RX Family

QE for Display [RX] Sample Program

Summary

This application note describes a sample application which interoperates with QE for Display [RX], a plugin for the e^2 studio integrated development environment with support for suitable Renesas RX microcontrollers. QE for Display [RX] provides a graphical interface for display control to support the development of embedded systems incorporating display devices. To develop a system using QE for Display [RX], a program is required to initialize the graphics LCD controller (GLCDC) which is in the RX family product. This application note provides a sample program which can be used as a basis for programs that are required to initialize the GLCDC.

Target Devices

- RX65N and RX651 groups (ROM capacity: 1.5 MB to 2 MB)
- RX72N groups
- RX72M groups
- RX66N groups

This application note deal with following sample programs.

- Renesas Starter Kit+ for RX72N
- Renesas Envision KIT RPBRX72N
- Renesas Starter Kit+ for RX65N-2MB
- Renesas Envision KIT RPBRX65N

When you apply this application note to another microcontroller, refer to chapter 10, Adapting the Sample Program to the User Environment.

Contents

1. Overview

As shown in Figure 1-1, multiple blocks make up the GLCDC, so simply checking the display attributes requires an understanding of the GLCDC specifications and several settings. However, by using this sample program and QE for Display [RX], it is possible to prepare an environment in which display device connections can be checked quickly, without the need to understand the GLCDC specifications. QE for Display [RX] is a tool that provides a graphical interface for display control. The user enters information on the display device to be used, and the tool produces a header file containing the information necessary for display control. Using the header file as a basis, the sample program makes settings for the GLCDC.

The tool also provides a facility for adjusting the timing in real time, making it possible to first make fine adjustments with the display device connected, and then to output the header file.

In addition, displays can be controlled more simply by using the Smart Configurator which graphically supports the settings of the device and display-related Firmware Integration Technology (FIT), which provides drivers and middleware for the RX family, and by adopting QE for Display [RX]. This sample program mainly uses QE for Display [RX] and the graphics LCD controller module Firmware Integration Technology (GLCDC FIT module) provided by the Smart Configurator and FIT.

The sample program and its requirements are described from the next section.

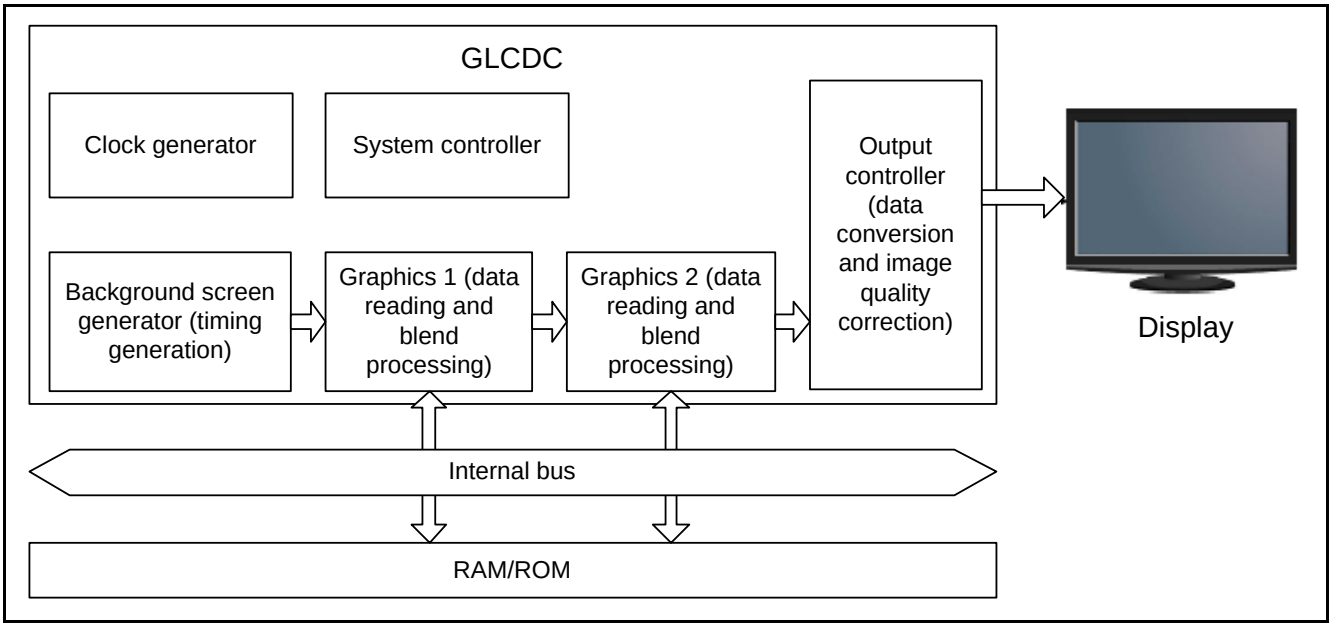


Figure 1-1 Block Configuration of the GLCDC

1.1 Flow of System Development with QE for Display [RX]

Figure 1-2 shows a flow of system development with the use of QE for Display [RX].

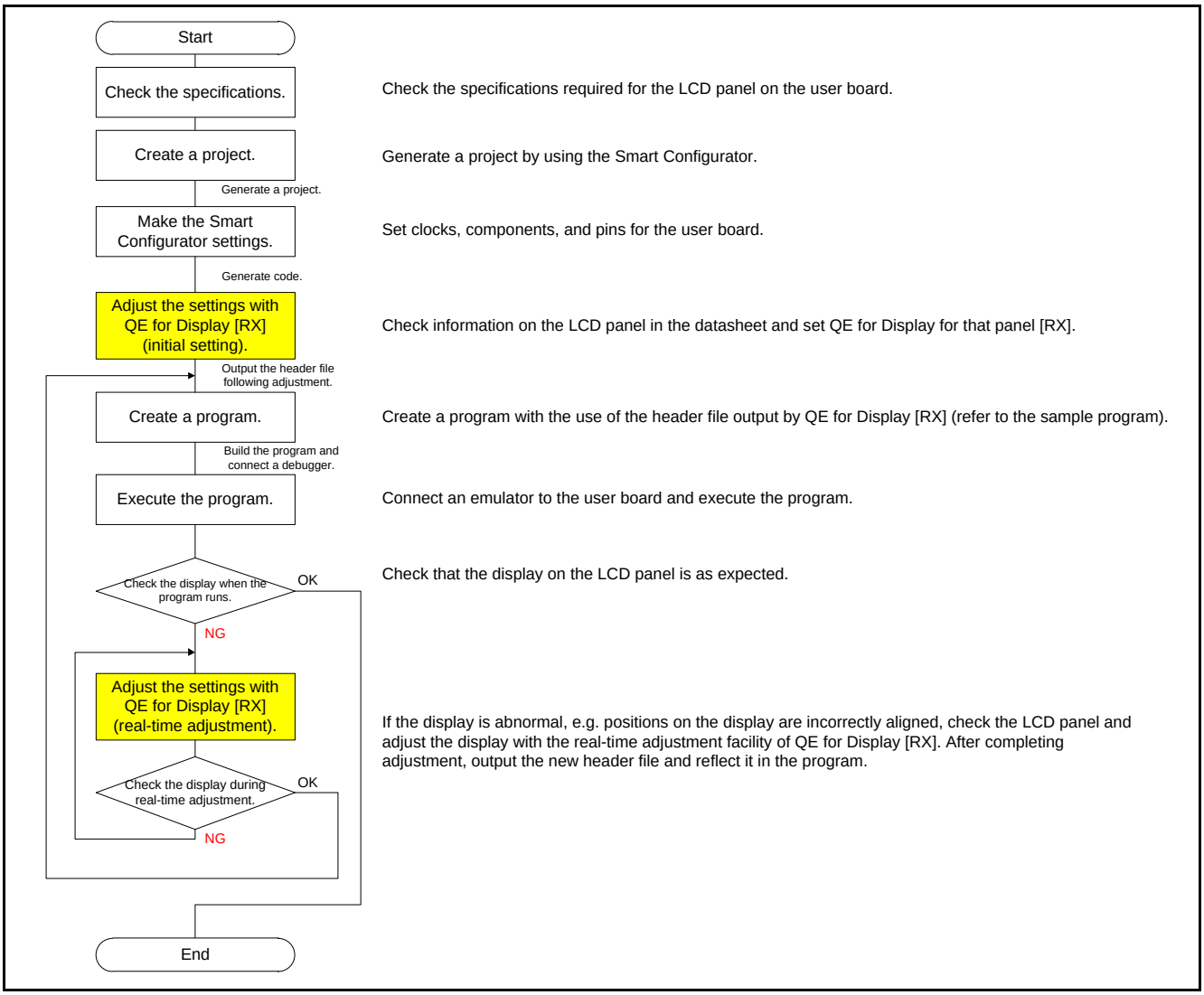


Figure 1-2 System Development by Using QE for Display [RX]

2. Operating Environment

This sample program has been run and confirmed with the Renesas Starter Kit+ for RX72N (RSK RX72N), the Renesas Envision Kit RPBRX72N (Envision RX72N), the Renesas Starter Kit+ for RX65N-2MB (RSK RX65N) and the Renesas Envision Kit RPBRX65N (Envision RX65N). Table 2- to Table2-4 list conditions for confirming operations for each board, respectively.

Table 2-1 Conditions for Confirming Operation (RSK RX72N)

Item	Contents
MCU used	R5F572NNDDBD (RX72N Group)
Operating frequency	• Main clock: 24 MHz • PLL: 240 MHz (main clock x 1/1 x 10) • System clock (ICLK): 240 MHz (PLL x 1/1) • Peripheral module clock A (PCLKA): 120 MHz (PLL x 1/2) • Peripheral module clock B (PCLKB): 60 MHz (PLL x 1/4) • LCD panel clock (LCD_CLK): 10 MHz (PLL x 1/24)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics e^2 studio Version 7.7.0
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V.3.01.00
	Compiler option -lang = C99
Version of iodefine.h	Version 1.0
Endian	Little endian, big endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample program version	Version 1.10
Emulator	E2 Lite
Board used	Renesas Starter Kit+ for RX72N (product No.: RTK5572NNXXXXXXXXX)

Table 2-2 Conditions for Confirming Operation (Envision RX72N)

Item	Contents
MCU used	R5F572NNHDFB (RX72N Group)
Operating frequency	• Main clock: 16 MHz • PLL: 240 MHz (main clock x 1/1 x 15) • System clock (ICLK): 240 MHz (PLL x 1/1) • Peripheral module clock A (PCLKA): 120 MHz (PLL x 1/2) • Peripheral module clock B (PCLKB): 60 MHz (PLL x 1/4) • LCD panel clock (LCD_CLK): 10 MHz (PLL x 1/24)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics e^2 studio Version 7.7.0
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V.3.01.00
	Compiler option -lang = C99
Version of iodefine.h	Version 1.0
Endian	Little endian, big endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample program version	Version 1.10
Emulator	E2 OB (E2 emulator On Board)
Board used	Renesas Envision KIT RPBRX72N (product No.: RTK5RX72N0CXXXXXBJ)
Board settings (jumper/switch)	<SW>1 Pin 1: don't care Pin 2: OFF (The debugger is used.) <Others> Default settings

Table 2-3 Conditions for Confirming Operation (RSK RX65N)

Item	Contents
MCU used	R5F565NEDDFC (RX65N Group)
Operating frequency	• Main clock: 24 MHz • PLL: 240 MHz (main clock x 1/1 x 10) • System clock (ICLK): 120 MHz (PLL x 1/2) • Peripheral module clock A (PCLKA): 120 MHz (PLL x 1/2) • Peripheral module clock B (PCLKB): 60 MHz (PLL x 1/4) • LCD panel clock (LCD_CLK): 10 MHz (PLL x 1/24)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics e^2 studio Version 7.7.0
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V.3.01.00
	Compiler option -lang = C99
Version of iodefine.h	Version 2.3
Endian	Little endian, big endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample program version	Version 1.10
Emulator	E2 Lite
Board used	Renesas Starter Kit+ for RX65N-2MB (product No.: RTK50565NXXXXXXXXX)
Board settings (jumper/switch)	<SW4> Pin 3: OFF Pin 4: ON (The LCD is used.) <Others> Default settings

Table 2-4 Conditions for Confirming Operation (Envision RX65N)

Item	Contents
MCU used	R5F565NEDDFB (RX65N Group)
Operating frequency	• Main clock: 12 MHz • PLL: 240 MHz (main clock x 1/1 x 20) • System clock (ICLK): 120 MHz (PLL x 1/2) • Peripheral module clock A (PCLKA): 120 MHz (PLL x 1/2) • Peripheral module clock B (PCLKB): 60 MHz (PLL x 1/4) • LCD panel clock (LCD_CLK): 10 MHz (PLL x 1/24)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics e^2 studio Version 7.7.0
C compiler	Renesas Electronics C/C++ Compiler Package for RX Family V.3.01.00
	Compiler Option -lang = C99
Version of iodefine.h	Version 2.3
Endian	Little endian, big endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample program version	Version 1.10
Emulator	E2 OB (E2 emulator On Board)
Board used	Renesas Envision KIT RPBRX65N (product No.: RTK5RX65N2CXXXXXBR)
Board settings (jumper/switch)	<SW>1 Pin 1: ON Pin 2: OFF (The debugger is used.) <SW4> Pin 1: OFF Pin 2: don't care (The debugger is used.) <Others> Default settings

3. Related Documents

Also refer to the following documents which are related to this sample program.

Firmware Integration Technology User's Manual (R01AN1833)
RX Family Board Support Package Firmware Integration Technology Module (R01AN1685)
RX Family Graphic LCD Controller Module Using Firmware Integration Technology (R01AN3609)
Renesas e^2 studio Smart Configurator User Guide (R20AN0451)
RX65N group Renesas Starter Kit+ for RX65N-2MB users' manual (R20UT3888)
RX65N Group RX65N Envision Kit User's Manual (R01UH0761)
RX65N Group, RX651 Group User's Manual: Hardware (R01UH0590)
RX72N Group Renesas Starter Kit+ for RX72N user’s manual (R20UT4436)
RX72N Group RX72N Envision Kit User's Manual (R20UT4788)
RX72N Group User’s Manual: Hardware (R01UH0824)

Please use the latest version when it is available. Visit the Renesas Electronics website to check and obtain the latest version.

4. Configuration of the Sample Project

The configuration of the sample project is shown below. However, the details of the FIT module and files that are automatically generated in the integrated development environment are excluded.

The sample project is provided for each board. All sample projects have same configuration, but header files output by QE for Display [RX] and files dependent on microcontroller are different.
Figure 4-1 shows the structure of RSK RX72N project folders in detail.

Table 4-1 Sample Projects

Project Name	Overview
QE_for_Display_sample_RX72N_RSK	Project operated on RSK RX72N
QE_for_Display_sample_RX72N_Envision	Project operated on Envision RX72N
QE_for_Display_sample_RX65N_RSK	Project operated on RSK RX65N
QE_for_Display_sample_RX65N_Envision	Project operated on Envision RX65N

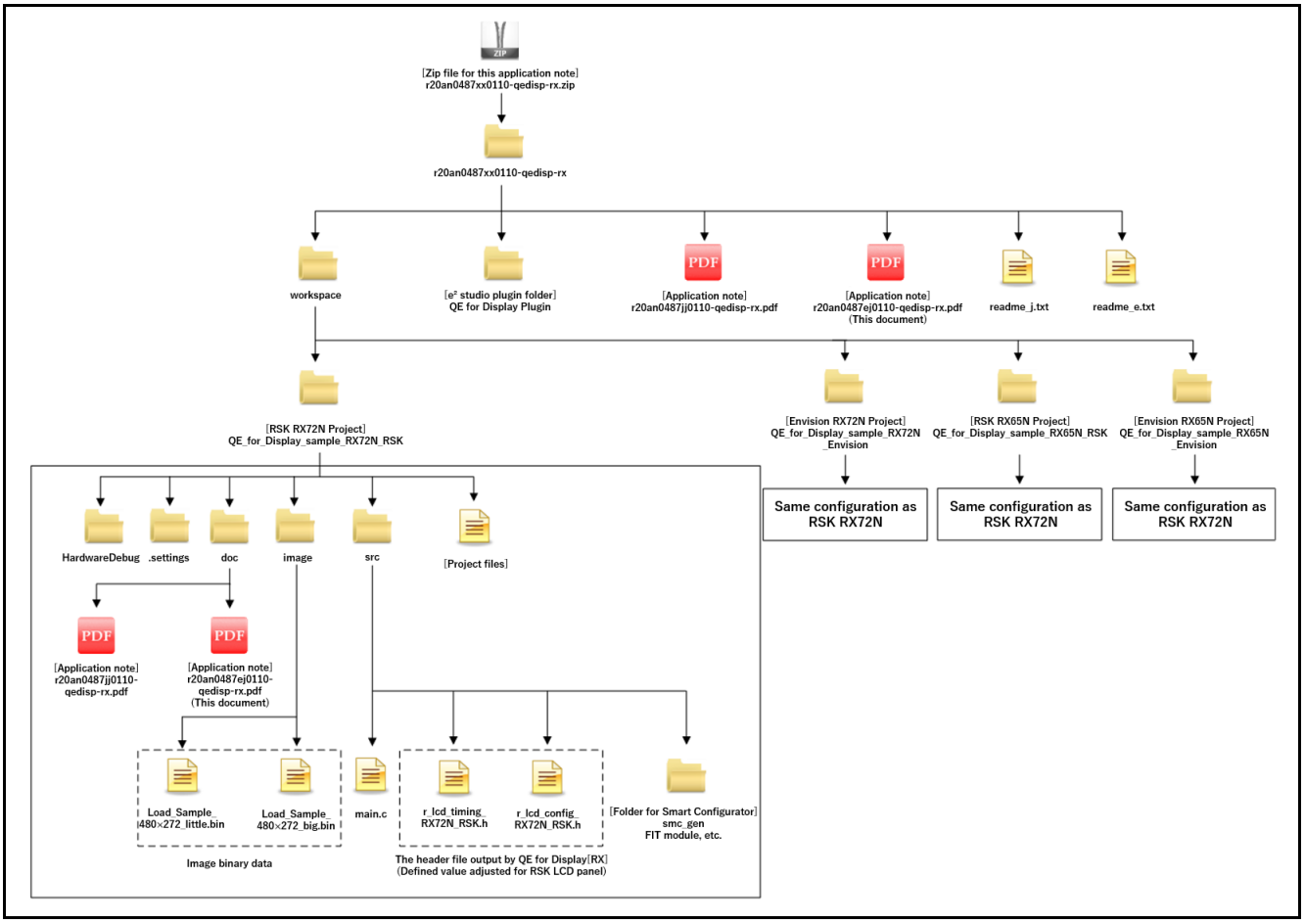


Figure 4-1 Structure of Project Folders

5. Procedure for Executing the Sample Project

This chapter describes the procedure for executing the sample project before real-time adjustment with the use of QE for Display [RX]. To use the real-time adjustment facility of QE for Display [RX], the GLCDC must have been initialized beforehand. The GLCDC is initialized by the user program.

In this sample program, the GLCDC is initialized by the glcdc_initialize function. After the glcdc_initialize function has been called, start real-time adjustment with QE for Display [RX]. For details on the sample program, refer to chapter 7, Software.

Before starting this project, if the jumper settings described in chapter 2, Operating Environment, be sure to make them.

For the usage of QE for Display [RX], refer to chapter 8, Using QE for Display [RX].

Preparation

1. Installing QE for Display [RX]

Procedure

2. Importing a project
3. Building a project
4. Connecting a debugger and executing the program
5. Real-time adjustment with QE for Display [RX]

5.1 Installing QE for Display [RX]

Install QE for Display [RX] in the integrated development environment e² studio. Use the following procedure to install this product.

Installing QE for Display [RX]

1. Start the e² studio.
2. From the [Help] menu, select [Install New Software...] to open the [Install] dialog box.
3. Click the [Add...] button to open the [Add Repository] dialog box.
4. Click the [Archive] button, select the zip file for installation (RenesasQE_display_package_V130.zip) in the opened dialog box, and click the [Open] button.
5. Click the [OK] button in the [Add Repository] dialog box.
6. Uncheck the [Contact all update site during install to find required software] check box.
7. Select the [Renesas QE for Display[RX]] check box displayed in the [Install] dialog box and click the [Next] button.
8. Check that [Renesas QE for Display[RX]] is selected as the target of installation, and click the [Next] button.
9. After confirming the license agreements, select the [I accept the terms of the license agreements] radio button, and click the [Finish] button.
10. If the dialog box of the trust certificate is displayed, check that certificate, and click the [OK] button to continue installation.
11. When prompted to restart the e² studio, restart it.

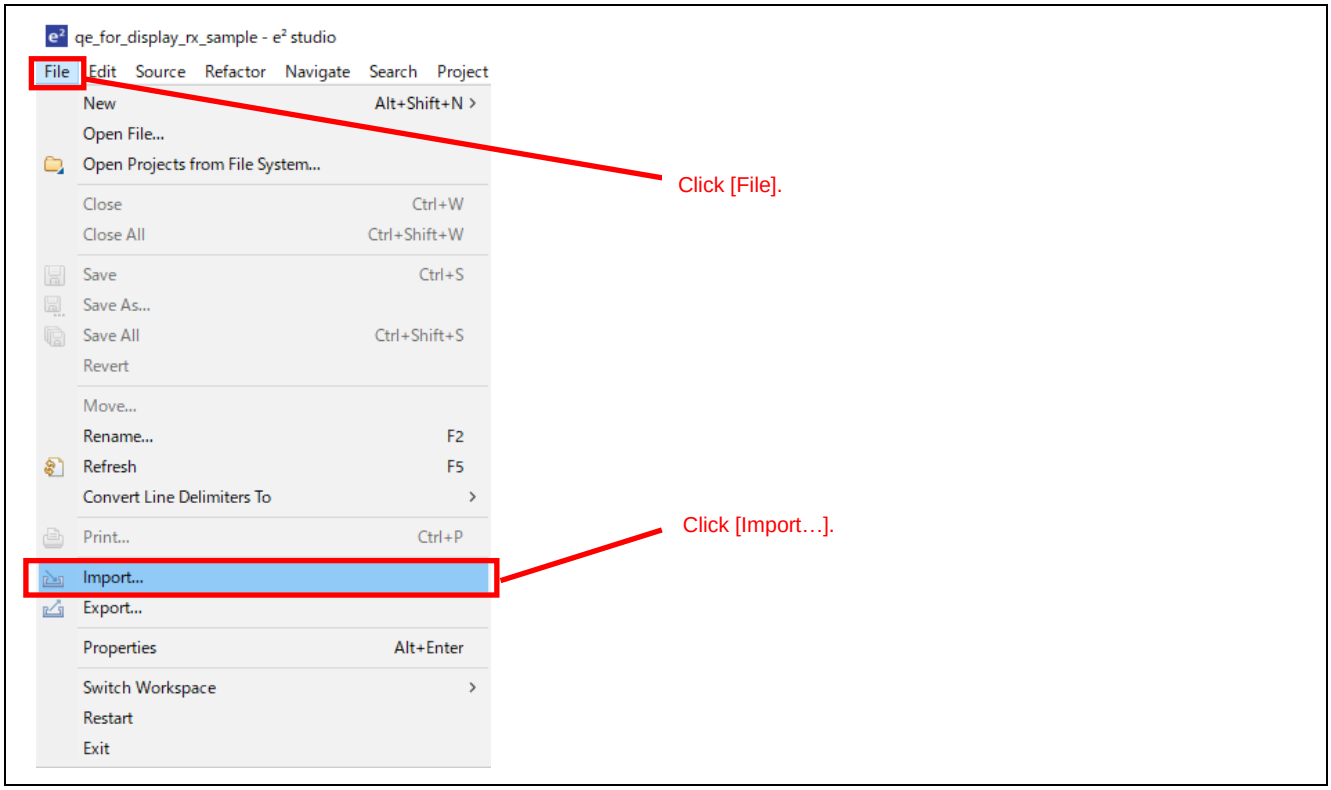
To uninstall QE for Display [RX], follow the procedure below.

Uninstalling QE for Display [RX]

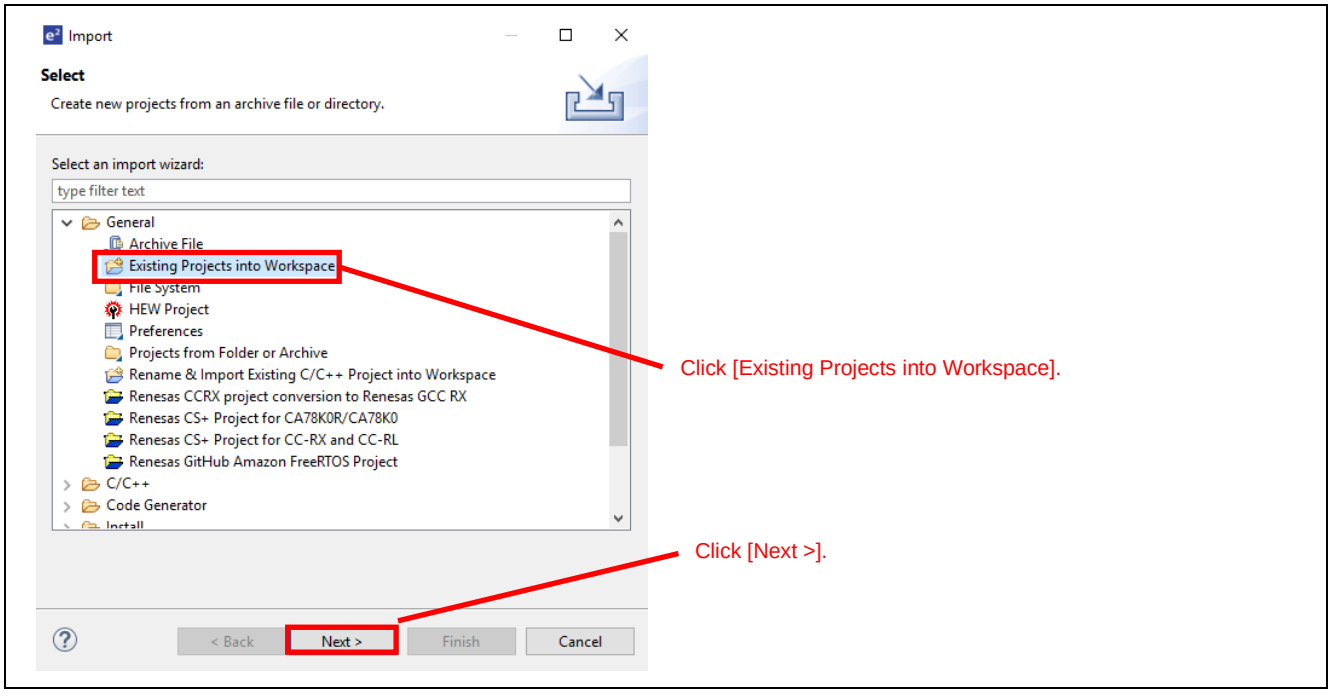
1. Start the e² studio.
2. From the [Help] → [About e² studio] menu, select [Installation Details] to open the [e² studio Installation Details] dialog box.
3. Select [Renesas QE for Display[RX]] displayed on the [Installed Software] tabbed page and click the [Uninstall...] button to open the [Uninstall] dialog box.
4. Check the displayed information and click the [Finish] button.
5. When prompted to restart the e² studio, restart it.

5.2 Importing the Project

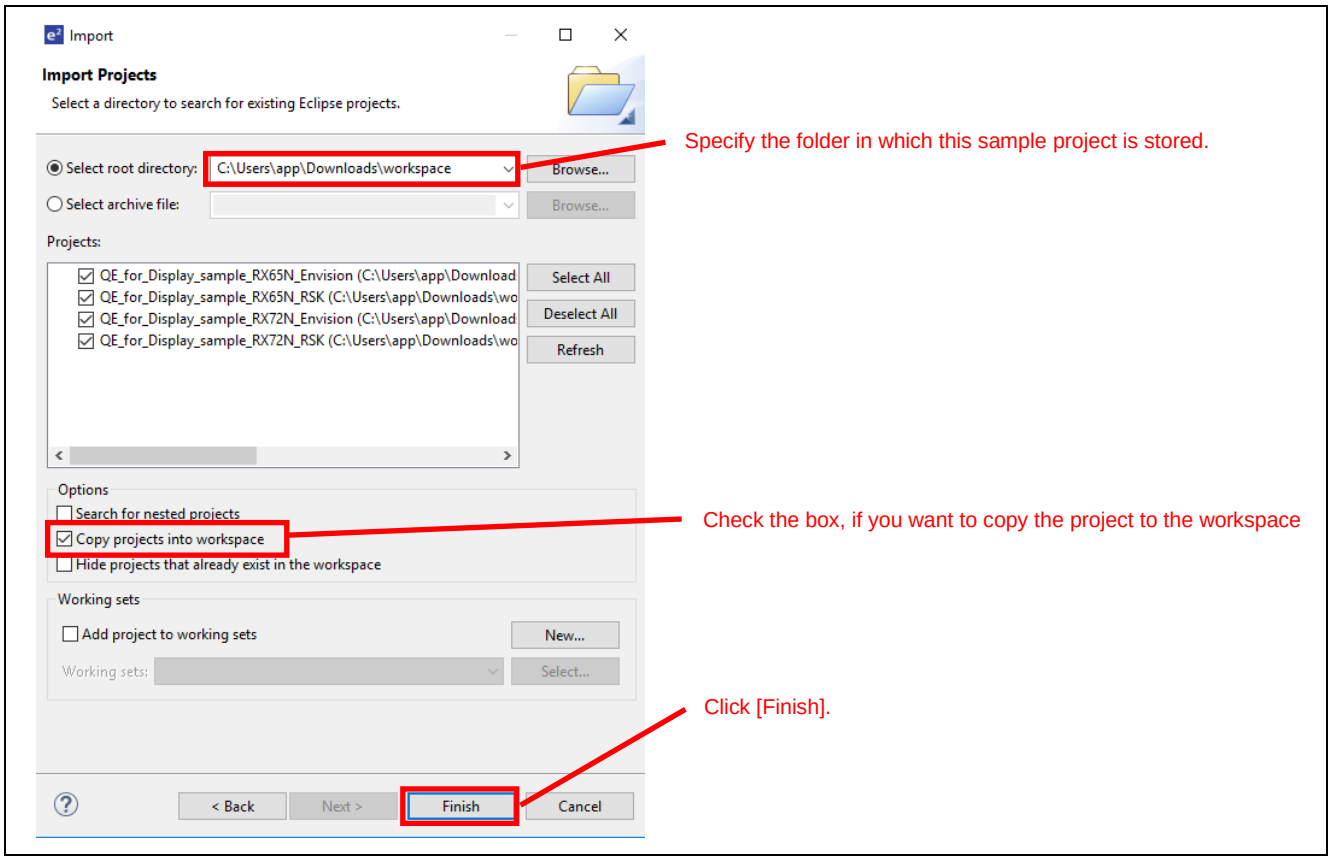
1. Click [File].
2. Click [Import...].



3. Click [General] > [Existing Projects into Workspace].
4. Click [Next >].



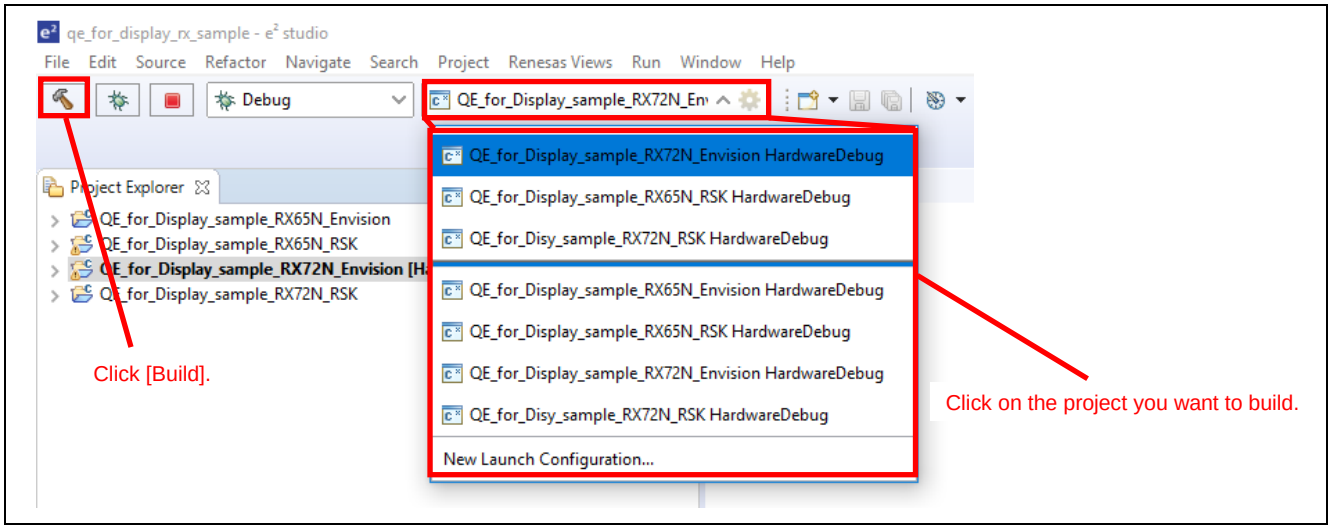
5. Specify the folder which has this sample project into the combo box in the [Select root directory:].
6. Click [Finish].



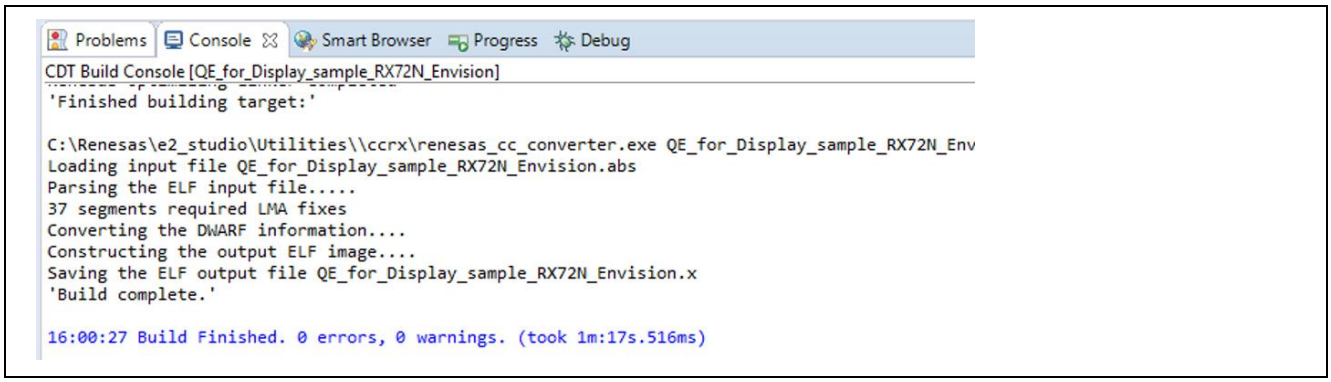
5.3 Building the Project

Build the project and make the load module according to the following procedure.

1. Click on the project you want to build (e.g. QE_for_Display_sample_RX72N_Envision HardwareDebug).
2. Click [Build].

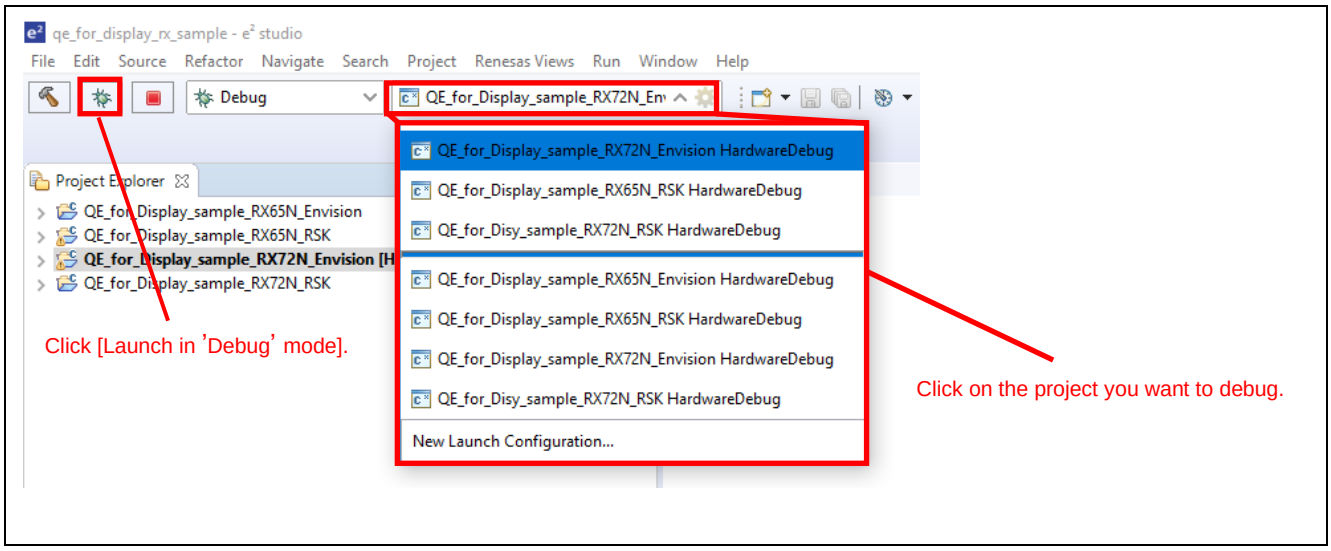


3. When the 'Console' panel displays 'Build complete.', the build operation is complete.

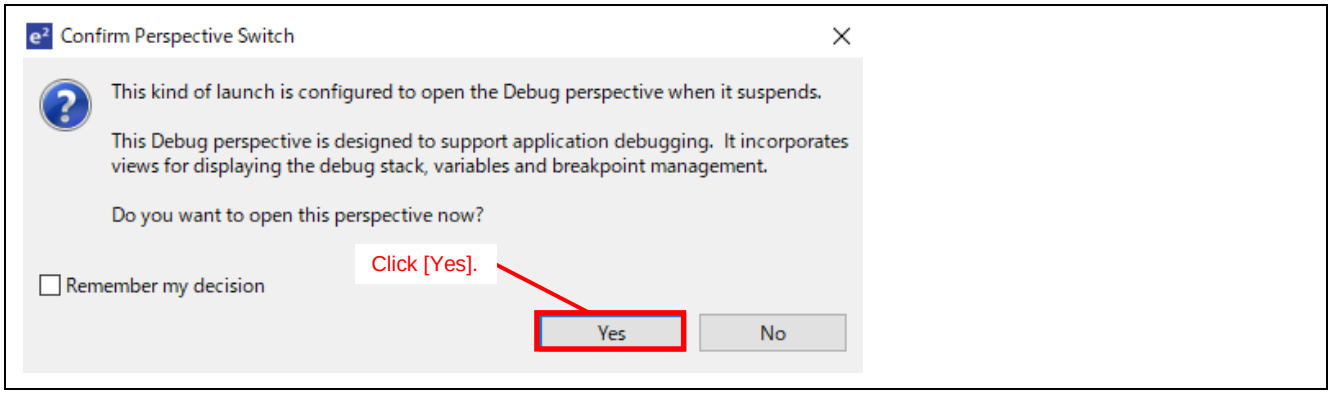


5.4 Connecting a Debugger and Executing the Program

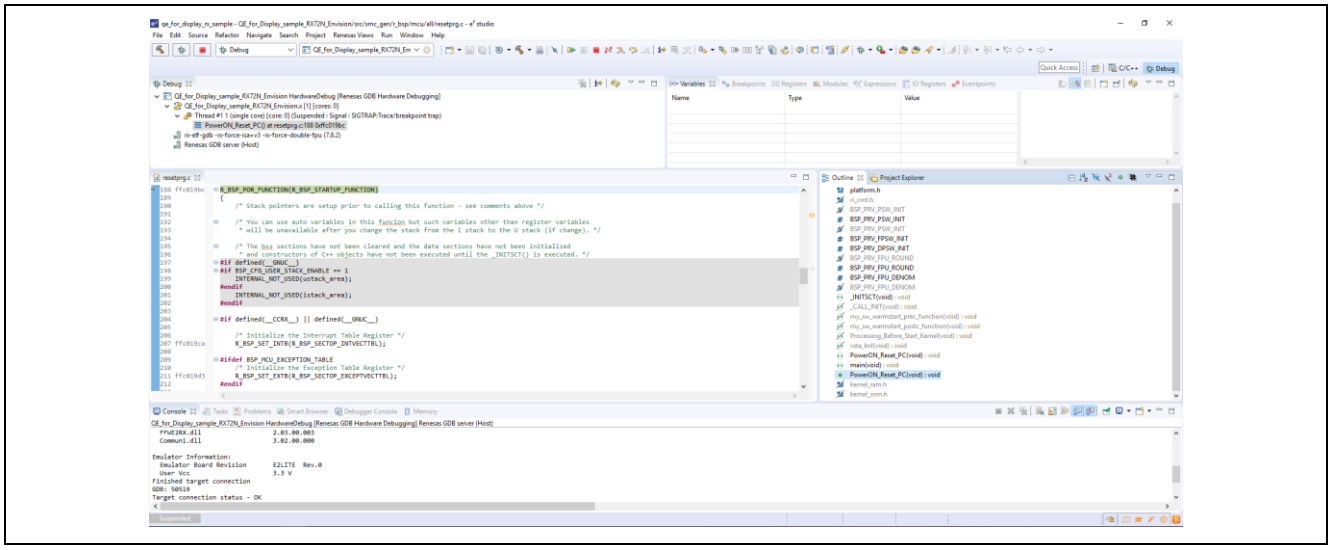
1. Click on the project you want to debug (e.g. QE_for_Display_sample_RX72N_Envision HardwareDebug).
2. Click [Launch in 'Debug' mode].



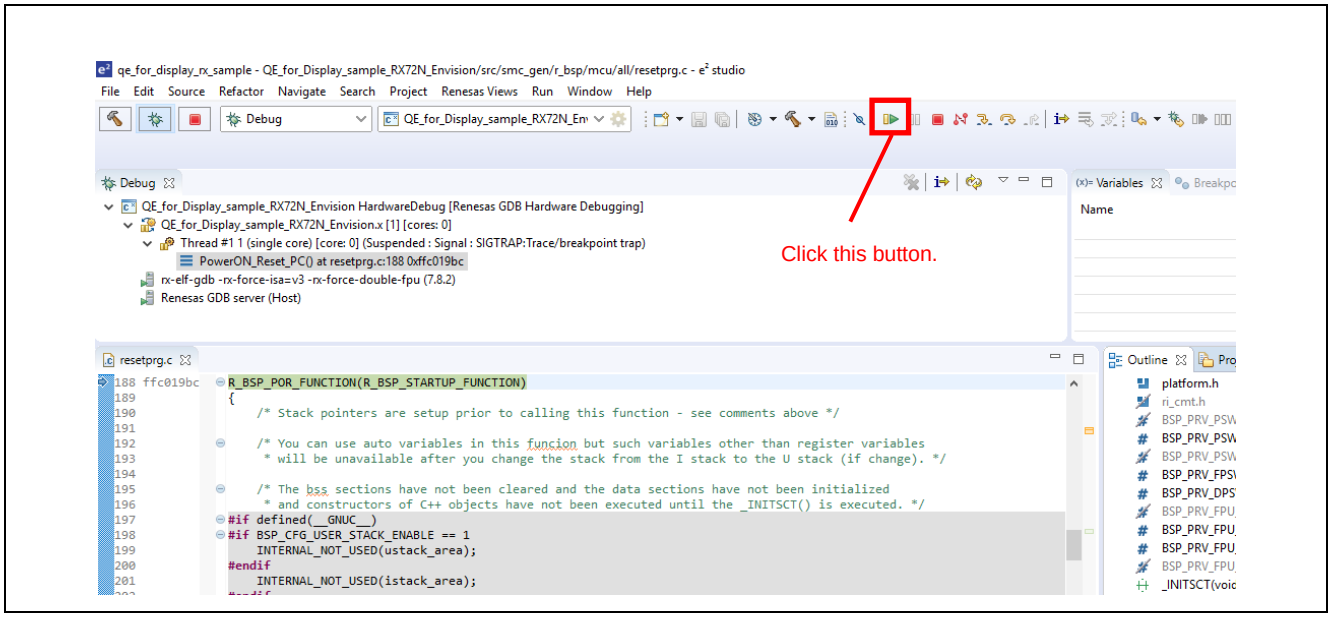
3. When the following message is displayed, click [Yes].



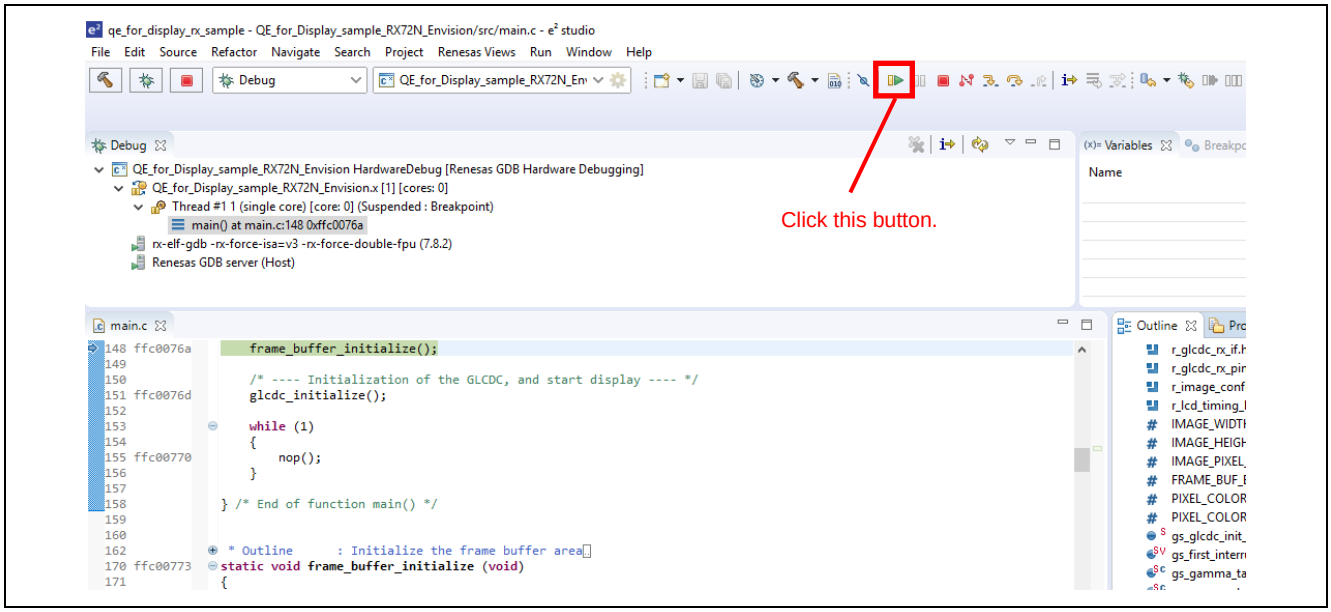
4. When downloading of the load module is completed, the [Debug] perspective opens.



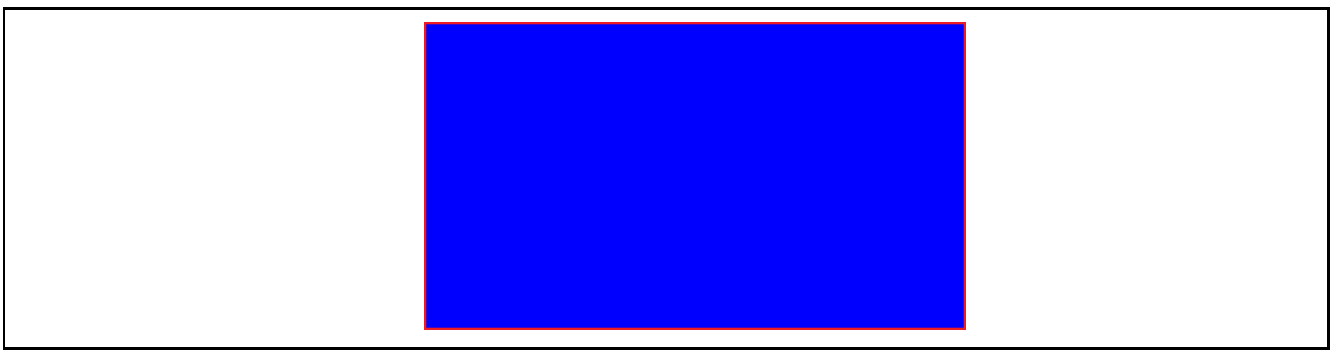
5. Click [Resume] on the tool bar. The program is executed and breaks at the beginning of the main function.



6. After a break occurs at the beginning of the main function, click [Resume] again on the toolbar.

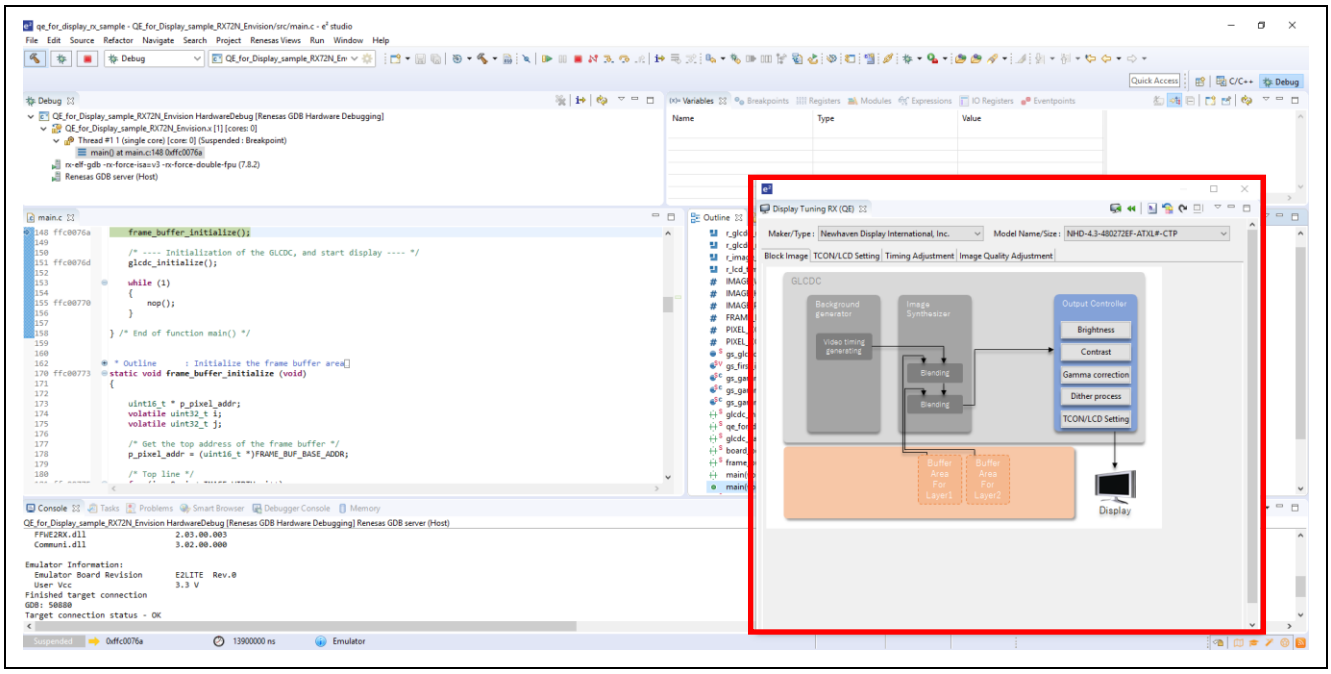


7. When the setting of the display device is done correctly, the following screen will be displayed on the LCD panel.



5.5 Real-time Adjustment with QE for Display [RX]

1. When the screen is displayed on the LCD panel, launch QE for Display [RX] and start real-time adjustment.



6. Hardware

6.1 Configuration of Hardware

Table 6-1 shows the LCD panel used in this sample.

Table 6-1 LCD Panel Used in the Sample.

Board	Information on the LCD Panel Product
RSK RX72N RSK RX65N	Manufacturer: Newhaven Display Co. Part number: NHD-4.3-480272EF-ATXL#-CTP Display size:480 x 272 Synchronization signal: VS, HS, DE (three signals) Built-in touch controller (not used in this sample)
Envision RX72N Envision RX65N	Manufacturer: EastRising Co. Part number: ER-TFT043-3 Display size:480 x 272 Synchronization signal: VS, HS, DE (three signals) Built-in touch controller (not used in this sample)

6.2 Pin Functions

The following shows pins used on each RSK and Envision and describes the pin functions used. Select the pins according to the product you are using. Pin functions can be set by using the Smart Configurator.

Table 6-2 Pins and Functions to be Used (RSK RX72N)

Device Connected	Pin Name	Input/Output	Description
NHD-4.3-480272EF-ATXL#-CTP	P14/LCD_CLK-B	Output	Outputs the panel clock.
	P13/LCD_TCON 0-B	Output	Outputs the synchronization signal (VSYNC).
	PJ2/LCD_TCON 2-B	Output	Outputs the synchronization signal (HSYNC).
	PB1/LCD_TCON 3-B	Output	Outputs the synchronization signal (DE).
	PC5/LCD_DATA 0-B	Output	Outputs the LCD signal R[3].
	P82/LCD_DATA 1-B	Output	Outputs the LCD signal R[4].
	P81/LCD_DATA 2-B	Output	Outputs the LCD signal R[5].
	P80/LCD_DATA 3-B	Output	Outputs the LCD signal R[6].
	PC4/LCD_DATA 4-B	Output	Outputs the LCD signal R[7].
	P55/LCD_DATA 5-B	Output	Outputs the LCD signal G[2].
	P54/LCD_DATA 6-B	Output	Outputs the LCD signal G[3].
	P11/LCD_DATA 7-B	Output	Outputs the LCD signal G[4].
	P83/LCD_DATA 8-B	Output	Outputs the LCD signal G[5].
	PC7/LCD_DATA 9-B	Output	Outputs the LCD signal G[6].
	PC6/LCD_DATA 10-B	Output	Outputs the LCD signal G[7].
	PJ0/LCD_DATA 11-B	Output	Outputs the LCD signal B[3].
	P85/LCD_DATA 12-B	Output	Outputs the LCD signal B[4].
	P84/LCD_DATA 13-B	Output	Outputs the LCD signal B[5].
	P57/LCD_DATA 14-B	Output	Outputs the LCD signal B[6].
	P56/LCD_DATA 15-B	Output	Outputs the LCD signal B[7].
	P27/general-purpose input/output port	Output	Backlight (controlled by the program)
	PK4/general-purpose input/output port	Output	Panel reset (controlled by the program)

Table 6-3 Pins and Functions to be Used (Envision RX72N)

Device Connected	Pin Name	Input/Output	Description
ER-TFT043-3	PB5/LCD_CLK-B	Output	Outputs the panel clock.
	PB4/LCD_TCON 0-B	Output	Outputs the synchronization signal (VSYNC).
	PB2/LCD_TCON 2-B	Output	Outputs the synchronization signal (HSYNC).
	PB1/LCD_TCON 3-B	Output	Outputs the synchronization signal (DE).
	PB0/LCD_DATA 0-B	Output	Outputs the LCD signal B[3].
	PA7/LCD_DATA 1-B	Output	Outputs the LCD signal B[4].
	PA6/LCD_DATA 2-B	Output	Outputs the LCD signal B[5].
	PA5/LCD_DATA 3-B	Output	Outputs the LCD signal B[6].
	PA4/LCD_DATA 4-B	Output	Outputs the LCD signal B[7].
	PA3/LCD_DATA 5-B	Output	Outputs the LCD signal G[2].
	PA2/LCD_DATA 6-B	Output	Outputs the LCD signal G[3].
	PA1/LCD_DATA 7-B	Output	Outputs the LCD signal G[4].
	PA0/LCD_DATA 8-B	Output	Outputs the LCD signal G[5].
	PE7/LCD_DATA 9-B	Output	Outputs the LCD signal G[6].
	PE6/LCD_DATA 10-B	Output	Outputs the LCD signal G[7].
	PE5/LCD_DATA 11-B	Output	Outputs the LCD signal R[3].
	PE4/LCD_DATA 12-B	Output	Outputs the LCD signal R[4].
	PE3/LCD_DATA 13-B	Output	Outputs the LCD signal R[5].
	PE2/LCD_DATA 14-B	Output	Outputs the LCD signal R[6].
	PE1/LCD_DATA 15-B	Output	Outputs the LCD signal R[7].
	P67/general-purpose input/output port	Output	Backlight (controlled by the program)
	PB3/general-purpose input/output port	Output	Panel reset (controlled by the program)

Table 6-4 Pins and Functions to be Used (RSK RX65N)

Device Connected	Pin Name	Input/Output	Description
NHD-4.3-480272EF-ATXL#-CTP	PB5/LCD_CLK-B	Output	Outputs the panel clock.
	PB4/LCD_TCON 0-B	Output	Outputs the synchronization signal (VSYNC).
	PB2/LCD_TCON 2-B	Output	Outputs the synchronization signal (HSYNC).
	PB1/LCD_TCON 3-B	Output	Outputs the synchronization signal (DE).
	PB0/LCD_DATA 0-B	Output	Outputs the LCD signal R[3].
	PA7/LCD_DATA 1-B	Output	Outputs the LCD signal R[4].
	PA6/LCD_DATA 2-B	Output	Outputs the LCD signal R[5].
	PA5/LCD_DATA 3-B	Output	Outputs the LCD signal R[6].
	PA4/LCD_DATA 4-B	Output	Outputs the LCD signal R[7].
	PA3/LCD_DATA 5-B	Output	Outputs the LCD signal G[2].
	PA2/LCD_DATA 6-B	Output	Outputs the LCD signal G[3].
	PA1/LCD_DATA 7-B	Output	Outputs the LCD signal G[4].
	PA0/LCD_DATA 8-B	Output	Outputs the LCD signal G[5].
	PE7/LCD_DATA 9-B	Output	Outputs the LCD signal G[6].
	PE6/LCD_DATA 10-B	Output	Outputs the LCD signal G[7].
	PE5/LCD_DATA 11-B	Output	Outputs the LCD signal B[3].
	PE4/LCD_DATA 12-B	Output	Outputs the LCD signal B[4].
	PE3/LCD_DATA 13-B	Output	Outputs the LCD signal B[5].
	PE2/LCD_DATA 14-B	Output	Outputs the LCD signal B[6].

Device Connected	Pin Name	Input/Output	Description
	PE1/LCD_DATA 15-B	Output	Outputs the LCD signal B[7].
	PB7/general-purpose input/output port	Output	Backlight (controlled by the program)
	P97/general-purpose input/output port	Output	Panel reset (controlled by the program)

Table 6-5 Pins and Functions to be Used (Envision RX65N)

Device Connected	Pin Name	Input/Output	Description
ER-TFT043-3	PB5/LCD_CLK-B	Output	Outputs the panel clock.
	PB4/LCD_TCON 0-B	Output	Outputs the synchronization signal (VSYNC).
	PB2/LCD_TCON 2-B	Output	Outputs the synchronization signal (HSYNC).
	PB1/LCD_TCON 3-B	Output	Outputs the synchronization signal (DE).
	PB0/LCD_DATA 0-B	Output	Outputs the LCD signal B[3].
	PA7/LCD_DATA 1-B	Output	Outputs the LCD signal B[4].
	PA6/LCD_DATA 2-B	Output	Outputs the LCD signal B[5].
	PA5/LCD_DATA 3-B	Output	Outputs the LCD signal B[6].
	PA4/LCD_DATA 4-B	Output	Outputs the LCD signal B[7].
	PA3/LCD_DATA 5-B	Output	Outputs the LCD signal G[2].
	PA2/LCD_DATA 6-B	Output	Outputs the LCD signal G[3].
	PA1/LCD_DATA 7-B	Output	Outputs the LCD signal G[4].
	PA0/LCD_DATA 8-B	Output	Outputs the LCD signal G[5].
	PE7/LCD_DATA 9-B	Output	Outputs the LCD signal G[6].
	PE6/LCD_DATA 10-B	Output	Outputs the LCD signal G[7].
	PE5/LCD_DATA 11-B	Output	Outputs the LCD signal R[3].
	PE4/LCD_DATA 12-B	Output	Outputs the LCD signal R[4].
	PE3/LCD_DATA 13-B	Output	Outputs the LCD signal R[5].
	PE2/LCD_DATA 14-B	Output	Outputs the LCD signal R[6].
	PE1/LCD_DATA 15-B	Output	Outputs the LCD signal R[7].
	P66/general-purpose input/output port	Output	Backlight (controlled by the program)
	P63/general-purpose input/output port	Output	Panel reset (controlled by the program)

7. Software

7.1 Overview of Operation

This sample program initializes clocks, interrupts, etc., on the CPU and makes settings of GLCDC operation based on the header file output by QE for Display [RX].

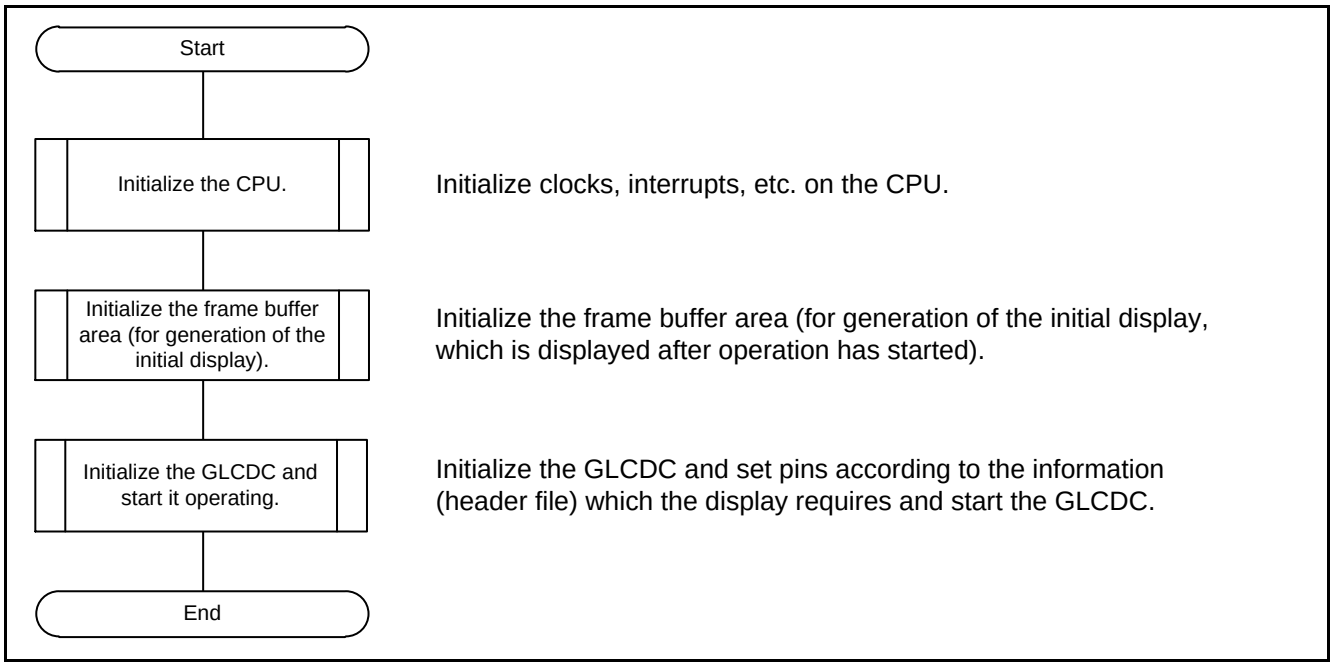


Figure 7-1 Overview of Operation of the Sample Program

7.2 Details of Settings for GLCDC Operation

Figure 7-2 shows the settings for GLCDC operation in detail. The GLCDC is set via the GLCDC FIT module. The member variables of the glcdc_cfg_t structure which is provided with the GLCDC FIT module become the arguments of the R_GLCDC_Open function, which initializes the GLCDC FIT module. The variables of that structure are set to multiple parameters to obtain the display on the screen.

Information regarding parameters is classified as follows.

a. Settings regarding synchronization signals and RGB signals output by the GLCDC → Edited by using QE for Display [RX].
b. Settings regarding output correction for the input data → Edited by using QE for Display [RX].
c. Settings regarding image data to be input to the GLCDC → The user directly edits the program.
d. Settings regarding interrupts → The user directly edits the program.

QE for Display [RX] mainly supports the setting of parameters a. and b., which depend on the specifications of the LCD panel. For settings c. and d., the user must directly edit the program according to the system and the format of the image to be used. That is, the user specifies values for the member variables of the structure corresponding to c. and d.

When the parameters have been set, the structure is passed to the initialization function (R_GLCDC_Open) of the GLCDC FIT module. After that, the pins are set.

Settings regarding pins of the GLCDC → Edited by using the Smart Configurator.
Settings regarding pins of the board → The user directly edits the program.

After pins have been set, display on the panel is started by the control function (R_GLCDC_Control) of the GLCDC FIT module.

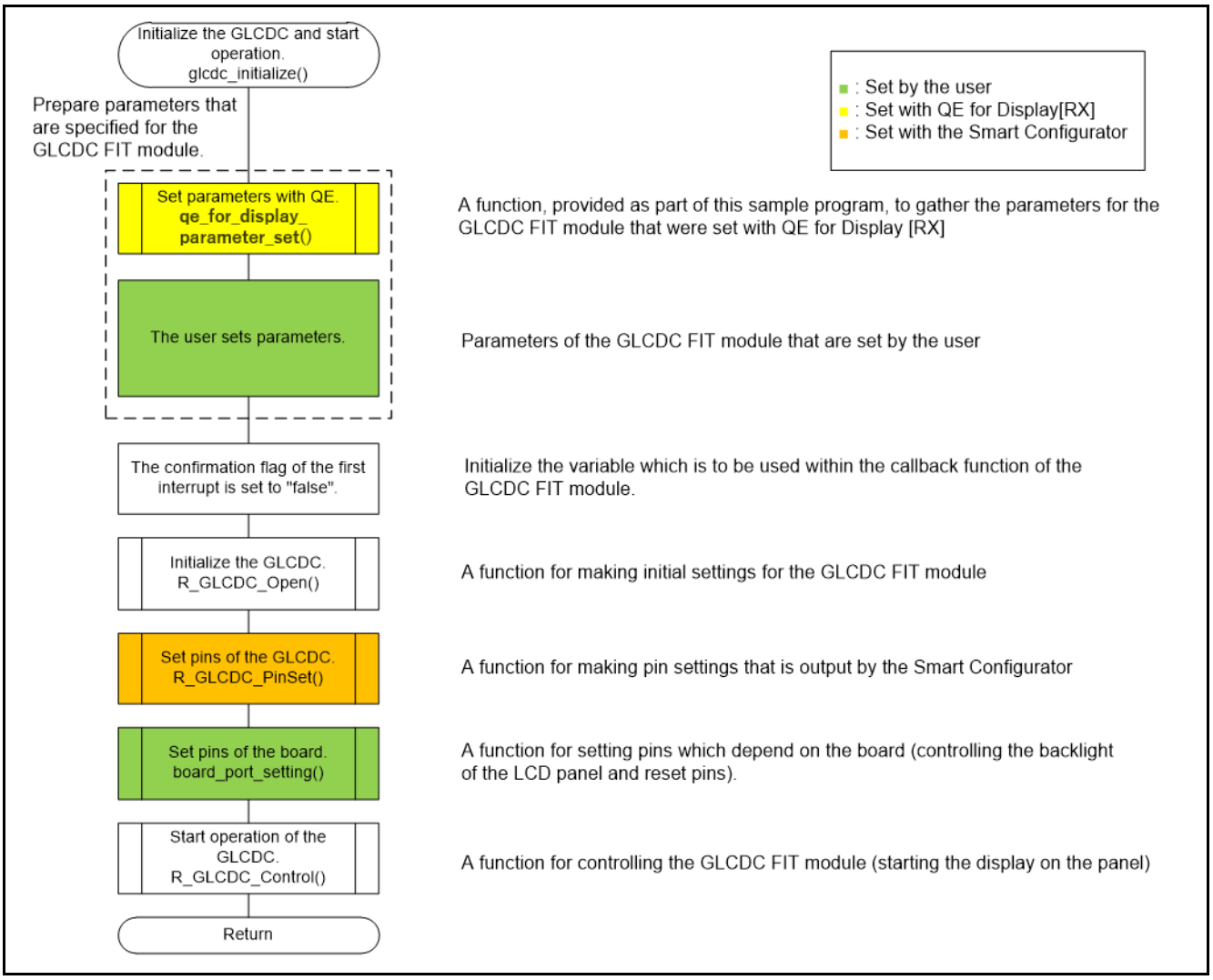


Figure 7-2 Details of Settings for GLCDC Operation

7.2.1 Parameter Settings Made by QE

The qe_for_display_parameter_set function to be called is provided as part of this sample program. The function collects parameters that use the define directives output by QE for Display [RX] among those set in the GLCDC FIT module.

7.2.2 Parameter Settings Made by the User

The user sets parameters that are not adjusted by QE for Display [RX] among those to be set in the GLCDC FIT module. The user must directly edit the program according to the system and the format of the image to be used. That is, the user specifies values for the member variables of the structure corresponding to c. and d. For setting values in this sample program, refer to section 7.3, Correspondence between Parameters in the GLCDC FIT Module and Header Files Output by QE for Display [RX].

7.2.3 Setting Pins of the GLCDC

As described in section 6.2, Pin Functions, pin functions can be set by using the Smart Configurator. The R_GLCDC_PinSet function to be called is implemented in the r_glcdc_rx_pinset.c file which has been generated by the Smart Configurator.

Figure 7-3 shows the setting of pins of the GLCDC by the Smart Configurator. Select whether or not each pin function is to be used and specify the port-pin numbers to which each pin function is to be assigned

according to the specifications and connections of the LCD panel. For the method of making pin settings, refer to the user's manual for the Smart Configurator.

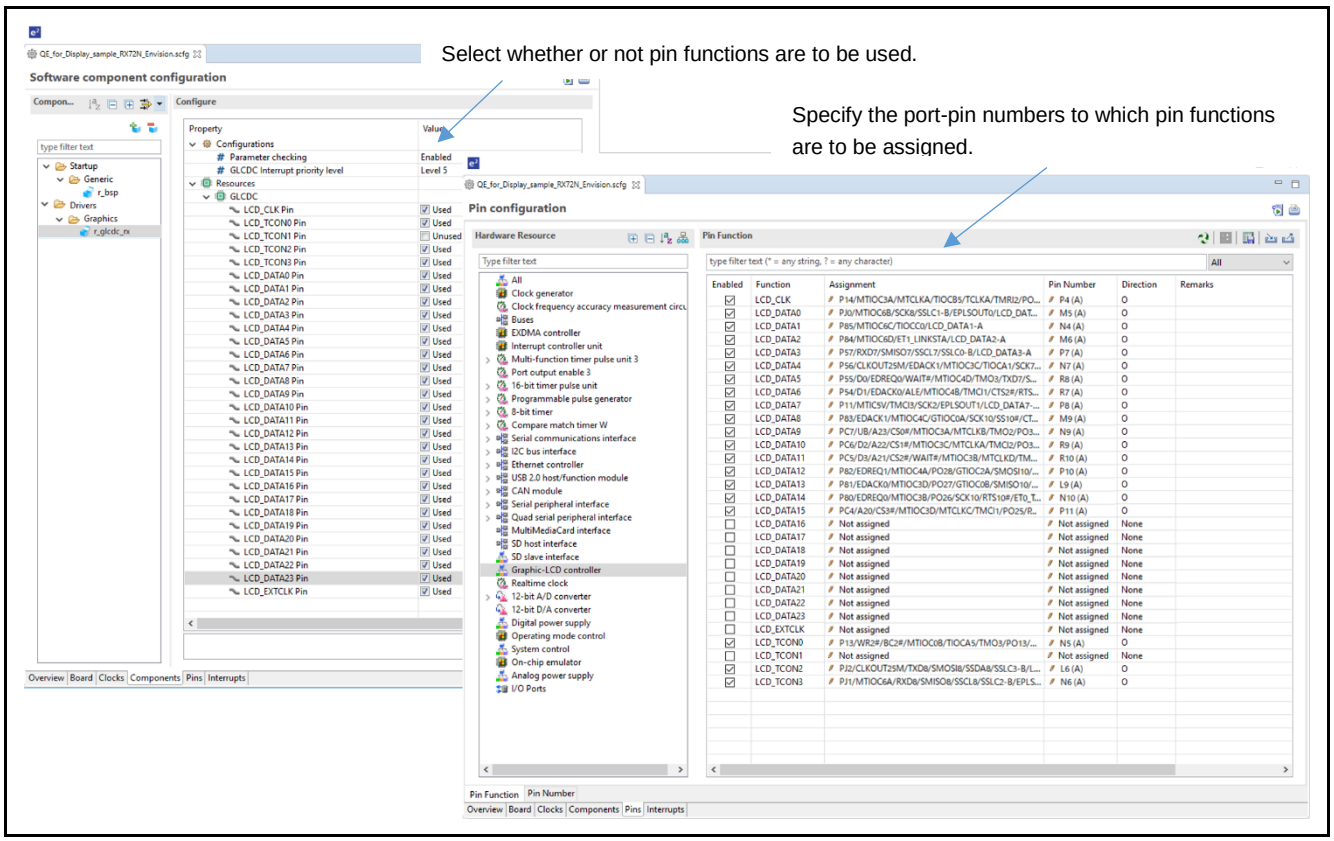


Figure 7-3 Windows for Setting Pins of the GLCDC (Smart Configurator)

7.2.4 Setting Pins on the Board

In addition to the pins controlled by the GLCDC, each board have connections to port pins that control the backlight and reset pins of the LCD panel.

The board_port_setting function that is to be called controls the backlight and reset pins of the LCD panel. The user must directly edit the program to set up control of these pins.

7.3 Correspondence between Parameters in the GLCDC FIT Module and Header Files Output by QE for Display [RX]

Table 7-1 lists the correspondences between parameters set in the GLCDC FIT module (members of the glcdc_cfg_t structure which is set as an argument of the R_GLCDC_Open function) and the define directives output by QE for Display [RX]. In the "Setting Value in the Sample Program" column, "←" means that the value is set with a define directive output by QE for Display [RX]. Entries other than "←" mean that the value is set by the user. For details of the parameters, refer to the user's manual for the GLCDC FIT module.

Table 7-1 Correspondence between Parameters in the GLCDC FIT Module and Definitions Output by QE for Display [RX]

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
Horizontal back porch	output.htiming. back_porch	LCD_CH0_W_HBP	←	Horizontal back porch period.
Horizontal assertion width	output.htiming. sync_width	LCD_CH0_W_HSYNC	←	Horizontal synchronization signal period.
Vertical back porch	output.vtiming. back_porch	LCD_CH0_W_VBP	←	Vertical back porch period.
Vertical assertion width	output.vtiming. sync_width	LCD_CH0_W_VSYNC	←	Vertical synchronization signal period.
Horizontal active display width	output.htiming. display_cyc	LCD_CH0_DISP_HW	←	Horizontal active display period.
Vertical active display width	output.vtiming. display_cyc	LCD_CH0_DISP_HW	←	Vertical active display period.
Horizontal front porch	output.htiming. front_porch	LCD_CH0_W_HFP	←	Horizontal front porch period.
Vertical front porch	output.vtiming. front_porch	LCD_CH0_W_VFP	←	Vertical front porch period.
Pointer to the callback function	p_callback	-	glcdc_callback	Executes the callback function at the address designated by the pointer when an interrupt source occurs.
Clock source	output.clksrc	-	GLCDC_CLK_SRC_INTERNAL	The PLL clock is used.
Clock division ratio	output. clock_div_ratio	-	GLCDC_PANEL_CLK_DIVISOR_24	Specifies the division ratio for LCD_CLK
Output data format	output.format	LCD_CH0_OUT_FORMAT	←	Output data format
Output phase control for TCON and DATA	output. sync_edge	LCD_CH0_OUT_EDGE	←	Outputs synchronizing with a rising or falling edge of LCD_CLK.
Output pin of the	output. tcon_hsync	LCD_CH0_TCON_PIN_HSYNC	←	Selects TCON which is used for the HSYNC output.

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
horizontal sync signal (HSYNC)				
Polarity of the horizontal sync signal (HSYNC)	output. hsync_polarity	LCD_CH0_TCON_POL_HSYNC	←	Sets polarity to low or high active.
Output pin of the vertical sync signal (VSYNC)	output. tcon_vsync	LCD_CH0_TCON_PIN_VSYNC	←	Selects TCON which is used for the VSYNC output.
Polarity of the vertical sync signal (VSYNC)	output. vsync_polarity	LCD_CH0_TCON_POL_VSYNC	←	Sets polarity to low or high active.
Output pin of the data enable signal (DE)	output. tcon_de	LCD_CH0_TCON_PIN_DE	←	Selects TCON which is used for the DE output.
Polarity of the data enable signal (DE)	output. data_enable_polarity	LCD_CH0_TCON_POL_DE	←	Sets polarity to low or high active.
R value for the background color	output.bg_color. byte.r	-	0xCC	Specifies the R value for the background color.
G value for the background color	output.bg_color. byte.g	-	0xCC	Specifies the G value for the background color.
B value for the background color	output.bg_color. byte.b	-	0xCC	Specifies the B value for the background color.
Graphics format of the frame buffer	input.format	-	Graphics 2: GLCDC_IN_FORMAT_16BITS_RGB565 Graphics 1: Not set	Selects the color format.
Start address of the frame buffer	input.p_base	-	Graphics 2: FRAME_BUF_BASE_ADDR(0x00800000) Graphics 1: NULL	Specifies the start address of the frame buffer. When NULL is set, the target graphics becomes disabled. (Setting values of structure members under glcdc_cfg_t.input are ignored.)

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
R value for the background color of graphics 1 and 2	input.bg_color.byte.r	-	Graphics 2: 0xCC Graphics 1: Not set	Specifies the R value for the background color of graphics 1 and 2.
G value for the background color of graphics 1 and 2	input.bg_color.byte.g	-	Graphics 2: 0xCC Graphics 1: Not set	Specifies the G value for the background color of graphics 1 and 2.
B value for the background color of graphics 1 and 2	input.bg_color.byte.b	-	Graphics 2: 0xCC Graphics 1: Not set	Specifies the B value for the background color of graphics 1 and 2.
Horizontal width of image data	input.hsize	-	Graphics 2: IMAGE_WIDTH(480) Graphics 1: Not set	Specifies the horizontal width of image for graphics 1 and 2.
Vertical width of image data	input.vsize	-	Graphics 2: IMAGE_HEIGHT(272) Graphics 1: Not set	Specifies the vertical width of image for graphics 1 and 2.
Macro line offset	input.offset	-	Graphics 2: (IMAGE_WIDTH(480) * IMAGE_PIXEL_SIZE(2)) Graphics 1: Not set	Specifies the macro line offset for graphics 1 and 2.
Show/hide setting of the graphics area frame	input.frame_edge	-	Graphics 2: false Graphics 1: Not set	Sets the graphics area frame to be displayed or not to be displayed.
X-coordinate of display start position	input.coordinate.x	-	Graphics 2: 0 Graphics 1: Not set	Specifies the horizontal start position of the graphics area.
Y-coordinate of display start position	input.coordinate.y	-	Graphics 2: 0	Specifies the vertical start position of the graphics area.

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
			Graphics 1: Not set	
Control setting for blending	blend. blend_control	-	Graphics 2: GLCDC_BLEND_CONTROL_NONE Graphics 1: Not set	Sets alpha blending.
Show/hide setting of the image	blend.visible	-	Graphics 2: true Graphics 1: Not set	Sets the image to be displayed or not to be displayed.
Show/hide setting of the rectangle alpha blending area frame	blend. frame_edge	-	Graphics 2: false Graphics 1: Not set	Sets the frame of the rectangle alpha blending area to be displayed or not to be displayed.
Fixed alpha value	blend.fixed_ blend_value	-	Graphics 2: 0 Graphics 1: Not set	Specifies the fixed alpha value (valid only when blend_control is GLCDC_BLEND_CONTROL_FIXED).
Alpha value to be increased/ decreased	blend. fade_speed	-	Graphics 2: 0 Graphics 1: Not set	Specifies the alpha value to be increased or decreased (valid only when blend_control is GLCDC_BLEND_CONTROL_FADEIN or GLCDC_BLEND_CONTROL_FADEOUT).
X-coordinate of the blending start position	blend.start_ coordinate.x	-	Graphics 2: 0 Graphics 1: Not set	Specifies the horizontal width of the rectangle alpha blending area and the horizontal start position of the rectangle alpha blending (invalid when blend_control is GLCDC_BLEND_CONTROL_NONE).
X-coordinate of the blending end position	blend.end_ coordinate.x	-	Graphics 2: IMAGE_WIDTH(480) Graphics 1: Not set	
Y-coordinate of the blending start position	blend.start_ coordinate.y	-	Graphics 2: 0 Graphics 1: Not set	Specifies the vertical width of the rectangle alpha blending area and the vertical start position of the rectangle alpha blending (invalid when blend_control is GLCDC_BLEND_CONTROL_NONE).
Y-coordinate of the	blend.end_ coordinate.y	-	Graphics 2: IMAGE_HEIGHT(272)	

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
blending end position			Graphics 1: Not set	
Enable/ disable setting of chroma key	chromakey. enbale	-	Graphics 2: false Graphics 1: Not set	Enables or disables chroma keying.
R value for chroma keying	chromakey. before.byte.r	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the R value for chroma keying (invalid when chromakey.enbale is false).
G value for chroma keying	chromakey. before.byte.g	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the G value for chroma keying (invalid when chromakey.enbale is false).
B value for chroma keying	chromakey. before.byte.b	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the B value for chroma keying (invalid when chromakey.enbale is false).
A value after chroma key replacement	chromakey. after.byte.a	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the A value after replacement by chroma keying (invalid when chromakey.enbale is false).
R value after chroma key replacement	chromakey. after.byte.r	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the R value after replacement by chroma keying (invalid when chromakey.enbale is false).
G value after chroma key replacement	chromakey. after.byte.g	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the G value after replacement by chroma keying (invalid when chromakey.enbale is false).
B value after chroma key replacement	chromakey. after.byte.b	-	Graphics 2: 0xFF Graphics 1: Not set	Specifies the B value after replacement by chroma keying (invalid when chromakey.enbale is false).
Bit endianness of the output data	output.endian	-	GLCDC_ENDIAN_LITTLE	Sets to little endian or big endian.
Pixel sequence of the output data	output. color_order	-	RSK RX65N: GLCDC_COLOR_ ORDER_BGR	Sets the pixel sequence of the output data to R-G-B or B-G-R in order.

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
			RSK RX72N, Envision RX72N, Envsion RX65N: GLCDC_COLOR_ORDER_RGB	*The setting differs for each board.
Sequence of correction processing	output.correction_proc_order	IMGC_OUTCTL_CALIB_ROUTE	←	Performs brightness and contrast adjustments first, and then gamma correction. or Performs gamma correction first, and then brightness and contrast adjustments.
Dithering mode selection	output.dithering.dithering_on	IMGC_DITHER_ACTIVE	←	For true, sets to '0: truncated, 1: rounded' or dithering with 2x2 pattern. For false, sets to 'truncated'. (Setting values of structure members under glcdc_cfg_t.output.dithering are ignored.)
Dithering mode selection 2	output.dithering.dithering_mode	IMGC_DITHER_MODE	←	Sets to '0: truncated, 1: rounded' or dithering with 2x2 pattern.
Dithering pattern value A	output.dithering.dithering_pattern_a	IMGC_DITHER_2X2_PA	←	Specifies pattern value A of dithering with 2x2 pattern (valid only when dithering_mode is 'GLCDC_DITHERING_MODE_2X2PATTERN').
Dithering pattern value B	output.dithering.dithering_pattern_b	IMGC_DITHER_2X2_PA	←	Specifies pattern value B of dithering with 2x2 pattern (valid only when dithering_mode is 'GLCDC_DITHERING_MODE_2X2PATTERN').
Dithering pattern value C	output.dithering.dithering_pattern_c	IMGC_DITHER_2X2_PC	←	Specifies pattern value C of dithering with 2x2 pattern (valid only when dithering_mode is 'GLCDC_DITHERING_MODE_2X2PATTERN').
Dithering pattern value D	output.dithering.dithering_pattern_d	IMGC_DITHER_2X2_PD	←	Specifies pattern value D of dithering with 2x2 pattern (valid only when dithering_mode is 'GLCDC_DITHERING_MODE_2X2PATTERN').
Enable/ disable setting of brightness adjustment	output.brightness.enable	IMGC_BRIGHT_OUTCTL_ACTIVE	←	Enables or disables brightness adjustment. (When it is invalid, the brightness adjustment value for RGB signal is set to 0 regardless of setting

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
				values of structure members under glcdc_cfg_t.output.brightness.) *It is always set to true by QE for Display [RX].
Brightness adjustment value for R signal	output. brightness.r	IMGC_BRIGHT_OUTCTL_OFFSET_R	←	Specifies the brightness adjustment value for the R signal.
Brightness adjustment value for G signal	output. brightness.g	IMGC_BRIGHT_OUTCTL_OFFSET_G	←	Specifies the brightness adjustment value for the G signal.
Brightness adjustment value for B signal	output. brightness.b	IMGC_BRIGHT_OUTCTL_OFFSET_B	←	Specifies the brightness adjustment value for the B signal.
Enable/ disable setting of contrast adjustment	output.contrast. enable	IMGC_CONTRAST_OUTCTL_ACTIVE	←	Enables or disables contrast adjustment. (When it is invalid, the contrast adjustment value for the RGB signal is set to 1.000 regardless of setting values of structure members under glcdc_cfg_t.output.contrast.) *It is always set to true by QE for Display [RX].
Contrast adjustment value for R signal	output. contrast.r	IMGC_CONTRAST_OUTCTL_GAIN_R	←	Specifies the contrast adjustment value for the R signal.
Contrast adjustment value for G signal	output. contrast.g	IMGC_CONTRAST_OUTCTL_GAIN_G	←	Specifies the contrast adjustment value for the G signal.
Contrast adjustment value for B signal	output. contrast.b	IMGC_CONTRAST_OUTCTL_GAIN_B	←	Specifies the contrast adjustment value for the B signal.
Enable/ disable setting of gamma correction	output.gamma. enable	IMGC_GAMMA_ACTIVE	←	Enables or disables contrast adjustment. (When it is invalid, setting values of structure members under glcdc_cfg_t.output.gamma are ignored.) *It is always set to true by QE for Display [RX].
Gamma correction table for the R signal	output.gamma. p_r	- gain[16] IMGC_GAMMA_R_GAIN_00~IMGC_GAMMA_R_GAIN_15 - Threshold[15]	← (&gs_gamma_table_r)	Specifies the gain value and the start threshold value for each R signal area.

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
		IMGC_GAMMA_R_TH_01~IMGC_GAMMA_R_TH_15		For the structure member, the address of the table variable (gs_gamma_table_r) is set.
Gamma correction table for the G signal	output.gamma.p_g	- gain[16] IMGC_GAMMA_R_GAIN_00~IMGC_GAMMA_R_GAIN_15 - Threshold[15] IMGC_GAMMA_G_TH_01~IMGC_GAMMA_G_TH_15	← (&gs_gamma_table_g)	Specifies the gain value and the start threshold value for each G signal area. For the structure member, the address of the table variable (gs_gamma_table_g) is set.
Gamma correction table for the B signal	output.gamma.p_b	- gain[16] IMGC_GAMMA_B_GAIN_00~IMGC_GAMMA_B_GAIN_15 - Threshold[15] IMGC_GAMMA_B_TH_01~IMGC_GAMMA_B_TH_15	← (&gs_gamma_table_b)	Specifies the gain value and the start threshold value for each B signal area. For the structure member, the address of the table variable (gs_gamma_table_b) is set.
Enable/ disable setting of CLUT memory	clut.enable	-	Graphics 2: false Graphics 1: Not set	Update or not update CLUT memory. (If CLUT memory is not updated, setting values of structure members under clut are ignored.)
Pointer to the start address of the CLUT memory	clut.p_base	-	Graphics 2: FIT_NO_PTR Graphics 1: Not set	Reads the value at the address designated by the pointer and copies it to the CLUT memory.
Start entry number of the CLUT memory to be updated	clut.start	-	Graphics 2: 0 Graphics 1: Not set	Starts updating the CLUT memory from the entry number specified.
Entry size of the CLUT memory to be updated	clut.size	-	Graphics 2: 256 Graphics 1: Not set	Updates the CLUT memory for the specified size.
Enable/ disable setting of VPOS detection	detection.vpos_detect	-	true	Enables or disables VPOS detection.
Enable/ disable setting of GR1UF detection	detection.gr1uf_detect	-	false	Enables or disables GR1UF detection.
Enable/ disable	detection.gr2uf_detect	-	true	Enables or disables GR2UF detection.

Outline	Structure Member	Define Directive Output by QE for Display[RX]	Setting Value in the Sample Program	Description
setting of GR2UF detection				
Enable/ disable setting of the VPOS interrupt	interrupt. vpos_enable	-	true	Enables or disables the VPOS interrupt.
Enable/ disable setting of the GR1UF interrupt	interrupt. gr1uf_enable	-	false	Enables or disables the GR1UF interrupt.
Enable/ disable setting of the GR2UF interrupt	interrupt. gr2uf_enable	-	true	Enables or disables the GR2UF interrupt.

----: There is no corresponding define directive.

← : The setting is made by a define directive output by QE for Display [RX].

7.4 Peripheral Devices

Table 7-2 shows the peripheral devices used by this sample program.

Table 7-2 Peripheral Devices to be Used

Peripheral Device	Usage
Graphics LCD controller (GLCDC)	Indicates the display. - Graphics 2 is used. - Graphics 1 is unused. - Color format: RGB565 - Output data format: RGB565 (parallel 16 bits)
Interrupt controller (ICU)	Controls interrupts generated by the GLCDC.
Extended RAM	Used for a frame buffer.

7.5 Memory

Table 7-3 shows the sizes of ROM and RAM used by this sample program.

Table 7-3 Sizes of ROM and RAM

Project	Memory Used	Size	Remarks
RSK RX72N	ROM	17266 bytes	Code, constant data, and initial value data
	RAM	8129 bytes	Data: 376 bytes Uninitialized data: 2633 bytes STACK: 5120 bytes
Envision RX72N	ROM	17270 bytes	Code, constant data, and initial value data
	RAM	8129 bytes	Data: 376 bytes Uninitialized data: 2633 bytes STACK: 5120 bytes
RSK RX65N	ROM	16842 bytes	Code, constant data, and initial value data
	RAM	8013 bytes	Data: 376 bytes Uninitialized data: 2517 bytes STACK: 5120 bytes
Envision RX65N	ROM	16862 bytes	Code, constant data, and initial value data
	RAM	8013 bytes	Data: 376 bytes Uninitialized data: 2517 bytes STACK: 5120 bytes

7.6 Memory Map

Table 7-4 shows the specific section used by this sample program.

Table 7-4 Specific Section Used by the Sample Program.

Section	Type	Description
FRAME_BUFFER	data	A buffer in which display data is stored. It is allocated to the fixed address (0x00800000) of the extended RAM area.

7.7 Interrupts Used

Table 7-5 lists interrupts used in the sample program. These interrupts are controlled by the GLCDC FIT module. For details, refer to the user's manual for the GLCDC FIT module.

Table 7-5 Interrupts Used in the Sample Program

Interrupt	Priority	Description
VPOS	5	Generated in response to the detection of lines in the graphics 2 block (interrupt of group AL1; source number: 8).
GR1UF		Generated when the supply of graphics data to the alpha-blending section of the graphics 1 block is delayed due to the load of access to the bus (interrupt of group AL1; source number: 9). This interrupt is inhibited in this sample program since the graphics 1 block is not used.
GR2UF		Generated when the supply of graphics data to the alpha-blending section of the graphics 2 block is delayed due to the load of access to the bus (interrupt of group AL1; source number: 10).

7.8 Using this Sample Program

The main.c file of this sample program includes code for processing required to initialize the GLCDC and produce a display on the panel. When you refer to code for a user system, equivalents of the following variables and functions in the main.c file will be required. Although some define directives are used to set up values to be assigned to parameters of the GLCDC FIT module, they are not absolutely required for the main.c file since immediate values can be assigned instead.

Check the following as a reference for a user system.

7.8.1 Variable List

Table 7-6 shows the required variables.

Table 7-6 Required Variables

Type	Variable Name	Contents
static glcdc_cfg_t	gs_glcdc_init_cfg	Structure variable passed to the R_GLCDC_Open function of the GLCDC FIT module. Specifies the information required for the display on the screen.
static volatile bool	gs_first_interrupt_flag	A variable to check the first VPOS interrupt in the callback function
static const gamma_correction_t	gs_gamma_table_r = {gain[16]={IMGC_GAMMA_R_GAIN_00~IMGC_GAMMA_R_GAIN_15} , Threshold[15]={IMGC_GAMMA_R_TH_01~IMGC_GAMMA_R_TH_15}}	Gamma (red) table. Table data use the define directive in the header file (r_image_config_{board name}.h) output from QE for Display [RX].
static const gamma_correction_t	gs_gamma_table_g = {gain[16]={IMGC_GAMMA_G_GAIN_00~IMGC_GAMMA_G_GAIN_15} , Threshold[15]={IMGC_GAMMA_G_TH_01~IMGC_GAMMA_G_TH_15}}	Gamma (green) table. Table data use the define directive in the header file (r_image_config_{board name}.h) output from QE for Display [RX].
static const gamma_correction_t	gs_gamma_table_b = {gain[16]={IMGC_GAMMA_B_GAIN_00~IMGC_GAMMA_B_GAIN_15} , Threshold[15]={IMGC_GAMMA_B_TH_01~IMGC_GAMMA_B_TH_15}}	Gamma (blue) table. Table data use the define directive in the header file (r_image_config_{board name}.h) output from QE for Display [RX].

7.8.2 Function List

Table 7-7 shows the required functions.

Table 7-7 Required Functions

Function Name	Outline
glcdc_initialize	Initializes and starts the GLCDC.
qe_for_display_parameter_set	Collects the parameters of the GLCDC FIT module which are set by QE for Display [RX].
glcdc_callback	A callback function of the GLCDC FIT module. It is not required when the GLCDC interrupt function is not used. When you do not use this function, disable the interrupt setting and the pointer to the callback function (set FIT_NO_FUNC) according to the manual of the GLCDC FIT module.
board_port_setting	A function for setting pins depending on the board (control pins for backlight and reset of the LCD panel). It is not required because it depends on the LCD panel to be used and the LCD connection type. In the RSK and Envision, this function is used to control the backlight and reset of the LCD panel with the general-purpose input/output port.

8. Using QE for Display [RX]

This chapter describes the usage of QE for Display [RX] according to the actual flow of display adjustment. For details on the facilities of QE for Display [RX], refer to the help file which comes with QE for Display [RX].

8.1 Starting QE for Display [RX]

Selecting [Renesas Views] -> [Renesas QE] -> [Display Tuning RX (QE)] from the menu of the e^2 studio starts QE for Display [RX] (Figure 8-1).

Figure 8-1 is the display of a block diagram of the hardware of the GLCDC, showing the path for the output of image data and the relationships between the positions where images are to be corrected. Clicking on [Brightness] or [Contrast] for the adjustment of image quality produces the [Image Quality Adjustment] tabbed page, which allows various adjustments.

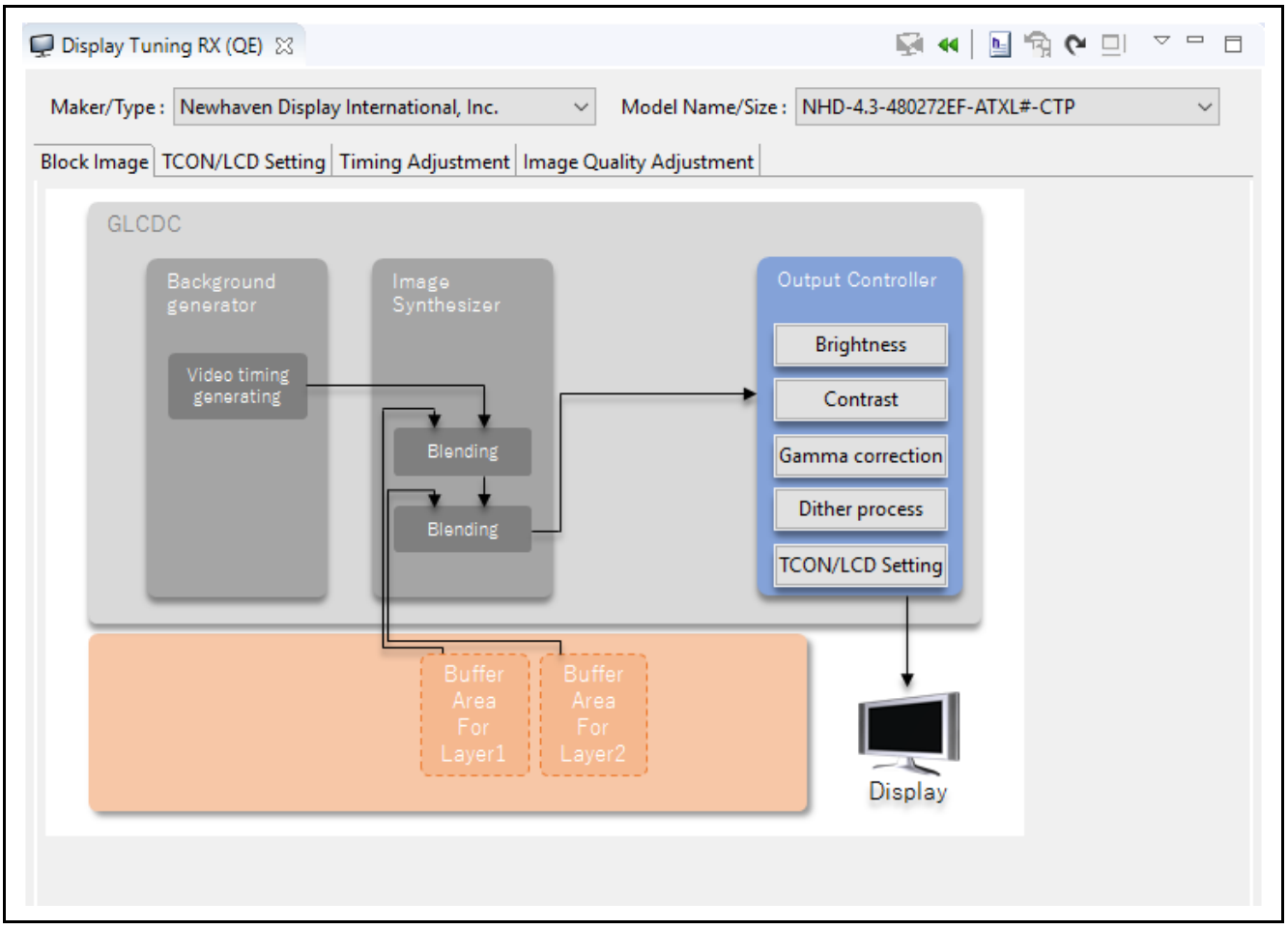


Figure 8-1 Initial State of QE for Display [RX]

8.2 Setting Data on the LCD Panel

Information on the LCD panel which is connected to the user system is specified. When the display is connected to a system under development, you need to compare and adjust the specifications of the LCD panel and the display controller and find specifiable and appropriate settings. Information that has been specified is used in comparison.

The LCD mounted on the RSK is an NHD-4.3-480272EF-ATXL#-CTP manufactured by Newhaven Display International. The LCD mounted on the Envision is an ER-TFT043-3 manufactured by EastRising Technology Co., Ltd.

The package of QE for Display [RX] V1.1.0 includes information on each of these LCD panels, so select the given type.

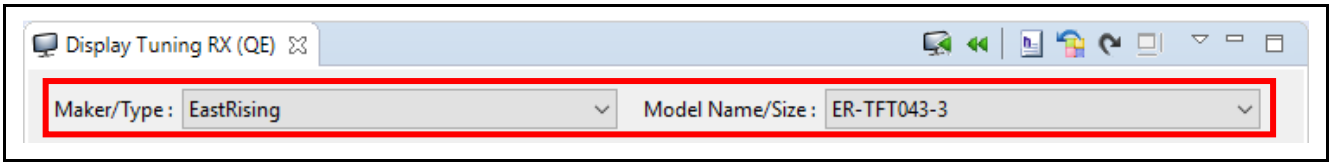


Figure 8-2 Selecting the LCD Panel

When information on the LCD panel is set, the display type can be specified by selecting from among three patterns. The display type adopted is 3 (the method of using Vsync, Hsync, and DE signals) for the LCD panel mounted on the RSK and Envision.

For details on setting information on the LCD panel, refer to chapter 9, Setting Detailed Data on the LCD Panel.

8.3 Setting the Output of Control Signals

Select the [TCON/LCD Setting] tabbed page of QE for Display [RX] and specify the settings for the output of control signals (Figure 8-3).

The following settings for the output of control signals are available on this page.

[Panel Driver Signal (TCON) Output Selection]
Selection of output pins:
Output to the LCD_TCON0 to LCD_TCON3 pins (TCON0 to TCON3)
Active sense of control signals:
Positive sense: [Active High]
Negative sense: [Active Low]
[LCD Setting]
[LCD Output Format]
24-bit RGB888 output: [24bit (GLCDC_OUT_FORMAT_24BITS_RGB888)]
18-bit RGB666 output: [18bit (GLCDC_OUT_FORMAT_18BITS_RGB666)]
16-bit RGB565 output: [16bit (GLCDC_OUT_FORMAT_16BITS_RGB565)]
[Timing of Output Data]
Output on rising edges of the panel clock: [Rising (GLCDC_SIGNAL_SYNC_EDGE_RISING)]
Output on falling edge of the panel clock: [Falling (GLCDC_SIGNAL_SYNC_EDGE_FALLING)]

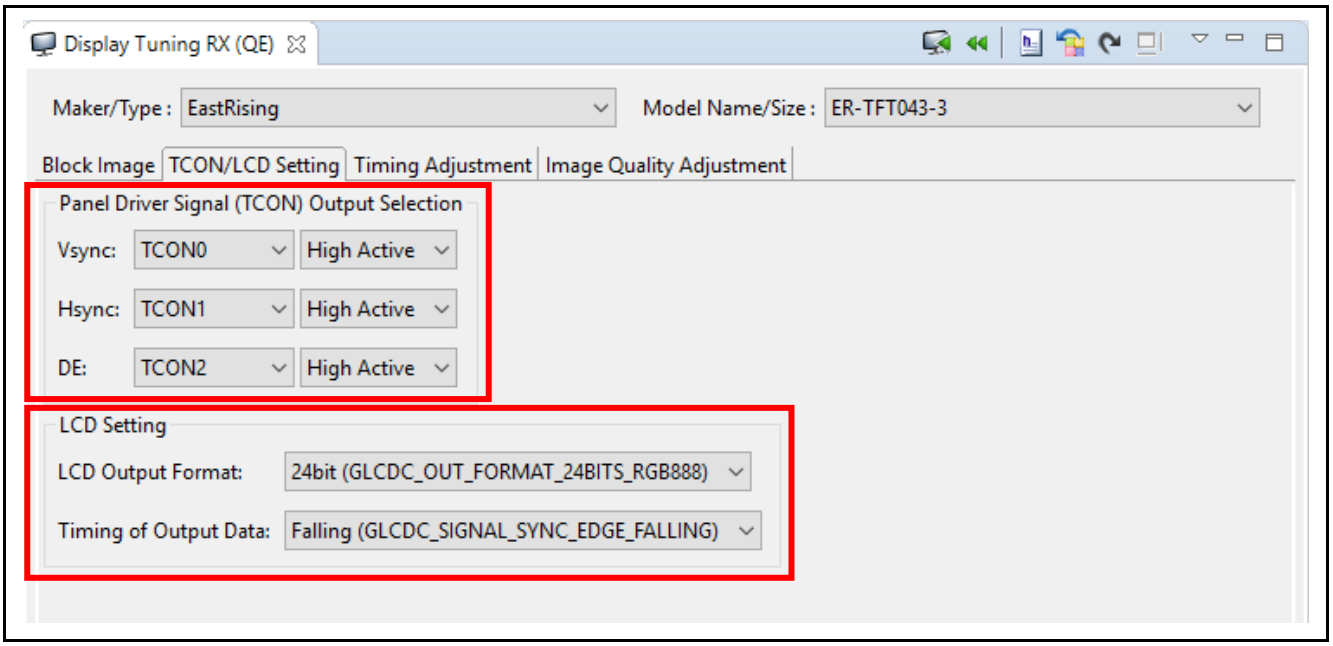


Figure 8-3 [TCON/LCD Setting] Tabbed Page

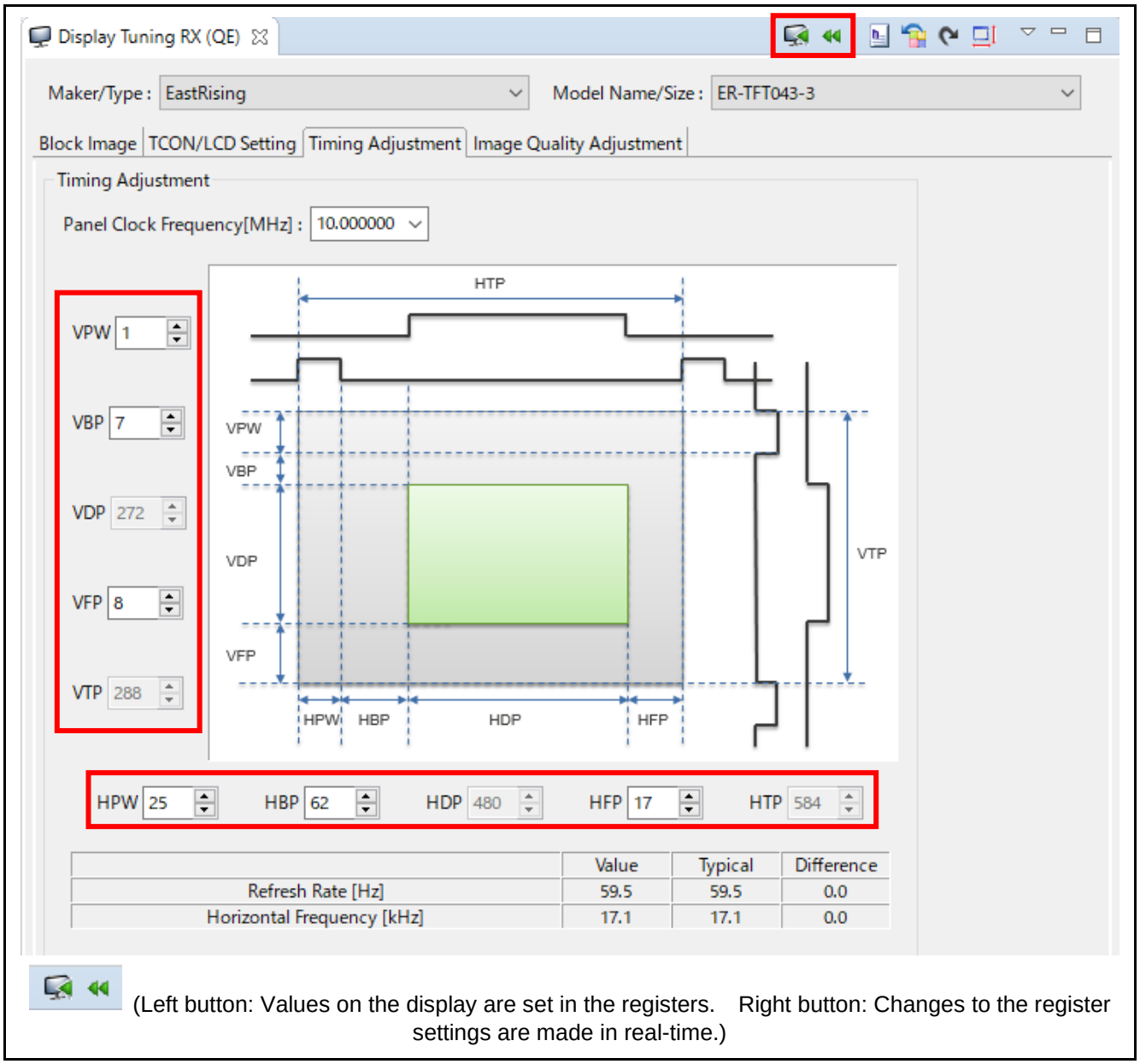
The following lists the settings that match the specifications of the each RSK and Envision board.

Table 8-1 LCD Panel Used for the Sample Program

	RSK RX72N / RX65N	Envision RX72N / RX65N
Selection of output pins		
Vsync	TCON0	TCON0
Hsync	TCON2	TCON2
DE	TCON3	TCON3
Active sense of control signals		
Vsync	Negative sense: [Active Low]	Negative sense: [Active Low]
Hsync	Negative sense: [Active Low]	Negative sense: [Active Low]
DE	Positive sense: [Active High]	Positive sense: [Active High]
[LCD Output Format]	16-bit RGB565 output [16bit (GLCDC_OUT_FORMAT_16BITS_RGB565)]	16-bit RGB565 output [16bit (GLCDC_OUT_FORMAT_16BITS_RGB565)]
[Timing of Output Data]	Output on rising edges of the panel clock [Rising (GLCDC_SIGNAL_SYNC_EDGE_RISING)]	Output on rising edges of the panel clock [Rising (GLCDC_SIGNAL_SYNC_EDGE_RISING)]

8.4 Adjusting the Timing of Control Signals for the LCD Panel

When the values shown in Figure 8-4 are changed after the debugger is connected and the sample program is executed, the timing of control signals can be changed. This tool directly writes the changed values to registers of the GLCDC so that they are reflected in the operation of the LCD panel on the Envision.



(Left button: Values on the display are set in the registers. Right button: Changes to the register settings are made in real-time.)

Figure 8-4 Adjusting the Timing of Control Signals

Enter the frequency of the panel clock in the upper-left box in the [Timing Adjustment] area. This frequency is used to calculate the refresh rate, which is indicated at the bottom of the page, along with a value for any difference from the recommended value for the LCD panel. The actual frequency of the panel clock must be specified in the program and cannot be specified by QE for Display [RX]. The recommended value of the LCD panel has been specified as the initial value. In this example, enter 10 MHz, which is the value specified in the sample program.

After that, adjust the individual parameters. The result of adjustment being shown in red numerals means that the value is out of the range of specifications of the GLCDC and of the LCD panel. In such a case, adjust the value so that it is within the range of the specifications of the GLCDC and of the LCD panel. Check the range of values which are allowable in the specifications of the GLCDC and of the LCD panel by hovering the mouse over the adjusted value that is being shown in red.

The recommended values for the LCD panel are also used for these initial values. If you use the RSK, the recommended value of the horizontal front porch (HFP) of the LCD panel is 2; however, this must be modified since it is out of the range of the specifications of the GLCDC. Modify the value to 3 or more to satisfy the specifications of both the LCD panel and GLCDC. After that, the display of the adjusted value is changed from red to black.

When you determine the adjusted values, write the values from this tool to the registers of the GLCDC and check the results.

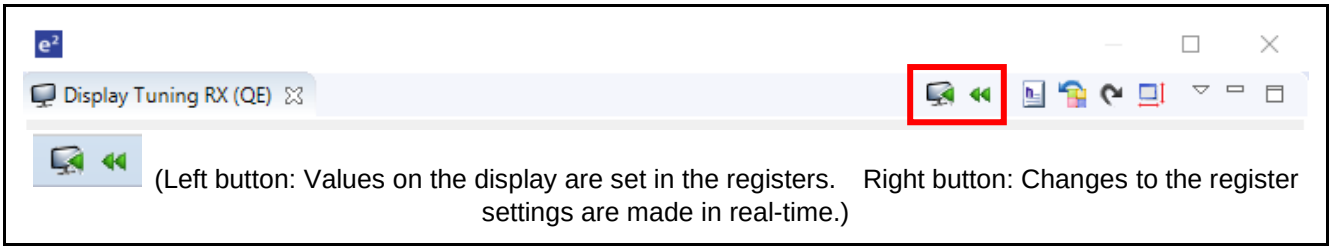


Figure 8-5 Buttons for Setting Registers

The following two methods are used to set or make changes to values in the registers.

Table 8-2 Facilities for Setting or Making Changes to Values in Registers

Button	Name	Description
	Set the Register	The settings are written to registers. This button is only effective if a debugger is connected.
	Set the Registers in Real-time when the Parameters are Changed	When this button is active, changes are automatically written to registers every time the setting is changed. This button is not active by default. Writing to registers is only possible when a debugger is connected; no operation proceeds if a debugger is not connected.

Note:

For the facility to write the adjusted values to registers in QE for Display [RX], the graphics screen is adjusted to be aligned with the upper left of the background screen when the timing is adjusted.

For definitions of the graphics and background screens, refer to the RX65N Group, RX651 Group User's Manual: Hardware (R01UH0590) or RX72N Group User's Manual: Hardware (R01UH0824).

In the sample program, a one-pixel red line is drawn around the outer periphery of a blue-colored image. Adjustment of the sample program will not be needed since values which are appropriate for the RSK and Envision have already been specified. In actual development, however, positions must be adjusted so that the red line around the periphery is displayed on the LCD panel.

Due to the display type and specifications of the LCD panel, fine changes to setting values (e.g. moving by several pixels) or changes to particular settings may not appear on the LCD panel. For example, the LCD panels mounted on the RSK and Envision are of display type 3, which does not allow the movement of positions in response to changes to the settings for the back porches and so on.

8.5 Output of Control Signals and Reflecting the Results of Timing Adjustment

The results of timing adjustment can be reflected in a program through the output of a header file. Clicking on the [Generating Header File] button of QE for Display [RX] (Figure 8-6) generates a header file that reflects the specified control timing.

Figure 8-6 [Generating Header File] Button

When you select [For Timing and TCON Settings] only and click on [Generate], a header file is generated at the specified destination for output. The name of the header file and the output destination can be specified as desired.

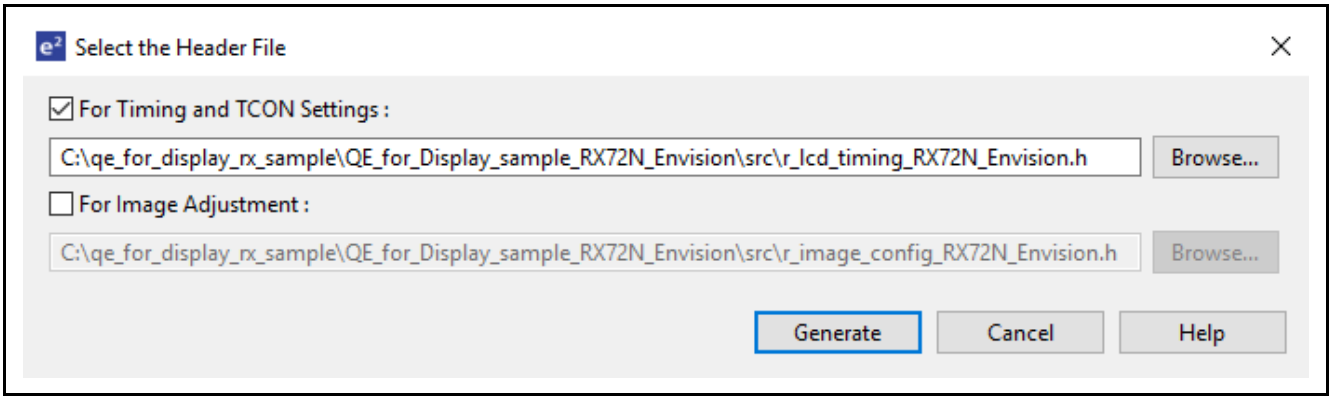


Figure 8-7 Selecting [For Timing and TCON Settings]

To reflect the timing of the sample project, output the header file with the name 'r_lcd_timing_< *RX72N / RX65N* >_< *RSK / Envision* >.h' in the following directory, and clean and build the project.

Directory:

<*workspace folder*> ¥QE_for_Display_sample_< *RX72N / RX65N* >_< *RSK / Envision* >¥src

8.6 Image-Downloading Facility

In QE for Display [RX], image quality is adjusted by checking the LCD according to the characteristics of the LCD. The image that is displayed on the LCD can be changed without changing the program.

Using the image-downloading facility downloads image data (binary file) from the personal computer to be displayed on the LCD.

Figure 8-8 [Send the Image File] Button

Click on the [Send the Image File] button on the toolbar.

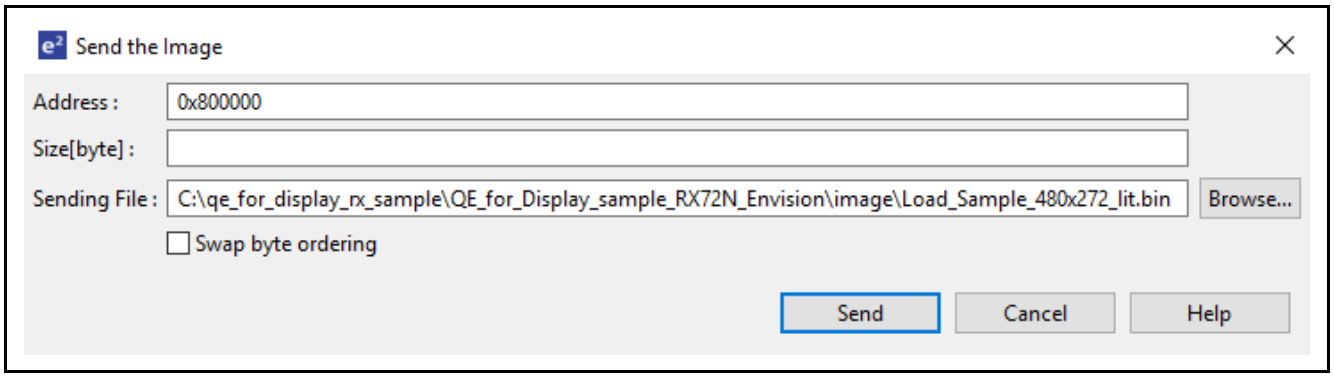


Figure 8-9 [Send the Image] Dialog Box

Specify the address of the destination and the file to be sent in the [Send the Image] dialog box. By default, the value that has been set in the graphics 2 frame buffer control register is specified as [Address]. The address need not be changed in the sample program since the address of graphics 2 has been used. When specification of [Size] is omitted, the entire file specified in the [Sending File] edit box is written to the range from the address specified in the [Address] edit box.

This application note includes sample image data. Send the following.

Directory:

<*workspace folder*> ¥QE_for_Display_sample_< *RX72N / RX65N* >_< *RSK / Envision* >¥image

File:

Load_Sample_480x272_lit.bin

When sending is successfully completed, color bars are displayed as shown in Figure 8-10.

Figure 8-10 Image on Completion of Sending

8.7 Adjusting Image Quality

Clicking on the items for image quality adjustment enclosed by red frames in Figure 8-11 on the [Block Image] tabbed page makes the [Image Quality Adjustment] tabbed page appear, enabling the adjustment of image quality.

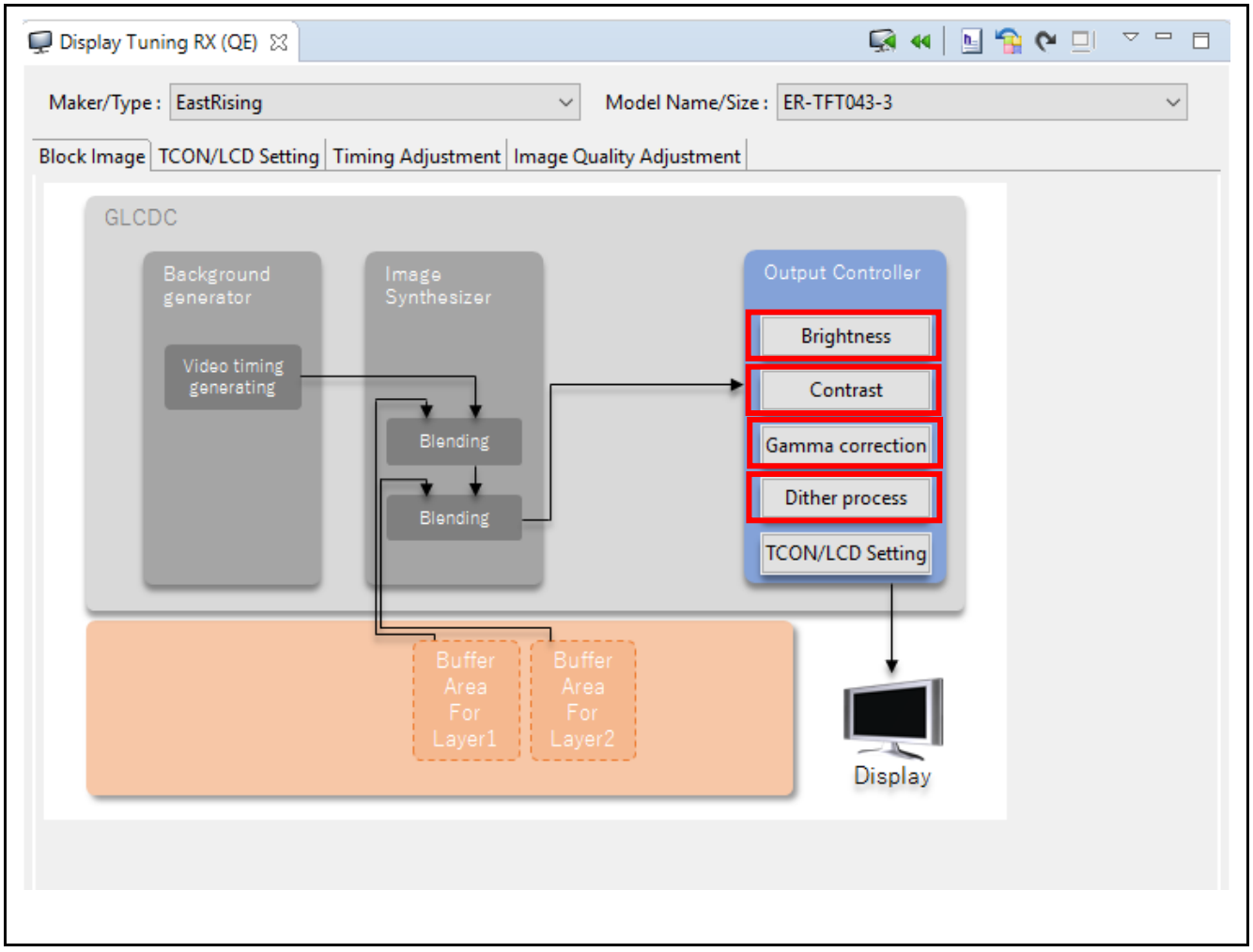


Figure 8-11 Buttons for Selecting the Adjustment of Image Quality

The [Image Quality Adjustment] tabbed page enables the adjustment of image quality. QE for Display [RX] supports [Calibration Route Setting] and four facilities for adjusting image quality: [Brightness], [Contrast], [Gamma correction], and [Dither process].

Changes to these settings are reflected in real-time, allowing the adjustment of image quality with reference to the display on the LCD panel.

Image quality is adjusted by using [Quick Setting] or [Custom]. If you select [Custom], refer to the RX65N Group, RX651 Group User's Manual: Hardware (R01UH0590), RX72N Group User's Manual: Hardware (R01UH0824) and the RX Family Graphic LCD Controller Module Using Firmware Integration Technology (R01AN3609), check the meanings of the settings made in each of the registers and the specifiable values, and adjust the image quality accordingly.

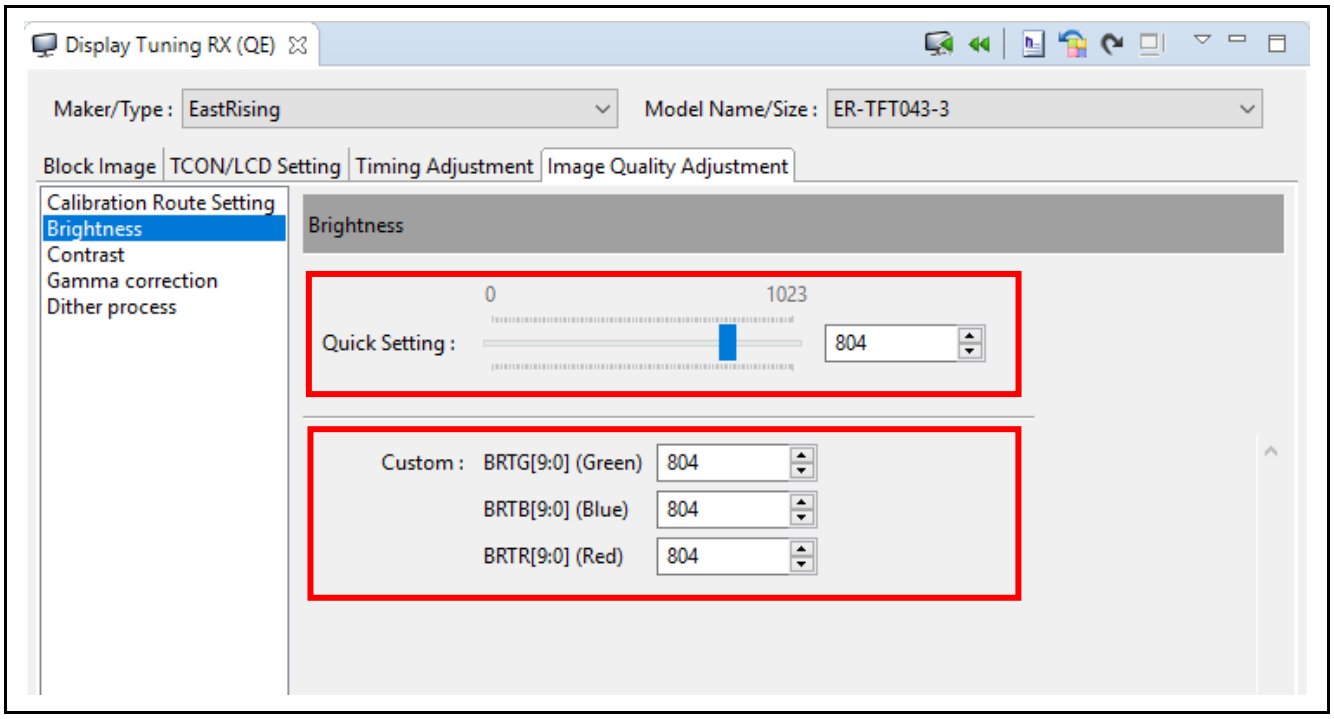


Figure 8-12 [Image Quality Adjustment] Tabbed Page

8.8 Generating a Header File with the Results of Adjusting Image Quality

Click on the [Generating Header File] icon of QE for Display [RX] to generate a header file that reflects the results of image quality adjustment which have been specified (see Figure 8-13).

When you select [For Image Adjustment] only and click on [Generate], a header file is generated at the specified destination for output. The name of the header file and the output destination can be specified as desired.

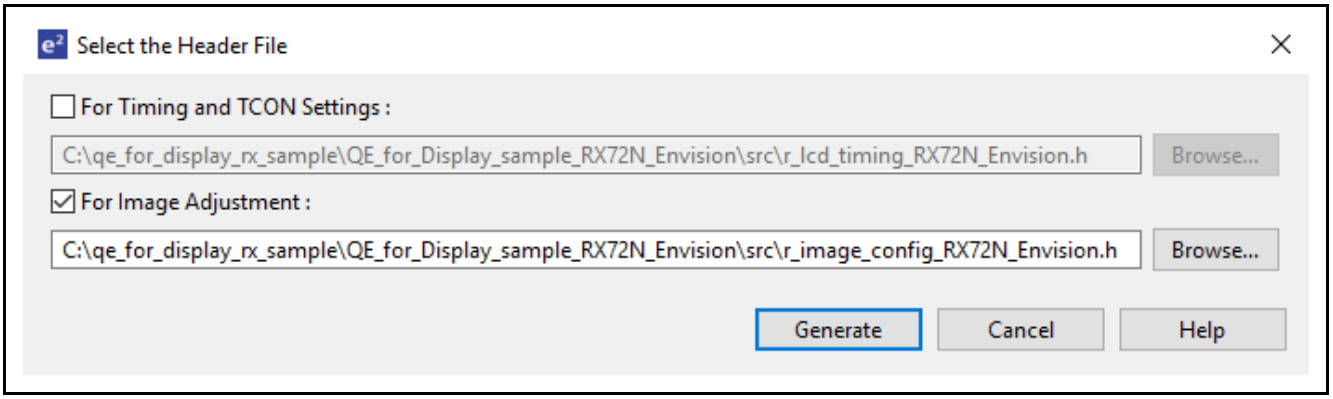


Figure 8-13 Generating [For Image Adjustment]

To reflect the settings of image quality adjustment in the sample project, output the header file with the name 'r_image_config_< *RX72N / RX65N* >_< *RSK / Envision* >.h' in the following directory, and clean and build the project.

Directory:

<*workspace folder*> ¥QE_for_Display_sample_< *RX72N / RX65N* >_< *RSK / Envision* >¥src

9. Setting Detailed Data on the LCD Panel

If you select [Custom] from the [Maker/Type] pull-down list in the upper section of the dialog box shown in Figure 8-1, the [Edit Custom Display Data] dialog box (Figure 9-1) appears. Enter information on the LCD panel in this dialog box.

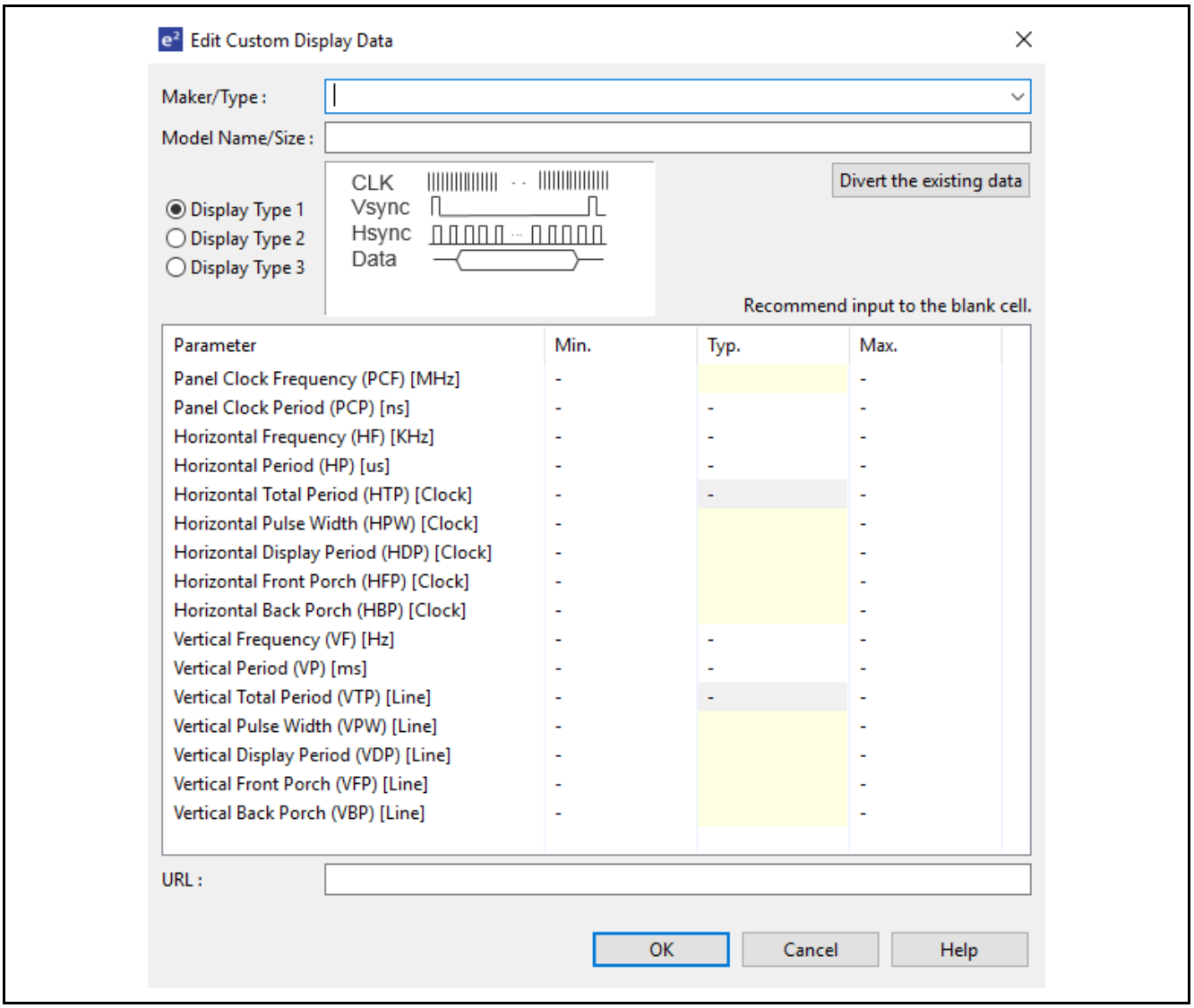


Figure 9-1 [Edit Custom Display Data] Dialog Box

9.1 Entering Names for Registration

Enter the desired names in [Maker/Type] and [Model Name/Size] in the [Edit Custom Display Data] dialog box (Figure 9-2). These names will be registered in the drop-down list for selection.

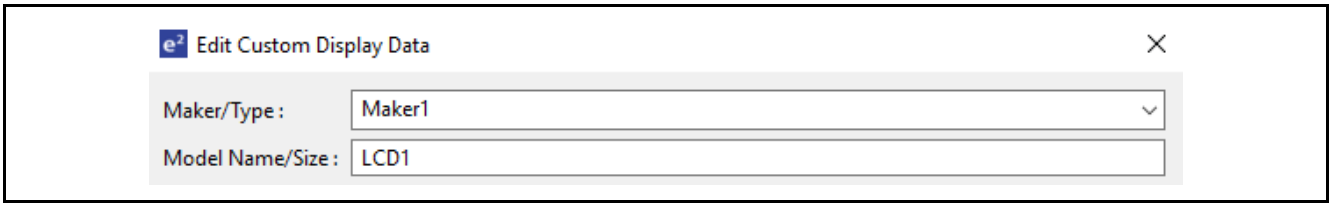


Figure 9-2 Registering a Name

9.2 Selecting the Display Type

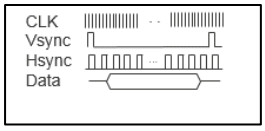
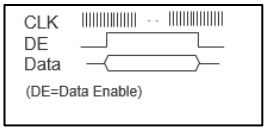
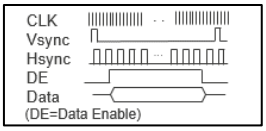
Table 9-1, Main Control Signals, lists the control signals required for connecting an LCD panel. QE for Display [RX] supports devices which have three display types with combination of those control signals.

Table 9-1 Main Control Signals

Name	Outline of Facility
Horizontal synchronization signal (Hsync)	The signal that generates the timing for one line to be displayed
Vertical synchronization signal (Vsync)	The signal that generates the timing for one screen to be displayed
Panel clock (CLK)	The signal that drives the sampling of pixels to be displayed
Display enable (DE)	The signal indicating that valid data are being output
Data (Data)	Data to be displayed

The user must check which control signals are required in the specifications of the LCD panel in use and select the appropriate one from among the three display types shown in Table 9-2, Display Types and Control Signals to be Used.

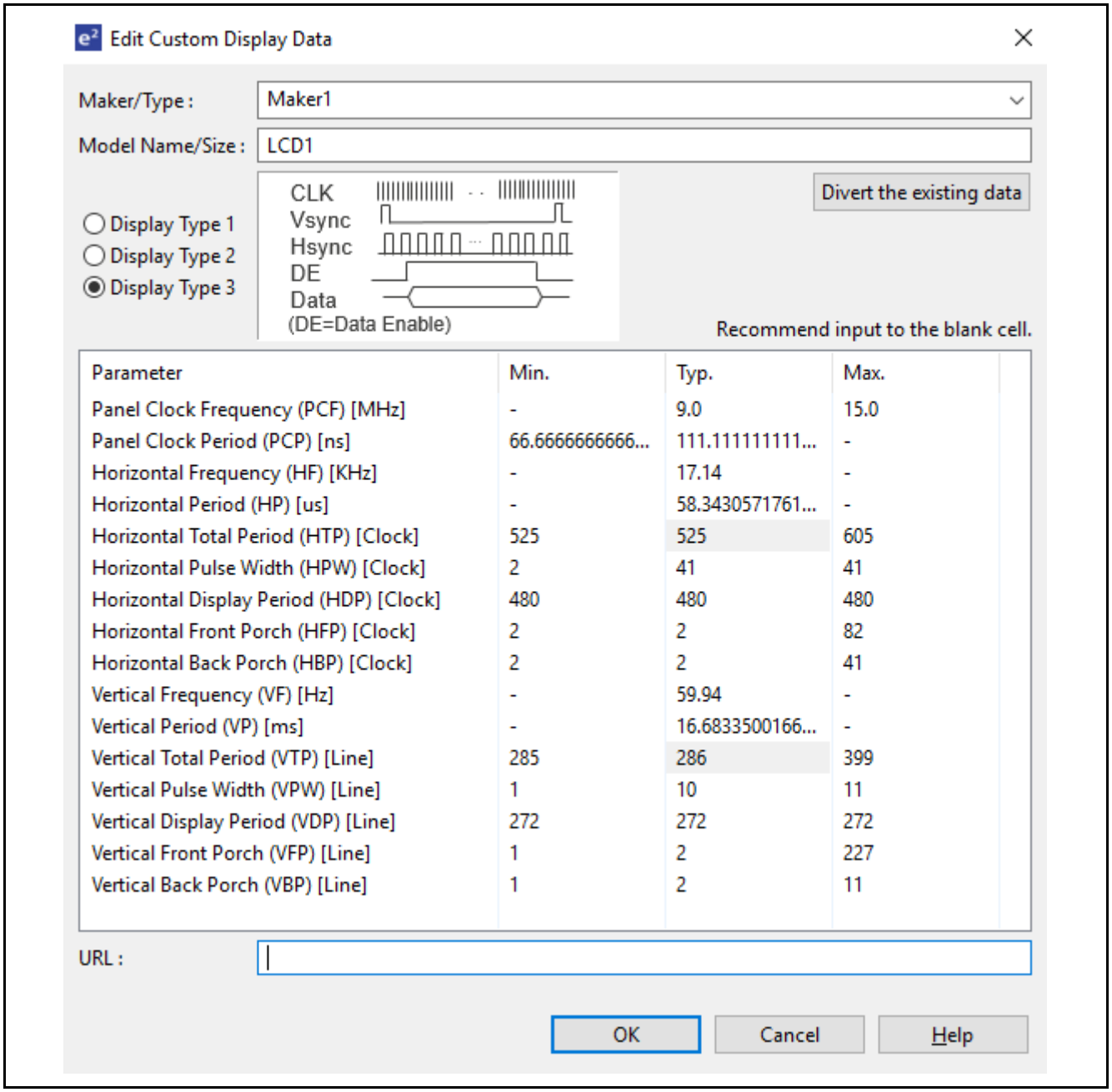
Table 9-2 Display Types and Control Signals to be Used

Name	Display type 1	Display type 2	Display type 3
			
Horizontal synchronization signal (Hsync)	Used	Unused	Used
Vertical synchronization signal (Vsync)	Used	Unused	Used
Panel clock (CLK)	Used	Used	Used
Display enable (DE)	Unused	Used	Used
Data (Data)	Used	Used	Used

9.3 Entering Control Timing

Enter the control timing with reference to the datasheet for the LCD panel. Values entered under Typ. are used as the initial values for timing control. Values entered under Min. and Max. are used to check whether or not the timing as adjusted by using the QE for Display [RX] GUI is within the range.

Figure 9-3 shows the result of data input for the LCD panel mounted on the Envision. Enter values with reference to Table 9-3, Excerpt from the Datasheet for the LCD Panel on the RSK.



Parameter	Min.	Typ.	Max.
Panel Clock Frequency (PCF) [MHz]	-	9.0	15.0
Panel Clock Period (PCP) [ns]	66.6666666666...	111.111111111...	-
Horizontal Frequency (HF) [KHz]	-	17.14	-
Horizontal Period (HP) [us]	-	58.3430571761...	-
Horizontal Total Period (HTP) [Clock]	525	525	605
Horizontal Pulse Width (HPW) [Clock]	2	41	41
Horizontal Display Period (HDP) [Clock]	480	480	480
Horizontal Front Porch (HFP) [Clock]	2	2	82
Horizontal Back Porch (HBP) [Clock]	2	2	41
Vertical Frequency (VF) [Hz]	-	59.94	-
Vertical Period (VP) [ms]	-	16.6833500166...	-
Vertical Total Period (VTP) [Line]	285	286	399
Vertical Pulse Width (VPW) [Line]	1	10	11
Vertical Display Period (VDP) [Line]	272	272	272
Vertical Front Porch (VFP) [Line]	1	2	227
Vertical Back Porch (VBP) [Line]	1	2	11

Figure 9-3 Result of Control Timing Input

Table 9-3 Excerpt from the Datasheet for the LCD Panel on the RSK

Parameter	Symbol	Spec.			Unit
		Min.	Typ.	Max.	
Clock cycle	fclk	-	9	15	MHz
Hsync cycle	1/th	-	17.14	-	KHz
Vsync cycle	1/th	-	59.94	-	Hz
Horizontal Signal					
Horizontal cycle	th	525	525	605	CLK
Horizontal display period	thd	480	480	480	CLK
Horizontal front porch	Thf	2	2	82	CLK
Horizontal pulse width	thp	2	41	41	CLK
Horizontal back porch	thb	2	2	41	CLK
Vertical Signal					
Vertical cycle	tv	285	286	399	H
Vertical display period	tvd	272	272	272	H
Vertical front porch	tvf	1	2	227	H
Vertical pulse width	tvp	1	10	11	H
Vertical back porch	rvb	1	2	11	H

9.4 Editing Created Display Data

When the [Edit and Delete the Custom Display...] menu item is executed after clicking on the menu button on the toolbar, the created display data can be re-edited.

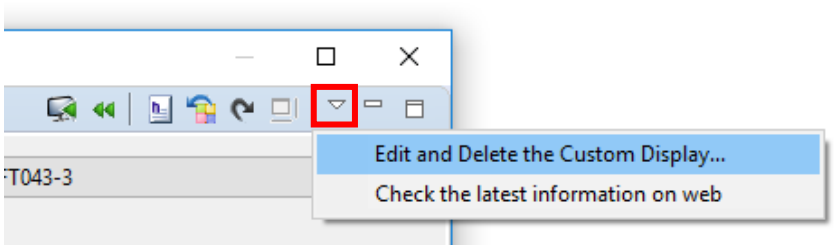


Figure 9-4 [Edit and Delete the Custom Display...] Menu Item

10. Adapting the Sample Program to the User Environment

To adapt this sample program to the user environment, the sample program must be modified according to the procedure in section 1.1, Flow of System Development with QE for Display [RX]. This chapter describes changed points and gives supplementary explanations and notes for each procedure.

The following takes the RSK RX72N as an example.

Points to be changed depending on the user environment

<Creating a project>
- Selecting the Smart Configurator when a new project is created
- Setting a section

<Settings in the Smart Configurator>
- Setting a clock
- Adding the GLCDC FIT module
- Setting pins for use by the GLCDC

<Creating a program (modification of the main.c file)>
- Panel clock
- Pin assignments (bit endian)
- Pixel order
- Other control pins (backlight, reset, etc.)
- Names of the header files
- Display size of graphics 2
- Base address of graphics 2

10.1 Confirming Specifications

Parameters that are not supported by QE for Display [RX] among those of the GLCDC that require setting must be set by the user (refer to section 7.3, Correspondence between Parameters in the GLCDC FIT Module and Header Files Output by QE for Display [RX]). The user also must control the LCD panel in terms of the user board.

Determine the following settings according to the specifications of the board in the user environment.

- Panel clock
- Pin assignments (bit endian)
- Pixel order
- Other control pins (backlight, reset, etc.)

10.1.1 Panel Clock

The panel clock from the GLCDC is derived by the frequency-dividing signal from a clock source (PLL) by a value from one to 32.

The frequency of the input clock of the LCD panel mounted on the RSK RX72N is 9 MHz (typ.) to 15 MHz (max.). In this sample program, the input clock is divided by 24 to provide the clock source.

PLL (240 MHz)* / 24 = 10 MHz
(*)PLL = EXTAL (24 MHz) x 10 x 1 = 240 MHz

```
/* Output clock */
gs_glcdc_init_cfg.output.clock_div_ratio = GLCDC_PANEL_CLK_DIVISOR_24;
```

Figure 10-1 Setting the Panel Clock (Divisor to Obtain the Clock Source)

In this sample program, the panel clock that is output runs at 10 MHz.

The user must adjust the horizontal frequency or refresh rate by using QE for Display [RX] since there is a difference of 1 MHz against the typical value for the LCD panel (9 MHz).

10.1.2 Pin Assignments (Bit Endian) and Pixel Order

In the GLCDC, select 'little endian' or 'big endian' for the order of pin assignments and 'RGB' or 'BGR' for the order of colors in pixel data, according to the connection between the MCU and the LCD panel.

The connection between the LCD panel mounted on the RSK RX72N and the MCU is the same as that described in section 6.2, Pin Functions; 'little endian' and 'RGB' are selected as the order of pin assignments (bit endian) and the order of colors in the pixel data, respectively.

The connection described above differs with the output data format. For details on output data formats, refer to the hardware manual for the target MCU.

```
/* Endian */
gs_glcdc_init_cfg.output.endian = GLCDC_ENDIAN_LITTLE;

/* Color order */
gs_glcdc_init_cfg.output.color_order = GLCDC_COLOR_ORDER_RGB;
```

Figure 10-2 Setting Pin Assignments (Bit Endian) and Pixel Order

10.1.3 Other Control Pins

In some cases, control of the backlight or reset may be required according to the connection between the LCD panel and the MCU on the user board.

In the RSK RX72N, as described in section 6.2, Pin Functions, the backlight for and resetting of the LCD panel are controlled by general I/O port pins of the MCU.

```
static void board_port_setting (void)
{
    /* ---- Port setting ---- */
    /* LCD back light and display-on */
    PORT2.PODR.BIT.B7 = 1;  /* Back light */
    PORTK.PODR.BIT.B4 = 1;  /* Display */
    PORT2.PDR.BIT.B7 = 1;
    PORTK.PDR.BIT.B4 = 1;

} /* End of function board_port_setting() */
```

Figure 10-3 Control of Backlight and Resetting

10.2 Creating a Project

10.2.1 Selecting the Smart Configurator

Create a new project for the MCU to be used by the user. Proceed in accord with the displays of the e² studio until the window shown in Figure 10-4 is displayed. Select [Smart Configurator].

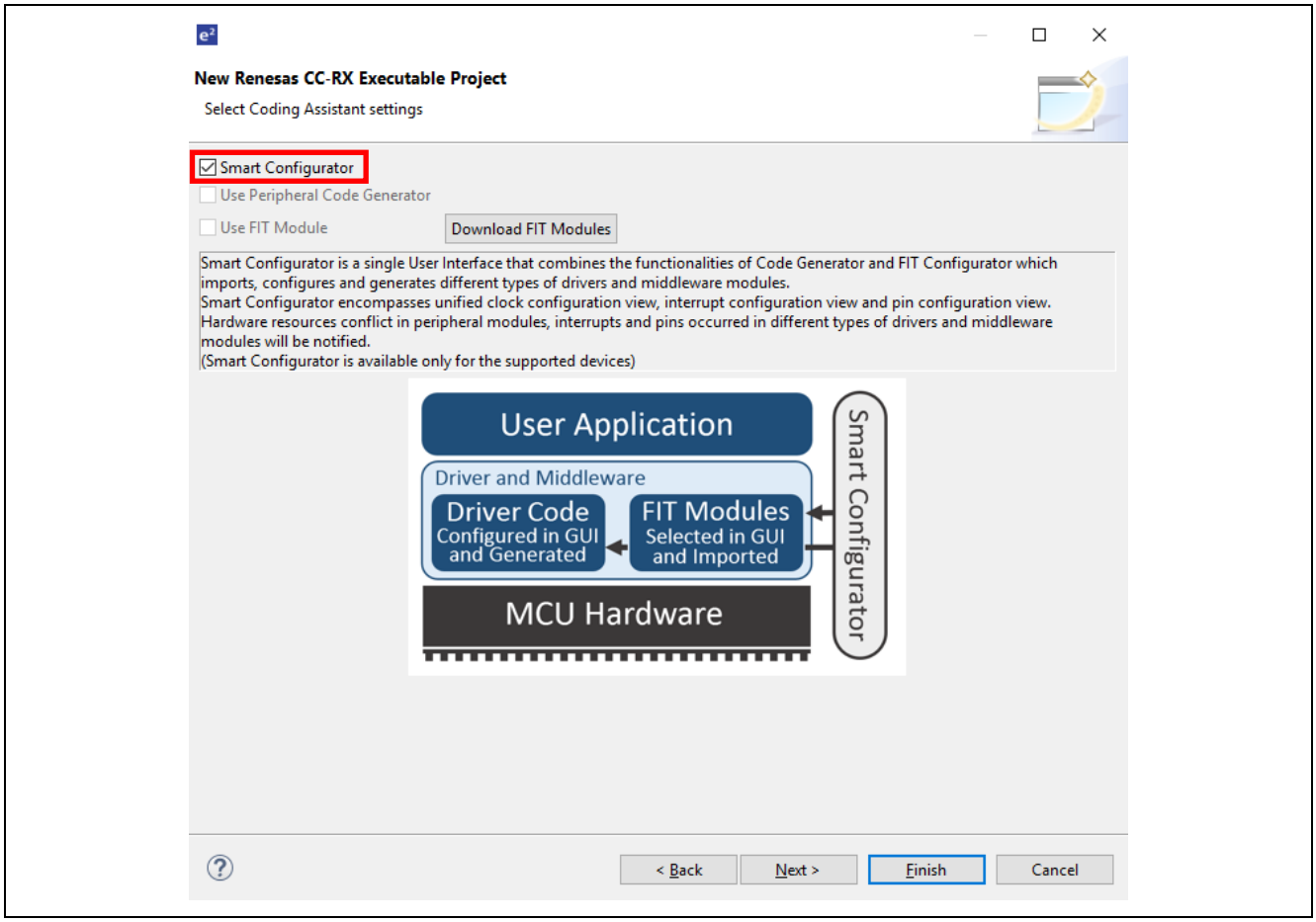


Figure 10-4 [Select Coding Assistant settings] Dialog Box

10.2.2 Setting a Section

Set the area to be used as the frame buffer as a section. Open the properties of the project and add the FRAME_BUFFER section. In the sample program, the initial screen is generated from the area where the FRAME_BUFFER section has been allocated. The address where this section starts is specified as the base address of graphics 2.

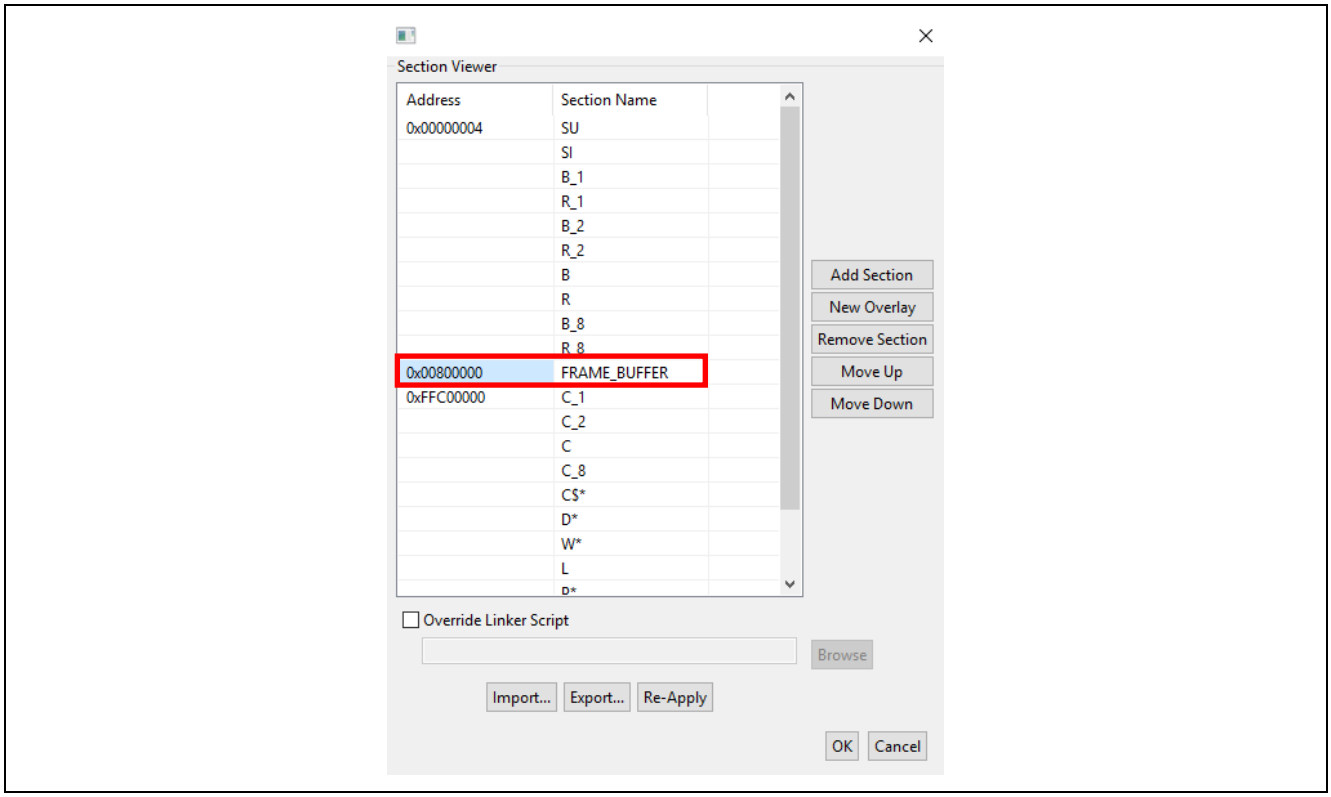


Figure 10-5 Setting a Section

This sample program uses the expanded on-chip RAM area of RX72N devices (refer to section 7.6 , Memory Map). Since the expansion is by 512 Kbytes in the case of the RX72N group, the user may need to consider allocating the area to the on-chip ROM depending on the display size of the LCD panel in the user environment. In such a case, this cannot be used as the initial screen by the sample program since the image data must be downloaded to the on-chip ROM along with the program. Also, downloading to the on-chip ROM is not possible with the image-downloading facility of QE for Display [RX]. For the size of the on-chip ROM, refer to the hardware manual for the target MCU.

10.3 Settings in the Smart Configurator

Use the Smart Configurator to set clocks, components, and pins. When all settings are completed, click on the [Generate Code] button to generate the corresponding code. For details on the Smart Configurator, refer to the user's manual for the Smart Configurator.

10.3.1 Setting a Clock

Make the clock settings on the [Clocks] tabbed page of the Smart Configurator to suit the panel in the user environment. Note that the correct frequency must be set since the PLL is the clock source from which the panel clock is derived.

10.3.2 Adding the GLCDC FIT Module

Add the GLCDC FIT module to the project on the [Components] tabbed page of the Smart Configurator. Click on the [Add component] button and select [GLCDC FIT module (r_glcdc_rx)] from the list of components in the [New Component] dialog box. If you have not already downloaded the module, you can download it from [Download more software components] in this dialog box.

10.3.3 Setting Pins for Use with the GLCDC

When the added GLCD FIT module (r_glcdc_rx) is selected in the [Components] tabbed page of the Smart Configurator, a list of pins available for use with the GLCDC is displayed. Set each pin to suit the connections in the user environment.

After the operations with the use of the [Components], select the [Pins] tabbed page and specify the port and pin numbers to which each pin function is assigned.

In the sample program for the RSK RX72N, settings have been made as described in section 7.2.3, Setting Pins of the GLCDC.

10.4 Adjustment by QE for Display [RX] (Initial Setting)

According to the descriptions in chapter 8, Using QE for Display [RX], and chapter 9, Setting Detailed Data on the LCD Panel, enter information on the LCD panel in the user environment and make the initial adjustments.

Take care that the settings are in the range of the GLCDC FIT module following adjustments. On the [Timing Adjustment] tabbed page of QE for Display [RX], the settings are displayed in red if they violate the panel data and the setting range of the GLCDC that have been registered. Even if the settings are displayed in black, they may still be beyond the range for the GLCDC FIT module.

After you have finished the adjustments, generate the header files.

In the sample program of the RSK RX72N, the header files are generated immediately below the QE_for_Display_sample_RX72N_RSK¥src directory with the following names.

r_lcd_timing_RX72N_RSK.h

r_image_config_RX72N_RSK.h

10.5 Creating a Program

Create a program on the basis of the sample program.

10.5.1 Copying the Sample Program

Delete the '<name of a new project>.c' file which was created when the project was generated and copy the main.c file of the sample program to the newly-created project.

10.5.2 Modifying the Program

Modify the main.c file with reference to section 7.8, Using this Sample Program, and section 10.1, Confirming Specifications, until the program is complete. In this case, you may need to modify the following items.

(a) Names of the header files

Change the header files included in the main.c file in the sample program to those generated by QE for Display [RX].

```
/* Header files for RSKRX72N board output by QE for Display [RX] */
#include "r_image_config_RX72N_RSK.h"
#include "r_lcd_timing_RX72N_RSK.h"
```

Figure 10-6 Including the Header Files

(b) Display size of graphics 2

Change the values of the define directives to suit the display size of the LCD panel to be used. Since these define directives are also used in generating data of the initial screen which is displayed by this sample program, the display size of the initial screen is also changed according to the changes of the defined values (the frame_buffer_initialize function in main.c).

```
/* Image info definition */
#define IMAGE_WIDTH          (480u)  /* Width of image used in this sample. */
#define IMAGE_HEIGHT         (272u)  /* Height of image used in this sample. */

/* ---- Graphic 2 setting ---- */
/* Image format */
gs_glcdc_init_cfg.input[GLCDC_FRAME_LAYER_2].hsize         = IMAGE_WIDTH;
gs_glcdc_init_cfg.input[GLCDC_FRAME_LAYER_2].vsize         = IMAGE_HEIGHT;
```

Figure 10-7 Specifying the Image Size of Graphics 2

(c) Base address of graphics 2

If you will not be using the area of the FRAME_BUFFER section or want to display data allocated to ROM, change the base address of graphics 2 to the address where the area that holds the image data starts.

```
/* ---- Graphic 2 setting ---- */
/* Image format */
gs_glcdc_init_cfg.input[GLCDC_FRAME_LAYER_2].p_base = (uint32_t *) FRAME_BUF_BASE_ADDR;
```

Figure 10-8 Specifying the Base Address of Graphics 2

10.6 From Execution to the End of Adjustment

After the program has been created, start the debugger and execute the program. If the initial screen is not correctly displayed, the settings are not correct. Check the values adjusted by QE for Display [RX] and the settings of parameters of the GLCDC FIT module.

Figure 10-9 shows the flow of troubleshooting.

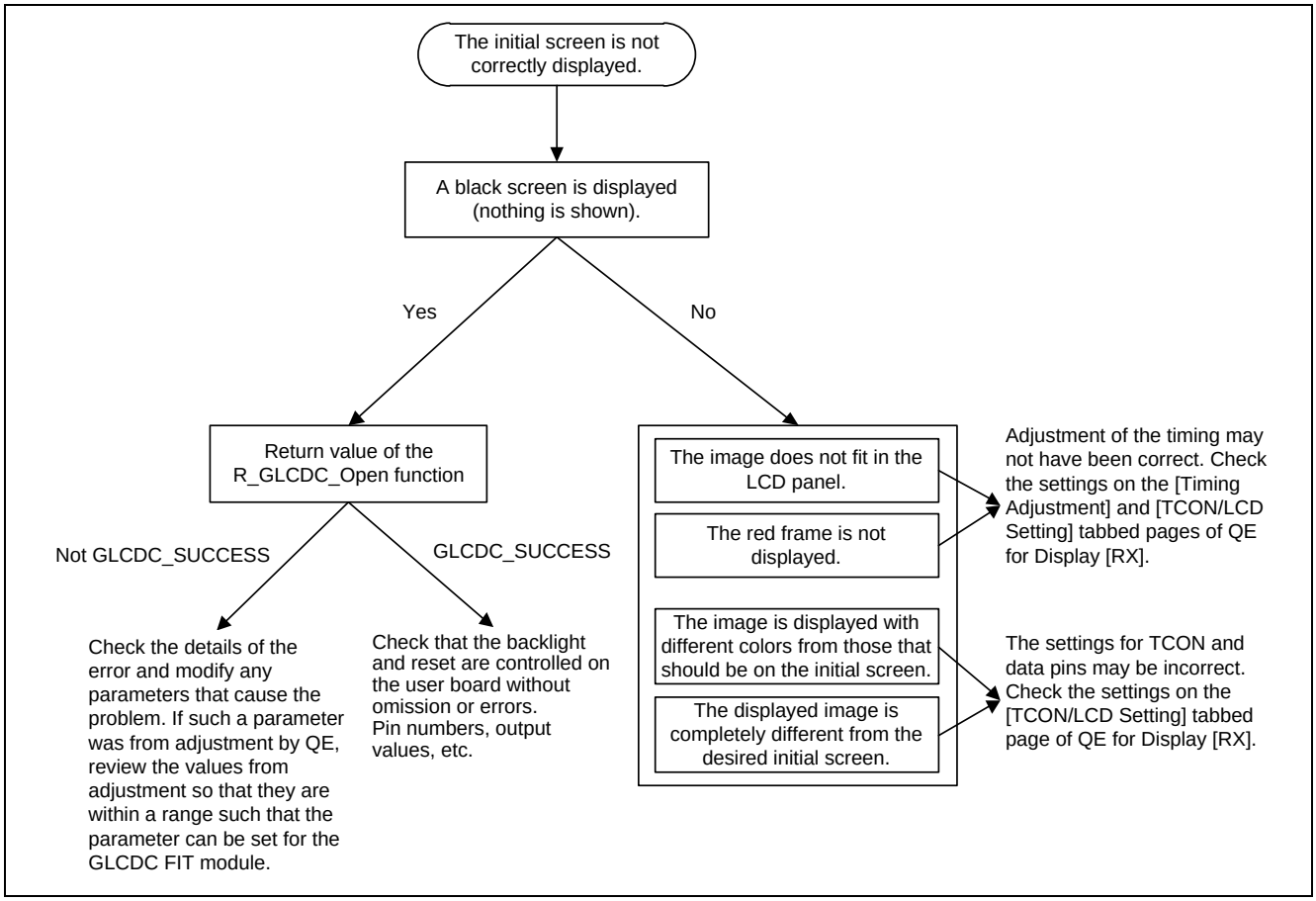


Figure 10-9 Troubleshooting

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Mar. 20, 2018	-	First edition issued.
1.10	Apr. 07, 2020	-	Added support for RSK RX72N, Envision Kit RX72N.
		11	Changed Figure 10-10, Structure of Project Folders.
		14 - 19 40 - 51	Changed the figures to screen using e^2 studio v7.7.0 and Envision RX72N project.
		27	Changed some description of Description column in Table 7-1.
		36	Updated Table 7-3, Sizes of ROM and RAM.
		52 - 55	Changed the figures to screen using e^2 studio v7.7.0.

eneral Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.