Getting started with the X-CUBE-DISPLAY display module Expansion Package for STM32Cube

Introduction

This user manual gives an overview of the X-CUBE-DISPLAY Expansion Package and explains how to get started in STM32CubeMX.

The X-CUBE-DISPLAY Expansion Package enables display and user interface features on top of STM32 boards. Various user interfaces are supported by the X-CUBE-DISPLAY Expansion Package such as LCD controllers, touch screen, user buttons, joystick, external memories, and others.

The X-CUBE-DISPLAY Expansion Package supports the X-NUCLEO-GFX01M1 expansion board, offering the needed drivers as well as simple examples.

X-CUBE-DISPLAY main features are:

- Software building block suppporting the addition of display modules
- Simple "hello world" example available on the QVGA X-NUCLEO-GFX01M1 STM32 Nucleo expansion board connected to the NUCLEO-G071RB STM32 Nucleo development board
- Easy portability across STM32 microcontrollers with STM32CubeMX in the STM32Cube ecosystem
- Free and user-friendly license terms

Figure 1. **X-CUBE-DISPLAY running on NUCLEO-G071RB equipped with X-NUCLEO-GFX01M1**

Picture is not contractual.

UM2787 - Rev 1 - December 2020
For further information contact your local STMicroelectronics sales office.

www.st.com

1 General information

This user manual briefly describes the X-CUBE-DISPLAY Expansion Package. It focuses on X-CUBE-DISPLAY usage and neither explains the LCD controllers, nor the inputs devices.

Table 1 presents the definitions of the acronyms that are relevant for a better understanding of this document.

Table 1. List of acronyms

Acronym	Definition
API	Application programming interface
BSP	Board support package
HAL	Hardware abstraction layer
IDE	Integrated development environment
LCD	Liquid-crystal display
LED	Light-emitting diode
SPI	Serial peripheral interface

The X-CUBE-DISPLAY Expansion Package runs on STM32 32-bit microcontrollers based on the Arm® Cortex® processor.

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

arm

2 Package description

This section details the X-CUBE-DISPLAY Expansion Package content and how to use it.

2.1 General description

The X-CUBE-DISPLAY Expansion Package provides simple and generic support software for the X-NUCLEO-GFX01M1 expansion board. It runs on the STM32 microcontroller.

The following integrated development environments are supported:

- IAR Systems® - IAR Embedded Workbench® for Arm® (EWARM)
- Keil® - Microcontroller Development Kit (MDK-ARM)
- STMicroelectronics - System Workbench for STM32 (SW4STM32)

Note: *Refer to the release note available in the delivery package root folder for information about the IDE versions supported.*

2.2 Architecture

The software runs on STM32 boards, allows the display and control of graphics and texts via add-on display, memories and controlling modules.

The package is split into the following components:

- LCD controllers
- External memories
- Buttons and joysticks
- LED

2.2.1 Architecture concept

This section provides a high-level view of the software architecture to support display modules.

Figure 2. **Architecture concept**
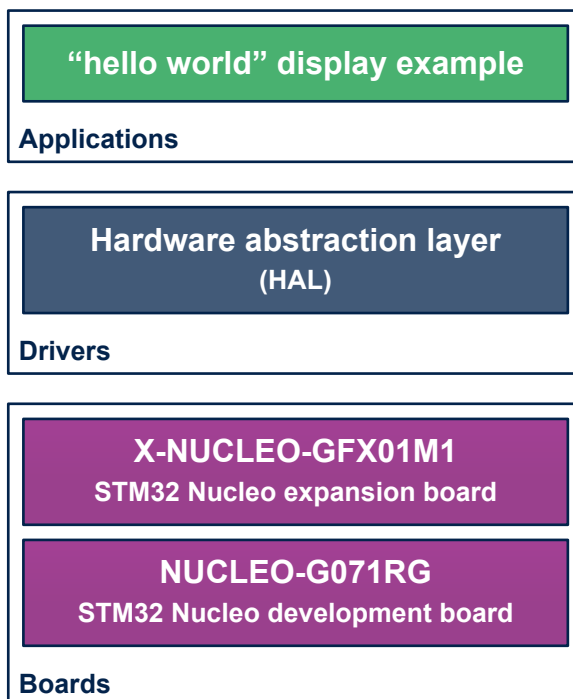
The display modules expose three main interfaces to applications:

- Display interface: via LCD controllers or dedicated display bus
- External memory interface: external memory devices to store graphical assets
- User interface: touch screen, joystick and buttons

The `LCD_IO` layer abstracts the actual hardware bus interface used with the LCD controller.

The `MEM_IO` layer abstracts the actual hardware bus interface used with external memory devices.

The `KEY_CONF` layer abstracts the actual hardware button interface used with the expansion board.

2.3 X-CUBE-DISPLAY Expansion Package content

The X-CUBE-DISPLAY Expansion Package is an expansion for STM32CubeMX with the following main characteristics:

- It complies with STM32CubeMX architecture recommendation (code generator and folders)
- It expands STM32CubeMX to enable the development of UIX applications
- It is based on STM32CubeHAL, which is the hardware abstraction layer for STM32 microcontrollers

The software components used by the application software to exchange with remote applications are the following ones:

- **STM32CubeHAL**: the HAL driver layer provides a generic multi-instance simple set of APIs (application programming interfaces) to interact with the upper layers (application, libraries and stacks).

  It is composed of generic and extension APIs. It is directly built around a generic architecture and allows the layers that are built upon, such as the middleware layer, to implement their functionalities without dependencies on the specific hardware configuration for a given microcontroller.

  This structure improves the library code reusability and guarantees easy portability onto other devices.

- **Board support package (BSP)**: The software package must support the peripherals on the STM32 boards apart from the MCU. This software is included in the board support package. This is a limited set of APIs, which provides a programming interface for certain board-specific peripherals such as the LED and the user button.

- **Configuration files**:
  - `lcd_conf.h` defines the mapping of the GPIO and hardware interface specific to logical name to ease software porting to another board. It provides also the hardware bus interface configuration (for example SPI) to be used for communication with the LCD controller.
  - `mem_conf.h` defines the mapping of the GPIO and hardware interface specific to logical name to ease software porting to another board. It provides also the hardware bus interface configuration (for example SPI) to be used for communication with the external memory device.
  - `key_conf.h` defines the mapping of the GPIO specific to logical name to ease software porting to another board. It provides the GPIO configurations to be used to provide controls for the user.

- **Example projects**: This release provides the "hello world" example only for the X-NUCLEO-GFX01M1 expansion board combined with the NUCLEO-G071RB STM32 development board.

3 Sample application description

This section provides a short overview of the sample applications and examples included in X-CUBE-DISPLAY.

The sample applications and examples:

- are ready-to-use projects that can be generated with STM32CubeMX for any STM32 Nucleo board used together with the X-NUCLEO-GFX01M1 expansion board
- show the users how to use the APIs to correctly initialize and use the X-NUCLEO-GFX01M1 peripherals

3.1 GFX01M1_HelloWorld application

This application provides a "hello world" example that configures the LCD controller and displays an image stored in the the X-NUCLEO-GFX01M1 expansion board external memory. The example provides controls to the user through the joystick buttons.

4 Installing X-CUBE-DISPLAY in STM32CubeMX

After downloading (from *www.st.com*), installing and launching STM32CubeMX (V6.0.0 or above), the X-CUBE-DISPLAY package can be installed following the few steps described below.

Step 1. From the menu, select [**Help**]>[**Manage embedded software packages**].

Figure 3. Managing embedded software packages in STM32CubeMX
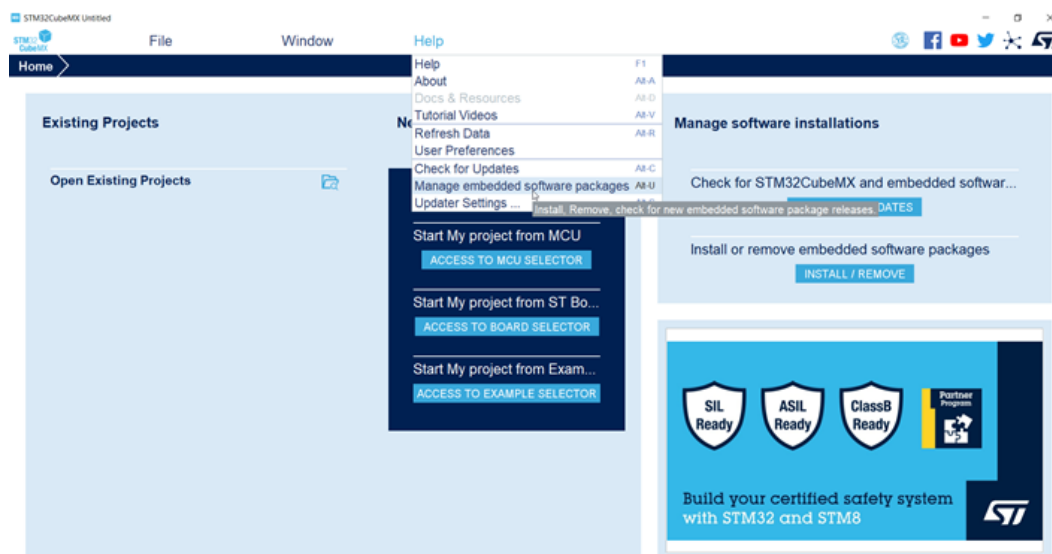
Step 2. From the *Embedded Software Packages Manager* window, press the [**Refresh**] button to get the updated add-on package list. Go to the *STMicroelectronics* tab to find the X-CUBE-DISPLAY package.

Figure 4. Installing the X-CUBE-DISPLAY package in STM32CubeMX
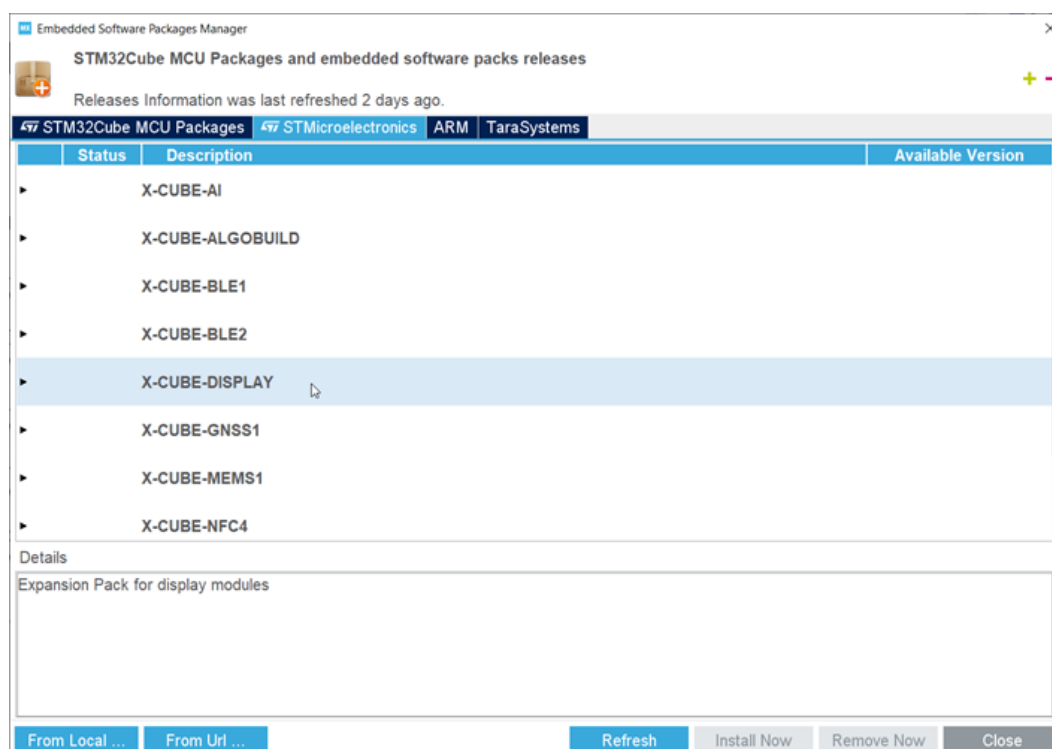
Step 3. Checking the corresponding checkbox to select X-CUBE-DISPLAY and install it by pressing the [**Install Now**] button. Once the installation is completed, the corresponding box becomes green, the [**Close**] button can be pressed, and the configuration of a new project can start.

Figure 5. The X-CUBE-DISPLAY package in STM32CubeMX
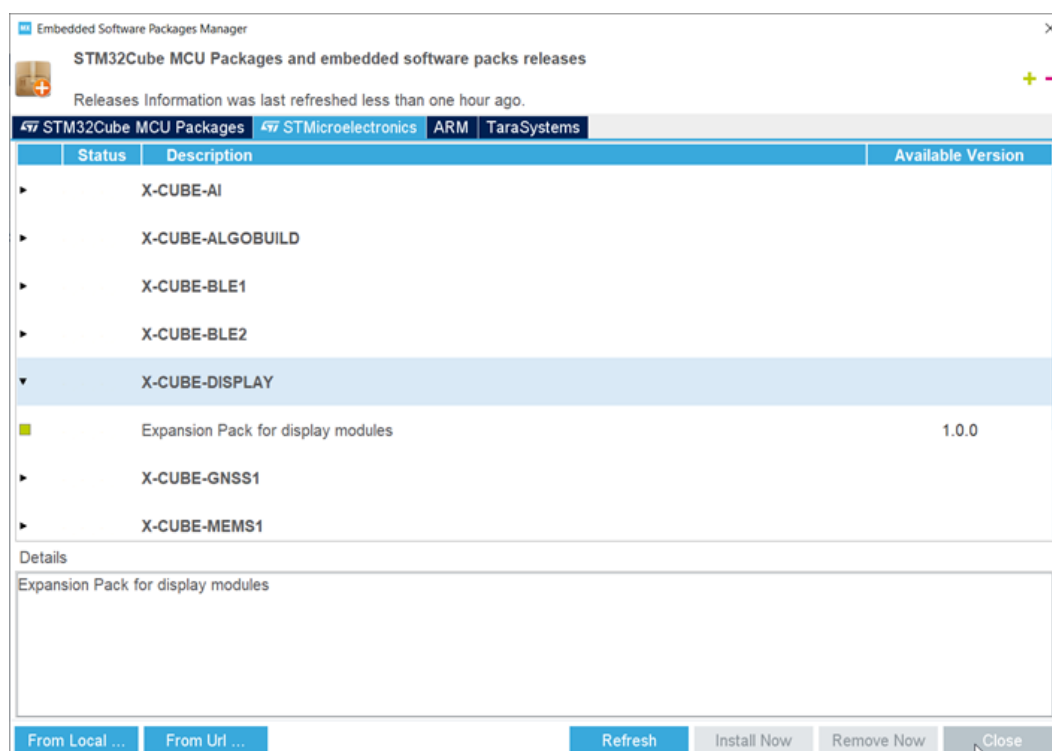
5 Starting a new project

After launching STM32CubeMX, start a new project as described in the steps below.

Step 1. Choose to start a new project from the *MCU Selector* or *Board Selector*.

Figure 6. STM32CubeMX main page
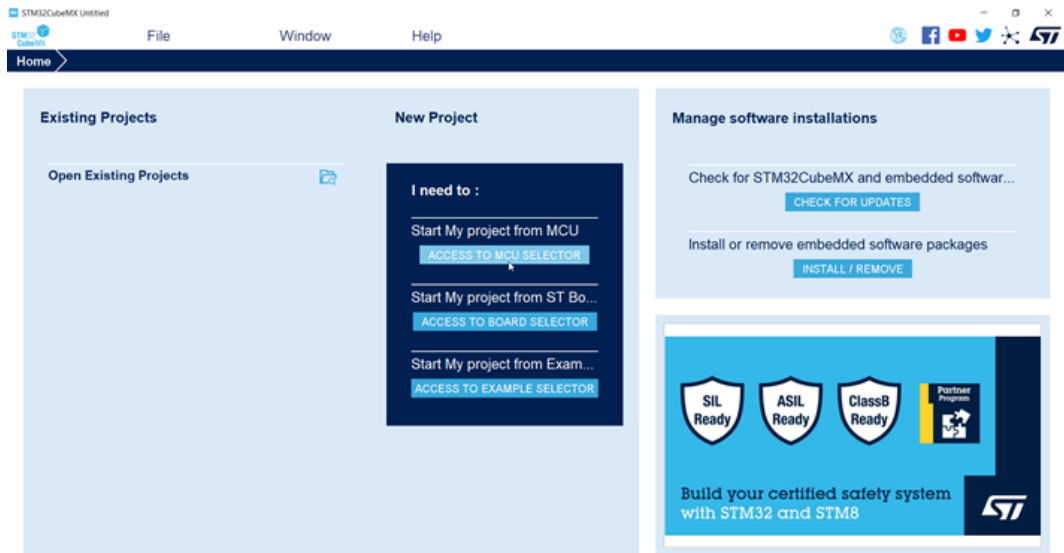
Step 2. Select the desired STM32 microcontroller or platform from the pop-up *MCU Selector* or *Board Selector* window.

Figure 7. STM32CubeMX *Board Selector* window example
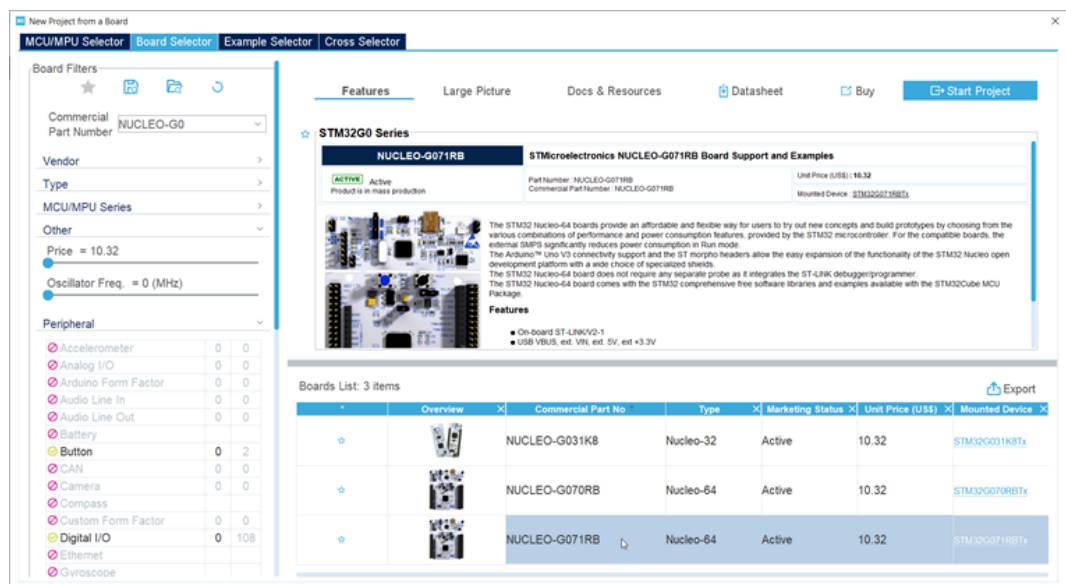
Step 3. Select the microcontroller or the board to display the pinout of the corresponding STM32 microcontroller.

From the window, the user can set up the project by adding one or more Additional Software and peripherals, and configuring the clock.

Figure 8. STM32CubeMX *Pinout & Configuration* window
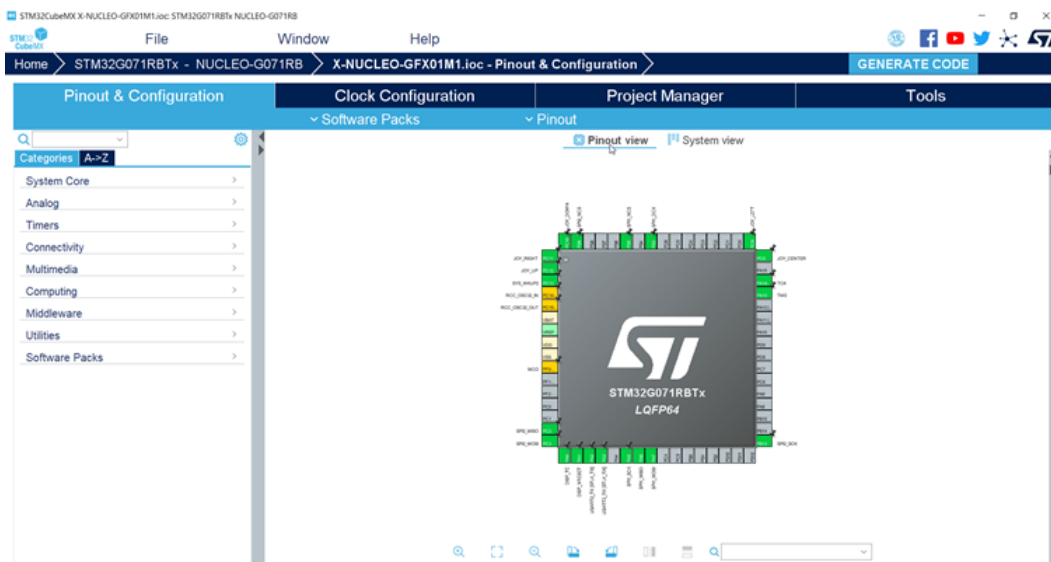
Step 4. Select the [**Software Packs**]>[**Select Components**] menu to add X-CUBE-DISPLAY as Additional Software to the project.

Figure 9. STM32CubeMX *Software Packs* window
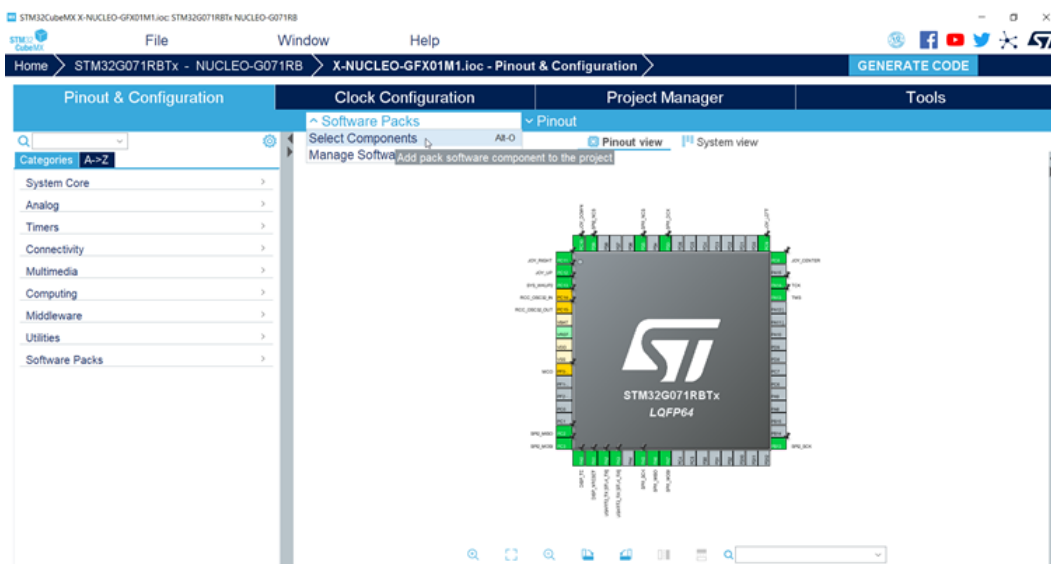
Step 5.   Choose to generate one of the enclosed sample applications or a new project from the *Software Packs Compoment Selector* for the selected microcontroller or board.

In this latter case, the user must simply implement the main application logic without bothering with the pinout and peripheral configuration code that is then automatically generated by STM32CubeMX.

Figure 10. STM32CubeMX *Software Packs Component Selector* window
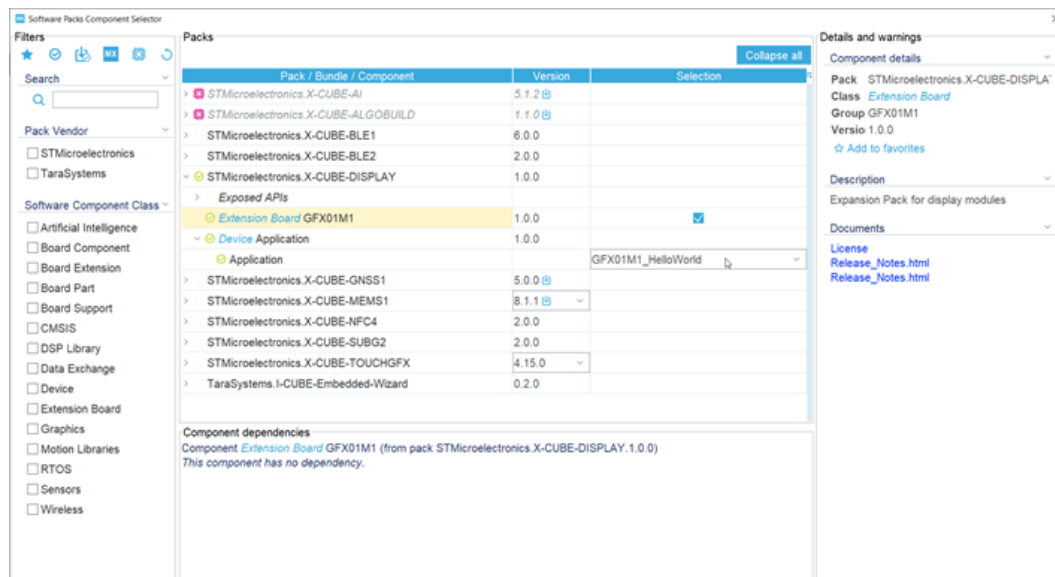
Step 6.   Press the [**OK**] button to return to STM32CubeMX *Pinout* view.

To enable and configure the selected X-CUBE-DISPLAY modules ([**Extension Board GFX01M1**], [**Device Application**] or both), select [**STMicroelectronics.X-CUBE-DISPLAY.1.0.0**] in *Software Packs* from the *Categories* list.

Figure 11. STMicroelectronics.X-CUBE-DISPLAY selection
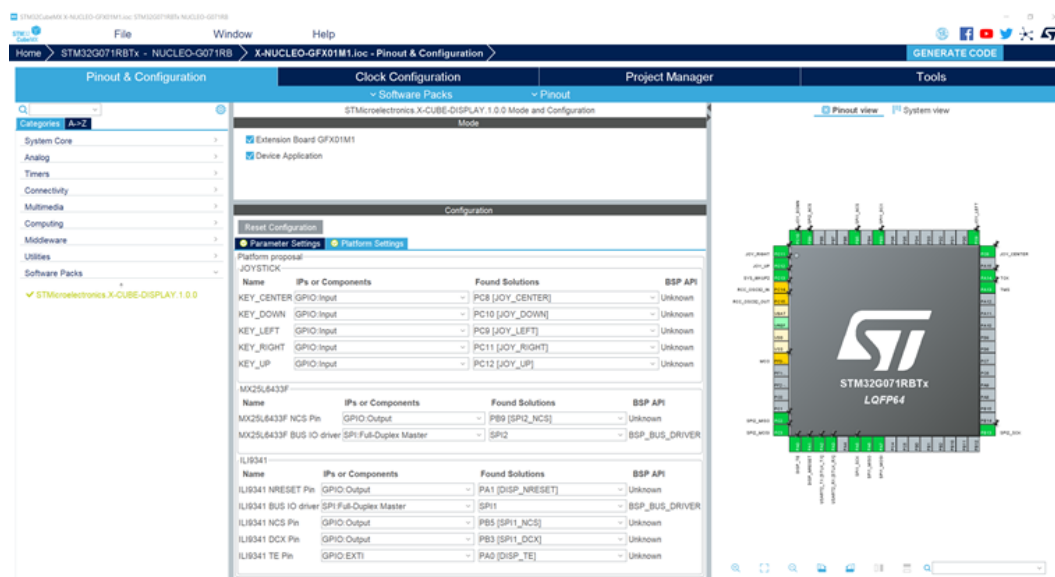
Revision history

Table 2. Document revision history

Date	Version	Changes
18-Dec-2020	1	Initial release.

Contents

List of tables

List of figures

IMPORTANT NOTICE – PLEASE READ CAREFULLY