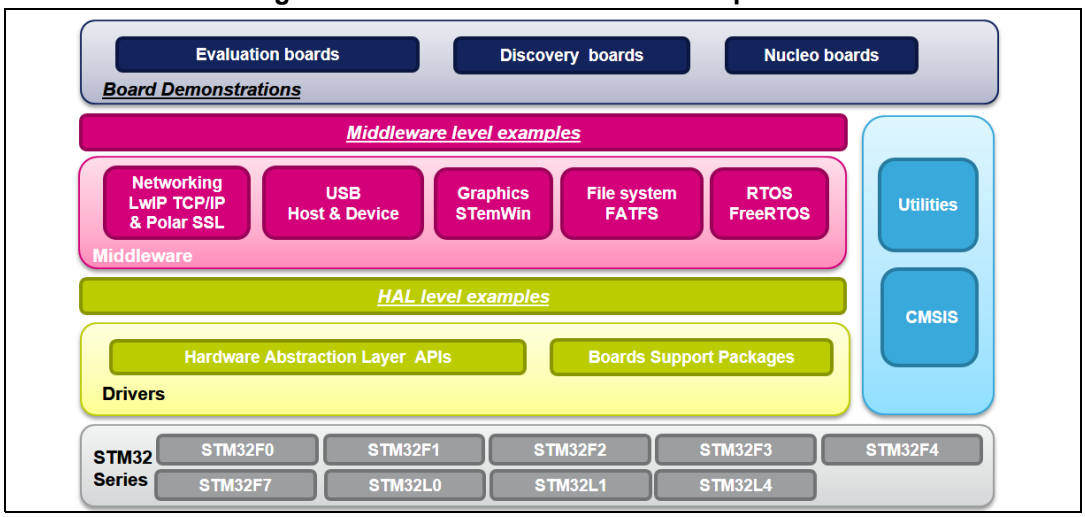
AN4739
Application note

STM32Cube firmware examples for STM32F4 Series

Introduction

The STM32CubeF4 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*).

Figure 1. STM32CubeF4 firmware components



Reference documents

The reference documents are available on www.st.com/stm32cube:

- Latest release of STM32CubeF4 firmware package
- *Getting started with the STM32CubeF4 firmware package for STM32F4 Series (UM1730)*
- *STM32CubeF4 demonstration platform (UM1743)*
- *Description of STM32F4xx HAL drivers (UM1725)*
- *STM32Cube USB Device library (UM1734)*
- *STM32Cube USB Host library (UM1720)*
- *Developing Applications on STM32Cube with FatFS (UM1721)*
- *Developing Applications on STM32Cube with RTOS (UM1722)*
- *Developing Applications on STM32Cube with LwIP TCP/IP stack (UM1713)*
- *STM32cube Ethernet IAP example (UM1709)*

STM32CubeF4 examples

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

- **Examples:** the examples use only the HAL and BSP drivers (middleware not used). Their objective is to demonstrate the product/peripherals features and usage. They are organized per peripheral (one folder per peripheral, e.g. TIM). 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.
- **Applications:** the applications demonstrate the product performance and how to use the available middleware stacks. They are organized either by middleware (a 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 to integrate and run the maximum number of peripherals and middleware stacks to showcase the product features and performance.
- **Template project:** the template project is provided to allow to quickly build a firmware application on a given board.

The examples are located under *STM32Cube_FW_F4_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, *SW4STM32*, and \TrueSTUDIO 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 the 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 STM32CubeF4 firmware package.

Table 1. STM32CubeF4 firmware examples

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E _EVAL	NUCLEO -F446RE	STM324xG _EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I _EVAL	NUCLEO -F401RE
Templates	-	Starter project	This projects provides a reference template that can be used to build any firmware application.	X	X	X	X	X	X	X	X	X
Total number of templates: 9				**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	-	BSP	This example provides a description of how to use the different BSP drivers.	X	X	X	-	X	-	X	X	-
	ADC	ADC_ DualMode Interleaved	This example provides a short description of how to use two ADC peripherals to perform conversions in interleaved dual-mode.	-	-	X	-	X	-	-	X	-
		ADC_Injected Conversion_ Interrupt	This example describes how to use the ADC in interrupt mode to convert data through the HAL API.	-	-	X	-	X	-	-	X	-
		ADC_Regular Conversion_ DMA	This example describes how to use the ADC and DMA to transfer continuously converted data from ADC to memory.	X	X	X	-	X	-	X	X	-
		ADC_Regular Conversion_ Interrupt	This example describes how to use the ADC in interrupt mode to convert data through the HAL API.	-	-	X	-	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	-	-	X	-
		ADC_Trigger Mode	This example describes how to use the ADC and TIM2 to convert continuously data from ADC channel. Each time an external trigger is generated by TIM2 a new conversion is started by ADC.	-	-	X	-	X	-	-	X	-
		ADC_Triple Mode Interleaved	This example provides a short description of how to use the ADC peripheral to convert a regular channel in Triple interleaved mode.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	CAN	CAN_LoopBack	This example provides a description of how to set a communication with the CAN in loopback mode.	-	-	-	-	X	-	-	X	-
		CAN_Loopback	This example provides a description of how to set a communication with the CAN in loopback mode.	-	-	X	-	-	-	-	-	-
		CAN_Networking	This example shows how to configure the CAN peripheral to send and receive CAN frames in normal mode. The sent frames are used to control LEDs by pressing key push button.	-	-	-	-	X	-	-	X	-
	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 doesn't acknowledge the received messages.	-	-	X	-	-	-	-	-	-
		CEC_ MultiAddress	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	-	-	-	-	-	-
	CRC	CRC_Example	This example guides you through the different configuration steps by mean of HAL API to ensure the use of the CRC (Cyclic Redundancy Check) calculation unit to get a CRC code of a given buffer of data word(32-bit), based on a fixed generator polynomial(0x4C11DB7).	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	CRYP	CRYP_ AESModes	This example provides a short description of how to use the CRYPTO peripheral to encrypt and decrypt data using AES in chaining modes (ECB, CBC, CTR) and all key sizes (128, 192, 256) Algorithm.	-	-	-	-	X	-	-	X	-
		CRYP_AES_ CCM	This example provides a short description of how to use the CRYPTO peripheral to encrypt data using AES with Combined Cipher Machine (CCM).	-	-	-	-	-	-	-	X	-
		CRYP_AES_ DMA	This example provides a short description of how to use the CRYPTO peripheral to encrypt and decrypt data using AES-128 Algorithm with ECB chaining mode.	-	-	-	-	X	-	-	X	-
		CRYP_AES_ GCM	This example provides a description of how to use the CRYPTO peripheral to encrypt and decrypt data using AES with Galois/Counter Mode (GCM).	-	-	-	-	-	-	-	X	-
		CRYP_ DESTDESmodes	This example provides a short description of how to use the CRYPTO peripheral to encrypt and decrypt data using DES and TDES in all modes (ECB, CBC) Algorithm.	-	-	-	-	X	-	-	X	-
		CRYP_TDES_ DMA	This example provides a short description of how to use the CRYPTO peripheral to encrypt data using TDES Algorithm.	-	-	-	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	Cortex	CORTEXM_MPU	This example presents the MPU features on STM32F4xx devices and it can be easily ported to any other STM32 device supporting MPU.	-	-	X	-	X	-	-	X	-
		CORTEXM_ ModePrivilege	This example shows how to modify CortexM4 Thread mode privilege access and stack.	-	-	X	-	X	-	-	X	-
		CORTEXM_ SysTick	This example shows how to use the default configuration of SysTick with a time base equal to 1 ms in order to insert a delay between LEDs toggling.	-	-	X	-	X	-	-	X	-
	DAC	DAC_Signals Generation	This example provides a short description of how to use the DAC peripheral to generate several signals using DMA controller.	X	-	X	-	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	-
	DCMI	DCMI_ CaptureMode	This example provides a short description of how to use the DCMI to interface with camera module and display in continuous mode the picture on LCD.	-	-	X	-	X	-	-	X	-
		DCMI_ SnapshotMode	This example provides a short description of how to use the DCMI to interface with camera module and display in snapshot mode the picture on LCD.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	DMA	DMA_FIFOMode	This example provides a description of how to use a DMA channel to transfer a word data buffer from FLASH memory to embedded SRAM memory with FIFO mode enabled through the STM32F4xx HAL API.	-	-	X	-	X	-	-	X	-
		DMA_ FLASHToRAM	This example provides a description of how to use a DMA channel to transfer a word data buffer from FLASH memory to embedded SRAM memory through the STM32F4xx HAL API.	X	X	X	-	X	-	X	X	-
	DMA2D	DMA2D_MemToMem WithBlending	This example provides a description of how to configure DMA2D peripheral in Memory_to_Memory with blending transfer mode.	X	-	-	-	-	-	-	X	-
		DMA2D_MemToMem WithLCD	This example provides a description of how to configure DMA2D peripheral in Memory_to_Memory transfer mode and display the result on LCD.	-	-	-	-	-	-	-	X	-
		DMA2D_MemToMem WithPFC	This example provides a description of how to configure DMA2D peripheral in Memory_to_Memory with pixel format conversion transfer mode.	X	-	-	-	-	-	-	X	-
		DMA2D_MemoryToMemory	This example provides a description of how to configure DMA2D peripheral in Memory_to_Memory transfer mode.	-	-	-	-	-	-	-	X	-
		DMA2D_RegToMemWithLCD	This example provides a description of how to configure DMA2D peripheral in Register_to_Memory transfer mode and display the result on LCD.	-	-	-	-	-	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	FLASH	FLASH_ DualBoot	This example guides you through the different configuration steps by mean of HAL API how to program bank1 and bank2 of the FLASH memory integrated within STM32F4xx devices and swap between both of them.	-	-	-	-	-	-	-	X	-
		FLASH_Erase Program	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	-
		FLASH_Write Protection	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	-
	FMC	FMC_SDRAM	This example describes how to configure the FMC controller to access the SDRAM memory.	X	-	X	-	-	-	-	X	-
		FMC_SDRAM_ DataMemory	This example describes how to configure the FMC controller to access the SDRAM memory including heap and stack.	-	-	X	-	-	-	-	X	-
		FMC_SDRAM_ LowPower	This example guides you through the different configuration steps by mean of HAL API to drive the IS42S16400J SDRAM memory in low-power mode (SDRAM Self-refresh mode).	X	-	X	-	-	-	-	X	-
		FMC_SRAM	This example describes how to configure the FMC controller to access the SRAM memory.	-	-	-	-	-	-	-	X	-
		FMC_SRAM_ DataMemory	This example guides you through the different configuration steps by mean of HAL API to configure the FMC controller to access the SRAM memory mounted on evaluation board (including heap and stack).	-	-	-	-	-	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	FSMC	FSMC_SRAM	This example describes how to configure the FSMC controller to access the SRAM memory.	-	-	-	-	X	-	-	-	-
		FSMC_SRAM_DataMemory	This example describes how to configure the FSMC controller to access the SRAM memory including heap and stack.	-	-	-	-	X	-	-	-	-
	GPIO	GPIO_EXTI	This example shows how to configure external interrupt lines.	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
	HAL	HAL_TimeBase	This example describes how to customize the HAL time base using a general purpose timer(TIM6) instead of Systick as main source of time base.	X	X	X	X	X	X	X	X	X
	HASH	HASH_HMAC_SHA1MD5	This example provides a short description of how to use the HASH peripheral to hash data using HMAC SHA-1 and HMAC MD5 Algorithms.	-	-	-	-	X	-	-	X	-
		HASH_SHA1MD5	This example provides a short description of how to use the HASH peripheral to hash data using SHA-1 and MD5 Algorithms.	-	-	-	-	X	-	-	X	-
		HASH_SHA1MD5_DMA	This example provides a short description of how to use the HASH peripheral to hash data using SHA-1 and MD5 Algorithms.	-	-	-	-	X	-	-	X	-
		HASH_SHA224SHA256_DMA	This example provides a short description of how to use the HASH peripheral to hash data using SHA224 and SHA256 Algorithms.	-	-	-	-	-	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	I2C	FMPI2C_EEPROM	This example describes how to perform I2C data buffer transmission/reception via DMA. The communication uses an I2C EEPROM memory.	-	-	X	-	-	-	-	-	-
		I2C_TwoBoards_AdvComIT	This example describes how to perform I2C data buffer transmission/reception between two boards, using an interrupt.	X	X	-	X	-	-	X	-	-
		I2C_TwoBoards_ComDMA	This example describes how to perform I2C data buffer transmission/reception between two boards, via DMA.	X	X	-	X	-	-	X	-	-
		I2C_TwoBoards_ComIT	This example describes how to perform I2C data buffer transmission/reception between two boards using an interrupt.	X	X	-	X	-	-	X	-	-
		I2C_TwoBoards_ComPolling	This example describes how to perform I2C data buffer transmission/reception between two boards in Polling mode.	X	X	-	X	-	-	X	-	-
	I2S	I2S_Audio	This example provides basic implementation of audio features.	-	-	-	-	X	-	-	-	-
	IWDG	IWDG_Example	This example guides you through the different configuration steps by mean of HAL API to ensure IWDG reload counter and simulate a software fault generating an MCU IWDG reset on expiry of a programmed time period.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	LTDC	LTDC_ColorKeying	This example describe how to enable and use the color keying functionality.	-	-	-	-	-	-	-	X	-
		LTDC_Display_1Layer	This example provides a description of how to configure LTDC peripheral to display BMP image on LCD using only one layer.	-	-	-	-	-	-	-	X	-
		LTDC_Display_2Layers	This example describes how to configure the LTDC peripheral to display two Layers at the same time.	X	-	-	-	-	-	-	X	-
	PWR	PWR_BOR	This example shows how to configure the programmable BOR thresholds using the FLASH option bytes.	-	-	-	-	X	-	-	X	-
		PWR_CurrentConsumption	This example shows how to configure the STM32F4xx system to measure different Low-power modes current consumption.	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. In this example, EXTI line 16 is configured to generate an interrupt on each rising or falling edge of the PVD output signal (which indicates that the Vdd voltage is below the PVD threshold).	-	-	-	-	X	-	-	X	-
		PWR_STANDBY	This example shows how to enters the system to STANDBY mode and wake-up from this mode using: external RESET, RTC Alarm A or WKUP pin.	-	-	X	-	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	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	QSPI	QSPI_ExecuteIn Place	This example describes how to erase part of the QSPI memory, write data in DMA mode and access to QSPI memory in memory-mapped mode to check the data in a forever loop.	-	-	X	-	-	-	-	-	-
		QSPI_Memory Mapped	This example describes how to erase part of the QSPI memory, write data in DMA mode and access to QSPI memory in memory-mapped mode to check the data in a forever loop.	-	-	X	-	-	-	-	-	-
		QSPI_ReadWrite_DMA	This example describes how to erase part of the QSPI memory, write data in DMA mode, read data in DMA mode and compare the result in a forever loop.	-	-	X	-	-	-	-	-	-
		QSPI_ReadWrite_IT	This example describes how to erase part of the QSPI memory, write data in IT mode, read data in IT mode and compare the result in a forever loop.	-	-	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	-
	RNG	RNG_MultiRNG	This example guides you through the different configuration steps by mean of HAL API to ensure RNG random 32bit numbers generation.	-	-	-	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	RTC	RTC_Alarm	This example guides you through the different configuration steps by mean of HAL API to ensure Alarm configuration and generation using the RTC peripheral.	-	-	X	-	X	-	-	X	-
		RTC_Calendar	This example guides you through the different configuration steps by mean of HAL API to ensure Calendar configuration using the RTC peripheral.	-	-	X	X	X	X	-	X	-
		RTC_Tamper	This example guides you through the different configuration steps by mean of HAL API to write/read data to/from RTC Backup data registers and demonstrate the Tamper detection feature using the RTC peripheral.	-	-	X	-	X	-	-	X	-
		RTC_TimeStamp	This example guides you through the different configuration steps by mean of HAL API to ensure Time Stamp configuration using the RTC peripheral.	-	-	X	-	X	-	-	X	-
	SAI	SAI_Audio	This example provides basic implementation of audio features.	-	-	X	-	-	-	-	X	-
		SAI_AudioPlay	This example show how to play an audio file using the DMA circular mode and how to handle the buffer update.	-	-	X	-	-	-	-	-	-
	SMARTCARD	SMARTCARD_T0	This example describes a firmware Smartcard Interface based on the USART peripheral. The main purpose of this firmware example is to provide resources facilitating the development of an application using the USART peripheral in smartcard mode.	-	-	-	-	X	-	-	-	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	SPI	SPI_FullDuplex_AdvComIT	This example guides you through the different configuration steps by mean of HAL API to ensure SPI Data buffer transmission and reception using Interrupt, in an advance communication mode: Master board is always sending command to slave before any transmission and Slave board is sending acknowledge before going further.	-	-	-	-	-	-	X	-	-
		SPI_FullDuplex_AdvComPolling	This example guides you through the different configuration steps by mean of HAL API to ensure SPI Data buffer transmission and reception using Polling, in an advance communication mode: Master board is always sending command to slave before any transmission and Slave board is sending acknowledge before going further.	-	-	-	-	-	-	X	-	-
		SPI_FullDuplex_ComDMA	This example shows how to perform SPI data buffer transmission/reception between two boards via DMA.	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	-	-
		SPI_FullDuplex_ComPolling	This example shows how to ensure SPI data buffer transmission/reception in Polling mode between two boards.	X	X	-	-	-	-	X	-	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	TIM	TIM_6Steps	This example shows how to configure the TIM1 peripheral to generate 6 Steps.	-	-	X	-	X	-	-	X	-
		TIM_ 7PWMOutput	This example shows how to configure the TIM1 peripheral to generate 7 PWM signals with 4 different duty cycles (50%, 37.5%, 25% and 12.5%).	-	-	X	-	X	-	-	X	-
		TIM_ CascadeSynchro	This example shows how to synchronize TIM peripherals in cascade mode.	-	-	X	-	X	-	-	X	-
		TIM_ Complementary Signals	This example shows how to configure the TIM1 peripheral to generate three complementary TIM1 signals, to insert a defined dead time value, to use the break feature and to lock the desired parameters.	-	-	X	-	X	-	-	X	-
		TIM_DMA	This example provides a description of how to use DMA with TIM1 Update request to transfer Data from memory to TIM1 Capture Compare Register 3 (CCR3).	-	-	X	-	X	-	-	X	-
		TIM_DMABurst	This example shows how to update the TIM1 channel1 period and the duty cycle using the TIM1 DMA burst feature.	-	-	X	-	X	-	-	X	-
		TIM_Encoder	This example shows how to configure the TIM1 peripheral in encoder mode to determinate the rotation direction.	-	-	X	-	X	-	-	X	-
		TIM_ExtTrigger Synchro	This example shows how to synchronize TIM peripherals in cascade mode with an external trigger.	-	-	X	-	X	-	-	X	-
		TIM_ InputCapture	This example shows how to use the TIM peripheral to measure the frequency of an external signal.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	TIM	TIM_OCActive	This example shows how to configure the TIM peripheral to generate four different signals with four different delays.	-	-	X	-	X	-	-	X	-
		TIM_OCInactive	This example shows how to configure the TIM peripheral in Output Compare Inactive mode with the corresponding Interrupt requests for each channel.	-	-	X	-	X	-	-	X	-
		TIM_OCToggle	This example shows how to configure the TIM3 peripheral to generate four different signals with four different frequencies.	-	-	X	-	X	-	-	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	-	X	-	-	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	X	X
		TIM_PWMOutput	This example shows how to configure the TIM peripheral in PWM (Pulse Width Modulation) mode.	-	-	X	-	X	-	-	X	-
		TIM_Parallel Synchro	This example shows how to synchronize TIM2 and Timers (TIM3 and TIM4) in parallel mode.	-	-	X	-	X	-	-	X	-
		TIM_Prescaler Selection	This example shows how to configure the TIM peripheral in PWM (Pulse Width Modulation) mode with clock prescaler selection feature activated using __HAL_RCC_TIMCLKPRESCALER() which allow to double the output frequency.	-	-	X	-	-	-	-	-	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	TIM	TIM_Prescaler_ Selection	This example shows how to configure the TIM peripheral in PWM (Pulse Width Modulation) mode with clock prescaler selection feature activated using __HAL_RCC_TIMCLKPRESCALER() which allow to double the output frequency.	-	-	-	-	-	-	-	X	-
		TIM_ Synchronization	This example shows how to synchronize TIM1 and Timers (TIM3 and TIM4) in parallel mode.	-	-	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	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
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	-	-	-	-	-	-
		UART_HyperTerminal_IT	This example describes an UART transmission (transmit/receive) between a board and an HyperTerminal PC application by using an interrupt.	-	-	X	-	-	-	-	-	-
		UART_Hyperterminal_DMA	This example describes an UART transmission (transmit/receive) in DMA mode between a board and an HyperTerminal PC application.	-	-	-	-	X	-	-	X	-
		UART_Hyperterminal_IT	This example describes an UART transmission (transmit/receive) between a board and an HyperTerminal PC application by using an interrupt.	-	-	-	-	X	-	-	X	-
		UART_Printf	This example shows how to use the STM32 Nucleo boards Virtual Com port feature.	-	-	X	X	X	X	-	X	X
		UART_TwoBoards_ComDMA	This example describes an UART transmission (transmit/receive) in DMA mode between two boards.	X	X	-	-	-	-	X	-	-
		UART_TwoBoards_ComIT	This example describes a UART transmission (transmit/receive) in interrupt mode between two boards.	X	X	-	-	-	-	X	-	-
		UART_TwoBoards_ComPolling	This example describes a UART transmission (transmit/receive) in polling mode between two boards.	X	X	-	-	-	-	X	-	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Examples	WWDG	WWDG_ Example	This example guides you through the different configuration steps by mean of HAL API to ensure WWDG counter update at regular period and simulate a software fault generating an MCU WWDG reset on expiry of a programmed time period.	-	-	X	-	X	-	-	X	-
Total number of examples: 315				**26**	**20**	**68**	**11**	**70**	**7**	**23**	**85**	**5**
Demons-trations	-	Demo	The STM32Cube Demonstration platform comes on top of the STM32CubeTM as a firmware package that offers a full set of software components based on a modules architecture allowing re-using them separately in standalone applications. All these modules are managed by the STM32Cube Demonstration kernel allowing to dynamically adding new modules and access to common resources (storage, graphical components and widgets, memory management, Real-Time operating system) The STM32Cube Demonstration platform is built around the powerful graphical library STemWin and the FreeRTOS real time operating system and uses almost the whole STM32 capability to offer a large scope of usage based on the STM32Cube HAL BSP and several middleware components.	X	X	X	-	X	X	X	X	-
Total number of demonstrations: 7				**1**	**1**	**1**	**0**	**1**	**1**	**1**	**1**	**0**

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	Audio	Audio_playback_and_record	This application provides a description of an audio play and record with different MEMS used on 32F401CDISCOVERY.	-	X	X	-	-	-	X	X	-
	Camera	Camera_To_USB Disk	This application provides a short description of how to use the DCMI to interface with camera module and display in continuous mode the picture on LCD and to save a picture in USB device.	-	-	X	-	X	-	-	X	-
	Display	LCD_Paint	This application describes how to configure LCD touch screen and attribute an action related to configured touch zone and how to save BMP picture in SDCard.	-	-	-	-	X	-	-	-	-
		LTDC_Animated PictureFrom SDCard	This application describes how to display on LCD an animated picture saved under microSD.	-	-	-	-	-	-	-	X	-
		LTDC_Animated PictureFromUSB	This application describes how to display on LCD pictures saved under USB mass storage.	X	-	-	-	-	-	-	-	-
		LTDC_Paint	This application describes how to configure LCD touch screen and attribute an action related to configured touch zone.	X	-	-	-	-	-	-	X	-
		LTDC_Pictures FromSDCard	This application describes how to display on LCD pictures saved under SD card.	-	-	-	-	-	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	FatFs	FatFs_ MultiDrivers	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with multidrives (RAMDisk, uSD) configuration.	-	-	X	-	-	-	-	-	-
		FatFs_ MultiDrives	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with multidrives (RAMDisk, uSD) configuration.	-	-	-	-	X	-	-	X	-
		FatFs_RAMDisk	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with RAM disk (SDRAM) drive configuration.	X	-	X	-	X	-	-	X	-
		FatFs_RAMDisk_RTOS	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with RAM disk (SRAM) drive in RTOS mode configuration.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	FatFs	FatFs_USBDisk	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module and STM32 USB On-The-Go (OTG) host library, in High Speed (HS) modes (configured in FS), in order to develop an application exploiting FatFs offred features with USB disk drive configuration.	X	X	X	-	X	-	X	X	-
		FatFs_USBDisk_MultipleAccess_RTOS	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, FreeRTOS as an RTOS module based on using CMSIS-OS wrapping layer common APIs, and also STM32 USB On-The-Go (OTG) host library, in both Full Speed (FS) and High Speed (HS) modes, in order to develop an application exploiting FatFs offered features with USB disk drive in RTOS mode configuration.	-	-	X	-	X	-	-	X	-
		FatFs_USBDisk_RTOS	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, FreeRTOS as an RTOS module based on using CMSIS-OS wrapping layer common APIs, and also STM32 USB On-The-Go (OTG) host library, in both Full Speed (FS) and High Speed (HS) modes, in order to develop an application exploiting FatFs offered features with USB disk drive in RTOS mode configuration.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	FatFs	FatFs_uSD	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with microSD drive configuration.	-	-	X	-	X	-	-	X	-
		FatFs_uSD_RTOS	This application provides a description on how to use STM32Cube firmware with FatFs middleware component as a generic FAT file system module, in order to develop an application exploiting FatFs offered features with microSD drive in RTOS mode configuration.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	FreeRTOS	FreeRTOS_ LowPower	This directory contains a set of source files that implement an application that uses message queues with CMSIS RTOS API This application creates two threads.	-	-	X	-	X	-	-	X	-
		FreeRTOS_ Mutexes	This directory contains a set of source files that implement an application that uses mutexes with CMSIS RTOS API This application creates three threads with different priorities, and access the same mutex MutexHighPriorityThread() has the highest priority so executes first and grabs the mutex and sleeps for a short period to let the lower priority threads execute. When it has completed its demo functionality it gives the mutex back before suspending itself.	-	-	X	-	X	-	-	X	-
		FreeRTOS_ Queues	This directory contains a set of source files that implement an application that uses message queues with CMSIS RTOS API This application creates two threads that send and receive an incrementing number to/from a queue.	-	-	X	-	X	-	-	X	-
		FreeRTOS_ Semaphore	This directory contains a set of source files that implement an application that uses semaphores with CMSIS RTOS API This application creates two threads that toggle LEDs through a shared semaphore.	-	-	X	-	X	-	-	X	-
		FreeRTOS_ SemaphoreFrom ISR	This directory contains a set of source files that implement an application that uses semaphore from ISR with CMSIS RTOS API This application creates a thread that toggle LED through semaphore given from ISR.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	FreeRTOS	FreeRTOS_ ThreadCreation	This directory contains a set of sources files that implement a thread creation application using CMSIS RTOS API This application creates two threads with the same priority, which executes in a periodic cycle of 15 seconds.	X	-	X	-	X	-	-	X	-
		FreeRTOS_ Timers	This directory contains a set of source files that implement an application that uses timers of CMSIS RTOS API This application creates a thread that toggles LED2 every 400 ms, and a periodic timer that calls a callback function every 200 ms to toggle the LED1.	-	-	X	-	X	-	-	X	-
	LibJPEG	LibJPEG_ Decoding	This application demonstrates how to read jpeg file from USB disk, decode it and display the final BMP image on the LCD.	X	-	X	-	X	-	-	X	-
		LibJPEG_ Encoding	This application demonstrates how to read BMP file from USB disk, encode it, save the jpeg file in USB disk then decode the jpeg file and display the final BMP image on the LCD.	X	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	LwIP	LwIP_HTTP_Server_Netconn_RTOS	This application guides STM32Cube HAL API users to run a http server application based on Netconn API of LwIP TCP/IP stack The communication is done with a web browser application in a remote PC.	-	-	-	-	X	-	-	X	-
		LwIP_HTTP_Server_Raw	This application guides STM32Cube HAL API users to run a http server application based on Raw API of LwIP TCP/IP stack The communication is done with a web browser application in a remote PC.	-	-	-	-	X	-	-	X	-
		LwIP_HTTP_Server_Socket_RTOS	This application guides STM32Cube HAL API users to run a http server application based on Socket API of LwIP TCP/IP stack The communication is done with a web browser application in a remote PC.	-	-	-	-	X	-	-	X	-
		LwIP_IAP	This application guides STM32Cube HAL API users to run In-Application Programming (IAP) over Ethernet.	-	-	-	-	X	-	-	X	-
		LwIP_TCP_Echo_Client	This application guides STM32Cube HAL API users to run TCP Echo Client application based on Raw API of LwIP TCP/IP stack To run this application, On the remote PC, open a command prompt window.	-	-	-	-	X	-	-	X	-
		LwIP_TCP_Echo_Server	This application guides STM32Cube HAL API users to run TCP Echo Server application based on Raw API of LwIP TCP/IP stack To run this application, On the remote PC, open a command prompt window.	-	-	-	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	LwIP	LwIP_TFTP_Server	This application guides STM32Cube HAL API users to run a tftp server demonstration for STM32F4xx devices.	-	-	-	-	X	-	-	X	-
		LwIP_UDPTCP_Echo_Server_Netconn_RTOS	This application guides STM32Cube HAL API users to run a UDP/TCP Echo Server application based on Netconn API of LwIP TCP/IP stack To run this application, On the remote PC, open a command prompt window.	-	-	-	-	X	-	-	X	-
		LwIP_UDP_Echo_Client	This application guides STM32Cube HAL API users to run a UDP Echo Client application based on Raw API of LwIP TCP/IP stack To run this application, On the remote PC, open a command prompt window.	-	-	-	-	X	-	-	X	-
		LwIP_UDP_Echo_Server	This application guides STM32Cube HAL API users to run UDP Echo Server application based on Raw API of LwIP TCP/IP stack To run this application, On the remote PC, open a command prompt window.	-	-	-	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	PolarSSL	SSL_Client	This application guides STM32Cube HAL API users to run an SSL client application based on PolarSSL crypto library and LwIP TCP/IP stack To off-load the CPU from encryption/decryption, hash and RNG, all these algorithms are implemented using the hardware acceleration AES 128/192/256, Triple DES, MD5, SHA-1 and analog RNG through the STM32Cube HAL APIs In this application the client (evaluation board) sends a crypted message to the server (test PC), which will decrypt the message then reply to the client.	-	-	-	-	X	-	-	X	-
		SSL_Server	This application guides STM32Cube HAL API users to run an SSL Server application based on PolarSSL crypto library and LwIP TCP/IP stack To off-load the CPU from encryption/decryption, hash and RNG, all these algorithms are implemented using the hardware acceleration AES 128/192/256, Triple DES, MD5, SHA-1, SHA2-2 and analog RNG through the STM32Cube HAL APIs the HTTP server (STM3241G-EVAL) contains a html page dynamically refreshed (every 1 s), it shows the RTOS statistics in runtime The HyperTerminal can be used to debug messages exchanged between client and server.	-	-	-	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	STemWin	STemWin_Hello World	This directory contains a set of source files that implement a simple "Hello Wolrd" application based on STemWin for STM32F4xx devices.	X	-	X	-	X	-	-	X	-
		STemWin_ SampleDemo	This directory contains a set of source files that implement demo based on STemWin for STM32F4xx devices.	X	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	USB_Device	AUDIO_ Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the AUDIO Class implementation of an audio streaming (Out: Speaker/Headset) capability on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		CDC_Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the Device Communication Class (CDC) following the PSTN subprotocol in the STM32F4xx devices using the OTG-USB and UART peripherals.	-	-	X	-	X	-	-	X	-
		CustomHID_ Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the Custom HID Class on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		DFU_Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the Device Firmware Upgrade (DFU) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		DualCore_ Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the STM32F4xx multi core support feature integrating Mass Storage (MSC) and Human Interface (HID) in the same project.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	USB_Device	HID_LPM_ Standalone	The STM32F446xx devices support the USB Link Power Management Protocol (LPM-L1) and complies with the USB 2.0 LPM-L1 ECN. The hpcd.Init.lpm_enable in the usbd_conf.c should be set to 1 to enable the support for LPM-L1 protocol in the USB stack.	-	-	X	-	-	-	-	-	-
		HID_Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the Human Interface (HID) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		MSC_Standalone	This application is a part of the USB Device Library package using STM32Cube firmware. It describes how to use USB device application based on the Mass Storage Class (MSC) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	USB_Host	AUDIO_ Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Audio OUT class on the STM32F4xx devices.	-	-	-	-	X	-	-	X	-
		CDC_Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Communication Class (CDC) on the STM32F4xx devices.	-	-	-	-	X	-	-	X	-
		DualCore_ Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the STM32F4xx multi core support feature integrating Mass Storage (MSC) and Human Interface (HID) in the same project.	-	-	-	-	X	-	-	X	-
		DynamicSwitch_ Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use dynamically switch, on the same port, between available USB host applications on the STM32F4xx devices.	-	-	-	-	X	-	-	X	-
		FWupgrade_ Standalone	The firmware upgrade application or In-Application programming (IAP) is a feature that allows a user application to erase and write to on-chip flash memory.	X	-	X	-	X	-	-	X	-
		HID_RTOS	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Human Interface Class (HID) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-

Table 1. STM32CubeF4 firmware examples (continued)

Level	Module Name	Project Name	Description	32F429I DISCOVERY	32F401C DISCOVERY	STM32446E_EVAL	NUCLEO-F446RE	STM324xG_EVAL	NUCLEO-F411RE	STM32F4 DISCOVERY	STM324x9I_EVAL	NUCLEO-F401RE
Applica-tions	USB_Host	HID_Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Human Interface Class (HID) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		MSC_RTOS	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Mass Storage Class (MSC) on the STM32F4xx devices in RTOS mode configuration.	-	-	X	-	X	-	-	X	-
		MSC_Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Mass Storage Class (MSC) on the STM32F4xx devices.	-	-	X	-	X	-	-	X	-
		MTP_Standalone	This application is a part of the USB Host Library package using STM32Cube firmware. It describes how to use USB host application based on the Media Transfer Protocol (MTP) on the STM32F4xx devices.	-	-	-	-	X	-	-	X	-
Total number of applications: 151				**10**	**2**	**34**	**0**	**50**	**0**	**2**	**53**	**0**
Total number of projects: 482				**38**	**24**	**104**	**12**	**122**	**9**	**27**	**140**	**6**

1 Revision history

Table 2. Document revision history

Date	Revision	Changes
27-Jul-2015	1	Initial release.