

STM32Cube WL firmware package

Firmware package

Revision 1.0

Hello, and welcome to this presentation of the STM32Cube firmware drivers including the hardware abstraction layer drivers.

STM32CubeMX



Initialization code generation made for all STM32 families depending on user configuration



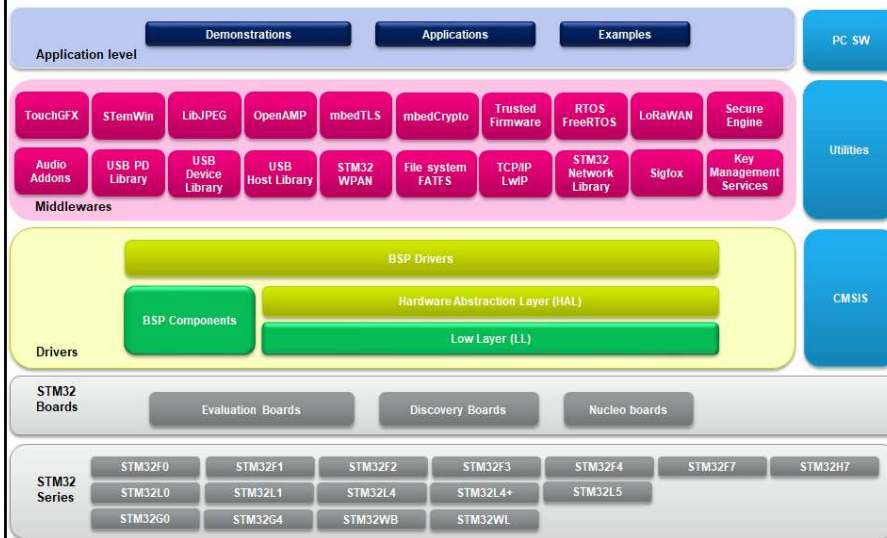
While this presentation is specifically about the STM32WL, the STM32Cube comprehensive software tool offers:

- STM32CubeMX graphical software configuration tool that allows the automatic generation of C initialization code using graphical wizards
- A comprehensive embedded software platform, delivered per series (such as STM32CubeWL for STM32WL Series):
 - The STM32Cube HAL, STM32 abstraction layer embedded software ensuring maximized portability across STM32 portfolio.
 - Low-layer APIs (LL) offering a fast light-weight expert-oriented layer which is closer to the hardware than the HAL. LL APIs are available only for a set of peripherals.
 - A consistent set of middleware components such as FatFS and FreeRTOS™, LoRaWAN®, Sigfox, SubGHz_Phy, KMS, Secure Engine &

mbedCrypto

- All embedded software utilities coming with a full set of examples.

Overview



(1) The set of middleware components depends on the STM32 Series.



Application benefits

- Single package
- Compatible with all STM32 series
- Source code with open-source BSD license

The embedded software package is a layered approach:

- Low Level: Hardware Abstraction Layer (HAL & LL), with libraries and examples
- Middleware level: Set of libraries with applications including LoRaWAN, Sigfox, KMS, SBSFU, RTOS, FileSystem ...
- Application level: Demonstrations for use on ST boards

Embedded software is delivered by series (STM32WL, STM32L5, STM32L4, STM32WB, STM32G0, STM32H7, ...) and common modules are covered with fully portable APIs.

Embedded software initialization code can be generated through STM32CubeMX allowing the customer to remain focused on the core application code.

Key features

Layer	Category	Provided embedded software	Provided examples
HAL/LL	Analog	Analog/Digital conversion, Comparator	~214 examples on STM32WL boards !
	Timers	Timers, RTC, Watchdogs, ...	
	Cryptography	CRC, AES, PKA and Random Number generator, ...	
	Security	Global TrustZone Controller	
	System	Flash, RAM, IO, DMA, Clock, Power (low power modes), Inter processor Communication, Hardware Semaphore	
	Connectivity	I2C, USART, SPI, I2S	
	Interface	Serial Audio	
Middleware	RTOS	FreeRTOS open source RTOS, with CMSIS-RTOS wrapper	~38 applications on STM32WL boards !
	File System	FatFS open-source file system	
	LoRaWAN	Lora Wide Area Network	
	Sigfox		
	KMS	Key Management Service	
	Secure Engine	Secure Boot Secure Firmware Update	
Application	Demonstration	Full demonstration for ST boards:	~2 demonstration projects for ST boards!



life.augmented

4

The STM32Cube package is a complete embedded software offer that ensures maximum portability between all STM32 series through 3 software layers: HAL, Middleware and Applications.

The HAL/LL Layer is providing an API for the STM32 embedded peripherals from analog to connectivity, and cryptography to security categories.

A rich set of examples is available to help developers start using the HAL and LL on the product.

The middleware layer for STM32CubeWL contains :

- CMSIS-RTOS implementation with FreeRTOS™ open source solution
- Mbed Crypto libraries
- FAT file system based on open source FatFS solution
- SubGHz_Phy, a common Phy layer according to the OSI model for all above MAC layers
- STMTouch touch sensing library solution.

- LoRaWAN®, Lora Wide Area Network
- Sigfox library
- KMS for Key Management Service
- Secure Engine is a secure enclave for sensible operations execution.

Advanced demos putting together all the embedded software components are also provided in the STM32CubeWL package.

There is a complete set of documents including release notes, readme files or associated user manual.

The packages come with free and user-friendly license terms.

Package organization

Contains STM32WLxx CMSIS files that defines Peripheral's registers declarations, bits definition and the address mapping

BSP drivers for the supported boards

STM32WLxx HAL & LL drivers

Secure Middleware to build SBSFU application

FatFS file system

RTOS FreeRTOS

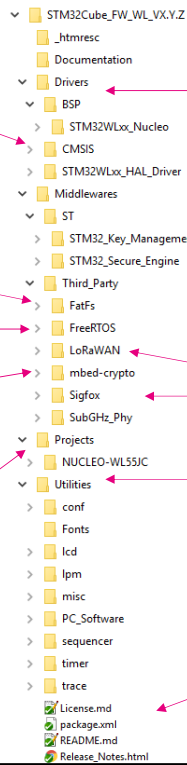
Mbed Crypto

Sub-GHz Middlewares (Phy Layer and MAC layers)

Set of useful utilities

Set of examples/ applications organized by board provided with preconfigured projects.

Package Release Note providing all the relevant information on its content: main changes, supported devices and boards, supported toolchains, firmware components and versions...



Like all STM32Cube firmware packages, the STM32CubeWL firmware solution comes in a single ZIP file having the structure shown in this slide.

It's organized in several main folders:

- The Documentation folder contains the STM32Cube WL getting starting document, helping developers to quickly become familiar with the firmware package and its contents.
- The Drivers folder contains all the ST-developed drivers.
 - CMSIS contains the files defining STM32WLxx supported devices, peripheral registers declarations, their associated bit definitions and address mapping.
 - STM32WLxx_HAL_Driver folder contains the drivers for all the peripherals.
 - The drivers for all supported boards are found in the BSP folder.

- Middleware contains the supported middleware libraries and stacks provided either by ST or third parties.
- The Projects folder contains templates, examples, applications and demonstrations for supported boards and with preconfigured projects and specific readme files that provides all necessary information for a quick and easy execution.
- The Utilities folder contains miscellaneous utility drivers that are used by the provided projects.

The Release Note lists the contents of all the packages, tracks the main changes and provides information on the supported devices and boards and any known limitations.

Supported devices & boards

Macro defined in stm32WLxx.h	STM32WL Series devices
STM32WL55xx	STM32WL55CC, STM32WL55JC
STM32WL54xx	STM32WL54CC, STM32WL54JC
STM32WLE5xx	STM32WLE5C8, STM32WLE5CB, STM32WLE5CC STM32WLE5J8, STM32WLE5JB, STM32WLE5JC
STM32WLE4xx	STM32WLE4C8, STM32WLE4CB, STM32WLE4CC STM32WLE4J8, STM32WLE4JB, STM32WLE4JC

Board	Example	Application	Demonstration
STM32WL55 Nucleo	214	39	2

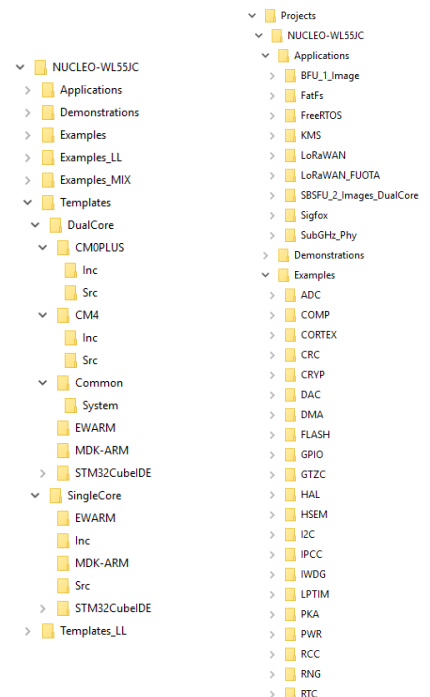


Through its generic architecture, STM32Cube offers a highly portable hardware abstraction layer (HAL). It allows developers to implement application functions by building on layers, such as the middleware layer, without requiring any in-depth knowledge of the MCU. This improves the re-usability of the library code and guarantees an easy portability to other devices.

In addition, thanks to its layered architecture, STM32CubeWL offers full support of all STM32WL microcontrollers and the development boards designed by ST. The user has only to define the correct macro in the stm32WLxx.h file and get in touch with BSP drivers and example/application projects specific to each board provided within the firmware package.

Examples overview

- For each board, a set of examples is provided with preconfigured projects for EWARM, MDK-ARM and STM32CubeIDE toolchains.
- This figure shows the projects structure for the STM32WL55 NUCLEO board, (order code: NUCLEO-WL55JC) which is identical for other boards.
- The examples are classified depending on the STM32Cube level they apply to, and are named as follows:
 - Examples in Level 0 are called Examples, and use HAL drivers without any middleware component
 - Examples in Level 1 are called Applications, and provide typical use cases of each middleware component
 - Examples in Level 2 are called Demonstration, and implement all the HAL, BSP and middleware components
- The Template and Templates_LL project is provided to build quickly any firmware application for all supported boards
- The STM32CubeProjectList file allows quick access and search for a given example within the firmware package
- All single core examples have the same structure,
 - \Inc folder contains all header files
 - \Src folder for the source code
 - \EWARM, \MDK-ARM and \STM32CubeIDE contain the preconfigured project for each toolchain.
 - readme.txt describes example behavior and the environment needed to make it work.
- All dual core examples have the same structure,
 - CM0PLUS\Inc & CM4\Inc folder contains all header files respectively for CM0PLUS & CM4
 - CM0PLUS\Src & CM4\Src folder contains the sources code files respectively for CM0PLUS & CM4
 - Common\ folder contains common files for CM0PLUS & CM4
 - \EWARM, \MDK-ARM and \STM32CubeIDE contain the preconfigured project for each toolchain.
 - readme.txt describes example behavior and the environment needed to make it work.



For each board, a set of examples is provided with preconfigured projects for EWARM, MDK-ARM and STM32CubeIDE toolchains.

The right-side figure shows the projects structure for the STM32WL55 Nucleo board, which is identical for all the other boards.

The examples are classified depending on the STM32Cube level they apply to, and are named as follows:

Examples in Level 0 are called Examples and use LL or HAL drivers without any middleware component.

Examples in Level 1 are called Applications and provide typical use cases of each middleware component.

Examples in Level 2 are called Demonstration and implement all the HAL, BSP and middleware components.

The Template and Template_LL projects are provided to quickly build any firmware application for all supported boards.

The STM32CubeProjectList file allows a quick access and search for a given example within the firmware package.

All single core examples have the same structure:

\Inc folder contains all header files.

\Src folder contains the sources code.

\EWARM, \MDK-ARM and \STM32CubeIDE folders contain the preconfigured project for each toolchain.

A readme text file describes the example behavior and environment needed to make it work.

All dual core examples have the same structure:

CM0PLUS\Inc & CM4\Inc folder contains all header files respectively for CM0PLUS & CM4

CM0PLUS\Src & CM4\Src folder contains the sources code files respectively for CM0PLUS & CM4

Common\ folder contains common files for CM0PLUS & CM4
\EWARM, \MDK-ARM and \STM32CubeIDE folders contain the preconfigured project for each toolchain.

A readme text file describes the example behavior and environment needed to make it work.

Exhaustive documentation list and STM32CubeWL Firmware package can be accessed from ST's web site at www.st.com/stm32cubefw

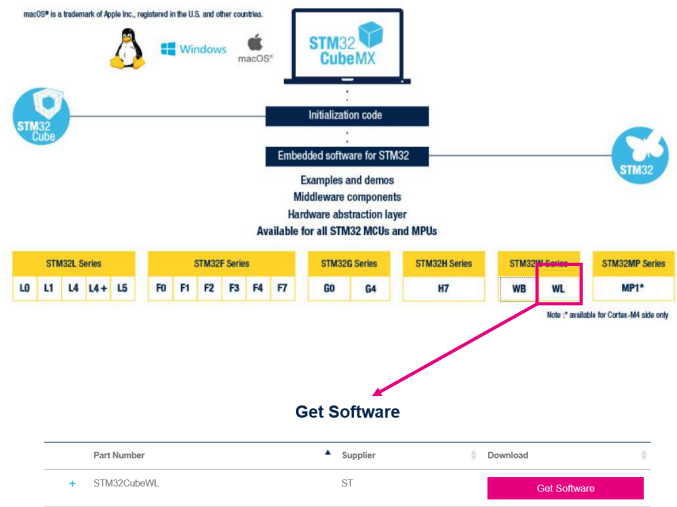
STM32Cube MCU & MPU Packages

STM32Cube is a set of tools and embedded software bricks available free of charge to enable fast and easy development on the STM32 platform which simplifies and speeds up developers' work.

A large number of code use examples are also included making it even easier to get started.

STM32Cube consists of the following components that can be used together or independently:

- The STM32CubeMX graphical user interface and initialization code generator that:
 - Provides graphical wizards to generate initialization C code and includes a utility tool for assisting developers with pin multiplexing, clock tree setting, peripheral configurations and setting up the middleware
 - Generates IDE-ready projects for a wide selection of integrated development environment toolchains
 - Calculates the power consumption for user-defined application sequences
 - Directly imports STM32Cube embedded software libraries from st.com
 - Keeps STM32CubeMX software up-to-date thanks to an integrated updater
- STM32Cube MCU and MPU Packages for each individual STM32 MCU and MPUs series that include:
 - The hardware abstraction layer (HAL) enabling portability between different STM32 devices via standardized API calls
 - Low-layer (LL) APIs, a light-weight, optimized, expert oriented set of APIs designed for both performance and runtime efficiency
 - A collection of middleware components including RTOS, USB library, file system, TCP/IP stack, touch-sensing library or graphics library (depending on the STM32 series)
 - For STM32 MPUs only, the BSP drivers are based on HAL drivers and provide an API Set to the evaluation board and 3rd party components.



The STM32CubeWL firmware can be downloaded from ST website at www.st.com/stm32cubefw
Thank you.