-up the STM32L1 software on the new STM32L4 platforms, do not forget to switch on the FPU. This can be done on makefile side or using software development tools:

- On Keil® µVision®
  - On “Project” open “Option for Target”
  - Go to “Target”...“Code Generation”
  - Set “Floating Point Hardware” to “Use Single Precision”
- On IAR Systems®
  - On “Project” open “Options...”
  - Go to “General Options”...“Target”
  - Set “Floating point settings”, ‘FPU” to “VFPv4 single precision”

5.3 Cortex mapping overview

Except for the Floating Point Unit, the mapping is similar on the Cortex®-M3 and the Cortex®-M4.

Table 31. Cortex overview mapping for STM32L1 Series and STM32L4 Series / STM32L4+ Series

		STM32L1 Series	STM32L4 Series / STM32L4+ Series
Core	Architecture	Cortex®-M3	Cortex®-M4
	Nested vectored interrupt controller (NVIC)	57 maskable interrupt channels	Maskable interrupt channel: – 94 (STM32L4+ Series) – 91 (STM32L49xxx/4Axxx) – 82 (STM32L47xxx/48xxx) – 67 (STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)
	Extended interrupts and events controller (EXTI)	Up to 24 event/interrupt	– Up to 41 event/interrupt (STM32L4+ Series, STM32L49xxx/4Axxx) – Up to 40 event/interrupt (STM32L47xxx/48xxx) – Up to 37 event/interrupt (STM32L45xxx/46xxx, STM32L43xxx/44xxx and STM32L41xxx/42xxx)

Table 31. Cortex overview mapping for STM32L1 Series and STM32L4 Series / STM32L4+ Series (continued)

		STM32L1 Series	STM32L4 Series / STM32L4+ Series
Mapping	System timer	0xE000 E010 to 0xE000 E01F	0xE000 E010 to 0xE000 E01F
	Nested vectored interrupt controller	0xE000E100 to 0xE000E4EF	0xE000E100 to 0xE000E4EF
	System control block	0xE000 ED00 to 0xE000 ED3F	0xE000 ED00 to 0xE000 ED3F
	Floating point unit coprocessor access control	NA	0xE000 ED88 to 0xE000 ED8B
	Memory protection unit	0xE000 ED90 to 0xE000 EDB8	0xE000 ED90 to 0xE000 EDB8
	Nested vectored interrupt controller	0xE000 EF00 to 0xE000 EF03	0xE000 EF00 to 0xE000 EF03
	Floating point unit	NA	0xE000 EF30 to 0xE000 EF44

6 Revision history

Table 32. Document revision history

Date	Revision	Changes
16-Jul-2015	1	Initial release.
23-Nov-2015	2	*Section 4.2: Memory mapping updated:* Stop 0 mode added for content preservation. *Table 14: PWR differences between the STM32L1 Series and the STM32L4 Series* updated: Stop 0 mode added.
04-Mar-2016	3	*Section 1: STM32L4 Series overview*: added category 2 and 4 for STLM32L4.
20-Feb-2017	4	Updated: – *Introduction*: STM32L4 Series reference manuals. – *Section 1: STM32L4 Series and STM32L4+ Series overview*. – Cat. 2 devices replaced by STM32L49xxx/4Axxx and STM32L47xxx/48xxx devices. – Cat. 4 devices replaced bySTM32L45xxx/46xxx and STM32L43xxx/44xxx devices. – *Table 6*, *Table 7*, *Table 8*, *Table 9*, *Table 10*, *Table 17*, *Table 26*, *Table 27*. – *Section 4.2: Memory mapping*. – *Section 2: Hardware migration*: *Table 1* and SMPS data. – *Section 4.5.3: Peripheral clock configuration*. Added: *Section 5: Software migration*. Removed Table *Product category overview.*
31-Aug-2017	5	Updated the whole document to include the information about STM32L4+ Series products.

Table 32. Document revision history (continued)

Date	Revision	Changes
11-Apr-2018	6	Updated: – *Table 7: Bootloader interfaces on STM32L1 and STM32L4 Series / STM32L4+ Series* – DAC naming: 1 DAC with 2 channels instead of 2 DACs
20-Sep-2018	7	Added – Information related to STM32L41xxx/42xxx to the whole document – *Table 1: STM32L4 Series / STM32L4+ Series memory availability* Updated – Cover page – *Section 1: STM32L4 Series and STM32L4+ Series overview* – *Section 3: Boot mode selection* – *Section 4.2: Memory mapping* – *Section 4.5.3: Peripheral clock configuration* – *Section 4.17: Liquid-crystal display controller (LCD)* – *Section 4.18: Universal serial bus interface (USB)* – *Section 5.1: References* – *Recommendations to migrate from the STM32L1 Series board to the STM32L4 Series and STM32L4+ Series board on page 15* – *Table 2: Packages available on STM32L4 Series and STM32L4+ Series* – *Table 3: Pinout differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (QFP)* – *Table 4: Pinout differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series (BGA)* – *Table 6: Boot modes for STM32L4+ Series, STM32L49xxx/4Axxx, STM32L45xxx/46xxx,, STM32L43xxx/44xxx and STM43L41xxx/42xxx devices* – *Table 7: Bootloader interfaces on STM32L1 and STM32L4 Series / STM32L4+ Series* – *Table 8: Peripheral compatibility analysis between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 9: Peripheral address mapping differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 12: RCC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 15: PWR differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 18: EXTI differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 19: Flash memory differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 21: I2C differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 27: ADC differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 30: OPAMP differences between STM32L1 Series and STM32L4 Series / STM32L4+ Series* – *Table 31: Cortex overview mapping for STM32L1 Series and STM32L4 Series / STM32L4+ Series*