



# FOCUS LCDs

LCDs MADE SIMPLE®

Ph. 480-503-4295 | [NOPP@FocusLCDs.com](mailto:NOPP@FocusLCDs.com)

TFT | OLED | GRAPHIC | CHARACTER | UWVD | SEGMENT | CUSTOM

## Application Note FAN4225

### *Focus LCDs MIPI Display Drivers Version 2.0*

This application note will present the new version 2.0 MIPI Display Drivers. It presents the drivers using the E43GB-I-MW405-C MIPI TFT Display Module with an STM32H747I-DISCO microcontroller development board.

## Contents

Introduction .....	5
Hardware.....	7
MIPI Driver Firmware .....	11
Development Tools.....	11
Project Setup .....	12
Code Compilation Selections.....	18
Additional Application Options .....	19
Project Build Steps.....	20
Project Layout.....	25
Demonstration Firmware .....	27
Basic Demo .....	27
Application Demo .....	28
Additional Information.....	31
Summary .....	31
LCD Handling Precautions .....	32
Disclaimer.....	32
Revision History.....	32

## List of Figures

Figure 1: E43GB-I-MW405-C and STM32H747I-Disco Running Basic Demo Firmware .....	4
Figure 2: E43GB-I-MW405-C 4.3" MIPI TFT Display .....	6
Figure 3: STM32H747I-Disco Development Board.....	7
Figure 4: KAB-M20SD-01 Focus LCDs MIPI DSI Adapter Board Top Side .....	8
Figure 5: KAB-M20SD-01 Focus LCDs MIPI DSI Adapter Board Bottom Side .....	8
Figure 6: E43GB-I-MW405-C TFT MIPI DSI Display .....	9
Figure 7: Jumper JP6 "STIK" Default Location .....	9
Figure 8: Jumper JP6 in "HS" Location .....	10
Figure 9: STM32H747I-Disco USB Cable Connections.....	10
Figure 10: STM32CubeIDE Development Environment .....	11
Figure 11: Project Explorer Import Projects.....	12
Figure 12: Import Existing Projects into Workspace .....	13
Figure 13: Project Selection Dialog .....	14
Figure 14: Folder Selection.....	15
Figure 15: Project is Selected .....	16
Figure 16: MIPI Driver Project open in Project Explorer .....	17
Figure 17: Select the Display .....	18
Figure 18: LVGL Selection.....	19
Figure 19: Select the Display Image Rotation .....	19
Figure 20: Backlight Control .....	20
Figure 21: Build Project "Hammer" .....	20
Figure 22: Build Project Selection .....	21
Figure 23: Build Log .....	21
Figure 24: Run Icon.....	22
Figure 25: Run Dialog .....	22
Figure 26: Debug Dialog .....	22
Figure 27: External Loader Section .....	23
Figure 28: External Loader Selection.....	24
Figure 29: Loader Selected.....	24
Figure 30: Project Layout .....	25
Figure 31: E43GB-I-MW405-C Running the Basic Demo with a Red Background.....	27
Figure 32: Main Application Profile Screen.....	28
Figure 33: Application Analytics Screen .....	29
Figure 34: Application Shop Screen .....	30

## Focus LCDs MIPI Display Drivers Version 2.0

---

This application note will present the version 2.0 Focus LCDs MIPI Display Drivers. The [E43GB-I-MW405-C](#) MIPI TFT Display with an [STM32H747I-DISCO](#) microcontroller board from [ST Microelectronics](#) are used to demonstrate the display drivers. The [KAB-M20SD-01](#) MIPI DSI Adapter Board interfaces between the display and microcontroller board. A basic demo showing the Focus LCDs logo and an application demo with [LVGL](#) are provided in the driver.

Previous [application notes](#) (FAN4221 through FAN4224) had discussed version 1.0 of the MIPI Display Drivers. Those drivers were built with a bare-metal approach that did not lend themselves to expansion. The version 2.0 drivers were designed with ease-of-use using the low-level STM32 HAL. [LVGL](#) is used as the graphics layer and middleware library for the demo. The application layer was designed to call the lower layers through their respective APIs (application programming interfaces).

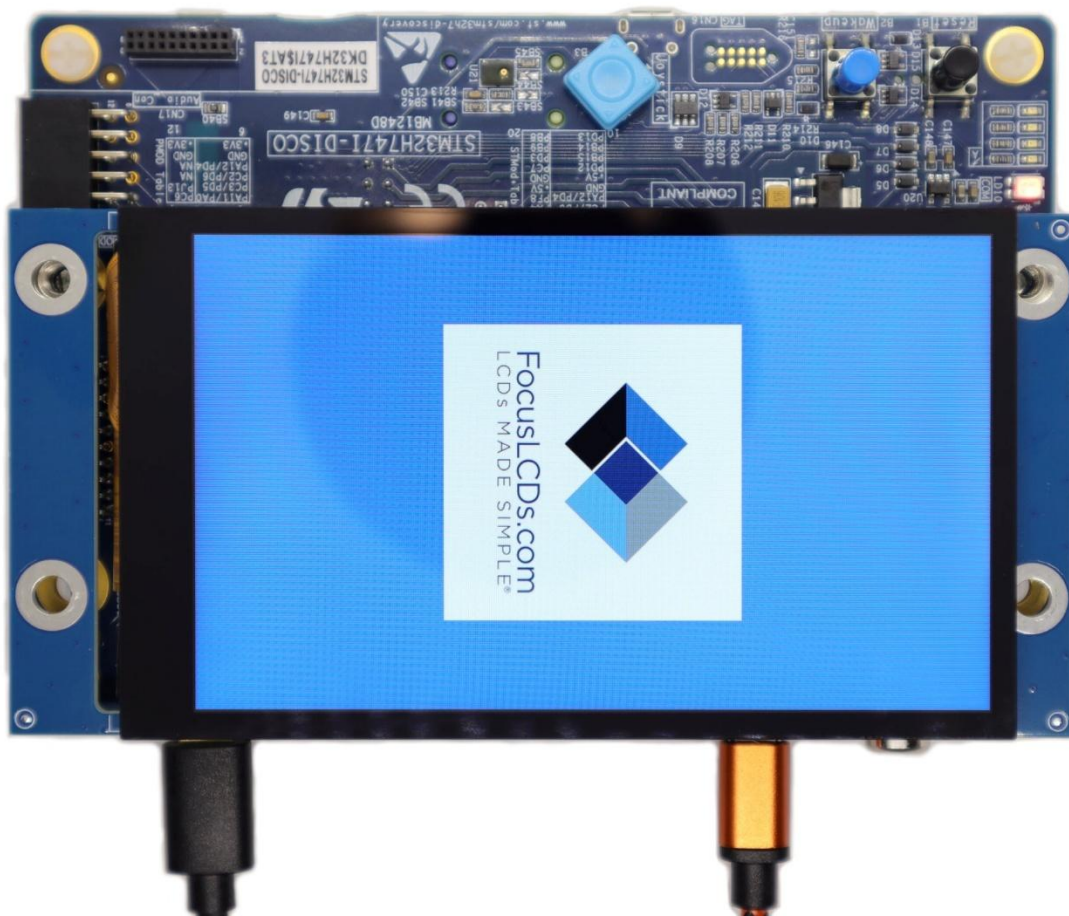


Figure 1: E43GB-I-MW405-C and STM32H747I-DISCO Running Basic Demo Firmware

## Introduction

---

Developing the new demonstration firmware for Focus LCDs MIPI TFT Displays was undertaken to address portability and ease of modification which was not present in version 1.0 drivers. The version 1.0 drivers (v1) did not consider portability between displays. Despite the shared code between the displays, the code from one display was not compatible with another similar display. This was considered when developing version 2.0 (v2) drivers.

Portability of the new drivers allows them to be used on several of Focus LCDs displays with different display controllers. Ease of modification facilitates making changes for new displays as they become available. Project and firmware layout make functional additions easier to implement.

Active development of the new drivers will include additional displays. As Focus LCDs designs new displays, they will be included in the firmware. A conditional compilation will build the firmware for a specific display without including unnecessary code. This allows displays to be easily incorporated into the driver.

The new drivers include 2 demos. The first demo shows the Focus LCDs logo on a colored background. On displays with touch capability the colored background can be changed to red, green, or blue. The second demo uses the LVGL library to generate a full application. The application has tabs, can be swiped left and right, scrolled up and down, and selections can be made that demonstrate a wide range of functionality.

The E43GB-I-MW405-C display used in the demo is a 4.3" TFT with a 480 x 800 RGB pixel resolution from Focus LCDs. This display is interfaced over a 2-lane MIPI DSI protocol with a 20-pin FPC cable. An ST Microelectronics STM32H747XI microcontroller, specifically the Discovery Development Board, will drive the display. Focus LCDs makes the KAB-M20SD-01 adapter board to connect the MIPI display to the ST Disco. This adapter board is available upon request.



*Figure 2: E43GB-I-MW405-C 4.3" MIPI TFT Display*

The main features of the E43GB-I-MW405-C are:

- 4.3-inch diagonal display, 480 x 800 RGB pixel resolution
- Up to 16.7M (24-bit) color
- 2-Lane MIPI DSI interface with 20-pin FPC cable
- Transmissive/Normally Black display mode
- White LED Backlight
- ILI9806E Display Controller
- Capacitive Touch Panel (GT911) – Touch Mode: 5-Point and Gesture
- Typical Operating Voltage: 3.3V

## Hardware

---

The development board that will drive the display is an STM32H747I-DISCO Discovery Kit. The Kit comes with its own display but for this app note, the stock display will be replaced with the E43GB-I-MW405-C.

An STM32H747I-Discovery board was chosen for this application due to the internal and external peripherals provided on the board. Internally, the Disco board has a 2-Lane capable MIPI DSI Host, LTDC (LCD-TFT Display Controller), QSPI interface, and an FMC (Flexible Memory Controller).

The DSI Host sends the image data to the display over a MIPI DSI communication interface. The LTDC takes the image data and formats it into a data stream for the DSI Host. The QSPI interfaces external flash memory for storing images. Finally, the FMC interfaces with SDRAM to contain 2 frame buffers to store each individual frame of imagery.

Other internal peripherals utilized in the demo include USART1 for sending debug messages out of a serial port. The debug messages can be used to troubleshoot the demo. Additionally, the Disco board has DMA (Direct Memory Access) for more efficient data movement without involving CPU cycles.

External peripherals on board are 128 MB QSPI NOR Flash, 32 MB SDRAM, a Micro SD Card socket, user buttons, and user LEDs. In this application the flash and SDRAM store images and frame buffers. This allows images to be preloaded in the flash and the final image sent to the display is held in the frame buffer.



Figure 3: STM32H747I-Disco Development Board

To connect the E43GB-I-MW405-C display to the STM32 Disco board the Focus LCDs MIPI DSI Adapter Board KAB-M20SD-01 adapts from the Q Strip connector to an FFC connector on the PCB. This adapter board is available to purchase from Focus LCDs.

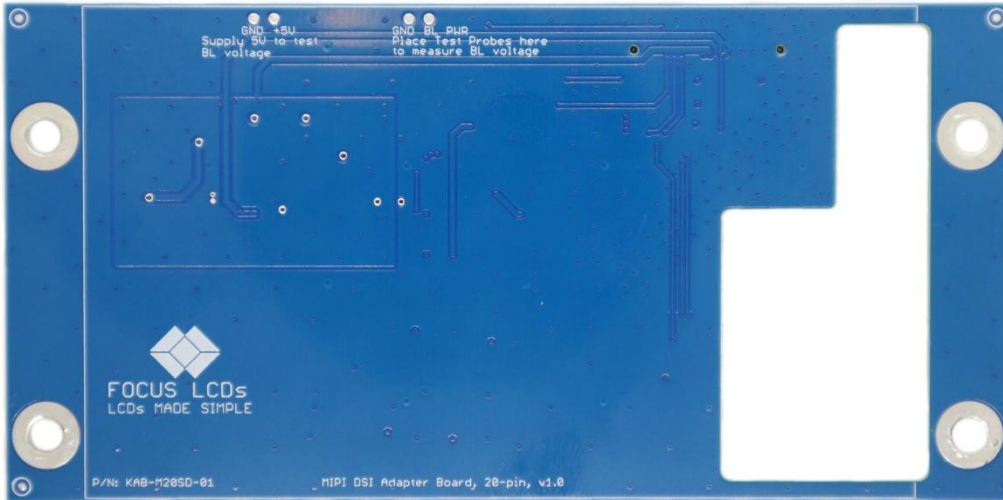


Figure 4: KAB-M20SD-01 Focus LCDs MIPI DSI Adapter Board Top Side

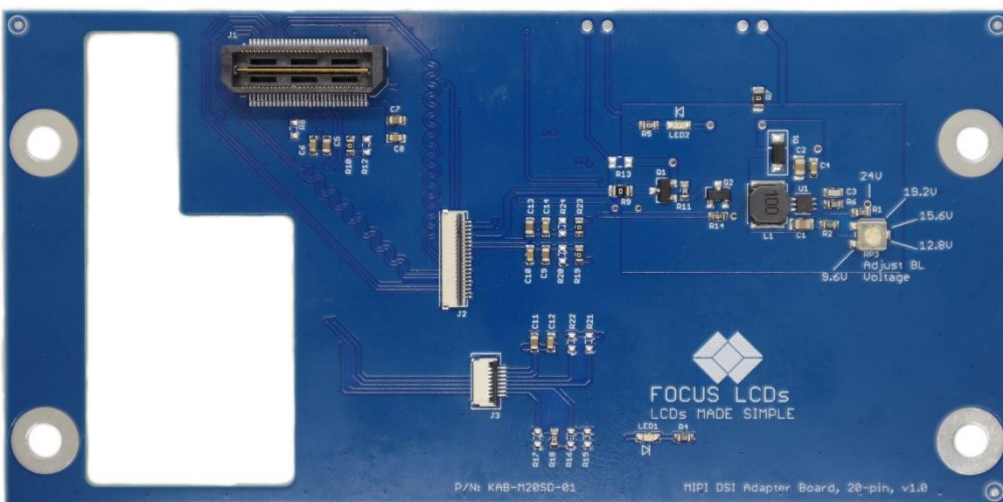


Figure 5: KAB-M20SD-01 Focus LCDs MIPI DSI Adapter Board Bottom Side

The E43GB-I-MW405-C display was chosen for this application note due to several factors. The resolution of the display, 480 x 800, is popular in many applications. The ILI9806E display controller and the GT911 touch controller are common to a variety of touch capable displays. The 4.3-inch display is mid-size between small 2.4 inch and large 7.0-inch displays.



Figure 6: E43GB-I-MW405-C TFT MIPI DSI Display

Power requirements of the demonstration necessitate changing the position of jumper JP6 and using 2 micro-USB cables. The backlight of different displays can draw up to 2 watts of power. Combine that with the power requirements of the MCU and external peripheral chips, a single USB cannot provide enough current to drive the display and all peripherals.

Jumper JP6 needs to be moved from “STIk” to “HS” to allow current to be drawn from the USB\_OTG\_HS micro-USB socket.

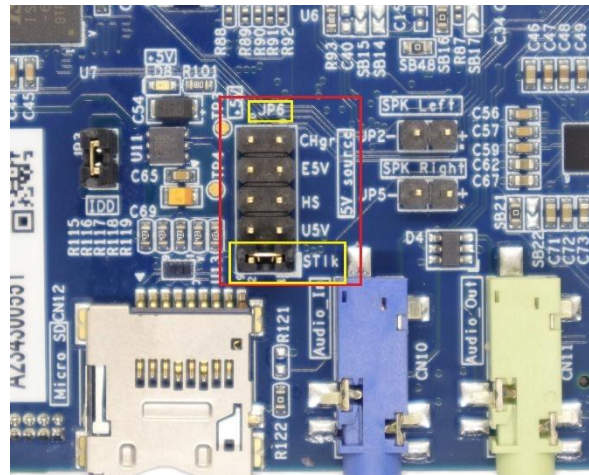


Figure 7: Jumper JP6 “STIk” Default Location

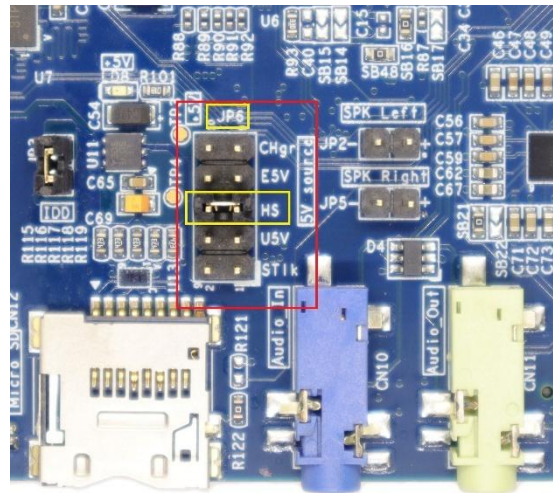


Figure 8: Jumper JP6 in “HS” Location

The 2 micro-USB cables are connected to the micro-USB sockets marked ST-LINK and USB\_OTG\_HS.

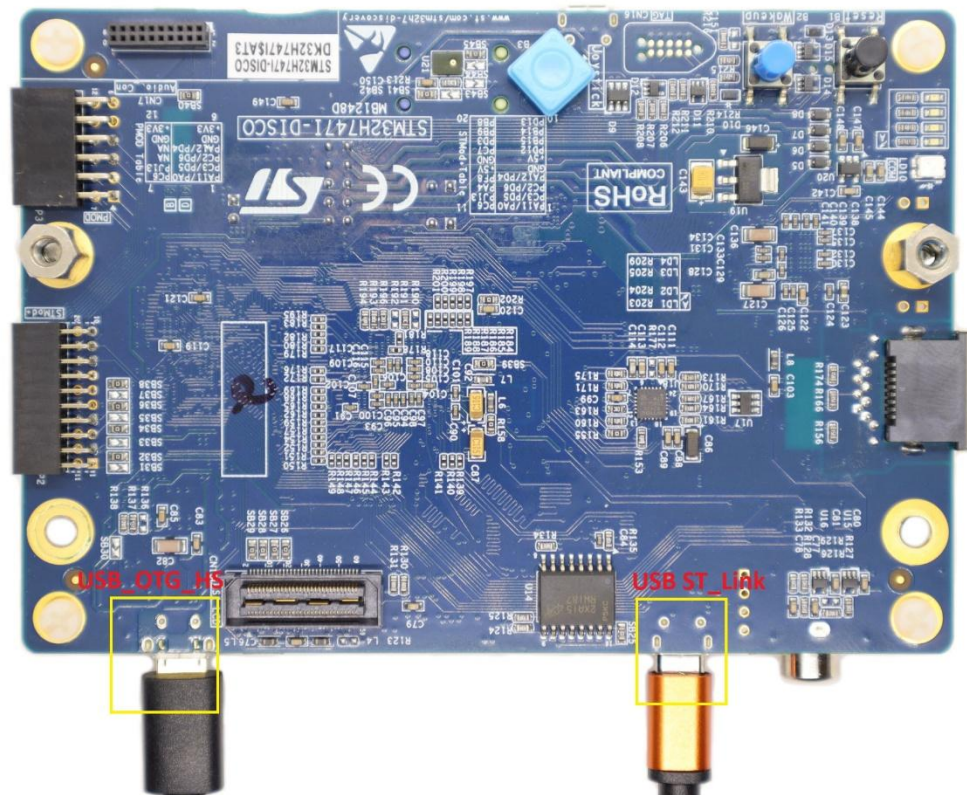


Figure 9: STM32H7471-Disco USB Cable Connections

## MIPI Driver Firmware

---

### Development Tools

Developing applications typically will use STM32CubeMX for setting up peripheral initialization and STM32CubeIDE (integrated development environment) to code the application side. These tools can be downloaded from the ST website.

After installation, STM32CubeIDE might request additional downloads depending on the device series integrated into the project. When developing the firmware, an additional download for the H7 series of devices was required.

In this application, only the STM32CubeIDE will be used in the development of the firmware. The project and firmware have already been created and configured. Importing the project will be required as all the other setup has been completed. Focus LCDs can provide the project files upon request.

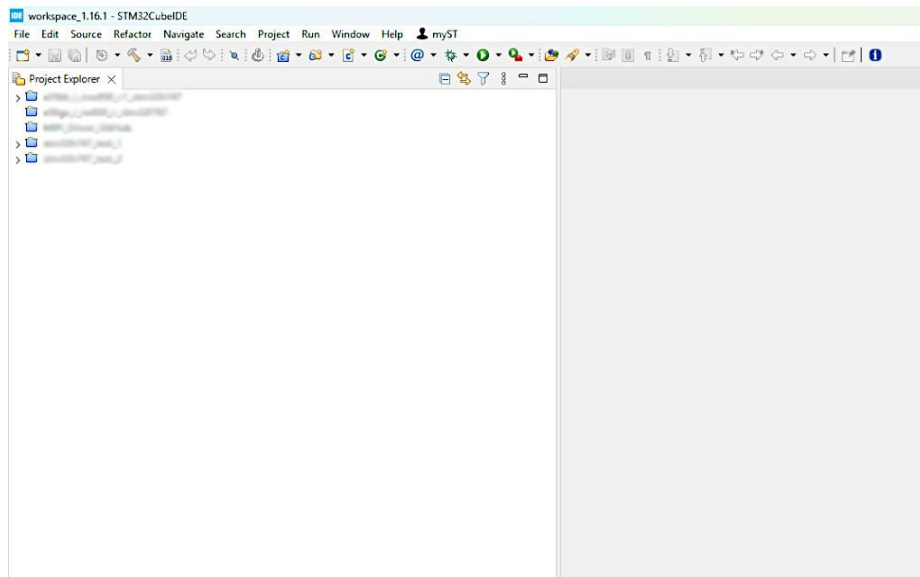


Figure 10: STM32CubeIDE Development Environment

## Project Setup

The project files are delivered as a zipped archive. Once the files have been unzipped, the main project can be imported into STM32CubeIDE. The steps are as follows:

Import the project into STM32CubeIDE by selecting “Import Projects” from Project Explorer.

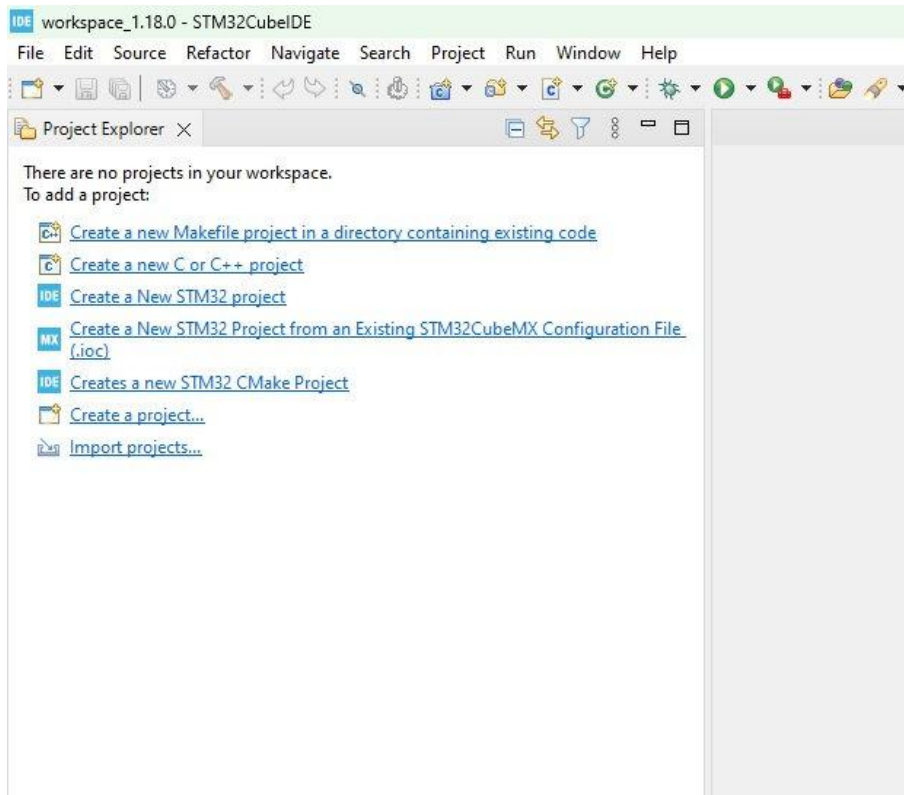


Figure 11: Project Explorer Import Projects

In the Import Dialog, select General -> Existing Projects into Workspace.

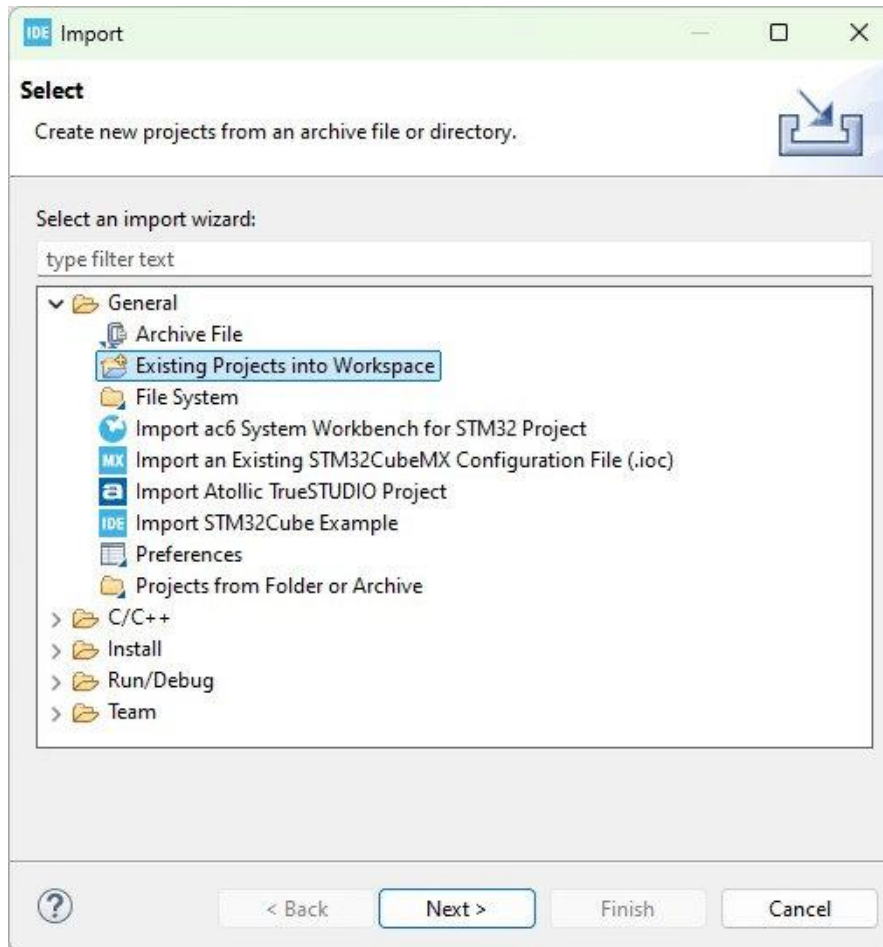


Figure 12: Import Existing Projects into Workspace

After selecting “Next”, the dialog to select the project is displayed.

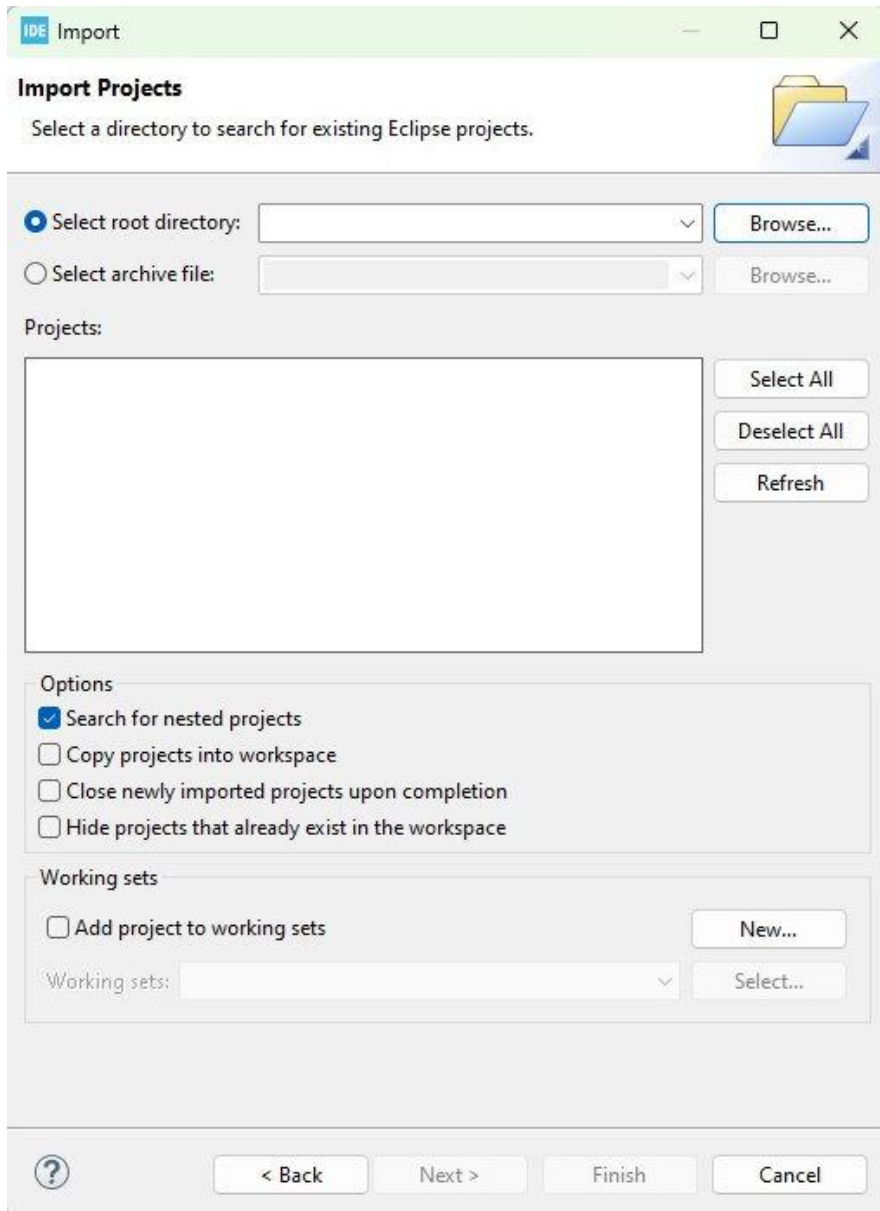


Figure 13: Project Selection Dialog

Select the folder where the project is located.

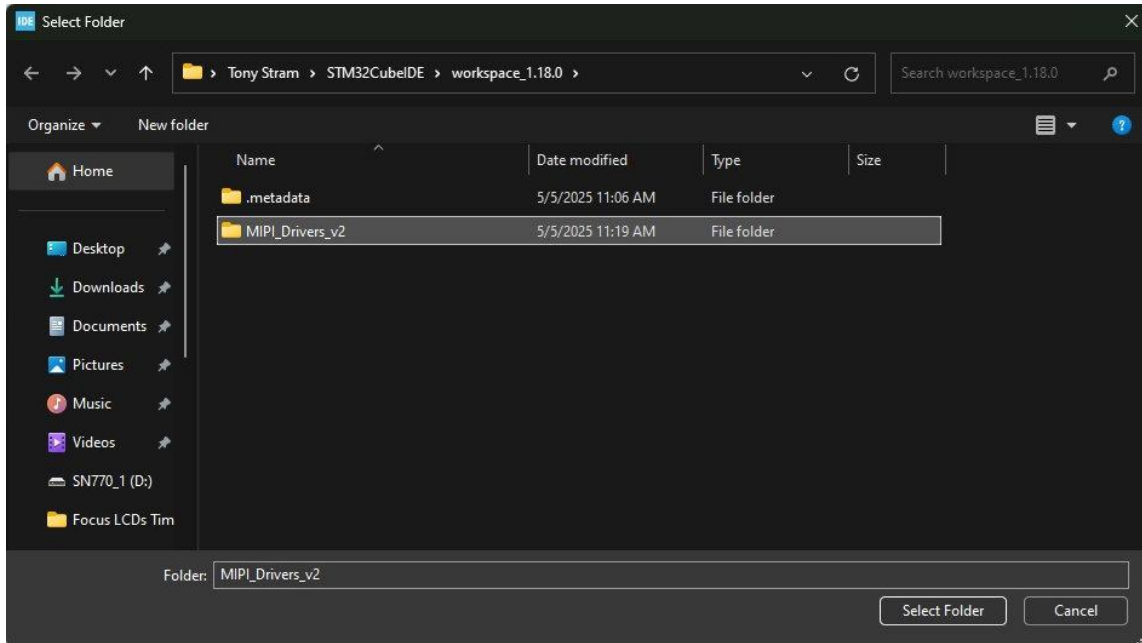


Figure 14: Folder Selection

Once the folder is selected, the Project Selection dialog will pop up. Under “Projects”, make sure the project in the folder is checked.

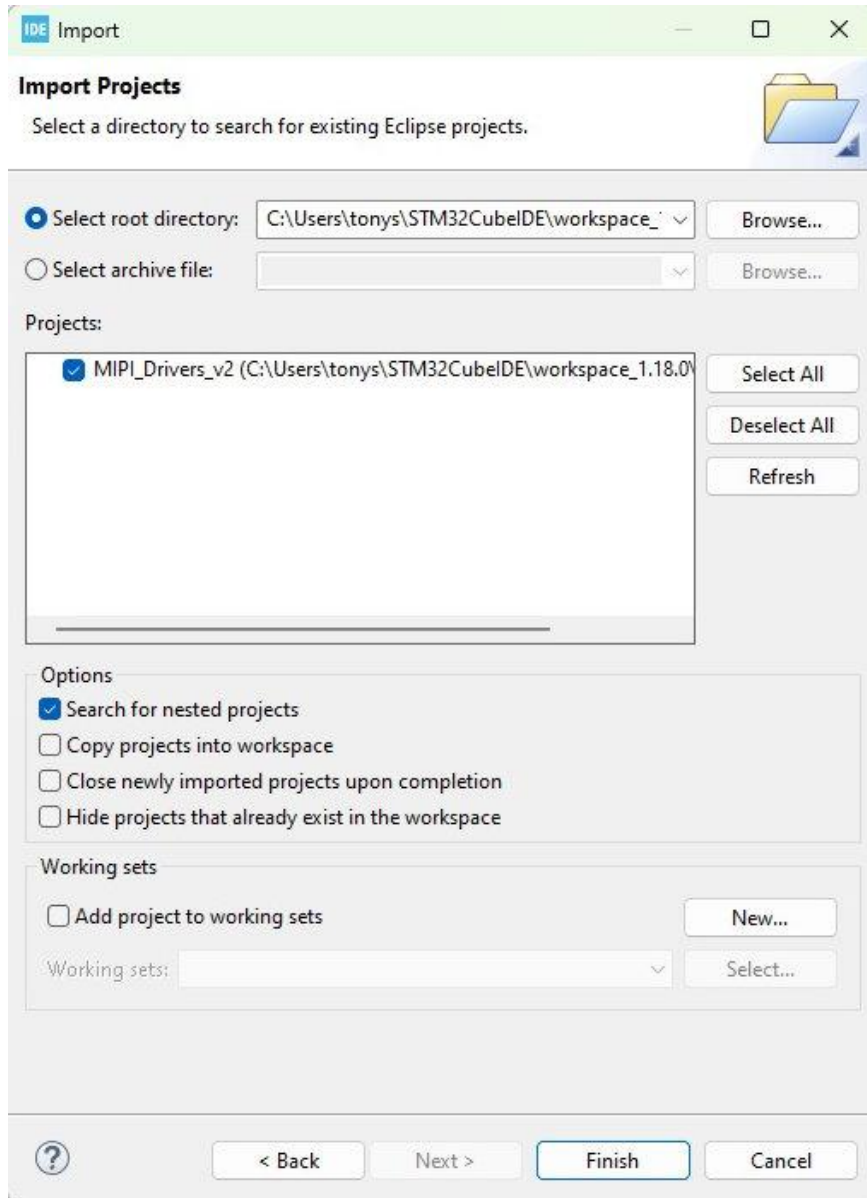


Figure 15: Project is Selected

Once those steps are completed the project will open in Project Explorer.

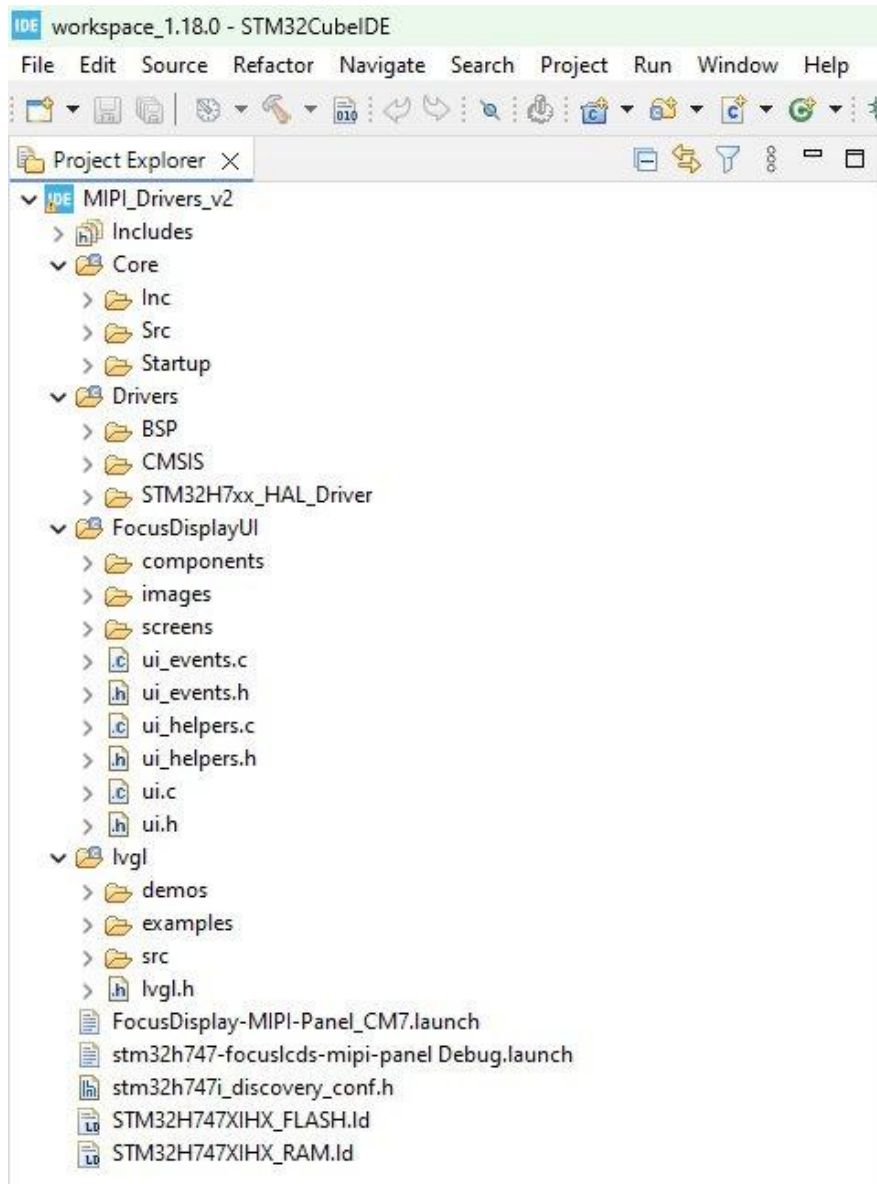
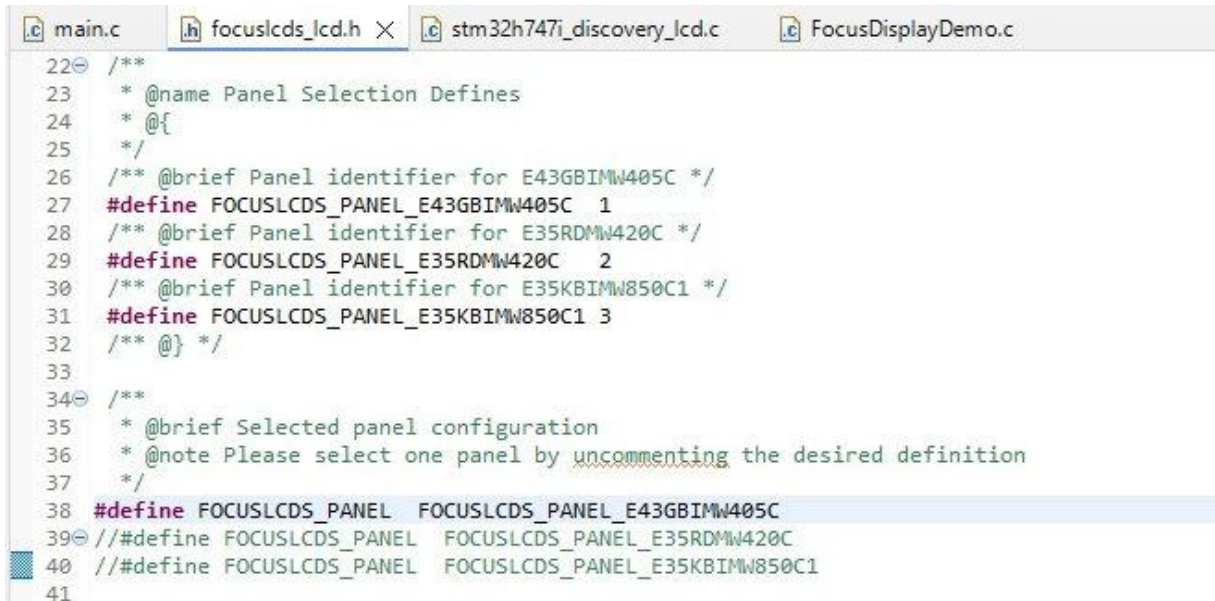


Figure 16: MIPI Driver Project open in Project Explorer

## Code Compilation Selections

Selecting the display is the next step in the process. Uncommenting the display causes conditional code compilation during the build step. This ensures that only the code for the selected display is included in the firmware.

The file is found in Drivers/BSP/Components/focuslcds/focuslcds\_hal/focuslcds\_lcd.h



```
22 //  
23 * @name Panel Selection Defines  
24 * @{  
25 */  
26 /** @brief Panel identifier for E43GBIMW405C */  
27 #define FOCUSLCDS_PANEL_E43GBIMW405C 1  
28 /** @brief Panel identifier for E35RDMW420C */  
29 #define FOCUSLCDS_PANEL_E35RDMW420C 2  
30 /** @brief Panel identifier for E35KBIMW850C1 */  
31 #define FOCUSLCDS_PANEL_E35KBIMW850C1 3  
32 /** @} */  
33  
34 /**  
35 * @brief Selected panel configuration  
36 * @note Please select one panel by uncommenting the desired definition  
37 */  
38 #define FOCUSLCDS_PANEL FOCUSLCDS_PANEL_E43GBIMW405C  
39 // #define FOCUSLCDS_PANEL FOCUSLCDS_PANEL_E35RDMW420C  
40 // #define FOCUSLCDS_PANEL FOCUSLCDS_PANEL_E35KBIMW850C1  
41
```

Figure 17: Select the Display

LVGL can be selected to provide a graphical and application library of components to display on the screen. The basic demo does not use LVGL and LVGL is commented out. Uncomment the line: `#define UNUSED_LVGL 1` to enable LVGL. LVGL will be uncommented for this section of the application note.



```

10 /* Includes */
11 #include "main.h"
12 #include <string.h>
13 #include <stdio.h>
14 #include "lvgl.h"
15 #include "lvgl_port_touchpad.h"
16 #include "lvgl_port_display.h"
17 #include "stm32h747i_discovery_qspi.h"
18 #include "../FocusDisplayUI/ui.h"
19 #include "lv_demos.h"
20 #include "focuslcds_log.h"
21
22 #define LOG_TAG "MAIN"
23 #define VERSION "1.0.0" /* Version of the application */
24
25 // #define UNUSED_LVGL 1 /* Uncomment to enable LVGL support */
26
27 #ifndef UNUSED_LVGL
28 // #define FOCUSLCDS_LOGO_DEMO_WITH_LVGL 1 /* Uncomment to enable LVGL demo */
29 #endif
30

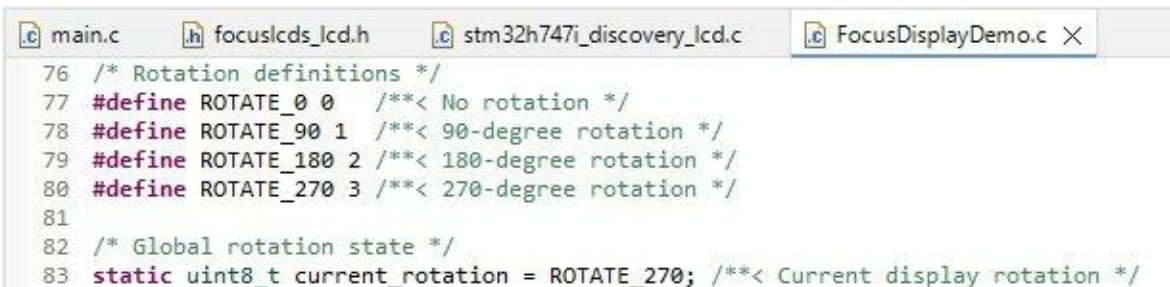
```

Figure 18: LVGL Selection

## Additional Application Options

Application options are also included in the code base. The ability to rotate the image in the display and to disable the backlight are options. These can be used within the application code for different effects.

Rotation:



```

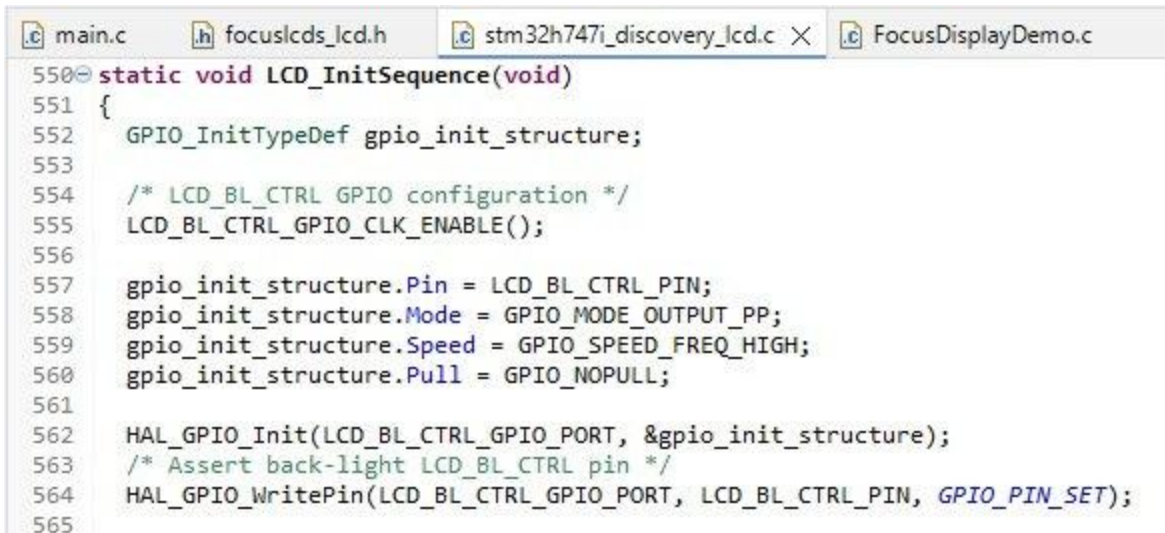
76 /* Rotation definitions */
77 #define ROTATE_0 0 /**< No rotation */
78 #define ROTATE_90 1 /**< 90-degree rotation */
79 #define ROTATE_180 2 /**< 180-degree rotation */
80 #define ROTATE_270 3 /**< 270-degree rotation */
81
82 /* Global rotation state */
83 static uint8_t current_rotation = ROTATE_270; /**< Current display rotation */

```

Figure 19: Select the Display Image Rotation

Update the variable `current_rotation` to any of the predefined rotations to display the image in 1 of 4 orientations on the screen.

Backlight:



```

550 static void LCD_InitSequence(void)
551 {
552     GPIO_InitTypeDef gpio_init_structure;
553
554     /* LCD_BL_CTRL GPIO configuration */
555     LCD_BL_CTRL_GPIO_CLK_ENABLE();
556
557     gpio_init_structure.Pin = LCD_BL_CTRL_PIN;
558     gpio_init_structure.Mode = GPIO_MODE_OUTPUT_PP;
559     gpio_init_structure.Speed = GPIO_SPEED_FREQ_HIGH;
560     gpio_init_structure.Pull = GPIO_NOPULL;
561
562     HAL_GPIO_Init(LCD_BL_CTRL_GPIO_PORT, &gpio_init_structure);
563     /* Assert back-light LCD_BL_CTRL pin */
564     HAL_GPIO_WritePin(LCD_BL_CTRL_GPIO_PORT, LCD_BL_CTRL_PIN, GPIO_PIN_SET);
565

```

Figure 20: Backlight Control

The backlight is set to turn on when the application starts. It can be turned off by replacing `GPIO_PIN_SET` with `GPIO_PIN_RESET` in the HAL write pin function.

## Project Build Steps

Building and loading the firmware onto the Disco board is the final steps in the process. There are several ways to build the project. The first is to select the hammer icon to build the project.

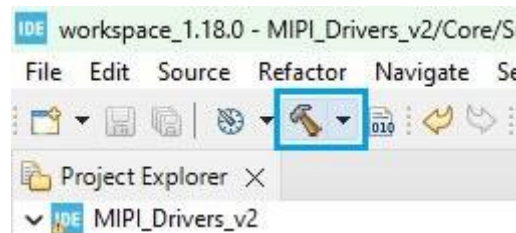


Figure 21: Build Project "Hammer"

Another option is to right-click the project in Project Explorer and select Build Project.

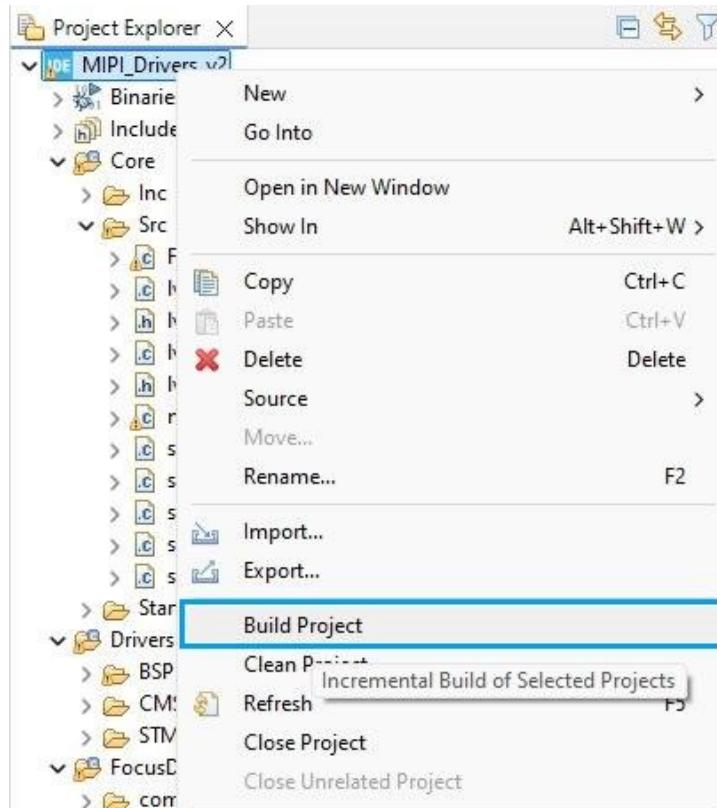


Figure 22: Build Project Selection

Once the build is complete, the Build Log will show any errors and warnings. The warnings shown in the image are due to unused variables that are part of the LVGL library. They do not cause any issues in the running of the demo.

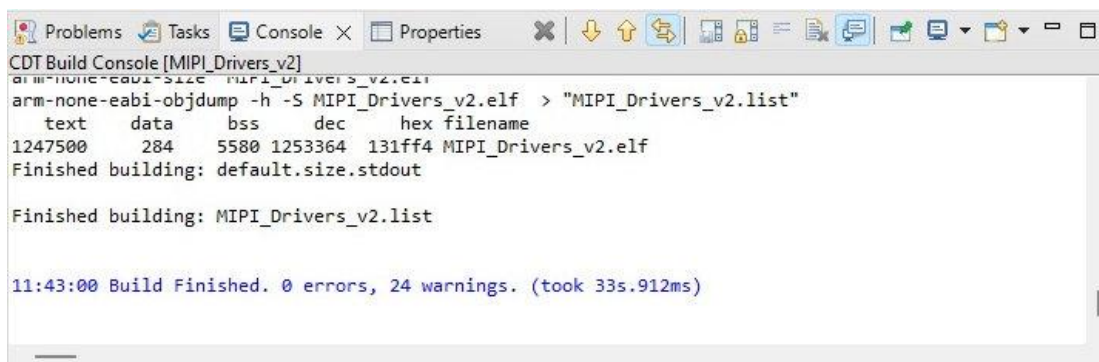


Figure 23: Build Log

Now the firmware can be downloaded into the Disco board by either clicking on the Run icon in the toolbar or selecting Run As in the right-click menu of the project.



Figure 24: Run Icon



Figure 25: Run Dialog

This will be the first time downloading and running the debugger for this project and will pull up the Debug Dialog.

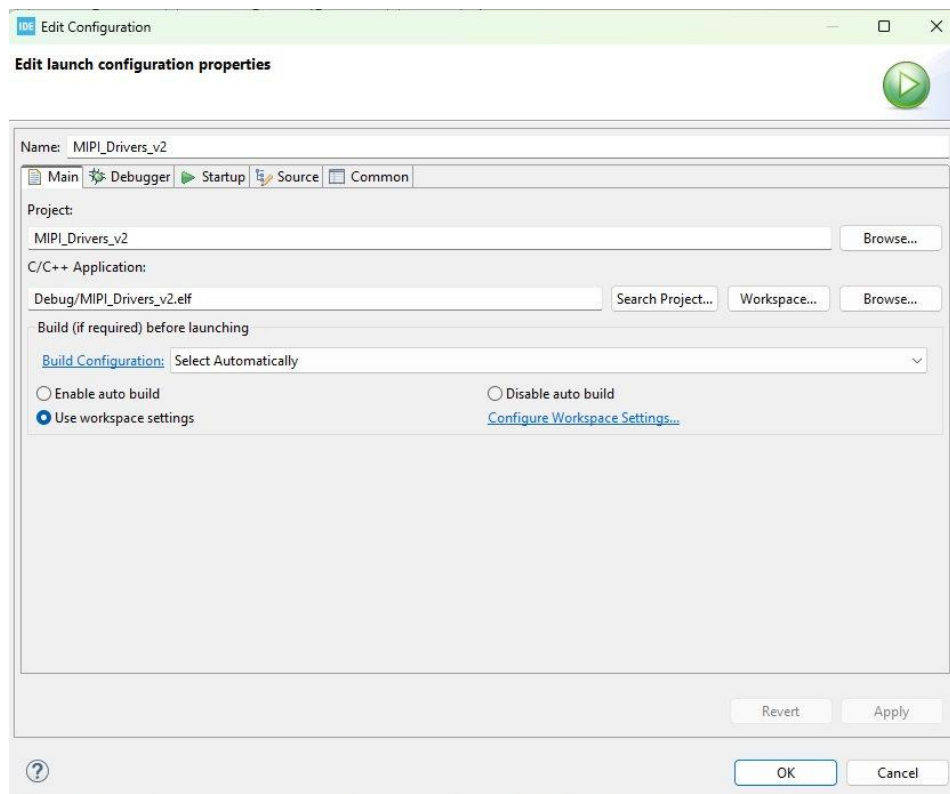


Figure 26: Debug Dialog

The project uses the external flash chips on the Disco board, 2 of the MT25TL01G QSPI Flash chips from [Micron Technology](#). An external flash loader will be invoked during the downloading process and requires a helper program to perform this operation. The correct loader must be selected, in this project it is MT25TL01G\_STM32H747I-DISCO.stldr.

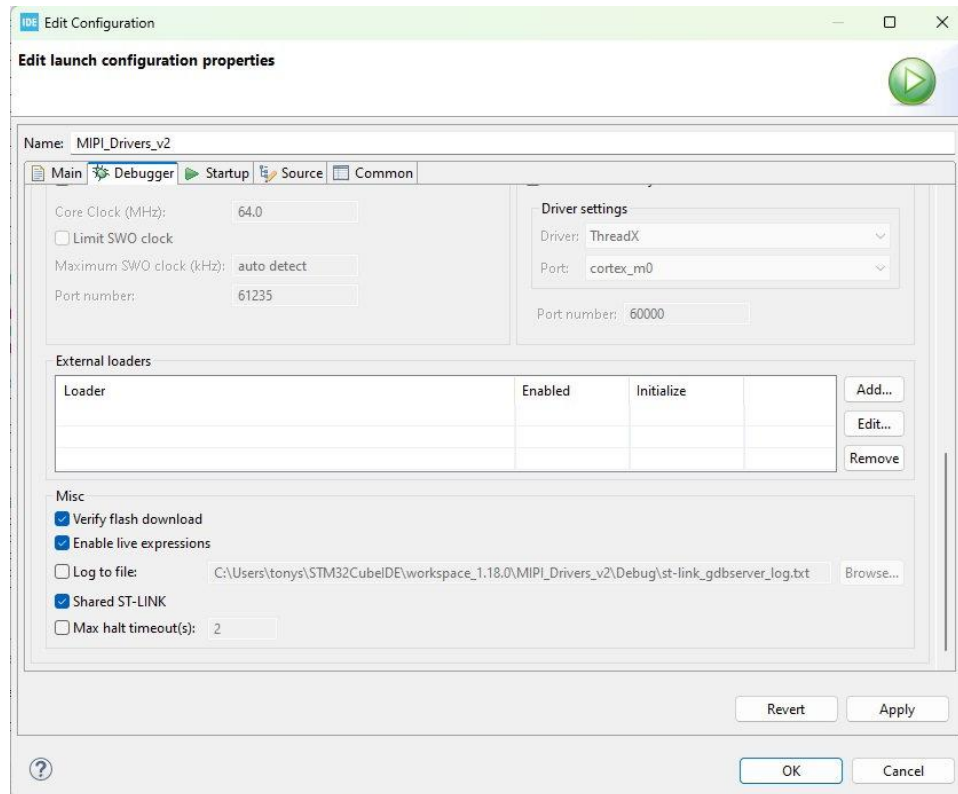


Figure 27: External Loader Section

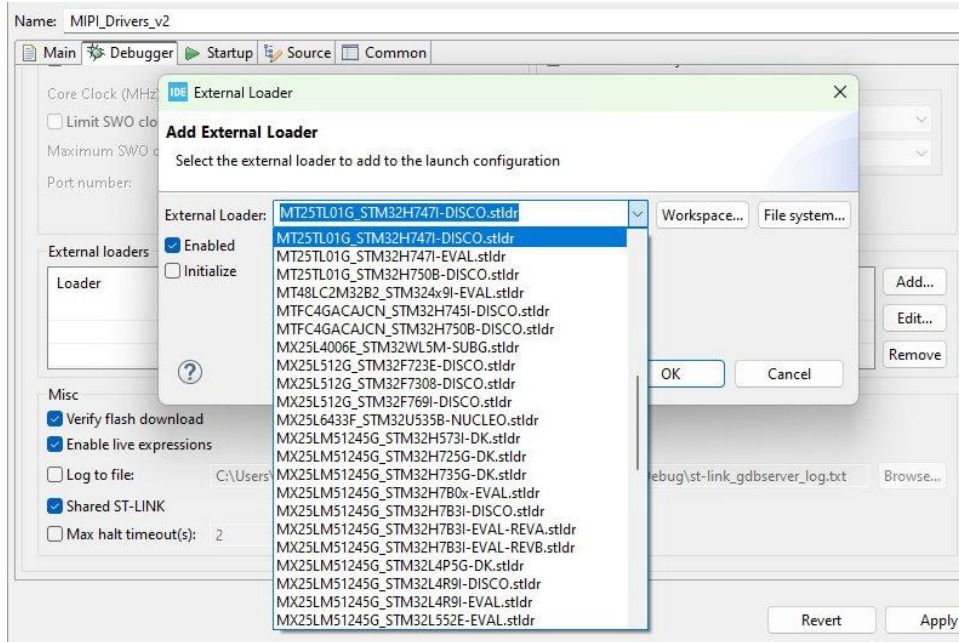


Figure 28: External Loader Selection

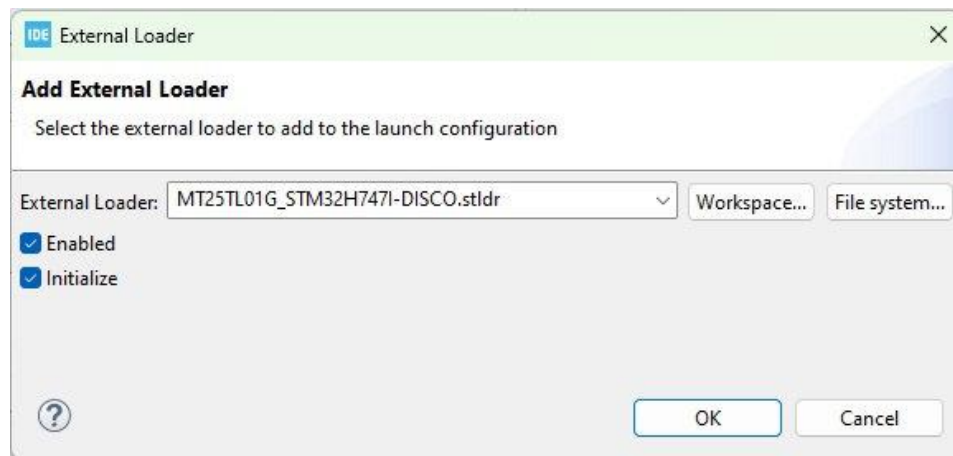


Figure 29: Loader Selected

When making the external loader selection. Make sure “Enable” and “Initialized” are checked otherwise the ST-Link will not start correctly and the download will fail.

## Project Layout

The project layout has been designed to separate the different layers of code. Core code is the main code driving the application. It contains “Main” and other files related to getting the microcontroller running. Drivers is where the BSP (board support package), the ARM CMSIS (ARM’s common microcontroller software interface standard), and ST Micro’s HAL (hardware abstraction layer) code is located. The LVGL folder contains the LVGL library code. Finally, any code specific to the Focus LCDs UI is in the FocusDisplayUI folder.

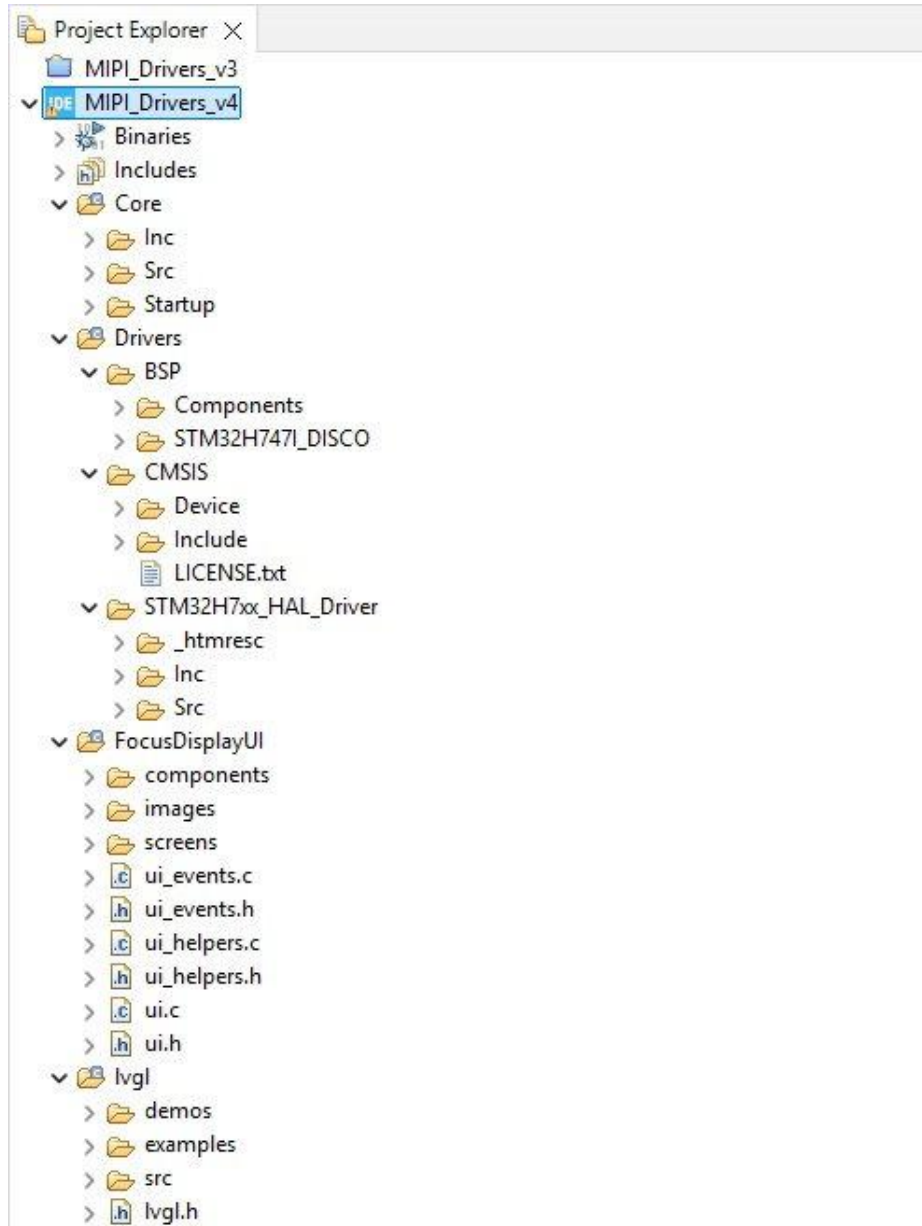


Figure 30: Project Layout

# FAN4225



When creating a new project in STM32 CubeIDE, it imports the code for ARM CMSIS and ST's HAL. Code in the BSP for the components on the Disco board are from STM32CubeH7 which contains all the low-level APIs and example code ST provides. The UI code comes as source and header files from LVGL and is incorporated in the application demo. All the low-level code for driving the E43GB-I-MW405-C display was created by Focus LCDs. The Focus LCDs specific application demo code was written by Focus LCDs using the LVGL library APIs.

## Demonstration Firmware

---

### Basic Demo

The firmware is configured to run the basic demonstration upon downloading into the Disco board. A blue background with the Focus LCDs logo centered on the screen starts the demo. When the corners of the screen are touched, the background color changes between red, green, and blue. The image below highlights the red background.

The demo was designed to be simplistic to show the display working. Displays that have touch capability allow touch interactions to control the background color and if the logo is shown.

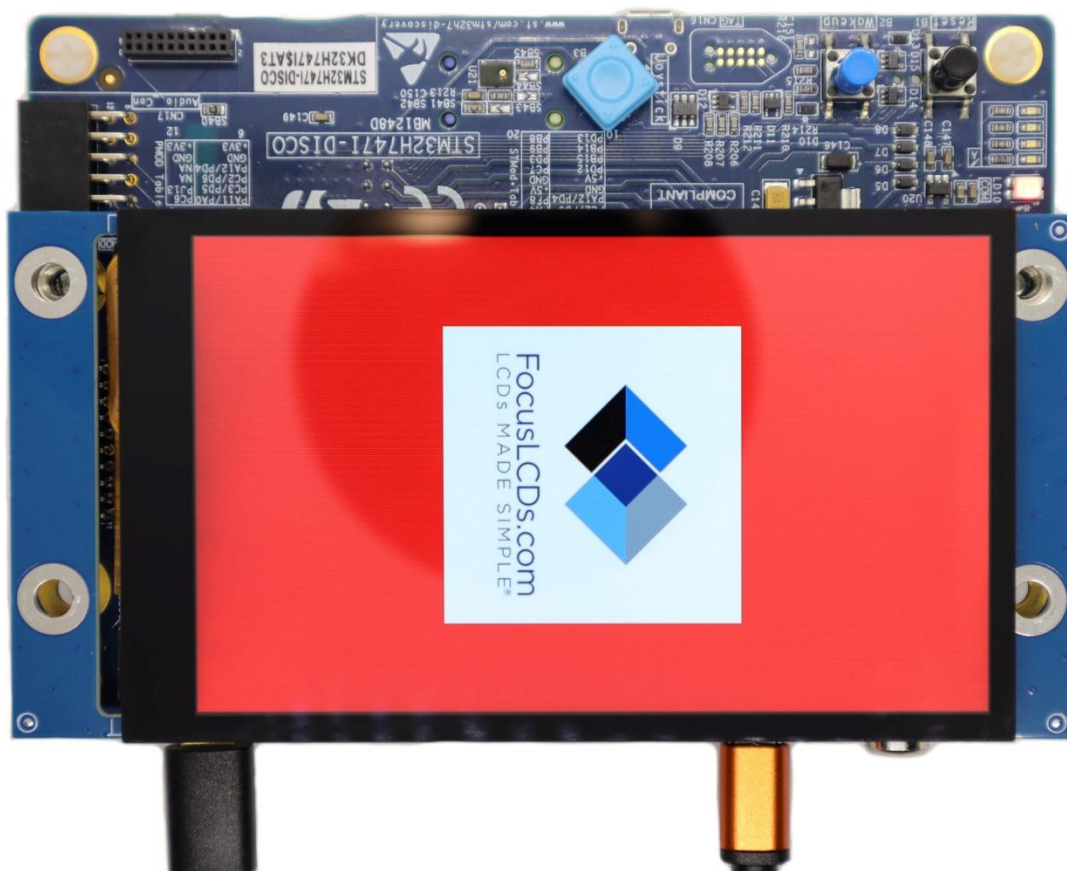


Figure 31: E43GB-I-MW405-C Running the Basic Demo with a Red Background

## Application Demo

The application demo shows how a user interface can look on the E43GB-I-MW405-C display. It provides a simulated, functional application like a cell phone application. The demo has 3 tabs corresponding to profile, analytics, and a shop. The tabs can be selected by touching them or swiping left and right. Each tab can be scrolled up and down to see additional simulated information. Buttons, check boxes, sliders, and radio button selections.

The profile tab simulates a social media profile page of a user.



Figure 32: Main Application Profile Screen

# FAN4225

The analytics tab simulates real-time data showing current marketing and revenue trends. In addition, a monthly target section simulates moving targets as concentric circles that increase and decrease.



Figure 33: Application Analytics Screen

The shop tab shows a simulation of the backend of a web shop. A monthly summary (static image), the top products sold, and selectable notifications highlight the display and its capabilities. The 4.3-inch screen size with a 480 x 800 resolution handles 24-bit color graphics, text, and HMI (human-machine interfaces) with ease.



Figure 34: Application Shop Screen

## Additional Information

The project source code can be provided by Focus LCDs upon request. The display and adapter board can be purchased from Focus LCDs.

## Summary

This application note explained the reason behind updating the MIPI drivers, discussed the hardware used, and how to import the project into the STM32CubeIDE. Due to power requirements, a jumper was changed, and a second USB connection was added to supply more current. The project folder layout was presented along with a quick discussion of the code in each folder. Next, how to download the firmware into the Disco board was shown with the addition of loading to external flash chips. In the last section the basic and full application demos were presented.

### LCD Handling Precautions

- Do not store the TFT-LCD module in direct sunlight, best stored in a dark place
- Do not leave it exposed to high temperature and high humidity for a long period of time
- Recommended temperature range is 0 to 35 °C, relative humidity should be less than 70%
- Stored modules away from condensation as formation of dewdrops may cause an abnormal operation or failure of the module.
- Protect the module from static discharge
- Do not press or scratch the surface and protect it from physical shock or any force

### Disclaimer

Buyers and others who are developing systems that incorporate FocusLCDs products (collectively, “Designers”) understand and agree that Designers remain responsible for using their independent analysis, evaluation, and judgment in designing their applications and that Designers have full and exclusive responsibility to assure the safety of Designers' applications and compliance of their applications (and of all FocusLCDs products used in or for Designers’ applications) with all applicable regulations, laws, and other applicable requirements.

Designer represents that, with respect to their applications, Designer has all the necessary expertise to create and implement safeguards that:

- (1) anticipate dangerous consequences of failures
- (2) monitor failures and their consequences, and
- (3) lessen the likelihood of failures that might cause harm and take appropriate actions.

The designer agrees that prior to using or distributing any applications that include FocusLCDs products, the Designer will thoroughly test such applications and the functionality of such FocusLCDs products as used in such applications.

### Revision History

Revision	Notes	Date
1.0.0	Initial Version	4/30/2025