

EXAMPLE NAME: GUI_SEGGERLIBRARY_EXAMPLE_XMC45

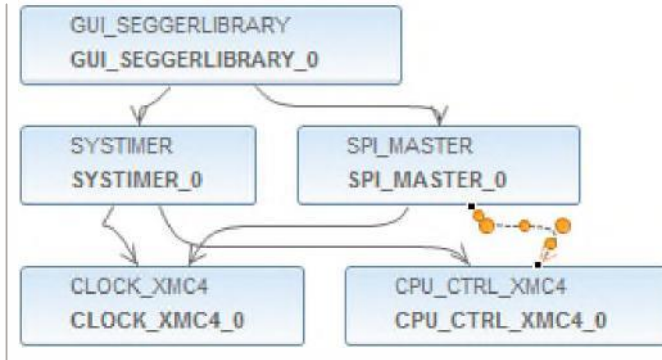
OVERVIEW: This example demonstrates how to use the GUI_SEGGERLIBRARY APP to display on the OLED.

DESCRIPTION:

Upon instantiation, GUI_SEGGERLIBRARY APP aggregates SPI_MASTER, SYSTIMER APP as shown in below snapshot.

It requires configuring:

1. SPI pins using SPI_MASTER APP
2. GPIO pin for GUI_SEGGERLIBRARY APP (Command Select/RS)

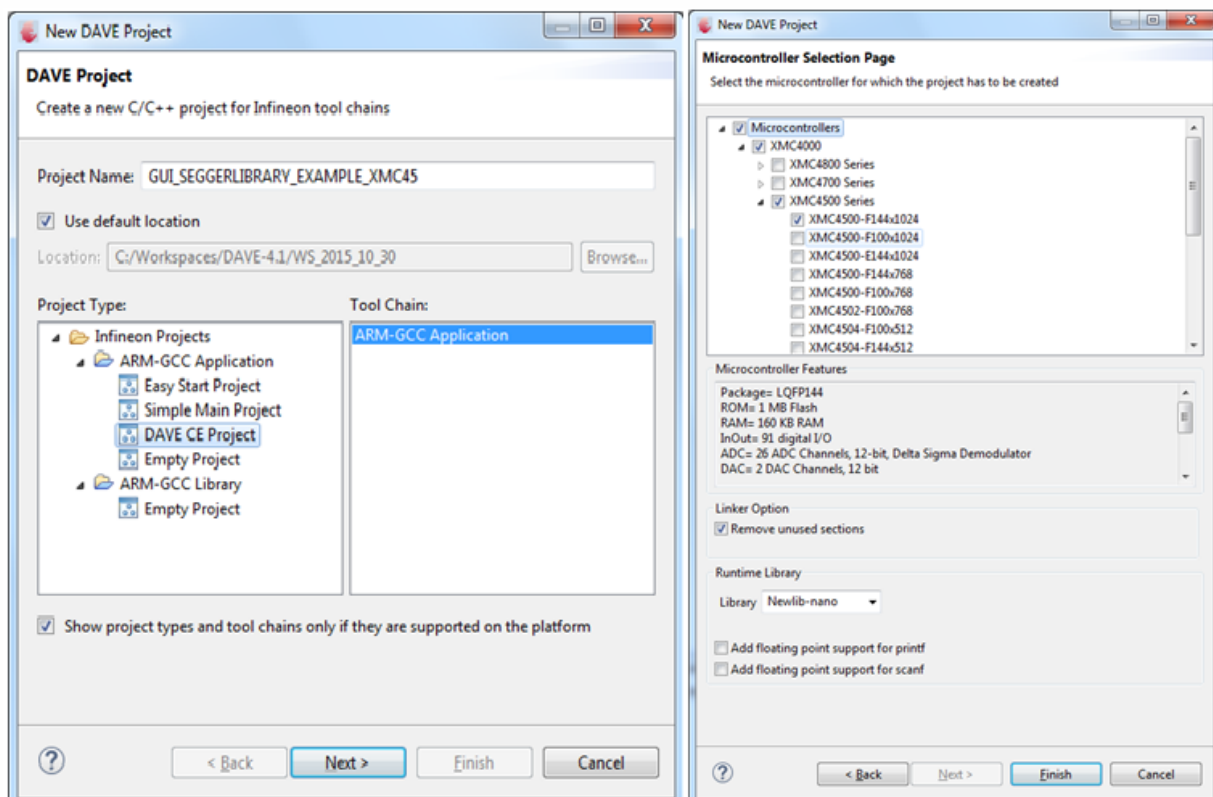


REQUIREMENTS:

Boards Required: XMC4500 Hexagon Board, HMI_OLED board

HOW TO CREATE THE PROJECT:

1. Open the DAVE CE and use “New” on the toolbar to add a new DAVE Project.
Select Microcontroller XMC4500-F144x1024.



2. Use the "Add New APP" in the toolbar. Instantiate GUI_SEGGERLIBRARY APP. Configure the APP with the following configurations.

The screenshot shows the 'Seggerlibrary Settings' dialog box with the 'Configuration' tab selected. The 'Initialize GUI library during APP initialization' checkbox is checked. The 'GUI library' dropdown is set to 'Basic'. A text box below shows 'Basic library: It provides the access only to text based graphics.' A note states: 'It is required to add the below mentioned library name in the project settings. For more details, refer Path Settings tab.' Below this, a text box contains 'GUI_basic_xmc4'. The 'OS support' dropdown is set to 'None'. The 'Enable cache for non-readable display' checkbox is unchecked, and the 'Cache size' is set to '4096'. The 'Enable user defined section in linker file for cache memory' checkbox is also unchecked. A 'NOTE' section explains that the user-defined section for cache memory is only applicable for XMC48/47/45 devices. A final note mentions that bigger displays require more cache size which cannot fit in internal RAM, suggesting a user-defined section in the linker file pointing to external RAM/SDRAM.

GUI_SEGGERLIBRARY_0

Seggerlibrary Settings LCD Settings Syncoam Display Settings Path Settings

Configuration

☒ Initialize GUI library during APP initialization

GUI library: Basic

Features supported by GUI Library:

Basic library: It provides the access only to text based graphics.

It is required to add the below mentioned library name in the project settings.
For more details, refer Path Settings tab.

GUI_basic_xmc4

OS support: None

☐ Enable cache for non-readable display

Cache size: 4096

☐ Enable user defined section in linker file for cache memory

NOTE:
User defined section for cache memory is applicable only for XMC48/47/45 devices.
For other devices, it is disabled due to non-availability of EBU support.

Bigger displays require more cache size which cannot fit in internal RAM.
So, user should write his own section called ".user_defined_section" in linker
file which can point to external RAM/SDRAM area for cache memory.

Refer usage section in documentation for more details about linker file modification.

The screenshot shows the 'Seggerlibrary Settings' dialog box with the 'LCD Settings' tab selected. The 'Display controller' dropdown is set to 'Syncoam(SEP5525F) Display'. The 'RGB interface' is set to '8-bit'.

GUI_SEGGERLIBRARY_0

Seggerlibrary Settings LCD Settings Syncoam Display Settings Path Settings

Selection

Display controller: Syncoam(SEP5525F) Display

RGB interface: 8-bit

GUI_SEGGERLIBRARY_0

Seggerlibrary Settings

LCD Settings

Syncoam Display Settings

Path Settings

Configuration

☒ SPI interface

Display orientation: 0 degree

Physical Size Configuration

X-Axis: 160

Y-Axis: 128

User Callback Functions

Display initialization:

LCD read:

LCD write(command):

LCD write(data):

LCD write(multiple):

NOTE: Please refer to help documentation (APP Configuration Parameters) about the callback function declaration in application code

GUI_SEGGERLIBRARY_0

Seggerlibrary Settings

LCD Settings

Syncoam Display Settings

Path Settings

In order to compile the GUI APPs, it is required to follow below two steps.

Step 1:
Add the below mentioned paths under Project Settings->C/C++ Build->Settings->ARM GCC C Compiler->Directories as shown in below snapshot.

"\${eclipse_home}/emWin/Start/GUI/inc"

"\${workspace_loc}/\${ProjName}/Dave/Generated/GUI_SEGGERLIBRARY/Config"

Step 2:
Add the library and library path in Project Settings -> C/C++ Build -> Settings -> ARM GCC C Linker -> Libraries as shown in below snapshot.

Below highlighted library name should match with the Seggerlibrary Settings tab.
Library search path(-L) should be the same as shown below highlighted one.

Libraries (-l)

m

GUI_basic_xmc4

Library search path (-L)

../libm

"\${eclipse_home}/emWin/Start/GUI"

Note: Refer help documentation Overview section to copy the above mentioned paths in Step 1 and paste them into the Settings tab as described above.

SPI_MASTER_0

General Settings | Advanced Settings | Interrupt Settings | Pin Settings

Operation mode: Full Duplex

Desired bus speed [KHz]: 1000

Actual bus speed [KHz]: 1000

Parity selection: None

SPI_MASTER_0

General Settings | Advanced Settings | Interrupt Settings | Pin Settings

Protocol Handling

Transmit mode: Interrupt

Receive mode: Interrupt

Number of slave select lines: 1

Frame and Word Length

☒ Enable frame end mode

Word length: 8

Frame length: 64

Delay Settings

Leading/Trailing delay [SCLK cycles]: 2

Actual Leading/Trailing delay [usec]: 2

Desired inter-word delay [SCLK cycles]: 0

Actual inter-word delay [usec]: 0.0

Control

Bit order: Transmit/receive MSB first

Clock settings: Low if inactive, transmit on falling clock edge, receive on rising clock edge

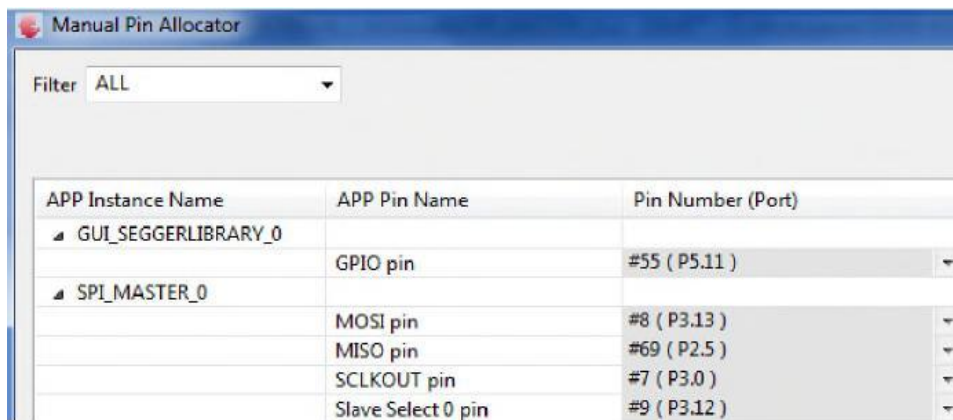
FIFO Settings

☒ Enable transmit FIFO Size: 16

☒ Enable receive FIFO Size: 16

A total of 64 FIFO entries are available to be configured as transmit and receive buffers. These 64 entries are additionally shared between the two channels of a USIC module.

3. Use the "Manual Pin Allocator" found in the toolbar, configure the pins for GUI_SEGGERLIBRARY and SPI_MASTER APPs as shown in below snapshot:



4. Press Solve and Save button.
5. Generate the code for configurations made.
6. Build and download to the microcontroller.

HOW TO USE THE GUI_SEGGERLIBRARY APP IN THE PROJECT:

- Refer to emWin documentation for API definition.

HOW TO TEST:

Download and run the demo in the uC.

OBSERVATIONS:

First screen: "Infineon" (image)

Second screen: "proudly presents..."

Third screen: "XMC" (image) will be displayed on the OLED.