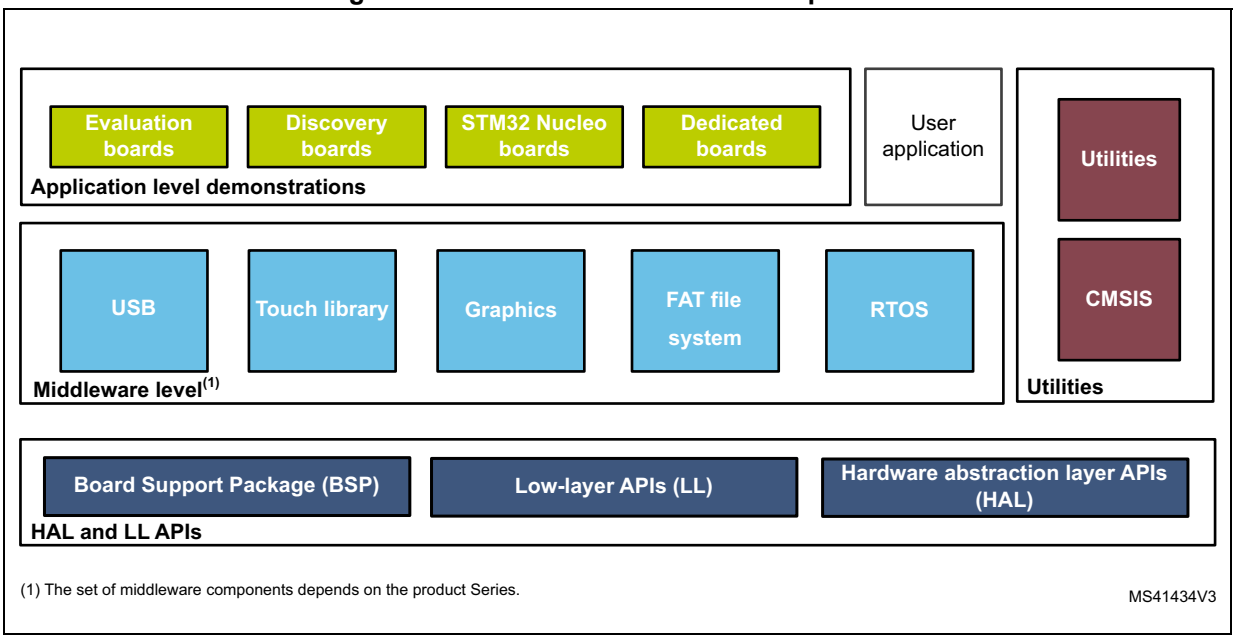
AN4734
Application note

STM32Cube firmware examples for STM32F3 Series

Introduction

The STM32CubeF3 firmware package comes with a rich set of examples running on STMicroelectronics boards. The examples are organized by board and provided with preconfigured projects for the main supported toolchains (see *Figure 1* and *Table 1*).

Figure 1. STM32CubeF3 firmware components



(1) The set of middleware components depends on the product Series.

MS41434V3

Contents

Reference documents

The reference documents are available on www.st.com/stm32cubefw:

- Latest release of STM32CubeF3 firmware package
- *Getting started with the STM32CubeF3 for STM32F3 Series* (UM1766)
- *Description of STM32F3 HAL and low-layer drivers* (UM1786)
- *STM32CubeF3 Nucleo demonstration firmware* (UM1784)
- *STM32Cube USB host library* (UM1720)
- *STM32Cube USB device library* (UM1734)
- *Developing Applications on STM32Cube with FatFS* (UM1721)
- *Developing Applications on STM32Cube with RTOS* (UM1722)

STM32CubeF3 examples

The examples are classified depending on the STM32Cube™ level they apply to. They are named as follows:

- **Examples**

  These examples use only the HAL and BSP drivers (the middleware is not used). Their objective is to demonstrate the product/peripherals features and usage. They are organized per peripheral (one folder for each peripheral, e.g. TIMER). Their complexity level ranges from the basic usage of a given peripheral (e.g. PWM generation using timer) to the integration of several peripherals (e.g. how to use DAC for signal generation with synchronization from TIM6 and DMA). The usage of the board resources is reduced to the strict minimum.

- **Examples_LL**

  These examples use only the LL drivers (HAL drivers and middleware components not used). They offer an optimum implementation of typical use cases of the peripheral features and configuration sequences. The examples are organized per peripheral (one folder for each peripheral, e.g. TIM) and run exclusively on Nucleo board.

- **Examples_MIX**

  These examples use only HAL, BSP and LL drivers (middleware components not used). They aim at demonstrating how to use both HAL and LL APIs in the same application to combine the advantages of both APIs:

  – HAL offers high-level function-oriented APIs with high portability level by hiding product/IPs complexity for end users.

  – LL provides low-level APIs at register level with better optimization.

  The examples are organized per peripheral (one folder for each peripheral, e.g. TIM) and run exclusively on Nucleo board.

- **Applications**

  The applications demonstrate the product performance and how to use the available middleware stacks. They are organized either by middleware (one folder per middleware, e.g. USB Host) or by product feature that require high-level firmware bricks (e.g. Audio). The integration of applications that use several middleware stacks is also supported.

- **Demonstrations**

  The demonstrations aim at integrating and running the maximum number of peripherals and middleware stacks to showcase the product features and performance.

- **Template project**

  The template project is provided to allow the user to quickly build a firmware application using HAL and BSP drivers on a given board.

- **Template_LL project**

  The template_LL project is provided to allow the user to quickly build a firmware application using LL drivers on a given board.

The examples are located under *STM32Cube_FW_STM32CubeF3_VX.Y.Z\Projects*\. They all have the same structure:

- *Inc* folder containing all header files
- *Src* folder containing the sources code
- *EWARM*, *MDK-ARM*, and \SW4STM32 folders containing the preconfigured project for each toolchain.
- readme.txt file describing the example behavior and the environment required to run the example.

To run an example, proceed as follows:

1. Open the example using your preferred toolchain.
2. Rebuild all files and load the image into target memory.
3. Run the example by following the readme.txt instructions

Note: Refer to "Development toolchains and compilers" and "Supported devices and evaluation boards" sections of the firmware package release notes to know more about the software/hardware environment used for the firmware development and validation. The correct operation of the provided examples is not guaranteed in other environments, for example when using different compiler or board versions.

The examples can be tailored to run on any compatible hardware: simply update the BSP drivers for your board, provided it has the same hardware functions (LED, LCD display, pushbuttons, etc.). The BSP is based on a modular architecture that can be easily ported to any hardware by implementing the low-level routines.

Table 1 contains the list of examples provided within STM32CubeF3 firmware package.

The board mnemonics used in the column headers of *Table 1* are further used as such in the firmware package. The correspondence with STMicroelectronics board references is as follows:

- STM32F302R8-Nucleo: NUCLEO-F302R8
- STM32303E_EVAL: STM32303E-EVAL
- STM32303C_EVAL: STM32303C-EVAL
- STM32F303ZE-Nucleo: NUCLEO-F303ZE
- STM32F303K8-Nucleo: NUCLEO-F303K8
- STM32F3-Discovery: STM32F3DISCOVERY
- STM32F3348-Discovery: 32F3348DISCOVERY
- STM32F334R8-Nucleo: NUCLEO-F334R8
- STM32373C_EVAL: STM32373C-EVAL
- STM32F303RE-Nucleo: NUCLEO-F303RE

Table 1. STM32CubeF3 firmware examples

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Templates_LL	-	Starter project	This project provides a reference template through the LL API that can be used to build any firmware application.	X	X	X	X	X	X	X	X	X	X
Total number of templates_LL: 10				**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Templates	-	Starter project	This project provides a reference template that can be used to build any firmware application.	X	X	X	X	X	X	X	X	X	X
Total number of templates: 10				**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Examples	-	BSP	This example describes the way to use the different BSP drivers.	-	X	-	-	-	X	-	X	-	X
	ADC	ADC_Analog Watchdog	This example provides a short description of the way to use the ADC peripheral to perform conversions with analog watchdog and out-of-window interruptions enabled.	X	-	-	-	X	-	-	-	X	-
		ADC_Conv_ Differential	This example provides a short description of the way to use ADC peripherals to perform a conversion in Differential mode, between 2 ADC channels.	-	-	-	-	-	-	-	-	-	X
		ADC_DMA_ Transfer	This example describes how to configure and use the ADC to convert an external analog input and get the result using a DMA transfer through the HAL API.	-	-	-	-	-	-	X	-	-	-
		ADC_ DifferentialMode	This example provides a short description of the way to use ADC peripherals to perform a conversion in Differential mode, between 2 ADC channels.	-	X	-	-	-	-	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	ADC	ADC_ DualMode Interleaved	This example provides a short description of the way to use two ADC peripherals to perform conversions in interleaved dual-mode.	-	-	-	-	-	-	-	-	-	X
		ADC_ DualMode_18Msps	This example provides a short description of the way to use two ADC peripherals to convert a regular channel in Dual interleaved mode.	-	X	-	-	-	-	-	-	-	X
		ADC_Regular Conversion_ Polling	This example describes how to use the ADC in Polling mode to convert data through the HAL API.	-	-	-	-	-	X	-	-	-	X
		ADC_ Sequencer	This example provides a short description of the way to use the ADC peripheral with sequencer to convert several channels.	X	-	-	-	X	-	-	-	X	-
		ADC_Trigger Mode	This example describes how to use the ADC and TIM2 to continuously convert data from ADC channel. Each time an external trigger is generated by TIM2 a new conversion is started by the ADC.	-	X	-	-	-	X	-	-	-	X
	CAN	CAN_ Networking	This example shows how to configure the CAN peripheral to send and receive CAN frames in normal mode.	-	-	-	-	-	X	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	CEC	CEC_Data Exchange	This example shows how to configure and use the CEC peripheral to receive and transmit messages.	-	-	-	-	-	X	-	-	-	-
		CEC_Listen Mode	This example shows how to configure and use the CEC peripheral to receive and transmit messages between two boards while a third one (the spy device) listens but does not acknowledge the received messages.	-	-	-	-	-	X	-	-	-	-
		CEC_Multi Address	This example shows how to configure and use the CEC peripheral to receive and transmit messages in the case where one device supports two distinct logical addresses at the same time.	-	-	-	-	-	X	-	-	-	-
	COMP	COMP_ Analog Watchdog	This example shows how to make an analog watchdog using the COMP peripherals in window mode.	-	-	-	-	-	-	-	-	-	X
		COMP_ Hygrometer	This example shows how to make an hygrometer using the capacitive humidity sensor. The capacitance measurement is performed by continuously charging/discharging the humidity sensor and measuring the associated time constant.	-	X	-	-	-	-	-	-	-	X
		COMP_ Output Blanking	This example shows how to use the output blanking feature of COMP peripheral.	-	X	-	-	-	-	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	CRC	CRC_Bytes_ Stream_7bit_CRC	This example guides the user through the different configuration steps by means of the HAL API. The CRC (Cyclic Redundancy Check) calculation unit computes 7-bit long CRC codes derived from buffers of 8-bit data (bytes).	-	-	-	X	-	-	X	-	-	X
		CRC_Data_ Reversing_ 16bit_CRC	This example guides the user through the different configuration steps by means of the HAL API. The CRC (Cyclic Redundancy Check) calculation unit computes a 16-bit long CRC code derived from a buffer of 8-bit data (bytes).	-	-	-	X	-	-	X	-	-	-
		CRC_ Example	This example guides the user through the different configuration steps by means of the HAL API. The CRC (Cyclic Redundancy Check) calculation unit computes the CRC code of a given buffer of 32-bit data words, using a fixed generator polynomial (0x4C11DB7).	-	-	-	-	-	X	X	-	X	X
	CRC	CRC_User Defined Polynomial	This example guides the user through the different configuration steps by means of the HAL API. The CRC (Cyclic Redundancy Check) calculation unit computes the 8-bit long CRC code of a given buffer of 32-bit data words, based on a user-defined generating polynomial.	-	-	-	-	-	-	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	CORTEX	CORTEXM_ MPU	This example presents the MPU feature. Its purpose is to configure a memory area as privileged read-only area and attempt to perform read and write operations in different modes.	-	-	-	-	-	X	-	X	X	X
		CORTEXM_ ModePrivilege	This example shows how to modify Thread mode privilege access and stack. Thread mode is entered on reset or when returning from an exception.	-	-	-	-	-	X	-	X	-	X
		CORTEXM_ SysTick	This example shows how to use the default SysTick configuration with a 1 ms timebase to toggle LEDs.	-	-	X	-	-	X	-	X	X	X
	DAC	DAC_Signals Generation	This example provides a description of how to use the DAC peripheral to generate several signals using DMA controller.	X	X	-	-	-	-	-	-	-	X
		DAC_Simple Conversion	This example provides a short description of how to use the DAC peripheral to do a simple conversion.	X	X	-	-	-	-	-	-	-	X
	DMA	DMA_FLASHTo RAM	This example provides a description of how to use a DMA to transfer a word data buffer from Flash memory to embedded SRAM through the HAL API.	X	X	-	-	X	X	X	-	X	X
	FLASH	FLASH_ EraseProgram	This example describes how to configure and use the FLASH HAL API to erase and program the internal Flash memory.	X	X	X	-	X	X	X	X	X	X
		FLASH_ WriteProtection	This example describes how to configure and use the FLASH HAL API to enable and disable the write protection of the internal Flash memory.	-	X	X	-	X	X	-	X	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	GPIO	GPIO_EXTI	This example shows how to configure external interrupt lines.	X	X	X	-	X	X	-	X	X	X
		GPIO_ IOToggle	This example describes how to configure and use GPIOs through the HAL API.	X	X	X	X	X	X	X	X	X	X
	HAL	HAL_TimeBase_ RTC_ALARM	This example describes how to customize the HAL time base using RTC alarm instead of Systick as main source of time base. The User push-button will be used to Suspend or Resume tick increment.	-	X	X	-	X	X	-	X	-	X
		HAL_TimeBase_ RTC_WKUP	This example describes how to customize the HAL time base using RTC wakeup instead of Systick as main source of time base. The User push-button will be used to Suspend or Resume tick increment.	-	X	X	-	X	X	-	X	-	X
		HAL_TimeBase_ TIM	This example describes how to customize the HAL time base using a general purpose timer instead of Systick as main source of time base.	-	X	X	-	X	X	-	X	-	X
	HRTIM	HRTIM_ BasicPWM	This example describes how to generate basic PWM waveforms with the HRTIM, using HRTIM Cookbook basic examples.	-	-	X	-	-	-	-	-	-	-
		HRTIM_ BuckBoost	This example shows how to configure the HRTIM to control a non-inverting buck-boost converter timer.	-	-	X	-	-	-	-	-	-	-
		HRTIM_Buck Boost_AN4449	This example is related to AN4449 buck-boost converter.	-	-	X	-	-	-	-	-	-	-
		HRTIM_BuckSync Rect	This example shows how to configure the HRTIM to control a buck converter with synchronous rectification.	-	-	X	-	-	-	-	-	-	-
		HRTIM_ DualBuck	This example shows how to configure the HRTIM to have 2 buck converters controlled by a single timer unit.	-	-	X	-	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples	HRTIM	HRTIM_LLC_HalfBridge	This example shows how to configure the HRTIM to control an half-bridge LLC converter with synchronous rectification, using timer units A and B and TA1/TA2/TB1/TB2 outputs.	-	-	X	-	-	-	-	-	-	-
		HRTIM_Multiphase	This example shows how to configure the HRTIM to control a multiphase buck converter. It handles here 5 phases on timer unit A, B C and D and outputs TA2, TB1, TC2, TD1, TD2.	-	-	X	-	-	-	-	-	-	-
		HRTIM_Snippets	This example describes how to generate basic PWM waveforms with the HRTIM, using HRTIM Cookbook basic examples.	-	-	X	-	-	-	-	-	-	-
		HRTIM_TM_PFC	This example shows how to configure the HRTIM to control a transition mode PFC.	-	-	X	-	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	I2C	I2C_EEPROM	This example describes how to ensure I^2C data buffer transmission and reception with DMA. The communication is done with an I2C EEPROM memory.	-	X	-	-	-	X	-	-	-	X
		I2C_ TwoBoards_ AdvComIT	This example describes how to perform I^2C data buffer transmission/reception between two boards, using an interrupt.	X	X	-	-	X	-	X	X	X	X
		I2C_ TwoBoards_ ComDMA	This example describes how to perform I^2C data buffer transmission/reception between two boards, via DMA.	X	X	-	-	X	-	X	X	X	X
		I2C_ TwoBoards_ ComIT	This example describes how to perform I^2C data buffer transmission/reception between two boards using an interrupt.	X	X	-	-	X	-	X	X	X	X
		I2C_ TwoBoards_ ComPolling	This example describes how to perform I^2C data buffer transmission/reception between two boards in Polling mode.	X	X	-	-	X	-	X	X	X	X
		I2C_ TwoBoards_ RestartAdv ComIT	This example describes how to perform a multiple I^2C data buffer transmission/reception between two boards in Interrupt mode and with a restart condition.	-	-	-	-	X	-	X	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	I2C	I2C_ TwoBoards_ RestartComIT	This example describes how to perform a single I^2C data buffer transmission/reception between two boards in Interrupt mode and with a restart condition.	-	-	-	-	X	-	X	-	-	-
		I2C_WakeUp FromStop	This example describes how to perform I^2C data buffer transmission/reception between two boards using an interrupt when the device is in STOP mode.	X	X	-	-	X	-	X	X	X	X
	I2S	I2S_Audio	This example provides basic implementation of audio features.	-	X	-	-	-	X	-	-	-	X
	IWDG	IWDG_Reset	This example describes how to ensure IWDG reload counter and simulate a software fault that generates an MCU IWDG reset when a programmed time period has elapsed.	-	-	-	-	-	X	-	X	-	X
		IWDG_ WindowMode	This example shows how to periodically update the IWDG reload counter and simulate a reload outside the window that generates an MCU IWDG reset.	-	-	-	-	-	X	-	X	-	X
	OPAMP	OPAMP_ CALIBRATION	This example shows how to calibrate the OPAMP.	-	-	-	-	-	-	-	-	-	X
		OPAMP_PGA	This example shows how to use the built-in PGA mode (OPAMP programmable gain).	-	-	-	-	-	-	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	PWR	PWR_Current Consumption	This example shows how to configure the system to measure the current consumption in different low-power modes.	X	X	X	X	X	X	-	X	-	X
		PWR_PVD	This example shows how to configure the programmable voltage detector using an external interrupt line.	X	X	X	-	X	X	X	X	-	X
		PWR_SLEEP	This example shows how to enter Sleep mode and wake up from this mode by using an interrupt.	-	-	-	-	-	-	-	-	X	-
		PWR_ STANDBY	This example shows how to enter the system in STANDBY mode and wake-up from this mode using external RESET or WKUP pin.	-	-	-	-	-	-	-	X	X	-
		PWR_STOP	This example shows how to enter Stop mode and wake up from this mode by using the RTC Wakeup timer event or an interrupt.	-	-	-	-	-	-	-	X	X	-
	RCC	RCC_ ClockConfig	This example describes how to use the RCC HAL API to configure the system clock (SYSCLK) and modify the clock settings on run time.	X	X	X	-	X	X	-	X	X	X
		RCC_ LSIConfig	This example describes how to use the RCC HAL API to enable or disable the low-speed internal (LSI) RC oscillator (about 40 KHz) at run time.	X	-	-	-	X	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	RTC	RTC_Calendar	This example guides the user through the different configuration steps by mean of HAL API to configure the RTC calendar.	X	-	-	-	-	X	X	X	X	X
		RTC_Tamper	This example guides the user through the different configuration steps by means of the RTC HAL API to write/read data to/from RTC Backup registers. It also demonstrates the tamper detection feature.	-	-	-	-	-	X	X	-	X	X
	SDADC	SDADC_ Pressure Measurement	This example aims to show how to use the 16-bit resolution Sigma-Delta Analog-to-Digital converter to perform differential pressure measurement.	-	-	-	-	-	X	-	-	-	-
		SDADC_TempMeasurement	This example aims to show how to use the 16-bit resolution Sigma-Delta Analog-to-Digital converter to perform accurate temperature measurement.	-	-	-	-	-	X	-	-	-	-
		SDADC_ Voltmeter	This example aims to show how to use the 16-bit resolution Sigma-Delta Analog-to-Digital converter to perform precise voltage measurement when input voltage range is between 0V and V_{REF}/gain.	-	-	-	-	-	X	-	-	-	-
	SMBUS	SMBUS_ TSENSOR	This example shows how to ensure SMBUS Data buffer transmission and reception with IT. The communication is done with an SMBUS temperature sensor.	-	X	-	-	-	-	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	SPI	SPI_FullDuplex_ AdvComIT	This example guides the user through the different configuration steps by mean of HAL API to transmit/receive SPI data buffer in Interrupt mode and in an advanced communication mode: The master board always sends the command to the slave before any transmission is performed.	-	-	-	-	X	-	-	-	-	-
		SPI_ FullDuplex_ AdvComPolling	This example guides the user through the different configuration steps by mean of HAL API to transmit/receive SPI data buffer in Polling mode and in an advanced communication mode: the master board always sends the command to the slave before any transmission is performed.	-	-	-	-	-	-	-	-	X	-
		SPI_ FullDuplex_ ComDMA	This example shows how to perform SPI data buffer transmission/reception between two boards via DMA.	X	X	-	-	X	-	X	X	X	-
		SPI_ FullDuplex_ComIT	This example shows how to ensure SPI data buffer transmission/reception between two boards by using an interrupt.	X	X	-	-	X	-	X	X	X	-
		SPI_ FullDuplex_ ComPolling	This example shows how to ensure SPI data buffer transmission/reception in Polling mode between two boards.	X	X	-	-	X	-	X	X	X	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	SPI	SPI_HalfDuplex_ ComPolling	This example shows how to ensure SPI data buffer half-duplex transmission/reception in Polling mode between two boards.	-	-	-	-	-	-	-	-	X	-
	TIM	TIM_Asymetric	This example shows how to configure the TIM peripheral to generate an asymmetric signal.	-	X	-	-	-	-	-	X	-	X
		TIM_Combined	This example shows how to configure the TIM peripheral to generate 3 PWM combined signals.	-	X	-	-	-	-	-	X	-	X
		TIM_ Complementary Signals	This example shows how to configure the TIM peripheral to generate three complementary TIM signals, to insert a defined dead time value, to use the break feature and to lock the desired parameters.	-	X	-	-	-	X	-	X	-	X
		TIM_DMA	This example provides a description of how to use DMA with TIMER Update request to transfer Data from memory to TIMER Capture Compare Register 3 (TIMx_CCR3).	X	X	-	-	X	X	X	X	X	X
		TIM_Input Capture	This example shows how to use the TIM peripheral to measure the frequency of an external signal.	-	X	-	-	-	X	X	X	X	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	TIM	TIM_OCActive	This example shows how to configure the TIM peripheral in Output Compare Active mode (when the counter matches the capture/compare register, the concerned output pin is set to its active state).	-	-	-	-	-	-	-	-	X	-
		TIM_OCToggle	This example shows how to configure the TIM peripheral to generate four different signals with four different frequencies.	-	-	-	-	-	-	-	-	X	-
		TIM_OnePulse	This example shows how to use the TIM peripheral to generate a One pulse mode after a rising edge of an external signal is received in Timer Input pin.	-	-	-	-	-	-	-	-	X	-
		TIM_PWMInput	This example shows how to use the TIM peripheral to measure the frequency and duty cycle of an external signal.	X	X	-	-	X	X	-	X	X	X
		TIM_ PWMOutput	This example shows how to configure the TIM peripheral in PWM (Pulse Width Modulation) mode.	X	X	-	-	X	X	X	X	X	X
		TIM_TimeBase	This example shows how to configure the TIM peripheral to generate a time base of one second with the corresponding Interrupt request.	X	X	-	-	X	X	X	X	X	X
	TSC	TSC_Basic Acquisition_ Interrupt	This example describes how to use the TSC to perform an acquisition of two channels in interrupt mode.	-	X	-	-	-	X	-	-	-	X
		TSC_Basic Acquisition_ Polling	This example describes how to use the TSC to perform continuous acquisitions of one channel in polling mode.	-	X	-	-	-	X	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples	UART	UART_ HyperTerminal_ DMA	This example describes an UART transmission (transmit/receive) in DMA mode between a board and an HyperTerminal PC application.	-	X	-	-	-	X	X	-	-	X
		UART_Printf	This example shows how to re-route the C library printf function to the UART.	-	-	-	-	-	-	-	-	X	-
		UART_ TwoBoards_ ComDMA	This example describes an UART transmission (transmit/receive) in DMA mode between two boards.	-	-	X	-	X	-	X	X	X	-
		UART_ TwoBoards_ ComIT	This example describes an UART transmission (transmit/receive) in interrupt mode between two boards.	-	-	X	-	X	-	X	X	X	-
		UART_ TwoBoards_ ComPolling	This example describes an UART transmission (transmit/receive) in polling mode between two boards.	-	-	X	-	X	-	X	X	X	-
	UART	UART_WakeUp FromStop	This example shows how to configure an UART to wake up the MCU from Stop mode when a given stimulus is received.	-	X	X	-	X	X	X	X	-	X
	WWDG	WWDG_ Example	This example guides the user through the different configuration steps by means of the HAL API to perform periodic WWDG counter update and simulate a software fault that generates an MCU WWDG reset when a predefined time period has elapsed.	X	X	-	-	-	X	X	X	X	X
Total number of examples: 337				**26**	**43**	**24**	**4**	**33**	**42**	**30**	**40**	**38**	**57**

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	ADC	ADC_Analog Watchdog	This example describes how to use an ADC peripheral with ADC analog watchdog to monitor a channel and detect when the corresponding conversion data are out of window thresholds. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
Examples_ LL	ADC	ADC_ Continuous Conversion_ TriggerSW	This example describes how to use an ADC peripheral to perform continuous ADC conversions of a channel, from a software start. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_ Continuous Conversion_ TriggerSW_Init	This example describes how to use an ADC peripheral to perform continuous ADC conversions of a channel, from a software start. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_ Continuous Conversion_ TriggerSW_ LowPower	This example describes how to use an ADC peripheral with ADC low-power features. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	ADC	ADC_GroupsRegularInjected	This example describes how to use an ADC peripheral with both ADC groups (ADC group regular and ADC group injected) in their intended use case. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_MultiChannelSingleConversion	This example describes how to use an ADC peripheral to convert several channels. ADC conversions are performed successively in a scan sequence. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_Multimode DualInterleaved	This example describes how to use several ADC peripherals in multimode, mode interleaved. This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	ADC	ADC_Single Conversion_TriggerSW	This example describes how to use an ADC peripheral to perform a single ADC conversion of a channel, at each software start. Example using programming model: polling (for programming models interrupt or DMA transfer, refer to other examples). This example is based on the STM32F3xx ADC LL API. Peripheral initialization done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_Single Conversion_TriggerSW_DMA	This example describes how to use an ADC peripheral to perform a single ADC conversion of a channel, at each software start. Example using programming model: DMA transfer (for programming models polling or interrupt, refer to other examples). This example is based on the STM32F3xx ADC LL API. Peripheral initialization done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	ADC	ADC_SingleConversion_TriggerSW_IT	This example describes how to use an ADC peripheral to perform a single ADC conversion of a channel, at each software start. Example using programming model: interrupt (for programming models polling or DMA transfer, refer to other examples). This example is based on the STM32F3xx ADC LL API. Peripheral initialization done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		ADC_SingleConversion_TriggerTimer_DMA	This example describes how to use an ADC peripheral to perform a single ADC conversion of a channel, at each trigger event from timer. Conversion data are transferred by DMA into a table, indefinitely (circular mode). This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	ADC	ADC_Temperature Sensor	This example describes how to use an ADC peripheral to perform a single ADC conversion of the internal temperature sensor and to calculate the temperature in Celsius degrees. Example using programming model: polling (for programming models interrupt or DMA transfer, refer to other examples). This example is based on the STM32F3xx ADC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	COMP	COMP_CompareGpioVsVrefInt_IT	This example describes how to use a comparator peripheral to compare a voltage level applied on a GPIO pin with the internal voltage reference (V_{REFINT}), in interrupt mode. This example is based on the STM32F3xx COMP LL AP. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	COMP	COMP_ CompareGpio VsVrefInt_IT_ Init	This example describes how to use a comparator peripheral to compare a voltage level applied to a GPIO pin versus the internal voltage reference (V_{REFINT}), in interrupt mode. This example is based on the STM32F3xx COMP LL API. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
		COMP_ CompareGpio VsVrefInt_ OutputGpio	This example describes how to use a comparator peripheral to compare a voltage level applied on a GPIO pin versus the internal voltage reference (V_{REFINT}). The comparator output is connected to a GPIO. This example is based on the STM32F3xx COMP LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	CORTEX	CORTEX_MPU	This example presents the MPU feature. Its purpose is to configure a memory area as privileged read-only area and attempt to perform read and write operations in different modes.	-	-	-	-	-	-	-	-	X	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	CRC	CRC_CalculateAndCheck	This example shows how to configure CRC calculation unit to get a CRC code of a given data buffer, based on a fixed generator polynomial (default value 0x4C11DB7). Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		CRC_User Defined Polynomial	This example shows how to configure and use CRC calculation unit to get a 8-bit long CRC of a given data buffer, based on a user-defined generating polynomial. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	DAC	DAC_GenerateConstantSignal_TriggerSW	This example describes how to use the DAC peripheral to generate a constant voltage signal. This example is based on the STM32F3xx DAC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		DAC_GenerateWaveform_TriggerHW	This example describes how to use the DAC peripheral to generate a waveform voltage from digital data stream transfered by DMA. This example is based on the STM32F3xx DAC LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_EVAL
Examples_LL	DAC	DAC_Generate Waveform_TriggerHW_Init	This example describes how to use the DAC peripheral to generate a waveform voltage from digital data stream transfered by DMA. This example is based on the STM32F3xx DAC LL API. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
	DMA	DMA_Copy FromFlashTo Memory	This example describes how to use a DMA channel to transfer a word data buffer from Flash memory to embedded SRAM. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		DMA_Copy FromFlashTo Memory_Init	This example describes how to use a DMA channel to transfer a word data buffer from Flash memory to embedded SRAM. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
	EXTI	EXTI_Toggle LedOnIT	This example describes how to configure the EXTI and use GPIOs using the STM32F3xx LL API to toggle the available user LEDs on the board when User button is pressed. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		EXTI_Toggle LedOnIT_Init	This example describes how to configure the EXTI and use GPIOs using the STM32F3xx LL API to toggle the available user LEDs on the board when User button is pressed. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	GPIO	GPIO_InfiniteLedToggling	This example describes how to configure and use GPIOs through the LL API to toggle the available user LEDs on the board each 250 ms. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		GPIO_InfiniteLedToggling_Init	This example describes how to configure and use GPIOs through the LL API to toggle the available user LEDs on the board each 250 ms. Peripheral initialization is one using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
	HRTIM	HRTIM_BuckBoost	This example shows how to configure the HRTIM to control a non-inverting buck-boost converter timer.	-	-	X	-	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	I2C	I2C_OneBoard_ Adv Communication_ DMAAndIT	This example describes how to exchange some data between an I^2C Master device using DMA mode and an I^2C Slave device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	-	X	-	-	-	-	-
		I2C_OneBoard_ Communication_ DMAAndIT	This example describes how to transmit some data bytes from an I^2C Master device using DMA mode to an I^2C Slave device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	-	X	-	-	-	-	-
		I2C_OneBoard_ Communication_ IT	This example describes how to receive data byte from an I^2C Slave device using IT mode to an I^2C Master device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	-	X	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	I2C	I2C_OneBoard_ Communication_ IT_Init	This example describes how to receive data byte from an I^2C Slave device using IT mode to an I^2C Master device using IT mode. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	-	X	-	-	-	-	-
		I2C_OneBoard_ Communication_ PollingAndIT	This example describes how to transmit data bytes from an I^2C Master device using Polling mode to an I^2C Slave device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	-	X	-	-	-	-	-
		I2C_TwoBoards_ MasterRx_ SlaveTx_IT	This example describes how to receive data byte from an I^2C Slave device using IT mode to an I^2C Master device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		I2C_TwoBoards_ MasterTx_ SlaveRx	This example describes how to transmit some data bytes from an I^2C Master device using Polling mode to an I^2C Slave device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		I2C_TwoBoards_ MasterTx_ SlaveRx_DMA	This example describes how to transmit some data bytes from an I^2C Master device using DMA mode to an I^2C Slave device using DMA mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	I2C	I2C_TwoBoards_ WakeUpFromStop _IT	This example describes how to receive data byte from an I^2C Slave device in Stop mode using IT mode to an I^2C Master device using IT mode. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	IWDG	IWDG_Refresh UntilUserEvent	This example describes how to configure the IWDG and ensure counter updates at regular period and generating an MCU IWDG reset at User Button pressed. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	OPAMP	OPAMP_PGA	This example describes how to use an operational amplifier peripheral in PGA mode (programmable gain amplifier). To test OPAMP in this example, a voltage waveform is generated by the DAC peripheral and can be connected to OPAMP input. This example is based on the STM32F3xx OPAMP LL AP. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		OPAMP_PGA_ Init	This example describes how to use an operational amplifier peripheral in PGA mode (programmable gain amplifier). To test OPAMP, a voltage waveform is generated by the DAC peripheral and can be connected to OPAMP input. This example is based on the STM32F3xx OPAMP LL API. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	PWR	PWR_Enter StandbyMode	This example shows how to enter the system in STANDBY mode and wake-up from this mode using external RESET or wake-up interrupt.	-	-	-	X	-	-	-	-	-	-
		PWR_Enter StopMode	This example shows how to enter the system in STOP_LPREGU mode.	-	-	-	X	-	-	-	-	-	-
	RCC	RCC_Output SystemClockOn MCO	This example describes how to configure MCO pin (PA8) to output the system clock.	-	-	-	X	-	-	-	-	-	-
		RCC_UseHSE asSystemClock	This example describes how to use the RCC LL API, how to start the HSE and use it as system clock.	-	-	-	X	-	-	-	-	-	-
		RCC_UseHSI_PL LasSystem Clock	This example shows how to modify the PLL parameters in run time.	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	RTC	RTC_Alarm	This example guides the user through the different configuration steps by mean of LL API to ensure Alarm configuration and generation using the RTC peripheral. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		RTC_Alarm_Init	This example guides the user through the different configuration steps by mean of LL API to ensure Alarm configuration and generation using the RTC peripheral. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
		RTC_Calendar	This example guides the user through the different configuration steps by mean of HAL API to configure the RTC calendar. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		RTC_Exit StandbyWith WakeUpTimer	This example shows how to configure the RTC in order to wake up from Standby mode using RTC Wakeup Timer. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		RTC_Tamper	This example guides the user through the different configuration steps by mean of LL API to ensure Tamper configuration using the RTC peripheral. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	RTC	RTC_TimeStamp	This example guides the user through the different configuration steps by mean of LL API to ensure Time Stamp configuration using the RTC peripheral. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	SPI	SPI_TwoBoards_FullDuplex_DMA	This example shows how to ensure SPI data buffer transmission and reception in DMA mode. The example is based on the STM32F3xx SPI LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		SPI_TwoBoards_FullDuplex_IT	This example shows how to ensure SPI data buffer transmission and reception in Interrupt mode. The example is based on the STM32F3xx SPI LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	TIM	TIM_BreakAndDeadtime	This example shows how to configure the TIMER to perform the following: – generate three center-aligned PWM and complementary PWM signals – insert a defined dead time value – use the break feature – lock the desired parameters This example is based on the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		TIM_DMA	This example provides a description of the way to use DMA with TIMER update request to transfer data from memory to TIMER Capture Compare Register 3 (TIMx_CCR3). Example using the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		TIM_InputCapture	This example shows how to use the TIM peripheral to measure the frequency of a periodic signal provided either by an external signal generator or by another timer instance. Example using the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	TIM	TIM_OnePulse	This example shows how to configure a timer to generate a positive pulse in Output Compare mode with a length of t_{PULSE} and after a delay of t_{DELAY}. This example is based on the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		TIM_Output Compare	This example shows how to configure the TIM peripheral to generate an output waveform in different output compare modes. Example using the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		TIM_PWM Output	This example describes how to use a timer peripheral to generate a PWM output signal and update PWM duty cycle. Example using the STM32F3xx TIM LL API. Peripheral initialization done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		TIM_PWM Output_Init	This example describes how to use a timer peripheral to generate a PWM output signal and update PWM duty cycle. Example using the STM32F3xx TIM LL API. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-
		TIM_TimeBase	This example shows how to configure the TIM peripheral to generate a time base. Example using the STM32F3xx TIM LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Examples_ LL	USART	USART_ Communication_ Rx_IT	This example shows how to configure GPIO and USART peripherals for receiving characters from HyperTerminal (PC) in Asynchronous mode using IT. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_ Communication_ Rx_IT_ Continuous	This example shows how to configure GPIO and USART peripherals for continuously receiving characters from HyperTerminal (PC) in Asynchronous mode using IT. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_ Communication_ Rx_IT_Init	This example shows how to configure GPIO and USART peripherals for receiving characters from HyperTerminal (PC) in Asynchronous mode using IT. Peripheral initialization is done using LL initialization function to demonstrate LL init usage.	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	USART	USART_Communication_Tx	This example shows how to configure GPIO and USART peripherals to send characters asynchronously to an HyperTerminal (PC) in Polling mode. If the transfer cannot be completed within the allocated time, a timeout allows to exit from the sequence with a Timeout error code. This example is based on STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_Communication_TxRx_DMA	This example shows how to configure GPIO and USART peripherals to send characters asynchronously to/from an HyperTerminal (PC) in DMA mode. This example is based on STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_Communication_Tx_IT	This example shows how to configure GPIO and USART peripherals to send characters asynchronously to HyperTerminal (PC) in Interrupt mode. This example is based on STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	USART	USART_HardwareFlowControl	This example shows how to configure GPIO and USART peripherals to receive characters asynchronously from HyperTerminal (PC) in Interrupt mode with Hardware Flow Control feature enabled. This example is based on STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_SyncCommunication_FullDuplex_DMA	This example shows how to configure GPIO, USART, DMA and SPI peripherals for transmitting bytes from/to an USART peripheral to/from an SPI peripheral (in slave mode) by using DMA mode through the STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
		USART_SyncCommunication_FullDuplex_IT	This example shows how to configure GPIO, USART, DMA and SPI peripherals for transmitting bytes from/to an USART peripheral to/from an SPI peripheral (in slave mode) by using IT mode through the STM32F3xx USART LL API (SPI is using DMA for receving/transmitting characters sent from/received by USART). Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_LL	USART	USART_WakeUpFromStop	This example shows how to configure GPIO and USART peripherals for allowing characters received on USART RX pin, to wake up MCU from low-power mode, using STM32F3xx USART LL API. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
	UTILS	UTILS_ConfigureSystemClock	This example describes how to use UTILS LL API to configure the system clock using PLL with HSI as source clock. The user application just needs to calculate PLL parameters using STM32CubeMX and call the UTILS LL API.	-	-	-	X	-	-	-	-	-	-
		UTILS_ReadDeviceInfo	This example describes how to Read UID, Device ID and Revision ID and save them into a global information buffer.	-	-	-	X	-	-	-	-	-	-
	WWDG	WWDG_RefreshUntilUserEvent	This example describes how to configure WWDG and update counter at regular period and generating an MCU WWDG reset at User Button pressed. Peripheral initialization is done using LL unitary services functions for optimization purpose (performance and size).	-	-	-	X	-	-	-	-	-	-
Total number of examples_LL: 74				**0**	**0**	**1**	**67**	**5**	**0**	**0**	**0**	**1**	**0**

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_MIX	ADC	ADC_SingleConversion_TriggerSW_IT	This example describes how to use an ADC peripheral to perform a single ADC conversion of a channel, at each software start; Example using programming model: interrupt (for programming models polling or DMA transfer, refer to other examples). This example is based on the STM32F3xx ADC HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-
	CRC	CRC_Polynomial Update	This example provides a description of the way to use CRC peripheral through the STM32F3xx CRC HAL & LL API (LL API used for performance improvement). The CRC (Cyclic Redundancy Check) calculation unit computes a 8-bit long CRC code of a given buffer of 32-bit data words, based on a user-defined generating polynomial. In this example, the polynomial is first set manually to 0x9B that is X^8 + X^7 + X^4 + X^3 + X + 1. Then, in a second step, generating polynomial value and length are updated (set to 0x1021 that is X^16 + X^12 + X^5 + 1), for new CRC calculation. These updates are performed using CRC LL API.	-	-	-	X	-	-	-	-	-	-
	DMA	DMA_FLASHToRAM	This example provides a description of how to use a DMA to transfer a word data buffer from Flash memory to embedded SRAM through the STM32F3xx DMA HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_MIX	I2C	I2C_OneBoard_ComSlave7_10bits_IT	This example describes how to perform I^2C data buffer transmission/reception between a master and 2 slaves with different address size (7-bit or 10-bit) through the STM32F3xx HAL & LL API (LL API used for performance improvement), using an interrupt.	-	-	-	-	X	-	-	-	-	-
	OPAMP	OPAMP_CALIBRATION	This example describes how to use an operational amplifier peripheral with OPAMP calibration and operation. This example is based on the STM32F3xx OPAMP HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-
	PWR	PWR_STANDBY_RTC	This example shows how to enter the system in STANDBY mode and wake-up from this mode using external RESET or RTC Wake-up Timer through the STM32F3xx RTC & RCC HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-
		PWR_STOP	This example shows how to enter the system in STOP with Low power regulator mode and wake-up from this mode using external RESET or wake-up interrupt (all the RCC functions calls use RCC LL API for footprint and performance improvements).	-	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Examples_MIX	TIM	TIM_6Steps	This example shows how to configure the TIM1 peripheral to generate 6 Steps PWM signal. The STM32F3xx TIM1 peripheral offers the possibility to program in advance the configuration for the next TIM1 outputs behaviour (step) and change the configuration of all the channels at the same time. This operation is possible when the COM (commutation) event is used. Example using the STM32F3xx TIM HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-
	UART	UART_HyperTerminal_IT	This example describes how to use an UART to transmit data (transmit/receive) between a board and an HyperTerminal PC application in Interrupt mode. This example provides a description of the way to use USART peripheral through the STM32F3xx UART HAL and LL API (LL API usage for performance improvement).	-	-	-	X	-	-	-	-	-	-
		UART_HyperTerminal_TxPolling_RxIT	This example describes how to use an UART to transmit data (transmit/receive) between a board and an HyperTerminal PC application both in Polling and Interrupt modes. This example provides a description of the way to use USART peripheral through the STM32F3xx UART HAL & LL API (LL API used for performance improvement).	-	-	-	X	-	-	-	-	-	-
Total number of examples_mix: 10				**0**	**0**	**0**	**9**	**1**	**0**	**0**	**0**	**0**	**0**

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Applications	EEPROM	EEPROM_Emulation	This application shows how to emulate EEPROM on internal Flash.	-	X	-	-	-	X	-	-	-	-
	FatFS	FatFs_uSD	This application provides a description on the way to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module. The objective is to develop an application using most of the features offered by FatFs to configure a microSD drive.	X	X	-	-	-	X	-	-	X	X
	FreeRTOS	FreeRTOS_LowPower	This application shows how to enter and exit low-power mode with CMSIS RTOS API.	-	-	-	-	-	X	-	-	-	X
		FreeRTOS_Mail	This application shows how to use mail queues with CMSIS RTOS API.	-	X	-	-	-	X	-	-	-	-
		FreeRTOS_Mutexes	This application shows how to use mutexes with CMSIS RTOS API.	-	X	-	-	-	X	-	-	-	X
		FreeRTOS_Queues	This application shows how to use message queues with CMSIS RTOS API.	X	X	-	-	-	X	-	-	-	X
		FreeRTOS_Semaphore	This application shows how to use semaphores with CMSIS RTOS API.	X	X	-	-	-	X	-	-	-	X
		FreeRTOS_SemaphoreFromISR	This application shows how to use semaphore from ISR with CMSIS RTOS API.	X	X	-	-	-	X	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Applications	FreeRTOS	FreeRTOS_ Signal	This application shows how to perform thread signaling using CMSIS RTOS API.	-	X	-	-	-	X	-	-	-	-
		FreeRTOS_ SignalFromISR	This application shows how to perform thread signaling from an interrupt using CMSIS RTOS API.	-	X	-	-	-	X	-	-	-	-
		FreeRTOS_ ThreadCreation	This application shows how to implement thread creation using CMSIS RTOS API.	X	X	X	X	X	X	X	X	-	X
		FreeRTOS_ Timers	This application shows how to use timers of CMSIS RTOS API.	X	X	-	-	-	X	-	-	-	X
	IAP	IAP_Binary_ Template	This directory contains a set of source files that build the application to be loaded into Flash memory using In-Application Programming (IAP) through USART.	-	X	-	-	-	X	-	-	-	-
		IAP_Main	This directory contains a set of source files and pre-configured projects that describe how to build an application to be loaded into Flash memory using In-Application Programming (IAP) through USART.	-	X	-	-	-	X	-	-	-	-
	STemWin	STemWin_ HelloWorld	This application shows how to implement a simple "Hello World" example based on STemWin.	-	X	-	-	-	X	-	-	-	X

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F303RE-Nucleo	STM32303E_EVAL	STM32F3348-Discovery	STM32F334R8-Nucleo	STM32F302R8-Nucleo	STM32373C_EVAL	STM32F303K8-Nucleo	STM32F3-Discovery	STM32F303ZE-Nucleo	STM32303C_EVAL
Applications	Touch Sensing	TouchSensing_2touchkeys	This firmware is a basic example on the way to use the STMTouch driver with 2 touchkey sensors. The ECS and DTO are also used.	-	X	-	-	-	-	-	-	-	X
		TouchSensing_2touchkeys_IT	This firmware is a basic example on the way to use the STMTouch driver with 2 touchkey sensors. The ECS and DTO are also used.	-	X	-	-	-	-	-	-	-	X
		TouchSensing_Linear	This firmware is a basic example on the way to use the STMTouch driver with 1 linear sensor. The ECS and DTO are also used.	-	-	-	-	-	X	-	-	-	-
		TouchSensing_Linear_IT	This firmware is a basic example on the way to use the STMTouch driver with 1 linear sensor. The ECS and DTO are also used.	-	-	-	-	-	X	-	-	-	-

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Applications	USB_ Device	CDC_ Standalone	This application shows how to use the USB device application based on the Device Communication Class (CDC) following the PSTN subprotocol using the USB Device and UART peripherals.	-	-	-	-	-	X	-	-	-	X
		CustomHID_ Standalone	This application shows how to use the USB device application based on the Human Interface (HID).	-	-	-	-	-	X	-	-	-	X
		DFU_ Standalone	This application presents a compliant implementation of the Device Firmware Upgrade (DFU) capability for programming the embedded Flash memory through the USB peripheral.	X	X	-	-	X	X	-	X	-	X
		HID_ Standalone	This application shows how to use the USB device application based on the Human Interface (HID).	X	X	-	-	X	X	-	X	X	X
		MSC_ Standalone	This application shows how to use the USB device application based on the Mass Storage Class (MSC).	-	X	-	-	-	X	-	-	X	X
Total number of applications: 77				**8**	**19**	**1**	**1**	**3**	**22**	**1**	**3**	**3**	**16**

Table 1. STM32CubeF3 firmware examples (continued)

Level	Module name	Project Name	Description	STM32F 303RE-Nucleo	STM32 303E_ EVAL	STM32 F3348-Discovery	STM32F 334R8-Nucleo	STM32F 302R8-Nucleo	STM32 373C_ EVAL	STM32F 303K8-Nucleo	STM32F3-Discovery	STM32F 303ZE-Nucleo	STM32 303C_ EVAL
Demonstrations	-	Adafruit_LCD_1_8_SD_Joystick	This demonstration firmware is based on STM32Cube. It helps to discover STM32 Cortex-M devices that can be plugged on a STM32 Nucleo board.	X	-	-	X	X	-	-	-	X	-
		Demo	This demonstration firmware is based on STM32Cube. It helps to discover STM32 Cortex-M devices that can be plugged on a STM32 Discovery board.	-	-	X	-	-	-	-	X	-	-
		Gravitech_ 4Digits_Counter	This demonstration shows how to use the Gravitech 7 segment 4 digits shield with a Nucleo 32 Board.	-	-	-	-	-	-	X	-	-	-
Total number of demonstrations: 7				**1**	**0**	**1**	**1**	**1**	**0**	**1**	**1**	**1**	**0**
Total number of projects: 525				**37**	**64**	**29**	**84**	**45**	**66**	**34**	**46**	**45**	**75**

Revision history

Table 2. Document revision history

Date	Revision	Changes
07-Jul-2015	1	Initial release.
15-Sep-2015	2	Updated list of reference documents. Added NUCLEO-F303K8 board.
09-Nov-2015	3	Added NUCLEO-F303ZE and updated *Table 1: STM32CubeF3 firmware examples*.
17-May-2016	4	Updated list of reference documents. Added new CRC and I2C examples, as well as touch sensing applications in *Table 1: STM32CubeF3 firmware examples*.
01-Jul-2016	5	Updated list of reference documents. Updated HAL examples. Added low-layer driver (LL) and mixed (MIX) examples.
12-Jan-2017	6	Updated the Template project item and added the Template_LL project item in *Chapter: STM32CubeF3 examples*. Updated *Table 1* for Templates_LL and HAL examples.
04-Jul-2017	7	Updated *Figure 1: STM32CubeF3 firmware components*. Removed TrueSTUDIO® from the list of supported toolchains and updated project descriptions in *Table 1: STM32CubeF3 firmware examples*.