User's Manual

Image Composer

User's Manual

Multimedia Processor for Mobile Applications
EMMA Mobile ™ EV2

Renesas Electronics
www.renesas.com

Rev.7.00 Dec 2011

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be 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, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

   Handle unused pins in accord 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 unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of 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. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment 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 moment 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 moment 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 moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has 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. Moreover, 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.

5. Differences between Products

   Before changing from one product to another, i.e. to one with a different part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MCU. It is intended for users designing application systems incorporating the MCU. A basic knowledge of electric circuits, logical circuits, and MCUs is necessary in order to use this manual.

The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and electrical characteristics; and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the EMMA Mobile EV2. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

Document Type	Description	Document Title	Document No.
Data Sheet	Hardware overview and electrical characteristics	EMMA Mobile EV2 Datasheet	R19DS0010EJxxxx
User's manual (1chip)	Hardware whole specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description	EMMA Mobile EV2 User's manual 1chip	R19UH0036EJxxxx
User's manual (module)	Hardware each macro specifications and operation description.	EMMA Mobile EV2 User's manual each module	See below

Document Name	Document No.	Document Name	Document No.
1chip	R19UH0036EJxxxx	DMA Controller	R19UH0043EJxxxx
System Management Unit	R19UH0037EJxxxx	LP-DDR/DDR2 Controller	R19UH0039EJxxxx
Timer	R19UH0054EJxxxx	SD Memory Card Interface	R19UH0061EJxxxx
System Timer	R19UH0055EJxxxx	SDIO Interface	R19UH0042EJxxxx
HD Video Decoder	R19UH0056EJxxxx	CF Card Interface	R19UH0062EJxxxx
Rotator	R19UH0057EJxxxx	Unified Serial Interface	R19UH0047EJxxxx
Resizer	R19UH0058EJxxxx	UART interface	R19UH0040EJxxxx
Image Composer (This manual)	R19UH0038EJxxxx	USB 2.0 Host Controller	R19UH0045EJxxxx
LCD Interface	R19UH0044EJxxxx	USB 2.0 Function Contoller	R19UH0034EJxxxx
ITU-R BT.656 Interface	R19UH0059EJxxxx	IIC Interface	R19UH0052EJxxxx
Digital Terrestrial TV Interface	R19UH0048EJxxxx	General Purpose I/O Interface	R19UH0041EJxxxx
Camera Interface	R19UH0060EJxxxx	Pulse Width Modulation Interface	R19UH0063EJxxxx

4 digits of end shows the version.

2. Notation of Numbers and Symbols

The notation conventions for register names, bit names, numbers, and symbols used in this manual are described below.

(1) Register Names, Bit Names, and Pin Names

Registers, bits, and pins are referred to in the text by symbols. The symbol is accompanied by the word “register,” “bit,” or “pin” to distinguish the three categories.

Examples the PM03 bit in the PM0 register

P3_5 pin, VCC pin

(2) Notation of Numbers

The indication “b” is appended to numeric values given in binary format. However, nothing is appended to the values of single bits. The indication “h” is appended to numeric values given in hexadecimal format. Nothing is appended to numeric values given in decimal format.

Examples Binary: 11b or 11

Hexadecimal: EFA0h

Decimal: 1234

3. Register Notation

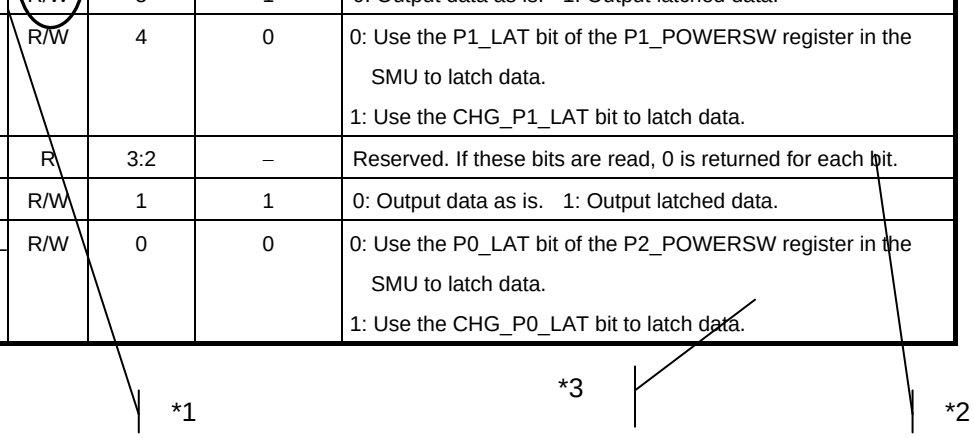
The symbols and terms used in register diagrams are described below.

x.x.x XXX register

This register (XXXXXXX: xxxx_xxxxh) …..

7	6	5	4	3	2	1	0
Reserved		CHG_P1_LAT	LATCH_P1_SEL	Reserved		CHG_P0_LAT	CHG_P0_LAT_SEL

Name	R/W	Bit No.	After Reset	Description
LATCH_P2_SEL	R/W	8	0	0: Use the P2_LAT bit of the P2_POWERSW register in the SMU to latch data. 1: Use the CHG_P2_LAT bit to latch data.
Reserved	R	7:6	–	Reserved. If these bits are read, 0 is returned for each bit.
CHG_P1_LAT	R/W	5	1	0: Output data as is. 1: Output latched data.
LATCH_P1_SEL	R/W	4	0	0: Use the P1_LAT bit of the P1_POWERSW register in the SMU to latch data. 1: Use the CHG_P1_LAT bit to latch data.
Reserved	R	3:2	–	Reserved. If these bits are read, 0 is returned for each bit.
CHG_P0_LAT	R/W	1	1	0: Output data as is. 1: Output latched data.
CHG_P0_LAT_SEL	R/W	0	0	0: Use the P0_LAT bit of the P2_POWERSW register in the SMU to latch data. 1: Use the CHG_P0_LAT bit to latch data.



*1

R/W: Read and Write.

R: Read only.

W: Write only.

-: Nothing is assigned.

*2

Reserved bit.

Reserved bit. Set to specified value.

*3

・ Nothing is assigned.

Nothing is assigned to the bit. As the bit may be used for future functions, if necessary, set to 0.

・ Do not set to a value.

Operation is not guaranteed when a value is set.

・ Function varies according to the operating mode.

The function of the bit varies with the peripheral function mode. Refer to the register diagram for information on the individual modes.

4. List of Abbreviations and Acronyms

Abbreviation	Full Form
ACIA	Asynchronous Communication Interface Adapter
bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
GSM	Global System for Mobile Communications
Hi-Z	High Impedance
IEBus	Inter Equipment Bus
I/O	Input/Output
IrDA	Infrared Data Association
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connect
PLL	Phase Locked Loop
PWM	Pulse Width Modulation
SFR	Special Function Register
SIM	Subscriber Identity Module
UART	Universal Asynchronous Receiver/Transmitter
VCO	Voltage Controlled Oscillator

Abbreviation	Full Form
APB	Advanced Peripheral Bus
AXI	Advanced eXtensible Interface
CAM	Camera Interface Module
FIFO	First In, First Out
HD	High Definition
ITU-R	International Telecommunication Union Radiocommunications sector
SIZ	Resizer

Table of Contents

1. Overview

The image composer (IMC, IMCW) generates images to be displayed in the LCD by the LCD controller. The IMC can superimpose up to seven images to synthesize them into one image and output it to the memory or LCD controller. Images can be output to the memory and the LCD controller at the same time. The IMCW can superimpose up to four images to synthesize them into one image and output it to the memory.

1.1 Features

The main features of the IMC and IMCW are as follows:

1.1.1 Layer Synthesis

- ○ Input images that have up to seven frames are divided into four layers using the synthesis depth. These four layers are further divided into three processes (front, middle, and back) for parallel processing.
  - Front process: Layer 0 (L0)
  - Middle process: Layer 1 (L1A, L1B, L1C)
  - Back process: Layer 2 (L2A, L2B), layer BG (BG) (IMC only)

  For details, see **3.3 Image Synthesis.**
- ○ Images are synthesized based on the ARGB8888 and RGB888 formats. All input images are converted to ARGB8888 (except for the back process, which is converted to RGB888), and then synthesis is performed. After synthesis, RGB888 images are converted to the specified format and output.
- ○ The input image format, synthesis position, and synthesis size can be separately specified.
- ○ An input image (sub-image) can be cut out of an image (main image) stored in a frame buffer.
- ○ Input images support double buffering and both interlaced and progressive scanning.
- ○ Resizing, mirroring, and wrapping around are only supported when inputting images from memory by using the AXI master read interface. These features are not supported when inputting images from the AXI slave write interface, and the register is ignored if specified.
- ○ The front and middle process each support transparency colors and alpha blending for the background layer.
- ○ If inputting a YUV image, gain offset and a byte lane are supported. Calculation coefficients can be selected from ITU-R BT.601-compliant, ITU-R BT.709-compliant, custom coefficient (with or without Y value subtracted by 16) for conversion. (This function is IMC only.)
- ○ The format and size of output images can be specified.
- ○ An output image can be used to update a portion (a sub-image) of an image stored in a frame buffer (a main image). (There are restrictions if the image is output in the RGB666 format.)
- ○ Output images support double buffering and both interlaced and progressive scanning.
- ○ Output images support mirroring. Wrapping around is only supported in the immediate startup mode.

1.1.2 Data Interface

I/O is handled by three AXI interfaces that comply with AMBA® AXI Protocol v1.0 and an LCD interface.

○ 64-bit AXI master read interface × 1
- This is used to read images input from memory
- Data can be supplied to multiple frames at the same time.
- Burst and non-burst settings are available.
- A burst length of up to 8 bits can be controlled using data remaining in the FIFO buffer.

○ 64-bit AXI master write interface ×1
- This is used exclusively to write synthesis results back to memory.
- Burst and non-burst settings are available.
- A burst length of up to 8 bits can be controlled using data remaining in the FIFO buffer.

○ 64-bit AXI slave write interface ×1
- This is used exclusively to receive image data output by the external master.
- Data can be supplied to up to one frame at the same time.

○ 32-bit LCD controller interface ×1
- This outputs synthesis results to a FIFO buffer in the LCD controller using a width of 32 bits.

1.1.3 Input

○ Input mode
- Input images defined in memory using two dimensions can be read by using the AXI master read interface.
  For details about the rules used to define images in the memory space, see **3.1.1 Images stored in the memory space.**
- Up to one frame of an input image can be directly input from the external AXI master by using the AXI slave write interface. For details, see **3.4 Inputting Images by Using the Slave Write Interface**.

○ Format
- RGB565, RGB666, RGB888
- ARGB1555, ARGB4444, ARGB8888
- YUV444, YUV422 **Note 1**, YUV420 **Note 2**
- Black background, fixed color

  Supported formats vary for each frame. For details, see **Table 3-1. Format Selectable for Each Frame**.

Notes 1. Three storage methods are defined for the YUV422 format used by EM/EV: the pixel interleaving method, where each two-pixel unit of data (Y component, U and V components) is set to a word for storage, the semi-planar method, where the Y components are stored as one plane and the UV components are stored as another, and the planar method, where the Y, U, and V components are stored as three separate planes.

2. Two storage methods are defined for the YUV420 format used by EM/EV: the semi-planar method, where the Y components are stored as one plane and the UV components are stored as another, and the planar method, where the Y, U, and V components are stored as three separate planes.

○ Format conversion
- The format of input images is converted to ARGB888 or RGB888 before performing synthesis.
  For details, see **3.2.4 Format conversion rule.**

- If converting from YUV to RGB data, approximately 0 to 2 times gain adjustment and offset adjustment are supported for each YUV component, as well as calculation coefficients from ITU-R BT.601-compliant, ITU-R BT.709-compliant, custom coefficient (with or without Y value subtracted by 16).
  For details, see **2.2.24 YUV2RGB conversion mode register** and **2.2.25 Custom coefficient registers.**

○ Scanning modes
- Interlaced and progressive scanning are supported.
- The scanning mode can be specified separately for each frame. For details, see **3.3.8 Progressive and interlaced scanning.**

○ Image size
- Progressive
  - Minimum size: 0 × 0 pixels
  - Maximum size: 4,094 × 4,094 pixels
  - Minimum block size: 1 × 1 pixel
- Interlaced
  - Minimum size: 0 × 0 pixels
  - Maximum size: 4,094 × 4,094 pixels
  - Minimum block size: 2 × 1 pixels

○ Buffer size
- Up to 16,382 × 16,382 pixels
- The minimum unit varies according to the format. For details, see **3.2.2 Restrictions on image setting.**

○ Wrapping around
- When reading image data, if the last (or first) column (in the horizontal direction) or row (in the vertical direction) in the frame buffer is reached, processing automatically wraps around to the first (or last) column or row to continue reading.
  For details, see **3.1.1 Images stored in the memory space.**
- Inputting from the AXI slave write interface is not supported.

○ Gamma adjustment (This function is IMC only.)
- Gamma conversion can be performed using the table look-up method for one of four locations (the three input processes and the output image) specified using a register.
- Gamma adjustment can be enabled or disabled separately for each frame in a selected process.
  For details, see **2.2.8 Gamma correction control register** and **3.6 Gamma correction.**

1.1.4 Output

○ Output mode
- Post-synthesis images can be written back to memory by using the AXI master write interface.
- Images can be output to the LCD controller by using the LCD interface. (IMC only)
- Only the AXI master write interface is enabled in the IMC immediate startup mode. For the LCD-synchronous mode, the status can be switched between only the LCD interface being enabled and both the LCD interface and AXI master write interface being enabled. (IMC only)

○ Format

- RGB565, RGB666, RGB888
- ARGB8888/ARGB4444

○ Format conversion
- Images are converted from RGB888 to the output format
- Dithering can be enabled or disabled. For details, see **3.2.4 Format conversion rule.**

○ Scanning modes
- Interlaced and progressive scanning are supported.

  For details, see **3.3.8 Progressive and interlaced scanning.**

○ Image size
- Progressive
  - Minimum size: 1 × 1 pixel
  - Maximum size: 4,094 × 4,094 pixels
  - Minimum block size:
    - RGB888, RGB565: 1 × 1 pixel
    - RGB666: 1 × 4 pixels
- Interlaced
  - Minimum size: 2 × 1 pixels
  - Maximum size: 4,094 × 4,094 pixels
  - Minimum block size:
    - RGB888, RGB565: 2 × 1 pixels
    - RGB666: 2 × 4 pixels

The same restrictions as RGB565 and RGB888 can conditionally be applied to RGB666. For details, see **2.2.14 WB image size register** and **2.2.16 WB memory frame start position register.**

○ Buffer size
- Up to 16,382 × 16,382
- 1 × 1-pixel units

○ Wrapping around
- When writing image data, if the last (or first) column (in the horizontal direction) or row (in the vertical direction) in the frame buffer is reached, processing automatically wraps around to the first (or last) column or row to continue writing. For details, see **3.1.1 Images stored in the memory space.**
- Wrapping around is disabled in the LCD synchronization mode

○ Gamma adjustment (This function is IMC only.)
- Gamma conversion can be performed using the table look-up method for one of four locations (the three input processes (front, middle, and back) and the output image) specified using a register. Gamma adjustment can be enabled or disabled separately for each frame in a selected process. Gamma conversion for frames other than those specified is disabled. For details, see **2.2.8 Gamma correction control register** and **3.6 Gamma correction.**

1.1.5 IMC Internal Memory

- ○ Three 32-word × 64-bit FIFO buffers (without bit masks) for storing input image data (one for each process) (IMCW is 2 mounted.)
- ○ One 32-word × 64-bit FIFO buffer (with a bit mask) for storing output WB image data
- ○ Three 256-word × 8-bit FIFO buffers (without bit masks) for storing the gamma adjustment table. (IMC only)

1.1.6 Low Power Mode

The IMC macro supports a low power standby mode (VGA standby mode) through use with an LCD controller. For details, see **3.4 Inputting Images by Using the Slave Write Interface**..

1.1.7 Restriction

・Phenomena：

After register setting is completed in LCD sync mode, when 0x1 is set to update reservation register (IMC_REFRESH), register update may not be issued.

・Condition：

－Operation is LCD sync mode.

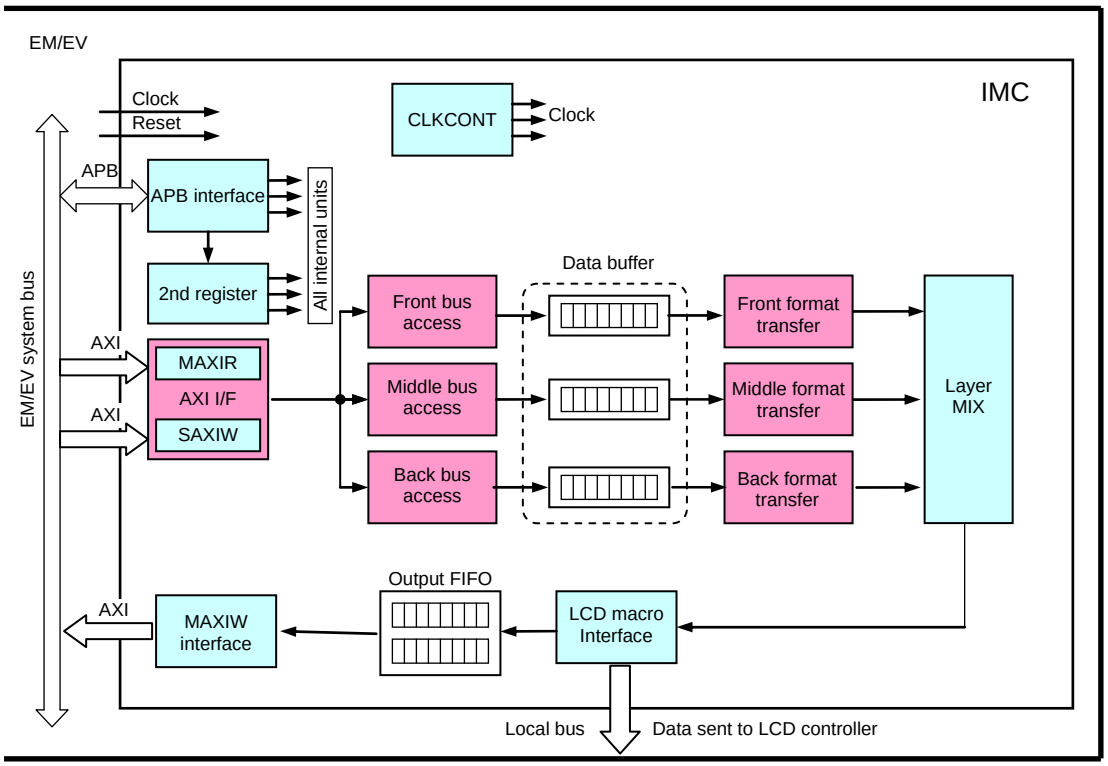
－When 0x1 is set to IMC_REFRESH register, register update is used.

・Workaround

After write “1” to IMC_REFRESH in LCD sync mode, confirm(read) the value of IMC_REFRESH, and if IMC_REFRESH is still “0”, 0x1 must be set in the Update Reserve Register once again.

1.2 Function Block Diagram

Figure 1-1. Function Block Diagram



2. Registers

The IMC registers can be accessed by way of the APB bus only in 32-bit (word) units. IMCW cannot be used in the LCD-synchronous mode.

2.1 Register List

IMC base address: E126_0000H
IMCW base address: E127_0000H

Caution Among addresses E126_0000H to E127_FFFCH, the addresses not listed in the following tables are reserved. Writing to reserved areas is prohibited. An undefined value is returned for read access. If attempted, the operation is not guaranteed.

(1/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
Function setting registers						
0000H	Control register	IMC_CONTROL	R/W	×	○	0000_0000H
0004H	Update reserve register	IMC_REFRESH	R/W	–	○	0000_0000H
0008H	Data request threshold register	IMC_DATAREQ	R/W	×	○	0000_0100H
000CH	Reserved	–	–	–	–	–
Image synthesis startup registers						
0010H	Reserved Startup register	IMC_START	R/W	○	○	0000_0000H
0014H	Status register	IMC_STATUS	R	–	○	0000_0000H
0018H	CPU double buffer control register	IMC_CPUBUFSEL	R/W	○	○	0000_0000H
001CH	Forced termination register	IMC_STOP	W	–	○	0000_0000H
Gamma correction registers						
0020H	Gamma correction control register	IMC_GAMMA_EN	R/W	○	×	0000_0000H
0024H	Gamma correction table address register	IMC_GAMMA_ADR	R/W	×	×	0000_0000H
0028H	Gamma correction table data register	IMC_GAMMA_DATA	R/W	×	×	xxxx_xxxxH
002CH to 003CH	Reserved	–	–	–	–	–
Registers for immediate startup settings						
0040H	Display area start address register	IMC_WB_AREAADR_P	R/W	○	○	0000_0000H
0044H	Address addition value register	IMC_WB_HOFFSET	R/W	○	○	0000_0000H
0048H	Format register	IMC_WB_FORMAT	R/W	○	○	0000_0000H
004CH	WB image size register	IMC_WB_SIZE	R/W	○	○	0000_0000H
0050H	Display area start address register	IMC_WB_AREAADR_Q	R/W	○	○	0000_0000H
0054H	WB double buffer control register	IMC_WB_BUFSEL	R/W	○	○	0000_0000H
0058H	WB memory frame start position register	IMC_WB_MPOSITION	R/W	○	○	0000_0000H
005CH	WB memory frame size register	IMC_WB_MSIZE	R/W	○	○	0000_0000H
0060H	Fixed color register	IMC_BACKCOLOR	R/W	○	○	0000_0000H
0064H	WB bytelane register	IMC_WB_BYTELANE	R/W	○	○	0000_E400H

(2/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0068H to 006CH	Reserved	–	–	–	–	–
0070H	WB output scanning mode register	IMC_WB_SCANMODE	R/W	○	○	0000_0000H
0074H to 0FFCH	Reserved	–	–	–	–	–
Registers for settings common to all layers						
0100H	Output image horizontal/vertical flip setting register	IMC_MIRROR	R/W	○	○	0000_0000H
0104H	Y gain offset register	IMC_YGAINOFFSET	R/W	×	×	0000_0080H
0108H	U gain offset register	IMC_UGAINOFFSET	R/W	×	×	0000_0080H
010CH	V gain offset register	IMC_VGAINOFFSET	R/W	×	×	0000_0080H
0110H	YUV2RGB conversion mode register	IMC_YUV2RGB	R/W	×	△	0000_0000H
0114H	Custom coefficient register (Coef R0)	IMC_COEF_R0	R/W	×	×	0000_0000H
0118H	Custom coefficient register (Coef R1)	IMC_COEF_R1	R/W	×	×	0000_0000H
011CH	Custom coefficient register (Coef R2)	IMC_COEF_R2	R/W	×	×	0000_0000H
0120H	Custom coefficient register (Coef R3)	IMC_COEF_R3	R/W	×	×	0000_0000H
0124H	Custom coefficient register (Coef G0)	IMC_COEF_G0	R/W	×	×	0000_0000H
0128H	Custom coefficient register (Coef G1)	IMC_COEF_G1	R/W	×	×	0000_0000H
012CH	Custom coefficient register (Coef G2)	IMC_COEF_G2	R/W	×	×	0000_0000H
0130H	Custom coefficient register (Coef G3)	IMC_COEF_G3	R/W	×	×	0000_0000H
0134H	Custom coefficient register (Coef B0)	IMC_COEF_B0	R/W	×	×	0000_0000H
0138H	Custom coefficient register (Coef B1)	IMC_COEF_B1	R/W	×	×	0000_0000H
013CH	Custom coefficient register (Coef B2)	IMC_COEF_B2	R/W	×	×	0000_0000H
0140H	Custom coefficient register (Coef B3)	IMC_COEF_B3	R/W	×	×	0000_0000H
0144H to 014CH	Reserved	–	–	–	–	–
0150H	Alpha select register 0	IMC_ALPHASEL0	R/W	○	○	0000_0000H
0154H	Alpha select register 1	IMC_ALPHASEL1	R/W	○	○	0000_0000H
0158H to 015CH	Reserved	–	–	–	–	–
0160H	Burst enable register	IMC_BURST_EN	R/W	×	○	0000_0101H
0164H	Maximum burst length switching threshold register	IMC_THRESHOLD	R/W	×	○	0000_1010H
0168H to 01FCH	Reserved	–	–	–	–	–
L0 setting registers						
0200H	L0 input image control register	IMC_L0_CONTROL	R/W	●	○	0000_0000H
0204H	L0 format register	IMC_L0_FORMAT	R/W	●	○	0000_0000H
0208H	L0 double buffer control register	IMC_L0_BUFSEL	R/W	●	○	0000_0000H
020CH	L0 byte lane register	IMC_L0_BYTELANE	R/W	●	○	0000_E400H
0210H	L0 transparency color control register	IMC_L0_KEYENABLE	R/W	●	○	0000_0000H

(3/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0214H	L0 transparency color register	IMC_L0_KEYCOLOR	R/W	●	○	0000_0000H
0218H	L0 alpha register	IMC_L0_ALPHA	R/W	●	○	0000_0000H
021CH	Reserved	–	–	–	–	–
0220H	L0 resize register	IMC_L0_RESIZE	R/W	●	○	0000_0000H
0224H	L0 horizontal/vertical flip setting register	IMC_L0_MIRROR	R/W	●	○	0000_0000H
0228H to 022CH	Reserved	–	–	–	–	–
0230H	L0 input address addition value register	IMC_L0_OFFSET	R/W	●	○	0000_0000H
0234H	L0 frame buffer start address register (P)	IMC_L0_FRAMEADR_P	R/W	●	○	0000_0000H
0238H to 023CH	Reserved	–	–	–	–	–
0240H	L0 frame buffer start address register (Q)	IMC_L0_FRAMEADR_Q	R/W	●	○	0000_0000H
0244H to 024CH	Reserved	–	–	–	–	–
0250H	L0 display position register	IMC_L0_POSITION	R/W	●	○	0000_0000H
0254H	L0 display size register	IMC_L0_SIZE	R/W	●	○	0000_0000H
0258H to 025CH	Reserved	–	–	–	–	–
0260H	L0 memory frame start position register	IMC_L0_MPOSITION	R/W	●	○	0000_0000H
0264H	L0 memory frame size register	IMC_L0_MSIZE	R/W	●	○	0000_0000H
0268H to 026CH	Reserved	–	–	–	–	–
0270H	L0 input scanning mode register	IMC_L0_SCANMODE	R/W	○	○	0000_0000H
0274H to 02FCH	Reserved	–	–	–	–	–
L1A setting registers						
0300H	L1A input image control register	IMC_L1A_CONTROL	R/W	●	○	0000_0000H
0304H	L1A format register	IMC_L1A_FORMAT	R/W	●	○	0000_0000H
0308H	L1A double buffer control register	IMC_L1A_BUFSEL	R/W	●	○	0000_0000H
030CH	L1A byte lane register	IMC_L1A_BYTELANE	R/W	●	○	0000_E400H
0310H	L1A transparency color control register	IMC_L1A_KEYENABLE	R/W	●	○	0000_0000H
0314H	L1A transparency color register	IMC_L1A_KEYCOLOR	R/W	●	○	0000_0000H
0318H	L1A alpha register	IMC_L1A_ALPHA	R/W	●	○	0000_0000H
031CH	Reserved	–	–	–	–	–
0320H	L1A resize register	IMC_L1A_RESIZE	R/W	●	○	0000_0000H
0324H	L1A horizontal/vertical flip setting register	IMC_L1A_MIRROR	R/W	●	○	0000_0000H
0328H to 032CH	Reserved	–	–	–	–	–
0330H	L1A input address addition value register	IMC_L1A_OFFSET	R/W	●	○	0000_0000H
0334H	L1A frame buffer start address register (P)	IMC_L1A_FRAMEADR_P	R/W	●	○	0000_0000H

(4/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0338H to 033CH	Reserved	–	–	–	–	–
0340H	L1A frame buffer start address register (Q)	IMC_L1A_FRAMEADR_Q	R/W	●	○	0000_0000H
0344H to 034CH	Reserved	–	–	–	–	–
0350H	L1A display position register	IMC_L1A_POSITION	R/W	●	○	0000_0000H
0354H	L1A display size register	IMC_L1A_SIZE	R/W	●	○	0000_0000H
0358H to 035CH	Reserved	–	–	–	–	–
0360H	L1A memory frame start position register	IMC_L1A_MPOSITION	R/W	●	○	0000_0000H
0364H	L1A memory frame buffer size register	IMC_L1A_MSIZE	R/W	●	○	0000_0000H
0368H to 036CH	Reserved	–	–	–	–	–
0370H	L1A input scanning mode register	IMC_L1A_SCANMODE	R/W	○	○	0000_0000H
0374H to 03FCH	Reserved	–	–	–	–	–
L1B setting registers						
0400H	L1B input image control register	IMC_L1B_CONTROL	R/W	●	○	0000_0000H
0404H	L1B format register	IMC_L1B_FORMAT	R/W	●	○	0000_0000H
0408H	L1B double buffer control register	IMC_L1B_BUFSEL	R/W	●	○	0000_0000H
040CH	L1B byte lane register	IMC_L1B_BYTELANE	R/W	●	○	0000_E400H
0410H	L1B transparency color control register	IMC_L1B_KEYENABLE	R/W	●	○	0000_0000H
0414H	L1B transparency color register	IMC_L1B_KEYCOLOR	R/W	●	○	0000_0000H
0418H	L1B alpha register	IMC_L1B_ALPHA	R/W	●	○	0000_0000H
041CH	Reserved	–	–	–	–	–
0420H	L1B resize register	IMC_L1B_RESIZE	R/W	●	○	0000_0000H
0424H	L1B horizontal/vertical flip setting register	IMC_L1B_MIRROR	R/W	●	○	0000_0000H
0428H to 042CH	Reserved	–	–	–	–	–
0430H	L1B input address addition value register	IMC_L1B_OFFSET	R/W	●	○	0000_0000H
0434H	L1B frame buffer start address register (P)	IMC_L1B_FRAMEADR_P	R/W	●	○	0000_0000H
0438H to 043CH	Reserved	–	–	–	–	–
0440H	L1B frame buffer start address register (Q)	IMC_L1B_FRAMEADR_Q	R/W	●	○	0000_0000H
0444H to 044CH	Reserved	–	–	–	–	–
0450H	L1B display position register	IMC_L1B_POSITION	R/W	●	○	0000_0000H
0454H	L1B display size register	IMC_L1B_SIZE	R/W	●	○	0000_0000H
0458H to 045CH	Reserved	–	–	–	–	–

(5/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0460H	L1B memory frame start position register	IMC_L1B_MPOSITION	R/W	●	○	0000_0000H
0464H	L1B memory frame buffer size register	IMC_L1B_MSIZE	R/W	●	○	0000_0000H
0468H to 046CH	Reserved	–	–	–	–	–
0470H	L1B input scanning mode register	IMC_L1B_SCANMODE	R/W	○	○	0000_0000H
0474H to 04FCH	Reserved	–	–	–	–	–
L1C setting registers						
0500H	L1C input image control register	IMC_L1C_CONTROL	R/W	●	○	0000_0000H
0504H	L1C format register	IMC_L1C_FORMAT	R/W	●	○	0000_0000H
0508H	L1C double buffer control register	IMC_L1C_BUFSEL	R/W	●	○	0000_0000H
050CH	L1C byte lane register	IMC_L1C_BYTELANE	R/W	●	○	0000_E400H
0510H	L1C transparency color control register	IMC_L1C_KEYENABLE	R/W	●	○	0000_0000H
0514H	L1C transparency color register	IMC_L1C_KEYCOLOR	R/W	●	○	0000_0000H
0518H	L1C alpha register	IMC_L1C_ALPHA	R/W	●	○	0000_0000H
051CH	Reserved	–	–	–	–	–
0520H	L1C resize register	IMC_L1C_RESIZE	R/W	●	○	0000_0000H
0524H	L1C horizontal/vertical flip setting register	IMC_L1C_MIRROR	R/W	●	○	0000_0000H
0528H to 052CH	Reserved	–	–	–	–	–
0530H	L1C input address addition value register	IMC_L1C_OFFSET	R/W	●	○	0000_0000H
0534H	L1C frame buffer start address register (P)	IMC_L1C_FRAMEADR_P	R/W	●	○	0000_0000H
0538H to 053CH	Reserved	–	–	–	–	–
0540H	L1C frame buffer start address register (Q)	IMC_L1C_FRAMEADR_Q	R/W	●	○	0000_0000H
0544H to 054CH	Reserved	–	–	–	–	–
0550H	L1C display position register	IMC_L1C_POSITION	R/W	●	○	0000_0000H
0554H	L1C display size register	IMC_L1C_SIZE	R/W	●	○	0000_0000H
0558H to 055CH	Reserved	–	–	–	–	–
0560H	L1C memory frame start position register	IMC_L1C_MPOSITION	R/W	●	○	0000_0000H
0564H	L1C memory frame buffer size register	IMC_L1C_MSIZE	R/W	●	○	0000_0000H
0568H to 056CH	Reserved	–	–	–	–	–
0570H	L1C input scanning mode register	IMC_L1C_SCANMODE	R/W	○	○	0000_0000H
0574H to 05FCH	Reserved	–	–	–	–	–
L2A setting registers						
0600H	L2A input image control register	IMC_L2A_CONTROL	R/W	●	×	0000_0000H

(6/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0604H	L2A format register	IMC_L2A_FORMAT	R/W	●	×	0000_0000H
0608H	L2A double buffer control register	IMC_L2A_BUFSEL	R/W	●	×	0000_0000H
060CH	L2A byte lane register	IMC_L2A_BYTELANE	R/W	●	×	0000_E4E4H
0610H to 061CH	Reserved	–	–	–	–	–
0620H	L2A resize register	IMC_L2A_RESIZE	R/W	●	×	0000_0000H
0624H	L2A horizontal/vertical flip control register	IMC_L2A_MIRROR	R/W	●	×	0000_0000H
0628H to 062CH	Reserved	–	–	–	–	–
0630H	L2A input address addition value register	IMC_L2A_OFFSET	R/W	●	×	0000_0000H
0634H	L2A frame buffer start address register (YP)	IMC_L2A_FRAMEADR_YP	R/W	●	×	0000_0000H
0638H	L2A frame buffer start address register (UP)	IMC_L2A_FRAMEADR_UP	R/W	●	×	0000_0000H
063CH	L2A frame buffer start address register (VP)	IMC_L2A_FRAMEADR_VP	R/W	●	×	0000_0000H
0640H	L2A frame buffer start address register (YQ)	IMC_L2A_FRAMEADR_YQ	R/W	●	×	0000_0000H
0644H	L2A frame buffer start address register (UQ)	IMC_L2A_FRAMEADR_UQ	R/W	●	×	0000_0000H
0648H	L2A frame buffer start address register (VQ)	IMC_L2A_FRAMEADR_VQ	R/W	●	×	0000_0000H
064CH	Reserved	–	–	–	–	–
0650H	L2A display position register	IMC_L2A_POSITION	R/W	●	×	0000_0000H
0654H	L2A display size register	IMC_L2A_SIZE	R/W	●	×	0000_0000H
0658H to 065CH	Reserved	–	–	–	–	–
0660H	L2A memory frame start position register	IMC_L2A_MPOSITION	R/W	●	×	0000_0000H
0664H	L2A memory frame buffer size register	IMC_L2A_MSIZE	R/W	●	×	0000_0000H
0668H to 066CH	Reserved	–	–	–	–	–
0670H	L2A input scanning mode register	IMC_L2A_SCANMODE	R/W	○	×	0000_0000H
0674H to 06FCH	Reserved	–	–	–	–	–
L2B setting registers						
0700H	L2B input image control register	IMC_L2B_CONTROL	R/W	●	×	0000_0000H
0704H	L2B format register	IMC_L2B_FORMAT	R/W	●	×	0000_0000H
0708H	L2B double buffer control register	IMC_L2A_BUFSEL	R/W	●	×	0000_0000H
070CH	L2B byte lane register	IMC_L2B_BYTELANE	R/W	●	×	0000_E4E4H
0710H to 071CH	Reserved	–	–	–	–	–
0720H	L2B resize register	IMC_L2B_RESIZE	R/W	●	×	0000_0000H
0724H	L2B horizontal/vertical flip control register	IMC_L2B_MIRROR	R/W	●	×	0000_0000H
0728H to 072CH	Reserved	–	–	–	–	–
0730H	L2B input address addition value register	IMC_L2B_OFFSET	R/W	●	×	0000_0000H

(7/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
0734H	L2B frame buffer start address register (YP)	IMC_L2B_FRAMEADR_YP	R/W	●	×	0000_0000H
0738H	L2B frame buffer start address register (UP)	IMC_L2B_FRAMEADR_UP	R/W	●	×	0000_0000H
073CH	L2B frame buffer start address register (VP)	IMC_L2B_FRAMEADR_VP	R/W	●	×	0000_0000H
0740H	L2B frame buffer start address register (YQ)	IMC_L2B_FRAMEADR_YQ	R/W	●	×	0000_0000H
0744H	L2B frame buffer start address register (UQ)	IMC_L2B_FRAMEADR_UQ	R/W	●	×	0000_0000H
0748H	L2B frame buffer start address register (VQ)	IMC_L2B_FRAMEADR_VQ	R/W	●	×	0000_0000H
074CH	Reserved	–	–	–	–	–
0750H	L2B display position register	IMC_L2B_POSITION	R/W	●	×	0000_0000H
0754H	L2B display size register	IMC_L2B_SIZE	R/W	●	×	0000_0000H
0758H to 075CH	Reserved	–	–	–	–	–
0760H	L2B memory frame start position register	IMC_L2B_MPOSITION	R/W	●	×	0000_0000H
0764H	L2B memory frame buffer size register	IMC_L2B_MSIZE	R/W	●	×	0000_0000H
0768H to 076CH	Reserved	–	–	–	–	–
0770H	L2B input scanning mode register	IMC_L2B_SCANMODE	R/W	○	×	0000_0000H
0774H to 07FCH	Reserved	–	–	–	–	–
BG setting registers						
0800H	Reserved	–	–	–	–	–
0804H	BG format register	IMC_BG_FORMAT	R/W	●	×	0000_0000H
0808H	BG double buffer control register	IMC_BG_BUFSEL	R/W	●	×	0000_0000H
080CH	BG byte lane register	IMC_BG_BYTELANE	R/W	●	×	0000_E400H
0820H	BG resize register	IMC_BG_RESIZE	R/W	●	×	0000_0000H
0824H	BG horizontal/vertical flip setting register	IMC_BG_MIRROR	R/W	●	×	0000_0000H
0828H to 082CH	Reserved	–	–	–	–	–
0830H	BG input address addition value register	IMC_BG_OFFSET	R/W	●	×	0000_0000H
0834H	BG frame buffer start address register (P)	IMC_BG_FRAMEADR_P	R/W	●	×	0000_0000H
0838H to 083CH	Reserved	–	–	–	–	–
0840H	BG frame buffer start address register (Q)	IMC_BG_FRAMEADR_Q	R/W	●	×	0000_0000H
0844H to 085CH	Reserved	–	–	–	–	–
0860H	BG memory frame start position register	IMC_BG_MPOSITION	R/W	●	×	0000_0000H
0864H	BG memory frame buffer size register	IMC_BG_MSIZE	R/W	●	×	0000_0000H
0868H to 086CH	Reserved	–	–	–	–	–
0870H	BG input scanning mode register	IMC_BG_SCANMODE	R/W	○	×	0000_0000H
0874H to 08FCH	Reserved	–	–	–	–	–

(8/8)

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
Interrupt control registers						
0900H	Interrupt status register	IMC_INTSTATUS	R	×	○	0000_0000H
0904H	Interrupt raw status register	IMC_INTRAWSTATUS	R	×	○	0000_0000H
0908H	Interrupt enable set register	IMC_INTENSET	R/W	×	○	0000_0000H
090CH	Interrupt enable clear register	IMC_INTENCLR	W	×	○	0000_0000H
0910H	Interrupt source clear register	IMC_INTFFCLR	W	×	○	0000_0000H
0914H	AXI read error address register	IMC_ERRORADR_R	R/W	×	○	0000_0000H
0918H	AXI write error address register	IMC_ERRORADR_W	R/W	×	○	0000_0000H
091CH	AXI write error address register	IMC_ERRORADR_SW	R/W	×	○	0000_0000H
0920H to FFF8H	Reserved	–	–	–	–	–

Address	Register Name	Symbol	R/W	Frame Sync	IMCW	After Reset
FFFCH	Composition register	IMC_COMP	R/W	×	○	0000_0000H

Registers marked with ○ in the Frame Sync column are registers with which setting changes made in the register take effect when the frame start signal is received from the LCD controller (V-sync register). While updating is reserved for a register, do not change the settings.
Registers marked with ● are registers with which setting changes made in the register take effect when the frame start signal is received from the LCD controller while the update reserve register is set (update target register). While updating is reserved for a register, do not change the settings.
Registers marked with × are registers with which setting changes made in the register take effect immediately (immediately-reflected register). Changing the settings during macro operation is prohibited.
Registers marked with ○ in the IMCW column are the same register as IMC.
In registers marked with △in the IMCW column, certain bit is like IMC.
When the IMC runs in the immediate startup mode, startup by setting the startup register is regarded as the request for frame transmission start from the LCD controller, and register values are updated.
For details about the update register, see **3.7.3 Register update.**

2.2 Register Details

2.2.1 Control register

This register (IMC_CONTROL: E126_0000H, IMCW_CONTROL: E127_0000H) sets up the basic IMC operation.

Changes to this register are immediately applied, so changing the settings during operation (STATUS bit of IMC_STATUS register ≠ 0) is prohibited.

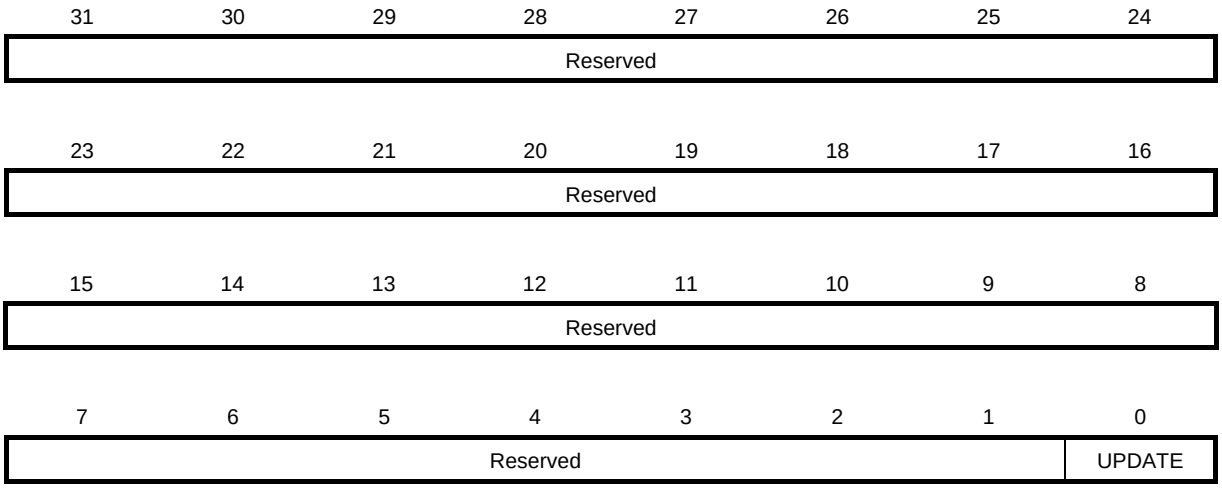
Name	R/W	Bit No.	After Reset	Description
DBGMODE	R/W	31	0	Specify the priority for supplying synthesized data to the LCD controller (with WB) (see **3.5.3 Overrun.**) 0: Availability of FIFO in the LCD controller (An IMC overrun is more likely to occur.) 1: Availability of FIFO in the LCD controller and transmission FIFO in the IMC (An IMC underrun is more likely to occur.)
Reserved	R	30:20	000H	Reserved. If these bits are read, 0 is returned for each bit.
Reserved	R/W	19:8	000H	Set 0 as all bits.
Reserved	R	7:4	0H	Reserved. If these bits are read, 0 is returned for each bit.
FORMAT	R	3:1	000b	Indicates the IMC output format. 000: RGB888 001: RGB666 010: RGB565 011: Reserved 100: ARGB8888 101: ARGB4444 110 – 111: Reserved
STARTMODE	R/W	0	0	Specify the IMC startup mode. 0: LCD-synchronous mode (started by a request from the LCD controller) 1: Immediate startup mode (started by setting the IMC_START register)

When using the IMC with the settings specified in the LCD controller, the startup signal is sent at the beginning of a frame transferred between the LCD controller and the IMC. If the STARTMODE bit is cleared to 0 in the IMC, the IMC starts operating at this timing.

Because IMCW is not connected to the LCD controller. Therefore, setting IMCW to the LCD-synchronous mode is prohibited.

2.2.2 Update reserve register

This register (IMC_REFRESH: E126_0004H, IMCW_REFRESH: E127_0004H) is used to apply the values specified for the update target registers
(registers marked with ● in the Frame Sync column in the register list) in the IMC, to the internal operation.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
UPDATE	R/W	0	0	Used to reserve register updates. 0: Do not update. 1: Update reserved.

An update end interrupt occurs when the register is updated. (The interrupt occurs immediately if the IMC has started immediately while an update is reserved, or when the next frame begins if the IMC has started in synchronization with the LCD controller.) The UPDATE bit is cleared to 0 at the same time as the update. For details about the circuit configuration, see **3.7.3 Register update.** For details about the operation timing and usage, see **3.8.2 Using the update register.**

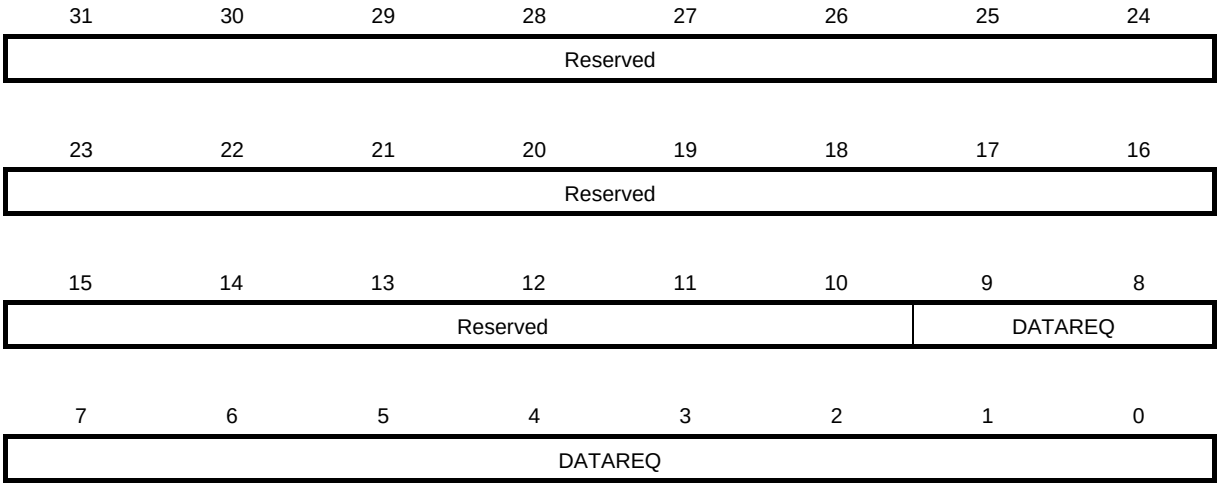
2.2.3 Data request threshold register

This register (IMC_DATAREQ: E126_0008H, IMCW_DATAREQ: E127_0008H) specifies the threshold for data requests from the LCD controller to the IMC.

This register is enabled only when bit 31 (DBGMODE) of the IMC_CONTROL register is cleared to 0. In the LCD-synchronous mode, if the amount of data remaining in the FIFO buffer in the LCD controller is less than the value specified for this register, data is requested for the IMC as long as the FIFOREADY signal from the LCD controller is H to prevent underruns from occurring. If the amount of data remaining in the FIFO buffer in the LCD controller is greater than or equal to the value specified for this register, because overruns are more likely to occur than underruns, data is only requested for the IMC as long as LCDREADY is H and the write-back FIFO buffer is empty.

For details about the register features and how to set the register, see **3.5.3 Overrun.**

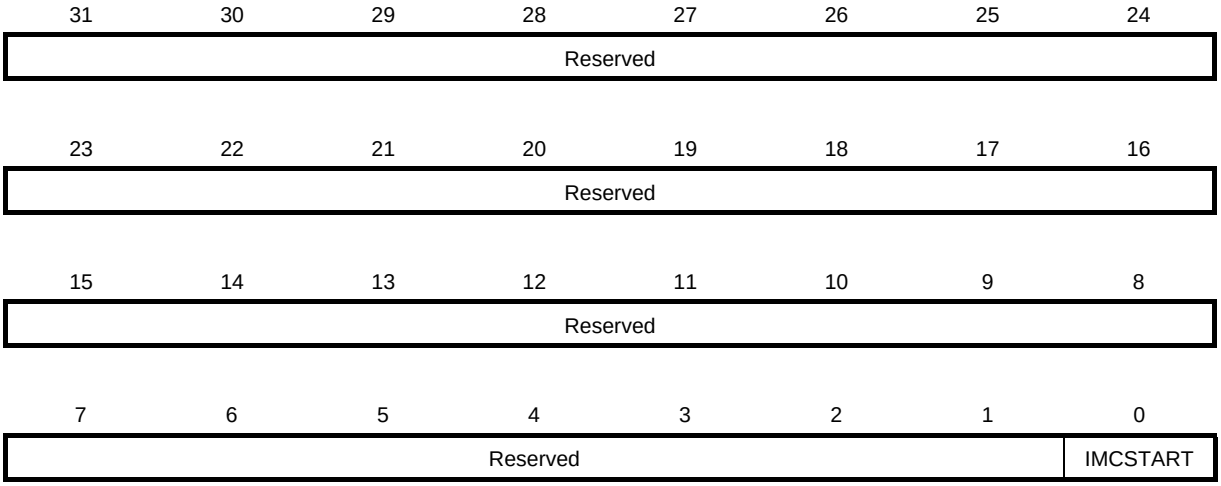
Changes to this register are immediately applied, so changing it during operation is prohibited.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:10	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
DATAREQ	R/W	9:0	100H	Specify the threshold for requesting data from the LCD controller to the IMC when supplying data to the LCD controller with WB processing. FIFO_REMAIN (LCD) ≥ DATAREQ: The LCD FIFO buffer and IMC-transmission FIFO buffer are empty. FIFO_REMAIN (LCD) < DATAREQ: The LCD FIFO buffer is empty.

2.2.4 Reserved Startup register

This register (IMC_START: E126_0010H, IMCW_START: E127_0010H) is used to start the IMC in the immediate startup mode (single startup).

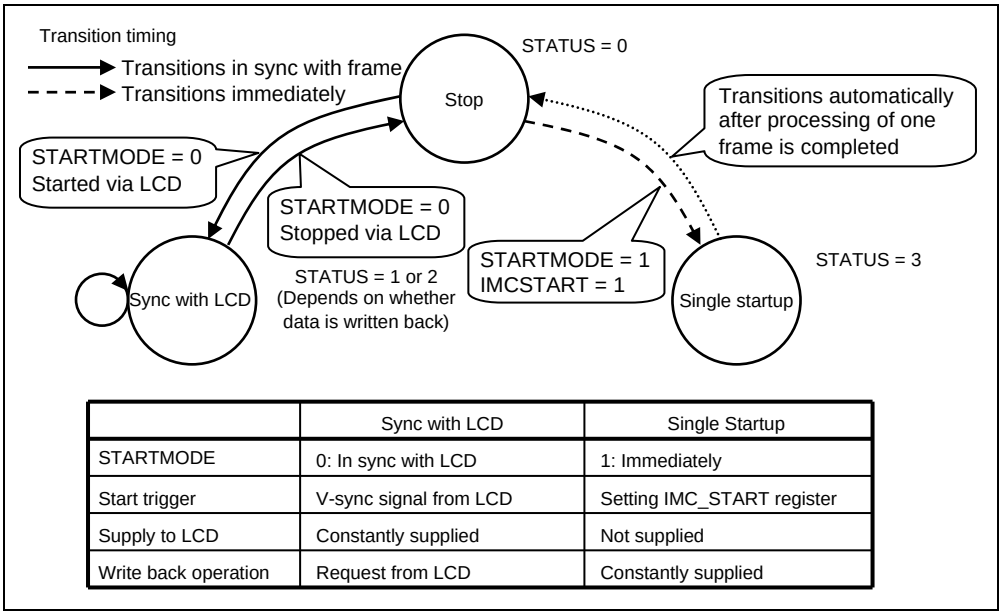


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
IMCSTART	R/W	0	0	Start the IMC in the immediate startup mode (STARTMODE = 1).
	W			0: Ignore 1: Reserved,
	R			0: no reserved 1: Reserved,

If the STARTMODE bit of the control register (IMC_CONTROL) is 1 (immediate startup) and the IMC_START register is set to 1, one frame of synthesis processing is performed, and then the IMC automatically starts through single startup. If the IMC_START register is set to 1 while the IMC is running, the operation is considered a reservation. When synthesis of the current frame ends, synthesis of the next frame automatically starts. Register settings for the next frame are applied when synthesis starts. Due to this, changing the register value between when the reservation is made and when the next frame starts is also applied. Therefore, if a reservation is made while there is already a reservation, the operation is not guaranteed.
For details about the operation timing, see **3.7.4 Reserved startup.**

For interlaced output, the odd and even fields must be started separately due to V-sync processing. However, because there are no V-sync problems in the immediate startup mode, by using the reservation function to set up registers and then setting the IMC_START register to 1 twice, it is possible to reserve synthesis of the even field. It is also possible to set the register to 1 once, wait for the field-writing completion interrupt, and then trigger startup again. If the register is only set to 1 once, the odd field is transferred and then processing stops.
If the STARTMODE bit of the control register (IMC_CONTROL) is 0 (LCD-synchronous), the IMC_START register is meaningless, and the IMC is started according to the LCD controller. If the LCD-synchronous mode is specified, writing is ignored.

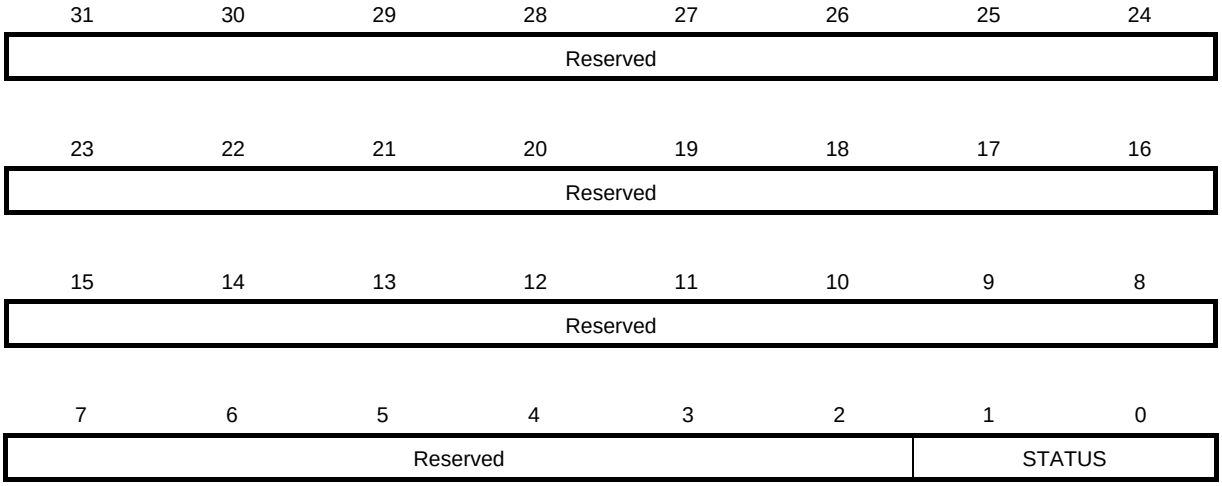
Figure 2-1. Status Transition



	Sync with LCD	Single Startup
STARTMODE	0: In sync with LCD	1: Immediately
Start trigger	V-sync signal from LCD	Setting IMC_START register
Supply to LCD	Constantly supplied	Not supplied
Write back operation	Request from LCD	Constantly supplied

2.2.5 Status register

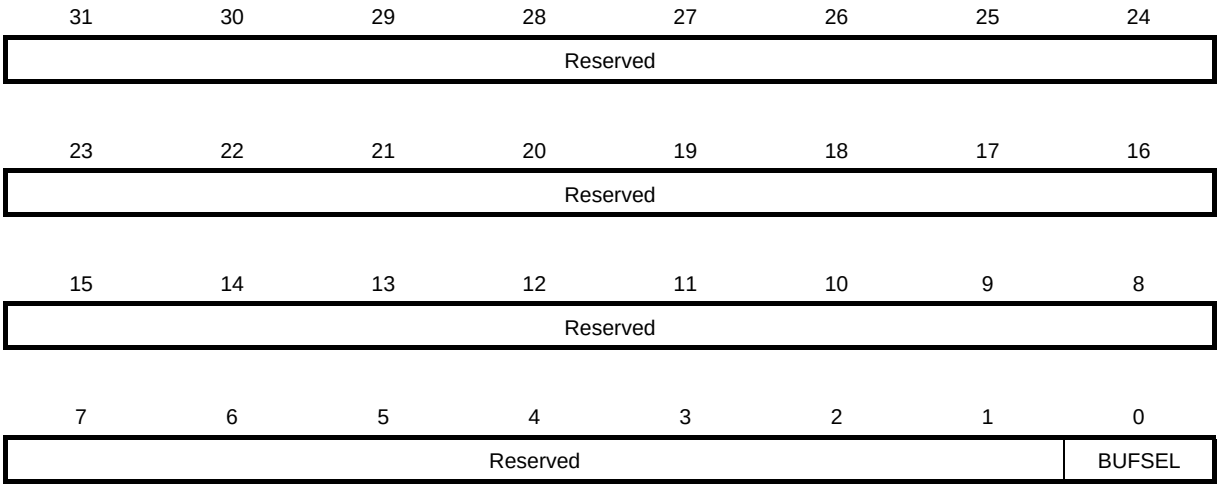
This register (IMC_STATUS: E126_0014H, IMCW_STATUS: E127_0014H) indicates the IMC running status. Read this register to check the IMC running status.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
STATUS	R	1:0	0H	Indicates the IMC running status. 0: Stopped 1: Supplying display data to the LCD controller (without WB). 2: Supplying display data to the LCD controller (with WB) 3: Running in the single startup mode (WB only).

2.2.6 CPU double buffer control register

This register (IMC_CPUBUFSEL: E126_0018H, IMCW_CPUBUFSEL: E127_0018H) switches buffers P and Q, when the buffer of a frame is determined by the CPU. The setting of this register is applied to each frame.
This is a V-sync register, so the setting is applied internally in synchronization with frame reception. It is not an update target register.

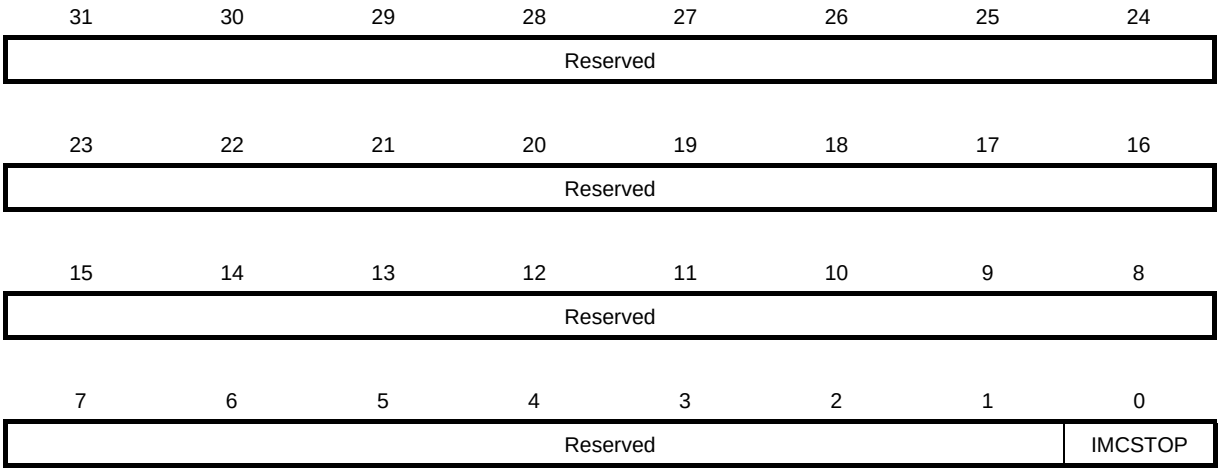


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
BUFSEL	R/W	0	0	Select buffer P or Q when double buffer of each frame is controlled by the CPU. 0: Buffer P 1: Buffer Q

This setting is ignored when double buffer of each frame is controlled in P/Q-fixed mode.

2.2.7 Forced termination register

This register (IMC_STOP: E126_001CH, IMCW_STOP: E127_001CH) retains register values and terminates running IMC processing. After triggering a forced termination, do not trigger the next activation before the IMC_STATUS register is cleared and the opposition master of the AXI slave write interface has finished the transfer or the transfer has stopped.

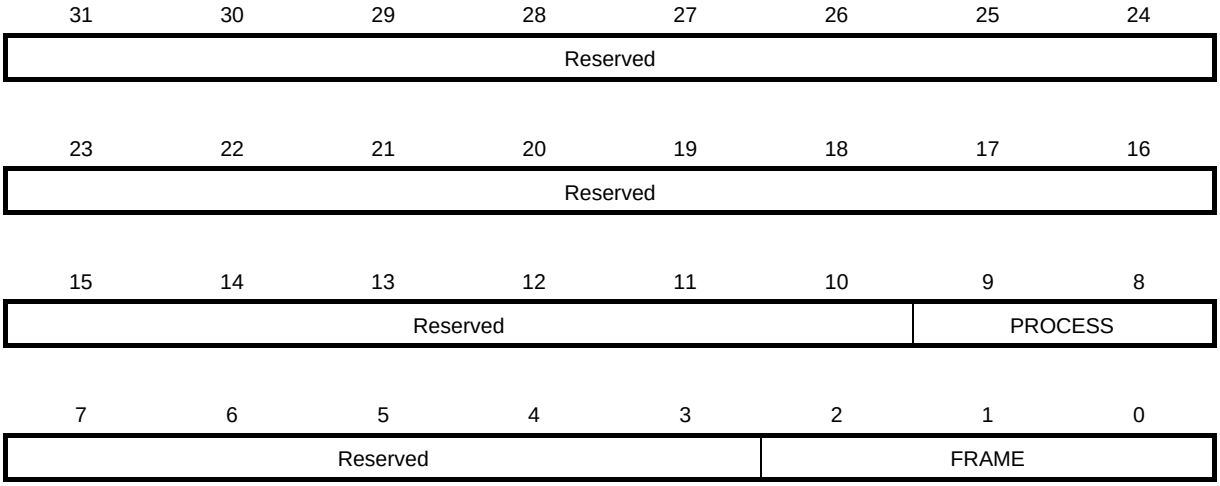


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
IMCSTOP	W	0	0	Terminate the IMC. 1: Terminate the IMC.

This register does the same thing as a reset, except register values are retained.

2.2.8 Gamma correction control register (IMC only)

This register (IMC_GAMMA_EN: E126_0020H) controls gamma correction.
Gamma conversion is performed using the table look-up method. Images are synthesized by dividing the processing into three parallel processes (front, middle, and back) and output. However, there is only one memory for the table look-up method. Therefore, it is not possible to access multiple memory locations in parallel. This register is used to enable only one of four locations for accessing the gamma conversion table. The frames in the same process are processed not in parallel but at different times. This register is also used to specify whether to enable or disable gamma correction for each frame.
This is a V-sync register. For details about gamma correction, see **3.6 Gamma Correction.**



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:10	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
PROCESS	R/W	9:8	00b	Specify the process for which to access the gamma correction table. 00: Output the synthesis result 01: Input images for the front process 10: Input images for the middle process 11: Input images for the back process
Reserved	R	7:3	00H	Reserved. If these bits are read, 0 is returned for each bit.
FRAME	R/W	2:0	0H	Specify whether to enable or disable gamma correction for each frame in the valid process specified for PROCESS. 0: Off 1: On The following table shows which frame corresponds to each bit.

PROCESS	0	1	2	3
Bit 2	–	–	L1C	BG
Bit 1	–	–	L1B	L2B
Bit 0	Synthesis result	L0	L1A	L2A

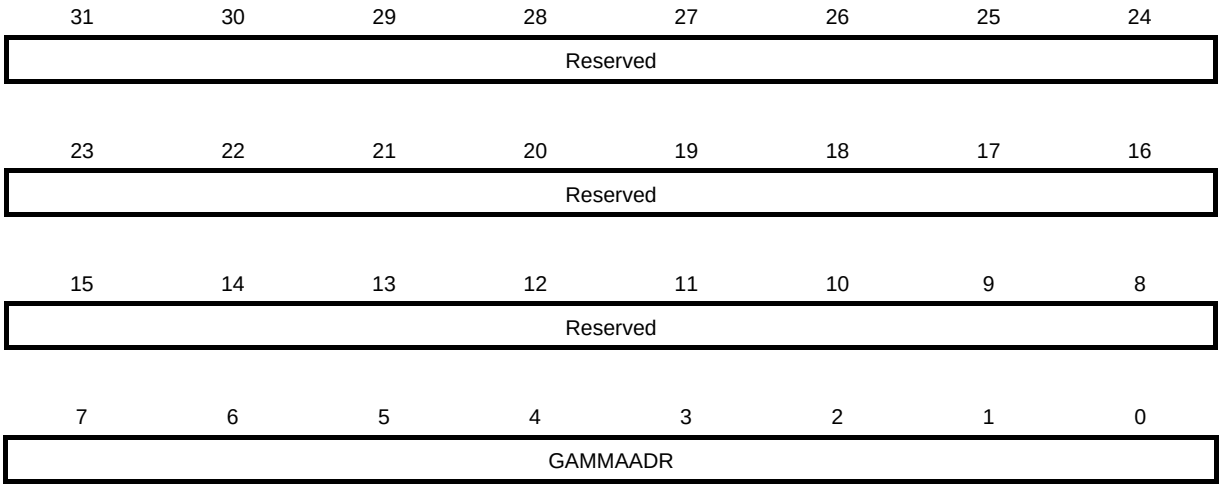
If gamma conversion is disabled for all frames, the privilege for the gamma correction table is passed to the APB interface. The gamma correction table values can only be changed at this time.

2.2.9 Gamma correction table address register (IMC only)

This register (IMC_GAMMA_ADR: E126_0024H) specifies the gamma correction table address to be accessed for reading or writing.

Changes to this register are immediately applied, so changing it during operation is prohibited.

For details about gamma correction, see **3.6 Gamma Correction.**.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
GAMMAADR	R/W	7:0	00H	Specify the gamma correction table address to be accessed.

To accelerate writing to the gamma table, perform write to the IMC_GAMMA_DATA register. The values of the IMC_GAMMA_ADR register are then incremented automatically.

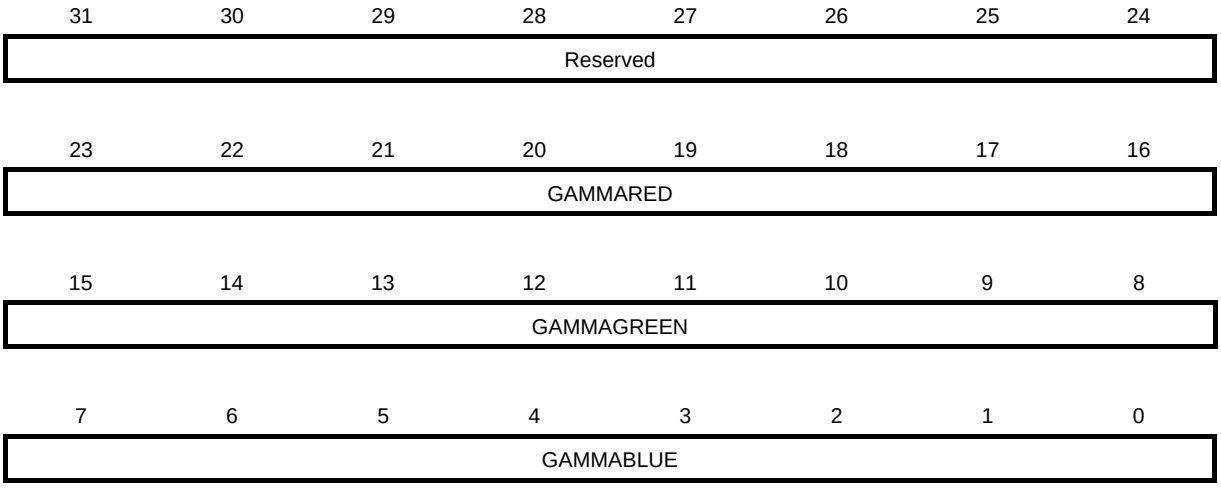
2.2.10 Gamma correction table data register (IMC only)

This register (IMC_GAMMA_DATA: E126_0028H) is used to access the gamma correction table based on the address specified by using GAMMAADR bit of the gamma correction table address register. The gamma correction table cannot be accessed when gamma correction is enabled. Setting this register is prohibited while gamma correction is enabled because the write operation is ignored and the read operation might read out invalid data.

When this register is read, data is latched from the memory at the first read, and is read out by way of the APB bus at the second read. Therefore, be sure to read this register twice in succession and use the data acquired at the second read.

Changes to this register are immediately applied, so changing it during operation is prohibited.

For details about gamma correction, see **3.6 Gamma Correction.**.

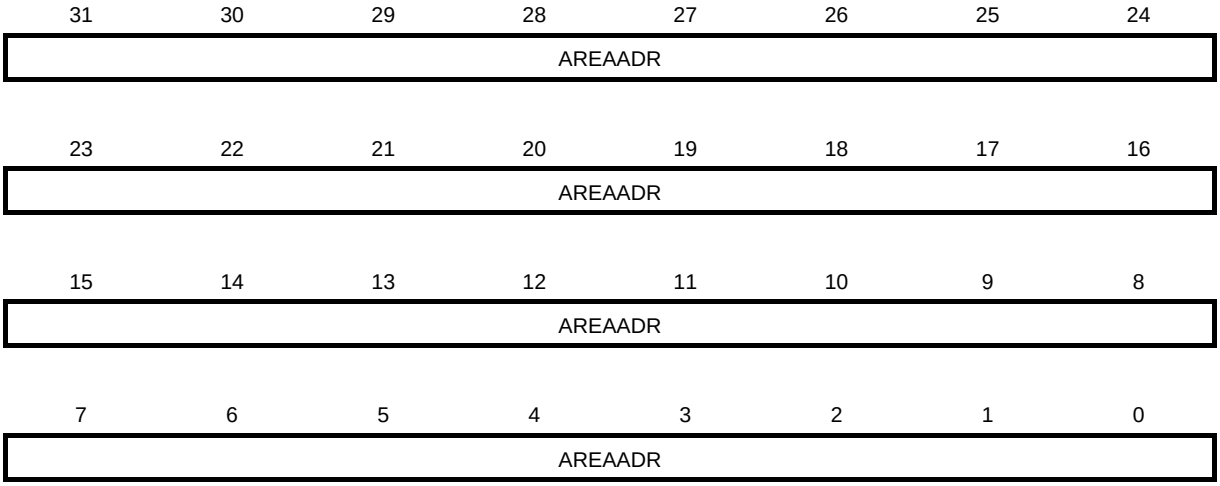


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	X	Reserved. If these bits are read, 0 is returned for each bit.
GAMMARED	R/W	23:16	X	Specify the correction value for the Red data at the address specified by using GAMMAADR.
GAMMAGREEN	R/W	15:8	X	Specify the correction value for the Green data at the address specified by using GAMMAADR.
GAMMABLUE	R/W	7:0	X	Specify the correction value for the Blue data at the address specified by using GAMMAADR.

To accelerate writing to the gamma table, perform write to this register. The values of the IMC_GAMMA_ADR register are then incremented automatically. This setting does not affect the read operation.

2.2.11 Display area address registers

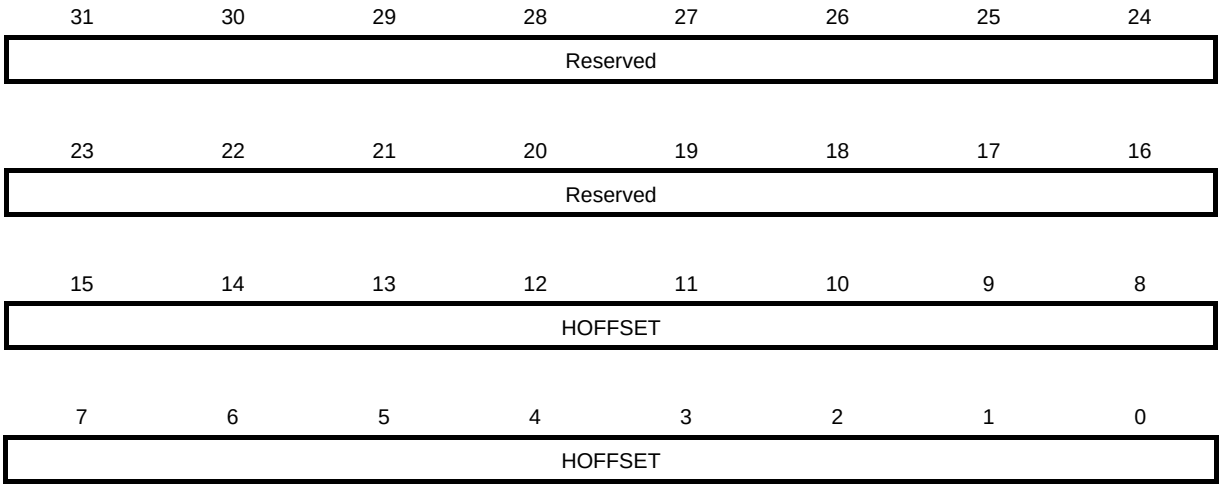
These registers (IMC_WB_AREAADR_P: E126_0040H and IMC_WB_AREAADDR_Q: E126_0050H, IMCW_WB_AREAADR_P: E127_0040H and IMCW_WB_AREAADDR_Q: E127_0050H) specify the start address of the frame buffer area for WB, when the IMC is running in the immediate startup mode. When the IMC is running in the LCD-synchronous mode, the values specified by using this register are ignored. Instead, the address specified by using the LCD_AREAADR_ODD or LCD_AREAADR_EVEN, LCD_HOFFSET, LCD_HAREA, and LCD_VAREA register in the LCD controller is used. At this time, double buffering is disabled depending on the setting of the IMC_WB_BUFSEL and IMC_CPUBUFSEL registers. For the progressive scanning mode, the address specified for the LCD_AREAADR_ODD register is used. For the interlaced scanning mode, the address specified for the LCD_AREAADR_ODD register is used as the odd field, and the address specified for the LCD_AREAADR_EVEN register is used as the even field.
This is a V-sync register.



Name	R/W	Bit No.	After Reset	Description
AREAADR	R/W	31:0	0000_0000H	Specify the first address of the frame buffer. For details, see **Figure 3-1. Frame Buffer Stored in a Memory Space**.

2.2.12 Address addition value register

This register (IMC_WB_HOFFSET: E126_0044H, IMCW_WB_HOFFSET: E127_0044H) specifies the total number of bytes in the horizontal direction of the frame buffer area for WB, when the IMC is running in the immediate startup mode.
Specify the value so that the added amount is not less than the minimum number of bytes required for storing one line of main image data in the memory.
When the IMC is running in the LCD-synchronous mode, the values specified by using this register are ignored. Instead, the value specified by using the LCD_HOFFSET register in the LCD controller is used.
This is a V-sync register.

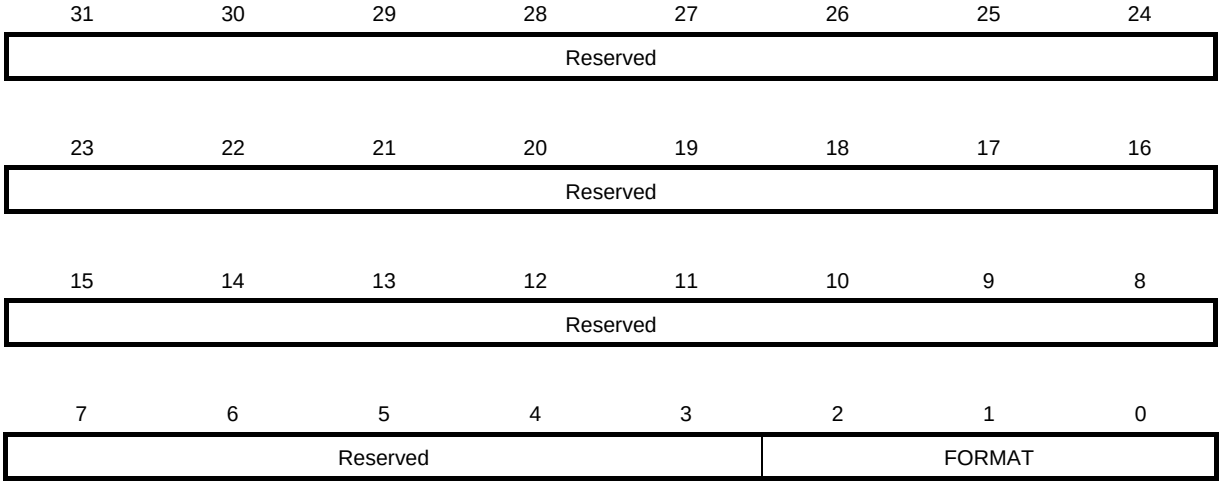


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
HOFFSET	R/W	15:0	0000H	Specify the total byte count in horizontal direction of the frame buffer area.

Note that the value required for storing valid pixel data varies depending on the horizontal pixel count specified by using the WB image size register (IMC_WB_SIZE) and the value specified by using the format register (IMC_WB_FORMAT).

2.2.13 Format register

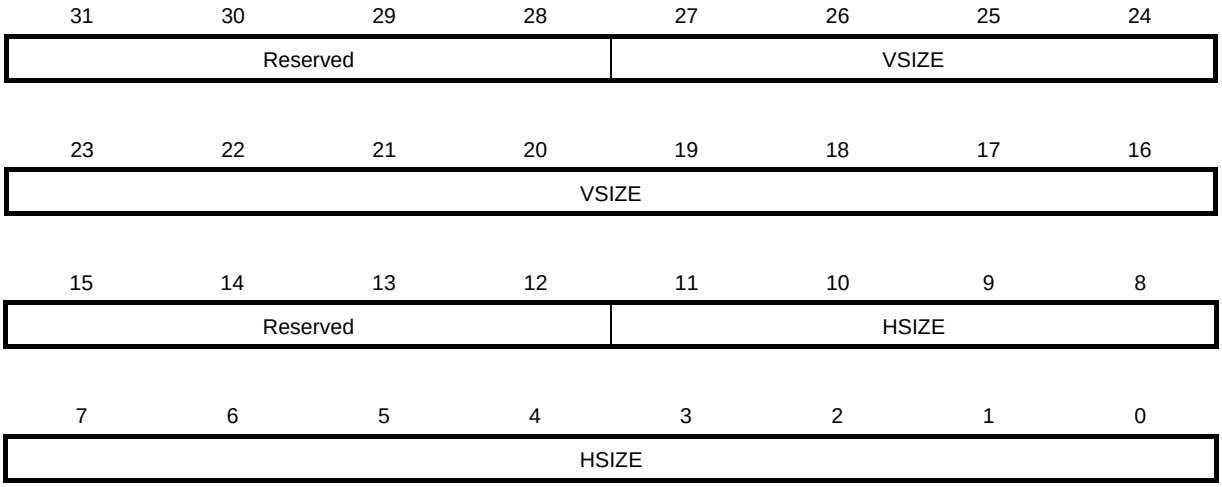
This register (IMC_WB_FORMAT: E126_0048H, IMCW_WB_FORMAT: E127_0048H) specifies the format of data input to the frame buffer area for WB, when the IMC is running in the immediate startup mode.
When the IMC is running in the LCD-synchronous mode, the values specified by using this register are ignored. Instead, the format specified by using the LCD_IFORMAT register of the LCD controller is used.
This is a V-sync register.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:3	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
FORMAT	R/W	2:0	000b	Specify the output data format. 000: RGB888 001: RGB666 010: RGB565 011: Setting prohibited 100: ARGB8888 101: ARGB4444 110 – 111: Reserved

2.2.14 WB image size register

This register (IMC_WB_SIZE: E126_004CH, IMCW_WB_SIZE: E127_004CH) specifies the size of WB images when the IMC is running in the immediate startup mode.
When the IMC is running in the LCD-synchronous mode, the values specified by using this register are ignored. Instead, the image size specified by using the LCD_VAREA and LCD_HAREA registers of the LCD controller is used for VSIZE and HSIZE, respectively.
This is a V-sync register.

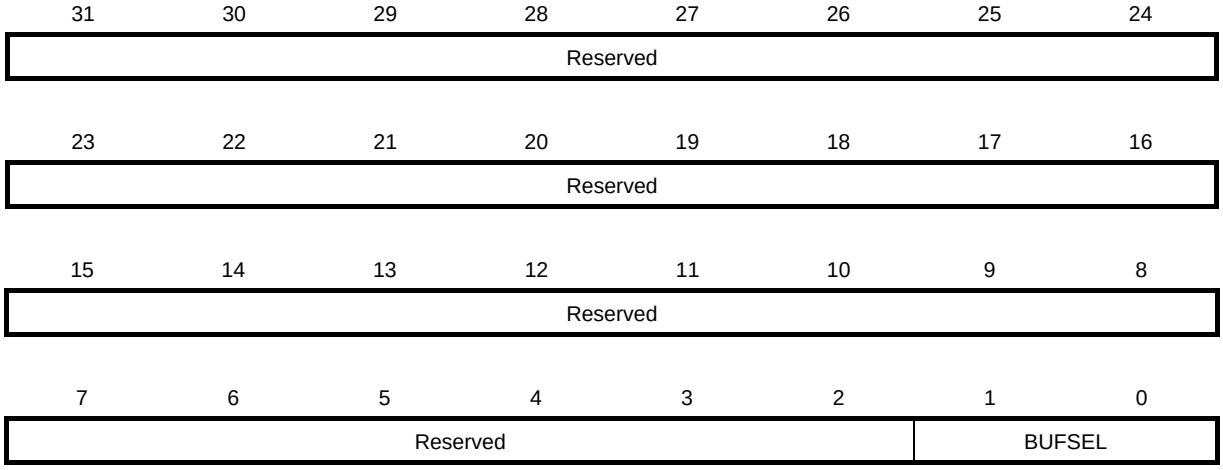


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:28	0H	Reserved. If these bits are read, 0 is returned for each bit.
VSIZE	R/W	27:16	000H	Specify the vertical size of WB images. (up to 4,096 lines, in lines)
Reserved	R	15:12	0H	Reserved. If these bits are read, 0 is returned for each bit.
HSIZE	R/W	11:0	000H	Specify the horizontal size of WB images. (up to 4,096 pixels, in pixels)

Caution When outputting data in the RGB666 format in the immediate startup mode, if wrapping around is enabled or a main image exists, specify an integer that is a multiple of 4 for the HSIZE bit. If you do not, a portion of the main image that you do not want to overwrite might be corrupted.

2.2.15 WB double buffer control register

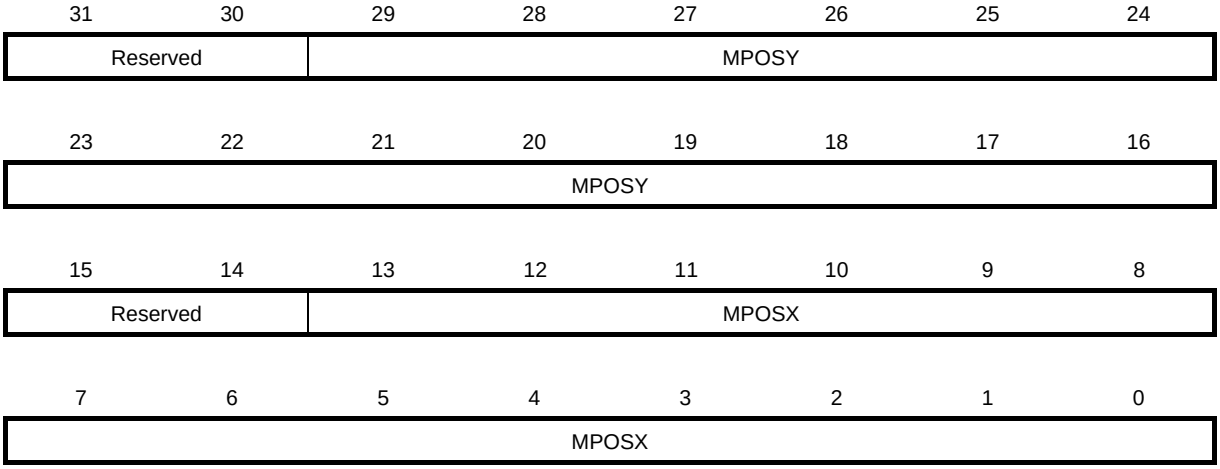
This register (IMC_WB_BUFSEL: E126_0054H, IMCW_WB_BUFSEL: E127_0054H) switches between the P buffer and Q buffer for progressive output or between the odd field and even field for interlaced output when the IMC is running in the immediate startup mode. When the IMC is running in the LCD-synchronous mode, the values specified by using this register are ignored.
This is a V-sync register.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
BUFSEL	R/W	1:0	0H	Select the start address for double buffering or interlaced output. For progressive output 0H: Fixed to the P buffer 1H: Fixed to the Q buffer 2H: CPU control 3H: CPU reverse control For interlaced output 0H: The P buffer is used for the odd field, and the Q buffer is used for the even field. 1H: The P buffer is used for the even field, and the Q buffer is used for the odd field. 2H: The buffer selected by the CPU is used for the odd field, and the unselected buffer is used for the even field. 3H: The buffer selected by the CPU is used for the even field, and the unselected buffer is used for the odd field.

2.2.16 WB memory frame start position register

This register (IMC_WB_MPOSITION: E126_0058H, IMCW_WB_MPOSITION: E127_0058H) specifies the start position of an output sub-image in its main image in pixels when the IMC is running in the immediate startup mode.
When the IMC is running in the LCD-synchronous mode, MPOSX and MPOSY are fixed to 0 in the circuit.
This is a V-sync register.



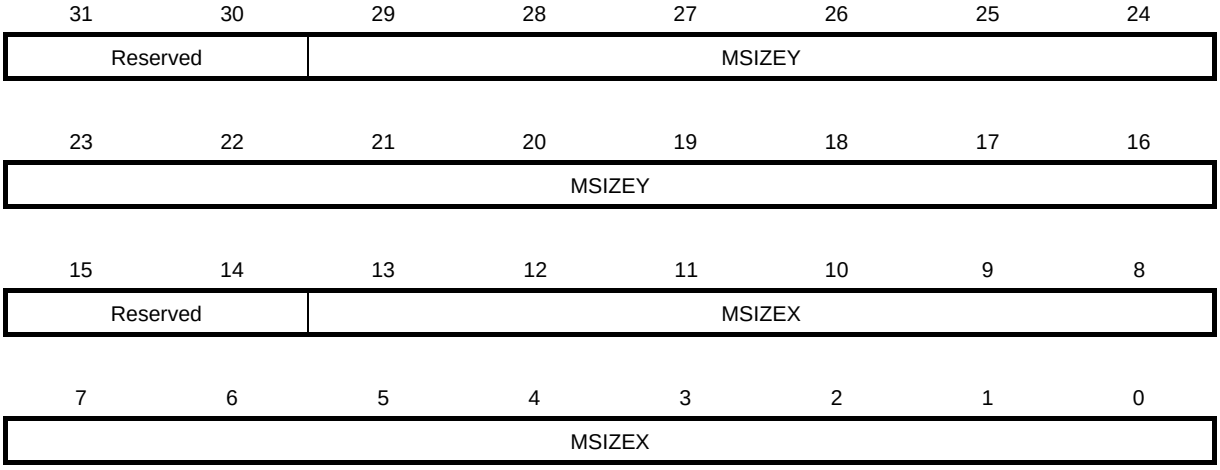
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:30	0H	Reserved. If these bits are read, 0 is returned for each bit.
MPOSY	R/W	29:16	0000H	Specify the vertical start position.
Reserved	R	15:14	0H	Reserved. If these bits are read, 0 is returned for each bit.
MPOSX	R/W	13:0	0000H	Specify the start position in the horizontal direction.

For details about the definition of frame buffers, see **Figure 3-1. Frame Buffer Stored in a Memory Space**.

Caution When outputting data in the RGB666 format in the immediate startup mode, if wrapping around is enabled or a main image exists, specify an integer that is a multiple of 4 for the MPOSX bit. If you do not, a portion of the main image that you do not want to overwrite might be corrupted.

2.2.17 WB memory frame size register

This register (IMC_WB_MSIZE: E126_005CH, IMCW_WB_MSIZE: E127_005CH) specify the vertical and horizontal size of a frame buffer in pixels when the IMC is running in the immediate startup mode.
When the IMC is running in the LCD-synchronous mode, MSIZEX and MSIZEY are fixed to a maximum value of 3FFFH in the circuit.
This is a V-sync register.



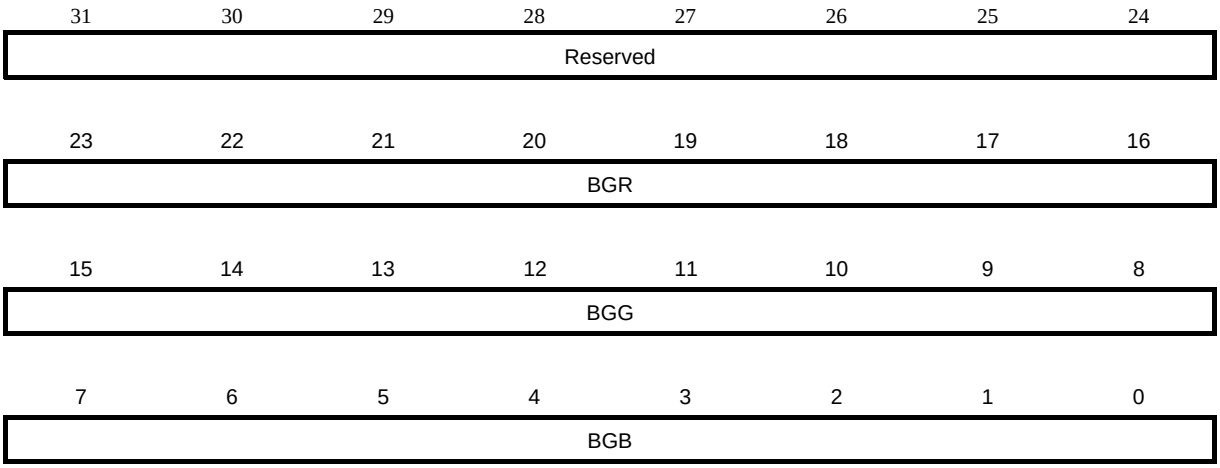
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:30	0H	Reserved. If these bits are read, 0 is returned for each bit.
MSIZEY	R/W	29:16	0000H	Specify the vertical size of a frame buffer in the memory in pixels.
Reserved	R	15:14	0H	Reserved. If these bits are read, 0 is returned for each bit.
MSIZEX	R/W	13:0	0000H	Specify the horizontal size of a frame buffer in the memory in pixels.

For details about the definition of frame buffers, see **Figure 3-1. Frame Buffer Stored in a Memory Space**.

Caution When outputting data in the RGB666 format in the immediate startup mode, if wrapping around is enabled or a main image exists, specify an integer that is a multiple of 4 for the MSIZEX bit. If you do not, a portion of the main image that you do not want to overwrite might be corrupted.

2.2.18 Fixed color register

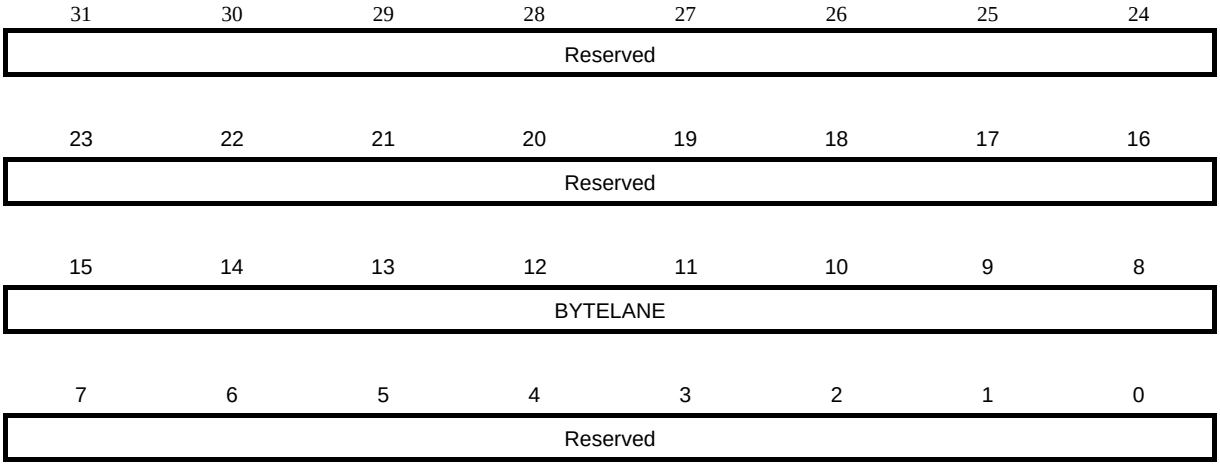
This register (IMC_BACKCOLOR: E126_0060H, IMCW_BACKCOLOR: E127_0060H) specify the fixed background color when the IMC is running in the immediate startup mode. The format matches the IMC output format. For RGB666 or RGB565, fill higher bits during specification.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	00H	Reserved. If these bits are read, 0 is returned for each bit.
BGR	R/W	23:16	00H	Specify the red data specified for the background color.
BGG	R/W	15:8	00H	Specify the green data specified for the background color.
BGB	R/W	7:0	00H	Specify the blue data specified for the background color.

2.2.19 WB byte lane register

This register (IMC_WB_BYTELANE: E126_0064H, IMCW_WB_BYTELANE: E127_0064H) specify changes a ARGB format input picture.



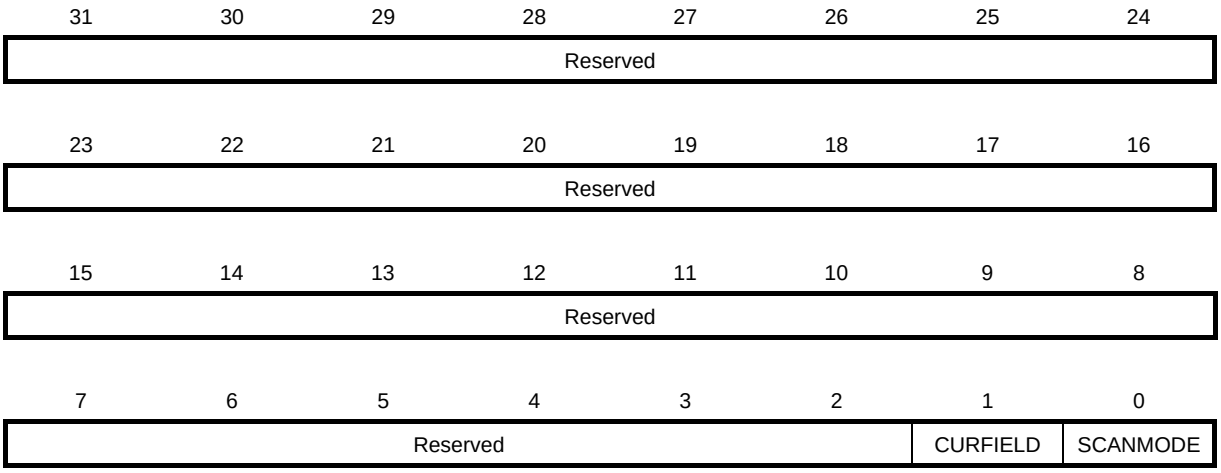
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
BYTELANE	R/W	15:8	E4H	ARGB/RGBA change in an input picture E4H : ARGB/RGB C6H : ABGR/BGR 93H : RGBA 1BH : BGRA Others : Setting prohibited
Reserved	R	7:0	00H	Reserved. If these bits are read, 0 is returned for each bit.

2.2.20 WB output scanning mode register

This register (IMC_WB_SCANMODE: E126_0070H, IMCW_WB_SCANMODE: E127_0070H) specifies the output scanning method.

For details about interlaced and progressive scanning, see **3.3.8 Progressive and interlaced scanning.**

This is a V-sync register.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
CURFIELD	R	1	0	For interlaced output, this bit indicates the field being output. 0: Odd field 1: Even field For progressive output, this bit is meaningless.
SCANMODE	R/W	0	0	Select the output scanning mode. 0: Progressive 1: Interlaced

2.2.21 Output image horizontal/vertical flip setting register

This register (IMC_MIRROR: E126_0100H, IMCW_MIRROR: E127_0100H) is used to flip images output from the IMC (data after synthesis) horizontally or vertically.
Horizontal or vertical flip is performed after layers are superimposed, so care must be exercised when displaying layers including texts.
This is a V-sync register, so the setting takes effect at the beginning of the first frame after setting change.

31	30	29	28	27	26	25	24
Reserved							

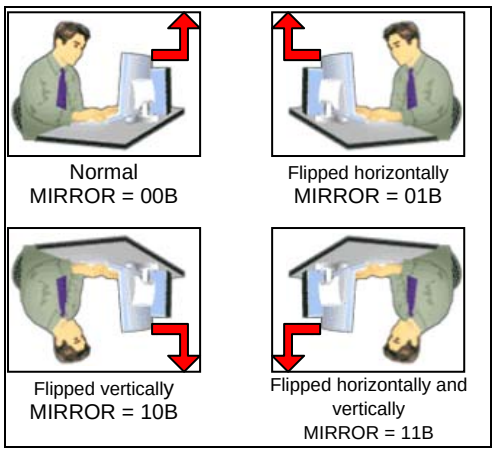
23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved						MIRROR	

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
MIRROR	R/W	1:0	0H	Specify flipping of images. 0: No flip 1: Horizontal flip 2: Vertical flip 3: Horizontal and vertical flip

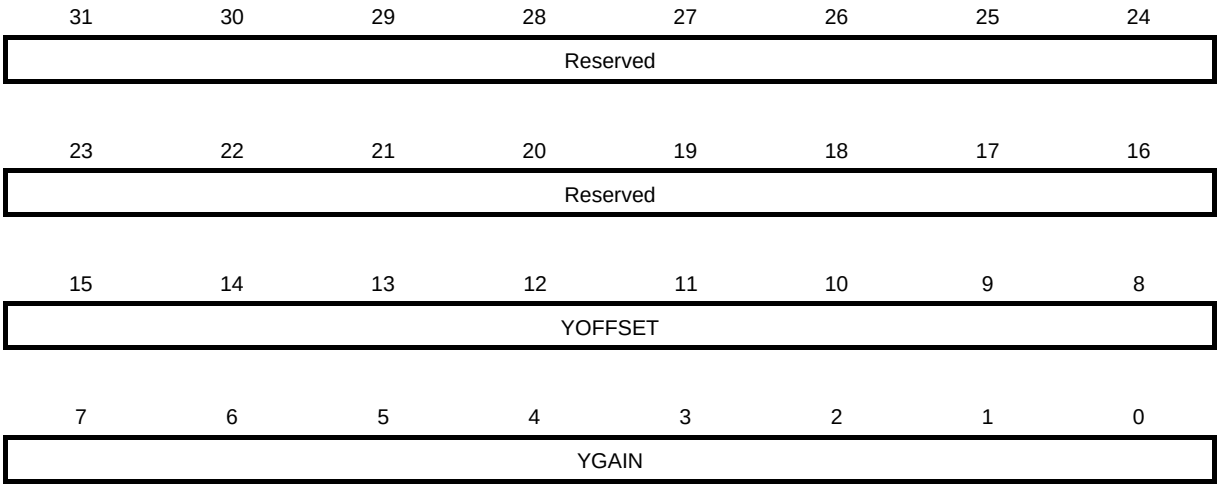
Figure 2-2. Flipping of Images



The flipping effect differs depending on whether the image is input from the AXI master read interface or the AXI slave write interface. See **Figure 3-14. The Effect of Mirroring**.

2.2.22 Y gain offset register (IMC only)

This register (IMC_YGAINOFFSET: E126_0104H) is used to adjust the gain for Y (luminance) by multiplying by 0 to 255/128 and adjusts the offset in the range of –128 to 127, for YUV format images to be input.
The YUV format can be specified for layers 2A and 2B, and the same setting is applied to both layers.
Changes to this register are immediately applied, so changing it during operation is prohibited.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
YOFFSET	R/W	15:8	00H	Specify the offset for the Y value by using signed 8-bit data. (two's complement) (–128 to 127)
YGAIN	R/W	7:0	80H	Specify the offset for the Y value. Values from 0 to 255 can be specified, and the gain is specified value/128.

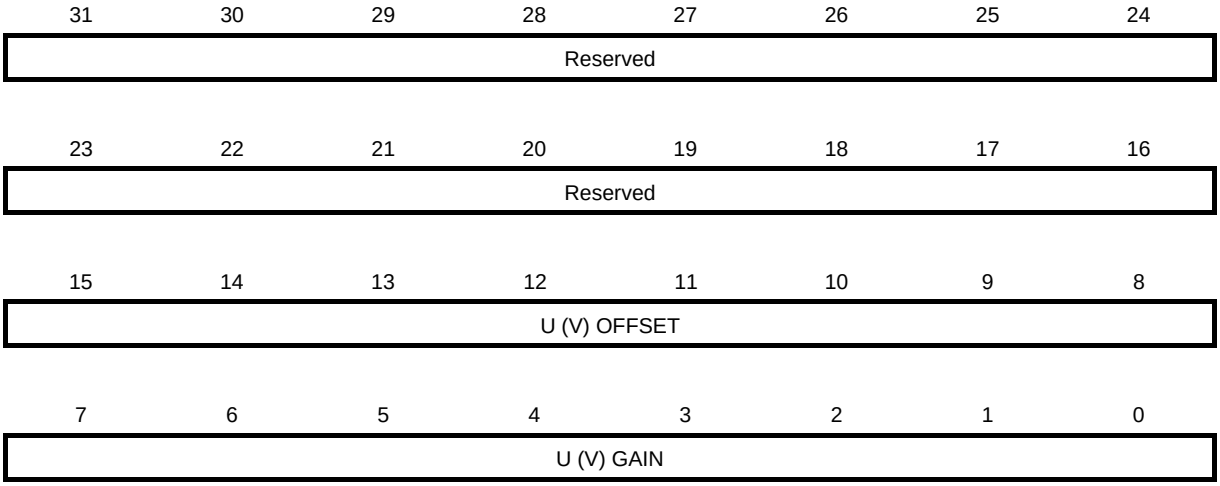
$$Yout = \left(\frac{YGAIN}{128} * Yin \right) + YOFFSET$$

Calculation of desired value: Multiply Yin by YGAIN divided by 128, rounding to one decimal place, and then add YOFFSET to calculate the approximate value (Yout) in the range of 0 to 255.

Caution When YGAIN = 0, the value specified for the YOFFSET bit is used as unsigned 8-bit data (0 to 255), and Yout is assumed to be equal to YOFFSET.

2.2.23 U (V) gain offset registers (IMC only)

These registers (IMC_UGAINOFFSET: E126_0108H and IMC_VGAINOFFSET: E126_010CH) are used to adjust the gain for U (color difference in blue) and V (color difference in red) by multiplying by 0 to 255/128 and the offset in the range of –128 to 127, for YUV format images to be input.
The YUV format can be specified for layers 2A and 2B, and the same setting is applied to both layers.
Because the U and V components are represented in offset binary coding with 80H positioned as the center, 80H is subtracted, converted to values ranging from –128 to 127, the gain is multiplied, and then the offset value and 80H are added.
Changes to this register are immediately applied, so changing it during operation is prohibited.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
UOFFSET (VOFFSET)	R/W	15:8	00H	Specify the offset for the U (V) value by using signed 8-bit data. (–128 to 127)
UGAIN (VGAIN)	R/W	7:0	80H	Specify the offset for the U (V) value. Values from 0 to 255 can be specified, and the gain is specified value/128.

$$Uout = \left(\frac{UGAIN}{128} * (Uin - 80H) \right) + UOFFSET + 80H$$

$$Vout = \left(\frac{VGAIN}{128} * (Vin - 80H) \right) + VOFFSET + 80H$$

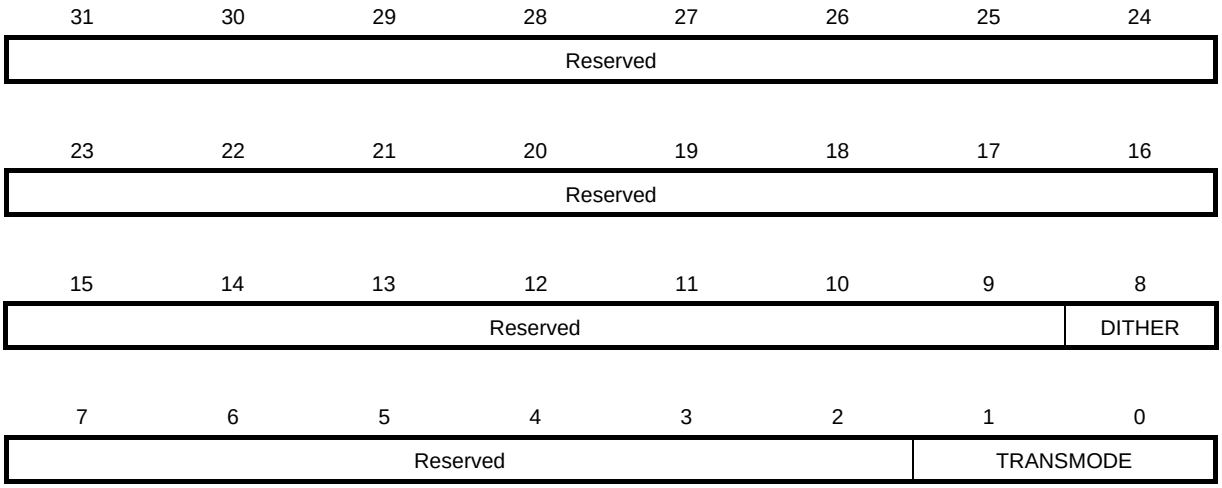
Calculation of desired value: Multiply (Uin – 80H) or (Yin – 80H) by UGAIN or VGAIN divided by 128, rounding to one decimal place, and then add 80H and UOFFSET or VOFFSET to calculate the approximate value in the range of 0 to 255.

Cautions 1. When UGAIN/VGAIN = 0, the value specified for the UOFFSET/VOFFSET bit is used as unsigned 8-bit data (0 to 255) and Uout is assumed to be equal to UOFFSET (Vout = VOFFSET).

2. During multiplication, the fraction is dropped by adding 0.5. For negative values, –0.5 is rounded up to 0.

2.2.24 YUV2RGB conversion mode register

This register (IMC_YUV2RGB: E126_0110H, IMCW_YUV2RGB: E127_0110H) selects a calculation coefficient from ITU-R BT.601-compliant, ITU-R BT.709-compliant, custom coefficient (with or without Y value subtracted by 16) for converting YUV format input data to RGB format data.
The data format after conversion is RGB888, each data consists of 8 bits.



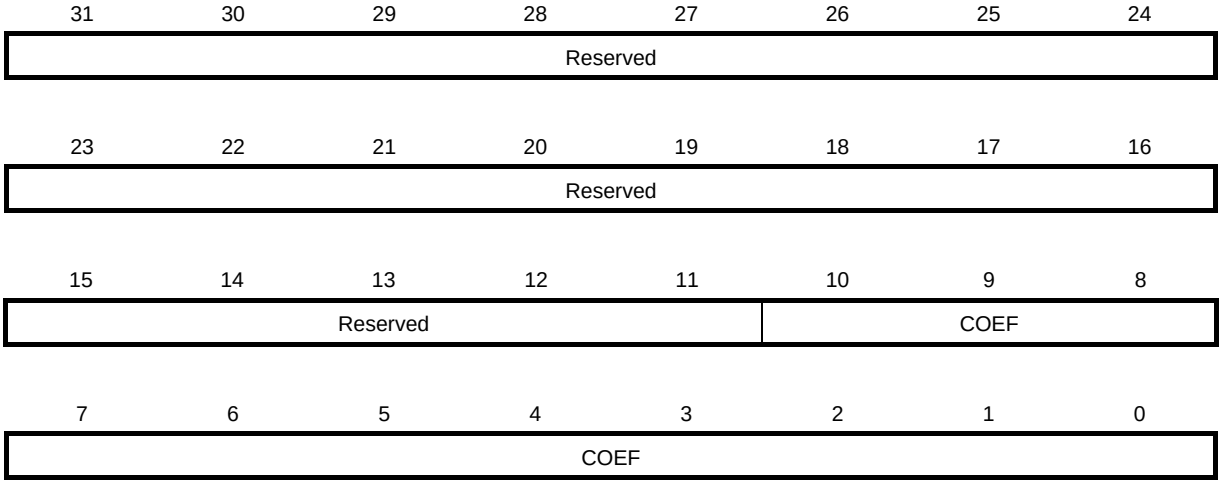
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:9	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
DITHER	R/W	8	0	Enable dithering. For details, see **3.2.7 Dithering.** 0: Do not perform dithering (rounding on or off). 1: Perform dithering.
Reserved	R	7:2	00H	Reserved. If these bits are read, 0 is returned for each bit.
TRANS MODE	R/W	1:0	0H	Select the coefficient for YUV-to-RGB conversion.(IMC only) 0: ITU-R BT.601-compliant 1: ITU-R BT.709-compliant 2: Custom coefficient (Y value is subtracted by 16) 3: Custom coefficient (Y value is used as is)

For details about conversion methods according to the TRANSMODE bit settings,
see **2.2.25 Custom coefficient registers.**

2.2.25 Custom coefficient registers (IMC only)

(IMC_COEF_R0: E126_0114H) (IMC_COEF_G0: E126_0124H) (IMC_COEF_B0: E126_0134H)
(IMC_COEF_R1: E126_0118H) (IMC_COEF_G1: E126_0128H) (IMC_COEF_B1: E126_0138H)
(IMC_COEF_R2: E126_011CH) (IMC_COEF_G2: E126_012CH) (IMC_COEF_B2: E126_013CH)
(IMC_COEF_R3: E126_0120H) (IMC_COEF_G3: E126_0130H) (IMC_COEF_B3: E126_0140H)

These registers specify the coefficient for YUV-to-RGB conversion. The specified values are used for matrix calculation when the TRANSMODE bit is set to 2 or 3. These values are ignored when the TRANSMODE bit is set to 0 or 1.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:11	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
COEF	R/W	10:0	000H	Signed 11-bit data (two's complement) (–1024 to 1023)

YUV-to RGB conversion is calculated with the 4 × 4 matrix calculation.

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R0 & R1 & R2 & R3 \\ G0 & G1 & G2 & G3 \\ B0 & B1 & B2 & B3 \end{pmatrix} * \begin{pmatrix} Y-16 \\ U-128 \\ V-128 \\ 1 \end{pmatrix} * \frac{1}{256}$$

When the TRANSMODE bit of the IMC_YUV2RGB register is set to 3, 16 is not subtracted from the Y value.

$$\begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} R0 & R1 & R2 & R3 \\ G0 & G1 & G2 & G3 \\ B0 & B1 & B2 & B3 \end{pmatrix} * \begin{pmatrix} Y \\ U-128 \\ V-128 \\ 1 \end{pmatrix} * \frac{1}{256}$$

As a result, the RGB values are calculated as follows:

$$R = (R0*(Y-16) + R1*(U-128) + R2*(V-128) + R3)/256$$
$$G = (G0*(Y-16) + G1*(U-128) + G2*(V-128) + G3)/256$$ 16 is subtracted from Y value
$$B = (B0*(Y-16) + B1*(U-128) + B2*(V-128) + B3)/256$$

$$R = (R0 * Y + R1 * (U - 128) + R2 * (V - 128) + R3) / 256$$
$$G = (G0 * Y + G1 * (U - 128) + G2 * (V - 128) + G3) / 256$$
$$B = (B0 * Y + B1 * (U - 128) + B2 * (V - 128) + B3) / 256$$

Y value is used as is

All twelve coefficients can be specified by using signed 11-bit values (–1024 to 1023).
When the TRANSMODE of the IMC_YUV2RGB register is set to 0 (ITU-R BT.601-compliant) or 1 (ITU-R BT.709-compliant), each coefficient is automatically replaced by the following coefficients:

• ITU-R BT601-compliant (TRANS MODE = 0)

$$\begin{pmatrix} R0 & R1 & R2 & R3 \\ G0 & G1 & G2 & G3 \\ B0 & B1 & B2 & B3 \end{pmatrix} = \begin{pmatrix} 298 & 0 & 409 & 0 \\ 298 & -100 & -208 & 0 \\ 298 & 517 & 0 & 0 \end{pmatrix}$$

Conversion is performed based on the above coefficient in the mode in which the Y value is subtracted by 16.

As a result, the RGB values are calculated as follows:

$$R = (298 * (Y - 16) + 0 * U + 409 * V + 0) / 256$$
$$G = (298 * (Y - 16) - 100 * U - 208 * V + 0) / 256$$
$$B = (298 * (Y - 16) + 517 * U + 0 * V + 0) / 256$$

• ITU-R BT709-compliant (TRANS MODE = 1)

$$\begin{pmatrix} R0 & R1 & R2 & R3 \\ G0 & G1 & G2 & G3 \\ B0 & B1 & B2 & B3 \end{pmatrix} = \begin{pmatrix} 298 & 0 & 459 & 0 \\ 298 & -55 & -137 & 0 \\ 298 & 541 & 0 & 0 \end{pmatrix}$$

Conversion is performed based on the above coefficient in the mode in which the Y value is subtracted by 16.

As a result, the RGB values are calculated as follows:

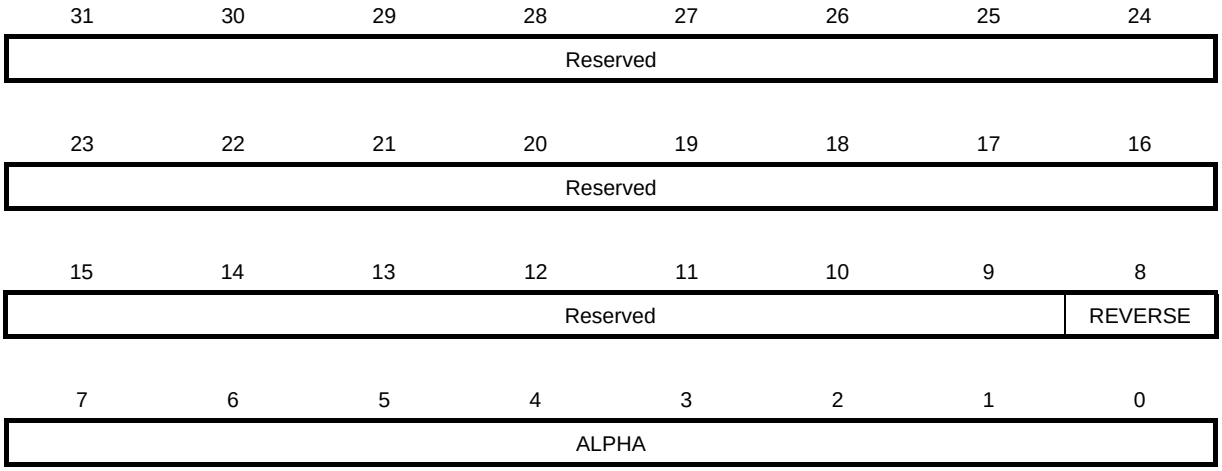
$$R = (298 * (Y - 16) + 0 * U + 459 * V + 0) / 256$$
$$G = (298 * (Y - 16) - 55 * U - 137 * V + 0) / 256$$
$$B = (298 * (Y - 16) + 541 * U + 0 * V + 0) / 256$$

Because the U and V components are represented in offset binary coding in the circuit, with 80H positioned as the center, 128 is subtracted before RGB calculation and substituted into the above formulas. The RGB value after conversion is obtained by rounding the resultant to one decimal place and being clipped in the range of 0 to 255.

2.2.26 Alpha select registers

These registers (IMC_ALPHASEL0: E126_0150H, and IMC_ALPHASEL1: E126_0154H, IMCW_ALPHASEL0: E127_0150H, and IMCW_ALPHASEL1: E127_0154H) specify the ALPHA value for input ARGB1555 images. If A is 0, the value specified using the IMC_ALPHASEL0 register is used. If A is 1, the value specified using the IMC_ALPHASEL1 register is used. This setting is applied to L0, L1A, L1B, and L1C.

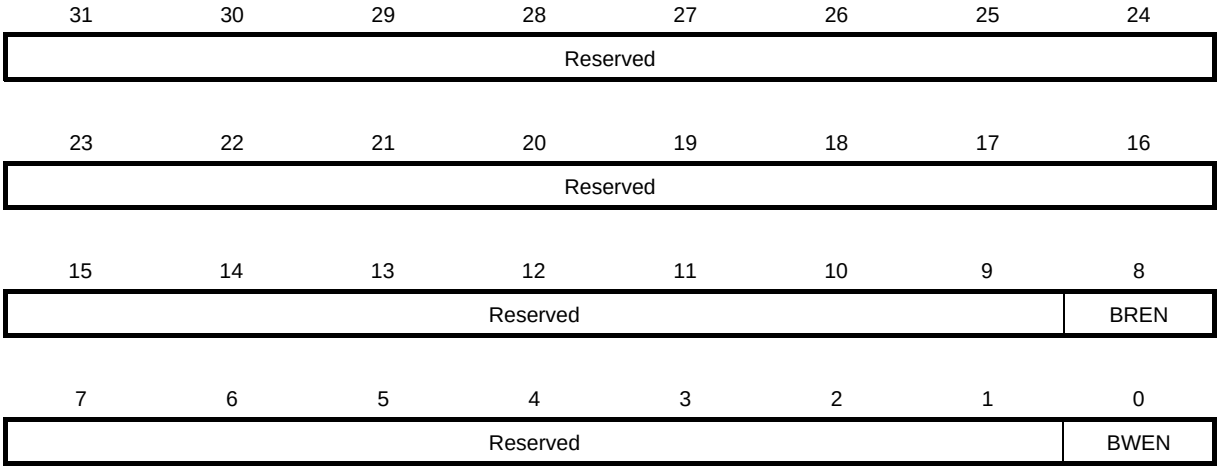
These are V-sync registers, so the setting takes effect at the beginning of the first frame after setting change.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:9	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
REVERSE	R/W	8	0	Reverse the ALPHA setting. Reads 255-ALPHA as newly specified ALPHA.
ALPHA	R/W	7:0	00H	Specify the transparency. Setting range: 0 to 255 (0 = transparent, 255 = opaque) If the REVERSE bit is set to 1, 0 = opaque and 255 = transparent.

2.2.27 Burst enable register

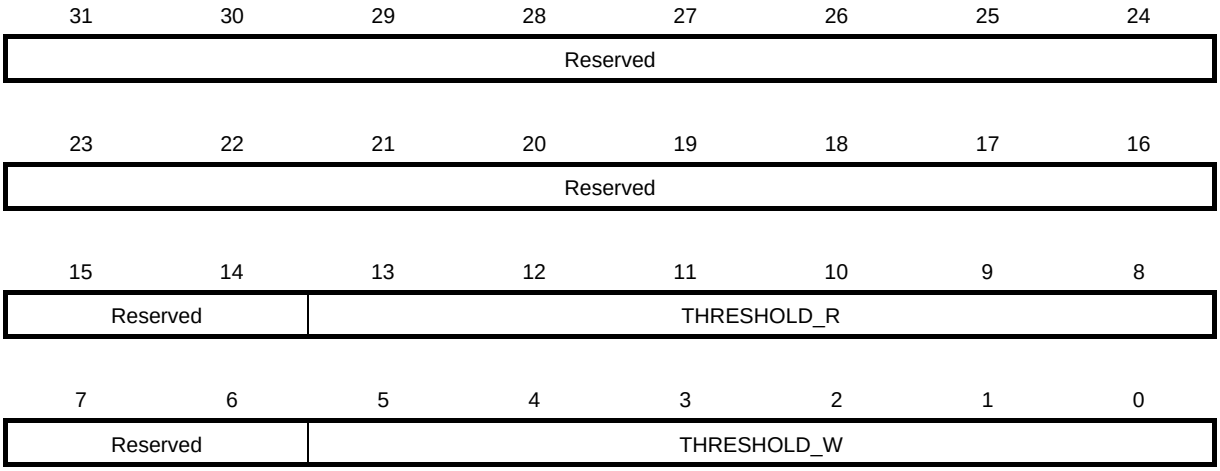
This register (IMC_BURST_EN: E126_0160H, IMCW_BURST_EN: E127_0160H) specifies whether to enable or disable burst transfers for the IMC AXI master read interface and AXI master write interface.
Changes to this register are immediately applied, so changing it during operation is prohibited.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:9	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
BREN	R/W	8	1	Specify whether to enable burst transfers for the AXI master read interface. 0: Non-burst 1: Burst
Reserved	R	7:1	00H	Reserved. If these bits are read, 0 is returned for each bit.
BWEN	R/W	0	1	Specify whether to enable burst transfers for the AXI master write interface. 0: Non-burst 1: Burst

2.2.28 Maximum burst length switching threshold register

This register (IMC_THRESHOLD: E126_0164H, IMCW_THRESHOLD: E127_0164H) specifies the threshold for switching the maximum burst length for the AXI master read interface and AXI master write interface.
The IMC supports HD video playback. For a 64-bit AXI bus that is operating at 266 MHz, if the burst length for the AXI master read interface is 8 words or less, a FIFO underrun might occur during HD display. However, fixing the burst length to 8 words might result in an adverse effect on the performance of another AXI master connected to the system when displaying a small screen.
To enable the system bus to provide sufficient performance without causing an underrun, the IMC has a feature that automatically switches the maximum burst length for the AXI master read interface according to the empty space of the FIFO buffer, as well as the maximum burst length for the AXI master write interface according to the amount of data stored in the FIFO buffer. The threshold can be specified separately for the each interface.
Changes to this register are immediately applied, so changing it during operation is prohibited.

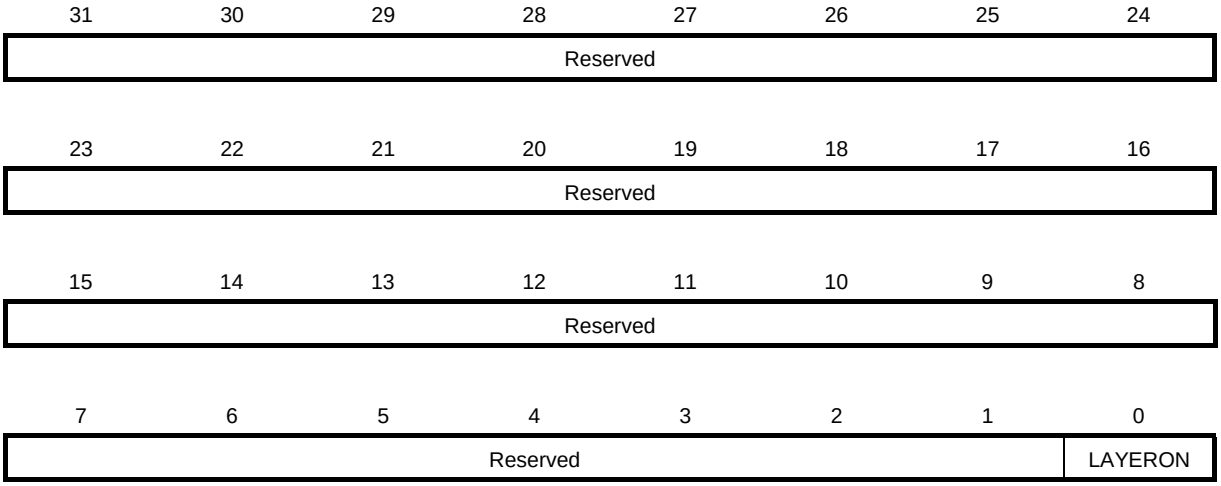


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:14	0_0000H	Reserved. If these bits are read, 0 is returned for each bit.
THRESHOLD_R	R/W	13:8	10H	Specify the maximum burst length for the AXI master read interface. FIFO empty space ≤ THRESHOLD_R: Up to a 4-word burst FIFO empty space > THRESHOLD_R: Up to an 8-word burst
Reserved	R	7:6	0H	Reserved. If these bits are read, 0 is returned for each bit.
THRESHOLD_W	R/W	5:0	10H	Specify the maximum burst length for the AXI master write interface. FIFO data amount > THRESHOLD_W: Up to an 8-word burst FIFO data amount ≤ THRESHOLD_W: Up to a 4-word burst

2.2.29 Input image control registers

L0 input image control register
(IMC_L0_CONTROL: E126_0200H, IMCW_L0_CONTROL: E127_0200H)
L1A input image control register
(IMC_L1A_CONTROL: E126_0300H, IMCW_L1A_CONTROL: E127_0300H)
L1B input image control register
(IMC_L1B_CONTROL: E126_0400H, IMCW_L1B_CONTROL: E127_0400H)
L1C input image control register
(IMC_L1C_CONTROL: E126_0500H, IMCW_L1C_CONTROL: E127_0500H)
L2A input image control register
(IMC_L2A_CONTROL: E126_0600H) (IMC only)
L2B input image control register
(IMC_L2B_CONTROL: E126_0700H) (IMC only)

These registers are used to enable synthesizing frames (the six frames other than the background). Synthesizing the background is always enabled. These are update target registers.
There is no background in case of IMCW. The input picture size is filled into a part smaller than the output picture size by a black picture element. It's recommended always to establish the picture size of the lowest rank hierarchy which is being used and the output picture size in the same way.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
LAYERON	R/W	0	0	Specify whether to enable synthesizing the target layers. 0: Disable layer 1: Enable layer

2.2.30 Layer format registers

L0 format register (IMC_L0_FORMAT: E126_0204H, IMCW_L0_FORMAT: E127_0204H)
L1A format register (IMC_L1A_FORMAT: E126_0304H, IMCW_L1A_FORMAT: E127_0304H)
L1A format register (IMC_L1B_FORMAT: E126_0404H, IMCW_L1B_FORMAT: E127_0404H)
L1A format register (IMC_L1C_FORMAT: E126_0504H, IMCW_L1C_FORMAT: E127_0504H)
L2A format register (IMC_L2A_FORMAT: E126_0604H) (IMC only)
L2B format register (IMC_L2B_FORMAT: E126_0704H) (IMC only)
BG format register (IMC_BG_FORMAT: E126_0804H) (IMC only)

These registers specify the format of input image data to be stored. These are update target registers. The supported format varies in each frame. The available formats are listed in the following tables:

L0 and L1A, L1B, and L1C format registers

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:3	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
FORMAT	R/W	2:0	0H	Specify the format of data to be stored in the target layer. 0: RGB888 4: ARGB8888 1: RGB666 5: ARGB4444 2: RGB565 6: ARGB1555 3: Setting prohibited 7: Setting prohibited

L2A and L2B format registers

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:4	000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
FORMAT	R/W	3:0	0H	Specify the format of data to be stored in the target layer. 0: RGB888 10: YUV422 planar 1: RGB666 11: Setting prohibited 2: RGB565 12: YUV444 3 to 7: Setting prohibited 13: YUV420 semi-planar 8: YUV422 interlaced 14: YUV420 planar 9: YUV422 semi-planar 15: Setting prohibited

BG format register

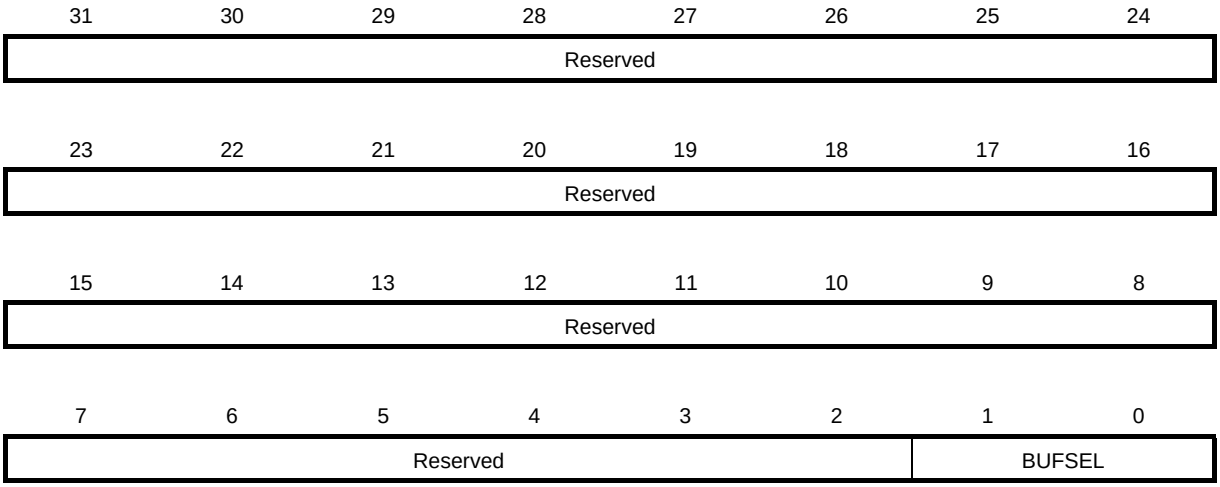
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:3	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
FORMAT	R/W	2:0	0H	Specify the format of data to be stored in the target layer. 0: RGB888 1: RGB666 2: RGB565 3: Fixed to black (ALL0) 4: Fixed background color (specified by using the LCD_BACKCOLOR or IMC_BACKCOLOR register)

L2A and L2B, which are processed during the back process, support the YUV formats. Data of formats that have a different number of planes (described in **Table 3-2. Formats and Start Address Setting Registers**) might be superimposed during the same processing. In this case, the performance might degrade temporarily when data is switched to that of another format that has a different number of planes.

2.2.31 Double buffer control registers

L0 double buffer control register (IMC_L0_BUFSEL: E126_0208H, IMCW_L0_BUFSEL: E127_0208H)
L1A double buffer control register (IMC_L1A_BUFSEL: E126_0308H, IMCW_L1A_BUFSEL: E127_0308H)
L1B double buffer control register (IMC_L1B_BUFSEL: E126_0408H, IMCW_L1B_BUFSEL: E127_0408H)
L1C double buffer control register (IMC_L1C_BUFSEL: E126_0508H, IMCW_L1C_BUFSEL: E127_0508H)
L2A double buffer control register (IMC_L2A_BUFSEL: E126_0608H) (IMC only)
L2B double buffer control register (IMC_L2B_BUFSEL: E126_0708H) (IMC only)
BG double buffer control register (IMC_BG_BUFSEL: E126_0808H) (IMC only)

These registers switch between the P buffer and Q buffer for progressive input or between the odd field and even line field for interlaced output.
These are update target registers.

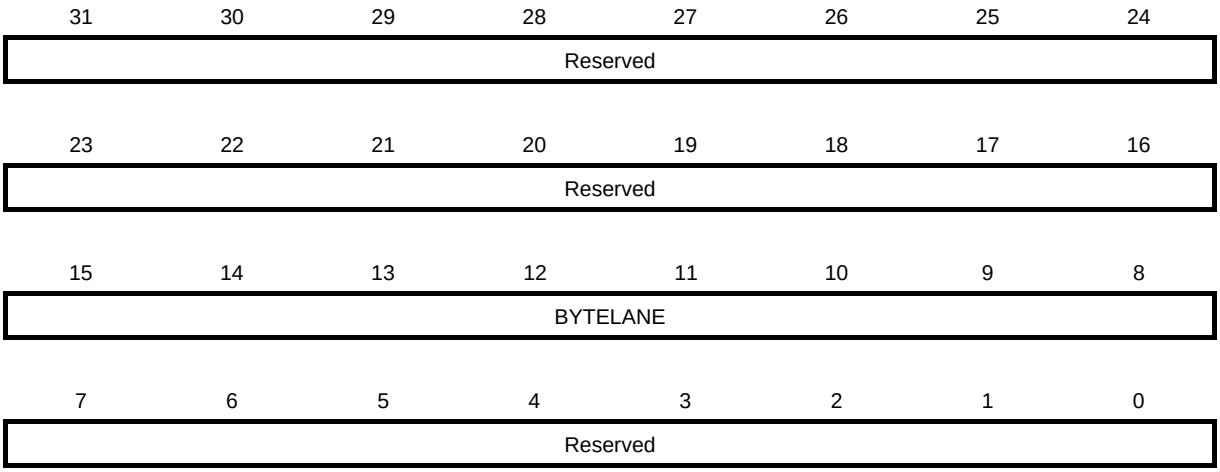


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
BUFSEL	R/W	1:0	0H	For progressive input 0: Fixed to the P buffer 1: Fixed to the Q buffer 2: CPU control 3: Inverted CPU control For interlaced input 0: The P buffer is used for the odd field, and the Q buffer is used for the even field. 1: The P buffer is used for the even field, and the Q buffer is used for the odd field. 2: The buffer selected by the CPU is used for the odd field, and the unselected buffer is used for the even field. 3: The buffer selected by the CPU is used for the even field, and the unselected buffer is used for the odd field.

2.2.32 Byte lane registers

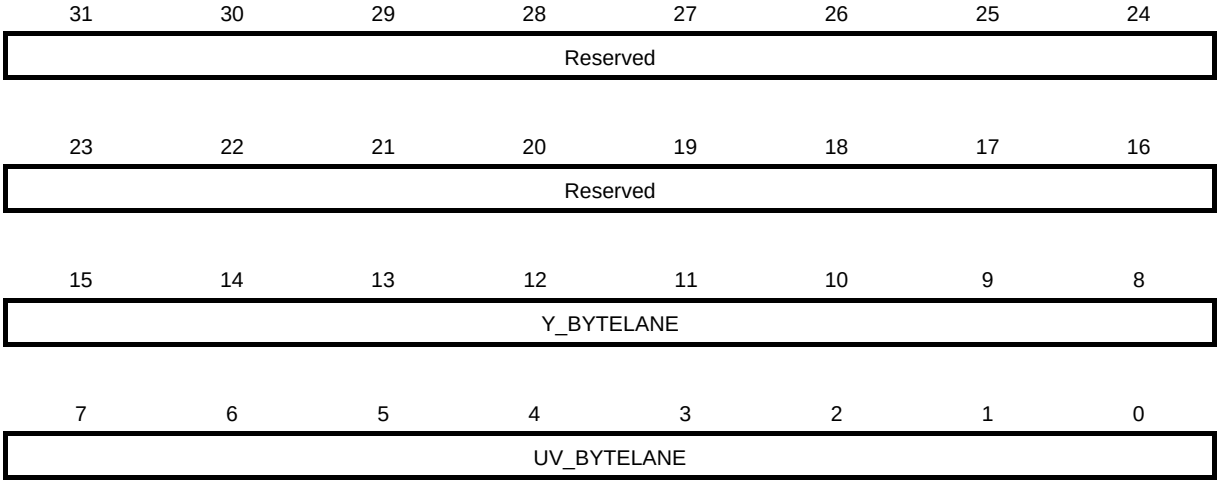
L0 byte lane register (IMC_L0_BYTELANE: E126_020CH, IMCW_L0_BYTELANE: E127_020CH)
L1A byte lane register (IMC_L1A_BYTELANE: E126_030CH, IMCW_L1A_BYTELANE: E127_030CH)
L1B byte lane register (IMC_L1B_BYTELANE: E126_040CH, IMCW_L1B_BYTELANE: E127_040CH)
L1C byte lane register (IMC_L1C_BYTELANE: E126_050CH, IMCW_L1C_BYTELANE: E127_050CH)
L2A byte lane register (IMC_L2A_BYTELANE: E126_060CH) (IMC only)
L2B byte lane register (IMC_L2B_BYTELANE: E126_070CH) (IMC only)
BG byte lane register (IMC_BG_BYTELANE: E126_080CH) (IMC only)

These registers (L0, L1A, L1B, L1C/BG) changes a ARGB format input picture.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
BYTELANE	R/W	15:8	E4H	ARGB/RGBA change in an input picture E4H : ARGB/RGB C6H : ABGR/BGR 93H : RGBA 1BH : BGRA Others : Setting prohibited
Reserved	R	7:0	00H	Reserved. If these bits are read, 0 is returned for each bit.

These registers switch the byte sequence of YUV information in word units when reading YUV format image data of L2A and L2B from the frame buffer. The supported format varies in each frame.



(1/2)

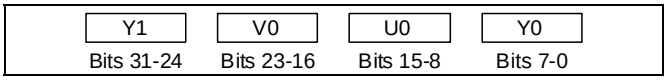
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
Y_BYTE LANE	R/W	15:8	E4H	• In RGB mode ARGB/RGBA change in an input picture 8'hE4 : ARGB/RGB 8'hC6 : ABGR/BGR 8'h93 : RGBA 8'h1B : BGRA Others : Setting prohibited • In YUV Interleave mode Corresponding Y_BYTELANE Bits / Value to Specify 15:14 – Bytes to which V0 is stored (0 to 3) 13:12 – Bytes to which U0 is stored (0 to 3) 11:10 – Bytes to which Y1 is stored (0 to 3) 9:8 – Bytes to which Y0 is stored (0 to 3) • In YUV semi-planar/ planar modes Corresponding Y_BYTELANE Bits / Value to Specify 15:14 – Bytes to which Y3 is stored (0 to 3) 13:12 – Bytes to which Y2 is stored (0 to 3) 11:10 – Bytes to which Y1 is stored (0 to 3) 9:8 – Bytes to which Y0 is stored (0 to 3) • In other modes These values are ignored.

(2/2)

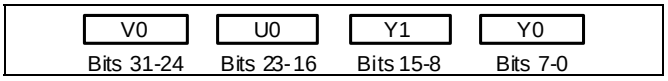
<table>
<tr><th>Name</th><th>R/W</th><th>Bit No.</th><th>After Reset</th><th>Description</th></tr>
<tr><td>UV_BYTE
LANE</td><td>R/W</td><td>7:0</td><td>E4H</td><td>• In YUV semi-planar mode
<table>
<tr><th>Corresponding UV_BYTELANE Bits</th><th>Value to Specify</th></tr>
<tr><td>7:6</td><td>Bytes to which V1 is stored (0 to 3)</td></tr>
<tr><td>5:4</td><td>Bytes to which U1 is stored (0 to 3)</td></tr>
<tr><td>3:2</td><td>Bytes to which V0 is stored (0 to 3)</td></tr>
<tr><td>1:0</td><td>Bytes to which U0 is stored (0 to 3)</td></tr>
</table>
• In YUV planar mode
<table>
<tr><th>Corresponding UV_BYTELANE Bits</th><th>Value to Specify</th></tr>
<tr><td>7:6</td><td>Bytes to which V3/V3 is stored (0 to 3)</td></tr>
<tr><td>5:4</td><td>Bytes to which U2/V2 is stored (0 to 3)</td></tr>
<tr><td>3:2</td><td>Bytes to which U1/V1 is stored (0 to 3)</td></tr>
<tr><td>1:0</td><td>Bytes to which U0/V0 is stored (0 to 3)</td></tr>
</table>
• In other modes
These values are ignored.</td></tr>
</table>

When the YUV format is specified for L2A and L2B, the IMC changes the byte sequence when reading data from the memory, according to the specified BYTELANE value. Specify the BYTELANE value so that the YUV data is reordered as expected by the IMC.

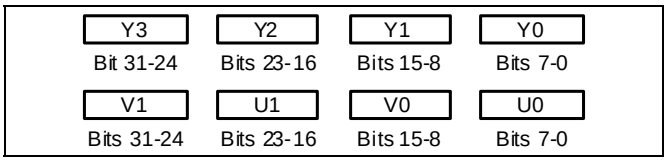
For YUV444, the IMC expects the following byte sequence:



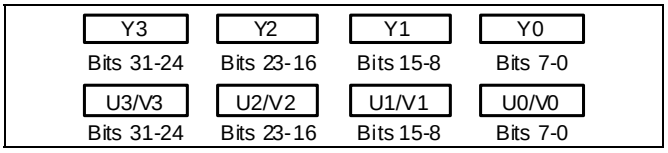
For YUV Interleave, the IMC expects the following byte sequence:



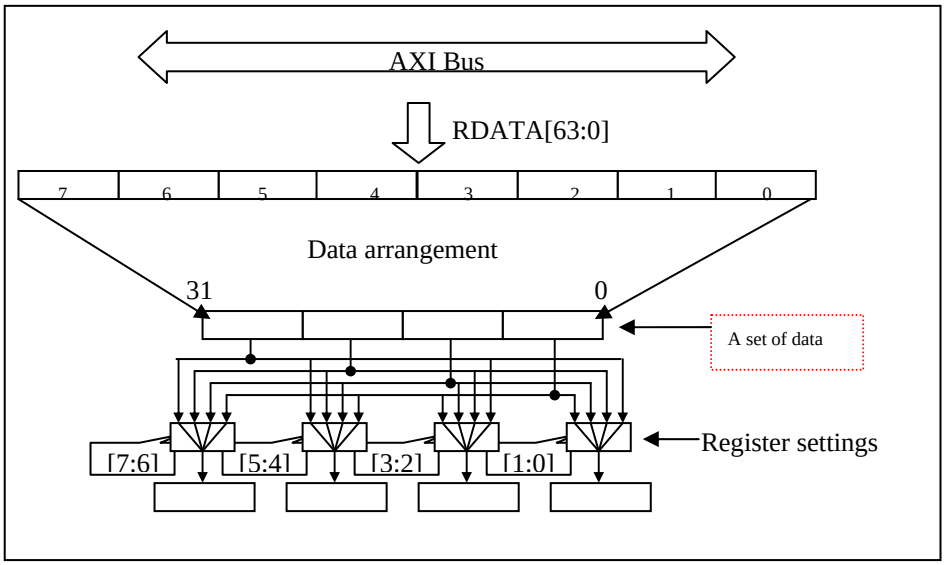
For YUV semi-planar, the IMC expects the following byte sequence:



For YUV planar, the IMC expects the following byte sequence:

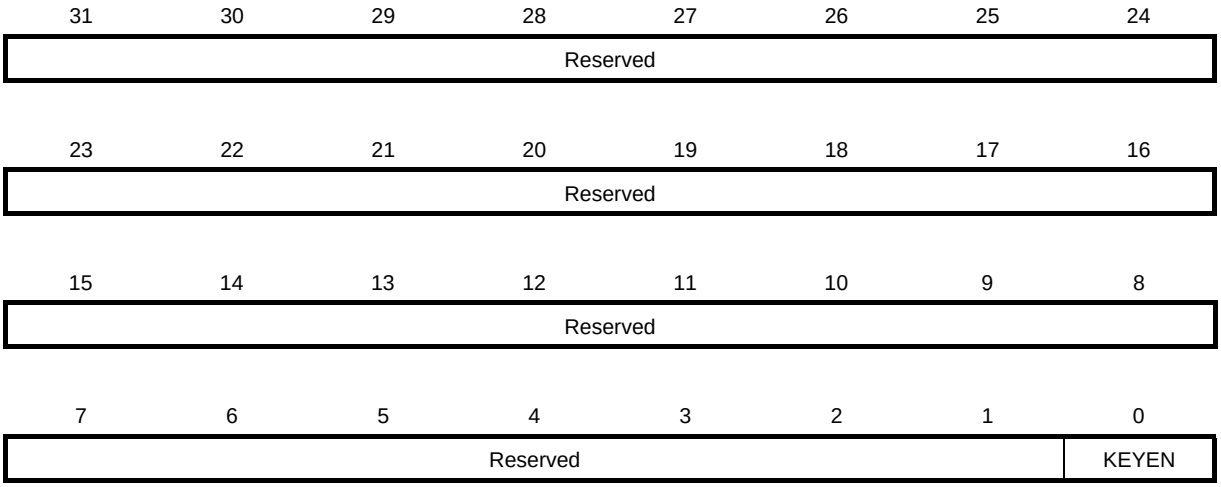


The following shows how data is arranged using the specified BYTELANE values.



2.2.33 Transparency color control registers

These registers (IMC_L0_KEYENABLE: E126_0210H, IMC_L1A_KEYENABLE: E126_0310H, IMC_L1B_KEYENABLE: E126_0410H and IMC_L1C_KEYENABLE: E126_0510H, IMCW_L0_KEYENABLE: E127_0210H, IMCW_L1A_KEYENABLE: E127_0310H, IMCW_L1B_KEYENABLE: E127_0410H and IMCW_L1C_KEYENABLE: E127_0510H) specify whether to enable or disable the transparency color for individual frame (layer 0 and layers 1A to 1C). These are update target registers.

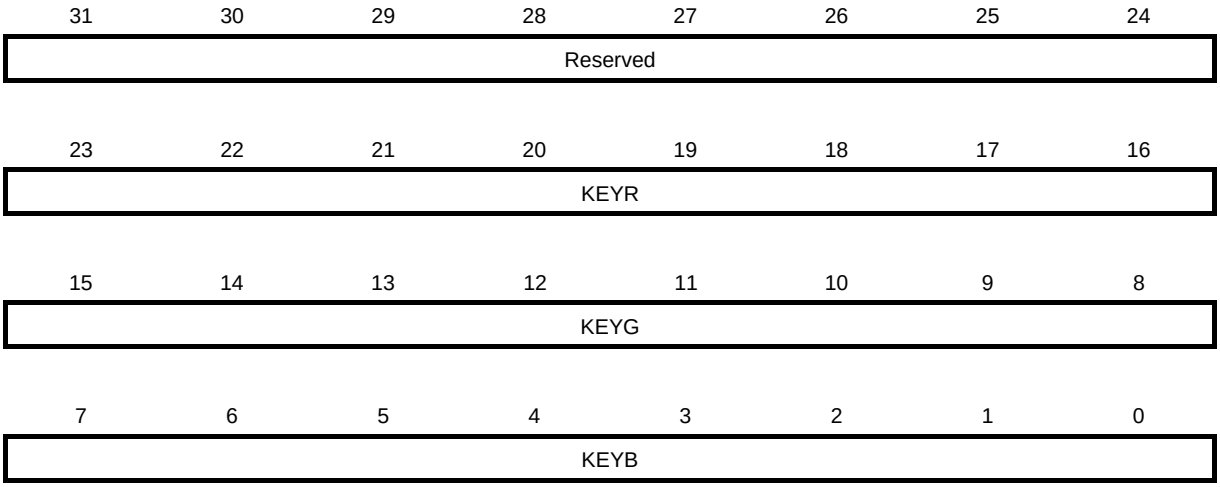


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
KEYEN	R/W	0	0	Specify whether to enable or disable the transparency color for the target layer. 0: Disable the transparency color. 1: Enable on the transparency color.

For details about transparency color, see **3.3.3 Transparency colors (key color)**.

2.2.34 Transparency color registers

These registers (IMC_L0_KEYCOLOR: E126_0214H, IMC_L1A_KEYCOLOR: E126_0314H, IMC_L1B_KEYCOLOR: E126_0414H and IMC_L1C_KEYCOLOR : E126_0514H, IMCW_L0_KEYCOLOR: E127_0214H, IMCW_L1A_KEYCOLOR: E127_0314H, IMCW_L1B_KEYCOLOR: E127_0414H and IMCW_L1C_KEYCOLOR : E127_0514H) specify the key colors used for using the transparency color for the target frame. The format matches the format of each layer. For RGB666, RGB565, ARGB4444, or ARGB1555, fill higher bits for R, G, and B during specification

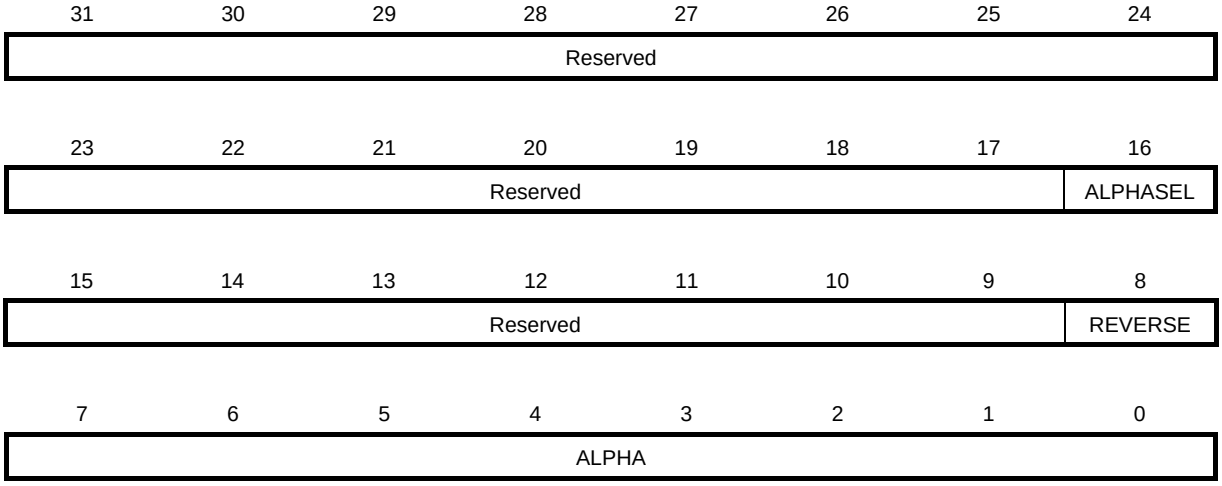


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	00H	Reserved. If these bits are read, 0 is returned for each bit.
KEYR	R/W	23:16	00H	Specify the red data specified for the transparency color.
KEYG	R/W	15:8	00H	Specify the green data specified for the transparency color.
KEYB	R/W	7:0	00H	Specify the blue data specified for the transparency color.

For details about transparency color, see **3.3.3 Transparency colors (key color)**..

2.2.35 Alpha registers

These registers (IMC_L0_ALPHA: E126_0218H, IMC_L1A_ALPHA: E126_0318H, IMC_L1B_ALPHA: E126_0418H and IMC_L1C_ALPHA: E126_0518H,
IMCW_L0_ALPHA: E127_0218H, IMCW_L1A_ALPHA: E127_0318H, IMCW_L1B_ALPHA: E127_0418H and IMCW_L1C_ALPHA: E127_0518H) set up alpha blending for synthesis of the target layers. These are update target registers.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:17	0000H	Reserved. If these bits are read, 0 is returned for each bit.
ALPHASEL	W	16	0	Specify the ALPHA switching signals of ARGB format. When an input format is ARGB8888/ARGB4444/ARGB1555, this register setting is effective. 0: The value of the ARGB format is used. 1: Alpha value is established at this register [8:0].
Reserved	R	15:9	00H	Reserved. If these bits are read, 0 is returned for each bit.
REVERSE	R/W	8	0	Reverse the ALPHA setting. Reads 255-ALPHA as newly specified ALPHA.
ALPHA	R/W	7:0	00H	Specify the transparency. Setting range: 0 to 255 (0 = transparent, 255 = opaque) If the REVERSE bit is set to 1, 0 = opaque and 255 = transparent.

Transparency of 0 to 100% is achieved by specified values ranging from 0 to 255.

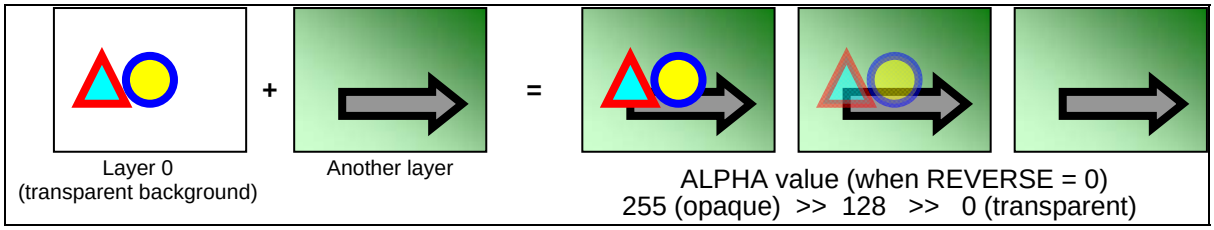
When calculating the desired value, PixOut after synthesis is as below, where pixel data of the layer subject to transparency processing = PixA, and background = PixB.

- If ALPHA = 0 to 254

  PixOut = {ALPHA × PixA + (256 – ALPHA) × PixB} /256 (pixels = 8 bits, with rounding up or down)

- If ALPHA = 255

  PixOut = PixA. The element of PixB is not reflected in the superimposed result.



For details about alpha blending, see **3.3.4 Alpha blending**. The IMC performs transparency color processing first, and then alpha blending.

2.2.36 Resize registers

L0 resize register (IMC_L0_RESIZE: E126_0220H, IMCW_L0_RESIZE: E127_0220H)
L1A resize register (IMC_L1A_RESIZE: E126_0320H, IMCW_L1A_RESIZE: E127_0320H)
L1B resize register (IMC_L1B_RESIZE: E126_0420H, IMCW_L1B_RESIZE: E127_0420H)
L1C resize register (IMC_L1C_RESIZE: E126_0520H, IMCW_L1C_RESIZE: E127_0520H)
L2A resize register (IMC_L2A_RESIZE: E126_0620H) (IMC only)
L2B resize register (IMC_L2B_RESIZE: E126_0720H) (IMC only)
BG resize register (IMC_BG_RESIZE: E126_0820H) (IMC only)

These registers enable resizing the superimposed layers.

Only resizing to double the simple copy is supported. Image data read from a frame buffer is copied to 2 pixels in horizontal and vertical directions and then output. The display area is not doubled during synthesis, but data stored in the frame buffer is enlarged to fit into the specified display area.

These are update target registers. For details, see **3.3.6 Simple resizing.**

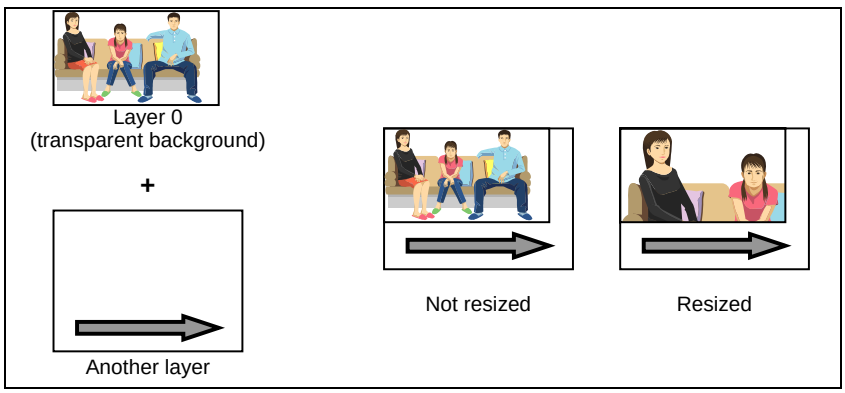
31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved							RESIZE

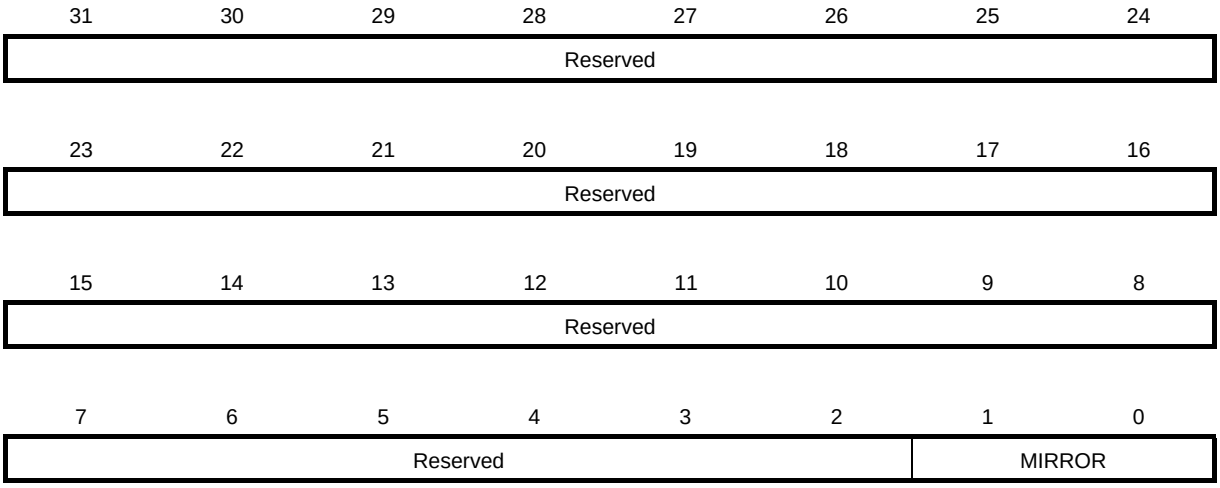
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
RESIZE	R/W	0	0	Specify whether to enable resizing. 0: Disable resizing. 1: Enable resizing.



2.2.37 Horizontal/vertical flip setting registers

L0 horizontal/vertical flip setting register
(IMC_L0_MIRROR: E126_0224H, IMCW_L0_MIRROR: E127_0224H)
L1A horizontal/vertical flip setting register
(IMC_L1A_MIRROR: E126_0324H, IMCW_L1A_MIRROR: E127_0324H)
L1B horizontal/vertical flip setting register
(IMC_L1B_MIRROR: E126_0424H, IMCW_L1B_MIRROR: E127_0424H)
L1C horizontal/vertical flip setting register
(IMC_L1C_MIRROR: E126_0524H, IMCW_L1C_MIRROR: E127_0524H)
L2A horizontal/vertical flip setting register
(IMC_L2A_MIRROR: E126_0624H) (IMC only)
L2B horizontal/vertical flip setting register
(IMC_L2B_MIRROR: E126_0724H) (IMC only)
BG horizontal/vertical flip setting register
(IMC_BG_MIRROR: E126_0824H) (IMC only)

These registers control horizontal/vertical flip of each frame.
These are update target registers.



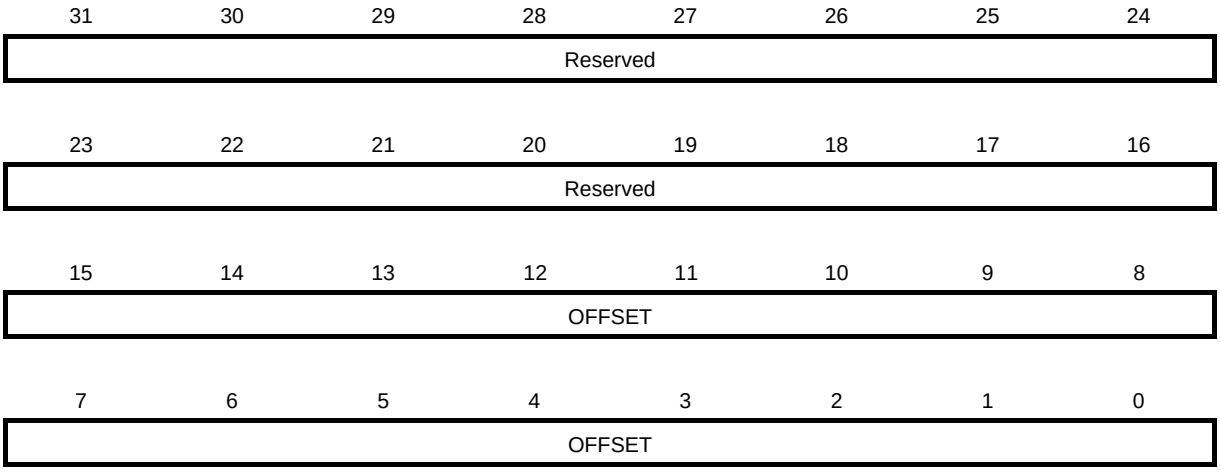
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:2	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
MIRROR	R/W	1:0	0H	Specify the image flip direction. 0: No flip 1: Horizontal flip 2: Vertical flip 3: Horizontal and vertical flip

The flipping effect differs depending on whether the image is input from the AXI master read interface or the AXI slave write interface. See **Figure 3-14. The Effect of Mirroring**.

2.2.38 Input address addition value registers

L0 input address addition value register
(IMC_L0_OFFSET: E126_0230H, IMCW_L0_OFFSET: E127_0230H)
L1A input address addition value register
(IMC_L1A_OFFSET: E126_0330H, IMCW_L1A_OFFSET: E127_0330H)
L1B input address addition value register
(IMC_L1B_OFFSET: E126_0430H, IMCW_L1B_OFFSET: E127_0430H)
L1C input address addition value register
(IMC_L1C_OFFSET: E126_0530H, IMCW_L1C_OFFSET: E127_0530H)
L2A input address addition value register
(IMC_L2A_OFFSET: E126_0630H) (IMC only)
L2B input address addition value register
(IMC_L2B_OFFSET: E126_0730H) (IMC only)
BG input address addition value register
(IMC_BG_OFFSET: E126_0830H) (IMC only)

These registers specify the horizontal size of the frame buffer of the target frames in bytes.
Specify the value so that the added amount is not less than the minimum number of bytes required for storing one line of main image data in the memory.
The values that can be specified depend on the image format and size. For details, see **Table 3-3. Minimum Unit That Can Be Specified for Image Size**.
This is an update target register.



Name	R/W	Bit No.	Reset	Description
Reserved	R	31:16	0000H	Reserved. If these bits are read, 0 is returned for each bit.
OFFSET	R/W	15:0	0000H	Specify the horizontal size of the frame buffer of the target frames in bytes.

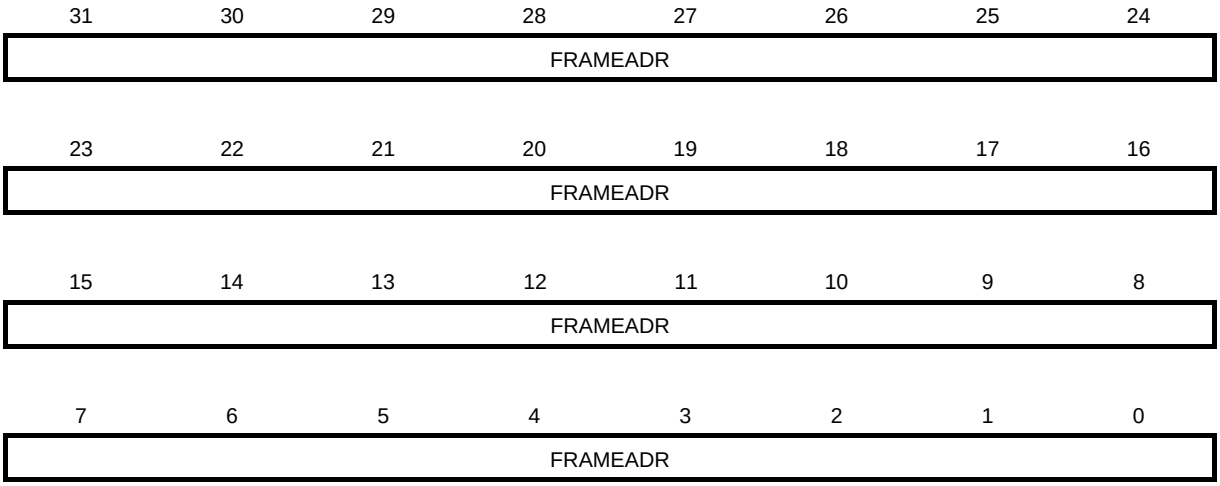
For details about the definition of frame buffers, see **Figure 3-1.** Frame Buffer Stored in a Memory Space.

2.2.39 Frame buffer start address registers

L0 frame buffer start address register (P)
(IMC_L0_FRAMEADR_P: E126_0234H, IMCW_L0_FRAMEADR_P: E127_0234H)
L0 frame buffer start address register (Q)
(IMC_L0_FRAMEADR_Q: E126_0240H, IMCW_L0_FRAMEADR_Q: E127_0240H)
L1A frame buffer start address register (P)
(IMC_L1A_FRAMEADR_P: E126_0334H, IMCW_L1A_FRAMEADR_P: E127_0334H)
L1A frame buffer start address register (Q)
(IMC_L1A_FRAMEADR_Q: E126_0340H, IMCW_L1A_FRAMEADR_Q: E127_0340H)
L1B frame buffer start address register (P)
(IMC_L1B_FRAMEADR_P: E126_0434H, IMCW_L1B_FRAMEADR_P: E127_0434H)
L1B frame buffer start address register (Q)
(IMC_L1B_FRAMEADR_Q: E126_0440H, IMCW_L1B_FRAMEADR_Q: E127_0440H)
L1C frame buffer start address register (P)
(IMC_L1C_FRAMEADR_P: E126_0534H, IMCW_L1C_FRAMEADR_P: E127_0534H)
L1C frame buffer start address register (Q)
(IMC_L1C_FRAMEADR_Q: E126_0540H, IMCW_L1C_FRAMEADR_Q: E127_0540H)
L2A frame buffer start address register (YP)
(IMC_L2A_FRAMEADR_YP: E126_0634H) (IMC only)
L2A frame buffer start address register (UP)
(IMC_L2A_FRAMEADR_UP: E126_0638H) (IMC only)
L2A frame buffer start address register (VP)
(IMC_L2A_FRAMEADR_VP: E126_063CH) (IMC only)
L2A frame buffer start address register (YQ)
(IMC_L2A_FRAMEADR_YQ: E126_0640H) (IMC only)
L2A frame buffer start address register (UQ)
(IMC_L2A_FRAMEADR_UQ: E126_0644H) (IMC only)
L2A frame buffer start address register (VQ)
(IMC_L2A_FRAMEADR_VQ: E126_0648H) (IMC only)
L2B frame buffer start address register (YP)
(IMC_L2B_FRAMEADR_YP: E126_0734H) (IMC only)
L2B frame buffer start address register (UP)
(IMC_L2B_FRAMEADR_UP: E126_0738H) (IMC only)
L2B frame buffer start address register (VP)
(IMC_L2B_FRAMEADR_VP: E126_073CH) (IMC only)
L2B frame buffer start address register (YQ)
(IMC_L2B_FRAMEADR_YQ: E126_0740H) (IMC only)
L2B frame buffer start address register (UQ)
(IMC_L2B_FRAMEADR_UQ: E126_0744H) (IMC only)
L2B frame buffer start address register (VQ)
(IMC_L2B_FRAMEADR_VQ: E126_0748H) (IMC only)
BG frame buffer start address register (P)
(IMC_BG_FRAMEADR_P: E126_0834H) (IMC only)
BG frame buffer start address register (Q)
(IMC_BG_FRAMEADR_Q: E126_0840H) (IMC only)

These registers specify the start address of the frame buffer of the target frames. All frames have two buffers. Which buffer is used can be selected by using the double buffer control register. For details, see **2.2.31 Double buffer control registers**.

These are update target registers.



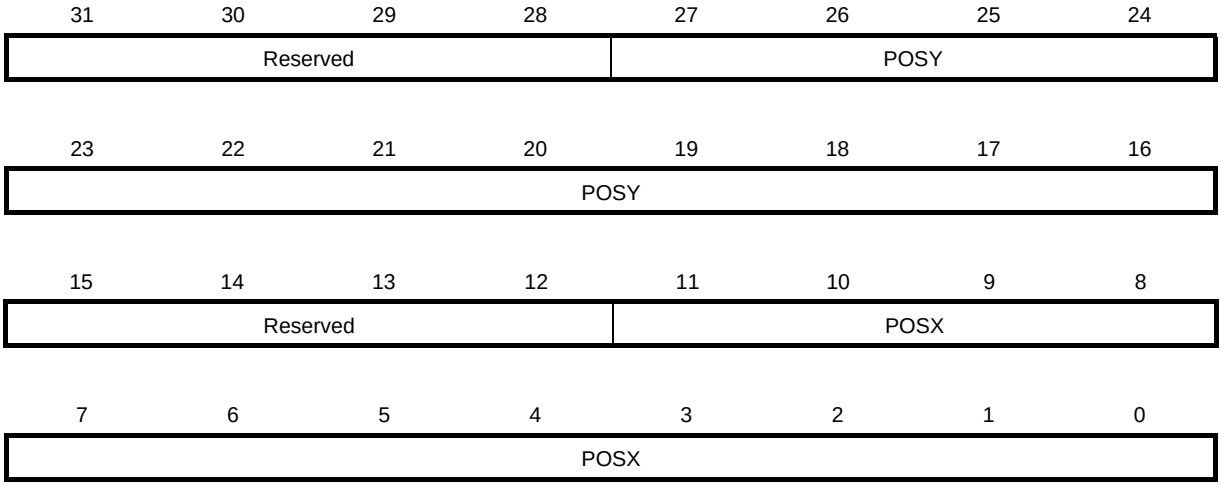
Name	R/W	Bit No.	After Reset	Description
FRAMEADR	R/W	31:0	0000_0000H	Specify the start address of the frame buffer of the target layer.

For details about the definition of frame buffers, see Figure 3-1. Frame Buffer Stored in a Memory Space. If the address of the plane selected by using the double buffer control register and CPU double buffer control register is specified in the corresponding address area (IMC : 0xC00000000 – 0xC7FFFFFFF, IMCW : 0xC8000000 – 0xCFFFFFFF) for the AXI slave write interface, the plane is started by way of the AXI slave write interface. For details about the usage and restrictions for the slave write interface, see **3.4 Inputting Images by Using the Slave Write Interface.**

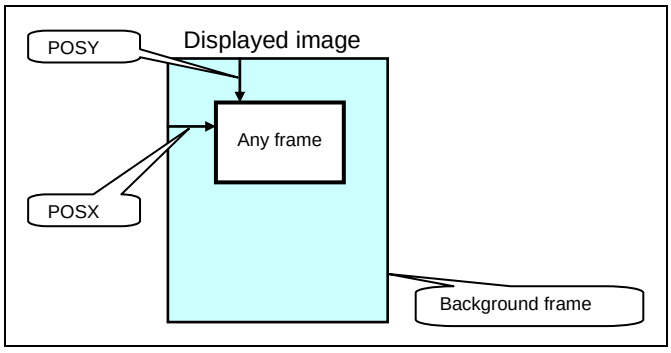
2.2.40 Display position registers

L0 display position register (IMC_L0_POSITION: E126_0250H, IMCW_L0_POSITION: E127_0250H)
L1A display position register (IMC_L1A_POSITION: E126_0350H, IMCW_L1A_POSITION: E127_0350H)
L1B display position register (IMC_L1B_POSITION: E126_0450H, IMCW_L1B_POSITION: E127_0450H)
L1C display position register (IMC_L1C_POSITION: E126_0550H, IMCW_L1C_POSITION: E127_0550H)
L2A display position register (IMC_L2A_POSITION: E126_0650H) (IMC only)
L2B display position register (IMC_L2B_POSITION: E126_0750H) (IMC only)

These registers specify where to superimpose the target frames in pixels.
These are update target registers



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:28	0H	Reserved. If these bits are read, 0 is returned for each bit.
POSY	R/W	27:16	000H	Specify the Y coordinate at which frame display begins. (0 to 4,094, in pixels)
Reserved	R	15:12	0H	Reserved. If these bits are read, 0 is returned for each bit.
POSX	R/W	11:0	000H	Specify the X coordinate at which frame display begins. (0 to 4,094, in pixels)



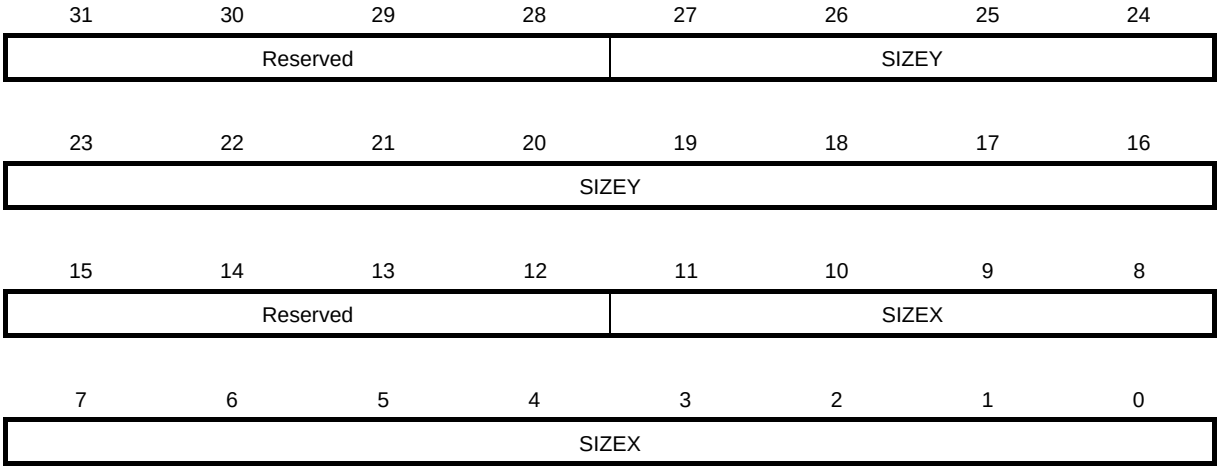
The position of the BG frame is fixed to (0, 0).

2.2.41 Display size registers

L0 display size register	(IMC_L0_SIZE: E126_0254H, IMCW_L0_SIZE: E127_0254H)
L1A display size register	(IMC_L1A_SIZE: E126_0354H, IMCW_L1A_SIZE: E127_0354H)
L1B display size register	(IMC_L1B_SIZE: E126_0454H, IMCW_L1B_SIZE: E127_0454H)
L1C display size register	(IMC_L1C_SIZE: E126_0554H, IMCW_L1C_SIZE: E127_0554H)
L2A display size register	(IMC_L2A_SIZE: E126_0654H) (IMC only)
L2B display size register	(IMC_L2B_SIZE: E126_0754H) (IMC only)

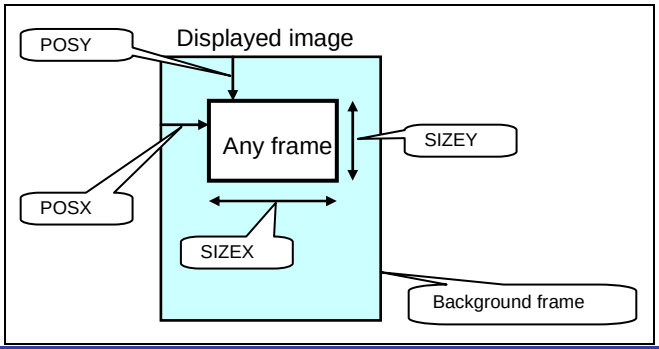
These registers specify the display size of the superimposed frames in pixels.
These are update target registers.
The values that can be specified depend on the scanning mode.
See **Table 3-3. Minimum Unit That Can Be Specified for Image Size**.

Frame indication size register besides BG



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:28	0H	Reserved. If these bits are read, 0 is returned for each bit.
SIZEY	R/W	27:16	000H	Specify the vertical size of the displayed frames in pixels. A value from 0 to 4,094 can be specified.
Reserved	R	15:12	0H	Reserved. If these bits are read, 0 is returned for each bit.
SIZEX	R/W	11:0	000H	Specify the horizontal size of the displayed frames in pixels. A value from 0 to 4,094 can be specified.

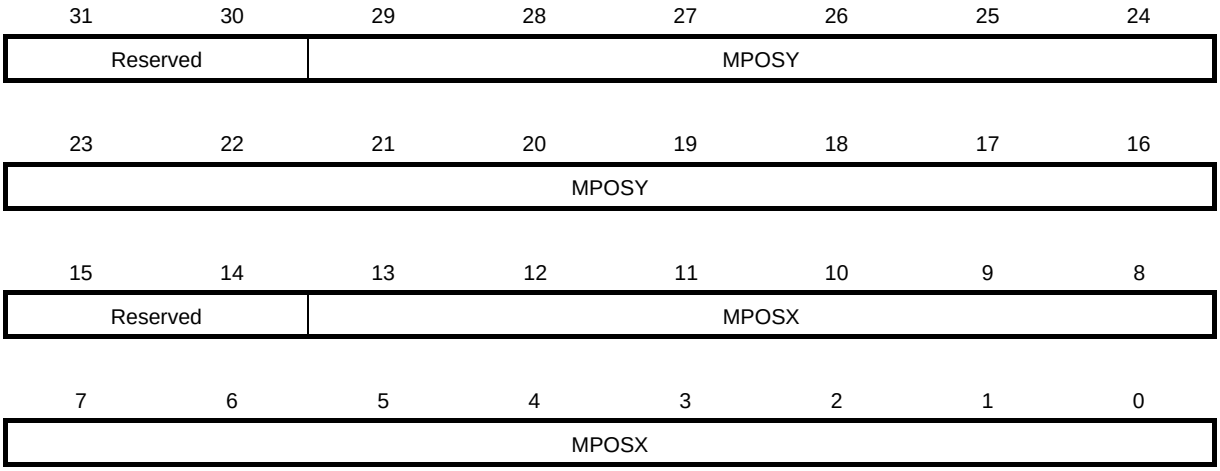
The size of the displayed BG frame is the same as the output image size.



2.2.42 Memory frame start position registers

L0 frame start position register
(IMC_L0_MPOSITION: E126_0260H, IMCW_L0_MPOSITION: E127_0260H)
L1A frame start position register
(IMC_L1A_MPOSITION: E126_0360H, IMCW_L1A_MPOSITION: E127_0360H)
L1B frame start position register
(IMC_L1B_MPOSITION: E126_0460H, IMCW_L1B_MPOSITION: E127_0460H)
L1C frame start position register
(IMC_L1C_MPOSITION: E126_0560H, IMCW_L1C_MPOSITION: E127_0560H)
L2A frame start position register
(IMC_L2A_MPOSITION: E126_0660H) (IMC only)
L2B frame start position register
(IMC_L2B_MPOSITION: E126_0760H) (IMC only)
BG frame start position register
(IMC_BG_MPOSITION: E126_0860H) (IMC only)

These registers specify the vertical and horizontal distance of a displayed frame sub-image from the position of its main image in lines and pixels, respectively.



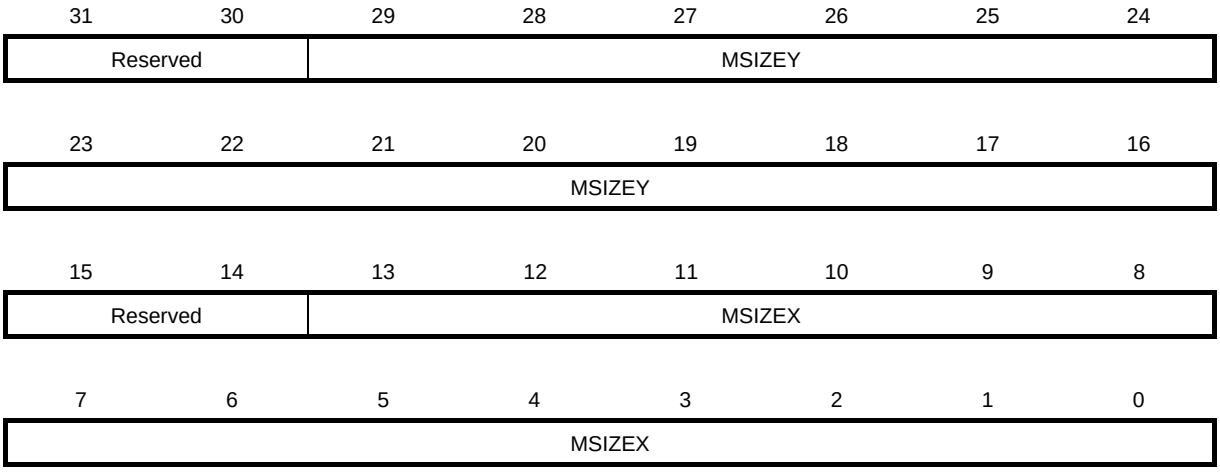
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:30	0H	Reserved. If these bits are read, 0 is returned for each bit.
MPOSY	R/W	29:16	0000H	Specify the vertical distance of a sub-image from the position of its main image in lines.
Reserved	R	15:14	0H	Reserved. If these bits are read, 0 is returned for each bit.
MPOSX	R/W	13:0	0000H	Specify the horizontal distance of a sub-image from the position of its main image in pixels.

For details about the definition of frame buffers, see **Figure 3-1. Frame Buffer Stored in a Memory Space**.

2.2.43 Memory frame buffer size registers

L0 memory frame size register (IMC_L0_MSIZE: E126_0264H, IMCW_L0_MSIZE: E127_0264H)
L1A memory frame size register (IMC_L1A_MSIZE: E126_0364H, IMCW_L1A_MSIZE: E127_0364H)
L1B memory frame size register (IMC_L1B_MSIZE: E126_0464H, IMCW_L1B_MSIZE: E127_0464H)
L1C memory frame size register (IMC_L1C_MSIZE: E126_0564H, IMCW_L1C_MSIZE: E127_0564H)
L2A memory frame size register (IMC_L2A_MSIZE: E126_0664H) (IMC only)
L2B memory frame size register (IMC_L2B_MSIZE: E126_0764H) (IMC only)
BG memory frame size register (IMC_BG_MSIZE: E126_0864H) (IMC only)

These registers specify the vertical and horizontal size of a frame buffer (main image) in lines and pixels, respectively.



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:30	0H	Reserved. If these bits are read, 0 is returned for each bit.
MSIZEY	R/W	29:16	0000H	Specify the vertical size of the main image in lines.
Reserved	R	15:14	0H	Reserved. If these bits are read, 0 is returned for each bit.
MSIZEX	R/W	13:0	0000H	Specify the horizontal size of the main image in pixels.

For details about the definition of frame buffers, see **Figure 3-1. Frame Buffer Stored in a Memory Space**.

2.2.44 Input scanning mode registers

L0 input scanning mode register
(IMC_L0_SCANMODE: E126_0270H, IMCW_L0_SCANMODE: E127_0270H)
L1A input scanning mode register
(IMC_L1A_SCANMODE: E126_0370H, IMCW_L1A_SCANMODE: E127_0370H)
L1B input scanning mode register
(IMC_L1B_SCANMODE: E126_0470H, IMCW_L1B_SCANMODE: E127_0470H)
L1C input scanning mode register
(IMC_L1C_SCANMODE: E126_0570H, IMCW_L1C_SCANMODE: E127_0570H)
L2A input scanning mode register
(IMC_L2A_SCANMODE: E126_0670H) (IMC only)
L2B input scanning mode register
(IMC_L2B_SCANMODE: E126_0770H) (IMC only)
BG input scanning mode register
(IMC_BG_SCANMODE: E126_0870H) (IMC only)

These registers specify the input scanning method.
For details about interlaced and progressive scanning, see **3.3.8 Progressive and interlaced scanning**.
These are V-sync registers.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved							SCANMODE

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:1	0000_0000H	Reserved. If these bits are read, 0 is returned for each bit.
SCANMODE	R/W	0	0	Select the input scanning mode. 0: Progressive 1: Interlaced

2.2.45 Interrupt setting registers

These registers specify interrupt parameters. The IMC can issue seven types of interrupts. Control of each interrupt is assigned to each bit of the interrupt setting registers. For details, see **Table 2-1 Interrupts**.

Table 2-1. Interrupts

Interrupt Name	Interrupt Source	Bit Assignment
Frame WB end interrupt	This interrupt occurs when one frame is written back to a cache frame.	6
SAXIW error response interrupt	This interrupt occurs when a response other than OKAY sent from the AXI slave write side is received (receiving image data from the AXI master).	5
MAXIR error response interrupt	This interrupt occurs when the IMC sends a response other than OKAY when the AXI master read side reads image data from the memory.	4
MAXW error response interrupt	This interrupt occurs when a response other than OKAY from the AXI master write side is received (writing back image data).	3
Field WB end interrupt	This interrupt occurs when one field is written back to a cache frame.	2
Overrun interrupt	This interrupt occurs when a writeback to a cache frame overruns.	1
Register update end interrupt	This interrupt occurs when the update target register is updated.	0

There are two types of WB end interrupts: frame WB end interrupts and field WB end interrupts. For progressive output, these interrupts are the same. These interrupts are issued when the IMC finishes writing image data back to a cache frame. For interlaced output, a field WB end interrupt occurs when the IMC finishes writing image data back to both the odd and even fields, and a frame WB end interrupt occurs when the IMC finishes writing image data back to an even field only. Latency occurs before the data is actually written to the memory because the data is written by way of bus bridges or bus switches.

An MAXIW error response interrupt occurs if a response other than OKAY is received from the bus when the IMC writes synthesized image data back to the memory. The IMC continues processing even if there is an illegal data access. The IMC stores the AXI address at which the interrupt occurred in the error address register.

An MAXIR error response interrupt occurs if a response other than OKAY is received from the bus when the IMC reads image data from the memory. The IMC stops processing. The IMC stores the AXI address at which the interrupt occurred in the error address register.

An SAXIW error response interrupt occurs if the IMC sends a response other than OKAY when the IMC receives data input from the other master by way of the AXI slave write interface. The IMC stops processing. The IMC stores the AXI address at which the interrupt occurred in the error address register.

Details about the interrupt setting registers are described below.

(1) Interrupt status register

This is a read-only register (IMC_INTSTATUS: E126_0900H, IMCW_INTSTATUS: E127_0900H) that indicates the status of interrupt sources. The statuses of the interrupt sources enabled by using the interrupt enable set register can be read.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWBEND	SAXIWERR	MAXIRERR	MAXIWERR	WBEND	OVERRUN	REFRESH

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:7	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
PWBEND	R	6	0	Indicates the status of a frame WB end interrupt. 0: No interrupt source 1: Interrupt source occurred
SAXIWERR	R	5	0	Indicates the status of an SAXIW error response interrupt. 0: No interrupt source 1: Interrupt source occurred
MAXIRERR	R	4	0	Indicates the status of an MAXIBR error response interrupt. 0: No interrupt source 1: Interrupt source occurred
MAXIWERR	R	3	0	Indicates the status of an MAXIBW error response interrupt. 0: No interrupt source 1: Interrupt source occurred
WBEND	R	2	0	Indicates the status of a field WB end interrupt. 0: No interrupt source 1: Interrupt source occurred
OVERRUN	R	1	0	Indicates the status of an overrun interrupt. 0: No interrupt source 1: Interrupt source occurred
REFRESH	R	0	0	Indicates the status of a register update end interrupt. 0: No interrupt source 1: Interrupt source occurred

(2) Interrupt raw status register

This is a read-only register (IMC_INTRAWSTATUS: E126_0904H, IMCW_INTRAWSTATUS: E127_0904H) that indicates the statuses of interrupt sources. The bits corresponding to the interrupt sources are set regardless of the settings of the interrupt enable set register and the interrupt enable clear register.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWBEND RAW	SAXIWERR RAW	MAXIRERR RAW	MAXIWERR RAW	WBEND RAW	OVERN RAW	REFRESH RAW

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:7	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
PWBENDRAW	R	6	0	Indicates the raw status of a frame WB end interrupt. 0: No interrupt source 1: Interrupt source occurred
SAXIWERRRAW	R	5	0	Indicates the raw status of an SAXIW error response interrupt. 0: No interrupt source 1: Interrupt source occurred
MAXIRERRRAW	R	4	0	Indicates the raw status of an MAXIBR error response interrupt. 0: No interrupt source 1: Interrupt source occurred
MAXIWERRRAW	R	3	0	Indicates the raw status of an MAXIBW error response interrupt. 0: No interrupt source 1: Interrupt source occurred
WBENDRAW	R	2	0	Indicates the raw status of a field WB end interrupt. 0: No interrupt source 1: Interrupt source occurred
OVERRUNRAW	R	1	0	Indicates the raw status of an overrun interrupt. 0: No interrupt source 1: Interrupt source occurred
REFRESHRAW	R	0	0	Indicates the raw status of a register update end interrupt. 0: No interrupt source 1: Interrupt source occurred

(3) Interrupt enable set register

This register (IMC_INTENSET: E126_0908H, IMCW_INTENSET: E127_0908H) enables issuance of interrupt requests. Only data of bits to which 1 is written is updated. If the bit corresponding to an interrupt source is set to 1 in this register, the interrupt source is set, the relevant interrupt request is issued, and the corresponding bit of the interrupt status register is set to 1. If no bits are set in this register, no interrupt requests are issued even if the interrupt source is enabled, but the corresponding bit of the interrupt raw status register is set to 1.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWBENDEN	SAXIWERR EN	MAXIRERR EN	MAXIWERR EN	WBENDEN	VERRUNEN	REFREHEN

(1/2)

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:7	00_0000H	Reserved. If these bits are read, 0 is returned for each bit.
PWBENDEN	R	6	0	Indicates whether issuance of frame WB end interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	6	–	1: Enable interrupt
SAXIWERREN	R	5	0	Indicates whether issuance of SAXIW error response interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	5	–	1: Enable interrupt
MAXIRERREN	R	4	0	Indicates whether issuance of MAXIBR error response interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	4	–	1: Enable interrupt
MAXIWERREN	R	3	0	Indicates whether issuance of MAXIBW error response interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	3	–	1: Enable interrupt

(2/2)

Name	R/W	Bit No.	After Reset	Description
WBENDEN	R	2	0	Indicates whether issuance of field WB end interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	2	–	1: Enable interrupt
OVERRUNEN	R	1	0	Indicates whether issuance of overrun interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	1	–	1: Enable interrupt
REFRESHEN	R	0	0	Indicates whether issuance of register update end interrupt requests is enabled. 0: Not enabled 1: Enabled
	W	0	–	1: Enable interrupt

(4) Interrupt enable clear register

This is a write-only register (IMC_INTENCLR: E126_090CH, IMCW_INTENCLR: E127_090CH) that disables issuance of interrupt sources. Only data of bits to which 1 is written is updated. If the bit corresponding to an interrupt source in this register is set to 1, no interrupt requests are issued even if the interrupt source occurs. The status of the corresponding bit in the interrupt status register also remains unchanged. If no bits are set in this register, an interrupt request is issued and the corresponding bit of the interrupt status register is set to 1 when the interrupt source is set, and the corresponding bit of the interrupt status register is set to 1.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWBEND MASK	SAXIWERR MASK	MAXIRERR MASK	MAXIWERR MASK	WBEND MASK	OVERRUN MASK	REFRESH MASK

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:7	00_0000H	Reserved.
PWBENDMASK	W	6	0	Disable issuance of frame WB end interrupt requests. 1: Disable interrupt.
SAXIWERRMASK	W	5	0	Disable issuance of SAXIW error response interrupt requests. 1: Disable interrupt.
MAXIRERRMASK	W	4	0	Disable issuance of MAXIBR error response interrupt requests. 1: Disable interrupt.
MAXIWERRMASK	W	3	0	Disable issuance of MAXIBW error response interrupt requests. 1: Disable interrupt.
WBENDMASK	W	2	0	Disable issuance of field WB end interrupt requests. 1: Disable interrupt.
OVERRUNMASK	W	1	0	Disable issuance of overrun interrupt requests. 1: Disable interrupt.
REFRESHMASK	W	0	0	Disable issuance of register update end interrupt requests. 1: Disable interrupt.

(5) Interrupt source clear register

This is a write-only register (IMC_INTFFCLR: E126_0910H, IMCW_INTFFCLR: E127_0910H) that requests clearing of interrupt sources. Only data of bits to which 1 is written is updated. Setting the bit corresponding to an interrupt source to 1 clears the interrupt source.

If setting of an interrupt source due to the internal operation and clearing of the interrupt source by writing to this register are performed at the same time, setting takes precedence.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

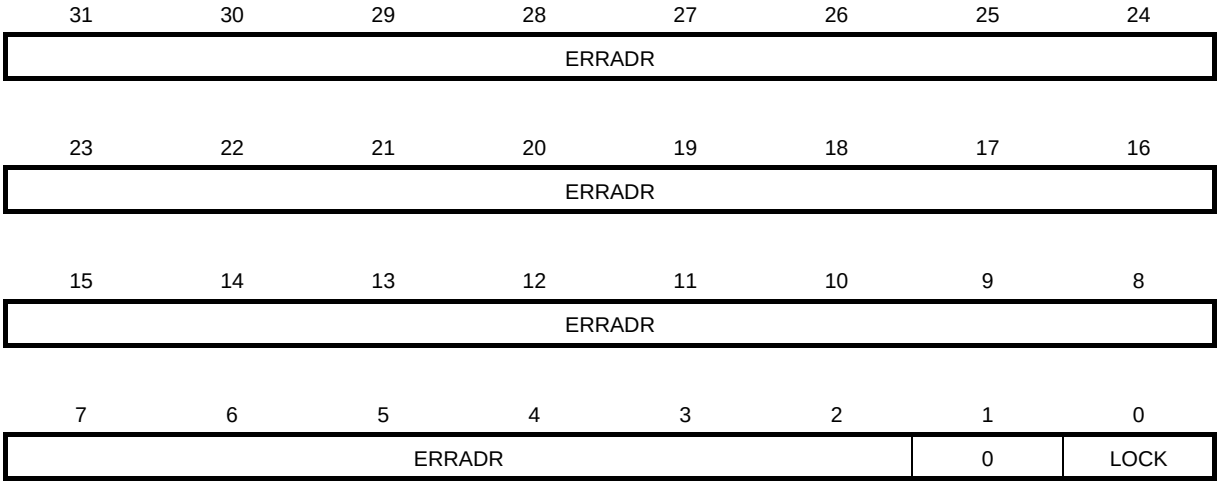
15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWBEND CLR	SAXIWERR CLR	MAXIRERR CLR	MAXIWERR CLR	WBEND CLR	OVERRUN CLR	REFRESH CLR

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:7	00_0000H	Reserved.
PWBENDCLR	W	6	0	Request clearing of a frame WB end interrupt source. 1: Clear the interrupt source.
SAXIWERRCLR	W	5	0	Request clearing of an SAXIW error response interrupt source. 1: Clear the interrupt source.
MAXIRERRCLR	W	4	0	Request clearing of an MAXIBR error response interrupt source. 1: Clear the interrupt source.
MAXIWERRCLR	W	3	0	Request clearing of an MAXIBW error response interrupt source. 1: Clear the interrupt source.
WBENDCLR	W	2	0	Request clearing of a field WB end interrupt source. 1: Clear the interrupt source.
OVERRUNCLR	W	1	0	Request clearing of an overrun interrupt source. 1: Clear the interrupt source.
REFRESHCLR	W	0	0	Request clearing of a register update end interrupt source. 1: Clear the interrupt source.

(6) Error address registers

These registers (IMC_ERRORADR_R: E126_0914H, IMC_ERRORADR_W: E126_0918H and IMC_ERRORADDR_SW: E126_091CH,
IMCW_ERRORADR_R: E127_0914H, IMCW_ERRORADR_W: E127_0918H and IMCW_ERRORADDR_SW: E127_091CH) retain the current address when an AXI bus response EXOKAY, SLVERR, or DECERR is received during data transfer by way of the AXI bus. Because the IMC has three AXI buses, the error address register is provided for each bus.
A delay occurs because of the AXI interface internal buffer. Due to this, an address later than the one at which an error occurred might be locked.



Name	R/W	Bit No.	After Reset	Description
ERRADR	R	31:2	0000_0000H	Store HADDR upon occurrence of an error response.
Reserved	R	1	0	Reserved. If this bit is read, 0 is returned.
LOCK	R/W	0	0	Check the error status. 0: Store the address when an error response occurs. 1: An error response occurred and the address was stored.

Caution If an error response occurs while the LOCK bit is set to 0, the current ADDR status is stored in the ERRADR bit and the LOCK bit is set to 1. While the LOCK bit is set to 1, ERRADR is not updated even if an error response occurs.
To acquire the error status again, clear the LOCK bit to 0.
Writing 1 to the LOCK bit does not affect the setting.

2.2.46 Composition register

This register (IMC_COMP: E126_FFFCH, IMCW_COMP: E127_FFFCH) selects the signal applied to MON_IMC[5:0] when MON_EN is at high level. This register is exclusively for debugging. Changes to this register are immediately applied.

31	30	29	28	27	26	25	24
Reserved	AXOR_MB	AXOR_FM	General			AMODE_FM	AMODE_MB

23	22	21	20	19	18	17	16
General							

15	14	13	12	11	10	9	8
General							

7	6	5	4	3	2	1	0
General							

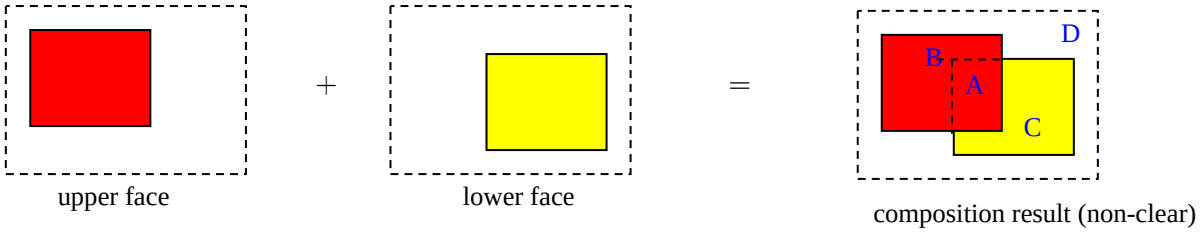
(1/2)

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31	0	Reserved. If this bit is read, 0 is returned.
AXOR_MB	R/W	30	0	When synthesizing a Middle face (L1A/L1B/L1C) and a Back face (L2A/L2B/BG), it decide about the processing when a picture in a one side in a Middle face or a Back face doesn't exist. 0 : The part where a Back face doesn't exist is filled by a black background or the fixed color and the filled color and Middle side are synthesized by alpha calculation. The part where a Middle face doesn't exist uses a Back face just as it is. 1 : A Back face isn't filled and the side of the one which exists is used just as it is without alpha calculation. ※A BG face is set by a black background or the fixed color and' the part where a Back face doesn't exist' indicates a part besides the L2A/L2B. ※' The part where a Middle face doesn't exist' indicates a part besides the L1A/L1B/L1C.

(2/2)

Name	R/W	Bit No.	After Reset	Description
AXOR_FM	R/W	29	0	When synthesizing a synthetic result (Middle face + Back face) of the Front face (L0) and the Middle face+ Back face, it decide about the processing when a picture in a face in a one side doesn't exist. 0 : The Middle face + part where a Back face doesn't exist is filled by a black background or the fixed color and the filled color and Front side are synthesized by alpha calculation. The part where a Front face doesn't exist uses a synthetic result of the Middle face + Back face just as it is. 1 : A Middle face + a Back face isn't filled and the side of the one which exists is used just as it is without alpha calculation.
General	R/W	28:26	0H	General purpose register.
AMODE_FM	R/W	25	0	The arithmetic expression when alpha blends a synthetic result of Front and Middle/Back 0 : Normal mode 1 : Premulti mode
AMODE_MB	R/W	24	0	The arithmetic expression when alpha blends a synthetic result of Middle and Back 0 : Normal mode 1 : Premulti mode
General	R/W	23:0	00_0000H	General purpose register.

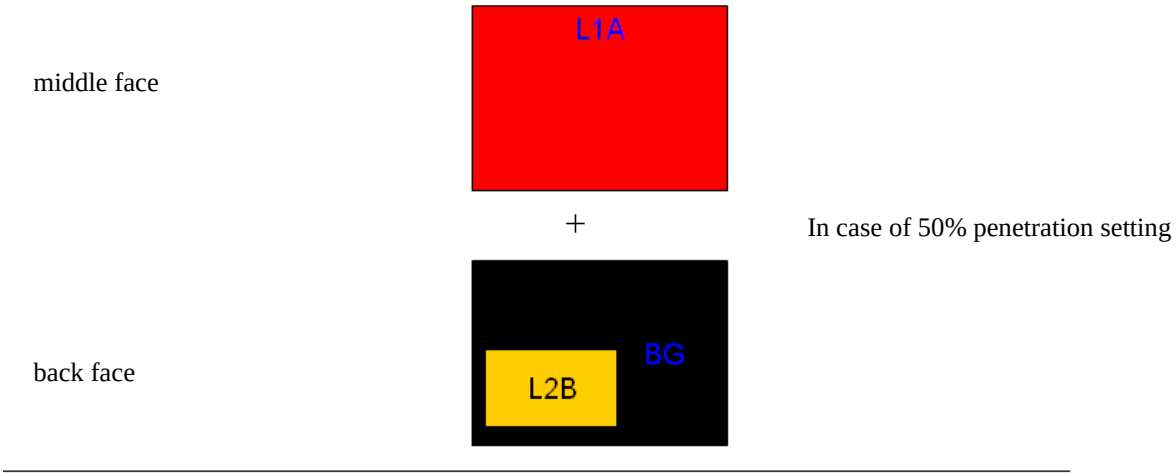
① Composition rule



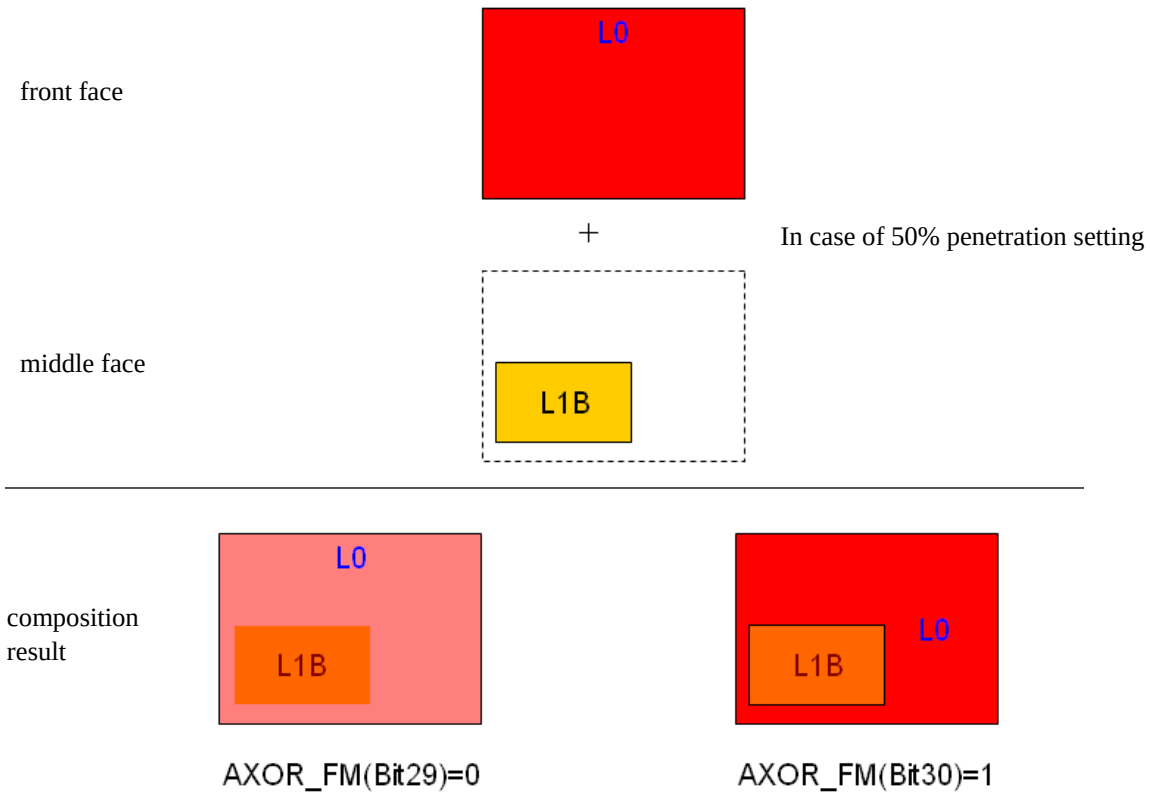
2 faces of synthesis is a basis of image composition. After synthesizing 2 faces of Middle and Back with a synthetic rule in a synthetic case, than 2 faces of the synthetic result and Front confrontation are synthesized with a synthetic rule, 3 faces are generated more. 2 synthetic rules are indicated in the following table.

	AXOR_*=0	**AXOR*=1**
Upper + lower (A)	It's synthesized with alpha blend system.	It's synthesized with alpha blend system.
Upper only (B)	A lower face is filled by the fixed color and black and it's synthesized with alpha blend system.	An upper face is reflected.
Lower only (C)	A lower face is reflected.	A lower face is reflected.
None (D)	A lower face is filled and reflected by the fixed color and black.	Black fixing

ex1)


middle face
L1A
+
In case of 50% penetration setting
back face
BG
L2B
composition result
L1A
BG
L2B
L1A
L2B
AXOR_MB(Bit30)=0
AXOR_MB(Bit30)=1

ex2)


front face
L0
+
In case of 50% penetration setting
middle face
L1B
composition result
L0
L1B
L0
L1B
AXOR_FM(Bit29)=0
AXOR_FM(Bit30)=1

② Alpha blend operation expression

There are a normal mode and a Premulti mode for an operation expression of alpha blend. The arithmetic expression of alpha value is just similar for two modes, and there is a difference only an operation expression of RGB value.

Normal mode (IMC)

PixTemp = AlphaMiddle × PixMiddle + (1 - AlphaMiddle) × PixBack
PixOut = AlphaFront × PixFront + (1 - AlphaFront) × PixTemp
AlphaOut = AlphaFront × (1 - AlphaMiddle) + AlphaMiddle

Normal mode (IMCW)

PixOut = AlphaFront × PixFront + (1 - AlphaFront) × PixMiddle
AlphaOut = AlphaFront × (1 - AlphaMiddle) + AlphaMiddle

Premulti mode (IMC)

PixTemp = PixMiddle + (1 - AlphaMiddle) × PixBack
PixOut = PixFront + (1 - AlphaFront) × PixTemp
AlphaOut = AlphaFront × (1 - AlphaMiddle) + AlphaMiddle

Premulti mode (IMCW)

PixOut = PixFront + (1 - AlphaFront) × PixMiddle
AlphaOut = AlphaFront × (1 - AlphaMiddle) + AlphaMiddle

※ AlphaMiddle and AlphaFront get the value of 0-1. Alpha value is being handled by 8bit in IMC/IMCW actually, so it's being calculated as follows.

preprocessing

```
if( m_alpha == 255 )  m_alpha = 256;
if( f_alpha == 255 )  f_alpha = 256;
```

AMODE_MB (bit24)	Arithmetic expression
1	pixtemp_r = L1_r + BG_r × (256 - m_alpha)/256;
	pixtemp_g = L1_g + BG_g × (256 - m_alpha)/256;
	pixtemp_b = L1_b + BG_b × (256 - m_alpha)/256;
0	pixtemp_r = L1_r × m_alpha/256 + BG_r × (256 - m_alpha)/256;
	pixtemp_g = L1_g × m_alpha/256 + BG_g × (256 - m_alpha)/256;
	pixtemp_b = L1_b × m_alpha/256 + BG_b × (256 - m_alpha)/256;
AMODE_FM (bit25)	**Arithmetic expression**
1	pixout_r = L0_r + pixtemp_r × (256 - f_alpha)/256;
	pixout_g = L0_g + pixtemp_g × (256 - f_alpha)/256;
	pixout_b = L0_b + pixtemp_b × (256 - f_alpha)/256;
0	pixout_r = L0_r × f_alpha/256 + pixtemp_r × (256 - f_alpha)/256;
	pixout_g = L0_g × f_alpha/256 + pixtemp_g × (256 - f_alpha)/256;
	pixout_b = L0_b × f_alpha/256 + pixtemp_b × (256 - f_alpha)/256;

③ Change rule to ARGB4444->ARGB8888

When changing ARGB4444-> to ARGB8888, a change rule of the R,G,B quantity copies 4bit of a MSB and expands to lower 4bit.

for example
right : 0xC(4)→0xCC(8)
wrong : 0xC(4)→0xC0(8)

A change rule of the A quantity expands lower 0bit.

for example
right : 0xA(4)→0xA0(8)
wrong : 0xA(4)→0xAA(8)

④ Making procedure of preMulti data (ARGB4444 function equivalence)

In case of R=0xC(4), A=0x8(4)

1) It's expanded to 4bit->8bit.
R=0xC(4)→0xCC(8)
A=0x8(4)→0x80(8)

2) Premulti calculation and lower 8bit are cut off by 8bit.
R'=R×A=0xCC(8) ×0x80(8)>>8=0x66

3) It's returned to 8bit->4bit.
As a rule : When being bigger than upper 4bit, the value of the lower 4bit of 8bit takes upper 4bit.
When being smaller than upper 4bit, the value of the lower 4bit of 8bit takes upper 4bit-1.

for example
R'=0x66(8)→0x6(4)
R'=0x65(8)→0x5(4)
R'=0x67(8)→0x6(4)

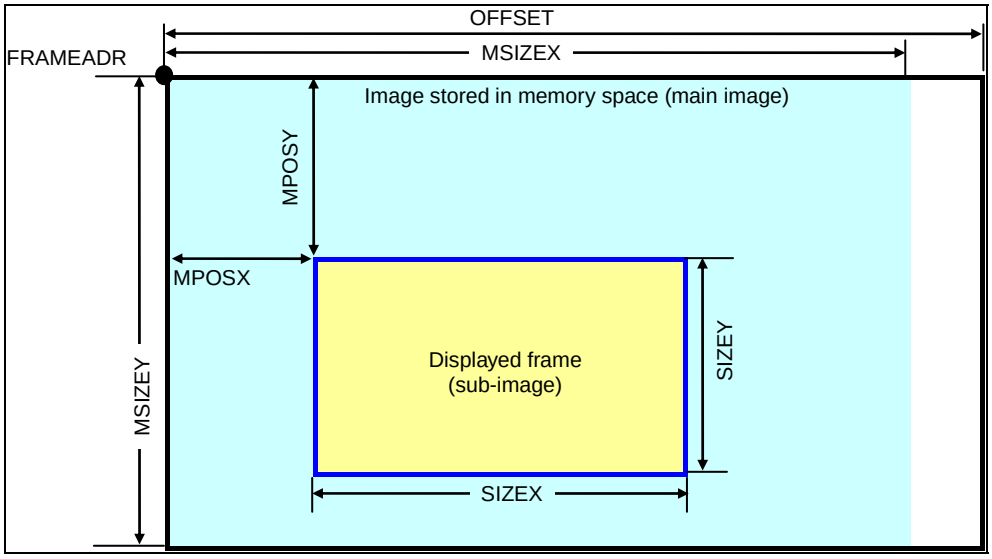
3. Description of Functions

3.1 Storing Images

3.1.1 Images stored in the memory space

An image is stored in the memory space as shown in Figure 3-1. The IMC uses this method to support wrapping around.

Figure 3-1. Frame Buffer Stored in a Memory Space

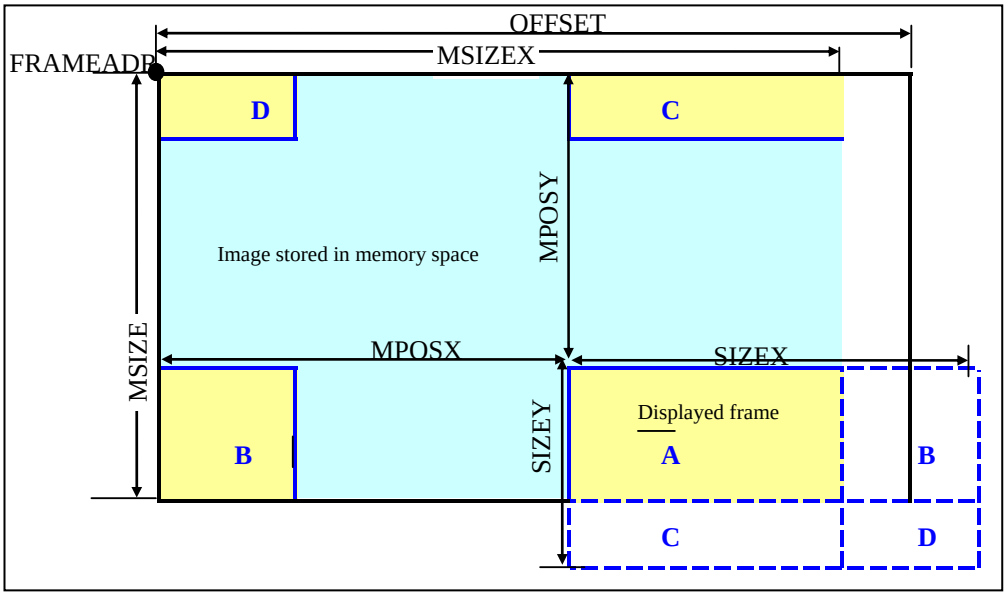


In Figure 3-1, FRAMEADR and OFFSET are defined in bytes. Other parameters are defined in pixels. In the IMC, the starting address of a sub-image is calculated in bytes according to the image format.

Frames stored in the memory space support wrapping around. *Wrapping around* refers to the process where, if a displayed frame exceeds the horizontal or vertical boundaries of the main image, the excess data wraps around to the opposite side of the same rows or columns. Figure 3-2 shows how an image is wrapped around and displayed in the memory space.

The white background portion enclosed in dotted blue lines (a logical address) is supposed to be accessed, but, because this portion exceeds the boundaries, the yellow background portion enclosed in solid blue lines (a physical address) is accessed instead. Wrapping around can be thought of as a means of converting a logical address to a physical address.

Figure 3-2. Wrapping Around in the Memory Space



If MPOSX + SIZEX > MSIZEX, wrapping around occurs in the horizontal direction. If MPOSY + SIZEY > MSIZEY, wrapping around occurs in the vertical direction.
The following conditions must be satisfied for data input and output from the IMC:

(A) MPOSX + SIZEX ≤ 2 × MSIZEX
(B) MPOSY + SIZEY ≤ 2 × MSIZEY

Be sure to follow restrictions (A) and (B) because, if either is violated, an area other than the main image is accessed.
The BG size is the same as the output-image size. If the above restrictions apply to BG frames, MPOSX, MPOSY, MSIZEX, and MSIZEY use the settings of their own registers. However, assign the SIZEX and SIZEY values of the output image to SIZEX and SIZEY.

BG acquires data from the memory territory designated in IMC_BG_FRAMEADR_x, IMC_BG_OFFSET, and IMC_WB_SIZE registers.

(A) and (B) restriction are applied about an input picture.
An output picture is applied from (A) to (D).

(C) SIZEX ≤ MSIZEX
(D) IZEY ≤ MSIZEY

The IMC supports progressive and interlaced scanning. How SIZE, MPOS, and MSIZE in the Y direction are used depends on the scanning method. For interlaced scanning, think of Figure 3-2 as showing a field, not a frame. Think of the SIZE value as half the register setting.
SIZE is used in both the memory space and synthesis space (as described in **3.1.2 Definition of a frame in the synthesis space**). For the synthesis space,
the scanning method can be specified separately between input frames or between input and output frames, but, to make the synthesis result easy to understand, specify the SIZE value of the progressive method for all frames, regardless of which scanning method they use.

The MSIZE and MPOS parameters are only for the memory space and are not related to the synthesis space. Because frames are independent, set up the memory buffer according to the scanning method for each frame. For the MSIZE and MPOS parameters of the progressive scanning method, specify the size of the frame memory buffer and the sub-image position. For the MSIZE and MPOS parameters of the interlaced scanning method, specify the size of the field memory buffer and the sub-image position. Due to these settings, for interlaced I/O, replace SIZEY with SIZEY/2 in restrictions (A) to (D) above.

3.1.2 Definition of a frame in the synthesis space

In the synthesis space, seven input images are superimposed to synthesize one image. The size of the synthesis space is the same as the output image size or BG frame size. The size and position of non-BG input images can be freely adjusted in this space. Figure 3-3 shows an image in the frame synthesis space.

Figure 3-3. Display in the Frame Synthesis Space

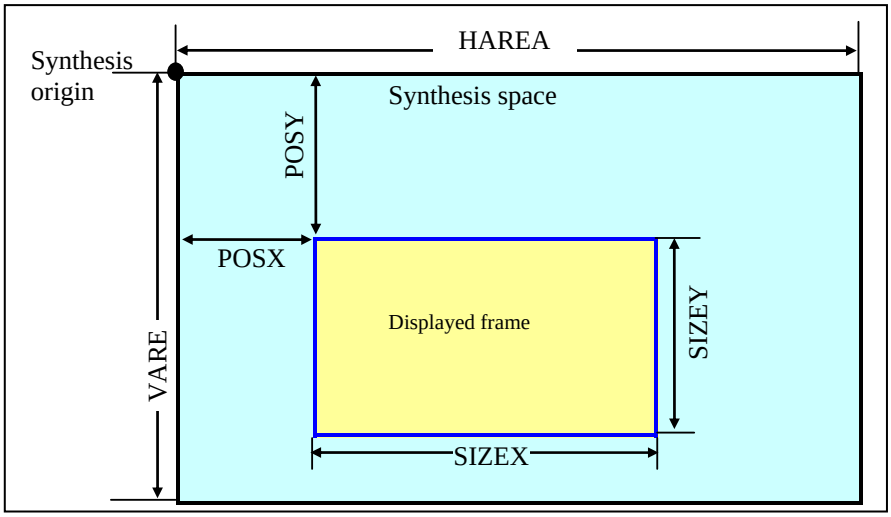


Figure 3-3 shows how an image is displayed in the frame synthesis space. For interlaced output, the odd and even fields are merged into one image and can be thought of as shown in Figure 3-3.
Because wrapping around is not supported for the synthesis space, follow the following restrictions for each frame:

(A) POSX + SIZEX ≤ HAREA
(B) POSY + SIZEY ≤ VAREA[Note]

Note Use the VAREA value specified for progressive output scanning.
In the LCD-synchronous mode, VAREA is controlled by using the LCD controller.
In the immediate startup mode, use the VAREA value specified using the IMC_WB_SIZE register.

3.2 Image Formats

3.2.1 Format selectable for each frame

In addition to a black background and fixed colors, the IMC supports 12 image formats. The formats supported by the IMC for each frame (L0, L1A, L1B, L1C, L2A, L2B, BG, and synthesis result) differ. Table 3-1 lists the frames and selectable format.

Table 3-1. Format Selectable for Each Frame

ID	Format	L0	L1A, L1B, L1C	L2A and L2B	BG	Synthesis Result (WB)[Note 3]
0	RGB888	○	○	○	○	○
1	RGB666	○	○	○	○	○
2	RGB565	○	○	○	○	○
4	ARGB8888	○	○			○
5	ARGB4444	○	○			○
6	ARGB1555	○	○			
8	YUV422I			○		
9	YUV422S			○		
10	YUV422P			○		
12	YUV444			○		
13	YUV420S			○		
14	YUV420P			○		
3	Black background				○	
4[Note 1]	Fixed colors[Note 2]				○	

Notes 1. Fixed colors and ARGB8888 share the same ID. ARGB8888 is used for layer 0 or layer 1*x* and a fixed color is used for the BG.

2. When STARTMODE = 1, the fixed color specified by the IMC_BACKCOLOR register is used. When STARTMODE = 0, the fixed color determined by the signal of the BGCOLOR[23:0] pin, which is connected to the LCD controller, is used.

3. In the LCD-synchronous mode, specify the synthesis result format by using the LCD_IFORMAT register in the LCD controller. In the immediate startup mode, it is determined according to the setting of the IMC_WB_FORMAT register.

Any image format can be selected for layer 2. The register that defines the start addresses of frame buffers varies depending on the specified format. Table 3-2 lists the definitions of start addresses in each format.

Table 3-2. Formats and Start Address Setting Registers

Selected Format	Number of Planes	Setting Method	
		Stored Data	Register That Specify Start Address
RGB565, RGB666, RGB888 ARGB8888, ARGB4444, ARGB1555	1	RGB plane	Address register for Y
YUV422 Pixel Interleave YUV444 Pixel Interleave	1	YUV plane	Address register for Y
YUV422 semi-planar YUV420 semi-planar	2	Y plane	Address register for Y
		UV plane	Address register for U
YUV422 planar YUV420 planar	3	Y plane	Address register for Y
		U plane	Address register for U
		V plane	Address register for V

3.2.2 Restrictions on image setting

Register settings must be restricted according to the I/O image format. For each register, specify an integer that is a multiple of the values shown in Table 3-3.

Table 3-3. Minimum Unit That Can Be Specified for Image Size

ID	Format	FRAMEADR (Bytes)	MSIZE		MPOS		OFFSET (Bytes)	SIZE		POS	
			Y (Lines)	X (Pixels)	Y (Lines)	X (Pixels)		Y (Lines)	X (Pixels)	Y (Lines)	X (Pixels)
0	RGB888	1	1	1	1	1	1	1 Note 1	1	1	1
1	RGB666	1	1	1 Note 2	1	1 Note 2	1	1 Note 1	1 Note 2	1	1
2	RGB565	1	1	1	1	1	1	1 Note 1	1	1	1
4	ARGB8888	1	1	1	1	1	1	1 Note 1	1	1	1
5	ARGB4444	1	1	1	1	1	1	1 Note 1	1	1	1
6	ARGB1555	1	1	1	1	1	1	1 Note 1	1	1	1
8	YUV422I	1	1	2	1	1	1	1 Note 1	1	1	1
9	YUV422S	1	1	4	1	1	1	1 Note 1	1	1	1
10	YUV422P	1	1	8	1	1	2	1 Note 1	1	1	1
12	YUV444	1	1	1	1	1	1	1 Note 1	1	1	1
13	YUV420S	1	1	4	1	1	1	1 Note 1	1	1	1
14	YUV420P	1	1	8	1	1	2	1 Note 1	1	1	1

Notes **1.** For the interlaced scanning mode, set the vertical size (SIZEY) of the corresponding frame to an even number.

2. If outputting in the RGB666 format, MSIZEX, MPOSX, and SIZEX might be restricted to integer multiples of 4. For details, see **2.2.14 WB image size register** and **2.2.16 WB memory frame start position register.**

3.2.3 Image format

A stock image on the memory of each image format IMC macro supports with default is indicated in figure 3-4 to 3-7..

The address line expresses the value out of which start address in a frame buffer was reduced in the byte unit.

Figure 3-4. Data Storage for RGB Format

Plane	Address	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
RGB	00	B1	R0	G0	B0
	04	G2	B2	R1	G1
	08	R3	G3	B3	R2
Data storage image for RGB8888					
RGB	00	R1	G1	B1	R0	G0	B0
	04	G3	B3	R2	G2	B2	R1
	08	B5	R4	G4	B4	R3	G3
	0C	B7	R6	G6	B6	R5	G5
	10	R8	G8	B8	R7	G7	B7
	14	G10	B10	R9	G9	B9	R8
	18	G12	B12	R11	G11	B11	R10
	1C	B14	R13	G13	B13	R12	G12
	20	R15	G15	B15	G14	R14	B14
Data storage image for RGB666							
RGB	00	R1	G1	B1	R0	G0	B0
Data storage image for RGB565							

Figure 3-5. Data Storage for RGB Format

プレーン	アドレス	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
ARGB	00	A	R	G	B
Data storage image for ARGB8888					
ARGB	00	A1	R1	G1	B1	A0	R0	G0	B0
Data storage image for ARGB4444									
ARGB	00	A1	R1	G1	B1	A0	R0	G0	B0
Data storage image for ARGB1555									

Figure 3-6. Data Storage for YUV422 and YUV444 Format

Plane	Address	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
YUV	00	V0/1	U0/1	Y1	Y0
Data storage image for YUV422 Interleave					
YUV	00	Y1	V0	U0	Y0
	04	U2	Y2	V1	U1
	08	V3	U3	Y3	V2
Data storage image for YUV444					

Figure 3-7. Data Storage for YUV422 Semi Planar and YUV422 Planar Format

Plane	Address	31 30 29 28 27 26 25 24	23 22 21 20 19 18 17 16	15 14 13 12 11 10 9 8	7 6 5 4 3 2 1 0
Y	00	Y3	Y2	Y1	Y0
UV	00	V2/3	U2/3	V0/1	U0/1

Data storage image for YUV422 Semi-Planar

Y	00	Y3	Y2	Y1	Y0
	04	Y7	Y6	Y5	Y4
U	00	U6/7	U4/5	U2/3	U0/1
V	00	V6/7	V4/5	V2/3	V0/1

Data storage image for YUV422 Planar

The data storage image of the YUV420 Semi-Planer /YUV420 Planer format is same as YUV422 Semi-Planer and YUV422 Planer.

3.2.4 Format conversion rule

The IMC macro internally converts the image data for each frame to the RGB888 or ARGB8888 format immediately before synthesis, and then, after the synthesis processing is performed, the data is converted back to the output format (RGB565, RGB666, or RGB888), and this data is supplied to the LCD controller and written back to a cache frame. Table 3-4 lists the format conversion rules.

Table 3-4. Format Conversion Rules

Conversion from each format to RGB888 or ARGB8888	
RGB565, RGB666	Adds the MSB to the LSB of R, G, and B.
ARGB1555, ARGB4444	In case of ARGB1555, the set value of IMC_ALPHASEL0 is used at A=0 and the set value of IMC_ALPHASEL1 is used at A=1. In case of ARGB4444, 0 is copied in LSB 4bit of A ingredient and it's expanded in 8bit.
YUV422 Pixel Interleave YUV422 semi-planar YUV422 planar YUV444 YUV420 semi-planar YUV420 planar	The following three steps are performed sequentially: • Gain/offset adjustment (8-bit accuracy before and after adjustment) • YUV-to-RGB conversion (8-bit accuracy before and after adjustment)
Conversion from RGB888 to RGB565 or RGB666 or ARGB4444	
RGB666	• If dithering is enabled, the dithering method is used to convert data to the RGB666 format. • If dithering is disabled, the lower bits are discarded.
RGB565	• If dithering is enabled, the dithering method is used to convert data to the RGB666 format, and then the LSB is discarded. • If dithering is disabled, the lower bits are discarded.
ARGB4444	The lower bits are discarded. Dithering non-correspondence.

3.2.5 Gain and offset adjustment

The gain and offset of YUV format data can be adjusted in the range from 0 to x255/x128.
Each item can be specified by using Y gain offset register (IMC_YGAINOFFSET), U gain offset register (IMC_UGAINOFFSET) and V gain offset register (IMC_VGAINOFFSET). For details about the specified values and adjusted items, see the descriptions of each register.

Table 3-5. Gain/Offset Adjustment Register

Setup Register	Symbol	See:
Y gain offset register	IMC_YGAINOFFSET	**Y gain offset register**
U gain offset register	IMC_UGAINOFFSET	**U (V) gain offset registers**
V gain offset register	IMC_VGAINOFFSET	

3.2.6 YUV-to-RGB conversion

YUV format data can be converted to the RGB format by using arithmetic calculation.
The IMC can select a calculation coefficient from ITU-R BT.601-compliant, ITU-R BT.709-compliant, custom coefficient (with or without Y value subtracted by 16) for YUV-to-RGB format conversion, by using the IMC_YUV2RGB register.
When using a custom coefficient for conversion, separately specify the coefficient in the IMC_COEF_[RGB][0:3] registers. For details about conversion formulas, see **2.2.25 Custom coefficient registers.**

3.2.7 Dithering

Dithering is a method of pseudo tone processing. With dithering, the rounding error in lower bits, which occurs when dropping the data tone (masking colors), is dispersed to the surrounding pixels. Dithering allows the expression of natural colors with low tone.
If converting a YUV format to RGB666, the IMC can perform dithering after conversion with a 3 × 4 matrix calculation. The IMC employs systematic 2 × 2 dithering.

Table 3-6. Dithering Table Values

LSB on Vertical Coordinate of Synthesized Image[Note]	LSB on Horizontal Coordinate of Synthesized Image	
	0	1
0	0	2
1	3	1

Note For interlaced output, the vertical coordinates for each field are used. Therefore, if dithering is enabled and the odd and even fields are combined into one image, if the same settings are used to output an image in the progressive mode, the images might not match.

Dithering is performed for each pixel of synthesized images to be output from the IMC.
First, select a table value listed in **Table 3-6 Dithering Table Values**, based on the position of the current pixel to be processed (coordinate based on the synthesized image). This table value is used to switch how the lower 2 bits are handled during 8-to-6-bit conversion. During ordinary conversion with rounding off, the value is rounded up if bit 1 of the original 8-bit data is 1, or rounded off if it is 0 (02H is added and then the lower 2 bits

are dropped). With dithering by the IMC, the table value is added and then the lower 2 bits are dropped. As a result, data is rounded with the probability shown in **Table 3-7 Probability of Dithering Error Dispersion**.

Table 3-7. Probability of Dithering Error Dispersion

Lower 3 Bits of Original Data	Probability for Round up	Probability for Round off
0	0/4	4/4
1	1/4	3/4
2	2/4	2/4
3	3/4	1/4

Remark The probability in the above table means the probability with which rounding up or off is performed for vertical and horizontal 2 pixels, four pixels in total.

The detailed operation performed during dithering conversion is shown below.

Figure 3-8. Color Masking by Using Dithering

■ Converting 8-bit Red data to 6-bit data

Original data before dithering performed (8 bits of red component)

	0	1	2	3	4	5
0	13H	14H	14H	21H	28H	3AH
1	32H	2CH	33H	40H	51H	55H
2	F4H	F0H	F1H	DDH	FAH	FAH
3	E0H	E5H	CCH	C2H	FAH	FAH

Dithering table values for each coordinate

	0	1	2	3	4	5
0	0	2	0	2	0	2
1	3	1	3	1	3	1
2	0	2	0	2	0	2
3	3	1	3	1	3	1

■ Dithering procedures

- First, expand the dithering table to the size of the layer to be processed (right table).
- Calculate the value of upper-left pixel (V, H) = (0, 0).
- Add table value 0 to 13H and drop the lower 2 bits. As a result, 4H is obtained.
- Next, calculate the second pixel (0, 1).
- Add table value 2 to 14H (original data) and drop the lower 2 bits. As a result, 5H is obtained.
- Pixels are processed as shown above and corrected to be 6-bit tones.

■ Comparison between results of converted data with dithering performed and converted data with rounding on/off

Converted data with dithering performed (6 bits of red component)

	0	1	2	3	4	5
0	04H	05H	05H	08H	0AH	0FH
1	0DH	0BH	0DH	10H	15H	15H
2	3DH	3CH	3CH	37H	3EH	3FH
3	38H	39H	33H	30H	3FH	3EH

Converted data with rounding on/off

	0	1	2	3	4	5
0	05H	05H	05H	08H	0AH	0FH
1	0DH	0BH	0DH	10H	14H	15H
2	3DH	3CH	3CH	37H	3FH	3FH
3	38H	39H	33H	31H	3FH	3FH

* Shaded pixels do not match the corresponding pixels of data with dithering performed

As shown in the right-bottom area, for example, where an intermediate color is filled (lower 2 bits of the original 8-bit data are not 0), differences between the intermediate tone resulting from dithering and converted data with rounding on/off appear significantly.

3.3 Image Synthesis

3.3.1 Definitions of layers and processes

In the IMC circuit, internal processing is performed by rearranging seven frames, which are organized into four layers, into three processes.

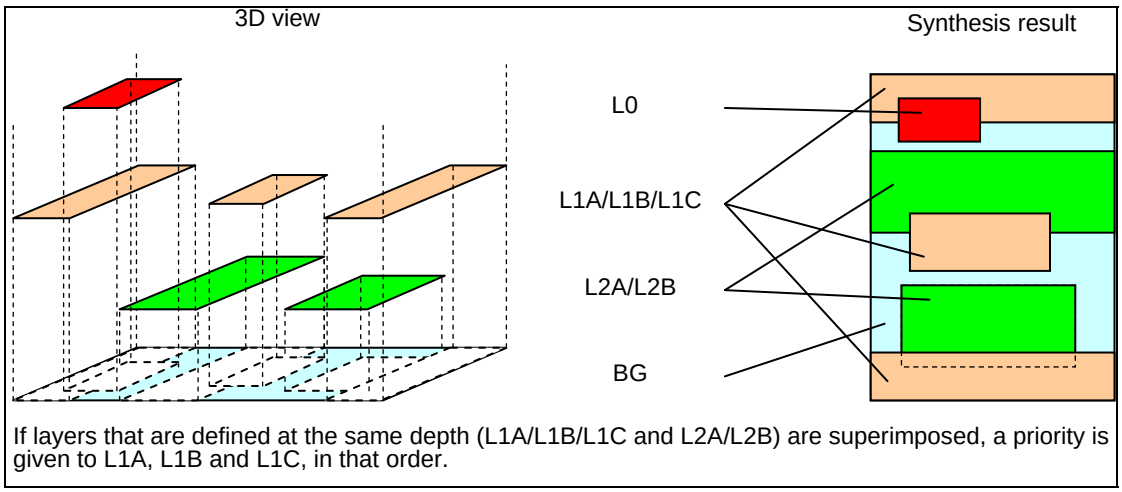
Front process: L0
Middle process: L1A, L1B, L1C (order of priority in the process: L1A > L1B > L1C)
Back process (IMC only) : L2A, L2B, BG (order of priority in the process: L2A > L2B > BG)

In each process, if a superimposed display area is specified, the displayed frame is selected according to a fixed priority order. In other words, if L1A and L1B are superimposed, L1A is selected first for the middle process. At this time, if semi-transparent processing is performed for L1A by using a transparency color and alpha blending, L1B is not visible behind L1A, and the back-process frame is displayed.

3.3.2 Synthesis rule

The IMC can handle four layers in the depth direction (front and back): layers 0, 1, 2, and background. Layer 1 has three frames at the same depth, and layer 2 has two frames at the same depth. The positional relationship between layers is fixed. The supported formats and settable parameters vary in each layer.
Figure 3-9 shows the concept of layers, and Table 3-8 lists the parameter features.

Figure 3-9. Layer Definition



If layers that are defined at the same depth (L1A/L1B/L1C and L2A/L2B) are set to different display positions, a selection priority is given to layer 1A, layer 1B and layer 1C, in that order. Between layers 2A and 2B, layer 2A is given a priority.
As shown in Figure 3-10, for example, if layers 1B and 1C are superimposed and the transparent background is specified for layer 1B, layer 1B is brought to the front at the position where two layers are superimposed. Layer 1C hidden behind the transparent portions of layer 1B is not displayed but the background is displayed.
Layer synthesis can superimpose up to three processes (layer 0 + layer 1x + layer 2x or layer BG).

Figure 3-10. Duplicated Setting of Layers at the Same Depth

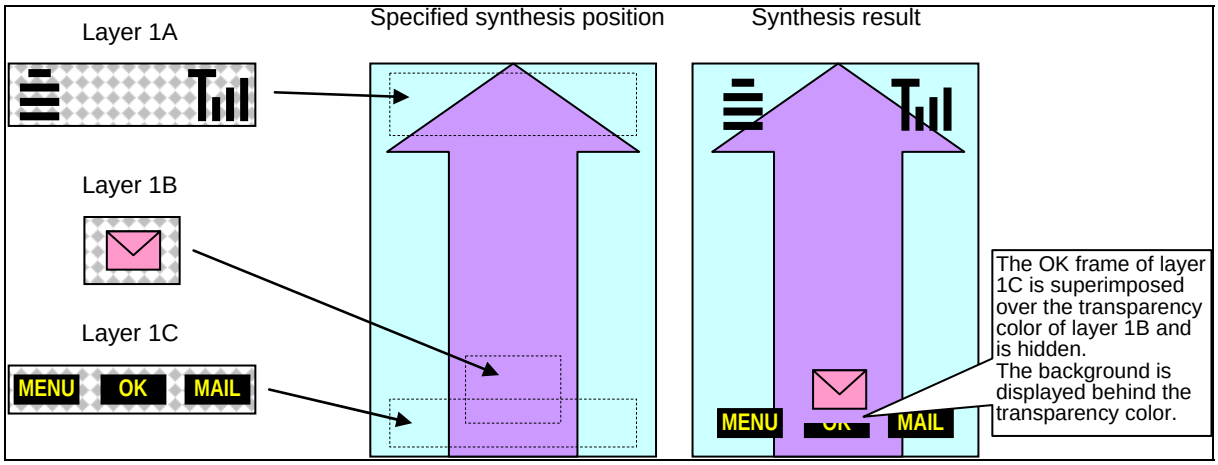


Table 3-8. Setting Parameters

In/Out		Input							Output	
Frame ID→	Offset	L0	L1A	L1B	L1C	L2A	L2B	BG	Offset	WB
Start address→	↓	0200H	0300H	0400H	0500H	0600H	0700H	0800H	↓	0040H
CONTROL	+00	○/○	○/○	○/○	○/○	○/×	○/×	×/×		×/×
FORMAT	+04	○/○	○/○	○/○	○/○	○/×	○/×	○/×	+08	○/○
BUFSEL	+08	○/○	○/○	○/○	○/○	○/×	○/×	○/×	+14	○/○
BYTELANE	+0C	○/○	○/○	○/○	○/○	○/×	○/×	○/×	+24	○/○
KEYENABLE	+10	○/○	○/○	○/○	○/○	×/×	×/×	×/×	–	×/×
KEYCOLOR	+14	○/○	○/○	○/○	○/○	×/×	×/×	×/×	–	×/×
ALPHA	+18	○/○	○/○	○/○	○/○	×/×	×/×	×/×	–	×/×
RESIZE	+20	○/○	○/○	○/○	○/○	○/×	○/×	○/×	–	×/×
MIRROR	+24	○/○	○/○	○/○	○/○	○/×	○/×	○/×	–	×/×
OFFSET	+30	○/○	○/○	○/○	○/○	○/×	○/×	○/×	+04	○/○
FRAMEADR(YP)	+34	○/○	○/○	○/○	○/○	○/×	○/×	○/×	+00	○/○
FRAMEADR(UP)	+38	×/×	×/×	×/×	×/×	○/×	○/×	×/×	–	×/×
FRAMEADR(VP)	+3C	×/×	×/×	×/×	×/×	○/×	○/×	×/×	–	×/×
FRAMEADR(YQ)	+40	○/○	○/○	○/○	○/○	○/×	○/×	○/○	+10	○/○
FRAMEADR(UQ)	+44	×/×	×/×	×/×	×/×	○/×	○/×	×/×	–	×/×
FRAMEADR(VQ)	+48	×/×	×/×	×/×	×/×	○/×	○/×	×/×	–	×/×
POSITION	+50	○/○	○/○	○/○	○/○	○/×	○/×	×/×	–	×/×
SIZE	+54	○/○	○/○	○/○	○/○	○/×	○/×	×/×	+0C	○/○
MPOSITION	+60	○/○	○/○	○/○	○/○	○/×	○/×	○/○	+18	○/○
MSIZE	+64	○/○	○/○	○/○	○/○	○/×	○/×	○/○	+1C	○/○
SCANMODE	+70	○/○	○/○	○/○	○/○	○/×	○/×	○/○	+30	○/○

○ Yes ×No IMC/IMCW format

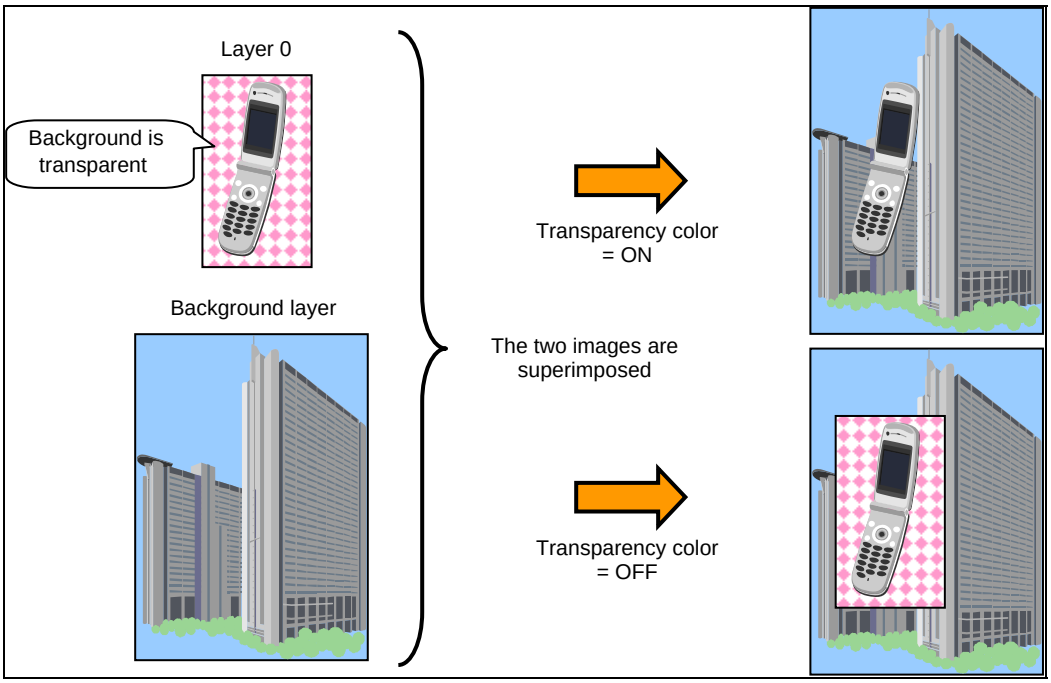
3.3.3 Transparency colors (key color)

Transparency colors can be specified separately for layers 0 and 1. When a value of pixel data expressed in the RGB format is defined as a transparency color and pixel data values of the target layer match the specified transparency color value, transparency processing is performed and the background layer can be seen.

For details about the register settings, see descriptions on each register.

Figure 3-11 shows how images are synthesized using transparency color.

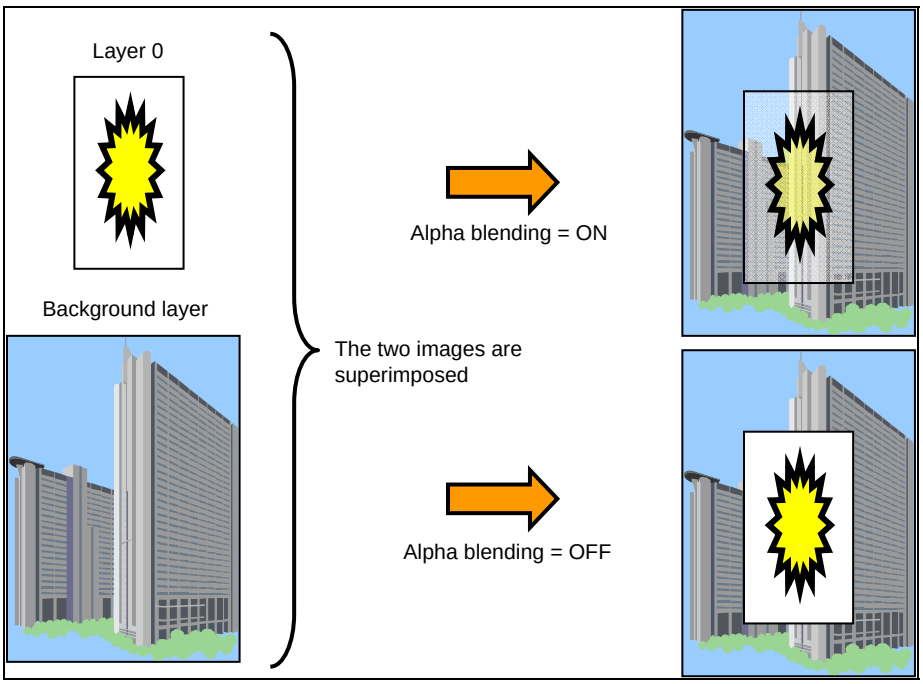
Figure 3-11. Transparency Color



3.3.4 Alpha blending

Alpha blending can be specified separately for the A, B and C planes of layers 0 and 1. Alpha blending is a method of processing to opaque images, in 256 steps. When alpha blending is enabled for a target frame, opaque processing is performed the frames and the background layer can be seen.
For details about the register settings, see descriptions on each register.
Figure 3-12 shows how images are synthesized by using alpha blending.

Figure 3-12. Alpha Blending



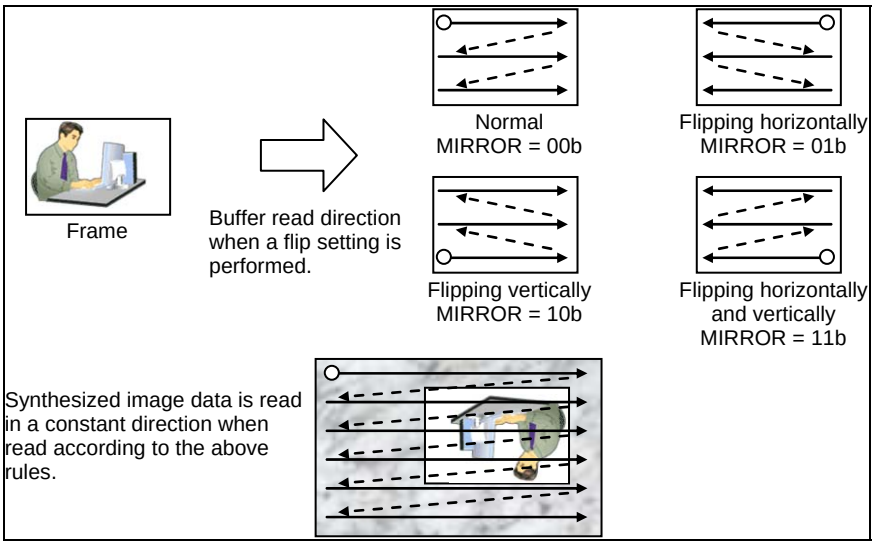
Calculation of desired values when layers 0 and 1x are superimposed and alpha blending is enabled for both layers

The IMC first processes the background data. When synthesizing front, middle, and back layers, the middle and back layers are synthesized first. If alpha blending settings have been specified for the middle layer, blending is performed according to the specified values. Synthesis with the front layer is then performed for the resulting 8-bit value.

3.3.5 Image flip

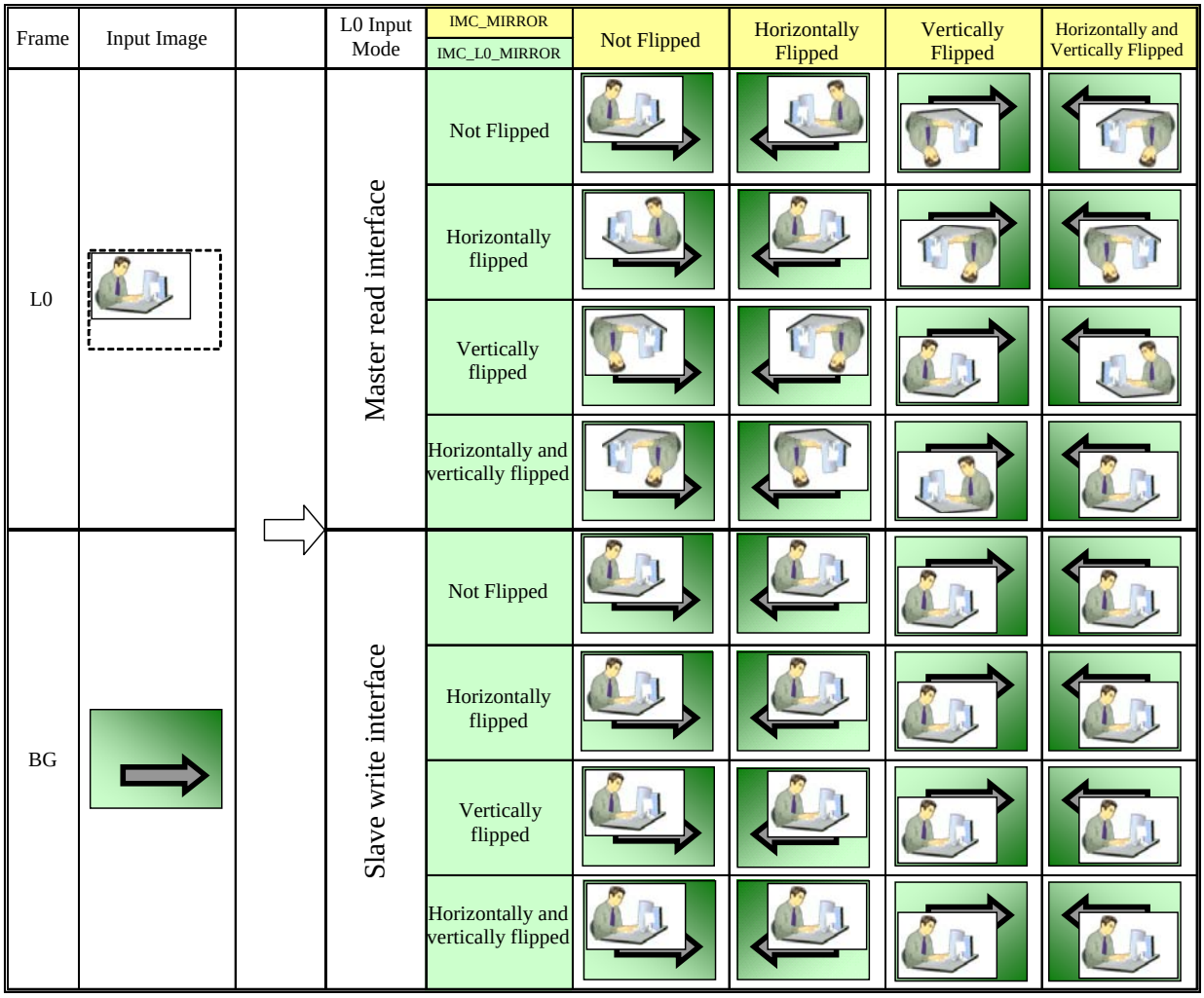
When inputting images by way of the AXI master read interface, flipping images horizontally or vertically can be specified separately each frame. When horizontal or vertical flipping is specified for each frame, the display position does not change but only image data is flipped.
Aside from this function, a similar flip setting can be performed for images output (synthesized images). If each frame image is flipped, synthesis is performed, and then the synthesis result is flipped, the effect of flipping is eliminated. Figure 3-13 shows this option.

Figure 3-13. Operation When Image Flipping Is Enabled



If inputting data from the AXI slave write interface, because data can only be received in the raster order, the order in which data is received does not change according to the MIRROR setting as shown in Figure 3-13, and operation is only performed as though MIRROR = 00b is specified, even if the image flipping setting for each frame is set to 0 to 3. However, in the synthesis result window, it is possible to flip the display position by using the IMC_MIRROR setting.

Figure 3-14. The Effect of Mirroring



Remark Figure 3-14 uses L0 as an example. This figure also applies to other frames.

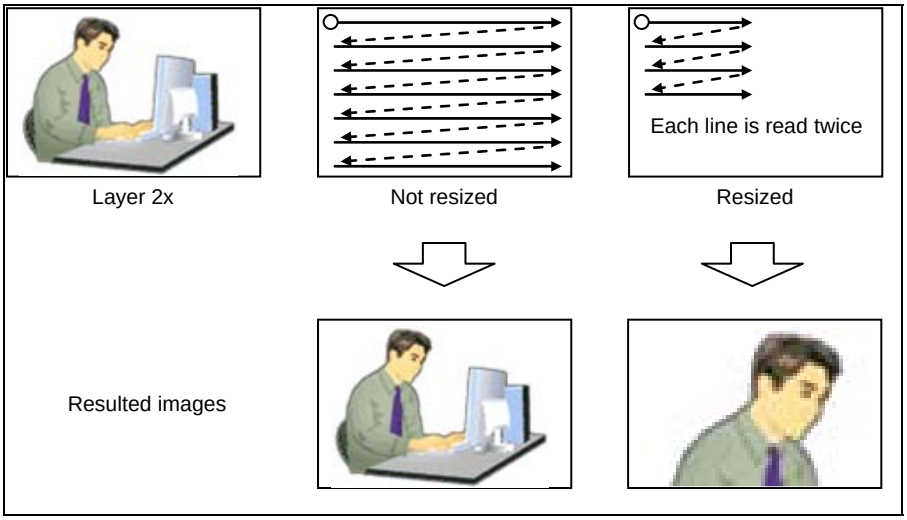
The MIRROR (IMC_L*_MIRROR/IMC_BG_MIRROR) setting of a frame subject to the AXI slave write interface is forcibly handled in the circuit as being the same value as that of the output image horizontal/vertical flip setting register (IMC_MIRROR). The effect of the I/O MIRROR setting is eliminated, and data is always displayed as though it has not been flipped. However, the start position for displaying a frame is flipped using the IMC_MIRROR setting, and the effect of the frame MIRROR setting is not eliminated, so only the display position is flipped. Check the effect of flipping in Figure 3-14.

If the opposition master of the AXI slave write interface flips the output according to the IMC_L0_MIRROR setting and outputs to the IMC, the same effect is possible as when inputting from the master read interface.

3.3.6 Simple resizing

Simple doubling is provided for each frame to reduce the image data size in frame buffers and the amount of data read from buffers. Data read by the IMC from a frame buffer is simply doubled horizontally. Data in the vertical direction is read from the same line, in 2-line units. As a result, the data buffer can be 1/4 and the bus bandwidth required for reading data can be halved.

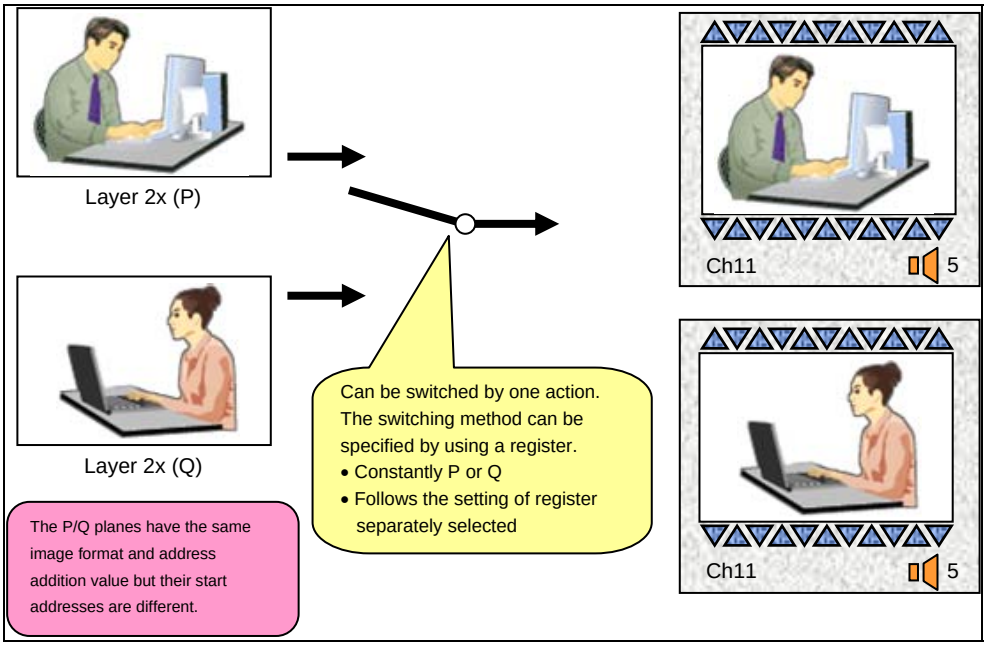
Figure 3-15. Operation When Simple Resizing Is Enabled



3.3.7 Double buffer

Each frame can have two sets of definitions, P and Q, for individual frame buffer.
Which of P and Q is selected can be specified by using double buffer control register for each frame. The fixed P/Q, CPU control/image flip control can be specified. If CPU control/image flip is specified, all frames can be changed at the same time, by separately setting the CPU double buffer control register (IMC_CPUBUFSEL).
For interlaced input, the two buffers support the odd and even fields. As for the specified double buffer rules, for progressive input, the specified buffer is used as the odd field buffer, and the unspecified buffer is used as the even field buffer.

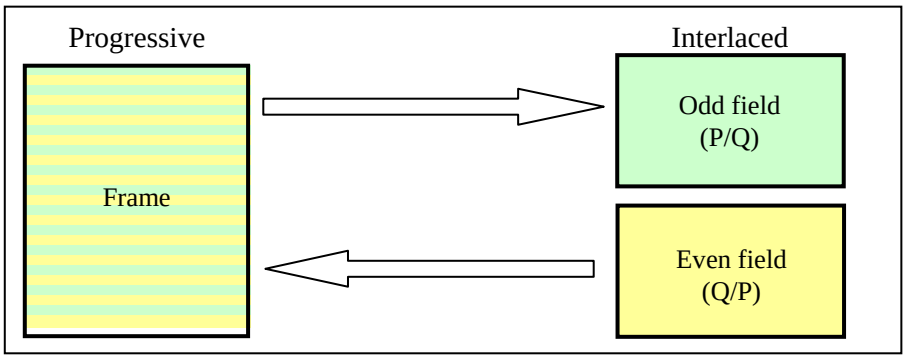
Figure 3-16. Operation When Double Buffer Function Is Enabled



3.3.8 Progressive and interlaced scanning

IMC input and output support the progressive and interlaced scanning modes. For interlaced scanning, the image is divided into an odd field and even field, and these fields are shared by the two buffers for double buffering. If interlaced scanning is enabled, double buffering is disabled. In this case, the correspondence between the odd/even field and the P/Q buffer can be specified or switched using the double buffer control register and CPU double buffer control register.
Figure 3-17 shows conversion from the progressive mode to the interlaced mode.

Figure 3-17. Conversion Between the Interlaced and Progressive Modes



When speaking, scanned lines are counted starting at 1. However, the HDL performs counting starting at 0. Therefore, there is a difference of one line between scanning and the hardware. When counting starting at 1, the odd field is the image made up of lines 1, 3, 5, and the following, and the even field is the image made up of lines 2, 4, 6, and the following. When counting starting at 0, the odd field is the image made up of lines 0, 2, 4, and the following, and the even field is the image made up of lines 1, 3, 5, and the following.

3.4 Inputting Images by Using the Slave Write Interface

3.4.1 Restrictions

For IMC input, an AXI-standard slave write interface is used and data is directly written from an external master (SIZ, M2M, or CAM) without going through memory to make the following possible:

- No intermediate buffer is required (which reduces the amount of used DRAM).
- The amount of DRAM R/W operations is reduced, which increases speed and reduces voltage consumption.
- The used DRAM bandwidth is reduced (which increases the surplus bandwidth for other masters).

However, there are restrictions on masters when inputting images from the slave interface:

(1) Input output images in the raster order.

(2) Set the number of frames that can be received when simultaneously inputting images from the slave interface to one or less.

(3) Only a format that has one plane is not supported.

(4) For the master, transfer a sub-image, not a main image.

(5) Specify settings such that the input image address is not outside the range from C000_0000H to C7FF_FFFFH for the IMC or C800_0000H to CFFF_FFFFH for the IMCW. Frame images for which FRAMEADR is set to a value in the above range are received from the slave write interface. Images for which a value outside this range is specified are read by way of the master read interface.

(6) For an input image received from the slave write interface, the mirror settings (IMC_Lx_MIRROR and IMC_BG_MIRROR) for the corresponding frame are forcibly set to the value of the output image horizontal/vertical flip setting register (IMC_MIRROR). The resize settings (IMC_Lx_RESIZE and IMC_BG_RESIZE) are forcibly handled as 0. To use the MIRROR and RESIZE functions, do so on the master side. In addition, follow the following restrictions so no wrap around occurs:

MSIZEX ≥ MPOSX + SIZEX

MSIZEY ≥ MPOXY + SIZEY

If IMC_MIRROR is not 0, only the display position for an input image is flipped, not the image data.

(7) Match the scanning mode for a frame subject to the slave write interface with the output scanning mode. For the interlaced scanning mode, set the vertical display position to an even number.

(8) The master must output images according to the settings for a frame subject to the slave write interface. Specifically, the following attributes must match:

- Format
- Frame buffer start address

  Match the start address of an output frame buffer for the master with the sub-image start address of a frame subject to the slave write interface. Calculate the sub-image start address after you understand Figure 3-1. Frame Buffer Stored in a Memory Space.

  If MPOSITION is not 0, the sub-image start address and value specified for the register differ.
- Address addition value
- Display size

  For the YUV422 interlaced format, note that, because pairs of pixels share U/V signals, the pair data must be transferred even if only one pixel in a pair belongs to a sub-image.

(9) For the master, data must not be enabled before the address.

3.4.2 Usage

The following describes how to use the slave write interface.

(1) Set up registers related to the slave write interface start frame of the IMC.

(2) Set up the master. Output images according to frames subject to the IMC slave write interface, as described in step (8) above.

(3) Start the IMC.

(4) Start the master.

3.5 VGA Standby

3.5.1 Usage

VGA standby is used to display the LCD while power to the L1 domain (where the IMC is mapped) in the EM/EV application block is turned off. This enables low-power scheduling by using the SDRAM interface cache in the MEMC macro, without IMC accessing by way of the AXI bus.

The features of the VGA standby function are as follows:

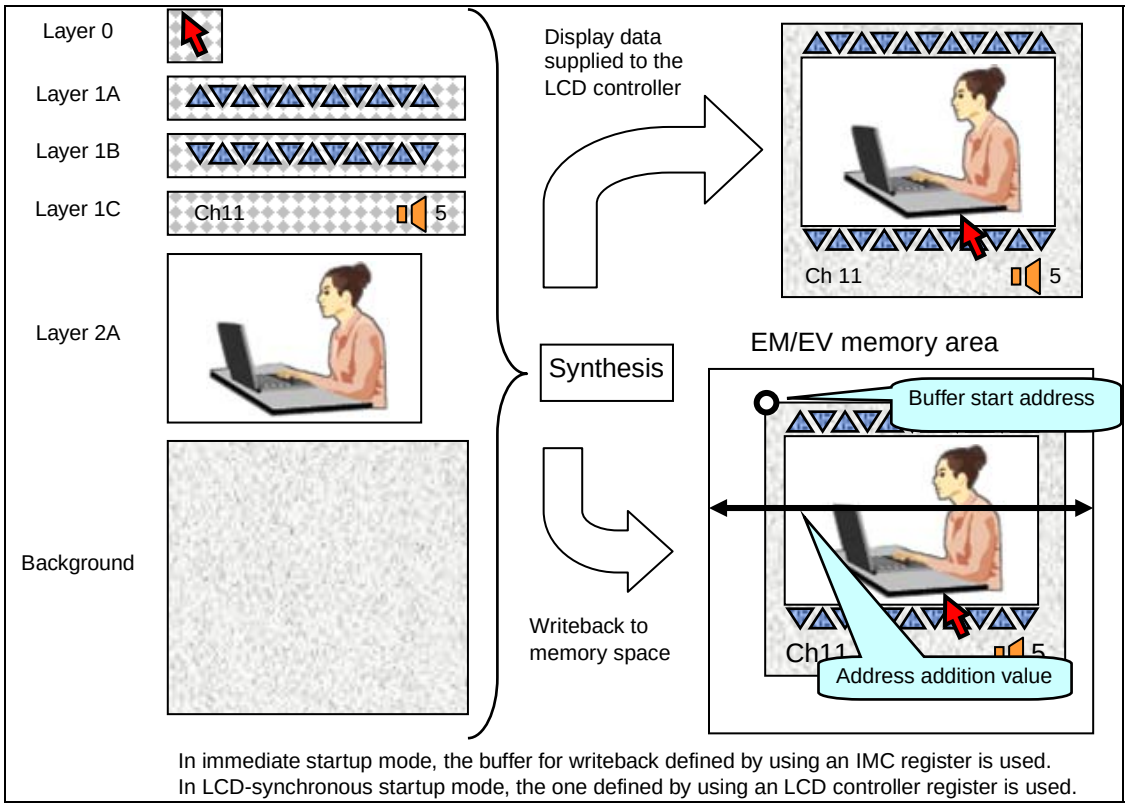
- The IMC writes images back to a frame cache while the supplying data to the LCD controller.
- A writeback might fail (due to an IMC output FIFO overrun).
- When a writeback has failed, it is retried at the next frame.
- After data is written back, the LCD controller directly reads the cache frame via direct path accessing.
- After the IMC enters the direct path mode, power to the L1 domain (including the IMC) is turned off.

Remark VGA standby is a special mode linked with the LCD controller and the MEMC. Use this feature with a thorough understanding of the features of each macro and power control feature.

3.5.2 Writeback to frame cache

If writing back a synthesized image is requested when the IMC runs in the immediate startup mode, the IMC writes back the image to the frame cache memory (the frame area defined using an IMC register). If such a writeback is requested in the LCD-synchronous mode, the IMC writes back the image to the frame area defined using an LCD controller register.
Image data written back here is the same as the display data supplied to the LCD controller during LCD-synchronous operation.

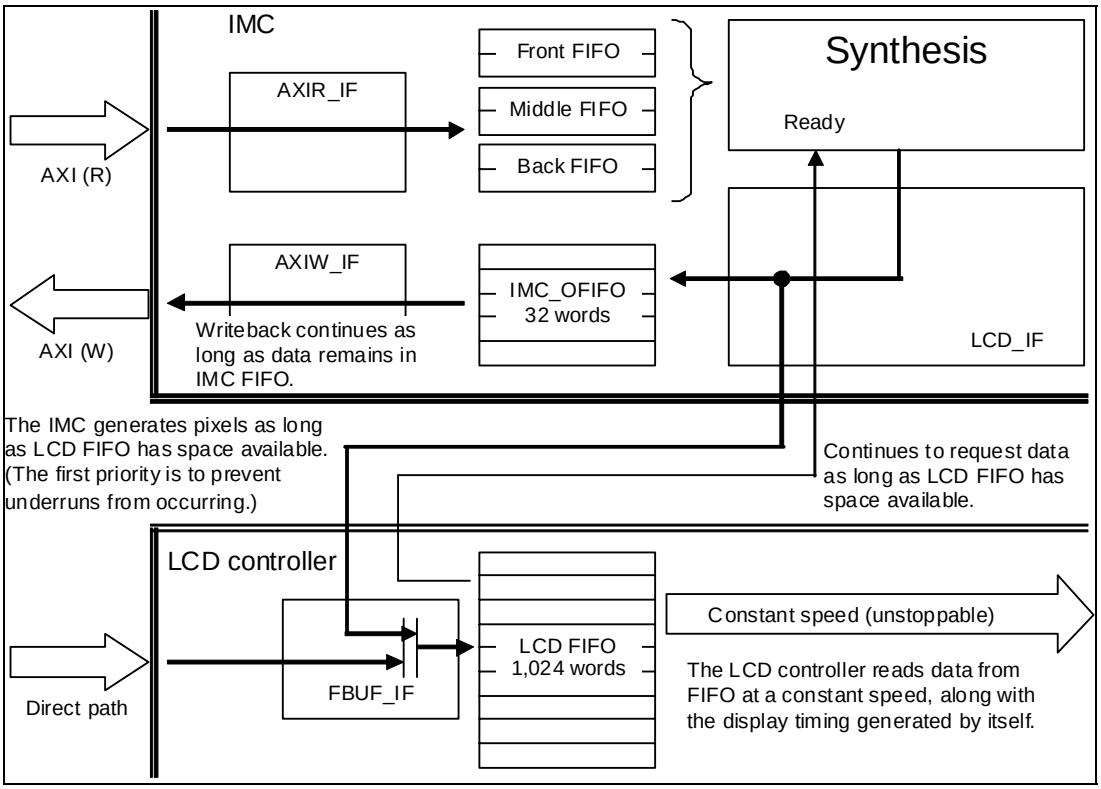
Figure 3-18. Writeback Operation



3.5.3 Overrun

When the IMC runs in the LCD-synchronous mode, a writeback is performed prior to supplying display data to the LCD controller. Depending on the AXI bus status on the write side, therefore, an overrun might occurs in the IMC FIFO. If an overrun occurs, the writeback is repeated because writeback requests from the LCD controller are not canceled. (They can be canceled by using the LCD controller.)
An overrun does not occur during writebacks in the immediate startup mode because writebacks and LCD display are performed asynchronously.

Figure 3-19. Overrun Occurrence Factor



In the above figure, the LCD_IF block in the IMC monitors availability of LCD controller FIFO and issues a request for the block that performs synthesis. The synthesized image is written to the LCD controller, at the same time as written to IMC FIFO.
If an underrun occurs in the LCD controller FIFO, it results in a fatal system error, but writebacks can be started over, in the worst case. Therefore, when a FIFO underrun can occur because data in the LCD FIFO buffer is running short, the IMC synthesizes images and supplies display data to the LCD controller, regardless of the availability of IMC FIFO. As a result, an overrun might occur in the IMC FIFO.

Remark In an actual one-chip environment, synthesis can be performed by setting the DBGMODE bit of the IMC_CONTROL register to 1, by considering the IMC FIFO availability, to obtain the optimum capacity value for FIFO on both the LCD controller and the IMC sides. (A request is issued if FIFO on both sides has space available.)
In this case, an overrun no longer occurs in the IMC FIFO, but underruns become more likely to occur in LCD controller FIFO.

To prevent both underruns and overruns, use the IMC_DATAREQ register to see the FIFO_REMAIN values sent from the LCD controller to allow automatic control of conditions for requesting data for the IMC. If the amount of data in the LCD controller FIFO buffer is sufficient, an underrun does not occur for a while. Therefore, the IMC performs synthesis by considering the available space in the IMC FIFO buffer. If the amount of data in the LCD controller FIFO buffer is running short, an underrun might occur soon. Therefore, the IMC ignores the available space in the IMC FIFO buffer and prioritizes supplying data to the LCD controller when performing synthesis. Use the IMC_DATAREQ register to specify the criteria for whether the amount of data in the LCD controller FIFO buffer is sufficient. If a large value is specified for the IMC_DATAREQ register, an overrun is more likely to occur. If a small value is specified, an underrun is more likely to occur. To suppress the occurrence of both underruns and overruns, specify a value appropriate for your system

The IMC_DATAREQ register and LCD_DATAREQ register for the LCD are a pair. It is generally recommended that the IMC_DATAREQ register use the setting of the LCD_DATAREQ register to open a buffer zone as shown in Figure 3-20. The faster the PXCLK value for the LCD, the wider the buffer zone has to be.

Figure 3-20. How the IMC_DATAREQ Register Works

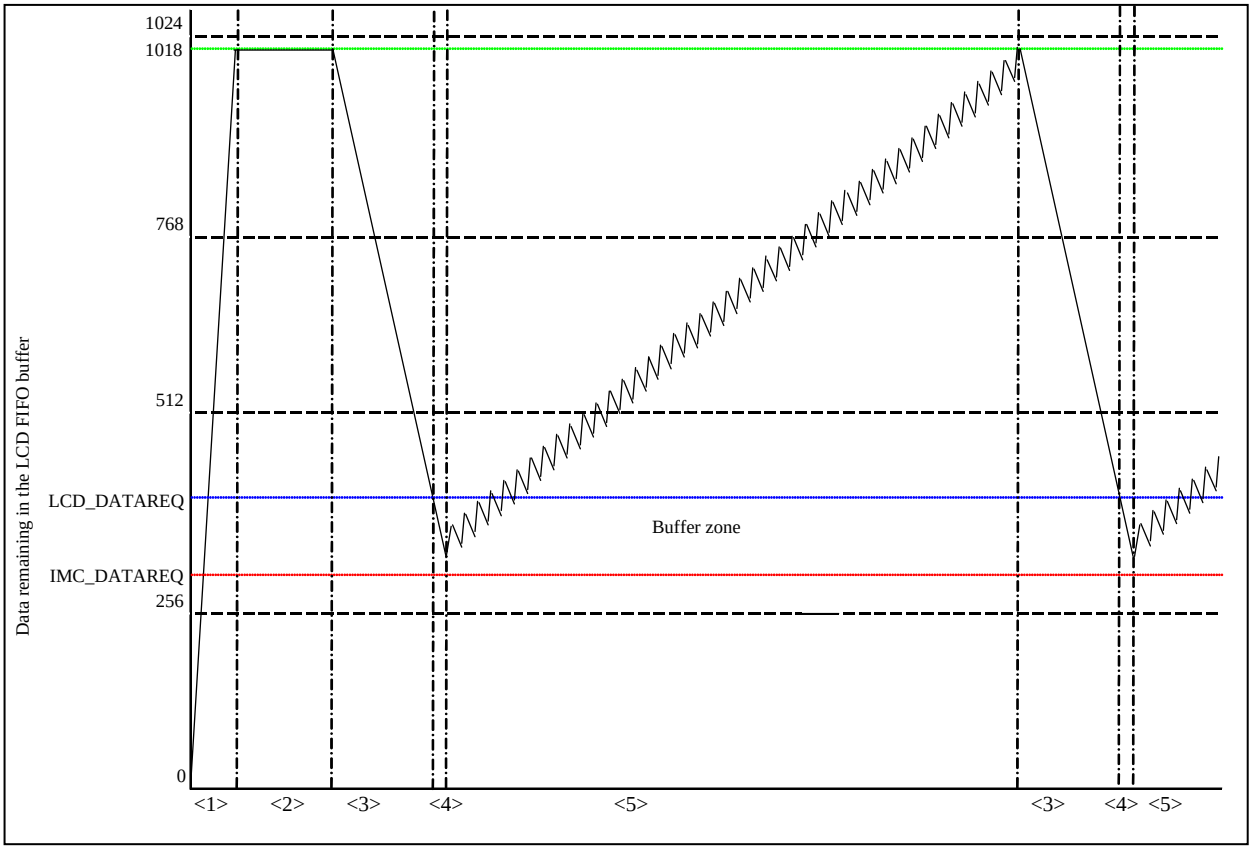


Figure 3-20 shows how the setting of the IMC_DATAREQ register works.

<1> The IMC starts up in the LCD-synchronous mode and writes data to the LCD controller FIFO buffer until it becomes full during the LCD vertical refresh period. No data is read from the LCD controller FIFO buffer.

<2> After the LCD controller FIFO buffer becomes full and before outputting the first line starts, LCD_FIFOREADY remains L and the image processing by the IMC is kept stopped. No data is written to or read from the LCD controller FIFO buffer.

<3> After outputting the first line starts and until the available space in the LCD controller FIFO buffer reaches the value specified for the LCD_DATAREQ register, LCD_FIFOREADY remains L. Data is read from the LCD controller FIFO buffer but not written to it.

<4> When the available space in the LCD controller FIFO buffer becomes less than the value specified for the LCD_DATAREQ register, LCD_FIFOREADY becomes H, and the IMC resumes image processing. Data is read from and written to the LCD controller FIFO buffer at the same time. When the IMC resumes image processing, it takes a while until the backlog of the IMC-internal pipeline is processed or the system bus returns to the normal level. During this period, the throughput of the IMC momentarily drops to a value less than that of the LCD controller and the available space in the LCD controller FIFO buffer continues to decrease.

<5> After the throughput on the IMC side returns to the normal level, the available space in the LCD controller FIFO buffer continuously increases and decreases, and LCD_FIFOREADY does not rise until the LCD controller FIFO buffer becomes full. After that, processing is repeated from <3>.

Processing steps <4> and <5> switch when the available space in the LCD controller FIFO buffer reaches the threshold while normally operating in the LCD-synchronous mode. To prevent overruns, specify a value smaller than this threshold for the IMC_DATAREQ register. However, if too small a value is specified for the LCD_DATAREQ register, specify a larger value for the IMC_DATAREQ register to prioritize preventing underruns.

Caution The setting specified for the IMC_DATAREQ register can reduce the possibility of overruns, but it does not guarantee complete prevention. If an overrun occurs, writing back is retried.

3.5.4 Interrupts related to the standby mode

The IMC can issue the following three types of interrupts related to the standby mode:

Table 3-9. Interrupts Related to Standby Mode

Block	Interrupt Name	Occurrence Timing
IMC	WB end interrupt	This interrupt occurs when the last data of a frame data to write back is written. This is the timing at which writing data is completed at the AXI side of the IMC, but latency occurs until the data is written to the memory by way of a bus switch or bus bridge.
IMC	Overrun interrupt	This interrupt occurs when data is written to the IMC FIFO and it becomes full. When this interrupt occurs, writebacks are not performed until the next frame.
LCD controller	VGA standby shift end interrupt	This interrupt occurs when the last data of a frame data to write back is written, transaction between the MEMC is traced, and data is actually passed to the MEMC. (This interrupt occurs only when a writeback is performed in response to a request from the LCD controller.)

When a writeback is performed in response to a request from the LCD controller, the LCD controller enters the direct path mode after completion of the writeback and the L1 power domain is turned off. When a writeback end interrupt is issued, the IMC has completed the writeback but data might remain in the bus, as seen from the EM/EV system. It is therefore necessary to know when the power can be safely shut down, and this operation is traced between the IMC and MEMC. When this trace is completed, the IMC reports the completion to the LCD controller, and the LCD controller issues a VGA standby shift end interrupt.
When the IMC runs in the immediate startup mode, the LCD controller is assumed to be stopped, or running in direct path mode. The direct path mode presumes the cases where the displayed frame is updated less frequently, and it causes problems concerning power-saving because the IMC must run each time along with the LCD display rate. In this case, the L1 power domain is probably not turned off immediately after completion of the writeback, the IMC is presumed to stop until the next update, and the LCD controller is presumed to switch the address of the frame cache. Therefore, it is not necessary to know when the writeback is completely finished, but rather necessary when the IMC can be activated next time. For this reason, the writeback end interrupt is provided.

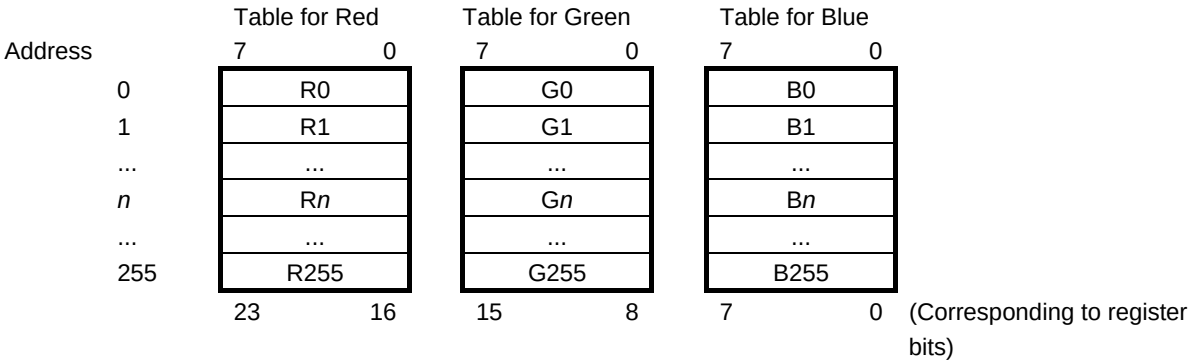
3.6 Gamma Correction

3.6.1 Gamma correction

The IMC performs gamma correction by table referencing. The tables used for conversion include 8 bits each for RGB (equivalent to 256 addresses), and can be accessed by way of the APB bus. Three areas in a 1-port SRAM, each of which consists of 8 bits × 256 words, are used for storing table data. RGB data before correction is handled as the table memory addresses, and the read values are handled as RGB data after correction.
Because it is a 1-port SRAM, accessing by way of the APB bus is possible only when gamma correction is disabled in the gamma correction control register (IMC_GAMMA_EN). Accessing by way of the APB bus is not guaranteed when gamma correction is enabled.
The gamma correction table address register (LCD_GAMMA_ADR) and gamma correction table data register (LCD_GAMMA_DATA) are used for accessing the table memory via APB. To write data to the table, first write the address where the target data is stored, to the gamma correction table address register. Next, write the data to the gamma correction table data register. Data is then written to the table memory indirectly. The gamma correction table address register is automatically incremented when the gamma correction table data register is written, so the table data register can be written successively.
To read data from the table memory, the procedure is the same but the registers must be read twice. First, set the address from which the target data is read, to the gamma correction table address register. Next, read the gamma correction table data register twice. Data is then read from the table memory indirectly, upon the second read.

Bits 23 to 16, bits 15 to 8 and bits 7 to 0 of the gamma correction table data register are assigned for accessing the table memory of red, green, and blue, respectively.
The figure below shows the structure of the gamma correction table. Rn, Gn, and Bn indicate the correction values for red, green, and blue.

Figure 3-21. Gamma Correction Tables



3.6.2 Usage of gamma correction

The following describes the usage of gamma correction:

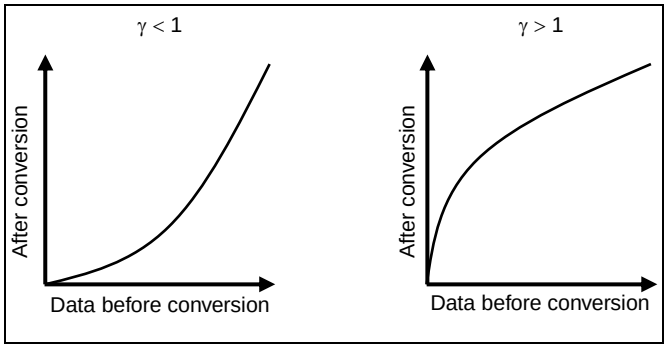
<1> Set IMC_GAMMA_EN to 0. (Do not set it when the IMC is running.)
<2> Set IMC_GAMMA_ADR to 0H.
<3> Write to IMC_GAMMA_DATA the value converted from input value 0.
(IMC_GAMMA_ADR is automatically incremented to 1.)
<4> Write the values converted from input values 1 to 63.
<5> After writing values to all the tables, set IMC_GAMMA_EN to 1 where gamma correction is to be used.

Gamma correction is then performed for images output from the IMC.
Even if <3> and <4> are not performed in succession, any address can be specified each time (by repeating <2> and <3>).

Generally, the relationship between a gamma value (γ) and I/O can be expressed as follows (as reference for calculating table values):

$$Out = In^{\frac{1}{\gamma}}$$

Figure 3-22. Gamma Correction



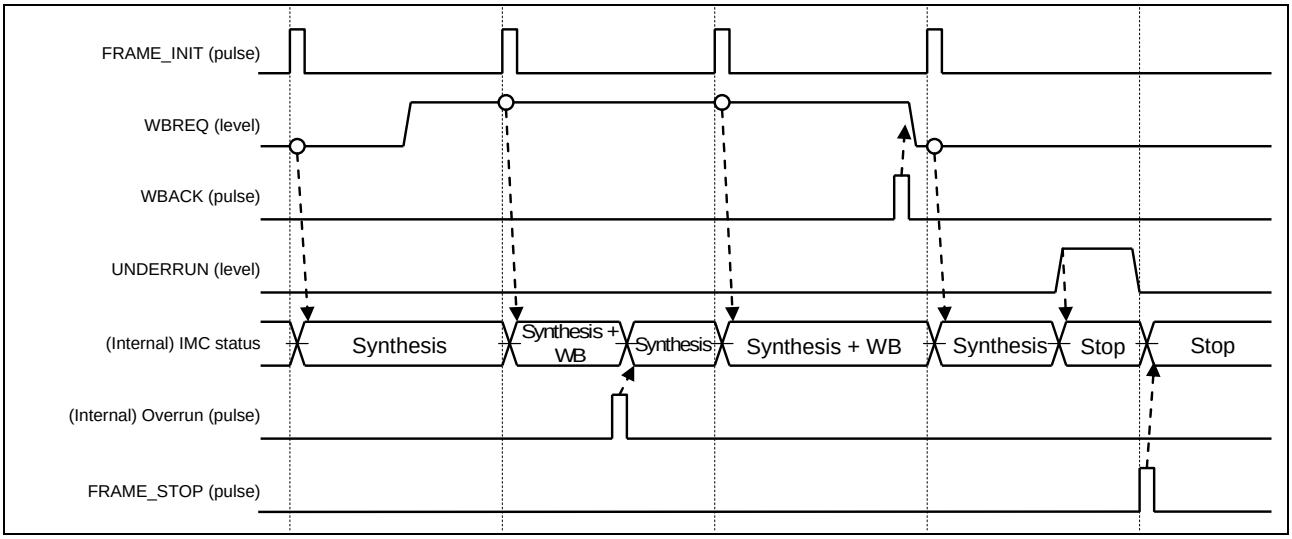
The IMC performs gamma adjustment by referencing tables for any value. Therefore, adjustment such as negative-positive reverse can be specified freely, without being constrained by the above conversion formula.

3.7 Operation Timing

3.7.1 LCD interface

In the LCD-synchronous mode, the IMC starts in synchronization with frame output by the LCD controller. The trigger signal is a pulse signal from the FRAMEINIT pin. A pulse signal from the FRAMESTOP pin is used as the operation stop request signal from the LCD controller. The IMC stops running when it receives this pulse. The IMC determines whether to perform writebacks at the same time as supplying data to the LCD controller, based on the level of the WBREQ pin when the trigger signal is generated. When a writeback is performed, WBACK is issued to indicate completion of WB from the IMC and WBTRACEEND (indicating that no data remains on the bus and therefore the L1 power domain can be turned off) is issued to indicate that the last data in the cache has been completely received in the MEMC. WBTRACEEND is used as an interrupt source for the LCD controller.
The IMC has a mechanism to terminate a frame when UNDERRUN, which indicates the occurrence of an LCD controller buffer underrun, is received. Figure 3-23 shows the timing of control operations performed between the LCD controller and IMC.

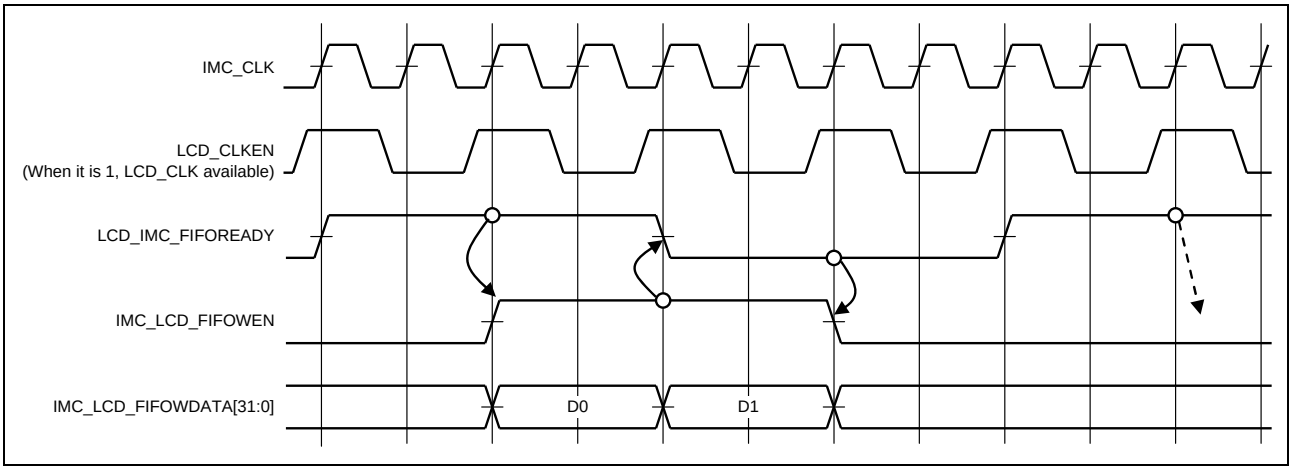
Figure 3-23. Startup Timing Between IMC and LCD Controller



The display data interface between the IMC and the LCD controller operates in accordance with the following rules:

- The LCD controller issues FIFOREADY if the FIFO has an empty space of 8 words.
- When the IMC detects that FIFOREADY is set to 1, it issues FIFOWEN and FIFOWDATA, if there is data that can be output.
- At least 2 clock cycles are required from when FIFOREADY rises to when FIFO_WEN rises.
- If no data is ready on the IMC side, no data is output even if the LCD controller is in the READY state.

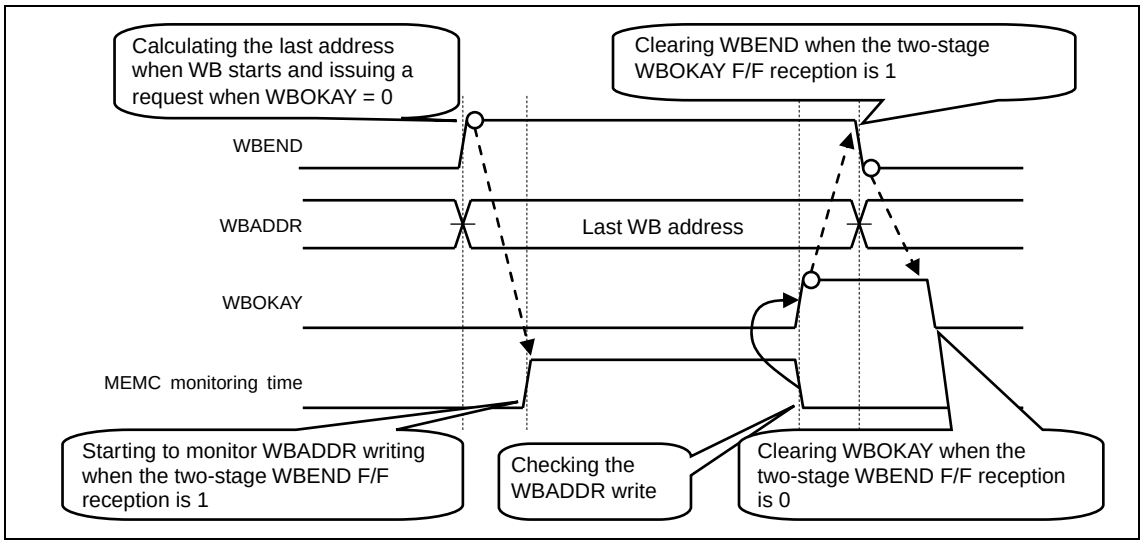
Figure 3-24. Timing of Display Data Interface Between IMC and LCD Controller



The figure below shows the operation that traces when the last data is written back in the LCD-synchronous mode.

Because allocation problems are assumed during communication between the IMC and MEMC, such communication is asynchronous. Therefore, multiple clock cycles are used for this communication, but a handshake is reliable achieved.

Figure 3-25. Timing Related to Writeback End Trace



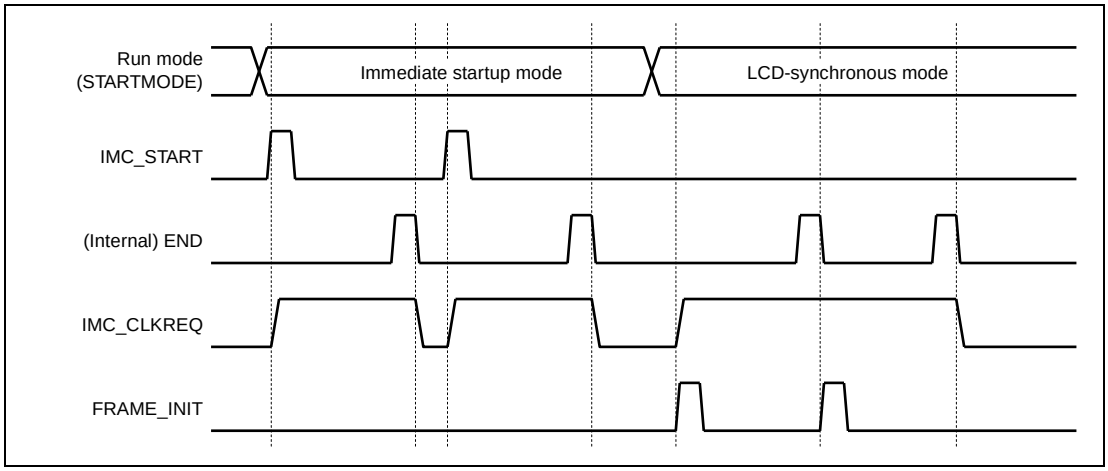
3.7.2 Clock control

Because EM/EV has a low power design, supplying the main clock is controlled for each module. The main clock is supplied when there is a clock request from a module or when the APB clock is supplied due to register access.

In the immediate startup mode, the IMC issues a clock request when the IMCSTART bit of the reserve startup register (IMC_START) is set to 1. This request is canceled when the IMC finishes synthesizing one frame (at the same time as when the WB end interrupt is issued).

When the IMC is running in the LCD-synchronous mode, the IMC requests that a clock be supplied by using the IMC_CLKREQ signal sent from the LCD controller. Using this clock, the IMC sets its CLKREQ value, and then clears it when synthesis is complete. After the LCD controller FIFO buffer becomes full, FIFO_READY rises, all data on the IMC AXI interface is transferred, and then CLKREQ is cleared. When the FIFO_READY signal on the LCD controller side becomes H again, the CLKREQ signal of the IMC is set.

Figure 3-26. IMC Clock Request Timing



In addition, each peripheralof the IMC is capable of automatic clock control. Automatic clock control can be enabled or disabled by using the automatic clock control register.

3.7.3 Register update

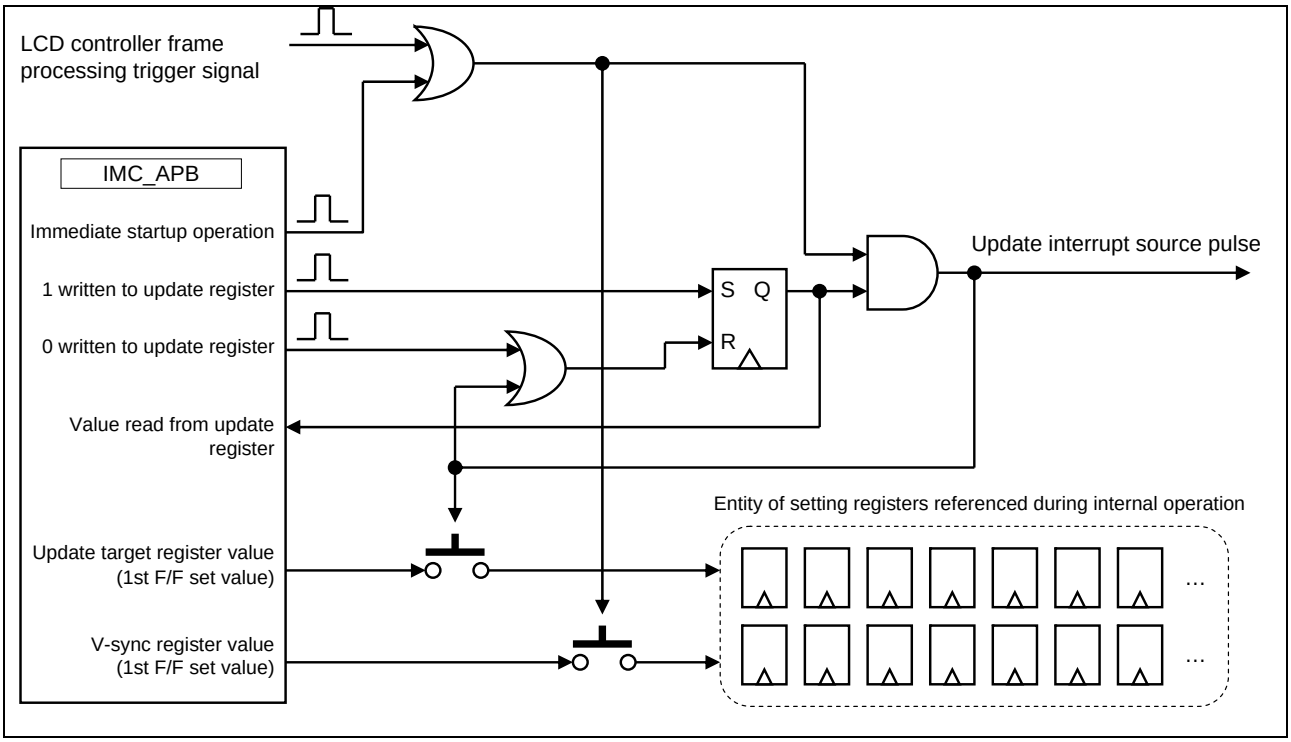
The IMC registers are updated (the specified value is applied to the internal operations) at three types of timing. According to the type, each register is classified as the immediately-reflected register, V-sync register, or update target register.

The entity of the F/Fs of immediately-reflected registers exists in the IMC_APB block, and their values are updated immediately when they are accessed by way of the APB bus. Each sub-module in the IMC operates by directly referencing the values of the F/Fs.

The V-sync and update target registers have the first F/F in the IMC_APB block, and the second F/F, which is referenced during actual circuit operation, in each block. The V-sync registers and update target registers are classified according to the conditions under which the set value is copied from the first F/F to the second F/F. With the V-sync registers, the value is copied when frame processing starts. With the update target registers, the value is copied if the register has been reserved for update when frame processing starts.

Figure 3-27 shows the circuit diagram of these registers and the update register (IMC_UPDATE).

Figure 3-27. Relationship Between Update Register and V-Sync/Update Target Registers

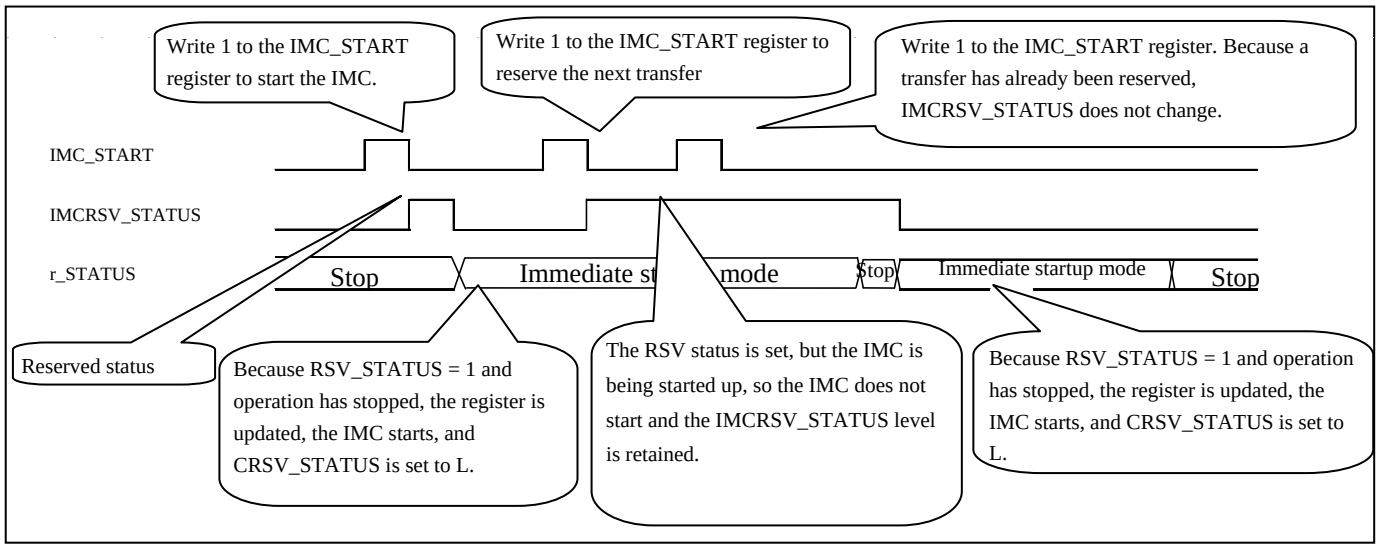


3.7.4 Reserved startup

In the immediate startup mode, the IMC supports reserved startup. Figure 3-28 shows the timing.
When 1 is written to the IMC_START register, the IMC_START register generates one clock pulse. (When passing data from PCLK to the CLK domain, one clock pulse is always generated for the PCLK and CLK domains by way of an FF.)
When the IMC_START register is set to 1, IMCRSV_STATUS is set to 1.
Synthesis processing for a new frame starts only when IMCRSV_STATUS is set to 1 and operation has stopped.
The IMCRSV_STATUS signal falls when synthesis starts.

Caution The register values are updated immediately before synthesis starts. If a register value is changed while xxx is being reserved (IMCRSV_STATUS = 1), the change is applied when synthesis starts the next time.

Figure 3-28. Startup Reservation Timing Chart



3.8 Usage Example

This section describes examples for setting up the LCD controller and the IMC, assuming use in various situations.

3.8.1 When there is no synthesis, there is gamma correction, all windows are horizontally flipped for display, and there are no writebacks

<1> Before startup, specify the gamma table values and enable gamma correction.
<2> Because there is no synthesis, use the BG layer. Disable other layer control registers.
<3> Specify the settings for the BG layer (format, resizing, address addition value, start address), and then store image data in the specified frame-buffer area.
<4> Specify any image flipping setting for the image flipping register.
<5> Set the IMC startup mode to the LCD-synchronization mode, and set the update reservation register to 1.

This concludes the IMC setup. Next, set up the LCD controller:
<6> Specify the SYNC synchronization and pixel size settings for the LCD panel to be connected.
<7> Set the LCD to operate by way of the IMC (without WB processing).
<8> Set up the display startup register.

After these settings are specified, the LCD controller starts displaying frames and issuing data requests to the IMC. The IMC receives these requests, generates image data, and supplies data to the LCD controller.
As long as the LCD controller does not stop, any number of image frames specified by the IMC can be consecutively displayed.
The following describes how to use the IMC to change the setting for horizontally or vertically flipping displayed images.

<9> Specify any image flipping setting (by changing the setting) for the image flipping register at any time.
<10> Set the update reserve register to 1.

After doing the above, the flipping settings take effect when the LCD controller starts displaying a new frame.
It is possible to determine when the settings will take effect according to the update completion interrupt.

If the image data to display has a format other than RGB565, RGB666, or RGB888:
When using the BG layer, a YUV format cannot be selected. YUV formats are supported if using only L2A (and L2B) to display windows.
When specifying the settings in step two, enable L2A. When specifying the settings in <3>, set up the L2A layer instead of the BG layer, set the display position to (0, 0), and set the display size to the output size of the LCD controller. Specifying these settings makes it possible to display all windows using a configuration that includes only L2A.

Note that, when using a YUV format, gain offset register settings for each component are required, as well as settings for the YUV2RGB conversion mode register and custom coefficient. In addition, if image data is stored in memory on multiple planes, the start address for each plane is required. For details,
see **3.2.1 Format selectable for each frame** to **3.2.7 Dithering.**

3.8.2 Using the update register

When performing synthesis, the IMC must change many registers at a time. The update reserve register that updates related registers independently is thus provided to prevent malfunction due to contradictions between settings that have been made extending over frames.
Registers marked with ● in the Frame Sync column in **2.2.1 Registers** are registers subject to update. These are mainly used to set up items related to layer synthesis, which update the settings frequently while successive frames are being displayed when the IMC runs in the LCD-synchronous mode.

General usage:

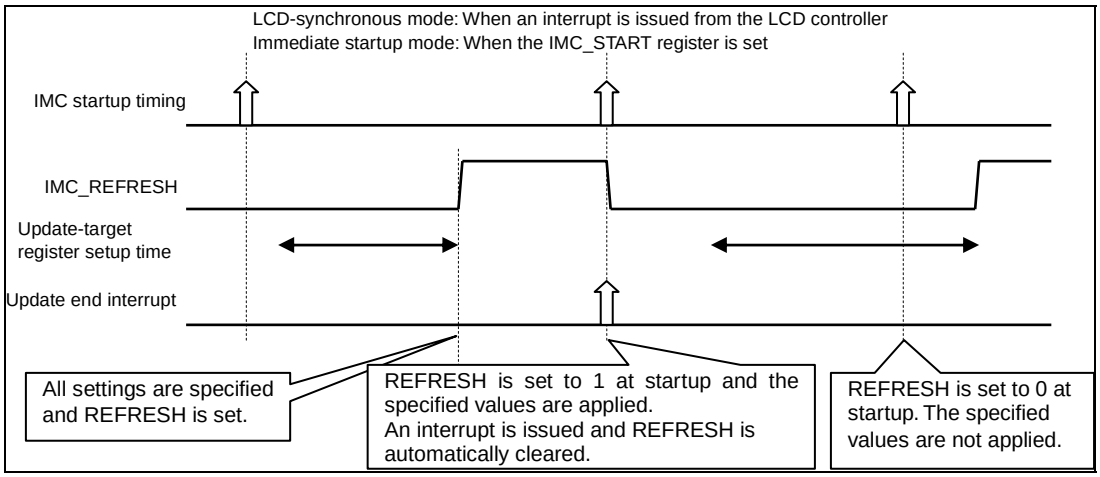
<1> Clear the update reserve register (IMC_REFRESH) to 0.

<2> Change the update target register settings. (Make sure that the settings do not contradict.)

<3> Set the update reserve register (IMC_REFRESH) to 1.

<4> When the IMC starts (when a startup trigger is sent from the LCD controller in the LCD-synchronous mode, and when the IMC_START register is set in the immediate startup mode) while the update reserve register is set to 1, the update reserve register value is cleared when a register update end interrupt occurs, and the values specified in step <2> take effect.

Figure 3-29. Timing Related to Update Register



3.8.3 Displaying cursor (by using L0) and upper and bottom bars (by using L1A and L1C)

The following describes an example of displaying a cursor and upper and bottom bars, which are generally used in cellular phone screens.

<1> Prepare image data for the cursor. Fill the area other than that of the cursor with a color not used in that layer. This color is set as the transparency color in L0.
<2> Specify settings related to L0 (format, transparency color control, transparency color, alpha blending, resizing, address addition value, start address, display size, and display position) and enable the L0 control register.
<3> Prepare image data for the upper bar (this area generally shows information such as the battery power, antenna strength, and silent mode status). Fill the background portions to be seen in this layer, with a color not used in that layer. This color is specified as the transparency color in L1.
<4> Specify settings related to L1A (format, transparency color control, transparency color, alpha blending, resizing, address addition value, start address, display size, and display position) and enable the L1A control register.
<5> Prepare image data for the bottom bar (this area generally shows the explanation of functions on the function buttons arranged under the LCD screen). Fill the background portions to be seen in this layer, with the same color as used in step <3>.
<6> Specify settings related to L1C (format, transparency color control, transparency color, alpha blending, resizing, address addition value, start address, display size, and display position) and enable the L1C control register.
<7> Specify settings on the BG when using BG, or on L2 when using L2 for the background.
<8> Set the update reserve register to 1.

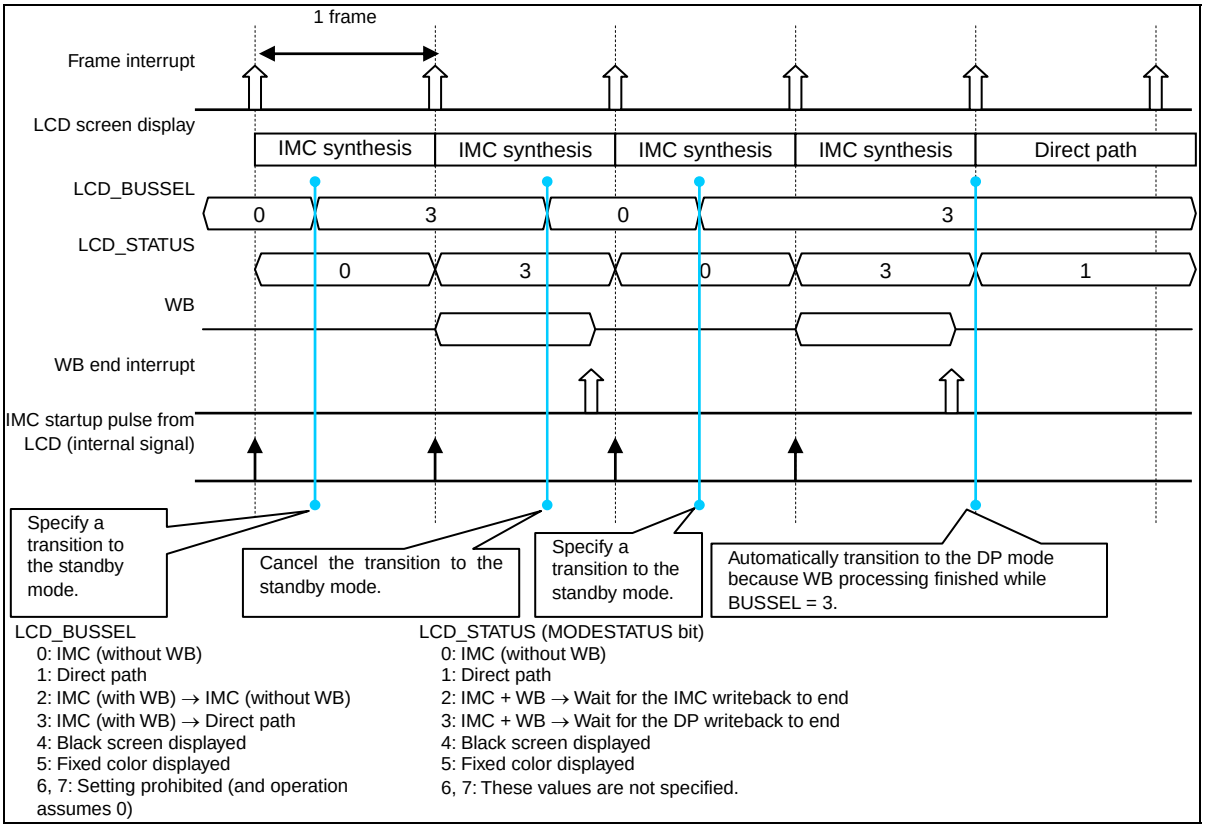
Specify the LCD controller settings in the same manner as described in **3.8.1 (1) How to use update reserve register**. The cursor and upper and bottom bands can then be displayed on the background set up in step <7>. To move the cursor position, perform the following:

<9> Change the display position of L0 to any coordinate.
<10> Set the update reserve register to 1.

3.8.4 Canceling a transition to the low-power mode

This section describes how to cancel a transition to the low-power mode when the displayed data is updated while using the function for automatically transitioning to the LCD direct path (DP) mode (that is, while waiting for WB processing to finish because LCD_BUSSEL is set to 3 (which specifies that there is local bus and WB processing and that the system will wait for WB processing to finish and then automatically switch to the direct path mode)).

Figure 3-30. Canceling an Automatic Transition to the DP Mode Due to a WB Request



As shown in the figure, the LCD controller determines the operating mode for each frame according to the value specified by LCD_BUSSEL. Because BUSSEL = 3 for the 2nd frame, the IMC is started during the next frame with WB processing.

For the 3rd frame, the WB processing has finished, but, because BUSSEL = 0, the IMC is started without WB processing. For the 4th frame, the IMC is again started with WB processing, and, for the 5th frame, because the WB processing has finished, the system automatically switches to the direct path mode.

When the LCD receives display data from the IMC, the startup pulse shown at the bottom of the figure is generated. At this time, if the IMC update register has the reserved status, the register processing is performed. Therefore, because the startup pulse is not generated during direct path display in the 5th frame, no register update is performed. However, caution is required because, if the register has the reserved status, it is updated when the next startup pulse is generated.

However, if the L1 power supply area allocated by the IMC is disabled while the system is in the direct path mode, the reserved status is canceled because the IMC is initialized.

APPENDIX. TERMINOLOGY

Frame

An independent image used during synthesis and the smallest unit used for synthesis processing. The IMC has seven frames (L0, L1A, L1B, L1C, L2A, L2B, and BG).

Layer

A combination of multiple frames placed at the same depth during synthesis. The seven frames of the IMC are divided into four layers depending on the depth (layer 0, layer 1, layer 2, and the BG layer).

Process

The IMC performs processing by using three parallel processes. Layer 0 and layer 1 are processed during the front and middle processes, respectively, while layer 2 and the BG layer are processed during the back process.

Layer synthesis

Processing to superimpose multiple layers to generate a new image

Frame buffer

Memory area for storing images. In the IMC, it might see a specific memory area.

Plane

Supporting double buffering and the interlaced scanning method makes two frame buffers available for each frame. These buffers are called P and Q.

Main image and sub-image

Because wrapping around is supported, a portion of an image stored in memory sometimes exceeds the boundaries during synthesis processing. In addition, only a portion of an image stored in memory is sometimes used for overwriting during synthesis processing. In such cases, an image stored in memory is called a *main image*, and a portion that exceeds the boundaries is called a *sub image*.

Background (BG)

In the IMC, it means the background frame.

Alpha blending (α blending)

Function to modify the transparency of images while superimposing images to make multiple images to be seen

Key color (mask color, transparency color)

Function to specify transparency colors while superimposing images to make the background layers to be seen as they are

Color space conversion

Two methods are available to express color data in numeric values: one is RGB, which uses three primary colors - red, green, and blue, and the other is YUV, which uses the luminance (γ) and color difference components (U and V). Conversion between these formats is called color space conversion.

RGB565 and RGB888

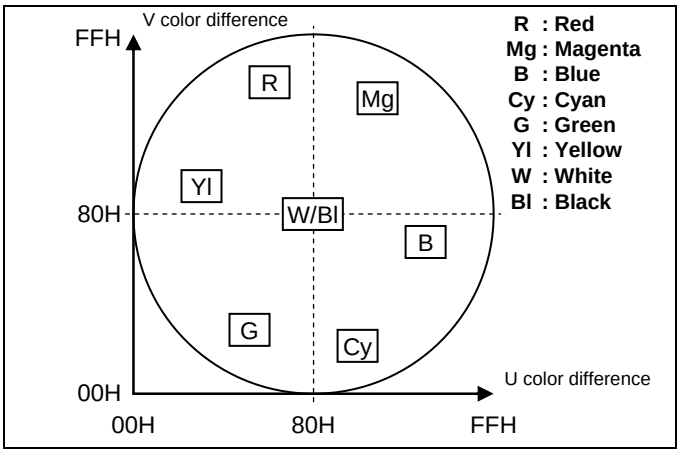
When color data is expressed in the RGB format, the format in which each component is scaled by 8 bits is called RGB888, the format in which each component is scaled by 6 bits is called RGB666, and the format in which G is scaled by 6 bits and R and B are scaled by 5 bits is called RGB565.

ARGB1555, ARGB4444, and ARGB8888

In addition to the color data of the RGB format, this format adds a parameter to each pixel that indicates the degree of transparency for alpha blending. The value after *ARGB* indicates the number of bits for each component. Using any of these formats makes it possible to achieve different alpha blending effects for different pixels, even in the same frame.

YUV422 and YUV420 (YUV444)

When color data is expressed in a YUV format, the format that defines a single color difference component for one luminance component is called YUV444, the format that defines a single color difference component for luminance data of two horizontal pixels is called YUV422, and the format that defines a single color difference component for luminance data of two horizontal and vertical pixels, four pixels in total, is called YUV420. The hue expressed with UV is illustrated as shown below. (The U and V components are represented in offset binary coding with 80H positioned as the center.)



Dithering

The method of displaying colors by combining displayable colors to express intermediate colors with few available colors. The image is displayed less sharp, but this method shows images as if the number of displayable colors is increased.

The IMC uses systematic dithering using a 2 × 2 matrix for converting RGB888 to RGB666 (color masking).

Writeback (WB)

Processing to write back display image data created by the IMC to the frame cache area which is specified in a system memory space. The written back data is used in the direct path mode in the LCD controller.

REVISION HISTORY	EMMA Mobile EV2 User's Manual: Image Composer

Rev.	Date	Description	
		Page	Summary
1.00	Jan 29, 2010	—	1st revision release
2.00	Jun 7, 2010	—	Incremental update from comments to the 1.0.
3.00	Sep 30, 2010	—	Incremental update from comments to the 2.0. (A change part from the old revision is "★" marked in the page left end.)
4.00	Apr 15, 2011	—	Incremental update from comments to the 3.0. (A change part from the old revision is "★" marked in the page left end.)
5.00	May 31, 2011	—	Incremental update from comments to the 4.0.
6.00	Sep 30, 2011	—	Incremental update from comments to the 5.0.
		68	Chapter 2.2.45(1) corrected. (bit[5] added.)
		69	Chapter 2.2.45(2) corrected. (bit[5] added.)
		70	Chapter 2.2.45(3) corrected. (bit[5] added.)
		72	Chapter 2.2.45(4) corrected. (bit[5] added.)
		73	Chapter 2.2.45(5) corrected. (bit[5] added.)
7.00	Dec 21, 2011	—	Incremental update from comments to the 6.0.
		5	Chapter 1.1.7 added.

EMMA Mobile EV2 User's Manual: Image Composer

Publication Date: Rev.1.00 Jan 29, 2010
Rev.7.00 Dec 21, 2011

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
7F, No. 363 Fu Shing North Road Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.0

Image Composer

EMMA Mobile EV2

Renesas Electronics Corporation

R19UH0038EJ0700