Hello, and welcome to this presentation of the STM32Cube firmware drivers including the hardware abstraction layer drivers.
While this presentation is specifically about the STM32G4, the STM32Cube comprehensive software tool offers:

- STM32CubeMX graphical software configuration tool used to generate initialization code based on user choices
- A complete embedded software package for each STM32 series (such as our STM32CubeG4) with:
  - Hardware Abstraction Layer (HAL) and Low-Layer (LL) APIs
  - A consistent set of middleware components: RTOS, USB, TCP/IP, graphics…
The STM32CubeG4 MCU Package runs on STM32G4 Series microcontrollers, based on the Arm® Cortex® -M4 processor.

STM32CubeG4 gathers, in a single package, all the generic embedded software components required to develop an application on STM32G4 Series microcontrollers.

The package includes low-layer (LL) and hardware abstraction layer (HAL) APIs that cover the microcontroller hardware, together with an extensive set of examples running on STMicroelectronics boards. The STM32CubeG4 MCU Package also contains a set of middleware components with the corresponding examples.
• Full USB device stack supporting many classes
• USB Power Delivery stack
• CMSIS-RTOS implementation with FreeRTOS™ open source solution
• FAT file system based on open source FatFs solution. Several applications and demonstrations implementing all these middleware components are also provided in the STM32CubeG4 MCU Package.

Embedded software is delivered by series (STM32G4, STM32L4, …) 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.
The STM32Cube™ firmware solution is built around three independent levels that interact easily:

- Level 0 contains the drivers and the HAL
- Level 1 contains middleware components and examples based on these components
- Level 2 is composed of a single layer which consists in a global real-time and graphical demonstration based on the middleware service layer, the low-level abstraction layer and the basic peripheral usage applications for board based features.
This slide describes the various software components per level.

Level 0 has 3 sub-layers:

- **The Board Software Package**, which offers a set of APIs corresponding to the hardware components of a board, that is LCD, Audio, microSD and MEMS drivers.
- **The HAL drivers** offer high-level function-oriented highly-portable APIs, they hide the MCU and peripheral complexity to end user
- **The Low-Layer drivers** provide low-level APIs at register level, with better optimization but less portability.
- **Basic peripheral usage examples**. These examples use either the HAL or the low-layer drivers APIs or both. They use the BSP resources as well.

Low-layer drivers require a deep knowledge of MCU and peripheral specifications. They are independent from
HAL and can to be used in standalone mode.
Level 1 is divided into two sub-layers:
• Middleware components, which is a set of libraries covering USB Device library, USB Power Delivery library, FreeRTOS™ and FatFs
• Examples based on the middleware components. Integration examples that use several middleware components are provided as well.
Level 2 is composed of a single layer which consists in a global real-time and graphical demonstration based on the middleware service layer, the low-level abstraction layer and the basic peripheral usage applications for board based features.
Like all STM32Cube firmware packages, the STM32CubeG4 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 G4 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, HAL and LL
  - CMSIS contains the files defining STM32G4xx supported devices, peripheral registers declarations, their associated bit definitions and address mapping
  - STM32G4xx_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 provide all necessary information for a quick and easy execution.
- The Utilities folder contains miscellaneous utility drivers that are used by the provided projects.
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, STM32CubeG4 offers full support of all STM32G4 microcontrollers and the development boards designed by ST.

The user has only to define the correct macro in the stm32g4xx.h file and get in touch with BSP drivers and example/application projects specific to each board provided within the firmware package.
For each board, a set of examples is provided with preconfigured projects for EWARM, MDK-ARM and SW4STM32 toolchains.

The right side figure shows the projects structure for NUCLEO-G431KB board, 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, Examples_LL and Examples_MIX, they use respectively HAL drivers, LL drivers and a mix of HAL and LL 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 Demonstrations, and implement all the HAL, BSP and middleware components.
• Examples in Level 2 are called Demonstration, and implement all the HAL, BSP and middleware components.
The Template project is provided to quickly build any firmware application for all supported boards.

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

All examples have the same structure:
- \Inc folder contains all header files
- \Src folder for the source code
- \EWARM, \MDK-ARM and \SW4STM32 contain the preconfigured project for each toolchain
- readme.txt describes example behavior and the environment needed to make it work
- *.ioc file that allows user to open most of FW examples within CubeMx (starting from CubeMx 5.0)

The Template project is 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 examples have the same structure:
- \Inc folder contains all header files.
- \Src folder contains the sources code.
- \EWARM, \MDK-ARM and \SW4STM32 folders contain the preconfigured project for each toolchain.
A readme text file describes the example behavior and environment needed to make it work.
Note that most of the examples have an .ioc file enabling the configuration of the firmware from CubeMX tool.
Rich documentation is associated to the STM32Cube G4 firmware package. Some documents are generic to all STM32 series and others are specific to the STM32G4 series. The Getting started with the STM32CubeG4 Firmware Package user manual is the document that you need to read first when you start using the STM32Cube Firmware Package.
The STM32Cube G4 firmware can be downloaded from ST website at www.st.com/stm32cubefw.
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:
- STM32Cube MCU and MPU Packages for each individual STM32 MCU and MPUs series.

This slide summarizes the features of STM32Cube. You can refer to the related presentation.