RZ/A1H Group

SDK for Camera Sample Program

Introduction

This document describes the functional specification of an SDK for Camera sample program running on the RZ/A1H Software Package that supports Renesas Starter Kit + for RZ/A1H with an RZ/A series RZ/A1H group MCU.

Target Device / Target Board

Target Device : RZ/A1H

Target Board : Renesas Starter Kit+ for RZ/A1H (YR0K77210C000BE)

Contents

List of Abbreviations and Acronyms

Abbreviation	Full Form
CEU	Capture Engine Unit
CMOS	Complementary Metal-Oxide-Semiconductor
CPU	Central Processing Unit
CUI	Character User Interface
DE	Display Enable
DRC	Dynamic Range Compression
GBR	Green Blue Red
GPIO	General Purpose Input / Output
GUI	Graphical User Interface
HS	Horizontal Sync
Hz	Hertz
IIC (or I^2C)	Inter-Integrated Circuit
I/O	Input/Output
IMR-LS2	IMage Renderer – LS2 (RZA1H on-chip peripheral)
JPEG	Joint Photographic Experts Group
KHz	KiloHertz
LCD	Liquid Crystal Display
LTI	Luminance Transient Improvement
LVDS	Low-Voltage Differential Signalling
MCU	MicroController Unit
MHz	MegaHertz
OS	Operating System
PC	Personal Computer
PLL	Phase-Locked Loop
QE	Quick and Effective tool solutions
RAM	Random Access Memory
RGB	Red Green Blue
RGB565	Red Green Blue 16-bit, 5 bits for red and blue, 6 bits for green
RGB666	Red Green Blue 18-bit, 6 bits for each of red, green and blue
RGB888	Red Green Blue 24-bit, 8 bits for each of red, green and blue
RIIC	Renesas Inter-Integrated Circuit
RSK	Renesas Starter Kit
RVAPI	Renesas Video Application Programming Interface
SDK	Software Development Kit
TCON	Timing Controller
TFT	Thin Film Transistor
USB	Universal Serial Bus
VDC5	Video Display Controller 5
VDEC	Video DECoder
VRAM	Video Random Access Memory
VS	Vertical Sync
YCbCr422	Y - Luma (4-bit), Cb - Blue chroma difference (2-bit), Red chroma difference (2-bit)
YUV	Y - Luma, U - blue projection, V - red projection

Table 1-1 List of Abbreviations and Acronyms

1. Overview

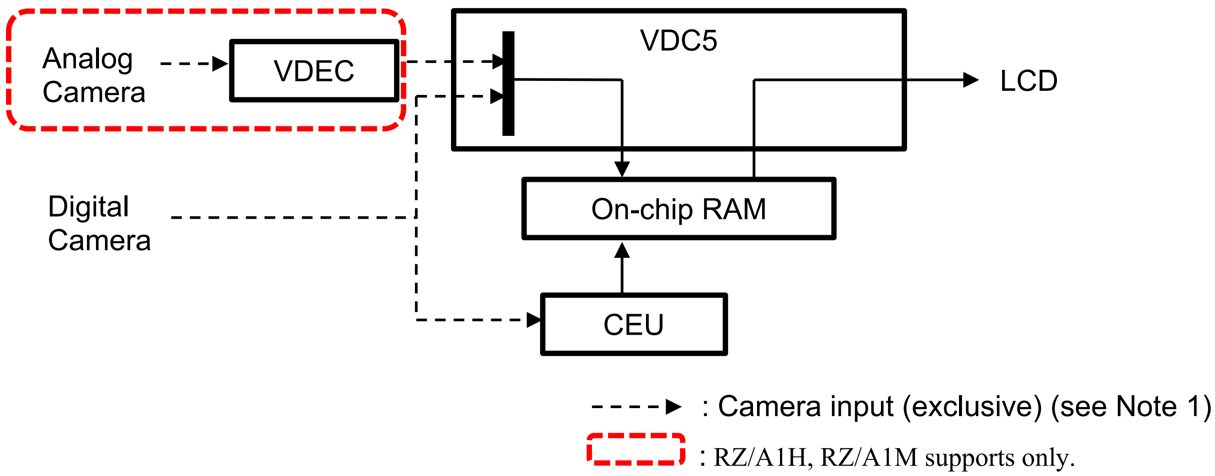
The SDK for Camera is an application program for cameras that functions to capture an image from the camera, adjust various image settings such as brightness, contrast, sharpness, an so on, and display the adjusted image. This sample program allows the image adjustment functionality of the RZ/A1H to be seen in real time, and verify that the adjusted image suits requirements. Once image data has been adjusted, the application software can obtain the configuration data from the sample application. This can then be incorporated into your own image processing software (such as image recognition or JPEG compression for storing JPEG-compressed data).

When using e^2 studio, this sample program can interoperate with the development support tools for RZ **QE for Display** and **QE for Camera**. **QE for Display** enables simple adjustment of image capture and display device parameters. **QE for Camera** enables easy access to set the camera module via RIIC commands containing camera device parameters. Refer to Section 6 for details.

2. Features

2.1 Camera Input

The RZ/A1 supports camera input and has on-chip peripheral devices VDC5, and CEU. The VDC5 and CEU are for digital camera input. The VDC5 also supports image output to display devices. This sample program offers sample applications for camera input which involve using those peripheral devices. For details about these peripheral devices, refer to “RZ/A1H Group, RZ/A1M Group User's Manual: Hardware (R01UH0403)”.



Note 1. The RZ/A1 can support two camera inputs, but this sample program only supports one input.

Figure 2.1 Data Flow for RZ/A1 Camera Input

2.2 Image Adjustment

By using the RZ/A1’s peripheral devices VDC5, this sample program adjusts an image from the camera as shown in Table 2-1. The adjustment results are reflected on the LCD display, allowing the user to observe the adjustment effects in real time. The user can directly change the register values related to each adjustment. The user can test the adjustment values to see if they suit the user’s requirements. For information on the image adjustment methods, refer to Section 5 Image Adjustment Effects and Adjustment Methods. Please refer to Section 6 if adjusting images with **QE for Display**.

Peripheral device	Adjustments	Description
VDC5	Brightness (Note 1)	Adjusts brightness by changing the luminance components
	Contrast (Note 1)	Adjusts contrast by changing the color components
	Sharpness	Sharpens the outline of an edge by adjusting overshoot and undershoot
	Gamma adjustment	Makes a gamma adjustment with a preset value (four types of preset gamma adjustments are enabled with preset values)
	Dithering process	Performs dithering with random patterns
	Rotation and Horizontal Mirroring (Note 2)	Rotate an image 180 degree or flip horizontally

Table 2-1 Image Adjustments Supported by This Sample Program

Note 1. The adjustment timing is limited for camera input through the CEU (see Section 2.4).
Note 2. Not available when using camera input with CEU.

2.3 Applying Image Adjustment Results

With this sample program, it is easy to apply the image adjustment results to its own default settings, so that these settings will be applied each time the program is started. Adjusted parameters can be output to the console in C source code header file format, and then copied over the appropriate header file in the sample source code. For full details, please refer to section 3.4.

Similarly, when adjusting images using the e^2 studio plug-in **QE for Display**, it can also output the adjustment parameters to a header file. Please refer to Section 6 for full details.

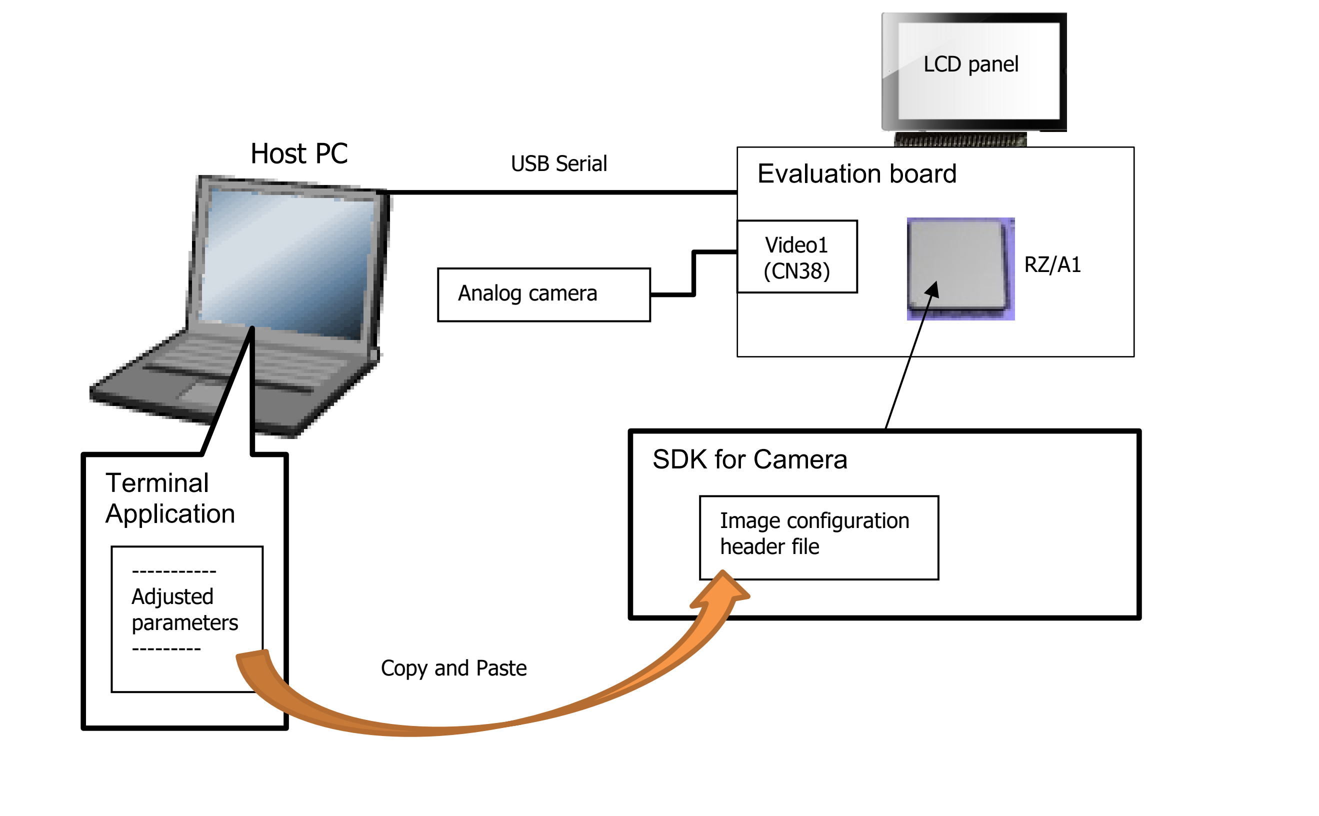


Figure 2.2 Obtaining Image Adjustment Result and Applying to SDK for Camera

2.4 Adjusted-Image Capture

Figure 2.3 shows the processing sequence for each adjusted image.

The image adjustments are all performed by hardware. In this sample program, the software can get an adjusted image at the position in the processing sequence indicated in red in Figure 2.3.

When performing image processing such as recognition processing and JPEG compression for storing JPEG-compressed data, the processing can perform “Brightness 1”, “Contrast 1”, and “Rotation and Horizontal Mirroring”. The result of all of this image processing will be passed on to the “Image quality improver block” and then the “Output controller block”.

However, when performing camera input with the Capture Engine Unit (CEU), “Brightness 1”, “Contrast 1”, “Rotation and Horizontal Mirroring” are not available, and it is also not possible to software capture a processed image at the point indicated in red in Figure 2.3.

For details about the hardware blocks of the VDC5 such as the “Input control block”, “Scaler block” and so on, refer to Section 5.1.

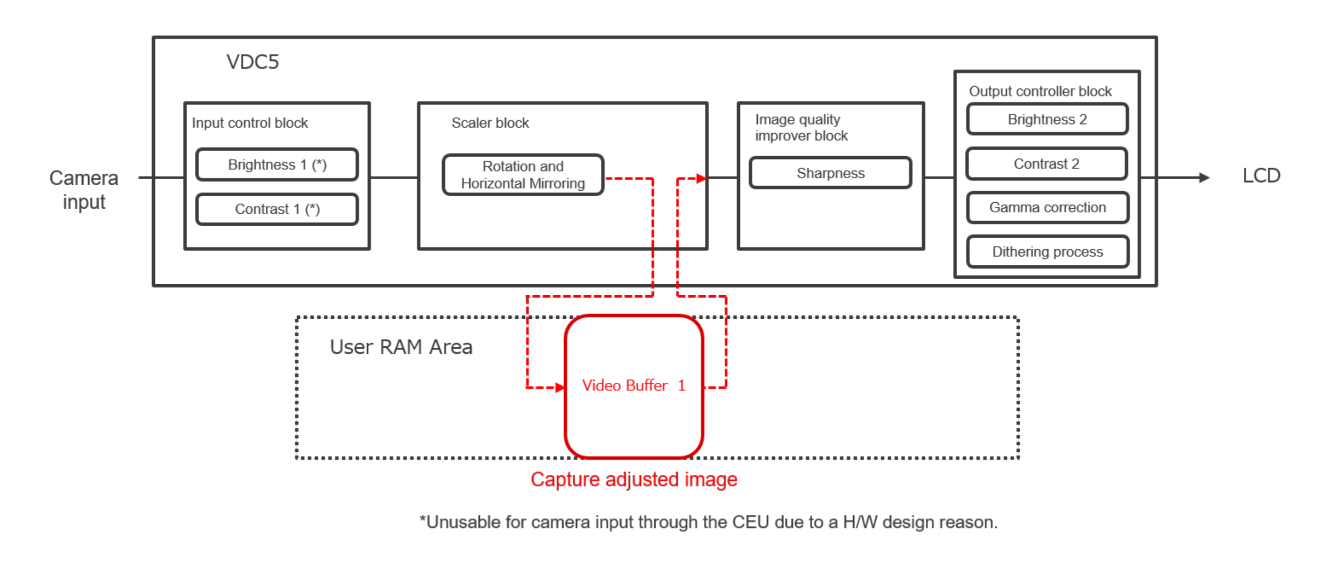


Figure 2.3 Capturing an Adjusted Image

.

3. Software Description

The content described in this chapter is specific for the RZ/A1H.

3.1 Software Blocks

Figure 3-1 SDK for Camera System Block Diagram shows the SDK for Camera system blocks.

This sample program has two tasks, graphics processing task that performs initial setting of camera input, display output, and image adjustment, and CUI (Character User Interface) task for performing image adjustment shown in Table 2-1. For details of the CUI task features, please see Chapter 3.4.

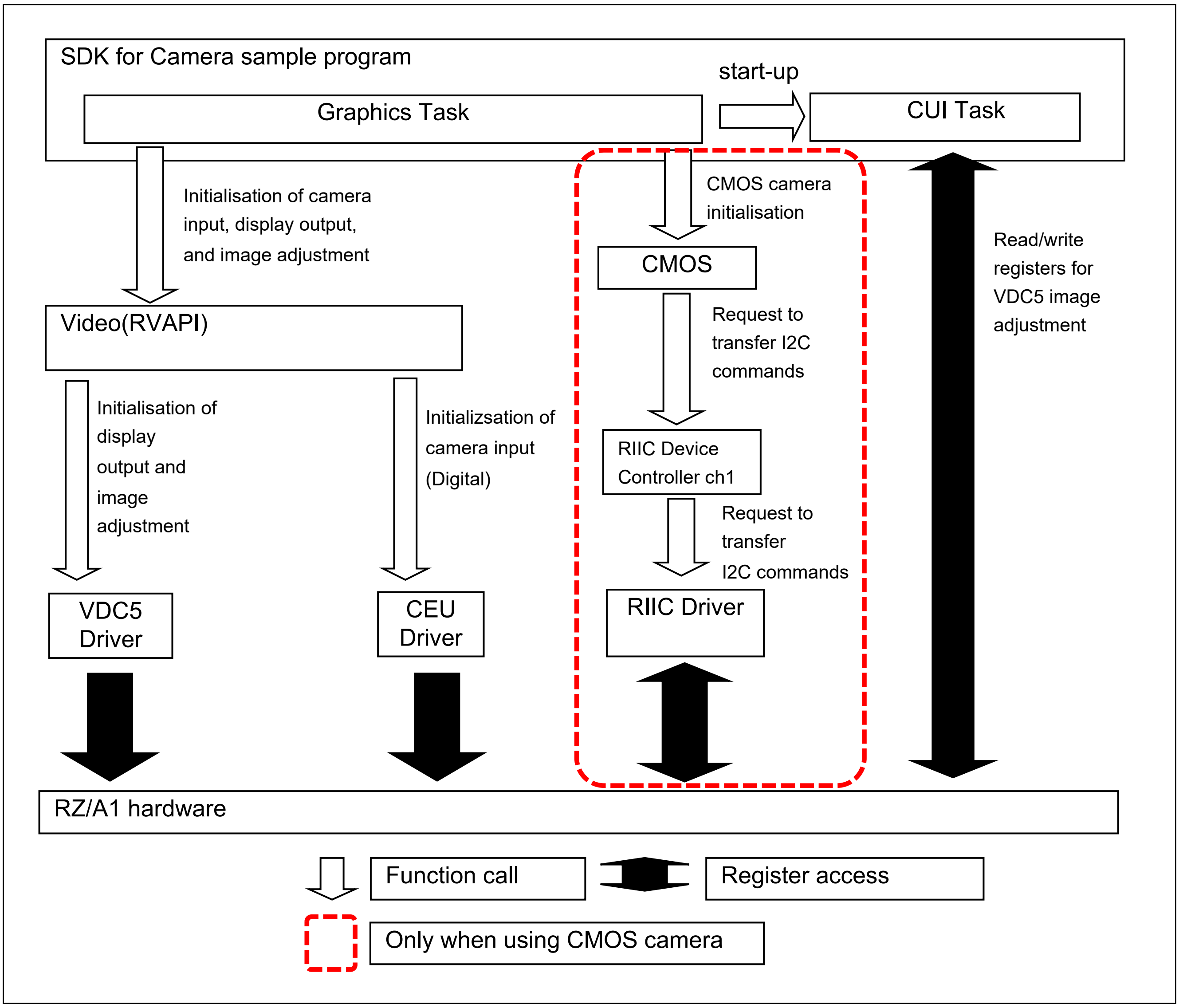


Figure 3-1 SDK for Camera System Block Diagram

3.2 Basic Data Types

Table 3-1 lists the basic data types that are used by the software products included in this package.

The software does not directly use the C language basic data types but uses only the basic data types listed below.

type	type definition
typedef char	char_t
typedef int	int_t
typedef unsigned int	bool_t
typedef signed long	long_t
typedef unsigned long	ulong_t
typedef signed char	int8_t
typedef unsigned char	uint8_t
typedef signed short	int16_t
typedef unsigned short	uint16_t
typedef signed int	int32_t
typedef unsigned int	uint32_t
typedef signed long long	int64_t
typedef unsigned long long	uint64_t
typedef float	float32_t
typedef double	float64_t
typedef long double	float128_t

Table 3-1 Basic Data Types

3.3 Launching SDK for Camera application

Type “sdk” into the RZ/A1H software package command line interface to launch the SDK for Camera application.

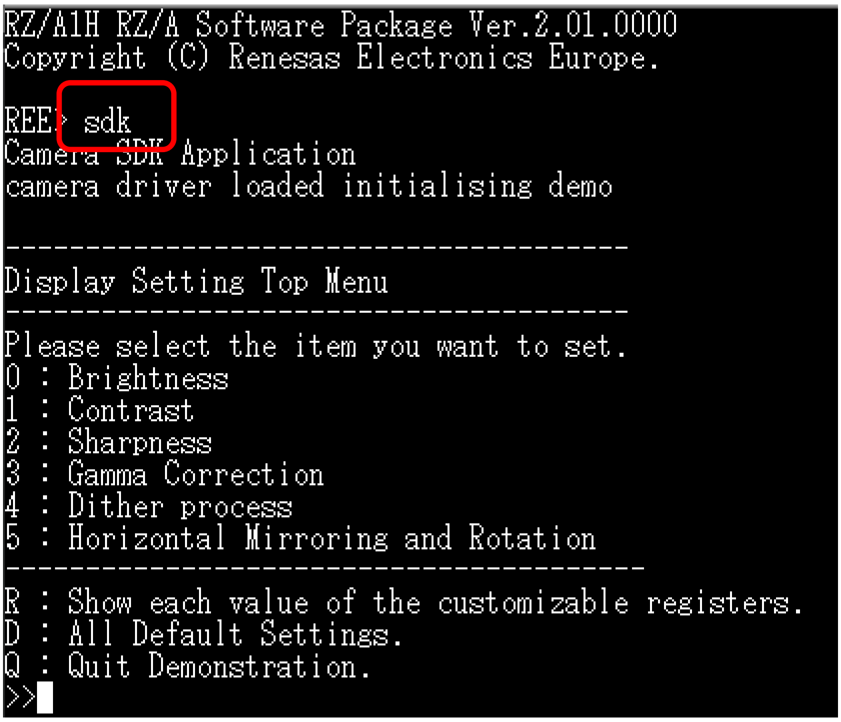


Figure 3-2 Launching the SDK for Camera Application

3.4 CUI (Character User Interface)

This sample program implements a Character User Interface (CUI) which can perform image adjustment in real time via a terminal console application on a PC. The CUI task updates the VDC5 register values by command input from the console to perform image adjustment.

This chapter describes the screen operation method and commands from the console. For details of the various adjustment settings and presets, please refer to Section 5 - **Image Adjustment Effects and Adjustment Methods.**

Note however, that when adjusting images using e^2 studio **QE for Display**, it is not permissable to combine its use with the CUI. Please refer to Section 6 regarding cooperation of this sample program and **QE for Display**.

3.4.1 Menu

The display menu on the console and operation overview is shown in Figure 3-3. Please refer to Chapter 3.4.2 for the details of any of the commands in Figure 3-3.

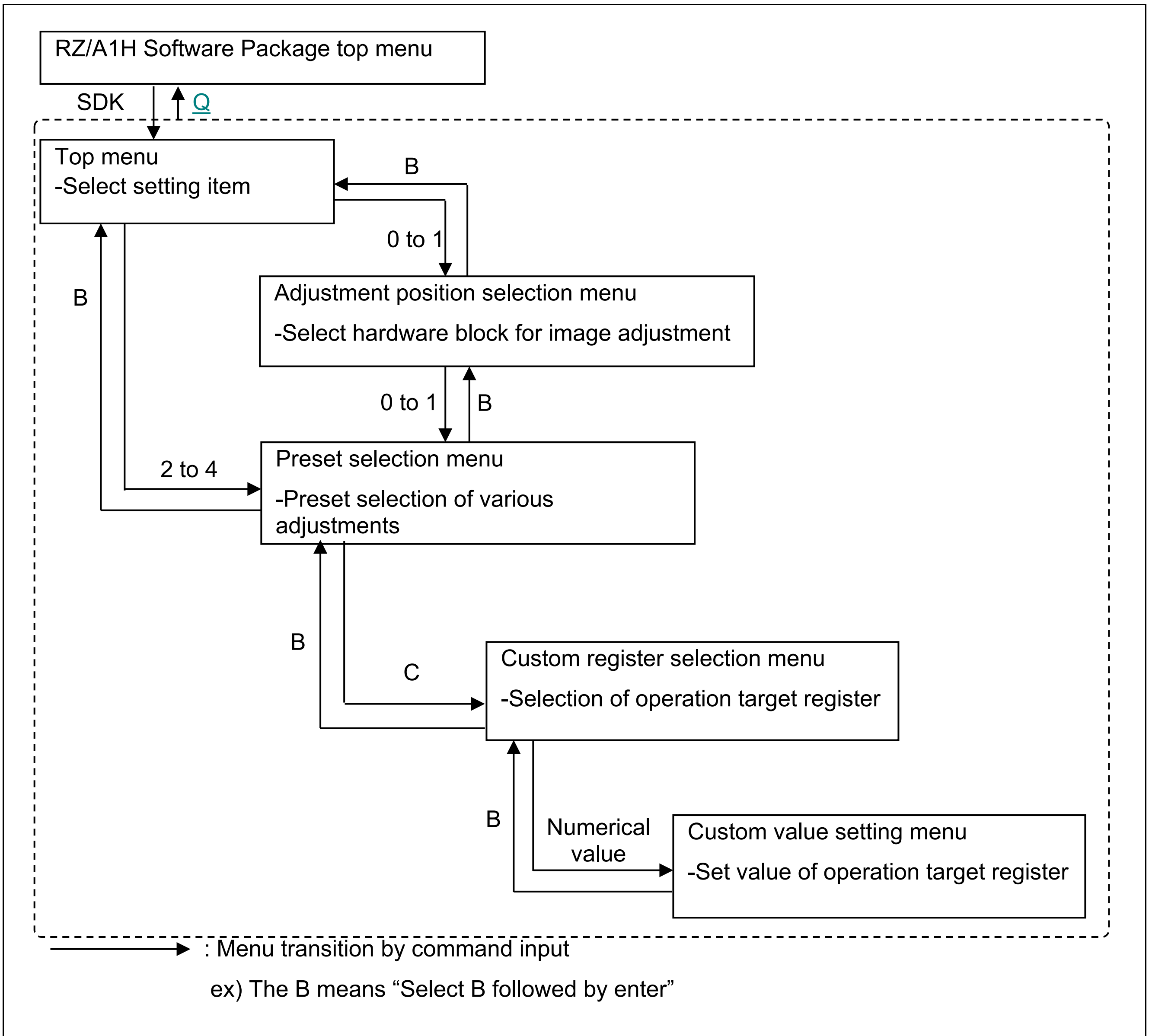


Figure 3-3 Display Menu Operation Overview

3.4.2 Commands

Table 3-2 lists the CUI operation commands on the console.

Command	Operation
Numerical value	Operations in each menu - Selection of image adjustment content - Selection of image adjustment position (selection of hardware block for image adjustment) - Selection of presets - Input custom value
C, c	Custom setting selection (Selected when user want to set a preset other than the various adjustment items)
D, d	Set image adjustment to default (Default value of each register described in hardware manual)
B, b	Return to the previous menu
R, r	Output current image adjustment value
T, t	Return to the Top menu
Q, q	Quit SDK for Camera application
Enter	Determine contents inputted
Delete/ Back space	Delete one character from the input character

Table 3-2 CUI Console Operation Commands

3.4.3 Acquisition of Image Adjustment Values

The user can adjust the camera image settings using the menu options provided by the application. They can choose between one of several presets supplied, or enter a custom value. The current state of all of the settings can be output at any time by entering the 'R' command. When this is done, the current values of the customisable registers are output on the terminal console in a format that can be directly copied and pasted over the source code (the C header file `r_image_config.h`). These settings will become the new defaults after the application is rebuilt and run again.

Image configuration header file path:
`\RZA1H_Sample\src\renesas\application\graphics\inc\r_image_config.h`

4. Processing Sequence

Figure 4.1 shows the processing sequence of this sample program.

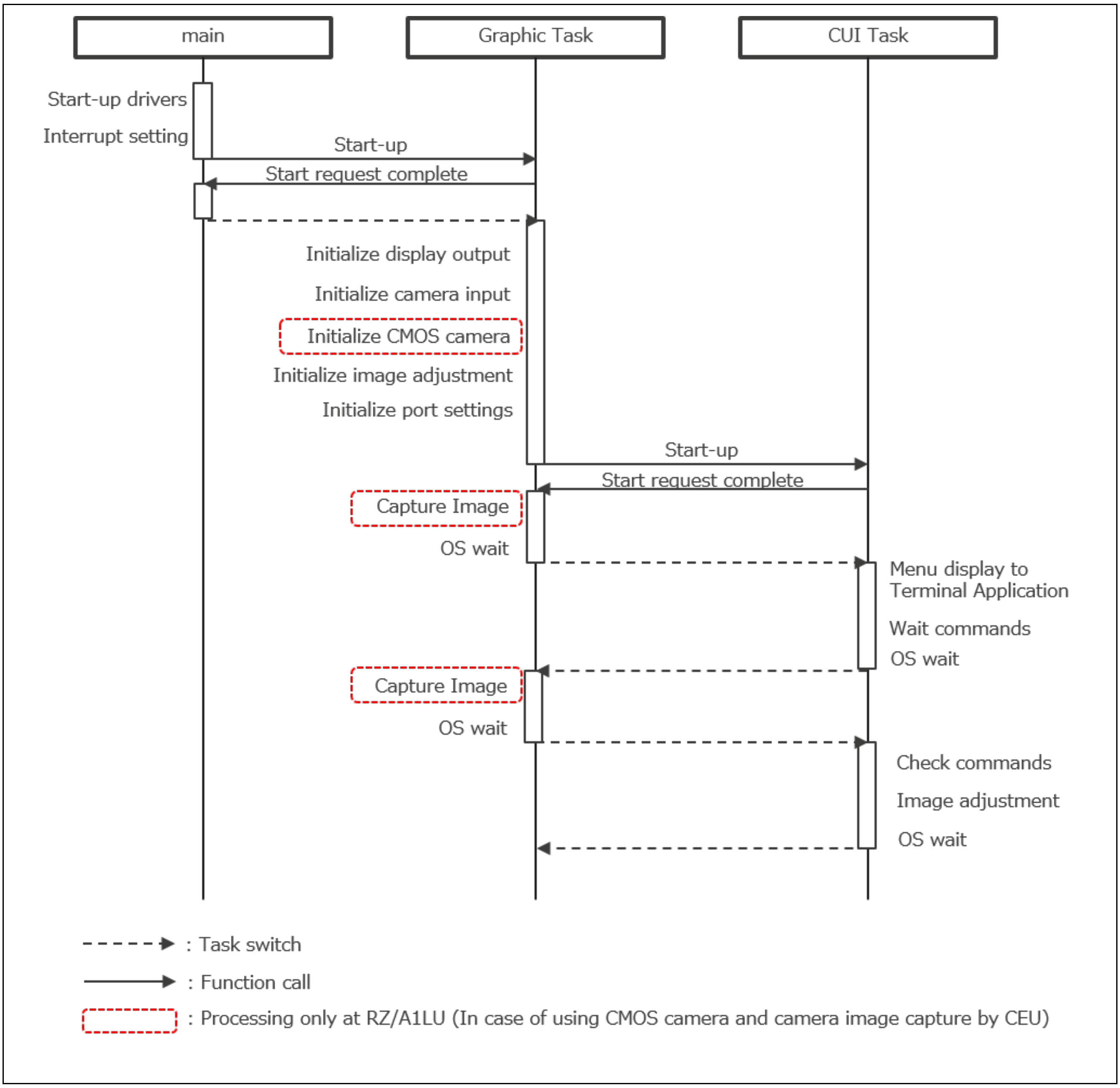


Figure 4.1 SDK for Camera Processing Sequence

5. Image Adjustment Effects and Adjustment Methods

This sample program provides several preset values for many of the image adjustments. This section describes the adjustment effects and preset values. It also shows which blocks in the hardware configuration for RZ/A1 image input/output are responsible for these adjustments.

5.1 Overall Configuration

Figure 5.1 shows the hardware configuration for RZ/A1 image input/output. Table 5-1 lists the functions of the VDC5 blocks.

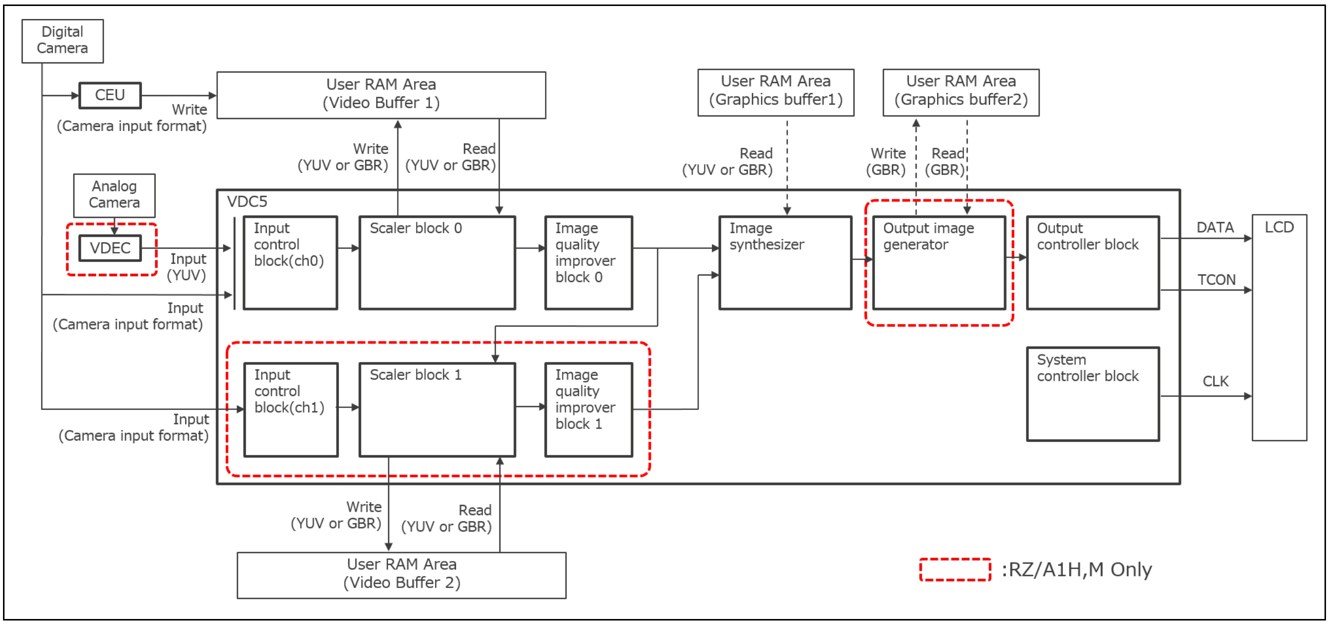


Figure 5.1 Block Diagram of the Hardware Configuration for RZ/A1 Image Input/Output

Block	Function
Input control block (ch0)	Captures an image from the VDEC or digital camera Converts the format through a color matrix and adjusts brightness/contrast Adjusts contrast with the DRC (see Note 1)
Scaler block 0	Writes/reads an image (YCbCr or RGB) to/from the RAM after format conversion Performs zoom in/out (see Note 4) Performs rotation/horizontal inversion Adjusts distortions with the IMR-LS2 (see Note 1)
Image quality improver block 0	Improves sharpness
Image synthesizer	Synthesizes camera images and graphics data stored in the RAM
Output image generator (Note 1)	Writes/reads graphics data (only RGB) to/from the RAM after creation (see Note 2) Adjusts distortions with the IMR-LS2
Output controller block	Outputs the DATA and TCON signals to the LCD Performs brightness/contrast adjustment, gamma adjustment, and dithering
System controller block	Outputs the panel clock signal
Input control block (ch1) (Note 1) (Note 3)	Captures an image from the digital camera Converts the format through a color matrix and adjusts brightness/contrast
Scaler block 1 (Note 1) (Note 3)	Functions the same as scaler block 0 (except that it performs zoom in differently than scaler block 0)
Image quality improver block 1 (Note 1) (Note 3)	Functions the same as image quality improver 0

Table 5-1 Functions of the VDC5 Blocks

[Note 1] This functionality is only provided by the RZ/A1H and RZ/A1M.
[Note 2] This sample code does not support the RAM read/write functionality of the output image generator.
[Note 3] This functionality is not available in this sample program (it is available in ch0 only).
[Note 4] This sample code does not support the VDC5 zoom in/out function.

5.2 Brightness

It is possible to adjust the brightness of the entire image. This makes both light and dark areas of the image lighter.

Brightness can be adjusted with the input control block (ch0) and the output control block of this sample program. If brightness is adusted using the input control block (ch0), then the brightness-adjusted image data is stored in RAM, and this adjusted image can be retrieved by software. If brightness is adjusted by the output controller block, then the brightness-adjusted data is not stored in the RAM, and is only reflected on the display panel.

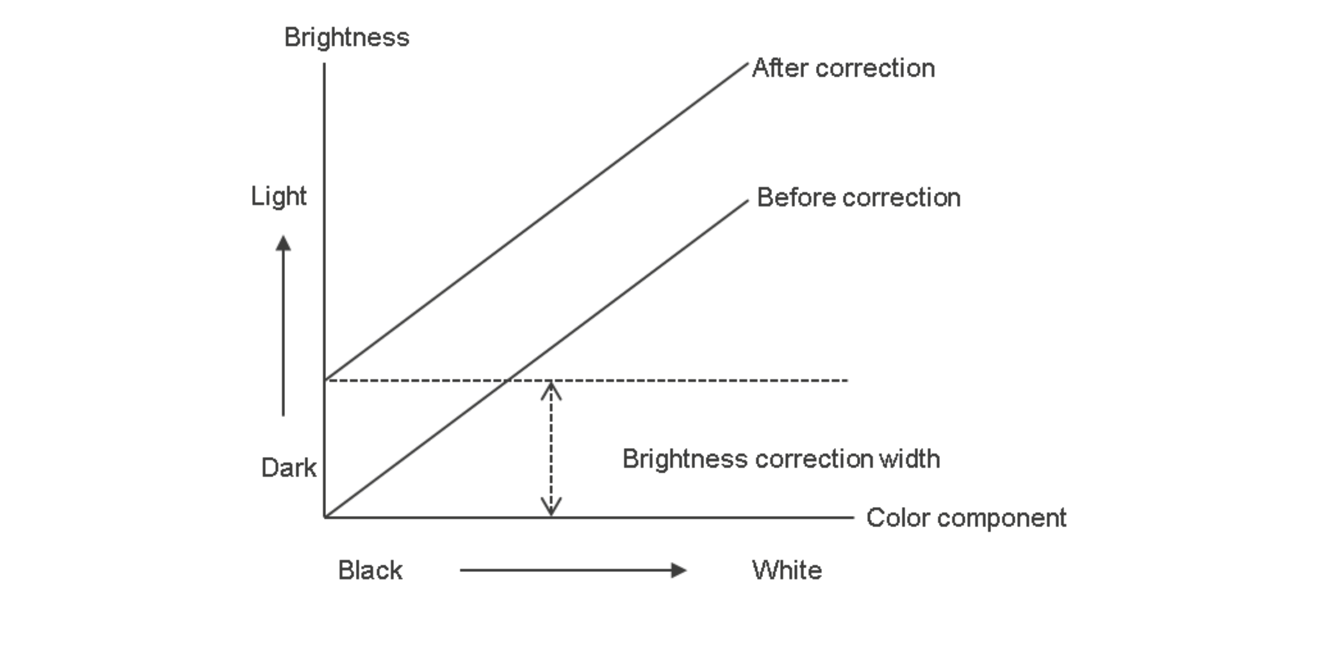


Figure 5.2 Effects of Brightness Adjustment

5.2.1 Preset Brightness Values

This sample program provides the preset brightness values listed in Table 5-2. Presets can be selected for the input control block (ch0) and the output controller block.

Adjustment block	Preset value	Description
Input control block (ch0)	High	Makes the entire screen lighter
	Middle	Makes the entire screen a little bit lighter
	Low	Makes the entire screen darker
Output controller block	High	Makes the entire screen lighter
	Middle	Makes the entire screen a little bit lighter
	Low	Makes the entire screen darker

Table 5-2 Preset Brightness Values

5.2.2 Custom Brightness Settings

Custom brightness settings can be configured using this sample program. Table 5-3 lists target register information.

Adjustment block	Target register	Target bit	Initial value	Setting range	Remarks
Input control block (ch0)	IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG [7:0]	128	0 to 255	Increasing the value makes the screen brighter
Output controller block	OUT_BRIGHT1	PBRT_G[9:0]	512	0 to 1023	Increasing the value makes the screen brighter For brightness adjustment, all three values should be equal
	OUT_BRIGHT2	PBRT_B[9:0]	512	0 to 1023	
	OUT_BRIGHT2	PBRT_R[9:0]	512	0 to 1023	

Table 5-3 Custom Settings for Brightness Adjustment

5.3 Contrast

By increasing the adjustment factor, it is possible to leave a dark part of the screen as it is and make a light part of the screen lighter, thereby emphasizing contrast.

Contrast can be adjusted with the input control block (ch0) and output controller block by this sample program. If the contrast is adjusted using the input control block (ch0), then the contrast-adjusted image data is stored in the RAM, and can by retrieved by software. If the contrast is adjusted using the output controller block, then the contrast-adjusted data is not stored in the RAM, and is only reflected on the display panel.

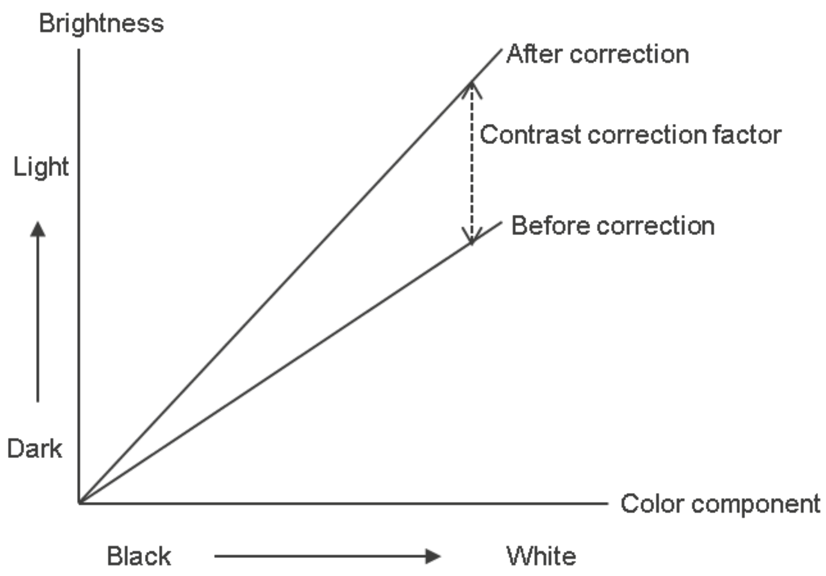


Figure 5.3 Effects of Contrast Adjustment

5.3.1 Preset Contrast Values

This sample program provides the preset contrast values listed in Table 5-4. Presets can be selected for the input control block (ch0) and the output controller block.

Adjustment block	Preset value	Description
Input control block (ch0)	High	Emphasizes the light and shade in the entire screen
	Middle	Slightly emphasizes the light and shade in the entire screen
	Low	Deemphasizes the light and shade in the entire screen
Output controller block	High	Emphasizes the light and shade in the entire screen
	Middle	Slightly emphasizes the light and shade in the entire screen
	Low	Deemphasizes the light and shade in the entire screen

Table 5-4 Preset Contrast Values

5.3.2 Custom Contrast Settings

Custom contrast settings can be configured using this sample program. Table 5-5 lists target register information.

Adjustment block	Target register	Target bit	Initial value	Setting range	Remarks
Input control block (ch0)	IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG[10:0]	256	-1024 to 1023	Increasing the value places greater emphasis on the contrasts between light and shade
Output controller block	OUT_CONTRAST-	CONT_G[7:0]	128	0 to 255	Increasing the value places greater emphasis on the contrast between light and shade For contrast adjustment, all three values should be equal
	OUT_CONTRAST-	CONT_B[7:0]	128	0 to 255	
	OUT_CONTRAST-	CONT_R[7:0]	128	0 to 255	

Table 5-5 Custom Settings for Contrast Adjustment

5.4 Sharpness

By increasing the brightness difference between adjacent pixels (LTI adjustment) and improving the brightness overshoot and undershoot (sharpness), contours can be emphasized.

With this sample program, sharpness can be adjusted by using the image quality improver block. Sharpness-adjusted data is not stored in RAM. It is only reflected in the display panel.

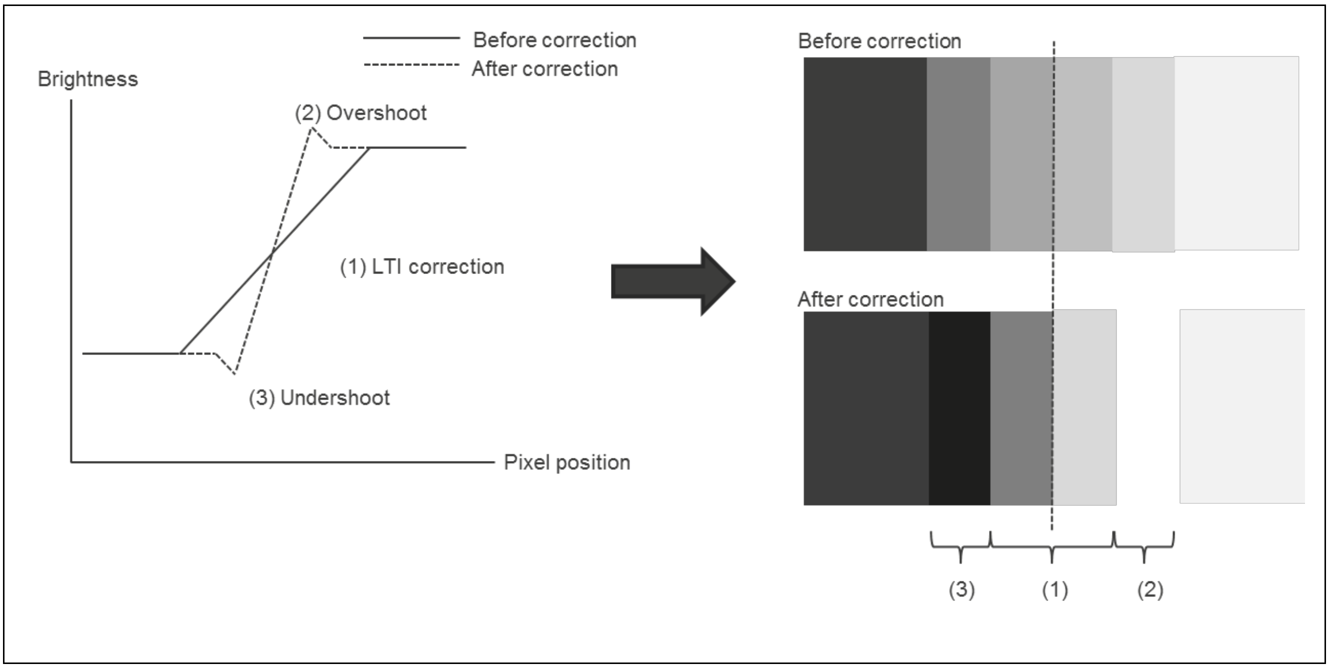


Figure 5.4 Effects of Sharpness Adjustment

5.4.1 Preset Sharpness Values

This sample program provides the values listed in Table 5-6 as preset values of sharpness.

Adjustment block	Preset value	Description
Image quality improver block	Strong	Puts maximum emphasis on the contour
	Semi-strong	Puts medium emphasis on the contour
	Weak	Puts minimum emphasis on the contour

Table 5-6 Preset Sharpness Values

5.4.2 Custom Sharpness Settings

Custom sharpness settings can be configured using this sample program. Table 5-7, Table 5-8, and Table 5-9 list target register information.

Adjustment block	Target register	Target bit	Initial value	Setting range	Remarks
Image quality improver block	ADJ0_ENH_SHP1	SHP_H_ON	0	0 to 1	0: Sharpness ON 1: Sharpness OFF
	ADJ0_ENH_SHP1	SHP_H1_CORE [6:0]	0	0 to 127	Used to specify the range of available sharpness settings for the horizontal sharpness band (H1) The contour is emphasized when the edge amplitude value is greater than or equal to this setting
	ADJ0_ENH_SHP2	SHP_H1_CLIP_O [7:0]	0	0 to 255	Used to adjust the clip value of overshoot for the horizontal sharpness band (H1)
	ADJ0_ENH_SHP2	SHP_H1_CLIP_U [7:0]	0	0 to 255	Used to adjust the clip value of undershoot for the horizontal sharpness band (H1)
	ADJ0_ENH_SHP2	SHP_H1_GAIN_O [7:0]	0	0 to 255	Gain value for the edge amplitude value of overshoot for the horizontal sharpness band (H1) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_SHP2	SHP_H1_GAIN_U [7:0]	0	0 to 255	Gain value for the edge amplitude value of undershoot for the horizontal sharpness band (H1) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_SHP3	SHP_H2_LPF_SEL	0	0 to 1	LPF settings for the horizontal sharpness band (H2) 0: LPF not present 1: LPF present
	ADJ0_ENH_SHP3	SHP_H2_CORE [6:0]	0	0 to 127	Used to specify the range of available sharpness settings for the horizontal sharpness band (H2) The contour is emphasized when the edge amplitude value is greater than or equal to this setting
	ADJ0_ENH_SHP4	SHP_H2_CLIP_O [7:0]	0	0 to 255	Used to adjust the clip value of overshoot for the horizontal sharpness band (H2)

Table 5-7 Custom Settings for Sharpness Adjustment (1/3)

Adjustment block	Target register	Target bit	Initial value	Setting range	Remarks
Image quality improver block	ADJ0_ENH_SHP4	SHP_H2_CLIP_U [7:0]	0	0 to 255	Used to adjust the clip value of undershoot for the horizontal sharpness band (H2)
	ADJ0_ENH_SHP4	SHP_H2_GAIN_O [7:0]	0	0 to 255	Gain value for the edge amplitude value of overshoot for the horizontal sharpness band (H2) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_SHP4	SHP_H2_GAIN_U [7:0]	0	0 to 255	Gain value for the edge amplitude value of undershoot for the horizontal sharpness band (H2) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_SHP5	SHP_H3_CORE [6:0]	0	0 to 127	Used to specify the range of available sharpness settings for the horizontal sharpness band (H3) The contour is emphasized when the edge amplitude value is greater than or equal to this setting
	ADJ0_ENH_SHP6	SHP_H3_CLIP_O [7:0]	0	0 to 255	Used to adjust the clip value of overshoot for the horizontal sharpness band (H3)
	ADJ0_ENH_SHP6	SHP_H3_CLIP_U [7:0]	0	0 to 255	Used to adjust the clip value of undershoot for the horizontal sharpness band (H3)
	ADJ0_ENH_SHP6	SHP_H3_GAIN_O [7:0]	0	0 to 255	Gain value for the edge amplitude value of overshoot for the horizontal sharpness band (H3) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_SHP6	SHP_H3_GAIN_U [7:0]	0	0 to 255	Gain value for the edge amplitude value of undershoot for the horizontal sharpness band (H3) 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_LTI1	LTI_H_ON	0	0 to 1	0: LTI ON 1: LTI OFF
	ADJ0_ENH_LTI1	LTI_H2_LPF_SEL	0	0 to 1	LPF settings for the horizontal LTI (H2) band 0: LPF present 1: LPF not present

Table 5-8 Custom Settings for Sharpness Adjustment (2/3)

Adjust ment block	Target register	Target bit	Initial value	Setting range	Remarks
Image quality improve r block	ADJ0_ENH_LTI1	LTI_H2_INC_ZERO [7:0]	10	0 to 255	LTI adjustment threshold value for the median filter (for noise elimination) LTI is not adjusted if the frequency difference from the adjacent pixel is less than or equal to this setting
	ADJ0_ENH_LTI1	LTI_H2_GAIN [7:0]	0	0 to 255	Gain value for the edge amplitude value of the LTI for the horizontal LTI (H2) band 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_LTI1	LTI_H2_CORE [7:0]	0	0 to 255	Coring for the LTI for the horizontal LTI (H2) band Coring is processed for amplitudes which are less than or equal to this setting
	ADJ0_ENH_LTI2	LTI_H4_MEDIAN_T AP_SEL	0	0 to 1	Used to select a reference pixel for the median filter (for noise elimination) for the horizontal LTI (H4) band 0: References the second adjacent pixel 1: References the first adjacent pixel
	ADJ0_ENH_LTI2	LTI_H4_INC_ZERO [7:0]	10	0 to 255	Adjustment threshold setting for median filter (for noise elimination) for the horizontal LTI (H4) band
	ADJ0_ENH_LTI2	LTI_H4_GAIN [7:0]	0	0 to 255	Gain value for the edge amplitude value of the LTI for the horizontal LTI (H4) band 0 (0x) - 64 (+1x) - 255 (+4x)
	ADJ0_ENH_LTI2	LTI_H4_CORE [7:0]	0	0 to 255	Coring for the LTI for the horizontal LTI (H4) band Coring is processed for amplitudes which are less than or equal to this setting

Table 5-9 Custom Settings for Sharpness Adjustment (3/3)

5.5 Gamma Adjustment

This sample program can apply gamma adjustments shown in Figure 5.5 to 256-tone input signals. The output control block makes gamma adjustments, and the results are only reflected on the display panel.

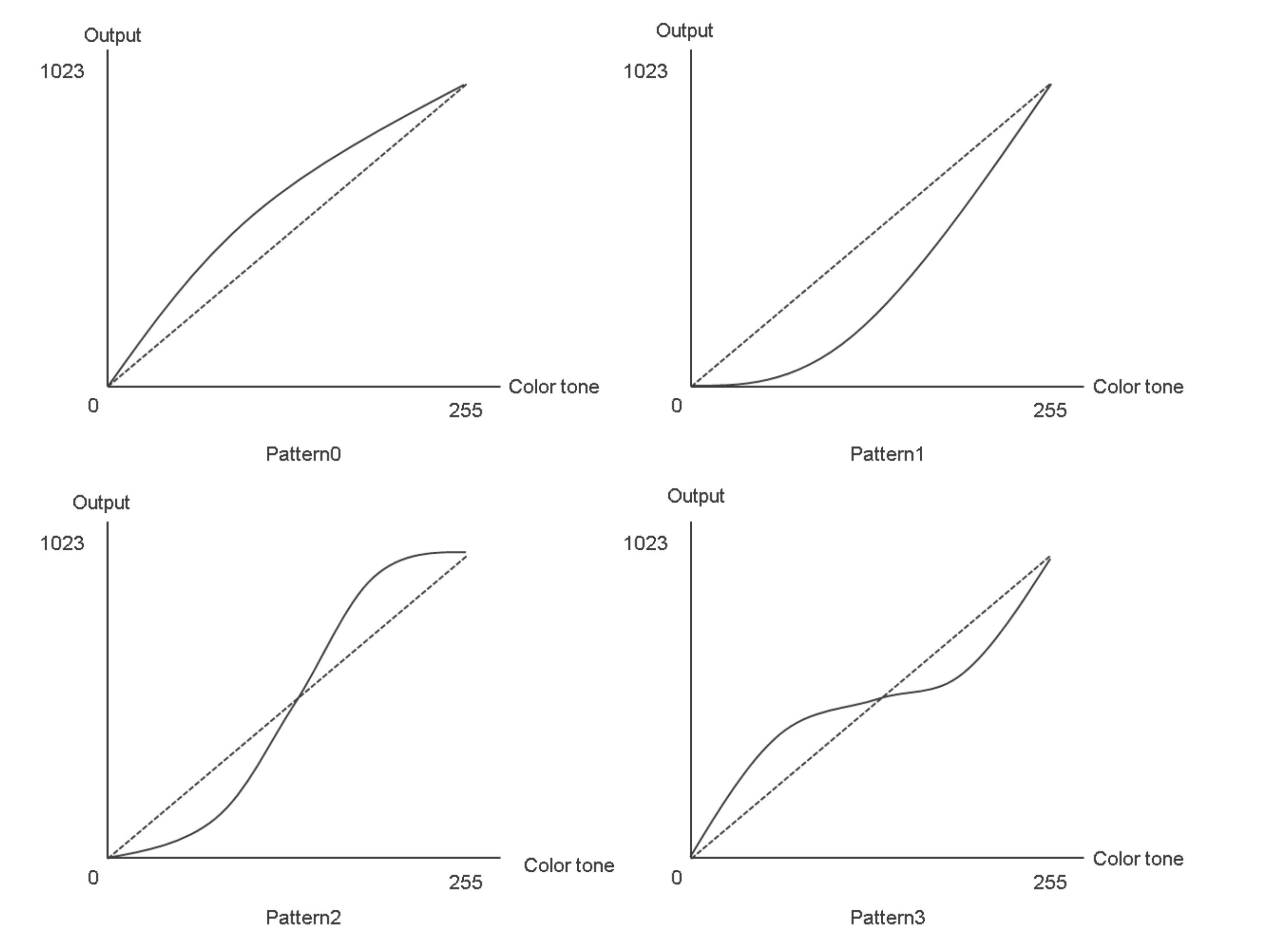


Figure 5.5 Gamma Adjustments Which Can Be Applied by This sample program

5.5.1 Preset Gamma Values

This sample program provides the values listed in Table 5-10 as preset values for gamma adjustments.

Adjustment block	Preset value	Description
Output controller block	Pattern0	Gamma adjustment with γ = 0.81. Makes the entire screen slightly lighter (see Note)
	Pattern1	Gamma adjustment with γ = 1.98. Makes the entire screen slightly darker (see Note)
	Pattern2	Gamma adjustment with an S-shaped curve. Slightly increases contrast
	Pattern3	Gamma adjustment with an inverse S-shaped curve. Slightly decreases contrast

Table 5-10 Preset Values for Gamma Adjustments

Note: Maximum and minimum values which can be set with the RZ/A1.

5.5.2 Custom Gamma Settings

This sample program does not support custom settings for gamma adjustments.

5.6 Dithering Process

The dithering process can prevent color banding. Color banding occurs if the number of bits for processing images with the RZ/A1 exceeds that of bits output to the display panel.

If, for example, image data processed in the RGB888 (24-bit) format is output in the RGB565 (16-bit) format, then the number of tones which can be expressed is reduced. As a result, a smooth gradation pattern might look like a striped pattern. This problem is called color banding. It can be fixed by dithering.

Note that if the number of bits for processing images with the RZ/A1 is less than or equal to that of bits output to the display panel, then dithering is ineffective.

5.6.1 Preset Dithering Values

This sample program provides the mode values listed in Table 5-11 as preset values for dithering.

Adjustment block	Preset value	Description
Output control block	Cutoff	Cutoff mode Truncates the fractional part of the calculation result for bit degeneration
	Round off	Roundoff mode Rounds the calculation result to the nearest integer for bit degeneration
	2x2 Pattern Dither	2x2 pattern dither Adds a pattern value to the figure in the first decimal place and truncates the fractional part of the result for bit degeneration
	Random Pattern Dither	Random pattern dither Adds a random pattern value to the figure in the first decimal place and truncates the fractional part of the result for bit degeneration

Table 5-11 Preset Values for Dithering

5.6.2 Custom Dithering Settings

This sample program does not support custom settings for dithering.

5.7 Rotation and Horizontal Mirroring

Rotation and horizontal mirroring are processed by “scaler block 0” of VDC5 and output the processed result to the camera input buffer (refer to “User RAM Area (Video Buffer 1)” in Figure 5.1).

It is not possible to use rotation and horizontal mirroring at the same time. The possible values are shown in Table 5-13. When setting “90 degrees rotation” or “270 degrees rotation”, the camera input buffer size needs to be changed because the vertical and horizontal dimensions of the image will change. Please note that this function is not available when using camera input with CEU.

5.7.1 Preset Rotation and Mirroring Values

This sample program provides the mode values listed in Table 5-12 as preset values for rotation and horizontal mirroring.

Adjustment block	Preset value	Description
Scaler block 0	Horizontal Mirroring ON	Horizontal mirroring
	Rotation (180 Degrees)	180 degrees Rotation

Table 5-12 Preset Values for Dithering

5.7.2 Custom Rotation and Mirroring Settings

Custom rotation and horizontal mirroring settings can be configured using this sample program. Table 5-13 lists target register information.

Adjustment block	Target register	Target bit	Initial value	Setting range	Remarks
Scaler block 0	SC0_SCL1_WR1	SC0_RES_DS_WR_MD[2:0]	0	0 to 4	0: Normal 1: Horizontal mirroring 2: 90 degrees rotation (Note 1) 3: 180 degrees rotation 4: 270 degrees rotation (Note 1)

Table 5-13 Custom Settings for Rotation and Mirroring

Note 1. The buffer size for camera input needs to be changed if using the “90 degrees rotation” or “270 degrees rotation” setting.

6. Using the Sample Program and QE (Only When Using e^2 studio)

If you use e^2 studio, this sample program can be linked with the development support tool **QE for RZ/A display and Camera**. QE is a plugin for the e^2 studio Integrated Development Environment. **QE for Display** can adjust the timing of the LCD panel via a GUI. The timing settings found using **QE for Display** can then be output as a header file, and used to replace the original sample program header file, thus changing its initial settings.

In a similar fashion, **QE for Camera** can set up the camera module using RIIC commands.

The procedure for applying the result adjusted by **QE for Display** to this sample program is described in this section.

To get instructions for installing e^2 studio and **QE for Display** or **QE for Camera**, please refer to the manuals for each of these tools.

Product page:

QE for Display: https://www.renesas.com/qe-display

QE for Camera: https://www.renesas.com/qe-camera

6.1 Launching QE for Display

From the e^2 studio menu select **Renesas Views** → **Renesas QE** → **Camera/Display Tuning RZ(QE)** to launch **QE for Display**. First, please change the setting **LCD Maker/Type** and **Device** (in the “Block Image” tab) as shown in Figure 6.1 .

The block diagram shown in Figure 6.1 indicates hardware block diagram of VDC5 and shows the image data flow between the camera input and the display output. It also shows the positional relationship of the various image correction blocks.

Various image corrections can be applied after clicking buttons such as Brightness, Contrast, Sharpness, and so on. The tab will move to the “Image Quality Adjustment” tab after clicking image correction buttons. For details of the image adjustment settings, please refer to Section 6.5.2.

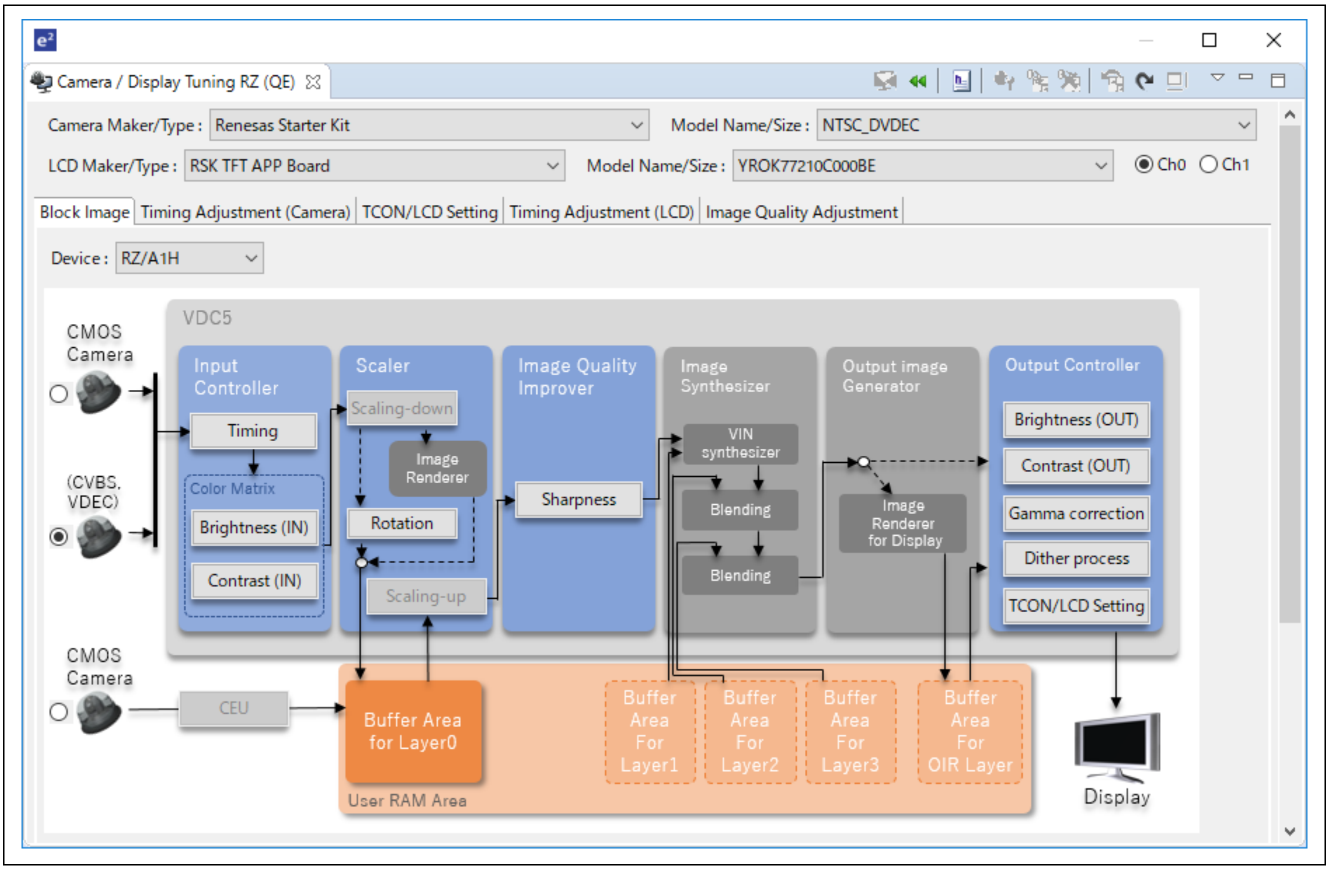


Figure 6.1 QE for Display Launch Screen

6.2 Display Device Information Settings

Select **Custom** from the **LCD Maker/Type** pull-down list at the top of the dialog box shown in Figure 6.1 to display the **Edit Custom Display Data** dialog box (Figure 6.2). The display device information should be entered into this dialog box.

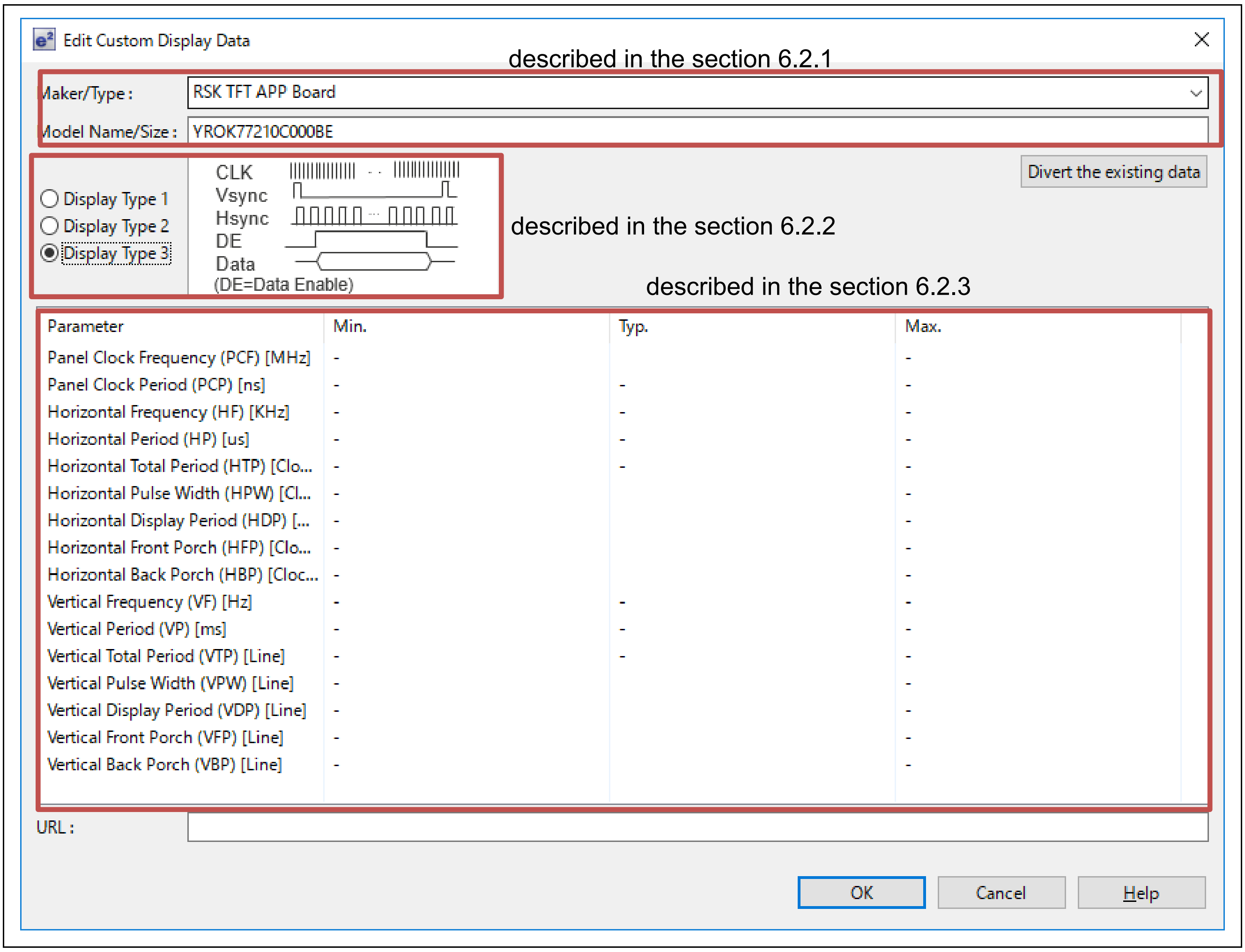


Figure 6.2 Edit Custom Display Data Dialog Box

6.2.1 Entering a Registered Name

Enter any name you wish in the **Maker/Type** and Model/Size fields of the **Edit Custom Display Data** dialog box (Figure 6.3). This name will be registered in the drop-down list and will become selectable.

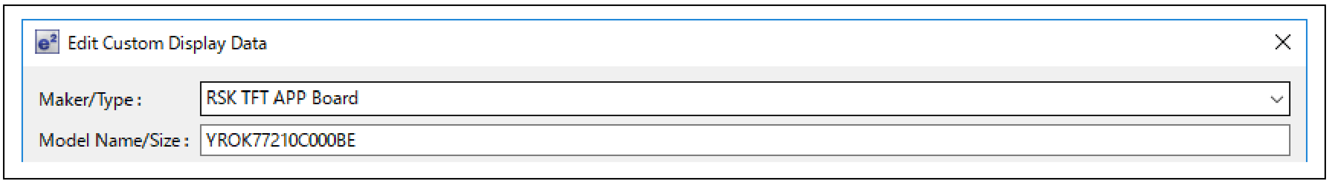


Figure 6.3 Name Registration

6.2.2 Selecting the Display Type

Table 6.1 lists the main control signals required when connecting a display device. Presently, three display device types combining these control signals are supported by **QE for Display**.

Name	Function Summary
Horizontal sync signal (Hsync)	Signal that creates timing for displaying one line
Vertical sync signal (Vsync)	Signal that creates timing for displaying one screen
Panel clock (CLK)	Signal that use to sample for displaying one pixel
Display enable (DE)	Signal indicating that valid data is being output
Data (Data)	Display data

Table 6.1 Main Control Signals

The user must select from the three display types listed in Table 6.2 to determine which control signals are required by the specifications of the display device used.

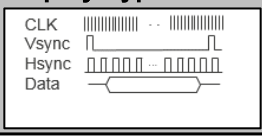
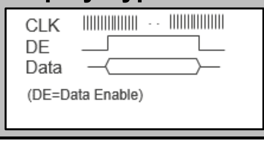
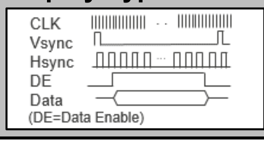
Name	Display Type 1 	Display Type 2 	Display Type 3 
Horizontal sync signal (Hsync)	Used	Not used	Used
Vertical sync signal (Vsync)	Used	Not used	Used
Panel clock (CLK)	Used	Used	Used
Display enable (DE)	Not used	Used	Used
Data (Data)	Used	Used	Used

Table 6.2 Control Signals Used

Figure 6.4 and Figure 6.5 show horizontal and vertical input timing charts for the RSK TFT APP BOARD. The Horizontal Sync (Hsync) signal, Vertical Sync (Vsync) signal, and Display Enable (DE) signal shown in these timing charts are required, so select display type 3 (Figure 6.6).

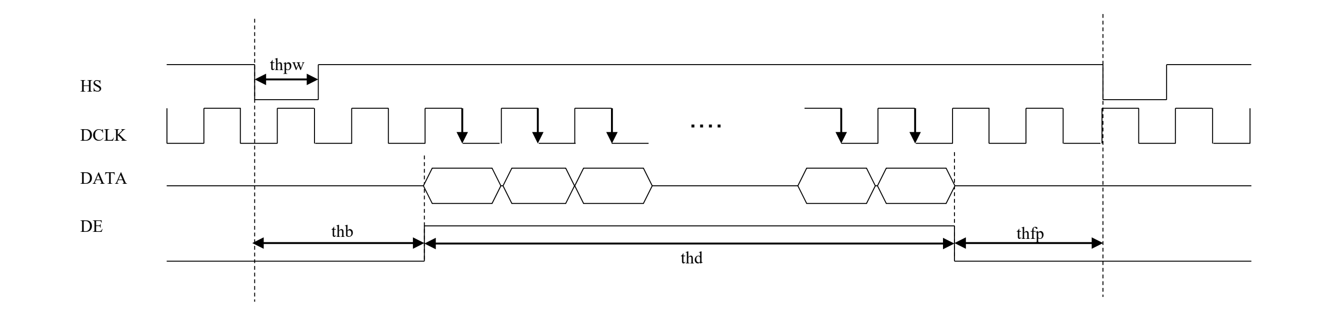


Figure 6.4 Horizontal Input Timing Diagram

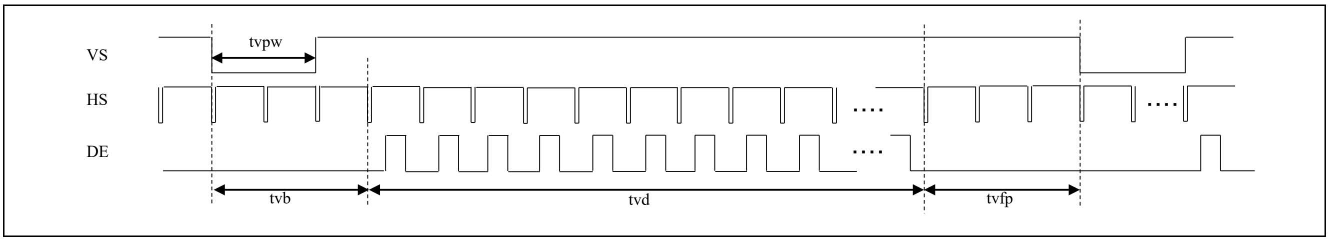


Figure 6.5 Vertical Input Timing Diagram

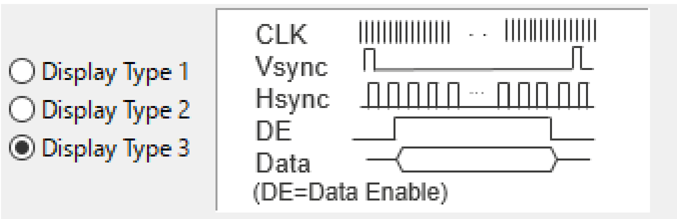


Figure 6.6 Display Type Selection

6.2.3 Entering Control Timing Values

Figure 6.4 and Figure 6.5 show horizontal and vertical input timing charts for the RSK TFT APP BOARD used in this example. Table 6.3 and Table 6.4 list the input timing of the horizontal and vertical sync signals. After entering these control timing values, the result is as shown in Figure 6.7. The values input under **Typ** are used for timing control, and the values input under **Min** and **Max** are used to confirm that values are within the allowable range when the GUI is used for timing adjustment in **QE for Display**.

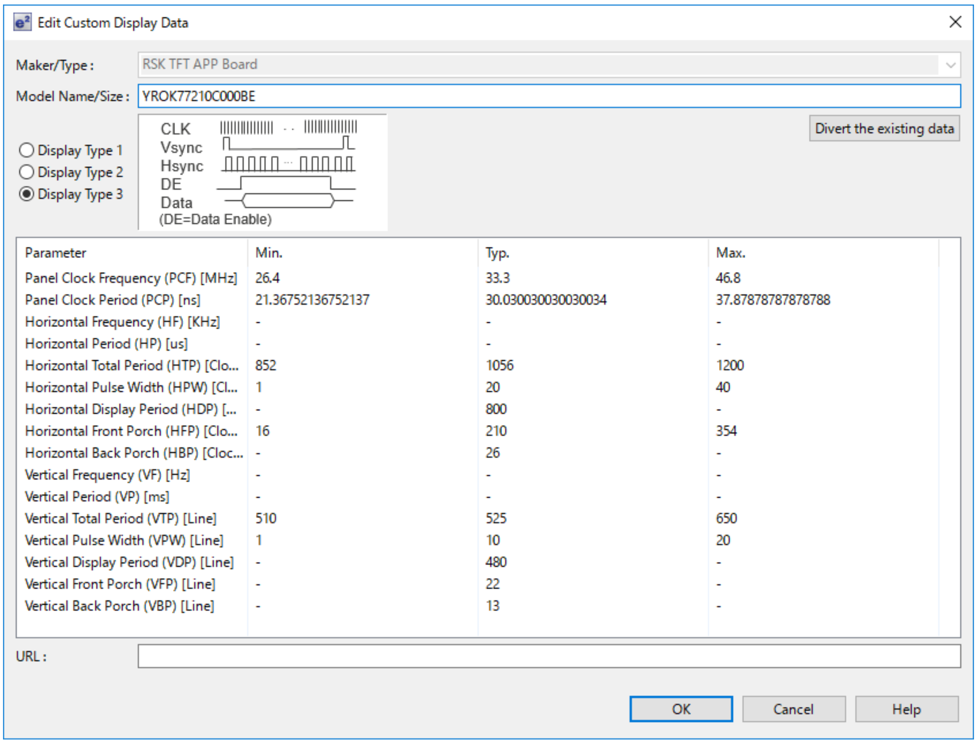


Figure 6.7 Result from Entering Control Timing Values

Refer to Table 6.3 when entering the following items:

- Panel Clock Frequency
- Horizontal Total Period
- Horizontal Pulse Width
- Horizontal Display Period
- Horizontal Front Porch
- Horizontal Back Porch

Refer to Table 6.4 when entering the following items:

- Vertical Total Period
- Vertical Pulse Width
- Vertical Display Period
- Vertical Front Porch
- Vertical Back Porch

Note: In Figure 6.7 the **Typ** values for Horizontal Pulse Width and Vertical Pulse Width are midway between the **Min** and **Max** values. Also, the values listed in Table 6.3 and Table 6.4 should be input for HS Blanking and VS Blanking, but since these HS Blanking and VS Blanking values include the HS pulse width and VS pulse width, respectively, the **Typ** values for Horizontal Pulse Width and Vertical Pulse Width are subtracted and the resulting values are input as the **Typ** values for Horizontal Back Porch and Vertical Back Porch, respectively.

Item	Symbol	Value			Unit
		Min.	Typ.	Max.	
Horizontal display area	thd	—	800	—	DCLK
DCLK frequency	fclk	26.4	33.3	46.8	MHz
One horizontal line	th	852	1056	1200	DCLK
HS pulse width	thpw	1	—	40	DCLK
HS blanking	thb	46	46	46	DCLK
HS front porch	thpf	16	210	354	DCLK

Table 6.3 Horizontal Input Timing

Item	Symbol	Value			Unit
		Min.	Typ.	Max.	
Vertical display area	tvd	—	480	—	TH
VS period time	Tv	510	525	650	TH
VS pulse width	tvpw	1	—	20	TH
VS blanking	tvb	23	23	23	TH
VS front porch	tvpf	7	22	147	TH

Table 6.4 Vertical Input Timing

6.2.4 Control Signal Output Settings

Control signal output can be set after clicking the “TCON/LCD Setting” tab in **QE for Display** (Figure 6.8).

In this dialog box, make the output settings for the control signals listed below.

[Panel driver signal (TCON) output selection]

- Output pin selection:
  Output to pins LCD_TCON0 to LCD_TCON6 (TCON0 to TCON6)
- Control signal polarity:
  Positive polarity (active high)
  Negative polarity (active low)

[LCD setting]

- Output data format selection:
  24-bit RGB888 output (24-bit (VDC5_LCD_OUTFORMAT_RGB888))
  18-bit RGB666 output (18-bit (VDC5_LCD_OUTFORMAT_RGB666))
  16-bit RGB565 output (16-bit (VDC5_LCD_OUTFORMAT_RGB565))
- Data output timing:
  Output at rising edge of panel clock (Rising (VDC5_EDGE_RISING))
  Output at falling edge of panel clock (Falling (VDC5_EFGE_FALLING))

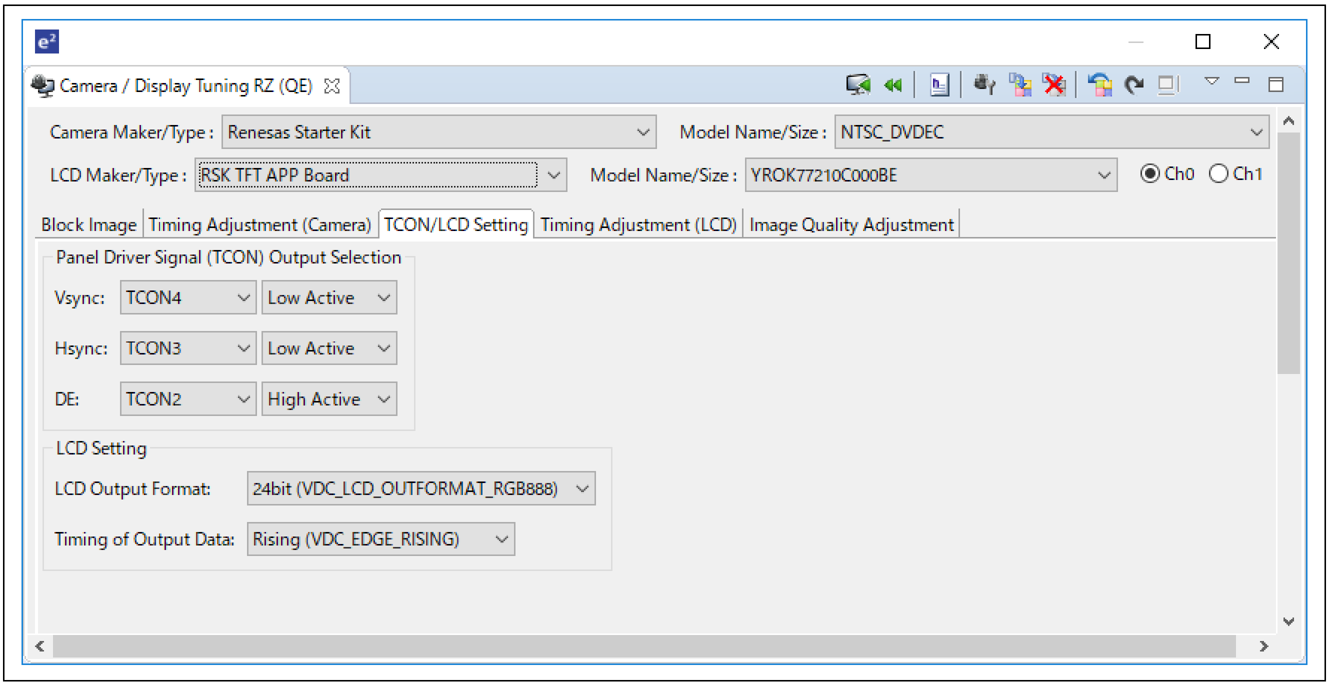


Figure 6.8 TCON and LCD Settings

The output pin selection items matching the specifications of the RSK TFT APP BOARD used in the present case are listed below.

- Output pin selection
  - Vsync: TCON4
  - Hsync: TCON3
  - DE: TCON2

Based on Figure 6.4 and Figure 6.5, the polarity of the control signals is listed below.

- Polarity of the control signals
  - Vsync: Negative polarity (active low)
  - Hsync: Negative polarity (active low)
  - DE: Positive polarity (active high)

The output data format matching the board specifications is listed below.

- Output data format selection:
  16-bit RGB565 output (16-bit (VDC5_LCD_OUTFORMAT_RGB565))

Regarding the data output timing, since sampling occurs at the falling edge of the DCLK signal, according to Figure 6.4, the timing of data output on the VDC5 side is at the rising edge of the panel clock.

- Data output timing:
  Output at rising edge of panel clock (Rising (VDC5_EDGE_RISING))

This completes the display device information settings.

6.3 Generating a Header File for Timing Settings of Display

A header file reflecting the control timing settings can be generated by clicking the generate header file icon (Figure 6.9) in **QE for Display**.

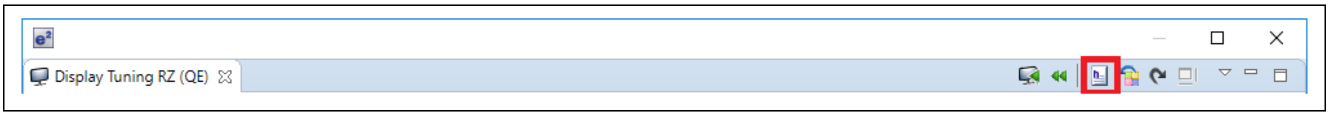


Figure 6.9 Icon for Generating a Header File

In the “Select the Header File” dialog box the header file name and output destination can be specified. Then, after selecting only the “For Timing and TCON Settings” checkbox, click the “Generate” button to generate the header file.

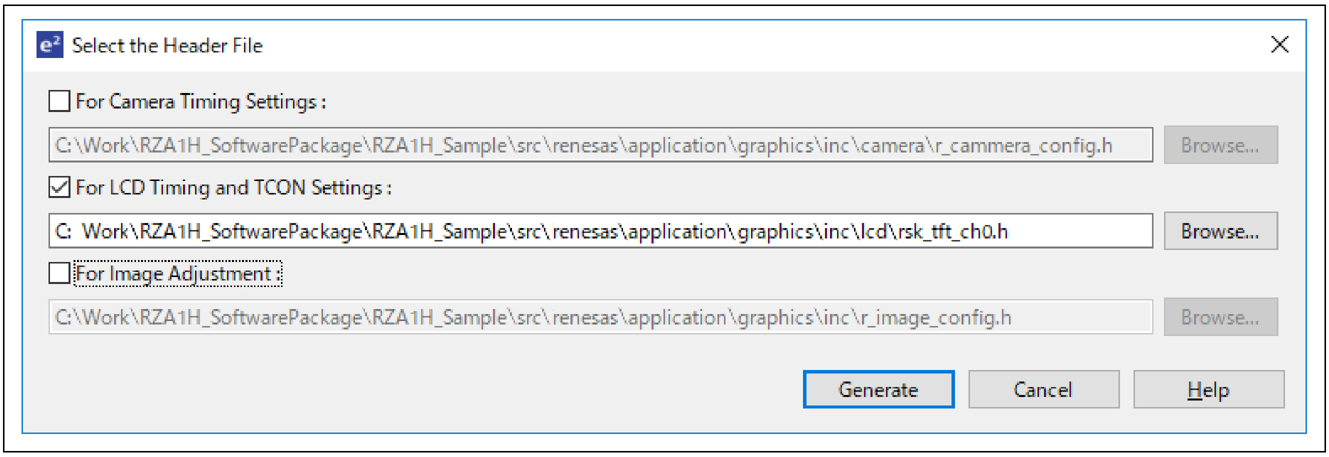


Figure 6.10 Generating a Header File for Timing and TCON Settings

The sample program references the generated header file and makes settings to VDC5. Save the header file using the folder name and file name shown below.

[Folder]

- `RZA1H_Sample\src\renesas\application\graphics\inc\lcd`

[File name]

- In case of Renesas Starter Kit + for RZ/A1H: `rsktft_ch0.h`

6.4 Apply the Generated Header File

After setting the display device information on the **QE for Display** and generating the header file, you can apply the settings in the generated header file with a project “clean” and “build”.

6.5 Real-Time Adjustment Function Using QE for Display

6.5.1 Adjusting the Position of Display

After connecting the debugger and launching the sample program, you can alter the control signal timing by changing the setting values shown in Figure 6.11. Make adjustments while viewing the results on the connected display device, then re-output the header file. Please refer to Section 6.3 regarding the process for generating a header file.

When the adjustment value is shown in red, it indicates that the value is outside VDC5 specifications. In this case, adjust it until the value comes within the VDC5 specification. The range of the value permitted by the specification can be seen by pointing the mouse to the adjustment value shown in red.

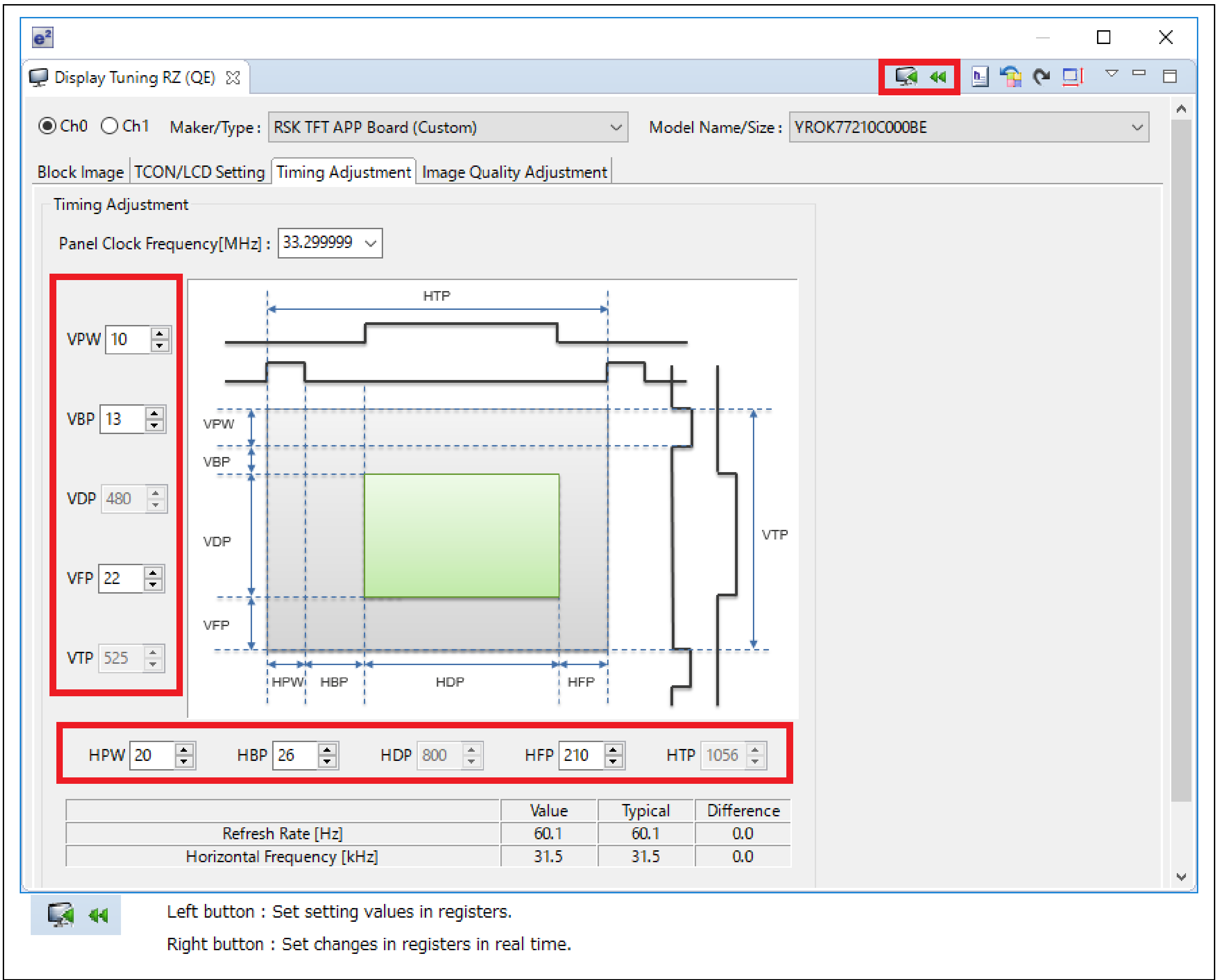


Figure 6.11 Debugging the Control Signal Timing

6.5.2 Various Image Quality Adjustments

Image Quality Adjustment

After connecting the debugger and launching the sample program, you can change various image adjustment settings by clicking on any of the image adjustment buttons indicated by a red outline in the “Block Image” tab in Figure 6.12.

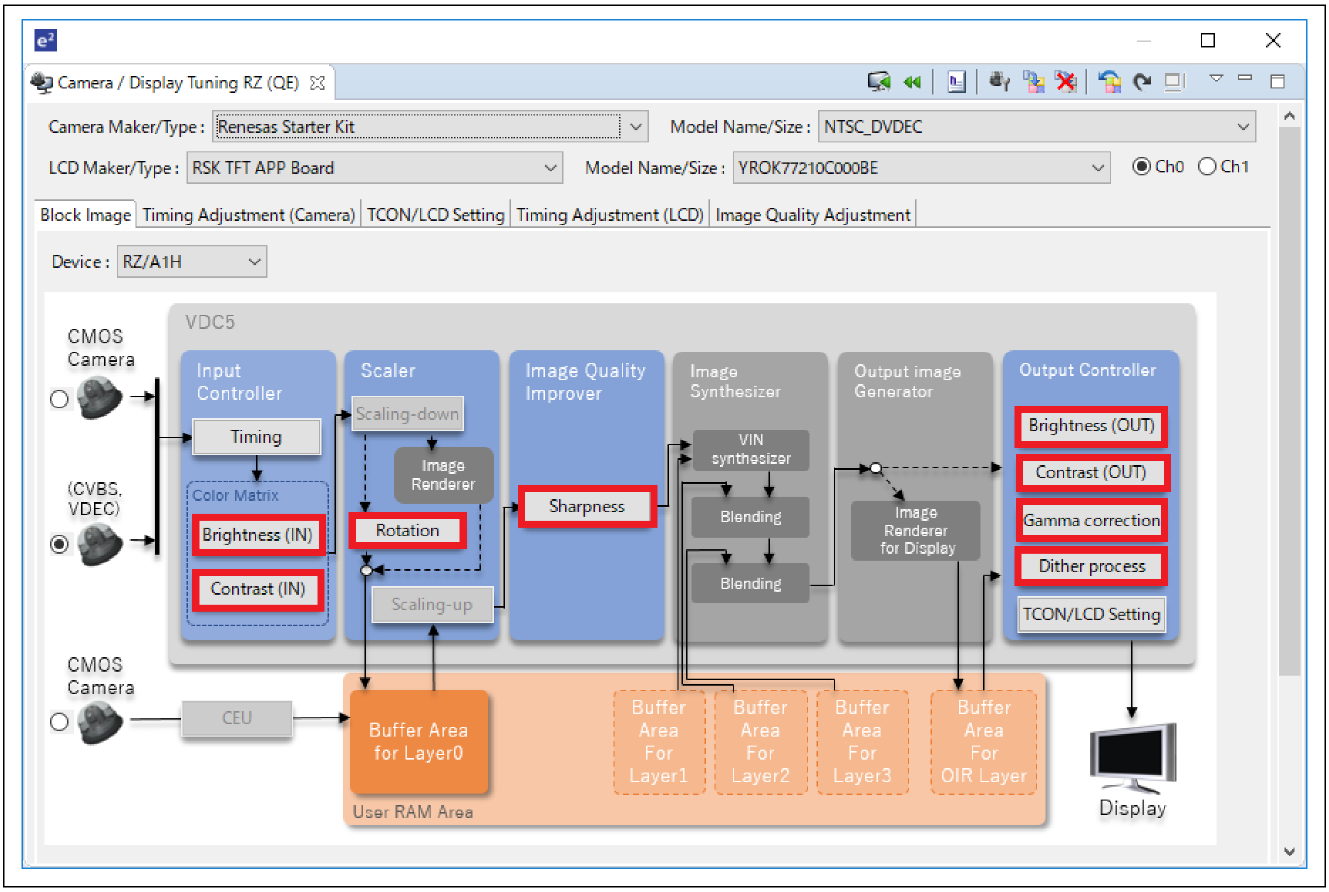


Figure 6.12 Selection of Image Quality Adjustment

You can also modify various image adjustments in the “Image Quality Adjustment” tab. The image adjustment can be performed with either the “Quick Setting” or “Custom” control. If using “Custom”, then please refer to the document “RZ/A1H Group, RZ/A1M Group User's Manual: Hardware (R01UH0403)” and check the setting contents and range of values that can be set, for each register.

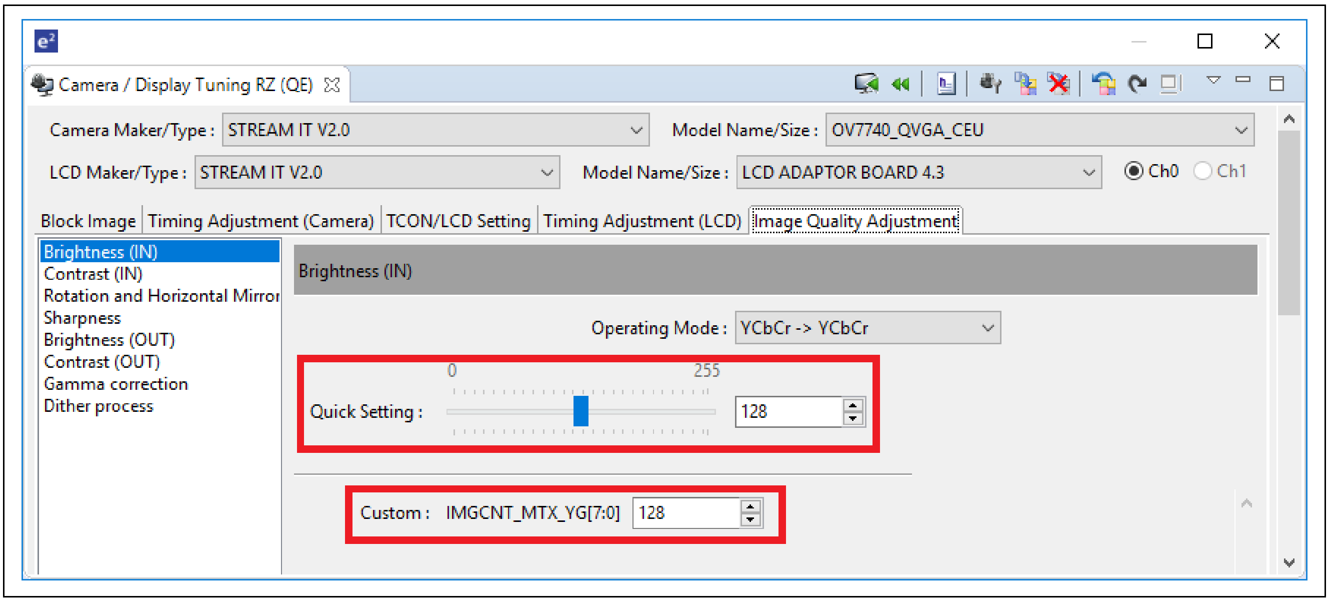


Figure 6.13 Image Quality Adjustment Display

Generating a Header File for Image Adjustment

A header file reflecting the control timing settings can be generated by clicking the generate header file icon (Figure 6.9) in **QE for Display**.

In the “Select the Header File” dialog box the header file name and output destination can be specified. Then, after selecting only the “For Image Adjustment” checkbox, click the “Generate” button to generate the header file.

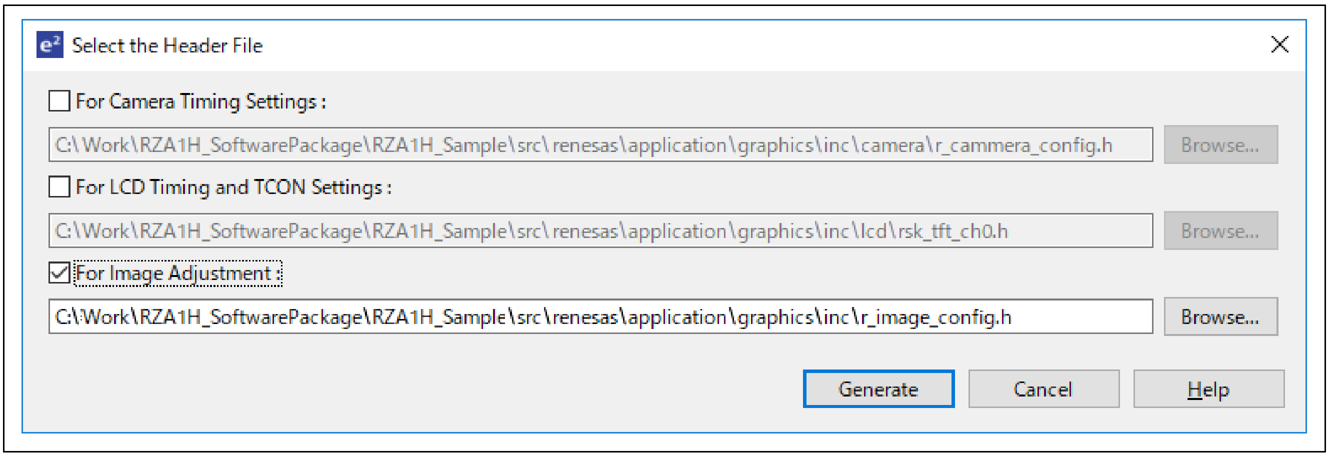


Figure 6.14 Generating a Header File for Image Adjustment

The sample program references the generated header file and uses the image adjustment settings to initialize VDC5. Save the header file using the folder name and file name shown below.

[Folder]

- `RZA1H_Sample\src\renesas\application\graphics\inc`

[File name]

- `r_image_config.h`

6.6 Image Download Function of QE for Display

The image download function of **QE for Display** can be used to display any image you wish. To download image data when using this sample program, please set a break point (see below) using the debugger and then download the image by following the procedure below.

[Folder]

- `\RZA1H_Sample\src\renesas\application\app_sdk_camera`

[File name]

- `r_sdk_camera_graphics.c`

[Function name]

- sdk_camera_graphics_sample_task() function. Place a break point inside the while loop.

Click the icon shown in Figure 6.15 to display the **Send the Image** dialog box (Figure 6.16).

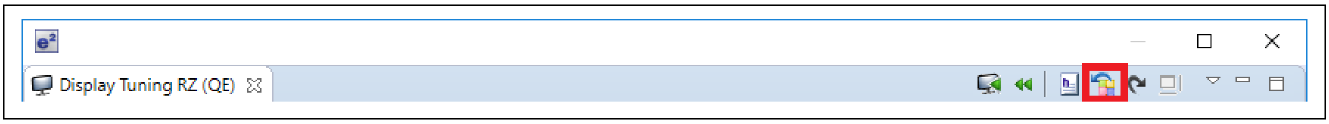


Figure 6.15 Image Download Icon

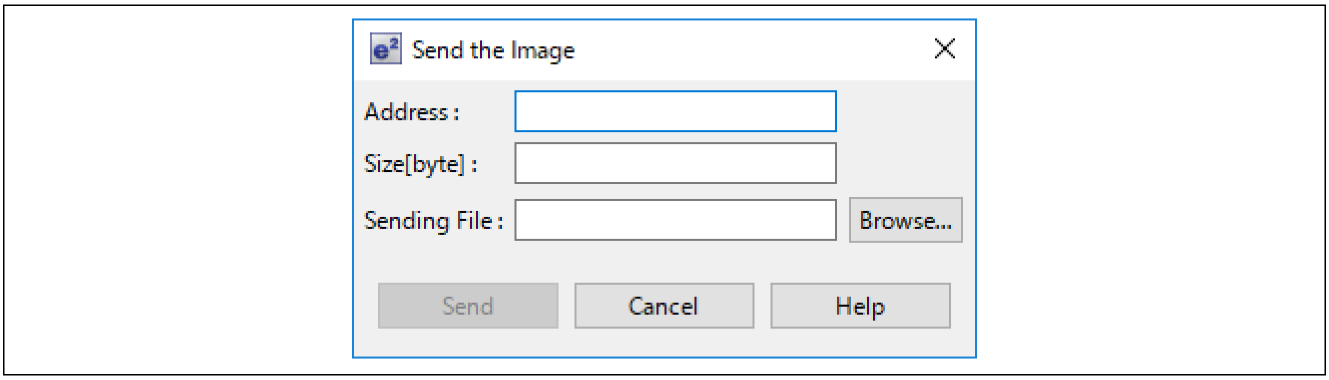


Figure 6.16 Send Image Dialog Box

Address:	Specify the address of the buffer for storing display data
Size [bytes]:	Image file size (when not specified, the file size is used)
Sending File:	Image file (binary format)

In the sample program, a buffer for storing display data is allocated by the **video_buffer** array variable and allocated to the VRAM section (0x60800000 for RZ/A1H). Enter the address of the **video_buffer** array variable into the "Address" field of the dialog box.

For the “Sending File” field use a binary image that matches the display data format set in the sample program. The display data format is set to YCbCr422 in the sample program.

7. Adapting the Sample Program to the User Environment

In order to use it for operating LCD displays in the your environment, the sample program must be modified to match that environment. The locations to be modified are listed below.

7.1 CPU and Board Initialization

The sample program performs initialization to match Renesas Starter Kit+ for RZ/A1H. It is therefore necessary to make modifications to match the user environment.

7.2 VDC5 Operation Settings

Some items related to display device control can be specified using **QE for Display** and some cannot. The settings that cannot be made using **QE for Display** are specified by the sample program instead. Therefore, the following three items need to be modified to match the user environment.

- Display device timing control
  The sample program references "`rsk_tft_ch0.h`" when making VDC5 settings. This header file is generated by **QE for Display**.
- Panel clock settings
  The sample program references "`rsk_tft_clk.h`". This file must be modified to match the user environment. Note that this item cannot be generated by **QE for Display**.
- GPIO settings
  The sample program uses setting file that is put in following directory. The file must be modified to match the user environment. Note that this item cannot be generated by **QE for Display**.

[Folder]

- `\RZA1H_Sample\src\renesas\application\graphics`

[File name]

- `vdc_portsetting.c`

The panel clock settings that cannot be specified using **QE for Display** are described below.

7.2.1 Panel Clock Settings

Source Clock Selection

VDC5 allows a variety of input clocks to be used as the source clock to generate the panel clock. The selectable source clocks are listed below. Note that the selected source clock is then frequency divided (1/1 to 1/32).

[Source clocks]

- Video clock (VIDEO_X1)
- Video clock (DV_CLK)
- External clock 0 (LCD0_EXTCLK)
- External clock 1 (LCD1_EXTCLK)
- Peripheral clock 1 (P1ϕ)
- LVDS PLL clock
- LVDS PLL clock × 1/7

The sample program uses the selections listed below to produce a "Panel Clock Frequency" of 27 MHz, as set in Figure 6.7, to match the specifications of the RSK TFT APP BOARD.

- Peripheral clock 1 (P1ϕ) 66.6 MHz
- 1/2 division ratio

These settings are located in "`rsk_tft_clk.h`".

- #define LCD_CH0_PANEL_CLK (VDC_PANEL_ICKSEL_PERI)
- #define LCD_CH0_PANEL_CLK_DIV (VDC_PANEL_CLKDIV_1_2)

8. Using Sample Program and QE for Camera

This sample program can be linked with the development support tool **QE for Camera**. **QE for Camera** is a plugin for the e[2] studio Integrated Development Environment. It can adjust the timing of the camera input using a GUI and can set the camera module registers via I^2C (Figure 8.1) (Figure 8.2).

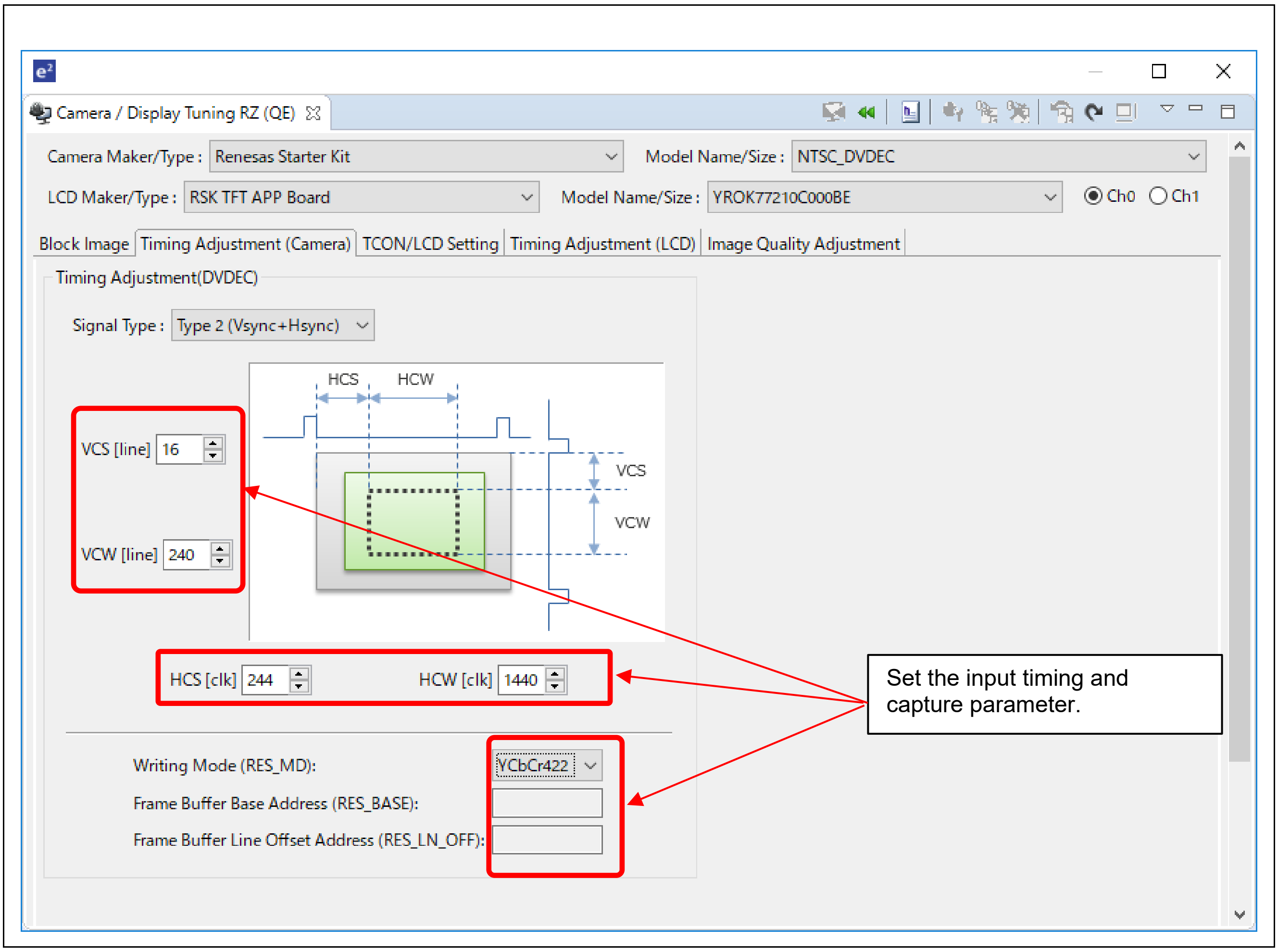


Figure 8.1 Adjusting Camera Input Timing

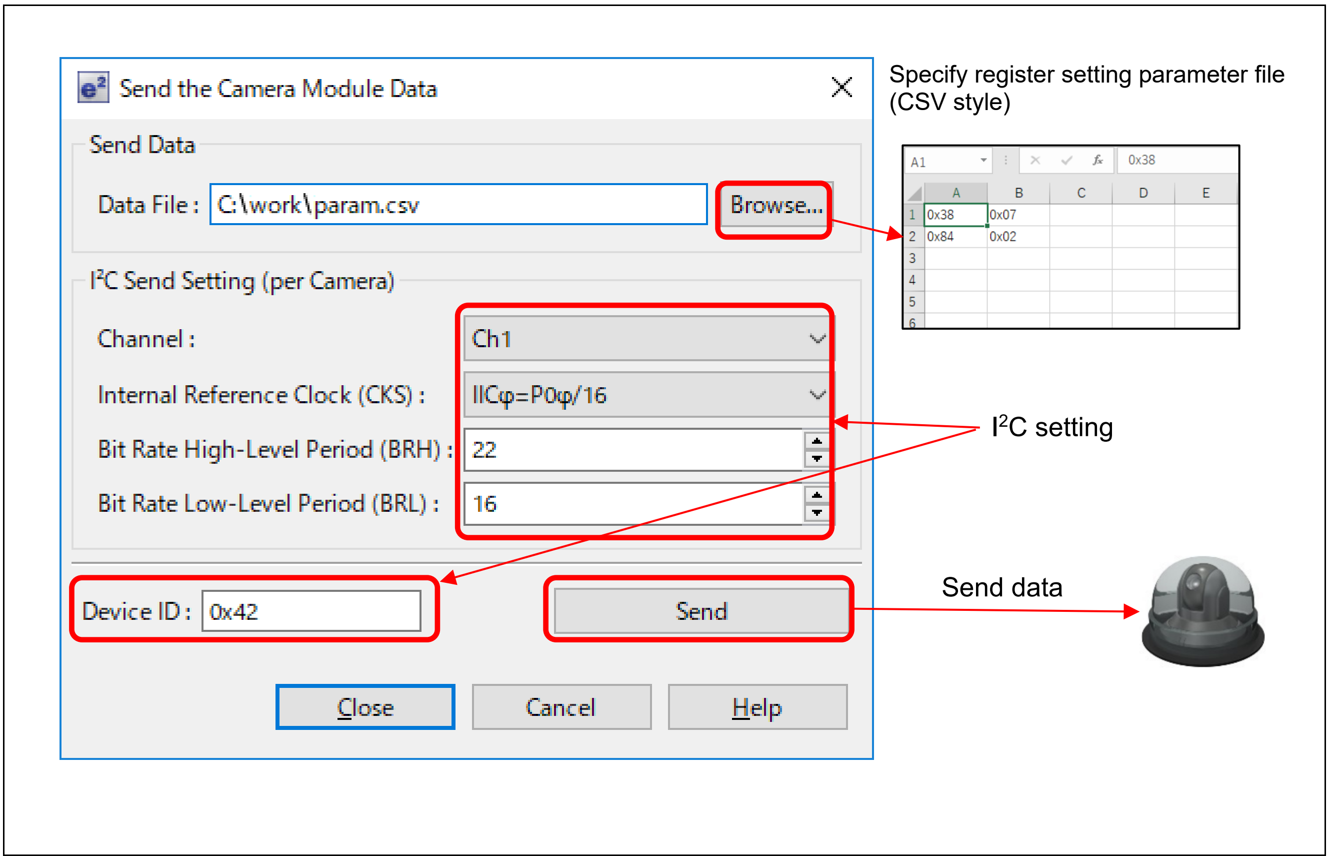


Figure 8.2 Setting of Camera Module Registers via I²C

Also, the timing settings configured by **QE for Camera** can be output as a header file, and by using it to replace the header file of the sample program, can be used as its new initial settings.

The sample program refers to a header file which is placed in path below for the initial camera input timing settings. Please refer to the following path contents and save / apply the new initial settings when after outputting the setting values from QE.

[path]

```
\RZA1H_Sample\src\renesas\application\graphics\inc\camera
```

For more detail of how to use QE for Camera, please refer to help documents in e^2 studio after installing **QE for Camera**.

Also, for instructions for installing e^2 studio and **QE for Camera**, refer to the manuals for each of these tools.

Product page:

https://www.renesas.com/qe-camera

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Nov. 29, 2019	-	First Edition issued

General 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.

Corporate Headquarters

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

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/.