User's Manual

DMA Controller

User's Manual

Multimedia Processor for Mobile Applications
EMMA Mobile ™ EV2

Renesas Electronics
www.renesas.com

Rev.3.00 May 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 (This manual)	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	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

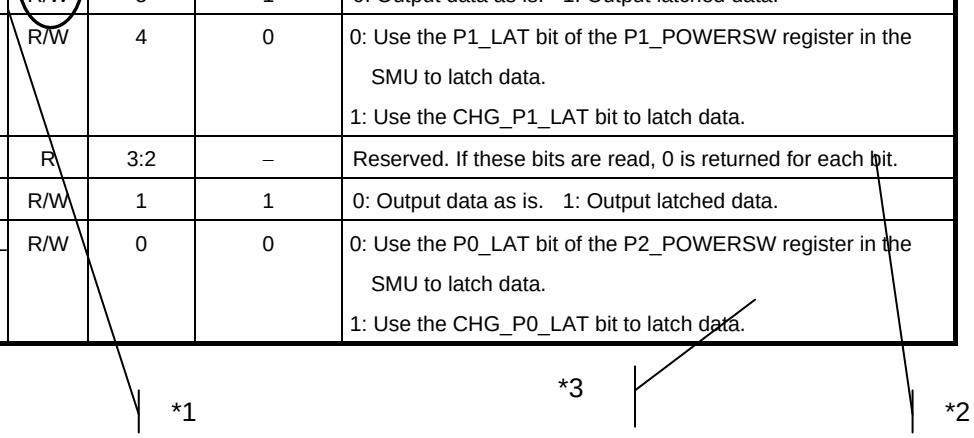
The symbols and terms used in register diagrams are described below.

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

Table of Contents

1. Overview

This document describes the DMA controller (DMAC) for EM/EV.

- Three physical channels (PCH0, PCH2, and PCH3) and 24 logical channels (LCH*x*) are incorporated. (PCH1 is a reserved channel).
  - PCH0 (LCH0 to LCH7): Memory-to-memory transfer
  - PCH2 (LCH0 to LCH7): Memory-to-peripheral transfer.
  - PCH3 (LCH0 to LCH7): Peripheral-to-memory transfer.

2. Registers

2.1 Register Overview

The DMA controller registers are used to control transfers between memory and memory, memory-to-peripheral, and peripheral-to-memory transfers.

(1) Addresses of setting registers

Table 2-1. Offset Address of Each Channel Parameter Setting Registers

Bits 15 to 12	Bits 11 to 8	Bits 7 to 0			
Physical Channels	Logical Channels	Register Functions			
				R/W	Bit Width
1: PCH0 3: PCH1 (Reserved) 5: PCH2 7: PCH3	0: LCH0 1: LCH1 2: LCH2 3: LCH3 4: LCH4 5: LCH5 6: LCH6 7: LCH7	00:	Source address	R/W	32
		04:	Source address pointer	R	32
		08:	Source address offset	R/W	17
		0C:	Source block size (only available for PCH2)	R/W	16
		10:	Source block count	R/W	6
		20:	Destination address	R/W	32
		24:	Destination address pointer	R	32
		28:	Destination address offset	R/W	17
		2C:	Destination block size (only available for PCH3)	R/W	16
		30:	Destination block count	R/W	6
		40:	Length	R/W	24
		44:	Read length count	R	24
		48:	Write length count	R	24
		4C:	Block size (only available for PCH0)	R/W	16
		50:	Mode	R/W	–
		54:	Timer (only available for PCH2 and PCH3)	R/W	24
		58:	Timer count (only available for PCH2 and PCH3)	R	24
		5C:	Physical channel (only available for PCH2 and PCH3)	R	5

Table 2-2. Offset Address of Each Channel Control and Interrupt Parameter Setting Registers

Bits 15 to 12	Bits 11 to 8	Bits 7 to 5	Bits 4 to 0		R/W	Bit Width
0: PCH0 2: PCH1 (Reserved) 4: PCH2 6: PCH3	0:	0:	00:	DMA control register	W	–
			04:	DMA status register	R	–
			08:	DMA end control register	W	–
	1:	0: LCH0 to LCH3 1: LCH4 to LCH7	00:	Interrupt status	R	–
			04:	Interrupt raw status	R	–
			08:	Interrupt enable set	R/W	–
			0C:	Interrupt enable clear	R/W	–
			10:	Interrupt source clear	W	–

(2) Types of setting registers

The base addresses of DMA controller registers are as follows:

- Memory-to-memory transfer registers: E109_0000H
- Memory-to-peripheral transfer registers: E107_0000H
- Peripheral-to-memory transfer registers: E108_0000H

Table 2-3. DMAC Address Map

Parameter	Address Range	Register Name
M2M (PCH0)	0000H to 00FFH	Control registers
	0100H to 0FFFH	Interrupt registers
	1000H to 10FFH	LCH0 setting registers
	1100H to 11FFH	LCH1 setting registers
	1200H to 12FFH	LCH2 setting registers
	1300H to 13FFH	LCH3 setting registers
	1400H to 14FFH	LCH4 setting registers
	1500H to 15FFH	LCH5 setting registers
	1600H to 16FFH	LCH6 setting registers
	1700H to 17FFH	LCH7 setting registers
	1800H to 1FFFH	Reserved
M2P (PCH2)	0000H to 00FFH	Control registers
	0100H to 0FFFH	Interrupt registers
	1000H to 10FFH	LCH0 setting registers
	1100H to 11FFH	LCH1 setting registers
	1200H to 12FFH	LCH2 setting registers
	1300H to 13FFH	LCH3 setting registers
	1400H to 14FFH	LCH4 setting registers
	1500H to 15FFH	LCH5 setting registers
	1600H to 16FFH	LCH6 setting registers
	1700H to 17FFH	LCH7 setting registers
	1800H to 1FFFH	Reserved
P2M (PCH3)	0000H to 00FFH	Control registers
	0100H to 0FFFH	Interrupt registers
	1000H to 10FFH	LCH0 setting registers
	1100H to 11FFH	LCH1 setting registers
	1200H to 12FFH	LCH2 setting registers
	1300H to 13FFH	LCH3 setting registers
	1400H to 14FFH	LCH4 setting registers
	1500H to 15FFH	LCH5 setting registers
	1600H to 16FFH	LCH6 setting registers
	1700H to 17FFH	LCH7 setting registers
	1800H to 1FFFH	Reserved

2.2 Registers

Do not access reserved registers. An undefined value is returned for a read access.
Do not write any value other than 0 to reserved bits in each register.

2.2.1 M2M (memory-to-memory) transfer registers

Base address: E109_0000H

(1) M2M DMA control registers

Address	Register Name	Symbol	R/W	After Reset
0000H	DMA start control register	CONT	W	0000_0000H
0004H	DMA control status register	CONTSTATUS	R	0000_0000H
0008H	DMA end control register	END	W	0000_0000H
000CH to 00FCH	Reserved	–	–	–

(2) M2M interrupt parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
0100H	Interrupt status register (LCH0 to LCH3)	LCH0LCH3_INT_CONT	R	0000_0000H
0104H	Interrupt raw status register (LCH0 to LCH3)	LCH0LCH3_INT_RAW	R	0000_0000H
0108H	Interrupt enable set register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE	R/W	0000_0000H
010CH	Interrupt enable clear register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE_CL	W	0000_0000H
0110H	Interrupt source clear register (LCH0 to LCH3)	LCH0LCH3_INT_REQ_CL	W	0000_0000H
0114H to 011CH	Reserved	–	–	–
0120H	Interrupt status register (LCH4 to LCH7)	LCH4LCH7_INT_CONT	R	0000_0000H
0124H	Interrupt raw status register (LCH4 to LCH7)	LCH4LCH7_INT_RAW	R	0000_0000H
0128H	Interrupt enable set register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE	R/W	0000_0000H
012CH	Interrupt enable clear register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE_CL	W	0000_0000H
0130H	Interrupt source clear register (LCH4 to LCH7)	LCH4LCH7_INT_REQ_CL	W	0000_0000H
0134H to 01FCH	Reserved	–	–	–

(3) M2M LCH0 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1000H	LCH0 source address register (start address)	LCH0_AADD	R/W	0000_0000H
1004H	LCH0 source address pointer register	LCH0_AADP	R	0000_0000H
1008H	LCH0 source address offset register	LCH0_AOFF	R/W	0000_0000H
100CH	Reserved	–	–	–
1010H	LCH0 source block count register	LCH0_ASIZE_COUNT	R/W	0000_0000H
1014H to 101CH	Reserved	–	–	–
1020H	LCH0 destination address register (start address)	LCH0_BADD	R/W	0000_0000H
1024H	LCH0 destination address pointer register	LCH0_BADP	R	0000_0000H
1028H	LCH0 destination address offset register	LCH0_BOFF	R/W	0000_0000H
102CH	Reserved	–	–	–
1030H	LCH0 destination block count register	LCH0_BSIZE_COUNT	R/W	0000_0000H
1034H to 103CH	Reserved	–	–	–
1040H	LCH0 length register	LCH0_LENG	R/W	0000_0000H
1044H	LCH0 read length count register	LCH0_LENG_RCOUNT	R	0000_0000H
1048H	LCH0 write length count register	LCH0_LENG_WCOUNT	R	0000_0000H
104CH	LCH0 block size register	LCH0_SIZE	R/W	0000_0000H
1050H	LCH0 mode register	LCH0_MODE	R/W	E4E4_0000H
1054H to 10FCH	Reserved	–	–	–

(4) M2M LCH1 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1100H	LCH1 source address register (start address)	LCH1_AADD	R/W	0000_0000H
1104H	LCH1 source address pointer register	LCH1_AADP	R	0000_0000H
1108H	LCH1 source address offset register	LCH1_AOFF	R/W	0000_0000H
110CH	Reserved	–	–	–
1110H	LCH1 source block count register	LCH1_ASIZE_COUNT	R/W	0000_0000H
1114H to 111CH	Reserved	–	–	–
1120H	LCH1 destination address register (start address)	LCH1_BADD	R/W	0000_0000H
1124H	LCH1 destination address pointer register	LCH1_BADP	R	0000_0000H
1128H	LCH1 destination address offset register	LCH1_BOFF	R/W	0000_0000H
112CH	Reserved	–	–	–
1130H	LCH1 destination block count register	LCH1_BSIZE_COUNT	R/W	0000_0000H
1134H to 113CH	Reserved	–	–	–
1140H	LCH1 length register	LCH1_LENG	R/W	0000_0000H
1144H	LCH1 read length count register	LCH1_LENG_RCOUNT	R	0000_0000H
1148H	LCH1 write length count register	LCH1_LENG_WCOUNT	R	0000_0000H
114CH	LCH1 block size register	LCH1_SIZE	R/W	0000_0000H
1150H	LCH1 mode register	LCH1_MODE	R/W	E4E4_0000H
1154H to 11FCH	Reserved	–	–	–

(5) M2M LCH2 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1200H	LCH2 source address register (start address)	LCH2_AADD	R/W	0000_0000H
1204H	LCH2 source address pointer register	LCH2_AADP	R	0000_0000H
1208H	LCH2 source address offset register	LCH2_AOFF	R/W	0000_0000H
120CH	Reserved	–	–	–
1210H	LCH2 source block count register	LCH2_ASIZE_COUNT	R/W	0000_0000H
1214H to 121CH	Reserved	–	–	–
1220H	LCH2 destination address register (start address)	LCH2_BADD	R/W	0000_0000H
1224H	LCH2 destination address pointer register	LCH2_BADP	R	0000_0000H
1228H	LCH2 destination address offset register	LCH2_BOFF	R/W	0000_0000H
122CH	Reserved	–	–	–
1230H	LCH2 destination block count register	LCH2_BSIZE_COUNT	R/W	0000_0000H
1234H to 123CH	Reserved	–	–	–
1240H	LCH2 length register	LCH2_LENG	R/W	0000_0000H
1244H	LCH2 read length count register	LCH2_LENG_RCOUNT	R	0000_0000H
1248H	LCH2 write length count register	LCH2_LENG_WCOUNT	R	0000_0000H
124CH	LCH2 block size register	LCH2_SIZE	R/W	0000_0000H
1250H	LCH2 mode register	LCH2_MODE	R/W	E4E4_0000H
1254H to 12FCH	Reserved	–	–	–

(6) M2M LCH3 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1300H	LCH3 source address register (start address)	LCH3_AADD	R/W	0000_0000H
1304H	LCH3 source address pointer register	LCH3_AADP	R	0000_0000H
1308H	LCH3 source address offset register	LCH3_AOFF	R/W	0000_0000H
130CH	Reserved	–	–	–
1310H	LCH3 source block count register	LCH3_ASIZE_COUNT	R/W	0000_0000H
1314H to 131CH	Reserved	–	–	–
1320H	LCH3 destination address register (start address)	LCH3_BADD	R/W	0000_0000H
1324H	LCH3 destination address pointer register	LCH3_BADP	R	0000_0000H
1328H	LCH3 destination address offset register	LCH3_BOFF	R/W	0000_0000H
132CH	Reserved	–	–	–
1330H	LCH3 destination block count register	LCH3_BSIZE_COUNT	R/W	0000_0000H
1334H to 133CH	Reserved	–	–	–
1340H	LCH3 length register	LCH3_LENG	R/W	0000_0000H
1344H	LCH3 read length count register	LCH3_LENG_RCOUNT	R	0000_0000H
1348H	LCH3 write length count register	LCH3_LENG_WCOUNT	R	0000_0000H
134CH	LCH3 block size register	LCH3_SIZE	R/W	0000_0000H
1350H	LCH3 mode register	LCH3_MODE	R/W	E4E4_0000H
1354H to 13FCH	Reserved	–	–	–

(7) M2M LCH4 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1400H	LCH4 source address register (start address)	LCH4_AADD	R/W	0000_0000H
1404H	LCH4 source address pointer register	LCH4_AADP	R	0000_0000H
1408H	LCH4 source address offset register	LCH4_AOFF	R/W	0000_0000H
140CH	Reserved	–	–	–
1410H	LCH4 source block count register	LCH4_ASIZE_COUNT	R/W	0000_0000H
1414H to 141CH	Reserved	–	–	–
1420H	LCH4 destination address register (start address)	LCH4_BADD	R/W	0000_0000H
1424H	LCH4 destination address pointer register	LCH4_BADP	R	0000_0000H
1428H	LCH4 destination address offset register	LCH4_BOFF	R/W	0000_0000H
142CH	Reserved	–	–	–
1430H	LCH4 destination block count register	LCH4_BSIZE_COUNT	R/W	0000_0000H
1434H to 143CH	Reserved	–	–	–
1440H	LCH4 length register	LCH4_LENG	R/W	0000_0000H
1444H	LCH4 read length count register	LCH4_LENG_RCOUNT	R	0000_0000H
1448H	LCH4 write length count register	LCH4_LENG_WCOUNT	R	0000_0000H
144CH	LCH4 block size register	LCH4_SIZE	R/W	0000_0000H
1450H	LCH4 mode register	LCH4_MODE	R/W	E4E4_0000H
1454H to 14FCH	Reserved	–	–	–

(8) M2M LCH5 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1500H	LCH5 source address register (start address)	LCH5_AADD	R/W	0000_0000H
1504H	LCH5 source address pointer register	LCH5_AADP	R	0000_0000H
1508H	LCH5 source address offset register	LCH5_AOFF	R/W	0000_0000H
150CH	Reserved	–	–	–
1510H	LCH5 source block count register	LCH5_ASIZE_COUNT	R/W	0000_0000H
1514H to 151CH	Reserved	–	–	–
1520H	LCH5 destination address register (start address)	LCH5_BADD	R/W	0000_0000H
1524H	LCH5 destination address pointer register	LCH5_BADP	R	0000_0000H
1528H	LCH5 destination address offset register	LCH5_BOFF	R/W	0000_0000H
152CH	Reserved	–	–	–
1530H	LCH5 destination block count register	LCH5_BSIZE_COUNT	R/W	0000_0000H
1534H to 153CH	Reserved	–	–	–
1540H	LCH5 length register	LCH5_LENG	R/W	0000_0000H
1544H	LCH5 read length count register	LCH5_LENG_RCOUNT	R	0000_0000H
1548H	LCH5 write length count register	LCH5_LENG_WCOUNT	R	0000_0000H
154CH	LCH5 block size register	LCH5_SIZE	R/W	0000_0000H
1550H	LCH5 mode register	LCH5_MODE	R/W	E4E4_0000H
1554H to 15FCH	Reserved	–	–	–

(9) M2M LCH6 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1600H	LCH6 source address register (start address)	LCH6_AADD	R/W	0000_0000H
1604H	LCH6 source address pointer register	LCH6_AADP	R	0000_0000H
1608H	LCH6 source address offset register	LCH6_AOFF	R/W	0000_0000H
160CH	Reserved	–	–	–
1610H	LCH6 source block count register	LCH6_ASIZE_COUNT	R/W	0000_0000H
1614H to 161CH	Reserved	–	–	–
1620H	LCH6 destination address register (start address)	LCH6_BADD	R/W	0000_0000H
1624H	LCH6 destination address pointer register	LCH6_BADP	R	0000_0000H
1628H	LCH6 destination address offset register	LCH6_BOFF	R/W	0000_0000H
162CH	Reserved	–	–	–
1630H	LCH6 destination block count register	LCH6_BSIZE_COUNT	R/W	0000_0000H
1634H to 163CH	Reserved	–	–	–
1640H	LCH6 length register	LCH6_LENG	R/W	0000_0000H
1644H	LCH6 read length count register	LCH6_LENG_RCOUNT	R	0000_0000H
1648H	LCH6 write length count register	LCH6_LENG_WCOUNT	R	0000_0000H
164CH	LCH6 block size register	LCH6_SIZE	R/W	0000_0000H
1650H	LCH6 mode register	LCH6_MODE	R/W	E4E4_0000H
1654H to 16FCH	Reserved	–	–	–

(10) M2M LCH7 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1700H	LCH7 source address register (start address)	LCH7_AADD	R/W	0000_0000H
1704H	LCH7 source address pointer register	LCH7_AADP	R	0000_0000H
1708H	LCH7 source address offset register	LCH7_AOFF	R/W	0000_0000H
170CH	Reserved	–	–	–
1710H	LCH7 source block count register	LCH7_ASIZE_COUNT	R/W	0000_0000H
1714H to 171CH	Reserved	–	–	–
1720H	LCH7 destination address register (start address)	LCH7_BADD	R/W	0000_0000H
1724H	LCH7 destination address pointer register	LCH7_BADP	R	0000_0000H
1728H	LCH7 destination address offset register	LCH7_BOFF	R/W	0000_0000H
172CH	Reserved	–	–	–
1730H	LCH7 destination block count register	LCH7_BSIZE_COUNT	R/W	0000_0000H
1734H to 173CH	Reserved	–	–	–
1740H	LCH7 length register	LCH7_LENG	R/W	0000_0000H
1744H	LCH7 read length count register	LCH7_LENG_RCOUNT	R	0000_0000H
1748H	LCH7 write length count register	LCH7_LENG_WCOUNT	R	0000_0000H
174CH	LCH7 block size register	LCH7_SIZE	R/W	0000_0000H
1750H	LCH7 mode register	LCH7_MODE	R/W	E4E4_0000H
1754H to 17FCH	Reserved	–	–	–

2.2.2 M2P (memory-to-peripheral) transfer registers

Base address: E107_0000H

(1) M2P DMA control registers

Address	Register Name	Symbol	R/W	After Reset
0000H	DMA start control register	CONT	W	0000_0000H
0004H	DMA control status register	CONTSTATUS	R	0000_0000H
0008H	DMA end control register	END	W	0000_0000H
000CH to 00FCH	Reserved	–	–	–

(2) M2P interrupt parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
0100H	Interrupt status register (LCH0 to LCH3)	LCH0LCH3_INT_CONT	R	0000_0000H
0104H	Interrupt raw status register (LCH0 to LCH3)	LCH0LCH3_INT_RAW	R	0000_0000H
0108H	Interrupt enable set register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE	R/W	0000_0000H
010CH	Interrupt enable clear register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE_CL	W	0000_0000H
0110H	Interrupt source clear register (LCH0 to LCH3)	LCH0LCH3_INT_REQ_CL	W	0000_0000H
0114H to 011CH	Reserved	–	–	–
0120H	Interrupt status register (LCH4 to LCH7)	LCH4LCH7_INT_CONT	R	0000_0000H
0124H	Interrupt raw status register (LCH4 to LCH7)	LCH4LCH7_INT_RAW	R	0000_0000H
0128H	Interrupt enable set register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE	R/W	0000_0000H
012CH	Interrupt enable clear register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE_CL	W	0000_0000H
0130H	Interrupt source clear register (LCH4 to LCH7)	LCH4LCH7_INT_REQ_CL	W	0000_0000H
0134H to 01FCH	Reserved	–	–	–

(3) M2P LCH0 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1000H	LCH0 source address register (start address)	LCH0_AADD	R/W	0000_0000H
1004H	LCH0 source address pointer register	LCH0_AADP	R	0000_0000H
1008H	LCH0 source address offset register	LCH0_AOFF	R/W	0000_0000H
100CH	LCH0 source block size register	LCH0_ASIZE	R/W	0000_0000H
1010H	LCH0 source block count register	LCH0_ASIZE_COUNT	R/W	0000_0000H
1014H to 101CH	Reserved	–	–	–
1020H	LCH0 destination address register	LCH0_BADD	R/W	0000_0000H
1024H to 103CH	Reserved	–	–	–
1040H	LCH0 length register	LCH0_LENG	R/W	0000_0000H
1044H	LCH0 read length count register	LCH0_LENG_RCOUNT	R	0000_0000H
1048H	LCH0 write length count register	LCH0_LENG_WCOUNT	R	0000_0000H
104CH	Reserved	–	–	–
1050H	LCH0 mode register	LCH0_MODE	R/W	E4E4_0000H
1054H	LCH0 timer register	LCH0_TIME	R/W	0000_0000H
1058H	LCH0 timer count register	LCH0_TIME_COUNT	R	0000_0000H
105CH	LCH0 physical channel register	LCH0_PCH	R/W	0000_0000H
1060H to 10FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(4) M2P LCH1 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1100H	LCH1 source address register (start address)	LCH1_AADD	R/W	0000_0000H
1104H	LCH1 source address pointer register	LCH1_AADP	R	0000_0000H
1108H	LCH1 source address offset register	LCH1_AOFF	R/W	0000_0000H
110CH	LCH1 source block size register	LCH1_ASIZE	R/W	0000_0000H
1110H	LCH1 source block count register	LCH1_ASIZE_COUNT	R/W	0000_0000H
1114H to 111CH	Reserved	–	–	–
1120H	LCH1 destination address register	LCH1_BADD	R/W	0000_0000H
1124H to 113CH	Reserved	–	–	–
1140H	LCH1 length register	LCH1_LENG	R/W	0000_0000H
1144H	LCH1 read length count register	LCH1_LENG_RCOUNT	R	0000_0000H
1148H	LCH1 write length count register	LCH1_LENG_WCOUNT	R	0000_0000H
114CH	Reserved	–	–	–
1150H	LCH1 mode register	LCH1_MODE	R/W	E4E4_0000H
1154H to 1158H	Reserved	–	–	–
115CH	LCH1 physical channel register	LCH1_PCH	R/W	0000_0001H
1160H to 11FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(5) M2P LCH2 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1200H	LCH2 source address register (start address)	LCH2_AADD	R/W	0000_0000H
1204H	LCH2 source address pointer register	LCH2_AADP	R	0000_0000H
1208H	LCH2 source address offset register	LCH2_AOFF	R/W	0000_0000H
120CH	LCH2 source block size register	LCH2_ASIZE	R/W	0000_0000H
1210H	LCH2 source block count register	LCH2_ASIZE_COUNT	R/W	0000_0000H
1214H to 121CH	Reserved	–	–	–
1220H	LCH2 destination address register	LCH2_BADD	R/W	0000_0000H
1224H to 123CH	Reserved	–	–	–
1240H	LCH2 length register	LCH2_LENG	R/W	0000_0000H
1244H	LCH2 read length count register	LCH2_LENG_RCOUNT	R	0000_0000H
1248H	LCH2 write length count register	LCH2_LENG_WCOUNT	R	0000_0000H
124CH	Reserved	–	–	–
1250H	LCH2 mode register	LCH2_MODE	R/W	E4E4_0000H
1254H to 1258H	Reserved	–	–	–
125CH	LCH2 physical channel register	LCH2_PCH	R/W	0000_0002H
1260H to 12FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(6) M2P LCH3 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1300H	LCH3 source address register (start address)	LCH3_AADD	R/W	0000_0000H
1304H	LCH3 source address pointer register	LCH3_AADP	R	0000_0000H
1308H	LCH3 source address offset register	LCH3_AOFF	R/W	0000_0000H
130CH	LCH3 source block size register	LCH3_ASIZE	R/W	0000_0000H
1310H	LCH3 source block count register	LCH3_ASIZE_COUNT	R/W	0000_0000H
1314H to 131CH	Reserved	–	–	–
1320H	LCH3 destination address register	LCH3_BADD	R/W	0000_0000H
1324H to 133CH	Reserved	–	–	–
1340H	LCH3 length register	LCH3_LENG	R/W	0000_0000H
1344H	LCH3 read length count register	LCH3_LENG_RCOUNT	R	0000_0000H
1348H	LCH3 write length count register	LCH3_LENG_WCOUNT	R	0000_0000H
134CH	Reserved	–	–	–
1350H	LCH3 mode register	LCH3_MODE	R/W	E4E4_0000H
1354H to 1358H	Reserved	–	–	–
135CH	LCH3 physical channel register	LCH3_PCH	R/W	0000_0003H
1360H to 13FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(7) M2P LCH4 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1400H	LCH4 source address register (start address)	LCH4_AADD	R/W	0000_0000H
1404H	LCH4 source address pointer register	LCH4_AADP	R	0000_0000H
1408H	LCH4 source address offset register	LCH4_AOFF	R/W	0000_0000H
140CH	LCH4 source block size register	LCH4_ASIZE	R/W	0000_0000H
1410H	LCH4 source block count register	LCH4_ASIZE_COUNT	R/W	0000_0000H
1414H to 141CH	Reserved	–	–	–
1420H	LCH4 destination address register	LCH4_BADD	R/W	0000_0000H
1424H to 143CH	Reserved	–	–	–
1440H	LCH4 length register	LCH4_LENG	R/W	0000_0000H
1444H	LCH4 read length count register	LCH4_LENG_RCOUNT	R	0000_0000H
1448H	LCH4 write length count register	LCH4_LENG_WCOUNT	R	0000_0000H
144CH	Reserved	–	–	–
1450H	LCH4 mode register	LCH4_MODE	R/W	E4E4_0000H
1454H to 1458H	Reserved	–	–	–
145CH	LCH4 physical channel register	LCH4_PCH	R/W	0000_0004H
1460H to 14FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(8) M2P LCH5 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1500H	LCH5 source address register (start address)	LCH5_AADD	R/W	0000_0000H
1504H	LCH5 source address pointer register	LCH5_AADP	R	0000_0000H
1508H	LCH5 source address offset register	LCH5_AOFF	R/W	0000_0000H
150CH	LCH5 source block size register	LCH5_ASIZE	R/W	0000_0000H
1510H	LCH5 source block count register	LCH5_ASIZE_COUNT	R/W	0000_0000H
1514H to 151CH	Reserved	–	–	–
1520H	LCH5 destination address register	LCH5_BADD	R/W	0000_0000H
1524H to 153CH	Reserved	–	–	–
1540H	LCH5 length register	LCH5_LENG	R/W	0000_0000H
1544H	LCH5 read length count register	LCH5_LENG_RCOUNT	R	0000_0000H
1548H	LCH5 write length count register	LCH5_LENG_WCOUNT	R	0000_0000H
154CH	Reserved	–	–	–
1550H	LCH5 mode register	LCH5_MODE	R/W	E4E4_0000H
1554H to 1558H	Reserved	–	–	–
155CH	LCH5 physical channel register	LCH5_PCH	R/W	0000_0005H
1560H to 15FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(9) M2P LCH6 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1600H	LCH6 source address register (start address)	LCH6_AADD	R/W	0000_0000H
1604H	LCH6 source address pointer register	LCH6_AADP	R	0000_0000H
1608H	LCH6 source address offset register	LCH6_AOFF	R/W	0000_0000H
160CH	LCH6 source block size register	LCH6_ASIZE	R/W	0000_0000H
1610H	LCH6 source block count register	LCH6_ASIZE_COUNT	R/W	0000_0000H
1614H to 161CH	Reserved	–	–	–
1620H	LCH6 destination address register	LCH6_BADD	R/W	0000_0000H
1624H to 163CH	Reserved	–	–	–
1640H	LCH6 length register	LCH6_LENG	R/W	0000_0000H
1644H	LCH6 read length count register	LCH6_LENG_RCOUNT	R	0000_0000H
1648H	LCH6 write length count register	LCH6_LENG_WCOUNT	R	0000_0000H
164CH	Reserved	–	–	–
1650H	LCH6 mode register	LCH6_MODE	R/W	E4E4_0000H
1654H to 1658H	Reserved	–	–	–
165CH	LCH6 physical channel register	LCH6_PCH	R/W	0000_0006H
1660H to 16FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

(10) M2P LCH7 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1700H	LCH7 source address register (start address)	LCH7_AADD	R/W	0000_0000H
1704H	LCH7 source address pointer register	LCH7_AADP	R	0000_0000H
1708H	LCH7 source address offset register	LCH7_AOFF	R/W	0000_0000H
170CH	LCH7 source block size register	LCH7_ASIZE	R/W	0000_0000H
1710H	LCH7 source block count register	LCH7_ASIZE_COUNT	R/W	0000_0000H
1714H to 171CH	Reserved	–	–	–
1720H	LCH7 destination address register	LCH7_BADD	R/W	0000_0000H
1724H to 173CH	Reserved	–	–	–
1740H	LCH7 length register	LCH7_LENG	R/W	0000_0000H
1744H	LCH7 read length count register	LCH7_LENG_RCOUNT	R	0000_0000H
1748H	LCH7 write length count register	LCH7_LENG_WCOUNT	R	0000_0000H
174CH	Reserved	–	–	–
1750H	LCH7 mode register	LCH7_MODE	R/W	E4E4_0000H
1754H to 1758H	Reserved	–	–	–
175CH	LCH7 physical channel register	LCH7_PCH	R/W	0000_0007H
1760H to 17FCH	Reserved	–	–	–

Remark Specify a memory address on the source side and a peripheral address on the destination side.

2.2.3 P2M (peripheral-to-memory) transfer registers

Base address: E108_0000H

(1) P2M DMA control registers

Address	Register Name	Symbol	R/W	After Reset
0000H	DMA start control register	CONT	W	0000_0000H
0004H	DMA control status register	CONTSTATUS	R	0000_0000H
0008H	DMA end control register	END	W	0000_0000H
000CH to 00FCH	Reserved	–	–	–

(2) P2M interrupt parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
0100H	Interrupt status register (LCH0 to LCH3)	LCH0LCH3_INT_CONT	R	0000_0000H
0104H	Interrupt raw status register (LCH0 to LCH3)	LCH0LCH3_INT_RAW	R	0000_0000H
0108H	Interrupt enable set register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE	R/W	0000_0000H
010CH	Interrupt enable clear register (LCH0 to LCH3)	LCH0LCH3_INT_ENABLE_CL	W	0000_0000H
0110H	Interrupt source clear register (LCH0 to LCH3)	LCH0LCH3_INT_REQ_CL	W	0000_0000H
0114H to 011CH	Reserved	–	–	–
0120H	Interrupt status register (LCH4 to LCH7)	LCH4LCH7_INT_CONT	R	0000_0000H
0124H	Interrupt raw status register (LCH4 to LCH7)	LCH4LCH7_INT_RAW	R	0000_0000H
0128H	Interrupt enable set register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE	R/W	0000_0000H
012CH	Interrupt enable clear register (LCH4 to LCH7)	LCH4LCH7_INT_ENABLE_CL	W	0000_0000H
0130H	Interrupt source clear register (LCH4 to LCH7)	LCH4LCH7_INT_REQ_CL	W	0000_0000H
0134H to 01FCH	Reserved	–	–	–

(3) P2M LCH0 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1000H	LCH0 source address register	LCH0_AADD	R/W	0000_0000H
1004H to 101CH	Reserved	–	–	–
1020H	LCH0 destination address register (start address)	LCH0_BADD	R/W	0000_0000H
1024H	LCH0 destination address pointer register	LCH0_BADP	R	0000_0000H
1028H	LCH0 destination address offset register	LCH0_BOFF	R/W	0000_0000H
102CH	LCH0 destination block size register	LCH0_BSIZE	R/W	0000_0000H
1030H	LCH0 destination block count register	LCH0_BSIZE_COUNT	R/W	0000_0000H
1034H to 103CH	Reserved	–	–	–
1040H	LCH0 length register	LCH0_LENG	R/W	0000_0000H
1044H	LCH0 read length count register	LCH0_LENG_RCOUNT	R	0000_0000H
1048H	LCH0 write length count register	LCH0_LENG_WCOUNT	R	0000_0000H
104CH	Reserved	–	–	–
1050H	LCH0 mode register	LCH0_MODE	R/W	E4E4_0000H
1054H	LCH0 timer register	LCH0_TIME	R/W	0000_0000H
1058H	LCH0 timer count register	LCH0_TIME_COUNT	R	0000_0000H
105CH	LCH0 physical channel register	LCH0_PCH	R/W	0000_0000H
1060H to 10FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(4) P2M LCH1 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1100H	LCH1 source address register	LCH1_AADD	R/W	0000_0000H
1104H to 111CH	Reserved	–	–	–
1120H	LCH1 destination address register (start address)	LCH1_BADD	R/W	0000_0000H
1124H	LCH1 destination address pointer register	LCH1_BADP	R	0000_0000H
1128H	LCH1 destination address offset register	LCH1_BOFF	R/W	0000_0000H
112CH	LCH1 destination block size register	LCH1_BSIZE	R/W	0000_0000H
1130H	LCH1 destination block count register	LCH1_BSIZE_COUNT	R/W	0000_0000H
1134H to 113CH	Reserved	–	–	–
1140H	LCH1 length register	LCH1_LENG	R/W	0000_0000H
1144H	LCH1 read length count register	LCH1_LENG_RCOUNT	R	0000_0000H
1148H	LCH1 write length count register	LCH1_LENG_WCOUNT	R	0000_0000H
114CH	Reserved	–	–	–
1150H	LCH1 mode register	LCH1_MODE	R/W	E4E4_0000H
1154H to 1158H	Reserved	–	–	–
115CH	LCH1 physical channel register	LCH1_PCH	R/W	0000_0001H
1160H to 11FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(5) P2M LCH2 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1200H	LCH2 source address register	LCH2_AADD	R/W	0000_0000H
1204H to 121CH	Reserved	–	–	–
1220H	LCH2 destination address register (start address)	LCH2_BADD	R/W	0000_0000H
1224H	LCH2 destination address pointer register	LCH2_BADP	R	0000_0000H
1228H	LCH2 destination address offset register	LCH2_BOFF	R/W	0000_0000H
122CH	LCH2 destination block size register	LCH2_BSIZE	R/W	0000_0000H
1230H	LCH2 destination block count register	LCH2_BSIZE_COUNT	R/W	0000_0000H
1234H to 123CH	Reserved	–	–	–
1240H	LCH2 length register	LCH2_LENG	R/W	0000_0000H
1244H	LCH2 read length count register	LCH2_LENG_RCOUNT	R	0000_0000H
1248H	LCH2 write length count register	LCH2_LENG_WCOUNT	R	0000_0000H
124CH	Reserved	–	–	–
1250H	LCH2 mode register	LCH2_MODE	R/W	E4E4_0000H
1254H to 1258H	Reserved	–	–	–
125CH	LCH2 physical channel register	LCH2_PCH	R/W	0000_0002H
1260H to 12FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(6) P2M LCH3 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1300H	LCH3 source address register	LCH3_AADD	R/W	0000_0000H
1304H to 131CH	Reserved	–	–	–
1320H	LCH3 destination address register (start address)	LCH3_BADD	R/W	0000_0000H
1324H	LCH3 destination address pointer register	LCH3_BADP	R	0000_0000H
1328H	LCH3 destination address offset register	LCH3_BOFF	R/W	0000_0000H
132CH	LCH3 destination block size register	LCH3_BSIZE	R/W	0000_0000H
1330H	LCH3 destination block count register	LCH3_BSIZE_COUNT	R/W	0000_0000H
1334H to 133CH	Reserved	–	–	–
1340H	LCH3 length register	LCH3_LENG	R/W	0000_0000H
1344H	LCH3 read length count register	LCH3_LENG_RCOUNT	R	0000_0000H
1348H	LCH3 write length count register	LCH3_LENG_WCOUNT	R	0000_0000H
134CH	Reserved	–	–	–
1350H	LCH3 mode register	LCH3_MODE	R/W	E4E4_0000H
1354H to 1358H	Reserved	–	–	–
135CH	LCH3 physical channel register	LCH3_PCH	R/W	0000_0003H
1360H to 13FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(7) P2M LCH4 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1400H	LCH4 source address register	LCH4_AADD	R/W	0000_0000H
1404H to 141CH	Reserved	–	–	–
1420H	LCH4 destination address register (start address)	LCH4_BADD	R/W	0000_0000H
1424H	LCH4 destination address pointer register	LCH4_BADP	R	0000_0000H
1428H	LCH4 destination address offset register	LCH4_BOFF	R/W	0000_0000H
142CH	LCH4 destination block size register	LCH4_BSIZE	R/W	0000_0000H
1430H	LCH4 destination block count register	LCH4_BSIZE_COUNT	R/W	0000_0000H
1434H to 143CH	Reserved	–	–	–
1440H	LCH4 length register	LCH4_LENG	R/W	0000_0000H
1444H	LCH4 read length count register	LCH4_LENG_RCOUNT	R	0000_0000H
1448H	LCH4 write length count register	LCH4_LENG_WCOUNT	R	0000_0000H
144CH	Reserved	–	–	–
1450H	LCH4 mode register	LCH4_MODE	R/W	E4E4_0000H
1454H to 1458H	Reserved	–	–	–
145CH	LCH4 physical channel register	LCH4_PCH	R/W	0000_0004H
1460H to 14FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(8) P2M LCH5 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1500H	LCH5 source address register	LCH5_AADD	R/W	0000_0000H
1504H to 151CH	Reserved	–	–	–
1520H	LCH5 destination address register (start address)	LCH5_BADD	R/W	0000_0000H
1524H	LCH5 destination address pointer register	LCH5_BADP	R	0000_0000H
1528H	LCH5 destination address offset register	LCH5_BOFF	R/W	0000_0000H
152CH	LCH5 destination block size register	LCH5_BSIZE	R/W	0000_0000H
1530H	LCH5 destination block count register	LCH5_BSIZE_COUNT	R/W	0000_0000H
1534H to 153CH	Reserved	–	–	–
1540H	LCH5 length register	LCH5_LENG	R/W	0000_0000H
1544H	LCH5 read length count register	LCH5_LENG_RCOUNT	R	0000_0000H
1548H	LCH5 write length count register	LCH5_LENG_WCOUNT	R	0000_0000H
154CH	Reserved	–	–	–
1550H	LCH5 mode register	LCH5_MODE	R/W	E4E4_0000H
1554H to 1558H	Reserved	–	–	–
155CH	LCH5 physical channel register	LCH5_PCH	R/W	0000_0005H
1560H to 15FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(9) P2M LCH6 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1600H	LCH6 source address register	LCH6_AADD	R/W	0000_0000H
1604H to 161CH	Reserved	–	–	–
1620H	LCH6 destination address register (start address)	LCH6_BADD	R/W	0000_0000H
1624H	LCH6 destination address pointer register	LCH6_BADP	R	0000_0000H
1628H	LCH6 destination address offset register	LCH6_BOFF	R/W	0000_0000H
162CH	LCH6 destination block size register	LCH6_BSIZE	R/W	0000_0000H
1630H	LCH6 destination block count register	LCH6_BSIZE_COUNT	R/W	0000_0000H
1634H to 163CH	Reserved	–	–	–
1640H	LCH6 length register	LCH6_LENG	R/W	0000_0000H
1644H	LCH6 read length count register	LCH6_LENG_RCOUNT	R	0000_0000H
1648H	LCH6 write length count register	LCH6_LENG_WCOUNT	R	0000_0000H
164CH	Reserved	–	–	–
1650H	LCH6 mode register	LCH6_MODE	R/W	E4E4_0000H
1654H to 1658H	Reserved	–	–	–
165CH	LCH6 physical channel register	LCH6_PCH	R/W	0000_0006H
1660H to 16FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

(10) P2M LCH7 parameter setting registers

Address	Register Name	Symbol	R/W	After Reset
1700H	LCH7 source address register	LCH7_AADD	R/W	0000_0000H
1704H to 171CH	Reserved	–	–	–
1720H	LCH7 destination address register (start address)	LCH7_BADD	R/W	0000_0000H
1724H	LCH7 destination address pointer register	LCH7_BADP	R	0000_0000H
1728H	LCH7 destination address offset register	LCH7_BOFF	R/W	0000_0000H
172CH	LCH7 destination block size register	LCH7_BSIZE	R/W	0000_0000H
1730H	LCH7 destination block count register	LCH7_BSIZE_COUNT	R/W	0000_0000H
1734H to 173CH	Reserved	–	–	–
1740H	LCH7 length register	LCH7_LENG	R/W	0000_0000H
1744H	LCH7 read length count register	LCH7_LENG_RCOUNT	R	0000_0000H
1748H	LCH7 write length count register	LCH7_LENG_WCOUNT	R	0000_0000H
174CH	Reserved	–	–	–
1750H	LCH7 mode register	LCH7_MODE	R/W	E4E4_0000H
1754H to 1758H	Reserved	–	–	–
175CH	LCH7 physical channel register	LCH7_PCH	R/W	0000_0007H
1760H to 17FCH	Reserved	–	–	–

Remark Specify a peripheral address on the source side and a memory address on the destination side.

2.3 Register Details

2.3.1 M2M DMA control/status registers

(1) M2M DMA start control register

This register (CONT: E109_0000H) is used to start a DMA transfer on each logical channel.

If this register is set up while the ARM_LCHx_RESERVE bit of the M2M DMA control status register is 0, the subsequent transfer starts immediately after the current transfer ends (simple reservation). A transfer parameter for the subsequent transfer must be specified before using simple reservation. For details, see **3.4.3 Continuous transfer**.

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
ARM_LCH7_CONT	ARM_LCH6_CONT	ARM_LCH5_CONT	ARM_LCH4_CONT	ARM_LCH3_CONT	ARM_LCH2_CONT	ARM_LCH1_CONT	ARM_LCH0_CONT

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH7_CONT	W	7	0	Set this bit to start a DMA transfer on LCH7. (1: Start a DMA transfer.)
ARM_LCH6_CONT	W	6	0	Set this bit to start a DMA transfer on LCH6. (1: Start a DMA transfer.)
ARM_LCH5_CONT	W	5	0	Set this bit to start a DMA transfer on LCH5. (1: Start a DMA transfer.)
ARM_LCH4_CONT	W	4	0	Set this bit to start a DMA transfer on LCH4. (1: Start a DMA transfer.)
ARM_LCH3_CONT	W	3	0	Set this bit to start a DMA transfer on LCH3. (1: Start a DMA transfer.)
ARM_LCH2_CONT	W	2	0	Set this bit to start a DMA transfer on LCH2. (1: Start a DMA transfer.)
ARM_LCH1_CONT	W	1	0	Set this bit to start a DMA transfer on LCH1. (1: Start a DMA transfer.)
ARM_LCH0_CONT	W	0	0	Set this bit to start a DMA transfer on LCH0. (1: Start a DMA transfer.)

(2) M2M DMA control status register

This register (CONTSTATUS: E109_0004H) indicates the DMA status.

This register also indicates whether a DMA transfer is reserved on a logical channel. If a bit is set to 1, the corresponding LCH has been reserved for the next transfer, so another transfer cannot be reserved.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
ARM_LCH7_RESERVE	ARM_LCH6_RESERVE	ARM_LCH5_RESERVE	ARM_LCH4_RESERVE	ARM_LCH3_RESERVE	ARM_LCH2_RESERVE	ARM_LCH1_RESERVE	ARM_LCH0_RESERVE

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
ARM_LCH7_CONTSTATUS	ARM_LCH6_CONTSTATUS	ARM_LCH5_CONTSTATUS	ARM_LCH4_CONTSTATUS	ARM_LCH3_CONTSTATUS	ARM_LCH2_CONTSTATUS	ARM_LCH1_CONTSTATUS	ARM_LCH0_CONTSTATUS

(1/2)

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH7_RESERVE	R	23	0	Indicates whether a DMA transfer is reserved on LCH7. 0: Not reserved 1: Reserved
ARM_LCH6_RESERVE	R	22	0	Indicates whether a DMA transfer is reserved on LCH6. 0: Not reserved 1: Reserved
ARM_LCH5_RESERVE	R	21	0	Indicates whether a DMA transfer is reserved on LCH5. 0: Not reserved 1: Reserved
ARM_LCH4_RESERVE	R	20	0	Indicates whether a DMA transfer is reserved on LCH4. 0: Not reserved 1: Reserved
ARM_LCH3_RESERVE	R	19	0	Indicates whether a DMA transfer is reserved on LCH3. 0: Not reserved 1: Reserved
ARM_LCH2_RESERVE	R	18	0	Indicates whether a DMA transfer is reserved on LCH2. 0: Not reserved 1: Reserved
ARM_LCH1_RESERVE	R	17	0	Indicates whether a DMA transfer is reserved on LCH1. 0: Not reserved 1: Reserved

(2/2)

Name	R/W	Bit No.	After Reset	Description
ARM_LCH0_RESERVE	R	16	0	Indicates whether a DMA transfer is reserved on LCH0. 0: Not reserved 1: Reserved
Reserved	R	15:8	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH7_CONTSTATUS	R	7	0	Indicates the DMA status on LCH7. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH6_CONTSTATUS	R	6	0	Indicates the DMA status on LCH6. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH5_CONTSTATUS	R	5	0	Indicates the DMA status on LCH5. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH4_CONTSTATUS	R	4	0	Indicates the DMA status on LCH4. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH3_CONTSTATUS	R	3	0	Indicates the DMA status on LCH3. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH2_CONTSTATUS	R	2	0	Indicates the DMA status on LCH2. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH1_CONTSTATUS	R	1	0	Indicates the DMA status on LCH1. 0: No DMA is being performed 1: DMA is being performed
ARM_LCH0_CONTSTATUS	R	0	0	Indicates the DMA status on LCH0. 0: No DMA is being performed 1: DMA is being performed

(3) M2M DMA end control register

This register (END: E109_0008H) is used to terminate DMA transfers.

If a DMA transfer is terminated, the transfer reserved on the channel becomes invalid. If a bit of this register is set, the DMA transfer on the corresponding channel is terminated when the current AHB transaction is completed. Therefore, the DMA transfer cannot be restarted until the AHB transaction ends. To restart the DMA transfer by setting the DMA transfer start bit, be sure to read the M2M DMA control status register to make sure the corresponding DMA status bit has been cleared to 0.

Once a DMA transfer starts, at least one write transfer must be performed before terminating the transfer. Otherwise, parameters are not updated in the internal circuits so the transfer is not executed correctly.

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
ARM_LCH7_END	ARM_LCH6_END	ARM_LCH5_END	ARM_LCH4_END	ARM_LCH3_END	ARM_LCH2_END	ARM_LCH1_END	ARM_LCH0_END

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH7_END	W	7	0	Set this bit to terminate a DMA transfer on LCH7.
ARM_LCH6_END	W	6	0	Set this bit to terminate a DMA transfer on LCH6.
ARM_LCH5_END	W	5	0	Set this bit to terminate a DMA transfer on LCH5.
ARM_LCH4_END	W	4	0	Set this bit to terminate a DMA transfer on LCH4.
ARM_LCH3_END	W	3	0	Set this bit to terminate a DMA transfer on LCH3.
ARM_LCH2_END	W	2	0	Set this bit to terminate a DMA transfer on LCH2.
ARM_LCH1_END	W	1	0	Set this bit to terminate a DMA transfer on LCH1.
ARM_LCH0_END	W	0	0	Set this bit to terminate a DMA transfer on LCH0.

Remark 0: The current status is retained, 1: Terminate the DMA transfer

2.3.2 M2M interrupt parameter setting registers

These registers are used to set up the parameters for three types of interrupts - length transfer end, block transfer end, and error end.

(1) M2M interrupt status registers

These registers (LCH*x*LCH*y*_INT_CONT) indicate the interrupt source statuses.
Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

LCH0LCH3_INT_CONT: E109_0100H (LCH0 to LCH3)
LCH4LCH7_INT_CONT: E109_0120H (LCH4 to LCH7)
Paraphrase as the next by the following detail description.
LCH0→LCH4, LCH1→LCH5, LCH2→LCH6, LCH3→LCH7

31	30	29	28	27	26	25	24
Reserved	ARM_LCH3_ INT_ERROR _W_CONT	ARM_LCH3_ INT_BLOCK _W_CONT	ARM_LCH3_ INT_LENG _W_CONT	Reserved	ARM_LCH3_ INT_ERROR _R_CONT	Reserved	

23	22	21	20	19	18	17	16
Reserved	ARM_LCH2_ INT_ERROR _W_CONT	ARM_LCH2_ INT_BLOCK _W_CONT	ARM_LCH2_ INT_LENG _W_CONT	Reserved	ARM_LCH2_ INT_ERROR _R_CONT	Reserved	

15	14	13	12	11	10	9	8
Reserved	ARM_LCH1_ INT_ERROR _W_CONT	ARM_LCH1_ INT_BLOCK _W_CONT	ARM_LCH1_ INT_LENG_ W_CONT	Reserved	ARM_LCH1_ INT_ERROR _R_CONT	Reserved	

7	6	5	4	3	2	1	0
Reserved	ARM_LCH0_ INT_ERROR _W_CONT	ARM_LCH0_ INT_BLOCK _W_CONT	ARM_LCH0_ INT_LENG_ W_CONT	Reserved	ARM_LCH0_ INT_ERROR_ R_CONT	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_W_CONT	R	30	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH3.
ARM_LCH3_INT_ BLOCK_W_CONT	R	29	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH3 finishes.
ARM_LCH3_INT_ LENG_W_CONT	R	28	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_R_CONT	R	26	0	Indicates the status of the interrupt caused by caused by an M2M transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH2_INT_ ERROR_W_CONT	R	22	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH2.
ARM_LCH2_INT_ BLOCK_W_CONT	R	21	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH2 finishes.
ARM_LCH2_INT_ LENG_W_CONT	R	20	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH2_INT_ ERROR_R_CONT	R	18	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH1_INT_ ERROR_W_CONT	R	14	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH1.
ARM_LCH1_INT_ BLOCK_W_CONT	R	13	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH1 finishes.
ARM_LCH1_INT_ LENG_W_CONT	R	12	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH1 finishes.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH1_INT_ ERROR_R_CONT	R	10	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH1.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH0_INT_ ERROR_W_CONT	R	6	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH0.
ARM_LCH0_INT_ BLOCK_W_CONT	R	5	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH0 finishes.
ARM_LCH0_INT_ LENG_W_CONT	R	4	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH0 finishes.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH0_INT_ ERROR_R_CONT	R	2	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(2) M2M interrupt raw status registers

These registers (LCH*x*LCH*y*_INT_RAW) can be used to read the status of the interrupt sources regardless of the settings of the interrupt enable set register and the interrupt enable clear register.
Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_RAW: E109_0104H (LCH0 to LCH3)
- LCH4LCH7_INT_RAW: E109_0124H (LCH4 to LCH7)

  Paraphrase as the next by the following detail description.
  LCH0→LCH4, LCH1→LCH5, LCH2→LCH6, LCH3→LCH7

31	30	29	28	27	26	25	24
Reserved	ARM_LCH3_INT_ERROR_W_RAW	ARM_LCH3_INT_BLOCK_W_RAW	ARM_LCH3_INT_LENG_W_RAW	Reserved	ARM_LCH3_INT_ERROR_R_RAW	Reserved	

23	22	21	20	19	18	17	16
Reserved	ARM_LCH2_INT_ERROR_W_RAW	ARM_LCH2_INT_BLOCK_W_RAW	ARM_LCH2_INT_LENG_W_RAW	Reserved	ARM_LCH2_INT_ERROR_R_RAW	Reserved	

15	14	13	12	11	10	9	8
Reserved	ARM_LCH1_INT_ERROR_W_RAW	ARM_LCH1_INT_BLOCK_W_RAW	ARM_LCH1_INT_LENG_W_RAW	Reserved	ARM_LCH1_INT_ERROR_R_RAW	Reserved	

7	6	5	4	3	2	1	0
Reserved	ARM_LCH0_INT_ERROR_W_RAW	ARM_LCH0_INT_BLOCK_W_RAW	ARM_LCH0_INT_LENG_W_RAW	Reserved	ARM_LCH0_INT_ERROR_R_RAW	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_W_RAW	R	30	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH3.
ARM_LCH3_INT_ BLOCK_W_RAW	R	29	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH3 finishes.
ARM_LCH3_INT_ LENG_W_RAW	R	28	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_R_RAW	R	26	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH2_INT_ ERROR_W_RAW	R	22	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH2.
ARM_LCH2_INT_ BLOCK_W_RAW	R	21	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH2 finishes.
ARM_LCH2_INT_ LENG_W_RAW	R	20	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH2_INT_ ERROR_R_RAW	R	18	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH1_INT_ ERROR_W_RAW	R	14	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH1.
ARM_LCH1_INT_ BLOCK_W_RAW	R	13	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH1 finishes.
ARM_LCH1_INT_ LENG_W_RAW	R	12	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH1 finishes.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH1_INT_ ERROR_R_RAW	R	10	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH1.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH0_INT_ ERROR_W_RAW	R	6	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH0.
ARM_LCH0_INT_ BLOCK_W_RAW	R	5	0	Indicates the status of the interrupt that occurs when an M2M block transfer on LCH0 finishes.
ARM_LCH0_INT_ LENG_W_RAW	R	4	0	Indicates the status of the interrupt that occurs when an M2M length transfer on LCH0 finishes.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH0_INT_ ERROR_R_RAW	R	2	0	Indicates the status of the interrupt caused by an M2M transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(3) M2M interrupt enable set registers

These registers (LCH*x*LCH*y*_INT_ENABLE) are used to enable interrupt sources. Only data of bits to which 1 is written is updated. Masking of interrupt sources corresponding to bits to which 1 is written is cancelled.
Read these registers to check whether an interrupt is enabled. Writing 0 to these registers is ignored.
To disable an interrupt source, set the corresponding bit of the interrupt enable clear register to 1.

- LCH0LCH3_INT_ENABLE: E109_0108H (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE: E109_0128H (LCH4 to LCH7)

  Paraphrase as the next by the following detail description.
  LCH0→LCH4, LCH1→LCH5, LCH2→LCH6, LCH3→LCH7

31	30	29	28	27	26	25	24
Reserved					ARM_LCH3_INT_ERROR_ENABLE	ARM_LCH3_INT_BLOCK_ENABLE	ARM_LCH3_INT_LENG_ENABLE

23	22	21	20	19	18	17	16
Reserved					ARM_LCH2_INT_ERROR_ENABLE	ARM_LCH2_INT_BLOCK_ENABLE	ARM_LCH2_INT_LENG_ENABLE

15	14	13	12	11	10	9	8
Reserved					ARM_LCH1_INT_ERROR_ENABLE	ARM_LCH1_INT_BLOCK_ENABLE	ARM_LCH1_INT_LENG_ENABLE

7	6	5	4	3	2	1	0
Reserved					ARM_LCH0_INT_ERROR_ENABLE	ARM_LCH0_INT_BLOCK_ENABLE	ARM_LCH0_INT_LENG_ENABLE

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH3_INT_ ERROR_ENABLE	R/W	26	0	Enable the LCH3 error interrupt source.
ARM_LCH3_INT_ BLOCK_ENABLE	R/W	25	0	Enable the LCH3 block interrupt source.
ARM_LCH3_INT_ LENG_ENABLE	R/W	24	0	Enable the LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH2_INT_ ERROR_ENABLE	R/W	18	0	Enable the LCH2 error interrupt source.
ARM_LCH2_INT_ BLOCK_ENABLE	R/W	17	0	Enable the LCH2 block interrupt source.
ARM_LCH2_INT_ LENG_ENABLE	R/W	16	0	Enable the LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH1_INT_ ERROR_ENABLE	R/W	10	0	Enable the LCH1 error interrupt source.
ARM_LCH1_INT_ BLOCK_ENABLE	R/W	9	0	Enable the LCH1 block interrupt source.
ARM_LCH1_INT_ LENG_ENABLE	R/W	8	0	Enable the LCH1 length interrupt source.
Reserved	–	7:3	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH0_INT_ ERROR_ENABLE	R/W	2	0	Enable the LCH0 error interrupt source.
ARM_LCH0_INT_ BLOCK_ENABLE	R/W	1	0	Enable the LCH0 block interrupt source.
ARM_LCH0_INT_ LENG_ENABLE	R/W	0	0	Enable the LCH0 length interrupt source.

Remark 0: Disable the interrupt source (default), 1: Enable the interrupt source

(4) M2M interrupt enable clear registers

These registers (LCH*x*LCH*y*_INT_ENABLE_CL) are used to disable interrupt sources.

If a bit of these registers is set to 1, the corresponding interrupt source is disabled. The bits to which 0 is written retain the current settings.

If interrupt sources are disabled in these registers, the corresponding bits in the M2M interrupt enable set register is set to 0 (disables the interrupt sources).

- LCH0LCH3_INT_ENABLE_CL: E109_010CH (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE_CL: E109_012CH (LCH4 to LCH7)

  Paraphrase as the next by the following detail description.
  LCH0→LCH4, LCH1→LCH5, LCH2→LCH6, LCH3→LCH7

31	30	29	28	27	26	25	24
Reserved					ARM_LCH3_INT_ERROR_ENABLE_CL	ARM_LCH3_INT_BLOCK_ENABLE_CL	ARM_LCH3_INT_LENG_ENABLE_CL

23	22	21	20	19	18	17	16
Reserved					ARM_LCH2_INT_ERROR_ENABLE_CL	ARM_LCH2_INT_BLOCK_ENABLE_CL	ARM_LCH2_INT_LENG_ENABLE_CL

15	14	13	12	11	10	9	8
Reserved					ARM_LCH1_INT_ERROR_ENABLE_CL	ARM_LCH1_INT_BLOCK_ENABLE_CL	ARM_LCH1_INT_LENG_ENABLE_CL

7	6	5	4	3	2	1	0
Reserved					ARM_LCH0_INT_ERROR_ENABLE_CL	ARM_LCH0_INT_BLOCK_ENABLE_CL	ARM_LCH0_INT_LENG_ENABLE_CL

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH3_INT_ ERROR_ENABLE_CL	W	26	0	Disable the LCH3 error interrupt source.
ARM_LCH3_INT_ BLOCK_ENABLE_CL	W	25	0	Disable the LCH3 block interrupt source.
ARM_LCH3_INT_ LENG_ENABLE_CL	W	24	0	Disable the LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH2_INT_ ERROR_ENABLE_CL	W	18	0	Disable the LCH2 error interrupt source.
ARM_LCH2_INT_ BLOCK_ENABLE_CL	W	17	0	Disable the LCH2 block interrupt source.
ARM_LCH2_INT_ LENG_ENABLE_CL	W	16	0	Disable the LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH1_INT_ ERROR_ENABLE_CL	W	10	0	Disable the LCH1 error interrupt source.
ARM_LCH1_INT_ BLOCK_ENABLE_CL	W	9	0	Disable the LCH1 block interrupt source.
ARM_LCH1_INT_ LENG_ENABLE_CL	W	8	0	Disable the LCH1 length interrupt source.
Reserved	–	7:3	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH0_INT_ ERROR_ENABLE_CL	W	2	0	Disable the LCH0 error interrupt source.
ARM_LCH0_INT_ BLOCK_ENABLE_CL	W	1	0	Disable the LCH0 block interrupt source.
ARM_LCH0_INT_ LENG_ENABLE_CL	W	0	0	Disable the LCH0 length interrupt source.

Remark 0: Enable the interrupt source (default), 1: Disable the interrupt source

(5) M2M interrupt source clear registers

These registers (LCH*x*LCH*y*_INT_REQ_CL) are used to clear the interrupt sources. Only data of bits to which 1 is written is updated.

Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_REQ_CL: E109_0110H (LCH0 to LCH3)
- LCH4LCH7_INT_REQ_CL: E109_0130H (LCH4 to LCH7)

  Paraphrase as the next by the following detail description.
  LCH0→LCH4, LCH1→LCH5, LCH2→LCH6, LCH3→LCH7

31	30	29	28	27	26	25	24
Reserved	ARM_LCH3_INT_ERROR_W_REQ_CL	ARM_LCH3_INT_BLOCK_W_REQ_CL	ARM_LCH3_INT_LENG_W_REQ_CL	Reserved	ARM_LCH3_INT_ERROR_R_REQ_CL	Reserved	

23	22	21	20	19	18	17	16
Reserved	ARM_LCH2_INT_ERROR_W_REQ_CL	ARM_LCH2_INT_BLOCK_W_REQ_CL	ARM_LCH2_INT_LENG_W_REQ_CL	Reserved	ARM_LCH2_INT_ERROR_R_REQ_CL	Reserved	

15	14	13	12	11	10	9	8
Reserved	ARM_LCH1_INT_ERROR_W_REQ_CL	ARM_LCH1_INT_BLOCK_W_REQ_CL	ARM_LCH1_INT_LENG_W_REQ_CL	Reserved	ARM_LCH1_INT_ERROR_R_REQ_CL	Reserved	

7	6	5	4	3	2	1	0
Reserved	ARM_LCH0_INT_ERROR_W_REQ_CL	ARM_LCH0 INT_BLOCK_W_REQ_CL	ARM_LCH0 INT_LENG_W_REQ_CL	Reserved	ARM_LCH0_INT_ERROR_R_REQ_CL	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_W_REQ_CL	W	30	0	Clear the LCH3 error interrupt source.
ARM_LCH3_INT_ BLOCK_W_REQ_CL	W	29	0	Clear the LCH3 block interrupt source.
ARM_LCH3_INT_ LENG_W_REQ_CL	W	28	0	Clear the LCH3 length interrupt source.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH3_INT_ ERROR_R_REQ_CL	W	26	0	Clear the LCH3 error interrupt source.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH2_INT_ ERROR_W_REQ_CL	W	22	0	Clear the LCH2 error interrupt source.
ARM_LCH2_INT_ BLOCK_W_REQ_CL	W	21	0	Clear the LCH2 block interrupt source.
ARM_LCH2_INT_ LENG_W_REQ_CL	W	20	0	Clear the LCH2 length interrupt source.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH2_INT_ ERROR_R_REQ_CL	W	18	0	Clear the LCH2 error interrupt source.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH1_INT_ ERROR_W_REQ_CL	W	14	0	Clear the LCH1 error interrupt source.
ARM_LCH1_INT_ BLOCK_W_REQ_CL	W	13	0	Clear the LCH1 block interrupt source.
ARM_LCH1_INT_ LENG_W_REQ_CL	W	12	0	Clear the LCH1 length interrupt source.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH1_INT_ ERROR_R_REQ_CL	W	10	0	Clear the LCH1 error interrupt source.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH0_INT_ ERROR_W_REQ_CL	W	6	0	Clear the LCH0 error interrupt source.
ARM_LCH0_INT_ BLOCK_W_REQ_CL	W	5	0	Clear the LCH0 block interrupt source.
ARM_LCH0_INT_ LENG_W_REQ_CL	W	4	0	Clear the LCH0 length interrupt source.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
ARM_LCH0_INT_ ERROR_R_REQ_CL	W	2	0	Clear the LCH0 error interrupt source.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No operation (retains the current setting), 1: Clear the interrupt source

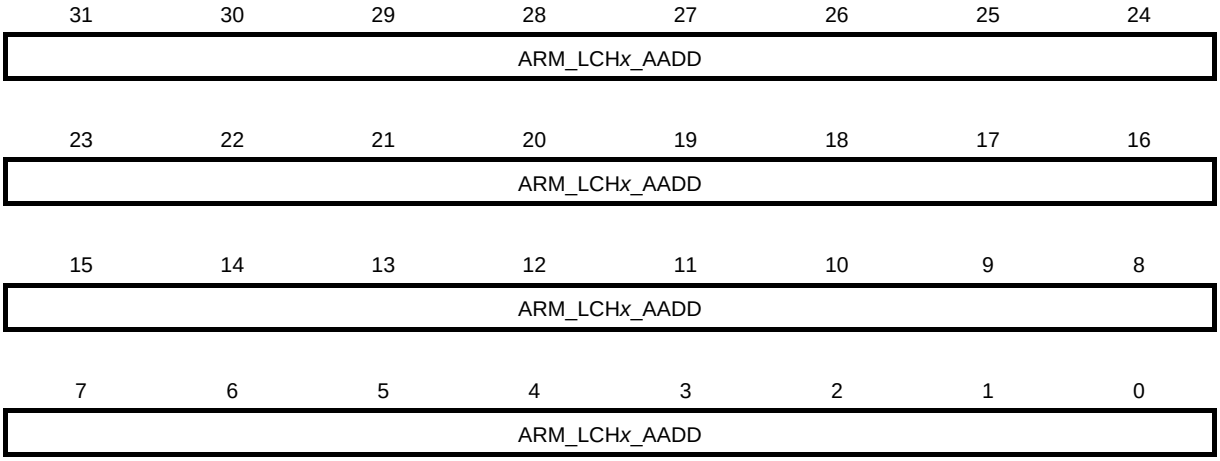
2.3.3 M2M LCH*x* parameter setting registers

These registers are used to specify the settings for each M2M logical channel. The letter *x* in LCH*x* represents a channel number, which ranges from 0 to 7.

(1) M2M LCH*x* source address registers

These registers (LCH*x*_AADD) are used to specify the address from which to send data in bytes.

- LCH0_AADD: E109_1000H (LCH0)
- LCH1_AADD: E109_1100H (LCH1)
- LCH2_AADD: E109_1200H (LCH2)
- LCH3_AADD: E109_1300H (LCH3)
- LCH4_AADD: E109_1400H (LCH4)
- LCH5_AADD: E109_1500H (LCH5)
- LCH6_AADD: E109_1600H (LCH6)
- LCH7_AADD: E109_1700H (LCH7)



Name	R/W	Bit No.	After Reset	Description
ARM_LCH*x*_AADD	R/W	31:0	0000_0000H	Specify the address from which to send data.

(2) M2M LCH*x* source address pointer registers

These registers (LCH*x*_AADP) indicate the address from which data is being sent.

- LCH0_AADP: E109_1004H (LCH0)
- LCH1_AADP: E109_1104H (LCH1)
- LCH2_AADP: E109_1204H (LCH2)
- LCH3_AADP: E109_1304H (LCH3)
- LCH4_AADP: E109_1404H (LCH4)
- LCH5_AADP: E109_1504H (LCH5)
- LCH6_AADP: E109_1604H (LCH6)
- LCH7_AADP: E109_1704H (LCH7)

31	30	29	28	27	26	25	24
ARM_LCH*x*_AADP							

23	22	21	20	19	18	17	16
ARM_LCH*x*_AADP							

15	14	13	12	11	10	9	8
ARM_LCH*x*_AADP							

7	6	5	4	3	2	1	0
ARM_LCH*x*_AADP							

Name	R/W	Bit No.	After Reset	Description
ARM_LCH*x*_AADP	R	31:0	0000_0000H	Indicates the address from which data is being sent on LCH*x*.

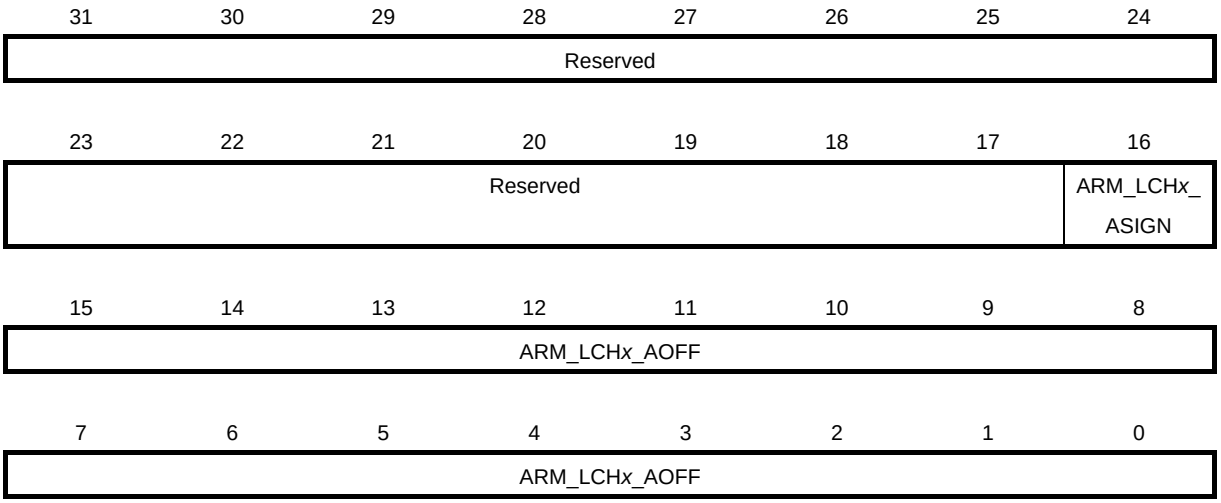
(3) M2M LCH*x* source address offset registers

These registers (LCH*x*_AOFF) are used to specify the offset between data blocks to send in bytes.

Bit 16 is a sign bit. Clearing this bit to 0 adds the specified value and setting this bit to 1 subtracts the specified value.

Up to 65,535 bytes can be specified.

- LCH0_AOFF: E109_1008H (LCH0)
- LCH1_AOFF: E109_1108H (LCH1)
- LCH2_AOFF: E109_1208H (LCH2)
- LCH3_AOFF: E109_1308H (LCH3)
- LCH4_AOFF: E109_1408H (LCH4)
- LCH5_AOFF: E109_1508H (LCH5)
- LCH6_AOFF: E109_1608H (LCH6)
- LCH7_AOFF: E109_1708H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:17	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_ASIGN	R/W	16	0	Specify the offset added to or subtracted from the start address on the source side. 0: Add the offset specified for ARM_LCH*x*_AOFF to the start address. 1: Subtract the offset specified for ARM_LCH*x*_AOFF from the start address.
ARM_LCH*x*_AOFF	R/W	15:0	0000H	Indicates the offset (absolute value) between data blocks to send on LCH*x* in bytes. 00000000_00000000: 0 bytes (no offset) 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

(4) M2M LCH*x* source block count registers

These registers (LCH*x*_ASIZE_COUNT) work differently when read than when written.

When these registers are written, the number of data blocks transferred per loop during a repeat transfer is set.

When these registers are read, the remaining number of data blocks to send is read. The number is decremented each time a block transfer ends, and the remaining number is shown in the corresponding register. To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_ASIZE_COUNT: E109_1010H (LCH0)
- LCH1_ASIZE_COUNT: E109_1110H (LCH1)
- LCH2_ASIZE_COUNT: E109_1210H (LCH2)
- LCH3_ASIZE_COUNT: E109_1310H (LCH3)
- LCH4_ASIZE_COUNT: E109_1410H (LCH4)
- LCH5_ASIZE_COUNT: E109_1510H (LCH5)
- LCH6_ASIZE_COUNT: E109_1610H (LCH6)
- LCH7_ASIZE_COUNT: E109_1710H (LCH7)

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							

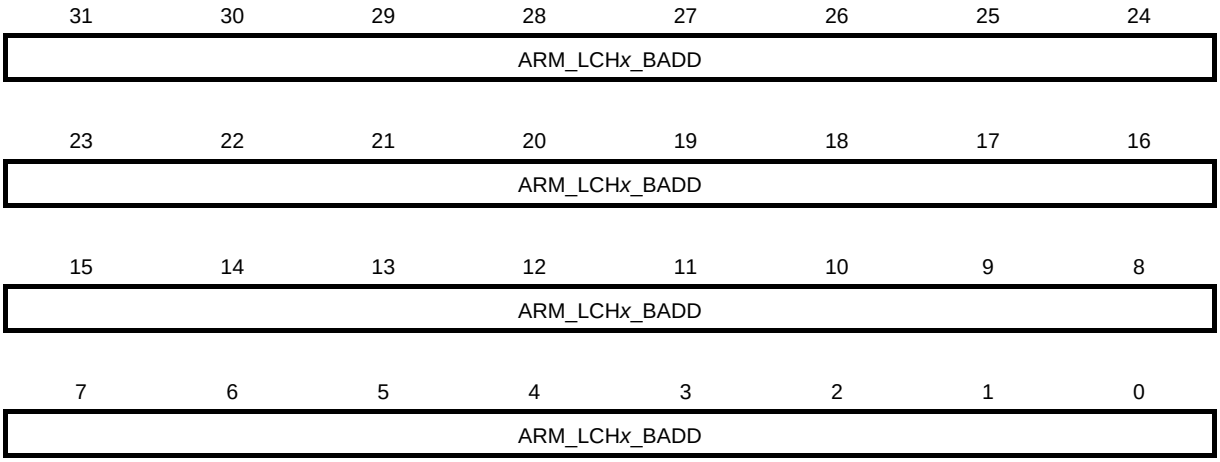
7	6	5	4	3	2	1	0
Reserved		ARM_LCH*x*_ASIZE_COUNT					

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:6	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_ASIZE_COUNT	R/W	5:0	000000b	When written: Specify the number of data blocks to send per loop during a repeat transfer. 000000: 1 block 000001: 2 blocks 000010: 3 blocks 000011: 4 blocks 000100: 5 blocks 000101: 6 blocks 000110: 7 blocks 000111: 8 blocks 001000: 9 blocks ... 111111: 64 blocks When read: Indicates the remaining number of data blocks to send. This number is applied when a DMA transfer starts, and is decremented each time a block transfer ends.

(5) M2M LCH*x* destination address registers

These registers (LCH*x*_BADD) are used to specify the address starting at which to store the received data in bytes.

- LCH0_BADD: E109_1020H (LCH0)
- LCH1_BADD: E109_1120H (LCH1)
- LCH2_BADD: E109_1220H (LCH2)
- LCH3_BADD: E109_1320H (LCH3)
- LCH4_BADD: E109_1420H (LCH4)
- LCH5_BADD: E109_1520H (LCH5)
- LCH6_BADD: E109_1620H (LCH6)
- LCH7_BADD: E109_1720H (LCH7)

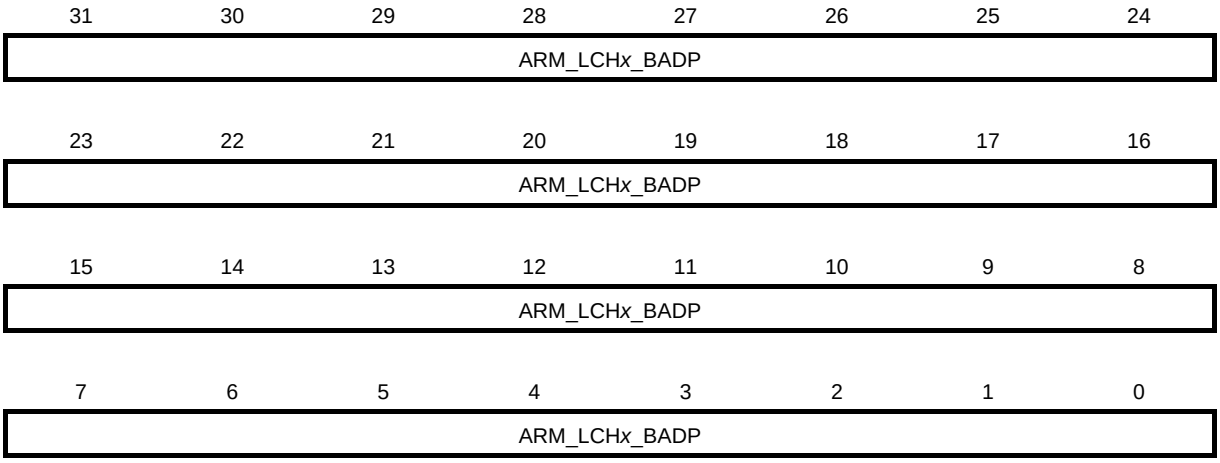


Name	R/W	Bit No.	After Reset	Description
ARM_LCH*x*_BADD	R/W	31:0	0000_0000H	Specify the address starting at which to store the received data.

(6) M2M LCH*x* destination address pointer register

These registers (LCH*x*_BADP) indicate the address where the received data is being stored.

- LCH0_BADP: E109_1024H (LCH0)
- LCH1_BADP: E109_1124H (LCH1)
- LCH2_BADP: E109_1224H (LCH2)
- LCH3_BADP: E109_1324H (LCH3)
- LCH4_BADP: E109_1424H (LCH4)
- LCH5_BADP: E109_1524H (LCH5)
- LCH6_BADP: E109_1624H (LCH6)
- LCH7_BADP: E109_1724H (LCH7)



Name	R/W	Bit No.	After Reset	Description
ARM_LCH*x*_BADP	R	31:0	0000_0000H	Indicates the address where the data received on LCH*x* is being stored.

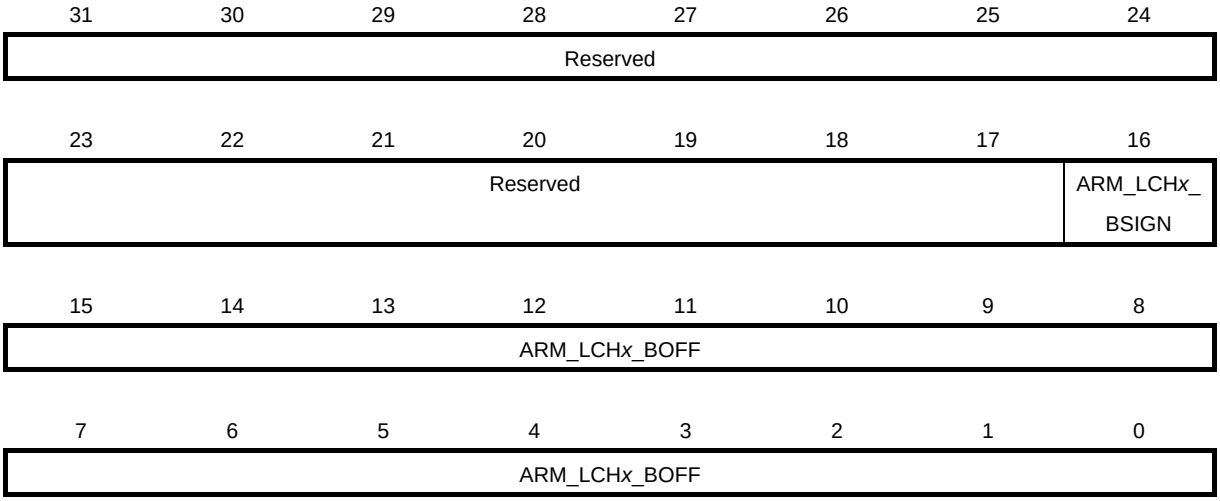
(7) M2M LCHx destination address offset registers

These registers (LCHx_BOFF) are used to specify the offset between data blocks to receive in bytes.

Bit 16 is a sign bit. Clearing this bit to 0 adds the specified value and setting this bit to 1 subtracts the specified value.

Up to 65,535 bytes can be specified.

- LCH0_BOFF: E109_1028H (LCH0)
- LCH1_BOFF: E109_1128H (LCH1)
- LCH2_BOFF: E109_1228H (LCH2)
- LCH3_BOFF: E109_1328H (LCH3)
- LCH4_BOFF: E109_1428H (LCH4)
- LCH5_BOFF: E109_1528H (LCH5)
- LCH6_BOFF: E109_1628H (LCH6)
- LCH7_BOFF: E109_1728H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:17	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_BSIGN	R/W	16	0	Specify the offset added to or subtracted from the start address on the destination side. 0: Add the offset specified for ARM_LCH*x*_BOFF to the start address. 1: Subtract the offset specified for ARM_LCH*x*_BOFF from the start address.
ARM_LCHx_BOFF	R/W	15:0	0000H	Indicates the offset between data blocks to receive on LCH*x* in bytes. 00000000_00000000: 0 bytes (no offset) 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

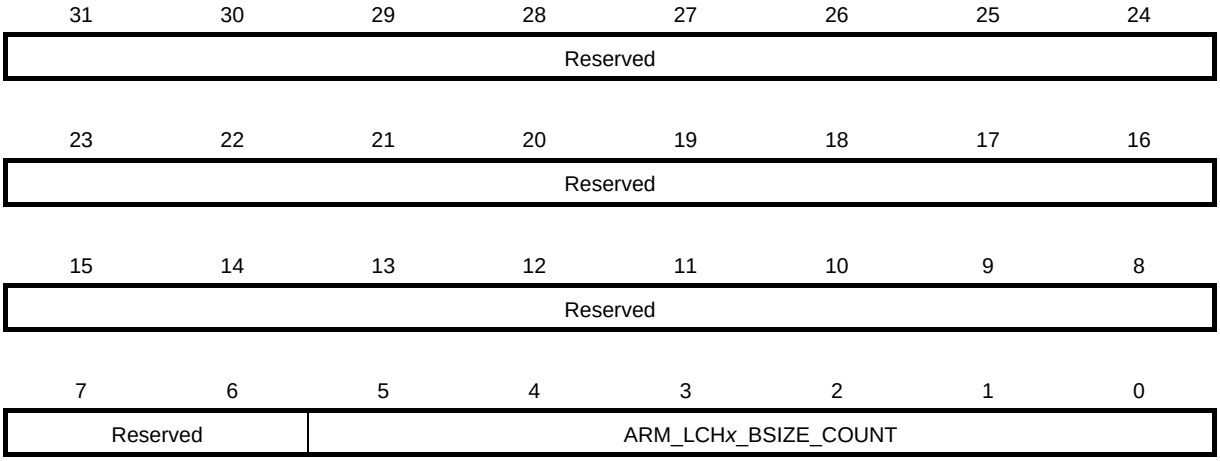
(8) M2M LCH*x* destination block count registers

These registers (LCH*x*_BSIZE_COUNT) work differently when read than when written.

When these registers are written, the number of data blocks transferred per loop during a repeat transfer is set.

When these registers are read, the remaining number of data blocks to receive is read. The number is decremented each time a block transfer ends, and the remaining number is shown in the corresponding register. To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_BSIZE_COUNT: E109_1030H (LCH0)
- LCH1_BSIZE_COUNT: E109_1130H (LCH1)
- LCH2_BSIZE_COUNT: E109_1230H (LCH2)
- LCH3_BSIZE_COUNT: E109_1330H (LCH3)
- LCH4_BSIZE_COUNT: E109_1430H (LCH4)
- LCH5_BSIZE_COUNT: E109_1530H (LCH5)
- LCH6_BSIZE_COUNT: E109_1630H (LCH6)
- LCH7_BSIZE_COUNT: E109_1730H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:6	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_BSIZE_COUNT	R/W	5:0	000000b	When written: Specify the number of data blocks to receive per loop during a repeat transfer. 000000: 1 block 000001: 2 blocks 000010: 3 blocks 000011: 4 blocks 000100: 5 blocks 000101: 6 blocks 000110: 7 blocks 000111: 8 blocks 001000: 9 blocks ... 111111: 64 blocks When read: Indicates the remaining number of transfer data blocks to receive. This number is applied when a DMA transfer starts, and is decremented each time a block transfer ends.

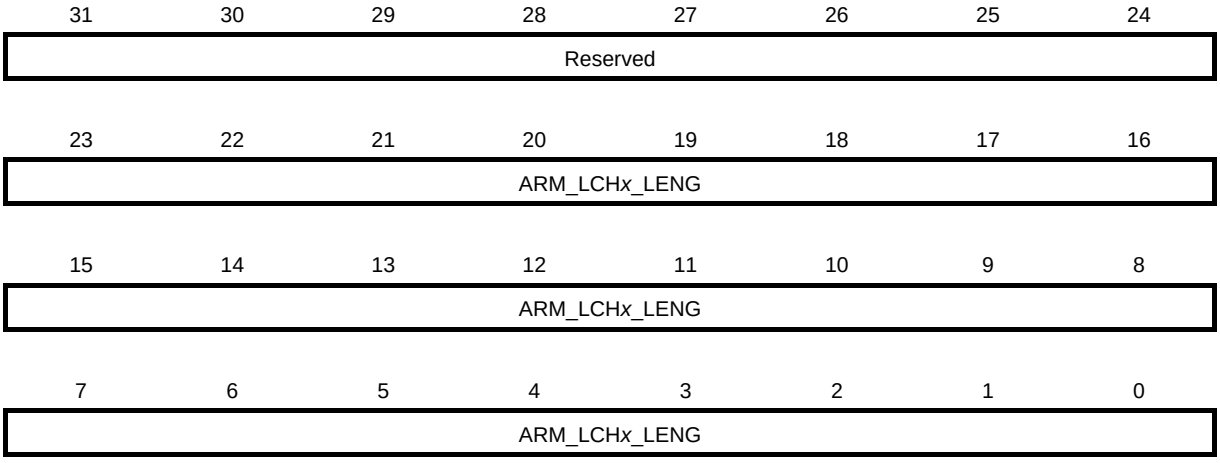
(9) M2M LCH*x* length registers

These registers (LCH*x*_LENG) are used to specify the number of bytes to transfer. Up to 16,777,215 bytes can be specified.

If one of these registers is cleared to 0 while the repeat mode is specified in its M2M LCH*x* mode register, an infinite-length transfer is specified.

Caution Do not clear these registers to 0 if the repeat mode is not specified. Specifying an offset for an infinite-length transfer is prohibited.

- LCH0_LENG: E109_1040H (LCH0)
- LCH1_LENG: E109_1140H (LCH1)
- LCH2_LENG: E109_1240H (LCH2)
- LCH3_LENG: E109_1340H (LCH3)
- LCH4_LENG: E109_1440H (LCH4)
- LCH5_LENG: E109_1540H (LCH5)
- LCH6_LENG: E109_1640H (LCH6)
- LCH7_LENG: E109_1740H (LCH7)

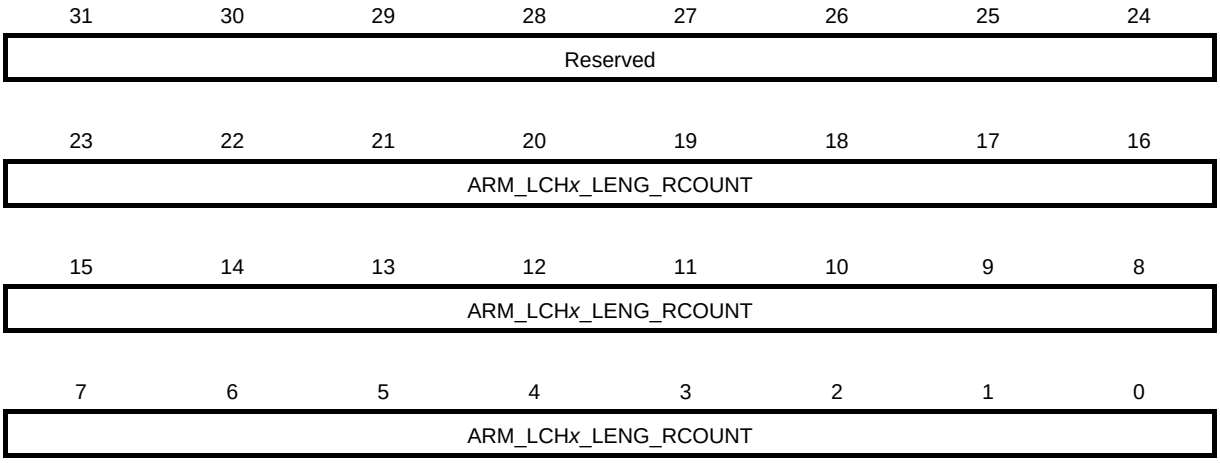


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_LENG	R/W	23:0	00_0000H	Specify the number of bytes to transfer on LCHx. 00000000_00000000_00000000: Infinite-length transfer 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(10) M2M LCH*x* read length count registers

These registers (LCH*x*_LENG_RCOUNT) indicate the remaining amount of data to send. The registers indicate the data amount decremented from the value specified for the M2M LCH*x* length register.

- LCH0_LENG_RCOUNT: E109_1044H (LCH0)
- LCH1_LENG_RCOUNT: E109_1144H (LCH1)
- LCH2_LENG_RCOUNT: E109_1244H (LCH2)
- LCH3_LENG_RCOUNT: E109_1344H (LCH3)
- LCH4_LENG_RCOUNT: E109_1444H (LCH4)
- LCH5_LENG_RCOUNT: E109_1544H (LCH5)
- LCH6_LENG_RCOUNT: E109_1644H (LCH6)
- LCH7_LENG_RCOUNT: E109_1744H (LCH7)

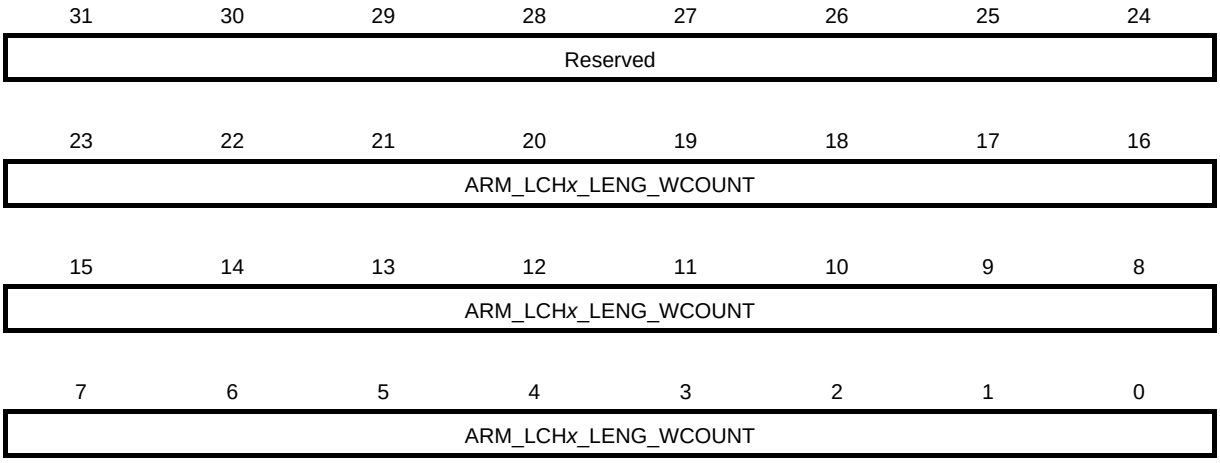


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_LENG_RCOUNT	R	23:0	00_0000H	Indicates the remaining amount of data to send on LCH*x* in bytes. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(11) M2M LCH*x* write length count registers

These registers (LCH*x*_LENG_WCOUNT) indicate the remaining amount of data to receive. The registers indicate the data amount decremented from the value specified for the M2M LCH*x* length register.

- LCH0_LENG_WCOUNT: E109_1048H (LCH0)
- LCH1_LENG_WCOUNT: E109_1148H (LCH1)
- LCH2_LENG_WCOUNT: E109_1248H (LCH2)
- LCH3_LENG_WCOUNT: E109_1348H (LCH3)
- LCH4_LENG_WCOUNT: E109_1448H (LCH4)
- LCH5_LENG_WCOUNT: E109_1548H (LCH5)
- LCH6_LENG_WCOUNT: E109_1648H (LCH6)
- LCH7_LENG_WCOUNT: E109_1748H (LCH7)

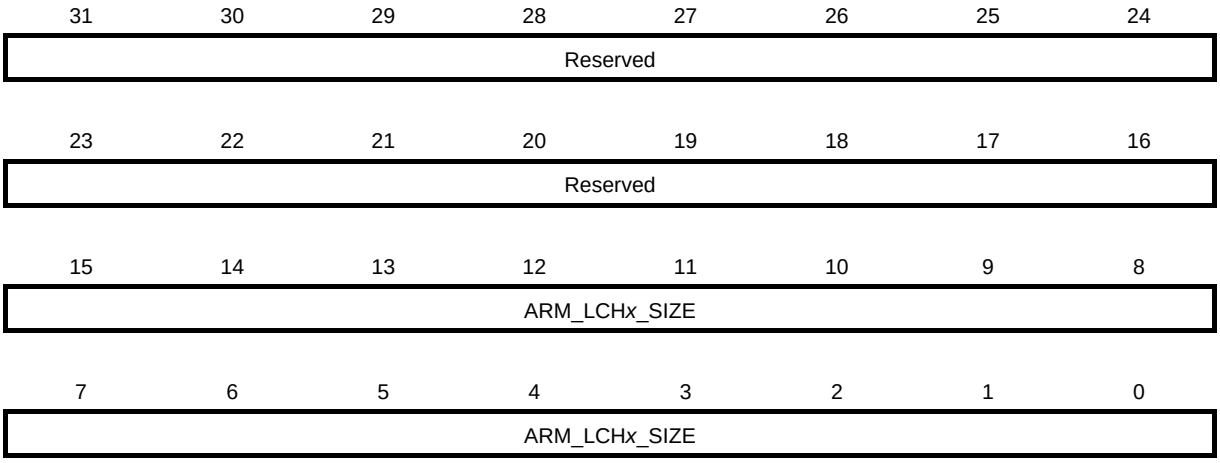


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_LENG_WCOUNT	R	23:0	00_0000H	Indicates the remaining amount of data to receive on LCH*x* in bytes. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(12) M2M LCH*x* block size registers

These registers (LCH*x*_SIZE) are used to specify the block size in bytes. Up to 65,535 bytes can be specified.

- LCH0_SIZE: E109_104CH (LCH0)
- LCH1_SIZE: E109_114CH (LCH1)
- LCH2_SIZE: E109_124CH (LCH2)
- LCH3_SIZE: E109_134CH (LCH3)
- LCH4_SIZE: E109_144CH (LCH4)
- LCH5_SIZE: E109_154CH (LCH5)
- LCH6_SIZE: E109_164CH (LCH6)
- LCH7_SIZE: E109_174CH (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:16	–	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_SIZE	R/W	15:0	0000H	Specify the size of data blocks to send on LCHx. 00000000_00000000: Setting prohibited 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

Cautions 1. If 0 is specified for one of these registers, a block interrupt might occur continuously. In this case, the normal DMA operation is not guaranteed.

2. To prevent a block interrupt from occurring each time the specified size is transferred, set the block size to the same value as the length. (In this case, a block interrupt and a length interrupt occur simultaneously when a DMA transfer finishes.)

(13) M2M LCH*x* mode registers

These registers (LCH*x*_MODE) are used to set up transfers.

To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_MODE: E109_1050H (LCH0)
- LCH1_MODE: E109_1150H (LCH1)
- LCH2_MODE: E109_1250H (LCH2)
- LCH3_MODE: E109_1350H (LCH3)
- LCH4_MODE: E109_1450H (LCH4)
- LCH5_MODE: E109_1550H (LCH5)
- LCH6_MODE: E109_1650H (LCH6)
- LCH7_MODE: E109_1750H (LCH7)

31 30	29 28	27 26	25 24
ARM_LCH*x*_MODE_ ENDI_W_HH	ARM_LCH*x*_MODE_ ENDI_W_HL	ARM_LCH*x*_MODE_ ENDI_W_LH	ARM_LCH*x*_MODE_ ENDI_W_LL

23 22	21 20	19 18	17 16
ARM_LCH*x*_MODE_ ENDI_R_HH	ARM_LCH*x*_MODE_ ENDI_R_HL	ARM_LCH*x*_MODE_ ENDI_R_LH	ARM_LCH*x*_MODE_ ENDI_R_LL

15 14 13 12 11 10 9	8
Reserved	ARM_LCH*x*_ BMODE_ REPEAT

7 6	5 4	3 2 1	0
CH_ORDERING	TYPE	Reserved	ARM_LCH*x*_ AMODE_ REPEAT

Name	R/W	Bit No.	After Reset	Description
ARM_LCH*x*_MODE_ENDI_W_HH	R/W	31:30	E4H	Specify the byte lane for writing data to the transfer destination. 00: Byte 0 01: Byte 1 10: Byte 2 11: Byte 3
ARM_LCH*x*_MODE_ENDI_W_HL		29:28		
ARM_LCH*x*_MODE_ENDI_W_LH		27:26		
ARM_LCH*x*_MODE_ENDI_W_LL		25:24		
ARM_LCH*x*_MODE_ENDI_R_HH	R/W	23:22	E4H	Specify the byte lane for reading data from the transfer source. 00: Byte 0 01: Byte 1 10: Byte 2 11: Byte 3
ARM_LCH*x*_MODE_ENDI_R_HL		21:20		
ARM_LCH*x*_MODE_ENDI_R_LH		19:18		
ARM_LCH*x*_MODE_ENDI_R_LL		17:16		
Reserved	–	15:9	00H	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_BMODE_REPEAT	R/W	8	0	Specify whether to use the repeat mode on the destination side. 0: Do not specify the repeat mode. 1: Specify the repeat mode.
CH_ORDERING	R	7:6	00b	00: Release the physical channel after a block transfer. 01: Use the U and V components of a YUV420 transfer for the resizer. 10: Release the physical channel after a burst transfer. 11: Reserved
TYPE	R	5:4	00b	00: Normal mode 01: Fill mode (Write the AADD value to the destination area.) 10: Mask mode (Selectable only when bits [15:0] of the read data do not equal bits [31:16] of this register) 11: Reserved
Reserved	–	3:1	0H	Reserved. If these bits are read, 0 is returned for each bit.
ARM_LCH*x*_AMODE_REPEAT	R/W	0	0	Specify whether to use the repeat mode on the source side. 0: Do not specify the repeat mode. 1: Specify the repeat mode.

2.3.4 M2P DMA control/status registers

(1) M2P DMA start control register

This register (CONT: E107_0000H) is used to start a DMA transfer on each logical channel.

If this register is set up while the M2P_LCHx_RESERVE bit of the M2P DMA control status register is 0, the subsequent transfer starts immediately after the current transfer ends (simple reservation). A transfer parameter for the subsequent transfer must be specified before using simple reservation. For details, see **3.4.3 Continuous transfer**.

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
M2P_LCH7_CONT	M2P_LCH6_CONT	M2P_LCH5_CONT	M2P_LCH4_CONT	M2P_LCH3_CONT	M2P_LCH2_CONT	M2P_LCH1_CONT	M2P_LCH0_CONT

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH7_CONT	W	7	0	Set this bit to start an M2P DMA transfer on LCH7. (1: Start a DMA transfer.)
M2P_LCH6_CONT	W	6	0	Set this bit to start an M2P DMA transfer on LCH6. (1: Start a DMA transfer.)
M2P_LCH5_CONT	W	5	0	Set this bit to start an M2P DMA transfer on LCH5. (1: Start a DMA transfer.)
M2P_LCH4_CONT	W	4	0	Set this bit to start an M2P DMA transfer on LCH4. (1: Start a DMA transfer.)
M2P_LCH3_CONT	W	3	0	Set this bit to start an M2P DMA transfer on LCH3. (1: Start a DMA transfer.)
M2P_LCH2_CONT	W	2	0	Set this bit to start an M2P DMA transfer on LCH2. (1: Start a DMA transfer.)
M2P_LCH1_CONT	W	1	0	Set this bit to start an M2P DMA transfer on LCH1. (1: Start a DMA transfer.)
M2P_LCH0_CONT	W	0	0	Set this bit to start an M2P DMA transfer on LCH0. (1: Start a DMA transfer.)

(2) M2P DMA control status register

This register (CONTSTATUS: E107_0004H) indicates the DMA status.

This register also indicates whether a DMA transfer is reserved on a logical channel. If a bit is set to 1, the corresponding LCH has been reserved for the next transfer, so another transfer cannot be reserved.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
M2P_LCH7_RESERVE	M2P_LCH6_RESERVE	M2P_LCH5_RESERVE	M2P_LCH4_RESERVE	M2P_LCH3_RESERVE	M2P_LCH2_RESERVE	M2P_LCH1_RESERVE	M2P_LCH0_RESERVE

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
M2P_LCH7_CONTSTATUS	M2P_LCH6_CONTSTATUS	M2P_LCH5_CONTSTATUS	M2P_LCH4_CONTSTATUS	M2P_LCH3_CONTSTATUS	M2P_LCH2_CONTSTATUS	M2P_LCH1_CONTSTATUS	M2P_LCH0_CONTSTATUS

(1/2)

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH7_RESERVE	R	23	0	Indicates whether an M2P DMA transfer is reserved on LCH7. 0: Not reserved 1: Reserved
M2P_LCH6_RESERVE	R	22	0	Indicates whether an M2P DMA transfer is reserved on LCH6. 0: Not reserved 1: Reserved
M2P_LCH5_RESERVE	R	21	0	Indicates whether an M2P DMA transfer is reserved on LCH5. 0: Not reserved 1: Reserved
M2P_LCH4_RESERVE	R	20	0	Indicates whether an M2P DMA transfer is reserved on LCH4. 0: Not reserved 1: Reserved
M2P_LCH3_RESERVE	R	19	0	Indicates whether an M2P DMA transfer is reserved on LCH3. 0: Not reserved 1: Reserved
M2P_LCH2_RESERVE	R	18	0	Indicates whether an M2P DMA transfer is reserved on LCH2. 0: Not reserved 1: Reserved
M2P_LCH1_RESERVE	R	17	0	Indicates whether an M2P DMA transfer is reserved on LCH1. 0: Not reserved 1: Reserved

(2/2)

Name	R/W	Bit No.	After Reset	Description
M2P_LCH0_RESERVE	R	16	0	Indicates whether an M2P DMA transfer is reserved on LCH0. 0: Not reserved 1: Reserved
Reserved	R	15:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH7_CONTSTATUS	R	7	0	Indicates the M2P DMA status on LCH7. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH6_CONTSTATUS	R	6	0	Indicates the M2P DMA status on LCH6. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH5_CONTSTATUS	R	5	0	Indicates the M2P DMA status on LCH5. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH4_CONTSTATUS	R	4	0	Indicates the M2P DMA status on LCH4. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH3_CONTSTATUS	R	3	0	Indicates the M2P DMA status on LCH3. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH2_CONTSTATUS	R	2	0	Indicates the M2P DMA status on LCH2. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH1_CONTSTATUS	R	1	0	Indicates the M2P DMA status on LCH1. 0: No DMA is being performed 1: DMA is being performed
M2P_LCH0_CONTSTATUS	R	0	0	Indicates the M2P DMA status on LCH0. 0: No DMA is being performed 1: DMA is being performed

(3) M2P DMA end control register

This register (END: E107_0008H) is used to terminate DMA transfers.

If a DMA transfer is terminated, the transfer reserved on the channel becomes invalid. If a bit of this register is set, the DMA transfer on the corresponding channel is terminated when the current AHB transaction is completed. Therefore, the DMA transfer cannot be restarted until the AHB transaction ends.

Once a DMA transfer starts, at least one write transfer must be performed before terminating the transfer. Otherwise, parameters are not updated in the internal circuits so the transfer is not executed correctly.

To restart the DMA transfer by setting the DMA transfer start bit, be sure to read the M2P DMA control status register to make sure the corresponding DMA status bit has been cleared to 0.

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
M2P_LCH7_END	M2P_LCH6_END	M2P_LCH5_END	M2P_LCH4_END	M2P_LCH3_END	M2P_LCH2_END	M2P_LCH1_END	M2P_LCH0_END

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH7_END	W	7	0	Set this bit to terminate an M2P DMA transfer on LCH7.
M2P_LCH6_END	W	6	0	Set this bit to terminate an M2P DMA transfer on LCH6.
M2P_LCH5_END	W	5	0	Set this bit to terminate an M2P DMA transfer on LCH5.
M2P_LCH4_END	W	4	0	Set this bit to terminate an M2P DMA transfer on LCH4.
M2P_LCH3_END	W	3	0	Set this bit to terminate an M2P DMA transfer on LCH3.
M2P_LCH2_END	W	2	0	Set this bit to terminate an M2P DMA transfer on LCH2.
M2P_LCH1_END	W	1	0	Set this bit to terminate an M2P DMA transfer on LCH1.
M2P_LCH0_END	W	0	0	Set this bit to terminate an M2P DMA transfer on LCH0.

Remark 0: The current status is retained, 1: Terminate the DMA transfer

2.3.5 M2P interrupt parameter setting registers

These registers are used to set up the parameters for four types of interrupts - length transfer end, block transfer end, error end, and timeout.

(1) M2P interrupt status registers

These registers (LCH*x*LCH*y*_INT_CONT) indicate the interrupt source statuses.
Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_CONT: E107_1100H (LCH0 to LCH3)
- LCH4LCH7_INT_CONT: E107_1120H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25 to 24
Reserved	M2P_LCH3_INT_ERROR_W_CONT	Reserved	M2P_LCH3_INT_LENG_W_CONT	Reserved	M2P_LCH3_INT_ERROR_R_CONT	Reserved

23	22	21	20	19	18	17 to 16
Reserved	M2P_LCH2_INT_ERROR_W_CONT	Reserved	M2P_LCH2_INT_LENG_W_CONT	Reserved	M2P_LCH2_INT_ERROR_R_CONT	Reserved

15	14	13	12	11	10	9 to 8
Reserved	M2P_LCH1_INT_ERROR_W_CONT	Reserved	M2P_LCH1_INT_LENG_W_CONT	Reserved	M2P_LCH1_INT_ERROR_R_CONT	Reserved

7	6	5	4	3	2	1 to 0
M2P_LCH0_INT_TIME_W_CONT	M2P_LCH0_INT_ERROR_W_CONT	Reserved	M2P_LCH0_INT_LENG_W_CONT	Reserved	M2P_LCH0_INT_ERROR_R_CONT	Reserved

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_W_CONT	R	30	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH3.
Reserved	–	29	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ LENG_W_CONT	R	28	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_R_CONT	R	26	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH2_INT_ ERROR_W_CONT	R	22	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH2.
Reserved	–	21	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_ LENG_W_CONT	R	20	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_ ERROR_R_CONT	R	18	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH1_INT_ ERROR_W_CONT	R	14	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH1.
Reserved	–	13	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ LENG_W_CONT	R	12	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH1 finishes.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ ERROR_R_CONT	R	10	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH1.
Reserved	–	9:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_INT_ TIME_W_CONT	R	7	0	Indicates the status of the interrupt that occurs when an M2P transfer times out on LCH0.
M2P_LCH0_INT_ ERROR_W_CONT	R	6	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH0.
Reserved	–	5	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ LENG_W_CONT	R	4	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH0 finishes.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ ERROR_R_CONT	R	2	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(2) M2P interrupt raw status registers

These registers (LCH*x*LCH*y*_INT_RAW) can be used to read the status of the interrupt sources regardless of the settings of the interrupt enable set register and the interrupt enable clear register.
Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_RAW: E107_1104H (LCH0 to LCH3)
- LCH4LCH7_INT_RAW: E107_1124H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25 to 24
Reserved	M2P_LCH3_INT_ERROR_W_RAW	Reserved	M2P_LCH3_INT_LENG_W_RAW	Reserved	M2P_LCH3_INT_ERROR_R_RAW	Reserved

23	22	21	20	19	18	17 to 16
Reserved	M2P_LCH2_INT_ERROR_W_RAW	Reserved	M2P_LCH2_INT_LENG_W_RAW	Reserved	M2P_LCH2_INT_ERROR_R_RAW	Reserved

15	14	13	12	11	10	9 to 8
Reserved	M2P_LCH1_INT_ERROR_W_RAW	Reserved	M2P_LCH1_INT_LENG_W_RAW	Reserved	M2P_LCH1_INT_ERROR_R_RAW	Reserved

7	6	5	4	3	2	1 to 0
M2P_LCH0_INT_TIME_W_RAW	M2P_LCH0_INT_ERROR_W_RAW	Reserved	M2P_LCH0_INT_LENG_W_RAW	Reserved	M2P_LCH0_INT_ERROR_R_RAW	Reserved

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_W_RAW	R	30	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH3.
Reserved	–	29	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ LENG_W_RAW	R	28	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_R_RAW	R	26	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH2_INT_ ERROR_W_RAW	R	22	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH2.
Reserved	–	21	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_LEN G_W_RAW	R	20	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_ ERROR_R_RAW	R	18	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH1_INT_ ERROR_W_RAW	R	14	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH1.
Reserved	–	13	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ LENG_W_RAW	R	12	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH1 finishes.
Reserved	–	11	0	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ ERROR_R_RAW	R	10	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH1.
Reserved	–	9:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_INT_ TIME_W_RAW	R	7	0	Indicates the status of the interrupt that occurs when an M2P transfer times out on LCH0.
M2P_LCH0_INT_ ERROR_W_RAW	R	6	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH0.
Reserved	–	5	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ LENG_W_RAW	R	4	0	Indicates the status of the interrupt that occurs when an M2P length transfer on LCH0 finishes.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ ERROR_R_RAW	R	2	0	Indicates the status of the interrupt caused by an M2P transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(3) M2P interrupt enable set registers

These registers (LCH*x*LCH*y*_INT_ENABLE) are used to enable interrupt sources. Only data of bits to which 1 is written is updated. Masking of interrupt sources corresponding to bits to which 1 is written is cancelled.
Read these registers to check whether an interrupt is enabled. Writing 0 to these registers is ignored.
To disable an interrupt source, set the corresponding bit of the interrupt enable clear register to 1.

- LCH0LCH3_INT_ENABLE: E107_1108H (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE: E107_1128H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved					M2P_LCH3_ INT_ERROR_ ENABLE	M2P_LCH3_ INT_BLOCK_ ENABLE	M2P_LCH3_ INT_LENG_ ENABLE

23	22	21	20	19	18	17	16
Reserved					M2P_LCH2_ INT_ERROR_ ENABLE	M2P_LCH2_ INT_BLOCK_ ENABLE	M2P_LCH2_ INT_LENG_ ENABLE

15	14	13	12	11	10	9	8
Reserved					M2P_LCH1_ INT_ERROR_ ENABLE	M2P_LCH1_ INT_BLOCK_ ENABLE	M2P_LCH1_ INT_LENG_ ENABLE

7	6	5	4	3	2	1	0
Reserved				M2P_LCH0_ INT_TIME_ ENABLE	M2P_LCH0_ INT_ERROR_ ENABLE	M2P_LCH0_ INT_BLOCK_ ENABLE	M2P_LCH0_ INT_LENG_ ENABLE

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH3_INT_ ERROR_ENABLE	R/W	26	0	Enable the M2P LCH3 error interrupt source.
M2P_LCH3_INT_ BLOCK_ENABLE	R/W	25	0	Enable the M2P LCH3 block interrupt source.
M2P_LCH3_INT_ LENG_ENABLE	R/W	24	0	Enable the M2P LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH2_INT_ ERROR_ENABLE	R/W	18	0	Enable the M2P LCH2 error interrupt source.
M2P_LCH2_INT_ BLOCK_ENABLE	R/W	17	0	Enable the M2P LCH2 block interrupt source.
M2P_LCH2_INT_ LENG_ENABLE	R/W	16	0	Enable the M2P LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH1_INT_ ERROR_ENABLE	R/W	10	0	Enable the M2P LCH1 error interrupt source.
M2P_LCH1_INT_ BLOCK_ENABLE	R/W	9	0	Enable the M2P LCH1 block interrupt source.
M2P_LCH1_INT_ LENG_ENABLE	R/W	8	0	Enable the M2P LCH1 length interrupt source.
Reserved	–	7:4	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_INT_ TIME_ENABLE	R/W	3	0	Enable the M2P LCH0 timeout interrupt source.
M2P_LCH0_INT_ ERROR_ENABLE	R/W	2	0	Enable the M2P LCH0 error interrupt source.
M2P_LCH0_INT_ BLOCK_ENABLE	R/W	1	0	Enable the M2P LCH0 block interrupt source.
M2P_LCH0_INT_ LENG_ENABLE	R/W	0	0	Enable the M2P LCH0 length interrupt source.

Remark 0: Disable the interrupt source (default), 1: Enable the interrupt source

(4) M2P interrupt enable clear registers

These registers (LCH*x*LCH*y*_INT_ENABLE_CL) are used to disable interrupt sources.

If a bit of these registers is set to 1, the corresponding interrupt source is disabled. The bits to which 0 is written retain the current settings.

If interrupt sources are disabled in these registers, the corresponding bits in the ARM interrupt enable set register is set to 0 (disables the interrupt sources).

- LCH0LCH3_INT_ENABLE_CL: E107_110CH (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE_CL: E107_112CH (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved					M2P_LCH3_INT_ERROR_ENABLE_CL	M2P_LCH3_INT_BLOCK_ENABLE_CL	M2P_LCH3_INT_LENG_ENABLE_CL

23	22	21	20	19	18	17	16
Reserved					M2P_LCH2_INT_ERROR_ENABLE_CL	M2P_LCH2_INT_BLOCK_ENABLE_CL	M2P_LCH2_INT_LENG_ENABLE_CL

15	14	13	12	11	10	9	8
Reserved					M2P_LCH1_INT_ERROR_ENABLE_CL	M2P_LCH1_INT_BLOCK_ENABLE_CL	M2P_LCH1_INT_LENG_ENABLE_CL

7	6	5	4	3	2	1	0
Reserved				M2P_LCH0_INT_TIME_ENABLE_CL	M2P_LCH0_INT_ERROR_ENABLE_CL	M2P_LCH0_INT_BLOCK_ENABLE_CL	M2P_LCH0_INT_LENG_ENABLE_CL

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH3_INT_ ERROR_ENABLE_CL	W	26	0	Disable the M2P LCH3 error interrupt source.
M2P_LCH3_INT_ BLOCK_ENABLE_CL	W	25	0	Disable the M2P LCH3 block interrupt source.
M2P_LCH3_INT_ LENG_ENABLE_CL	W	24	0	Disable the M2P LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH2_INT_ ERROR_ENABLE_CL	W	18	0	Disable the M2P LCH2 error interrupt source.
M2P_LCH2_INT_ BLOCK_ENABLE_CL	W	17	0	Disable the M2P LCH2 block interrupt source.
M2P_LCH2_INT_LEN G_ENABLE_CL	W	16	0	Disable the M2P LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH1_INT_ ERROR_ENABLE_CL	W	10	0	Disable the M2P LCH1 error interrupt source.
M2P_LCH1_INT_ BLOCK_ENABLE_CL	W	9	0	Disable the M2P LCH1 block interrupt source.
M2P_LCH1_INT_ LENG_ENABLE_CL	W	8	0	Disable the M2P LCH1 length interrupt source.
Reserved	–	7:4	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_INT_ TIME_ENABLE_CL	W	3	0	Disable the M2P LCH0 timeout interrupt source.
M2P_LCH0_INT_ ERROR_ENABLE_CL	W	2	0	Disable the M2P LCH0 error interrupt source.
M2P_LCH0_INT_ BLOCK_ENABLE_CL	W	1	0	Disable the M2P LCH0 block interrupt source.
M2P_LCH0_INT_ LENG_ENABLE_CL	W	0	0	Disable the M2P LCH0 length interrupt source.

Remark 0: Enable the interrupt source (default), 1: Disable the interrupt source

(5) M2P interrupt source clear registers

These registers (LCH*x*LCH*y*_INT_REQ_CL) are used to clear interrupt sources. Only data of bits to which 1 is written is updated.

Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_REQ_CL: E107_1110H (LCH0 to LCH3)
- LCH4LCH7_INT_REQ_CL: E107_1130H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved	M2P_LCH3_INT_ERROR_W_REQ_CL	Reserved	M2P_LCH3_INT_LENG_W_REQ_CL	Reserved	M2P_LCH3_INT_ERROR_R_REQ_CL	Reserved	

23	22	21	20	19	18	17	16
Reserved	M2P_LCH2_INT_ERROR_W_REQ_CL	Reserved	M2P_LCH2_INT_LENG_W_REQ_CL	Reserved	M2P_LCH2_INT_ERROR_R_REQ_CL	Reserved	

15	14	13	12	11	10	9	8
Reserved	M2P_LCH1_INT_ERROR_W_REQ_CL	Reserved	M2P_LCH1_INT_LENG_W_REQ_CL	Reserved	M2P_LCH1_INT_ERROR_R_REQ_CL	Reserved	

7	6	5	4	3	2	1	0
M2P_LCH0_INT_TIME_W_REQ_CL	M2P_LCH0_INT_ERROR_W_REQ_CL	Reserved	M2P_LCH0_INT_LENG_W_REQ_CL	Reserved	M2P_LCH0_INT_ERROR_R_REQ_CL	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	0	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_W_REQ_CL	W	30	0	Clear the M2P LCH3 error interrupt source.
Reserved	–	29	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ LENG_W_REQ_CL	W	28	0	Clear the M2P LCH3 length interrupt source.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH3_INT_ ERROR_R_REQ_CL	W	26	0	Clear the M2P LCH3 error interrupt source.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH2_INT_ ERROR_W_REQ_CL	W	22	0	Clear the M2P LCH2 error interrupt source.
Reserved	–	21	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_ LENG_W_REQ_CL	W	20	0	Clear the M2P LCH2 length interrupt source.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH2_INT_ ERROR_R_REQ_CL	W	18	0	Clear the M2P LCH2 error interrupt source.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH1_INT_ ERROR_W_REQ_CL	W	14	0	Clear the M2P LCH1 error interrupt source.
Reserved	–	13	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ LENG_W_REQ_CL	W	12	0	Clear the M2P LCH1 length interrupt source.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH1_INT_ ERROR_R_REQ_CL	W	10	0	Clear the M2P LCH1 error interrupt source.
Reserved	–	9:8	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_INT_ TIME_W_REQ_CL	W	7	0	Clear the M2P LCH0 timeout interrupt source.
M2P_LCH0_INT_ ERROR_W_REQ_CL	W	6	0	Clear the M2P LCH0 error interrupt source.
Reserved	–	5	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ LENG_W_REQ_CL	W	4	0	Clear the M2P LCH0 length interrupt source.
Reserved	–	3	–	Reserved. If this bit is read, 0 is returned.
M2P_LCH0_INT_ ERROR_R_REQ_CL	W	2	0	Clear the M2P LCH0 error interrupt source.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No operation (retains the current setting), 1: Clear the interrupt source

2.3.6 M2P LCH*x* parameter setting registers

These registers are used to specify the settings for each M2P logical channel. The letter *x* in LCH*x* represents a channel number, which ranges from 0 to 7.

(1) M2P LCHx source address register

These registers (LCH*x*_AADD) are used to specify the address from which to send data in bytes.

- LCH0_AADD: E107_1000H (LCH0)
- LCH1_AADD: E107_1100H (LCH1)
- LCH2_AADD: E107_1200H (LCH2)
- LCH3_AADD: E107_1300H (LCH3)
- LCH4_AADD: E107_1400H (LCH4)
- LCH5_AADD: E107_1500H (LCH5)
- LCH6_AADD: E107_1600H (LCH6)
- LCH7_AADD: E107_1700H (LCH7)

31	30	29	28	27	26	25	24
M2P_LCH*x*_AADD							

23	22	21	20	19	18	17	16
M2P_LCH*x*_AADD							

15	14	13	12	11	10	9	8
M2P_LCH*x*_AADD							

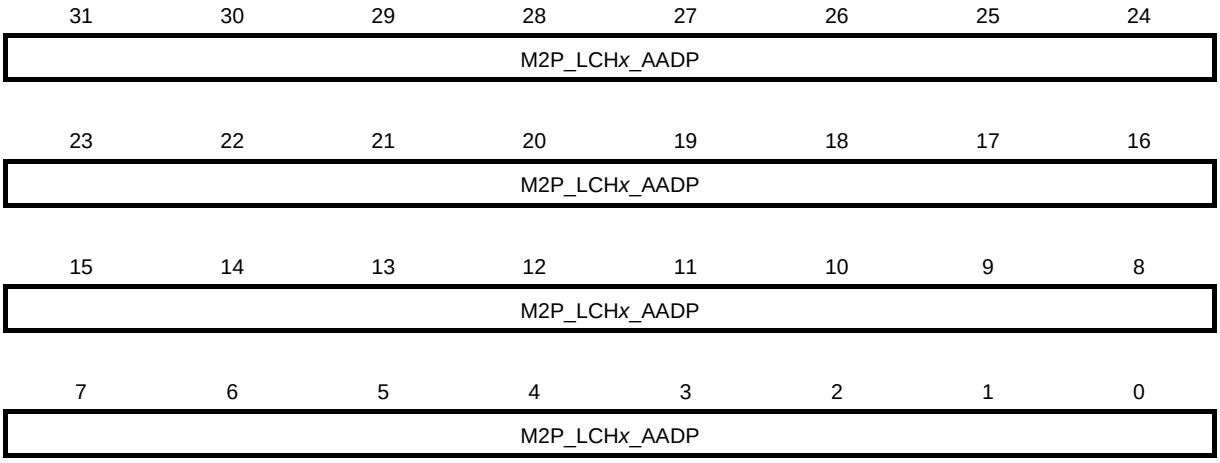
7	6	5	4	3	2	1	0
M2P_LCH*x*_AADD							

Name	R/W	Bit No.	After Reset	Description
M2P_LCH*x*_AADD	R/W	31:0	0000_0000H	Specify the address from which to send data.

(2) M2P LCH*x* source address pointer register

These registers (LCH*x*_AADP) indicate the address from which data is being sent.

- LCH0_AADP: E107_1004H (LCH0)
- LCH1_AADP: E107_1104H (LCH1)
- LCH2_AADP: E107_1204H (LCH2)
- LCH3_AADP: E107_1304H (LCH3)
- LCH4_AADP: E107_1404H (LCH4)
- LCH5_AADP: E107_1504H (LCH5)
- LCH6_AADP: E107_1604H (LCH6)
- LCH7_AADP: E107_1704H (LCH7)

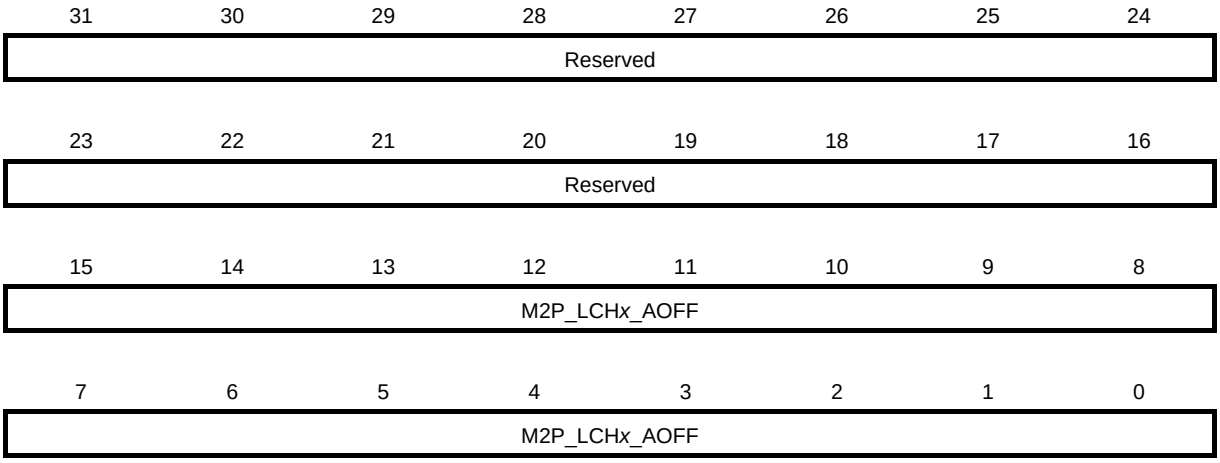


Name	R/W	Bit No.	After Reset	Description
M2P_LCH*x*_AADP	R	31:0	0000_0000H	Indicates the address from which data is being sent on LCH*x* by way of the M2P transfer.

(3) M2P LCH*x* source address offset register

These registers (LCH*x*_AOFF) are used to specify the offset between data blocks to send in bytes. Up to 65,535 bytes can be specified.

- LCH0_AOFF: E107_1008H (LCH0)
- LCH1_AOFF: E107_1108H (LCH1)
- LCH2_AOFF: E107_1208H (LCH2)
- LCH3_AOFF: E107_1308H (LCH3)
- LCH4_AOFF: E107_1408H (LCH4)
- LCH5_AOFF: E107_1508H (LCH5)
- LCH6_AOFF: E107_1608H (LCH6)
- LCH7_AOFF: E107_1708H (LCH7)



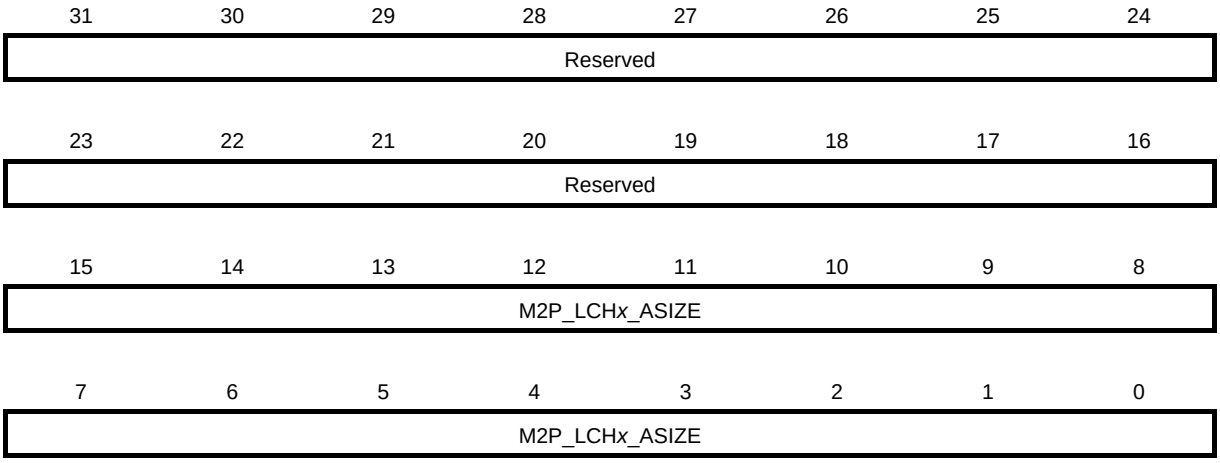
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_AOFF	R/W	15:0	0000H	Indicates the offset between data blocks to send on LCH*x* by way of the M2P transfer in bytes. 00000000_00000000: 0 bytes (no offset) 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

(4) M2P LCH*x* source block size register

These registers (LCH*x*_ASIZE) are used to specify the size of the data blocks to send in bytes.

Up to 65,535 bytes can be specified.

- LCH0_ASIZE: E107_100CH (LCH0)
- LCH1_ASIZE: E107_110CH (LCH1)
- LCH2_ASIZE: E107_120CH (LCH2)
- LCH3_ASIZE: E107_130CH (LCH3)
- LCH4_ASIZE: E107_140CH (LCH4)
- LCH5_ASIZE: E107_150CH (LCH5)
- LCH6_ASIZE: E107_160CH (LCH6)
- LCH7_ASIZE: E107_170CH (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:16	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_ASIZE	R/W	15:0	0000H	Specify the size of data blocks to send on LCH*x* by way of the M2P transfer. 00000000_00000000: Setting prohibited 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

Cautions 1. If 0 is specified for one of these registers, a block interrupt might occur continuously. In this case, the normal DMA operation is not guaranteed.

2. To prevent a block interrupt from occurring each time the specified size is transferred, set the block size to the same value as the length. (In this case, a block interrupt and a length interrupt occur simultaneously when a DMA transfer finishes.)

(5) M2P LCH*x* source block count registers

These registers (LCH*x*_ASIZE_COUNT) work differently when read than when written.

When these registers are written, the number of data blocks to send per loop during a repeat transfer is set. When these registers are read, the remaining number of data blocks to send is read. The number is decremented each time a block transfer ends, and the remaining number is shown in the corresponding register. To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_ASIZE_COUNT: E107_1010H (LCH0)
- LCH1_ASIZE_COUNT: E107_1110H (LCH1)
- LCH2_ASIZE_COUNT: E107_1210H (LCH2)
- LCH3_ASIZE_COUNT: E107_1310H (LCH3)
- LCH4_ASIZE_COUNT: E107_1410H (LCH4)
- LCH5_ASIZE_COUNT: E107_1510H (LCH5)
- LCH6_ASIZE_COUNT: E107_1610H (LCH6)
- LCH7_ASIZE_COUNT: E107_1710H (LCH7)

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							

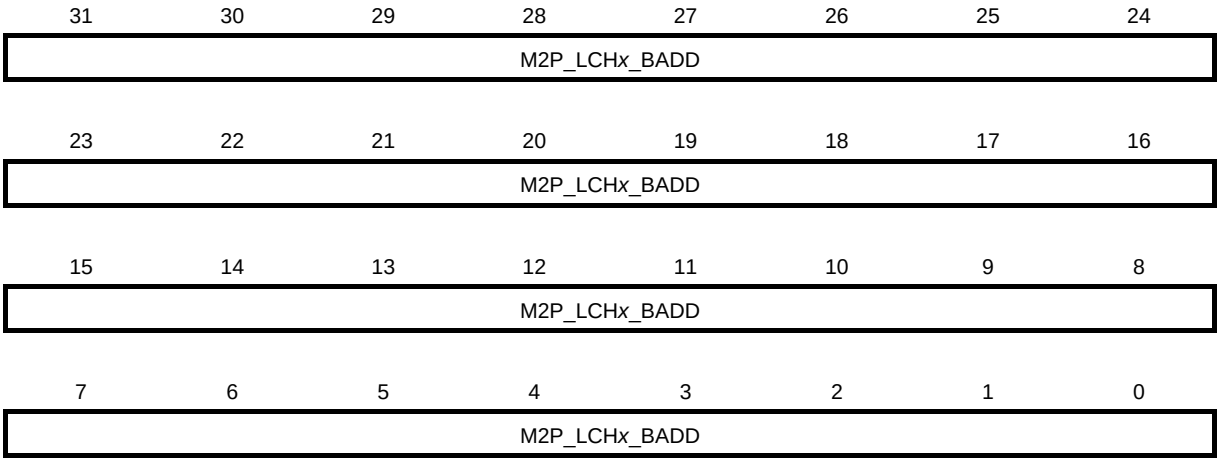
7	6	5	4	3	2	1	0
Reserved		M2P_LCH*x*_ASIZE_COUNT					

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:6	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_ASIZE _COUNT	R/W	5:0	000000b	When written: Specify the number of data blocks to send per loop during a repeat transfer. 000000: 1 block 000001: 2 blocks 000010: 3 blocks 000011: 4 blocks 000100: 5 blocks 000101: 6 blocks 000110: 7 blocks 000111: 8 blocks 001000: 9 blocks ... 111111: 64 blocks When read: Indicates the remaining number of data blocks to send. This number is applied when a DMA transfer starts, and is decremented each time a block transfer ends.

(6) M2P LCH*x* destination address register

These registers (LCH*x*_BADD) are used to specify the address starting at which to store the received data in bytes.

- LCH0_BADD: E107_1020H (LCH0)
- LCH1_BADD: E107_1120H (LCH1)
- LCH2_BADD: E107_1220H (LCH2)
- LCH3_BADD: E107_1320H (LCH3)
- LCH4_BADD: E107_1420H (LCH4)
- LCH5_BADD: E107_1520H (LCH5)
- LCH6_BADD: E107_1620H (LCH6)
- LCH7_BADD: E107_1720H (LCH7)



Name	R/W	Bit No.	After Reset	Description
M2P_LCH*x*_BADD	R/W	31:0	0000_0000H	Specify the address starting at which to store the data received on LCH*x* by way of the M2P transfer.

(7) M2P LCH*x* length registers

These registers (LCH*x*_LENG) are used to specify the number of bytes to transfer. Up to 16,777,215 bytes can be specified.

If one of these registers is cleared to 0 while the repeat mode is specified in its M2P LCH*x* mode register, an infinite-length transfer is specified.

Caution Do not clear these registers to 0 if the repeat mode is not specified. Specifying an offset for an infinite-length transfer is prohibited.

- LCH0_LENG: E107_1040H (LCH0)
- LCH1_LENG: E107_1140H (LCH1)
- LCH2_LENG: E107_1240H (LCH2)
- LCH3_LENG: E107_1340H (LCH3)
- LCH4_LENG: E107_1440H (LCH4)
- LCH5_LENG: E107_1540H (LCH5)
- LCH6_LENG: E107_1640H (LCH6)
- LCH7_LENG: E107_1740H (LCH7)

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
M2P_LCH*x*_LENG							

15	14	13	12	11	10	9	8
M2P_LCH*x*_LENG							

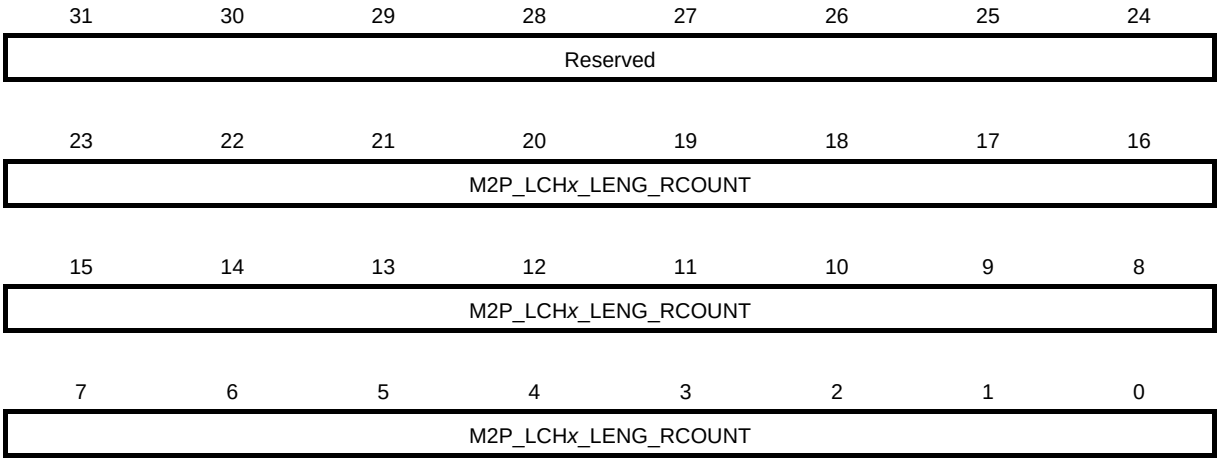
7	6	5	4	3	2	1	0
M2P_LCH*x*_LENG							

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_LENG	R/W	23:0	00_0000H	Specify the number of bytes to transfer on LCH*x* by way of the M2P transfer. 00000000_00000000_00000000: Infinite-length transfer 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(8) M2P LCH*x* read length count registers

These registers (LCH*x*_LENG_RCOUNT) indicate the remaining amount of data to send. The registers indicate the data amount decremented from the value specified for the M2P LCH*x* length register.

- LCH0_LENG_RCOUNT: E107_1044H (LCH0)
- LCH1_LENG_RCOUNT: E107_1144H (LCH1)
- LCH2_LENG_RCOUNT: E107_1244H (LCH2)
- LCH3_LENG_RCOUNT: E107_1344H (LCH3)
- LCH4_LENG_RCOUNT: E107_1444H (LCH4)
- LCH5_LENG_RCOUNT: E107_1544H (LCH5)
- LCH6_LENG_RCOUNT: E107_1644H (LCH6)
- LCH7_LENG_RCOUNT: E107_1744H (LCH7)

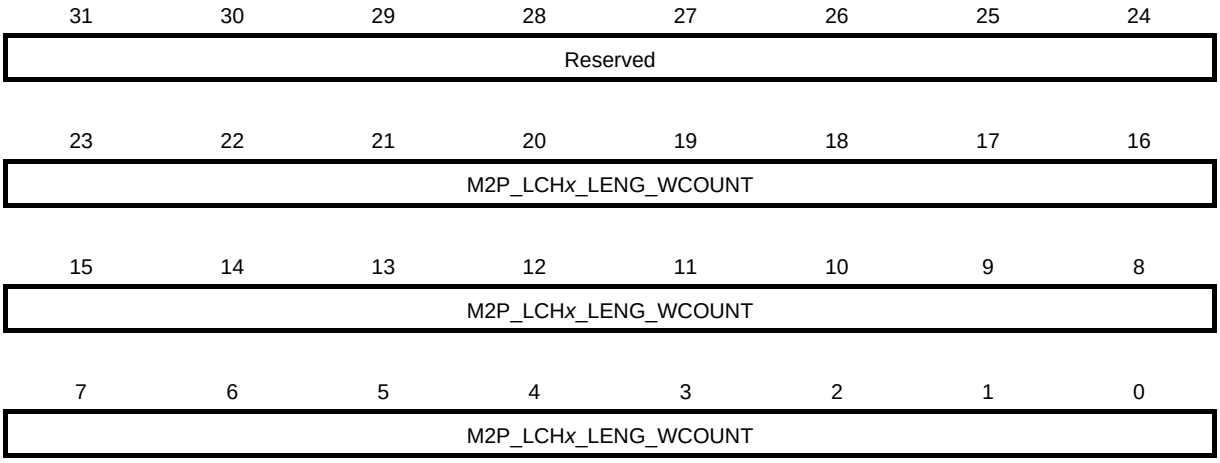


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_LENG_RCOUNT	R	23:0	00_0000H	Specify the number of bytes to send on LCH*x* by way of the M2P transfer in bytes. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(9) M2P LCH*x* write length count registers

These registers (LCH*x*_LENG_WCOUNT) indicate the remaining amount of data to receive. The registers indicate the data amount decremented from the value specified for the M2M LCH*x* length register.

- LCH0_LENG_WCOUNT: E107_1048H (LCH0)
- LCH1_LENG_WCOUNT: E107_1148H (LCH1)
- LCH2_LENG_WCOUNT: E107_1248H (LCH2)
- LCH3_LENG_WCOUNT: E107_1348H (LCH3)
- LCH4_LENG_WCOUNT: E107_1448H (LCH4)
- LCH5_LENG_WCOUNT: E107_1548H (LCH5)
- LCH6_LENG_WCOUNT: E107_1648H (LCH6)
- LCH7_LENG_WCOUNT: E107_1748H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_LENG_WCOUNT	R	23:0	00_0000H	Specify the number of bytes to receive on LCH*x* by way of the M2P transfer. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(10) M2P LCH*x* mode registers

These registers (LCH*x*_MODE) are used to set up transfers.

To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_MODE: E107_1050H (LCH0)
- LCH1_MODE: E107_1150H (LCH1)
- LCH2_MODE: E107_1250H (LCH2)
- LCH3_MODE: E107_1350H (LCH3)
- LCH4_MODE: E107_1450H (LCH4)
- LCH5_MODE: E107_1550H (LCH5)
- LCH6_MODE: E107_1650H (LCH6)
- LCH7_MODE: E107_1750H (LCH7)

31 30	29 28	27 26	25 24
M2P_LCH*x*_MODE_ ENDI_W_HH	M2P_LCH*x*_MODE_ ENDI_W_HL	M2P_LCH*x*_MODE_ ENDI_W_LH	M2P_LCH*x*_MODE_ ENDI_W_LL

23 22	21 20	19 18	17 16
M2P_LCH*x*_MODE_ ENDI_R_HH	M2P_LCH*x*_MODE_ ENDI_R_HL	M2P_LCH*x*_MODE_ ENDI_R_LH	M2P_LCH*x*_MODE_ ENDI_R_LL

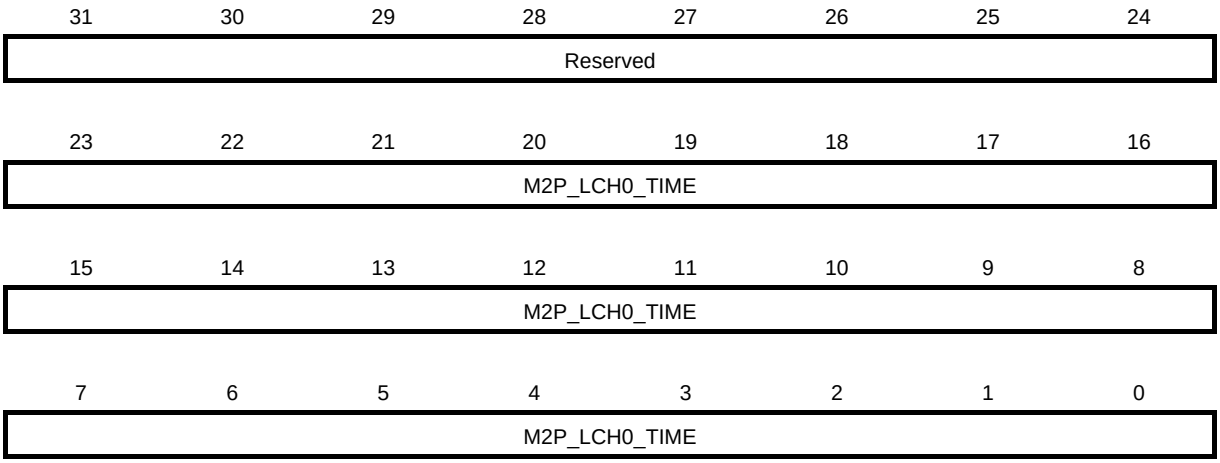
15 14 13 12 11 10 9 8
Reserved

7 6 5 4 3 2 1	0
Reserved	M2P_LCH*x*_ AMODE_ REPEAT

Name	R/W	Bit No.	After Reset	Description
M2P_LCH*x*_MODE_ ENDI_W_HH	R/W	31:30	E4H	Specify the byte lane for writing data to the transfer destination. 00: Byte 0 01: Byte 1 02: Byte 2 03: Byte 3
M2P_LCH*x*_MODE_ ENDI_W_HL	R/W	29:28		
M2P_LCH*x*_MODE_ ENDI_W_LH	R/W	27:26		
M2P_LCH*x*_MODE_ ENDI_W_LL	R/W	25:24		
M2P_LCH*x*_MODE_ ENDI_R_HH	R/W	23:22	E4H	Specify the byte lane for reading data from the transfer source. 00: Byte 0 01: Byte 1 02: Byte 2 03: Byte 3
M2P_LCH*x*_MODE_ ENDI_R_HL	R/W	21:20		
M2P_LCH*x*_MODE_ ENDI_R_LH	R/W	19:18		
M2P_LCH*x*_MODE_ ENDI_R_LL	R/W	17:16		
Reserved	R	15:1	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH*x*_AMODE_ REPEAT	R/W	0	0	Specify whether to use the repeat mode on the source side. 0: Do not specify the repeat mode 1: Specify the repeat mode.

(11) M2P LCH0 timer register

This register (LCH0_TIME: E107_1054H) is used to specify the wait time for the next request to receive. If no DMA request is issued for a specific period and the time specified in this register expires, the current DMA transfer is terminated. Up to 24 bits can be set.



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_TIME	R/W	23:0	00_0000H	Specify the time allowed to elapse before an M2P transfer on LCH0 times out.

Caution Be sure to supply a timer clock (DMA_TCLK) when using a timer. Otherwise, DMA clock control might malfunction.

(12) M2P LCH0 timer count register

This register (LCH0_TIME_COUNT: E107_1058H) is used to count down to the time when the next request is received. The timer is decremented in units of the DMA_TCLK cycle output from the SMU. When the count reaches 0, the current DMA transfer is terminated.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
M2P_LCH0_TIME_COUNT							

15	14	13	12	11	10	9	8
M2P_LCH0_TIME_COUNT							

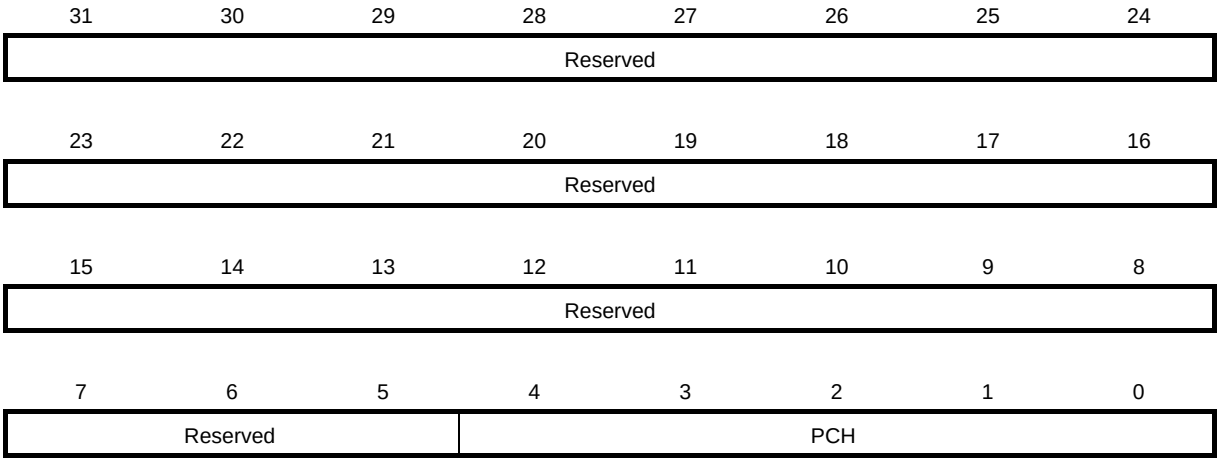
7	6	5	4	3	2	1	0
M2P_LCH0_TIME_COUNT							

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
M2P_LCH0_TIME_ COUNT	R	23:0	00_0000H	Indicates the time allowed to elapse before an M2P transfer on LCH0 times out. DMA_TCLK is used for timer counting.

(13) M2P LCHx physical channel registers

These registers (LCHx_PCH) are used to select the module that sends the DMA request signal (DMARQ).

- LCH0_PCH: E107_105CH (LCH0)
- LCH1_PCH: E107_115CH (LCH1)
- LCH2_PCH: E107_125CH (LCH2)
- LCH3_PCH: E107_135CH (LCH3)
- LCH4_PCH: E107_145CH (LCH4)
- LCH5_PCH: E107_155CH (LCH5)
- LCH6_PCH: E107_165CH (LCH6)
- LCH7_PCH: E107_175CH (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:5	–	Reserved. If these bits are read, 0 is returned for each bit.
PCH	R/W	4:0	00H	Select the module that sends the DMA request signal (DMARQ). 0 to 5: Reserved 6: UART0 7: UART1 8: UART2 9: UART3 10: SIO0 11: SIO1 12: SIO2 13: SIO3 14: SIO4 15: SIO5 16 and 17: Reserved

2.3.7 P2M DMA control/status registers

(1) P2M DMA start control register

This register (CONT: E108_0000H) is used to start a DMA transfer on each logical channel.

If this register is set up while the P2M_LCH*x*_RESERVE bit of the P2M DMA control status register is 0, the subsequent transfer starts immediately after the current transfer ends (simple reservation). A transfer parameter for the subsequent transfer must be specified before using simple reservation. For details, see **3.4.3 Continuous transfer**.

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
P2M_LCH7_CONT	P2M_LCH6_CONT	P2M_LCH5_CONT	P2M_LCH4_CONT	P2M_LCH3_CONT	P2M_LCH2_CONT	P2M_LCH1_CONT	P2M_LCH0_CONT

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH7_CONT	W	7	0	Set this bit to start a P2M DMA transfer on LCH7. (1: Start a DMA transfer.)
P2M_LCH6_CONT	W	6	0	Set this bit to start a P2M DMA transfer on LCH6. (1: Start a DMA transfer.)
P2M_LCH5_CONT	W	5	0	Set this bit to start a P2M DMA transfer on LCH5. (1: Start a DMA transfer.)
P2M_LCH4_CONT	W	4	0	Set this bit to start a P2M DMA transfer on LCH4. (1: Start a DMA transfer.)
P2M_LCH3_CONT	W	3	0	Set this bit to start a P2M DMA transfer on LCH3. (1: Start a DMA transfer.)
P2M_LCH2_CONT	W	2	0	Set this bit to start a P2M DMA transfer on LCH2. (1: Start a DMA transfer.)
P2M_LCH1_CONT	W	1	0	Set this bit to start a P2M DMA transfer on LCH1. (1: Start a DMA transfer.)
P2M_LCH0_CONT	W	0	0	Set this bit to start a P2M DMA transfer on LCH0. (1: Start a DMA transfer.)

(2) P2M DMA control status register

This register (CONTSTATUS: E108_0004H) indicates the DMA status.

This register also indicates whether a DMA transfer is reserved on a logical channel. If a bit is set to 1, the corresponding LCH has been reserved for the next transfer, so another transfer cannot be reserved.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
P2M_LCH7_RESERVE	P2M_LCH6_RESERVE	P2M_LCH5_RESERVE	P2M_LCH4_RESERVE	P2M_LCH3_RESERVE	P2M_LCH2_RSERVE	P2M_LCH1_RSERVE	P2M_LCH0_RSERVE

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
P2M_LCH7_CONTSTATUS	P2M_LCH6_CONTSTATUS	P2M_LCH5_CONTSTATUS	P2M_LCH4_CONTSTATUS	P2M_LCH3_CONTSTATUS	P2M_LCH2_CONTSTATUS	P2M_LCH1_CONTSTATUS	P2M_LCH0_CONTSTATUS

(1/2)

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH7_RESERVE	R	23	0	Indicates whether a P2M DMA transfer is reserved on LCH7. 0: Not reserved 1: Reserved
P2M_LCH6_RESERVE	R	22	0	Indicates whether a P2M DMA transfer is reserved on LCH6. 0: Not reserved 1: Reserved
P2M_LCH5_RESERVE	R	21	0	Indicates whether a P2M DMA transfer is reserved on LCH5. 0: Not reserved 1: Reserved
P2M_LCH4_RESERVE	R	20	0	Indicates whether a P2M DMA transfer is reserved on LCH4. 0: Not reserved 1: Reserved
P2M_LCH3_RESERVE	R	19	0	Indicates whether a P2M DMA transfer is reserved on LCH3. 0: Not reserved 1: Reserved
P2M_LCH2_RESERVE	R	18	0	Indicates whether a P2M DMA transfer is reserved on LCH2. 0: Not reserved 1: Reserved
P2M_LCH1_RESERVE	R	17	0	Indicates whether a P2M DMA transfer is reserved on LCH1. 0: Not reserved 1: Reserved

(2/2)

Name	R/W	Bit No.	After Reset	Description
P2M_LCH0_RESERVE	R	16	0	Indicates whether a P2M DMA transfer is reserved on LCH0. 0: Not reserved 1: Reserved
Reserved	R	15:8	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH7_CONTSTATUS	R	7	0	Indicates the P2M DMA status on LCH7. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH6_CONTSTATUS	R	6	0	Indicates the P2M DMA status on LCH6. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH5_CONTSTATUS	R	5	0	Indicates the P2M DMA status on LCH5. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH4_CONTSTATUS	R	4	0	Indicates the P2M DMA status on LCH4. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH3_CONTSTATUS	R	3	0	Indicates the P2M DMA status on LCH3. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH2_CONTSTATUS	R	2	0	Indicates the P2M DMA status on LCH2. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH1_CONTSTATUS	R	1	0	Indicates the P2M DMA status on LCH1. 0: No DMA is being performed 1: DMA is being performed
P2M_LCH0_CONTSTATUS	R	0	0	Indicates the P2M DMA status on LCH0. 0: No DMA is being performed 1: DMA is being performed

(3) P2M DMA end control register

This register (END: E108_0008H) is used to terminate DMA transfers.

If a DMA transfer is terminated, the transfer reserved on the channel becomes invalid. If a bit of this register is set, the DMA transfer on the corresponding channel is terminated when the current AHB transaction is completed. Therefore, the DMA transfer cannot be restarted until the AHB transaction ends.

Once a DMA transfer starts, at least one write transfer must be performed before terminating the transfer. Otherwise, parameters are not updated in the internal circuits so the transfer is not executed correctly. Before terminating a transfer on a P2M channel, DMA requests from peripherals must be stopped first and data on the LCH stored in the FIFO buffer in the DMA controller must be read out.

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
P2M_LCH7_END	P2M_LCH6_END	P2M_LCH5_END	P2M_LCH4_END	P2M_LCH3_END	P2M_LCH2_END	P2M_LCH1_END	P2M_LCH0_END

Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:8	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH7_END	W	7	0	Set this bit to terminate a P2M DMA transfer on LCH7.
P2M_LCH6_END	W	6	0	Set this bit to terminate a P2M DMA transfer on LCH6.
P2M_LCH5_END	W	5	0	Set this bit to terminate a P2M DMA transfer on LCH5.
P2M_LCH4_END	W	4	0	Set this bit to terminate a P2M DMA transfer on LCH4.
P2M_LCH3_END	W	3	0	Set this bit to terminate a P2M DMA transfer on LCH3.
P2M_LCH2_END	W	2	0	Set this bit to terminate a P2M DMA transfer on LCH2.
P2M_LCH1_END	W	1	0	Set this bit to terminate a P2M DMA transfer on LCH1.
P2M_LCH0_END	W	0	0	Set this bit to terminate a P2M DMA transfer on LCH0.

Remark 0: The current status is retained, 1: Terminate the DMA transfer

2.3.8 P2M interrupt parameter setting registers

These registers are used to set up the parameters for four types of interrupts - length transfer end, block transfer end, error end, and timeout.

(1) P2M interrupt status registers

These registers (LCH*x*LCH*y*_INT_CONT) indicate the interrupt source statuses.
Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_CONT: E108_0100H (LCH0 to LCH3)
- LCH4LCH7_INT_CONT: E108_0120H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved	P2M_LCH3_INT_ERROR_W_CONT	P2M_LCH3_INT_BLOCK_W_CONT	P2M_LCH3_INT_LENG_W_CONT	Reserved	P2M_LCH3_INT_ERROR_R_CONT	Reserved	

23	22	21	20	19	18	17	16
Reserved	P2M_LCH2_INT_ERROR_W_CONT	P2M_LCH2_INT_BLOCK_W_CONT	P2M_LCH2_INT_LENG_W_CONT	Reserved	P2M_LCH2_INT_ERROR_R_CONT	Reserved	

15	14	13	12	11	10	9	8
Reserved	P2M_LCH1_INT_ERROR_W_CONT	P2M_LCH1_INT_BLOCK_W_CONT	P2M_LCH1_INT_LENG_W_CONT	Reserved	P2M_LCH1_INT_ERROR_R_CONT	Reserved	

7	6	5	4	3	2	1	0
Reserved	P2M_LCH0_INT_ERROR_W_CONT	P2M_LCH0_INT_BLOCK_W_CONT	P2M_LCH0_INT_LENG_W_CONT	P2M_LCH0_INT_TIME_R_CONT	P2M_LCH0_INT_ERROR_R_CONT	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_W_CONT	R	30	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH3.
P2M_LCH3_INT_ BLOCK_W_CONT	R	29	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH3 finishes.
P2M_LCH3_INT_ LENG_W_CONT	R	28	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_R_CONT	R	26	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH2_INT_ ERROR_W_CONT	R	22	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH2.
P2M_LCH2_INT_ BLOCK_W_CONT	R	21	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH2 finishes.
P2M_LCH2_INT_ LENG_W_CONT	R	20	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH2_INT_ ERROR_R_CONT	R	18	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH1_INT_ ERROR_W_CONT	R	14	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH1.
P2M_LCH1_INT_ BLOCK_E_CONT	R	13	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH1 finishes.
P2M_LCH1_INT_ LENG_W_CONT	R	12	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH1 finishes.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH1_INT_ ERROR_R_CONT	R	10	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH1.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_INT_ ERROR_W_CONT	R	6	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH0.
P2M_LCH0_INT_ BLOCK_W_CONT	R	5	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH0 finishes.
P2M_LCH0_INT_ LENG_W_CONT	R	4	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH0 finishes.
P2M_LCH0_INT_ TIME_R_CONT	R	3	0	Indicates the status of the interrupt that occurs when a P2M transfer on LCH0 times out.
P2M_LCH0_INT_ ERROR_R_CONT	R	2	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(2) P2M interrupt raw status registers

These registers (LCH*x*LCH*y*_INT_RAW) can be used to read the status of the interrupt sources regardless of the settings of the interrupt enable set register and the interrupt enable clear register. Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_RAW: E108_0104H (LCH0 to LCH3)
- LCH4LCH7_INT_RAW: E108_0124H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved	P2M_LCH3_INT_ERROR_W_RAW	P2M_LCH3_INT_BLOCK_W_RAW	P2M_LCH3_INT_LENG_W_RAW	Reserved	P2M_LCH3_INT_ERROR_R_RAW	Reserved	

23	22	21	20	19	18	17	16
Reserved	P2M_LCH2_INT_ERROR_W_RAW	P2M_LCH2_INT_BLOCK_W_RAW	P2M_LCH2_INT_LENG_W_RAW	Reserved	P2M_LCH2_INT_ERROR_R_RAW	Reserved	

15	14	13	12	11	10	9	8
Reserved	P2M_LCH1_INT_ERROR_W_RAW	P2M_LCH1_INT_BLOCK_W_RAW	P2M_LCH1_INT_LENG_W_RAW	Reserved	P2M_LCH1_INT_ERROR_R_RAW	Reserved	

7	6	5	4	3	2	1	0
Reserved	P2M_LCH0_INT_ERROR_W_RAW	P2M_LCH0_INT_BLOCK_W_RAW	P2M_LCH0_INT_LENG_W_RAW	P2M_LCH0_INT_TIME_R_RAW	P2M_LCH0_INT_ERROR_R_RAW	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_W_RAW	R	30	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH3.
P2M_LCH3_INT_ BLOCK_W_RAW	R	29	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH3 finishes.
P2M_LCH3_INT_ LENG_W_RAW	R	28	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH3 finishes.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_R_RAW	R	26	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH3.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH2_INT_ ERROR_W_RAW	R	22	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH2.
P2M_LCH2_INT_ BLOCK_W_RAW	R	21	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH2 finishes.
P2M_LCH2_INT_ LENG_W_RAW	R	20	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH2 finishes.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH2_INT_ ERROR_R_RAW	R	18	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH2.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH1_INT_ ERROR_W_RAW	R	14	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH1.
P2M_LCH1_INT_ BLOCK_W_RAW	R	13	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH1 finishes.
P2M_LCH1_INT_ LENG_W_RAW	R	12	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH1 finishes.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH1_INT_ ERROR_R_RAW	R	10	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH1.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_INT_ ERROR_W_RAW	R	6	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH0.
P2M_LCH0_INT_ BLOCK_W_RAW	R	5	0	Indicates the status of the interrupt that occurs when a P2M block transfer on LCH0 finishes.
P2M_LCH0_INT_ LENG_W_RAW	R	4	0	Indicates the status of the interrupt that occurs when a P2M length transfer on LCH0 finishes.
P2M_LCH0_INT_ TIME_R_RAW	R	3	0	Indicates the status of the interrupt caused by a P2M LCH0 timeout.
P2M_LCH0_INT_ ERROR_R_RAW	R	2	0	Indicates the status of the interrupt caused by a P2M transfer error on LCH0.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No interrupt source (default), 1: Interrupt source occurred

(3) P2M interrupt enable set registers

These registers (LCH*x*LCH*y*_INT_ENABLE) are used to enable interrupt sources. Only data of bits to which 1 is written is updated. Masking of interrupt sources corresponding to bits to which 1 is written is cancelled.
Read these registers to check whether an interrupt is enabled. Writing 0 to these registers is ignored.
To disable an interrupt source, set the corresponding bit of the interrupt enable clear register to 1.

- LCH0LCH3_INT_ENABLE: E108_0108H (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE: E108_0128H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved					P2M_LCH3_INT_ERROR_ENABLE	P2M_LCH3_INT_BLOCK_ENABLE	P2M_LCH3_INT_LENG_ENABLE

23	22	21	20	19	18	17	16
Reserved					P2M_LCH2_INT_ERROR_ENABLE	P2M_LCH2_INT_BLOCK_ENABLE	P2M_LCH2_INT_LENG_ENABLE

15	14	13	12	11	10	9	8
Reserved					P2M_CH1_INT_ERROR_ENABLE	P2M_CH1_INT_BLOCK_ENABLE	P2M_CH1_INT_LENG_ENABLE

7	6	5	4	3	2	1	0
Reserved				P2M_LCH0_INT_TIME_ENABLE	P2M_LCH0_INT_ERROR_ENABLE	P2M_LCH0_INT_BLOCK_ENABLE	P2M_LCH0_INT_LENG_ENABLE

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH3_INT_ ERROR_ENABLE	R/W	26	0	Enable the P2M LCH3 error interrupt source.
P2M_LCH3_INT_ BLOCK_ENABLE	R/W	25	0	Enable the P2M LCH3 block interrupt source.
P2M_LCH3_INT_ LENG_ENABLE	R/W	24	0	Enable the P2M LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH2_INT_ ERROR_ENABLE	R/W	18	0	Enable the P2M LCH2 error interrupt source.
P2M_LCH2_INT_ BLOCK_ENABLE	R/W	17	0	Enable the P2M LCH2 block interrupt source.
P2M_LCH2_INT_ LENG_ENABLE	R/W	16	0	Enable the P2M LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH1_INT_ ERROR_ENABLE	R/W	10	0	Enable the P2M LCH1 error interrupt source.
P2M_LCH1_INT_ BLOCK_ENABLE	R/W	9	0	Enable the P2M LCH1 block interrupt source.
P2M_LCH1_INT_ LENG_ENABLE	R/W	8	0	Enable the P2M LCH1 length interrupt source.
Reserved	–	7:4	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_INT_ TIME_ENABLE	R/W	3	0	Enable the P2M LCH0 timeout interrupt source.
P2M_LCH0_INT_ ERROR_ENABLE	R/W	2	0	Enable the P2M LCH0 error interrupt source.
P2M_LCH0_INT_ BLOCK_ENABLE	R/W	1	0	Enable the P2M LCH0 block interrupt source.
P2M_LCH0_INT_ LENG_ENABLE	R/W	0	0	Enable the P2M LCH0 length interrupt source.

Remark 0: Disable the interrupt source (default), 1: Enable the interrupt source

(4) P2M interrupt enable clear registers

These registers (LCH*x*LCH*y*_INT_ENABLE_CL) are used to disable interrupt sources.

If a bit of these registers is set to 1, the corresponding interrupt source is disabled. The bits to which 0 is written retain the current settings.

If interrupt sources are disabled in these registers, the corresponding bits in the P2M interrupt enable set register is set to 0 (disables the interrupt sources).

- LCH0LCH3_INT_ENABLE_CL: E108_010CH (LCH0 to LCH3)
- LCH4LCH7_INT_ENABLE_CL: E108_012CH (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved					P2M_LCH3_INT_ERROR_ENABLE_CL	P2M_LCH3_INT_BLOCK_ENABLE_CL	P2M_LCH3_INT_LENG_ENABLE_CL

23	22	21	20	19	18	17	16
Reserved					P2M_LCH2_INT_ERROR_ENABLE_CL	P2M_LCH2_INT_BLOCK_ENABLE_CL	P2M_LCH2_INT_LENG_ENABLE_CL

15	14	13	12	11	10	9	8
Reserved					P2M_LCH1_INT_ERROR_ENABLE_CL	P2M_LCH1_INT_BLOCK_ENABLE_CL	P2M_LCH1_INT_LENG_ENABLE_CL

7	6	5	4	3	2	1	0
Reserved				P2M_LCH0_INT_TIME_ENABLE_CL	P2M_LCH0_INT_ERROR_ENABLE_CL	P2M_LCH0_INT_BLOCK_ENABLE_CL	P2M_LCH0_INT_LENG_ENABLE_CL

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:27	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH3_INT_ ERROR_ENABLE_CL	W	26	0	Disable the P2M LCH3 error interrupt source.
P2M_LCH3_INT_ BLOCK_ENABLE_CL	W	25	0	Disable the P2M LCH3 block interrupt source.
P2M_LCH3_INT_ LENG_ENABLE_CL	W	24	0	Disable the P2M LCH3 length interrupt source.
Reserved	–	23:19	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH2_INT_ ERROR_ENABLE_CL	W	18	0	Disable the P2M LCH2 error interrupt source.
P2M_LCH2_INT_ BLOCK_ENABLE_CL	W	17	0	Disable the P2M LCH2 block interrupt source.
P2M_LCH2_INT_ LENG_ENABLE_CL	W	16	0	Disable the P2M LCH2 length interrupt source.
Reserved	–	15:11	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH1_INT_ ERROR_ENABLE_CL	W	10	0	Disable the P2M LCH1 error interrupt source.
P2M_LCH1_INT_ BLOCK_ENABLE_CL	W	9	0	Disable the P2M LCH1 block interrupt source.
P2M_LCH1_INT_ LENG_ENABLE_CL	W	8	0	Disable the P2M LCH1 length interrupt source.
Reserved	–	7:4	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_INT_ TIME_ENABLE_CL	W	3	0	Disable the P2M LCH0 timeout interrupt source.
P2M_LCH0_INT_ ERROR_ENABLE_CL	W	2	0	Disable the P2M LCH0 error interrupt source.
P2M_LCH0_INT_ BLOCK_ENABLE_CL	W	1	0	Disable the P2M LCH0 block interrupt source.
P2M_LCH0_INT_ LENG_ENABLE_CL	W	0	0	Disable the P2M LCH0 length interrupt source.

Remark 0: Enable the interrupt source (default), 1: Disable the interrupt source

(5) P2M interrupt source clear registers

These registers (LCH*x*LCH*y*_INT_REQ_CL) are used to clear interrupt sources. Only data of bits to which 1 is written is updated.

Separate source registers are provided for the AHB write side (*_W_*) and the AHB read side (*_R_*).

- LCH0LCH3_INT_REQ_CL: E108_0110H (LCH0 to LCH3)
- LCH4LCH7_INT_REQ_CL: E108_0130H (LCH4 to LCH7)

The bit assignment for LCH0 to LCH3 is shown below. The other channels have the same structure. However, because the timeout interrupt (INT_TIME) occurs only on LCH0, the timeout interrupt bits for other channels are not available (i.e., they are reserved).

31	30	29	28	27	26	25	24
Reserved	P2M_LCH3_INT_ERROR_W_REQ_CL	P2M_LCH3_INT_BLOCK_W_REQ_CL	P2M_LCH3_INT_LENG_W_REQ_CL	Reserved	P2M_LCH3_INT_ERROR_R_REQ_CL	Reserved	

23	22	21	20	19	18	17	16
Reserved	P2M_LCH2_INT_ERROR_W_REQ_CL	P2M_LCH2_INT_BLOCK_W_REQ_CL	P2M_LCH2_INT_LENG_W_REQ_CL	Reserved	P2M_LCH2_INT_ERROR_R_REQ_CL	Reserved	

15	14	13	12	11	10	9	8
Reserved	P2M_LCH1_INT_ERROR_W_REQ_CL	P2M_LCH1_INT_BLOCK_W_REQ_CL	P2M_LCH1_INT_LENG_W_REQ_CL	Reserved	P2M_LCH1_INT_ERROR_R_REQ_CL	Reserved	

7	6	5	4	3	2	1	0
Reserved	P2M_LCH0_INT_ERROR_W_REQ_CL	P2M_LCH0_INT_BLOCK_W_REQ_CL	P2M_LCH0_INT_LENG_W_REQ_CL	P2M_LCH0_INT_TIME_R_REQ_CL	P2M_LCH0_INT_ERROR_R_REQ_CL	Reserved	

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_W_REQ_CL	W	30	0	Clear the P2M LCH3 error interrupt source.
P2M_LCH3_INT_ BLOCK_W_REQ_CL	W	29	0	Clear the P2M LCH3 block interrupt source.
P2M_LCH3_INT_ LENG_W_REQ_CL	W	28	0	Clear the P2M LCH3 length interrupt source.
Reserved	–	27	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH3_INT_ ERROR_R_REQ_CL	W	26	0	Clear the P2M LCH3 error interrupt source.
Reserved	–	25:23	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH2_INT_ ERROR_W_REQ_CL	W	22	0	Clear the P2M LCH2 error interrupt source.
P2M_LCH2_INT_ BLOCK_W_REQ_CL	W	21	0	Clear the P2M LCH2 block interrupt source.
P2M_LCH2_INT_ LENG_W_REQ_CL	W	20	0	Clear the P2M LCH2 length interrupt source.
Reserved	–	19	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH2_INT_ ERROR_R_REQ_CL	W	18	0	Clear the P2M LCH2 error interrupt source.
Reserved	–	17:15	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH1_INT_ ERROR_W_REQ_CL	W	14	0	Clear the P2M LCH1 error interrupt source.
P2M_LCH1_INT_ BLOCK_W_REQ_CL	W	13	0	Clear the P2M LCH1 block interrupt source.
P2M_LCH1_INT_ LENG_W_REQ_CL	W	12	0	Clear the P2M LCH1 length interrupt source.
Reserved	–	11	–	Reserved. If this bit is read, 0 is returned.
P2M_LCH1_INT_ ERROR_R_REQ_CL	W	10	0	Clear the P2M LCH1 error interrupt source.
Reserved	–	9:7	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_INT_ ERROR_W_REQ_CL	W	6	0	Clear the P2M LCH0 error interrupt source.
P2M_LCH0_INT_ BLOCK_W_REQ_CL	W	5	0	Clear the P2M LCH0 block interrupt source.
P2M_LCH0_INT_ LENG_W_REQ_CL	W	4	0	Clear the P2M LCH0 length interrupt source.
P2M_LCH0_INT_ TIME_R_REQ_CL	W	3	0	Clear the P2M LCH0 timeout interrupt source.
P2M_LCH0_INT_ ERROR_R_REQ_CL	W	2	0	Clear the P2M LCH0 error interrupt source.
Reserved	–	1:0	–	Reserved. If these bits are read, 0 is returned for each bit.

Remark 0: No operation (retains the current setting), 1: Clear the interrupt source

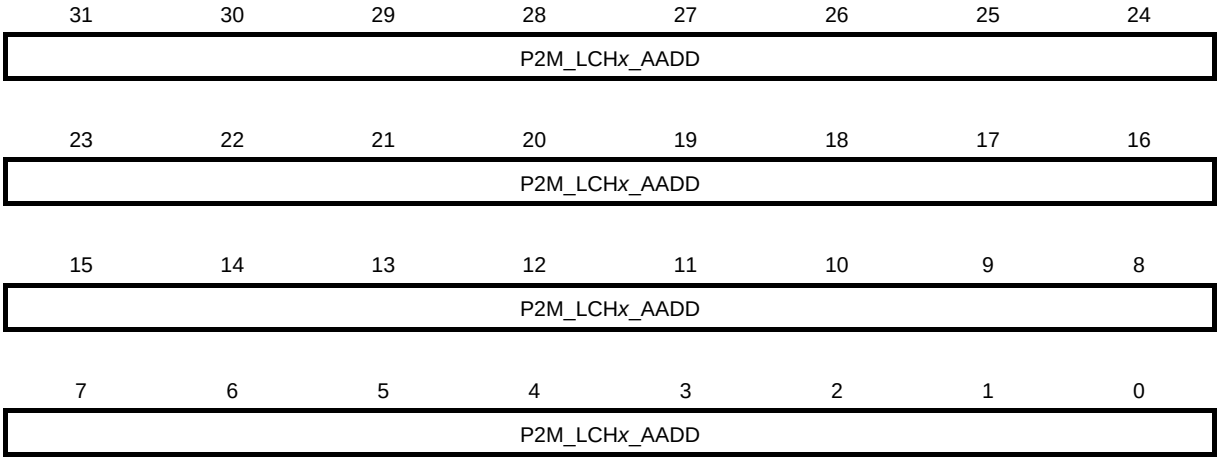
2.3.9 P2M LCH*x* parameter setting registers

These registers are used to specify the settings for each P2M logical channel. The letter *x* in LCH*x* represents a channel number, which ranges from 0 to 7.

(1) P2M LCH*x* source address registers

These registers (LCH*x*_AADD) are used to specify the address from which to send data in bytes.

- LCH0_AADD: E108_1000H (LCH0)
- LCH1_AADD: E108_1100H (LCH1)
- LCH2_AADD: E108_1200H (LCH2)
- LCH3_AADD: E108_1300H (LCH3)
- LCH4_AADD: E108_1400H (LCH4)
- LCH5_AADD: E108_1500H (LCH5)
- LCH6_AADD: E108_1600H (LCH6)
- LCH7_AADD: E108_1700H (LCH7)

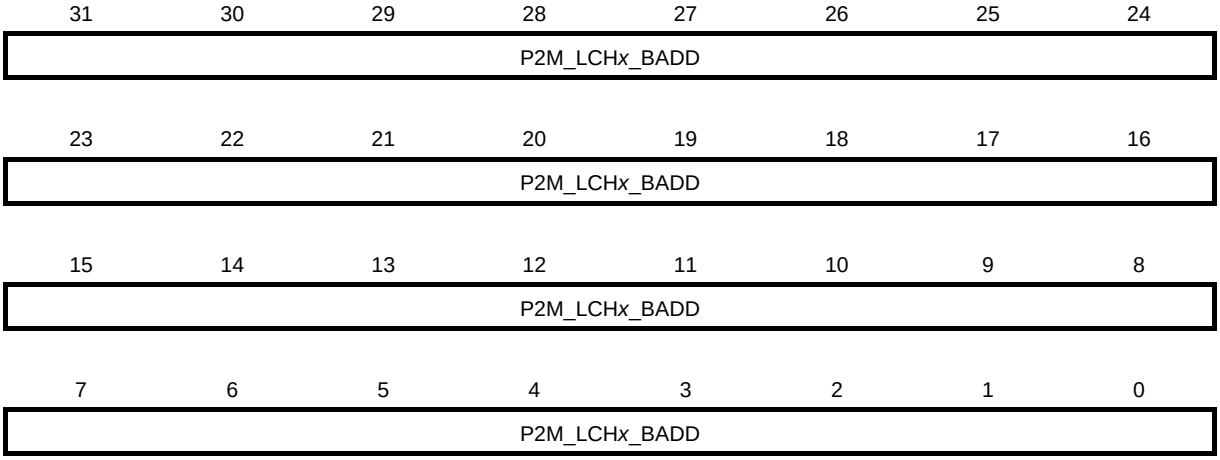


Name	R/W	Bit No.	After Reset	Description
P2M_LCH*x*_AADD	R/W	31:0	0000_0000H	Specify the address from which to send data.

(2) P2M LCH*x* destination address registers

These registers (LCH*x*_BADD) are used to specify the address starting at which to store the received data in bytes.

- LCH0_BADD: E108_1020H (LCH0)
- LCH1_BADD: E108_1120H (LCH1)
- LCH2_BADD: E108_1220H (LCH2)
- LCH3_BADD: E108_1320H (LCH3)
- LCH4_BADD: E108_1420H (LCH4)
- LCH5_BADD: E108_1520H (LCH5)
- LCH6_BADD: E108_1620H (LCH6)
- LCH7_BADD: E108_1720H (LCH7)

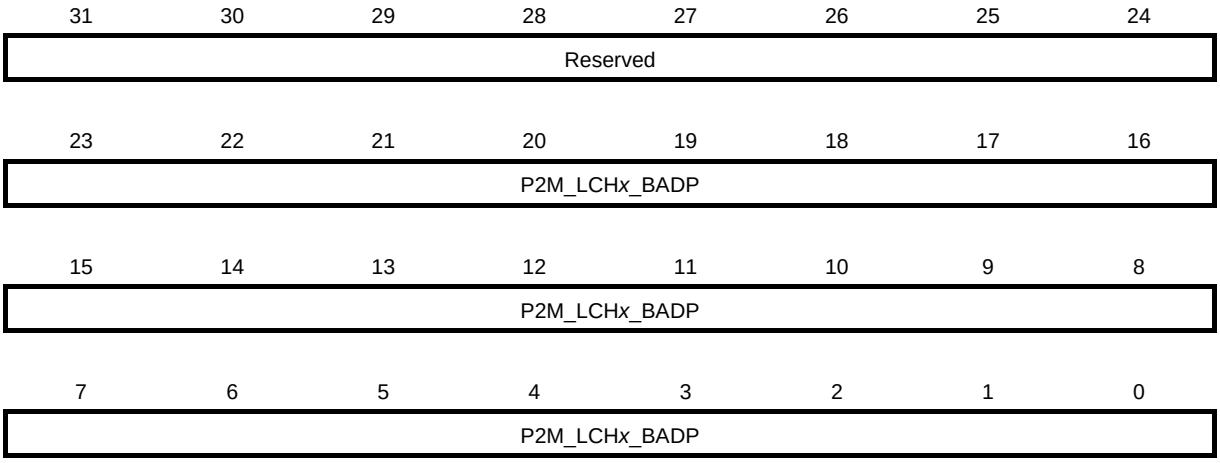


Name	R/W	Bit No.	After Reset	Description
P2M_LCH*x*_BADD	R/W	31:0	0000_0000H	Specify the address starting at which to store the data received on LCH*x* by way of the P2M transfer.

(3) P2M LCH*x* destination address pointer registers

These registers (LCH*x*_BADP) indicate the address where the received data is being stored.

- LCH0_BADP: E108_1024H (LCH0)
- LCH1_BADP: E108_1124H (LCH1)
- LCH2_BADP: E108_1224H (LCH2)
- LCH3_BADP: E108_1324H (LCH3)
- LCH4_BADP: E108_1424H (LCH4)
- LCH5_BADP: E108_1524H (LCH5)
- LCH6_BADP: E108_1624H (LCH6)
- LCH7_BADP: E108_1724H (LCH7)

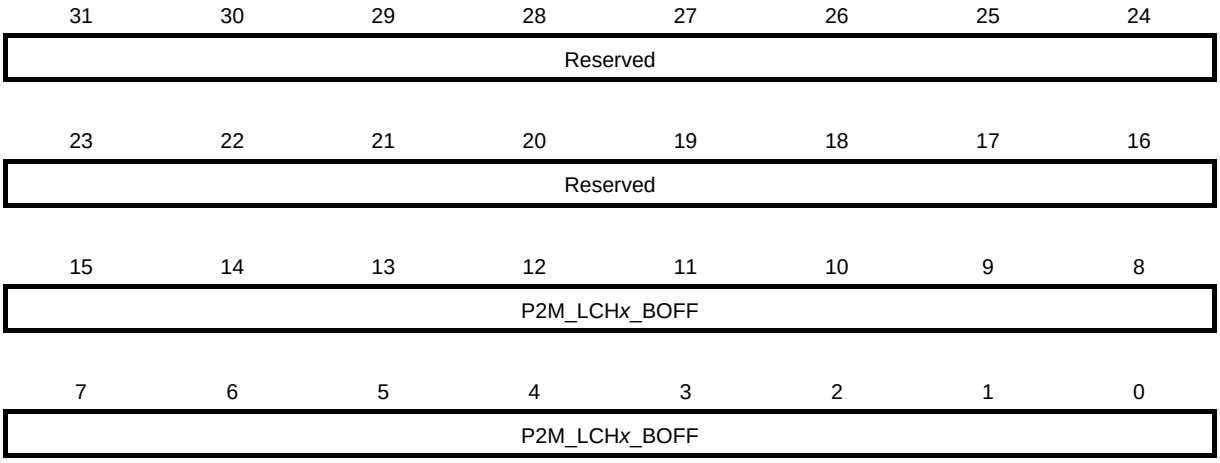


Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_BADP	R	23:0	00_0000H	Indicates the address where the data received on LCH*x* by way of the P2M transfer is being stored.

(4) P2M LCH*x* destination address offset registers

These registers (LCH*x*_BOFF) are used to specify the offset between data blocks to receive in bytes. Up to 65,535 bytes can be specified.

- LCH0_BOFF: E108_1028H (LCH0)
- LCH1_BOFF: E108_1128H (LCH1)
- LCH2_BOFF: E108_1228H (LCH2)
- LCH3_BOFF: E108_1328H (LCH3)
- LCH4_BOFF: E108_1428H (LCH4)
- LCH5_BOFF: E108_1528H (LCH5)
- LCH6_BOFF: E108_1628H (LCH6)
- LCH7_BOFF: E108_1728H (LCH7)



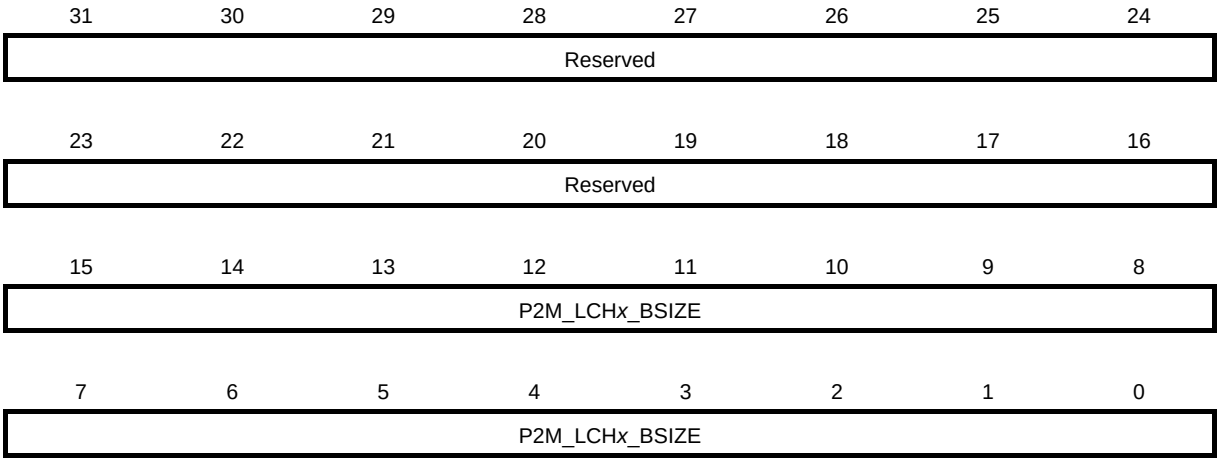
Name	R/W	Bit No.	After Reset	Description
Reserved	R	31:16	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_BOFF	R/W	15:0	0000H	Indicates the offset between data blocks to receive on LCH*x* by way of the P2M transfer in bytes. 00000000_00000000: 0 bytes (no offset) 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

(5) P2M LCH*x* destination block size registers

These registers (LCH*x*_BSIZE) are used to specify the size of data blocks to receive in bytes.

Up to 65,535 bytes can be specified.

- LCH0_BSIZE: E108_102CH (LCH0)
- LCH1_BSIZE: E108_112CH (LCH1)
- LCH2_BSIZE: E108_122CH (LCH2)
- LCH3_BSIZE: E108_132CH (LCH3)
- LCH4_BSIZE: E108_142CH (LCH4)
- LCH5_BSIZE: E108_152CH (LCH5)
- LCH6_BSIZE: E108_162CH (LCH6)
- LCH7_BSIZE: E108_172CH (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:16	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_BSIZE	R/W	15:0	0000H	Specify the size of data blocks to receive on LCH*x* by way of the P2M transfer. 00000000_00000000: Setting prohibited 00000000_00000001: 1 byte 00000000_00000010: 2 bytes (halfword) 00000000_00000011: 3 bytes 00000000_00000100: 4 bytes (1 word) 00000000_00000101: 5 bytes ... 11111111_11111111: 65,535 bytes

Cautions 1. If 0 is specified for one of these registers, a block interrupt might occur continuously. In this case, the normal DMA operation is not guaranteed.

2. To prevent a block interrupt from occurring each time the specified size is transferred, set the block size to the same value as the length. (In this case, a block interrupt and a length interrupt occur simultaneously when a DMA transfer finishes.)

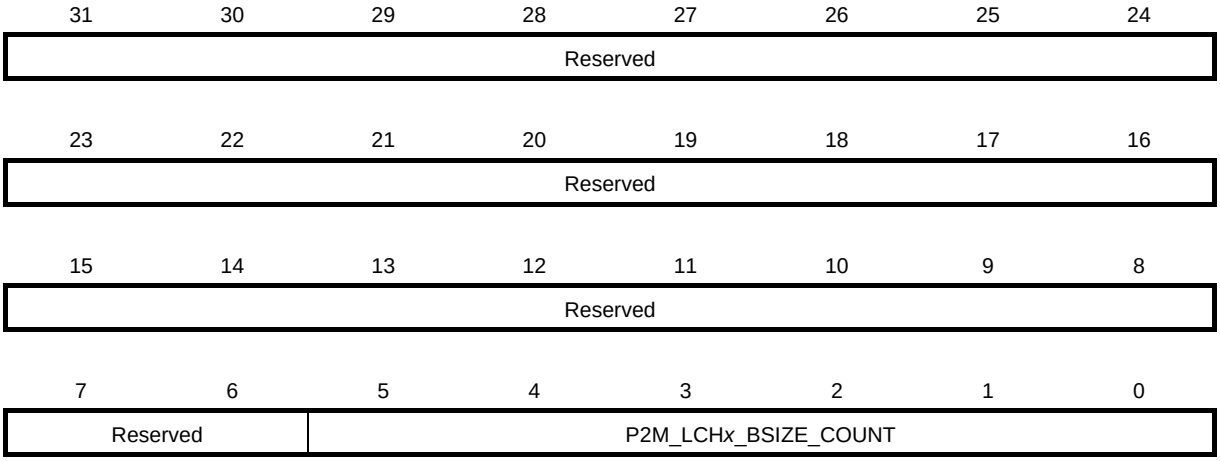
(6) P2M LCH*x* destination block count registers

These registers (LCH*x*_BSIZE_COUNT) work differently when read than when written.

When these registers are written, the number of data blocks transferred per loop during a repeat transfer is set.

When these registers are read, the remaining number of data blocks to receive is read. The number is decremented each time a block transfer ends, and the remaining number is shown in the corresponding register. To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_BSIZE_COUNT: E108_1030H (LCH0)
- LCH1_BSIZE_COUNT: E108_1130H (LCH1)
- LCH2_BSIZE_COUNT: E108_1230H (LCH2)
- LCH3_BSIZE_COUNT: E108_1330H (LCH3)
- LCH4_BSIZE_COUNT: E108_1430H (LCH4)
- LCH5_BSIZE_COUNT: E108_1530H (LCH5)
- LCH6_BSIZE_COUNT: E108_1630H (LCH6)
- LCH7_BSIZE_COUNT: E108_1730H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:6	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_BSIZE_COUNT	R/W	5:0	000000b	When written: Specify the number of data blocks to receive per loop during a repeat transfer. 000000: 1 block 000001: 2 blocks 000010: 3 blocks 000011: 4 blocks 000100: 5 blocks 000101: 6 blocks 000110: 7 blocks 000111: 8 blocks 001000: 9 blocks ... 111111: 64 blocks When read: Indicates the remaining number of data blocks to receive. This number is applied when a DMA transfer starts, and is decremented each time a block transfer ends.

(7) P2M LCH*x* length registers

These registers (LCH*x*_LENG) are used to specify the number of bytes to transfer. Up to 16,777,215 bytes can be specified.

If one of these registers is cleared to 0 while the repeat mode is specified in its P2M LCH*x* mode register, an infinite-length transfer is specified.

Caution Do not clear these registers to 0 if the repeat mode is not specified. Specifying an offset for an infinite-length transfer is prohibited.

- LCH0_LENG: E108_1040H (LCH0)
- LCH1_LENG: E108_1140H (LCH1)
- LCH2_LENG: E108_1240H (LCH2)
- LCH3_LENG: E108_1340H (LCH3)
- LCH4_LENG: E108_1440H (LCH4)
- LCH5_LENG: E108_1540H (LCH5)
- LCH6_LENG: E108_1640H (LCH6)
- LCH7_LENG: E108_1740H (LCH7)

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
P2M_LCH*x*_LENG							

15	14	13	12	11	10	9	8
P2M_LCH*x*_LENG							

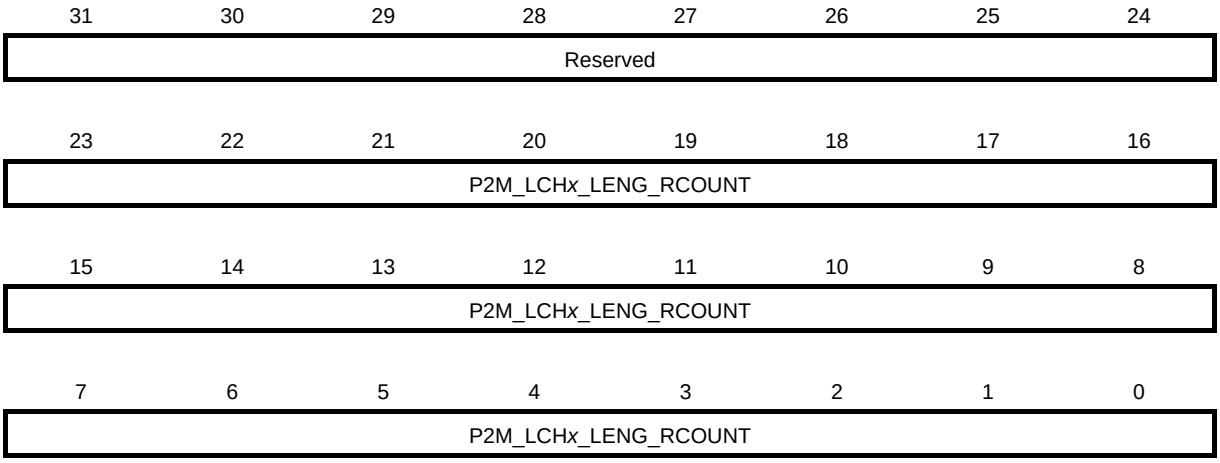
7	6	5	4	3	2	1	0
P2M_LCH*x*_LENG							

Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_LENG	R/W	23:0	00_0000H	Specify the number of bytes to transfer on LCH*x* by way of the P2M transfer. 00000000_00000000_00000000: Infinite-length transfer 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(8) P2M LCH*x* read length count registers

These registers (LCH*x*_LENG_RCOUNT) indicate the remaining amount of data to send. The registers indicate the data amount decremented from the value specified for the P2M LCH*x* length register.

- LCH0_LENG_RCOUNT: E108_1044H (LCH0)
- LCH1_LENG_RCOUNT: E108_1144H (LCH1)
- LCH2_LENG_RCOUNT: E108_1244H (LCH2)
- LCH3_LENG_RCOUNT: E108_1344H (LCH3)
- LCH4_LENG_RCOUNT: E108_1444H (LCH4)
- LCH5_LENG_RCOUNT: E108_1544H (LCH5)
- LCH6_LENG_RCOUNT: E108_1644H (LCH6)
- LCH7_LENG_RCOUNT: E108_1744H (LCH7)

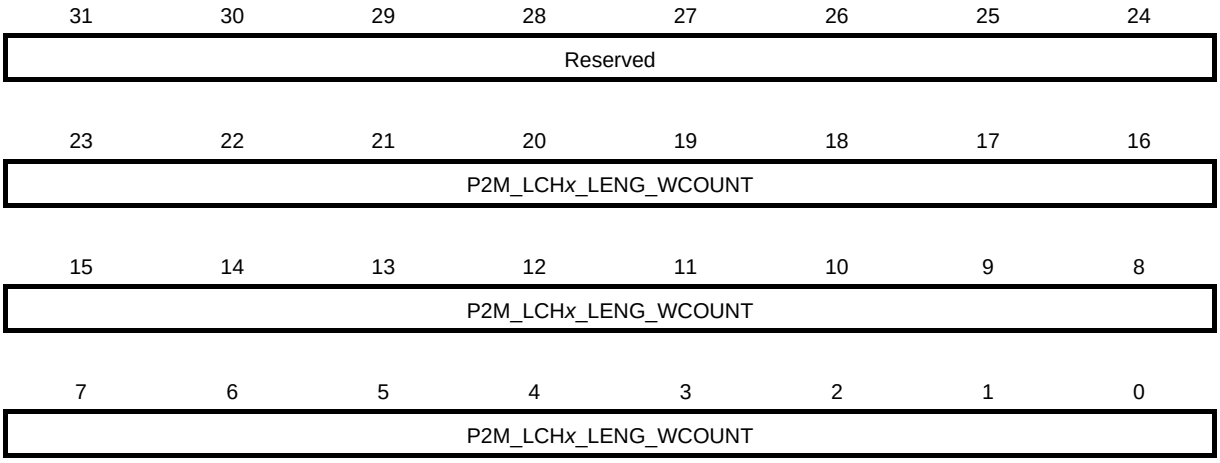


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_LENG_RCOUNT	R	23:0	00_0000H	Specify the number of bytes to send on LCH*x* by way of the P2M transfer. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(9) P2M LCH*x* write length count registers

These registers (LCH*x*_LENG_WCOUNT) indicate the remaining amount of data to receive. The registers indicate the data amount decremented from the value specified for the P2M LCH*x* length register.

- LCH0_LENG_WCOUNT: E108_1048H (LCH0)
- LCH1_LENG_WCOUNT: E108_1148H (LCH1)
- LCH2_LENG_WCOUNT: E108_1248H (LCH2)
- LCH3_LENG_WCOUNT: E108_1348H (LCH3)
- LCH4_LENG_WCOUNT: E108_1448H (LCH4)
- LCH5_LENG_WCOUNT: E108_1548H (LCH5)
- LCH6_LENG_WCOUNT: E108_1648H (LCH6)
- LCH7_LENG_WCOUNT: E108_1748H (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_LENG_WCOUNT	R	23:0	00_0000H	Specify the number of bytes to receive on LCH*x* by way of the P2M transfer. 00000000_00000000_00000000: 0 bytes 00000000_00000000_00000001: 1 byte 00000000_00000000_00000010: 2 bytes 00000000_00000000_00000011: 3 bytes 00000000_00000000_00000100: 4 bytes ... 11111111_11111111_11111111: 16,777,215 bytes

(10) P2M LCH*x* mode registers

These registers (LCH*x*_MODE) are used to set up transfers.

To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

- LCH0_MODE: E108_1050H (LCH0)
- LCH1_MODE: E108_1150H (LCH1)
- LCH2_MODE: E108_1250H (LCH2)
- LCH3_MODE: E108_1350H (LCH3)
- LCH4_MODE: E108_1450H (LCH4)
- LCH5_MODE: E108_1550H (LCH5)
- LCH6_MODE: E108_1650H (LCH6)
- LCH7_MODE: E108_1750H (LCH7)

31 30	29 28	27 26	25 24
P2M_LCH*x*_MODE_ ENDI_W_HH	P2M_LCH*x*_MODE_ ENDI_W_HL	P2M_LCH*x*_MODE_ ENDI_W_LH	P2M_LCH*x*_MODE_ ENDI_W_LL

23 22	21 20	19 18	17 16
P2M_LCH*x*_MODE_ ENDI_R_HH	P2M_LCH*x*_MODE_ ENDI_R_HL	P2M_LCH*x*_MODE_ ENDI_R_LH	P2M_LCH*x*_MODE_ ENDI_R_LL

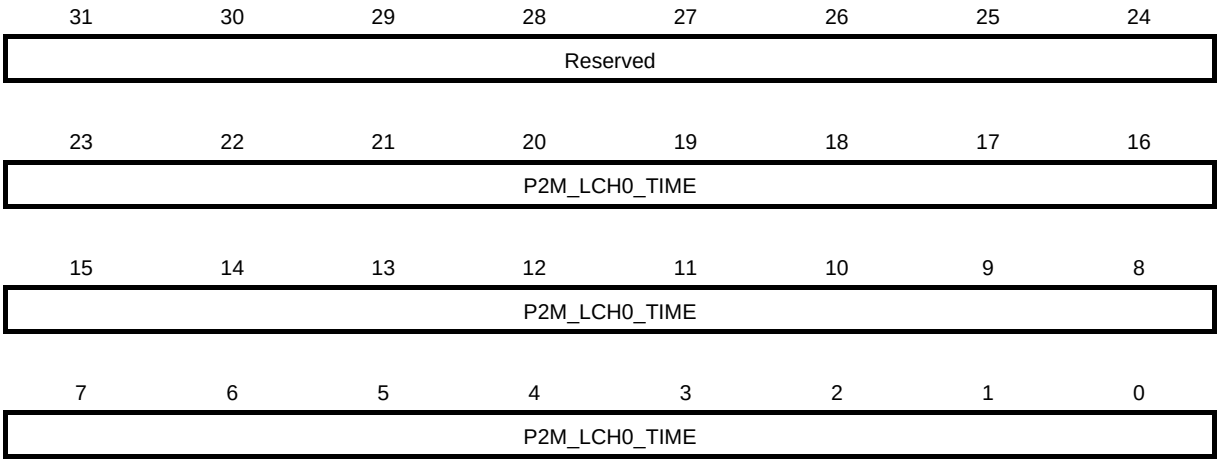
15 14 13 12 11 10 9	8
Reserved	P2M_LCH*x*_ BMODE_ REPEAT

7 6 5 4 3 2 1 0
Reserved

Name	R/W	Bit No.	After Reset	Description
P2M_LCH*x*_MODE_ ENDI_W_HH	R/W	31:30	E4H	Specify the byte lane for writing data to the transfer destination. 00: Byte 0 01: Byte 1 02: Byte 2 03: Byte 3
P2M_LCH*x*_MODE_ ENDI_W_HL	R/W	29:28		
P2M_LCH*x*_MODE_ ENDI_W_LH	R/W	27:26		
P2M_LCH*x*_MODE_ ENDI_W_LL	R/W	25:24		
P2M_LCH*x*_MODE_ ENDI_R_HH	R/W	23:22	E4H	Specify the byte lane for reading data from the transfer source. 00: Byte 0 01: Byte 1 02: Byte 2 03: Byte 3
P2M_LCH*x*_MODE_ ENDI_R_HL	R/W	21:20		
P2M_LCH*x*_MODE_ ENDI_R_LH	R/W	19:18		
P2M_LCH*x*_MODE_ ENDI_R_LL	R/W	17:16		
Reserved	R	15:9	00H	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH*x*_BMODE_ REPEAT	R/W	8	0	Specify whether to use the repeat mode on the destination side. 0: Do not specify the repeat mode. 1: Specify the repeat mode.
Reserved	R	7:0	00H	Reserved. If these bits are read, 0 is returned for each bit.

(11) P2M LCH0 timer register

This register (LCH0_TIME: E108_1054H) is used to specify the wait time for the next request to receive. If no DMA request is issued for a specific period and the time specified in this register expires, the current DMA transfer is terminated. Up to 24 bits can be set.

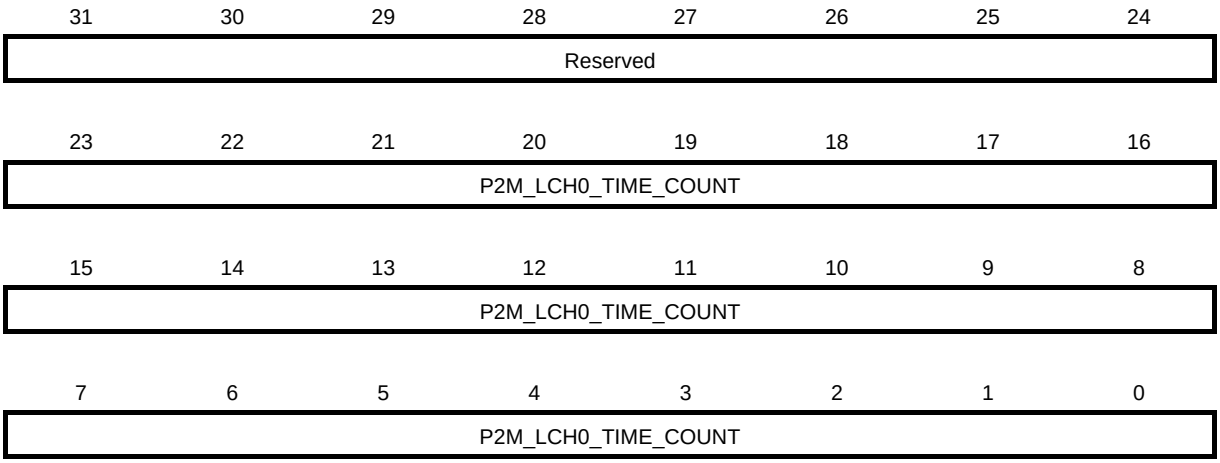


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_TIME	R/W	23:0	00_0000H	Specify the time allowed to elapse before a P2M transfer on LCH0 times out.

Caution Be sure to supply a timer clock (DMA_TCLK) when using a timer. Otherwise, DMA clock control might malfunction.

(12) P2M LCH0 timer count register

This register (LCH0_TIME_COUNT: E108_1058H) is used to count down to the time when the next request is received. The timer is decremented in units of the DMA_TCLK cycle output from the SMU. When the count reaches 0, the current DMA transfer is terminated.

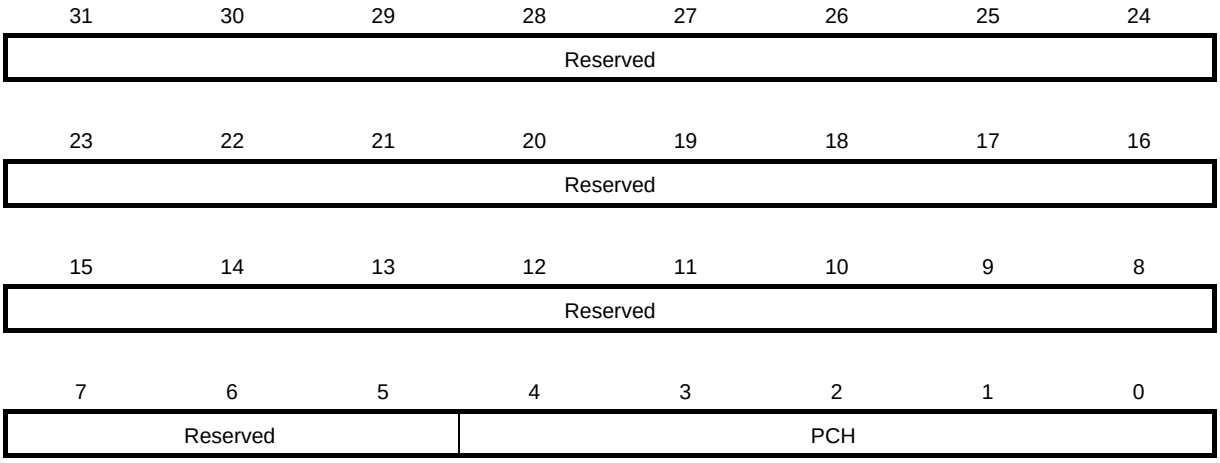


Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:24	–	Reserved. If these bits are read, 0 is returned for each bit.
P2M_LCH0_TIME_ COUNT	R	23:0	00_0000H	Indicates the time allowed to elapse before a P2M transfer on LCH0 times out. DMA_TCLK is used for timer counting.

(13) P2M LCH*x* physical channel registers

These registers (LCHx_PCH) select the module that sends the DMA request signal (DMARQ).

- LCH0_PCH: E108_105CH (LCH0)
- LCH1_PCH: E108_115CH (LCH1)
- LCH2_PCH: E108_125CH (LCH2)
- LCH3_PCH: E108_135CH (LCH3)
- LCH4_PCH: E108_145CH (LCH4)
- LCH5_PCH: E108_155CH (LCH5)
- LCH6_PCH: E108_165CH (LCH6)
- LCH7_PCH: E108_175CH (LCH7)



Name	R/W	Bit No.	After Reset	Description
Reserved	–	31:5	–	Reserved. If these bits are read, 0 is returned for each bit.
PCH	R/W	4:0	00H	Select the module that sends the DMA request signal (DMARQ). 0 to 5: Reserved 6: UART0 7: UART1 8: UART2 9: UART3 10: SIO0 11: SIO1 12: SIO2 13: SIO3 14: SIO4 15: SIO5 16 and 17: Reserved

3. Description of Functions

3.1 Overview of DMA Transfer

DMA transfers are classified into the types shown in the following table. A separate physical channel is provided for each transfer type.
Physical channel 1 (PCH1) does not exist.

Table 3-1. Transfer Types and Number of Channels

Transfer Type	Memory-to-Memory	Memory-to-Peripheral	Peripheral-to-Memory
Physical channel	PCH0	PCH2	PCH3
Number of logical channels	8	8	8
Two-dimensional transfer	○	△Note	△Note
Reverse transfer	○	×	×

Note Two-dimensional transfer is supported only on the memory side.

Remark ○: Supported, △: Supported with certain conditions, ×: Not supported

3.2 Physical Channel Functions

3.2.1 Memory-to-memory transfer (on PCH0)

Memory-to-memory transfers send data among areas AB0, MEMC and SRC in various combinations, except for transfers that involve overwriting (such as right-scrolling of an image). The minimum transfer unit is one byte. PCH0 has eight logical channels (LCH0 to LCH7) and they are arbitrated on a per-transaction basis in a round-robin fashion.

Table 3-2. Memory-to-Memory Transfer Combinations

Destination / Source	AB0 Area	SRC Area	MEMC Area
AB0 area	○	○	○
SRC area	×	×	×
MEMC area	○	○	○

Remark ○: Available, ×: Not available

As shown in the table above, the SRC area cannot be specified as the source address.

(1) LCH arbitration control

(a) Read control block

To select a logical channel subject to transfer, round-robin arbitration is performed among those logical channels for which starting a DMA transfer is directed by using the DMA start control register (CONT).

Note that a logical channel cannot participate in arbitration if the FIFO buffer assigned to that channel does not have 68 bytes (64 bytes + 4 bytes) or more of free space.

(b) Write control block

As with the read control block, to select a logical channel subject to transfer, round-robin arbitration is performed among those logical channels for which starting a DMA transfer is directed by using the DMA start control register. Note that a logical channel can participate in arbitration if the FIFO buffer assigned to that channel has 64 bytes or more of valid data or if the last data transfer completion signal sent from the read control block has been asserted indicating that the specified length of data has been transferred.

(2) AHB access control

(a) Read control block

To enhance the AHB transfer efficiency, only 8-beat incrementing bursts (INCR8) and 16-beat incrementing bursts (INCR16) are used. The transfer start address for INCR16 must be on a 64-byte boundary, and the transfer start address for INCR8 must be on a 32-byte address boundary. If 32 bytes or less are to be transferred, an INCR8 burst including null data is sent. If less than 32 bytes are to be transferred, the data is transferred using an INCR8 burst and the DMA controller stores only necessary received data in the FIFO buffer.

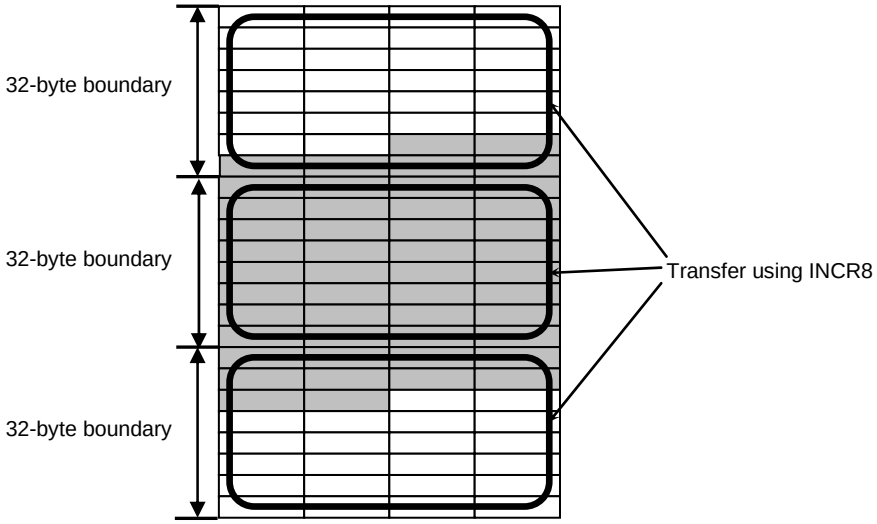
The transfer burst size is selected according to the rules described in Table 3-3.

Table 3-3. Transfer Burst Size Selection (Read Control Block)

Transfer Start Address	Transfer Burst Size
64-byte boundary	INCR16 (16-beat burst)
32-byte boundary	INCR8 (8-beat burst)
Other	INCR8 (8-beat burst), unnecessary data is discarded.

An overview of an AHB read transfer is shown below.

Figure 3-1. Selecting Transfer Burst Size (for AHB Read Transfer)



Remark : Valid transferred data

(b) Write control block

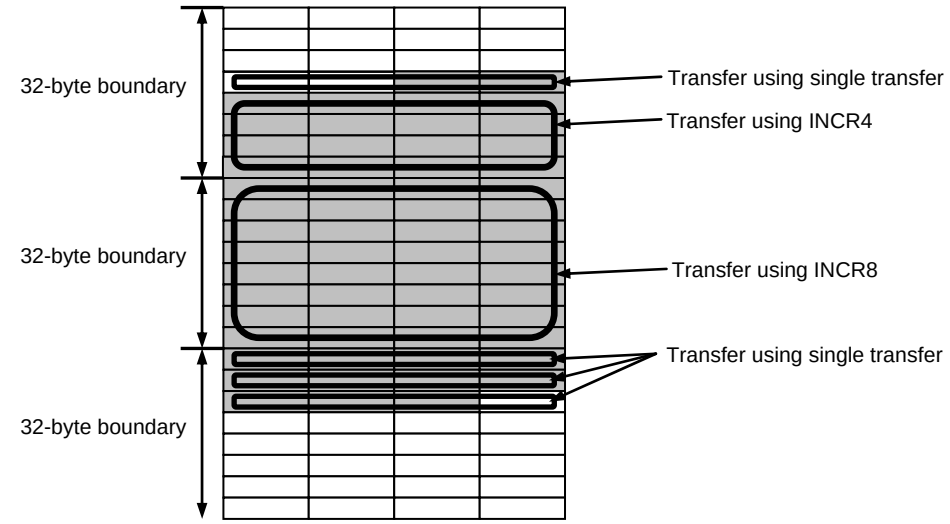
For AHB transfers handled by the write control block, 16-beat incrementing bursts (INCR16) are generally used. The transfer start address for INCR16 must be on a 64-byte boundary. If the transfer start address is not on a 64-byte boundary, the data is segmented so that the largest possible transfer burst size is selected according to the rules described in Table 3-4.

Table 3-4. Transfer Burst Size Selection (Write Control Block)

Transfer Start Address	Transfer Burst Size
64-byte boundary	INCR16 (16-beat burst)
32-byte boundary	INCR8 (8-beat burst)
16-byte boundary	INCR4 (4-beat burst)
Other	Single (including byte/halfword transfer)

An overview of an AHB write transfer is shown below.

Figure 3-2. Selecting Transfer Burst Size (for AHB Write Transfer)



Remark : Valid transferred data

(3) Address control

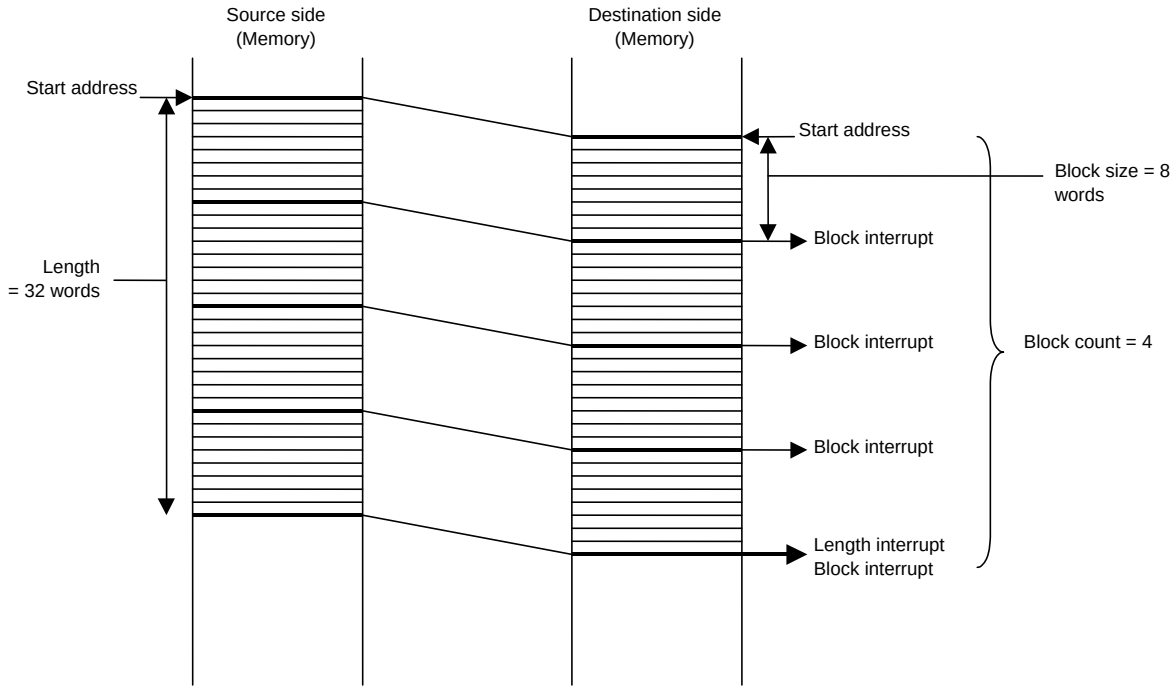
(a) One-dimensional memory-to-memory transfer

Memory-to-memory transfers support offsets for both the source and destination sides.

Figure 3-3 shows an example of a one-dimensional memory-to-memory transfer.

To continue a transfer, the address is incremented from the start address on both the source and destination sides. When the specified block size or length of data has been transferred, an interrupt request signal is output. The types of block and length interrupts are managed separately.

Figure 3-3. Example of One-Dimensional Memory-to-Memory Transfer



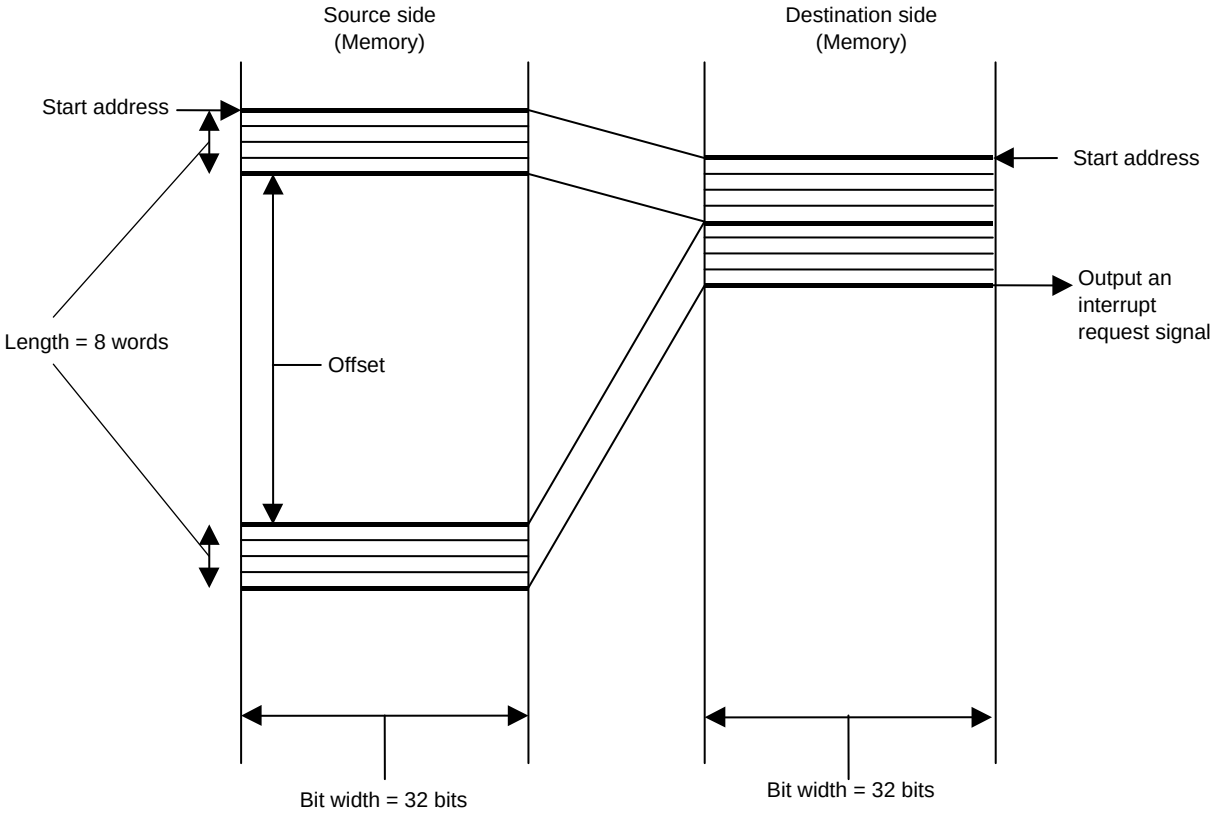
Remark To prevent a block interrupt from occurring each time the specified size is transferred, set the block size to the same value as the length. In this case, a length interrupt and a block interrupt occur simultaneously when DMA transfer ends.

(b) Two-dimensional memory-to-memory transfer

Figure 3-4 shows an example of a two-dimensional memory-to-memory transfer.

On the source side, the start address is incremented by the block size, the offset is added to the incremented address value, and then another block of data is accessed. On the destination side, addresses are generated in an incremental manner. When the specified amount of data has been transferred, an interrupt request signal is output. Figure 3-4 shows an example when offsets are not used on the destination side, but transfers with the block size or offset specified are also possible.

Figure 3-4. Example of Two-Dimensional Memory-to-Memory Transfer



(c) Memory-to-memory repeat transfer

Figure 3-5 shows an example of a memory-to-memory transfer for which the same destination addresses are repeatedly used.

Data is sent in an incremental manner beginning with the start address. When the specified amount of data has been sent, a length interrupt is issued (a block interrupt can be issued as well).

When the specified amount of data has been received, a block interrupt is issued and the start address is reset. (If the number of data blocks is specified by using the block count register, the start address is reset as soon as the specified number of data blocks is transferred.)

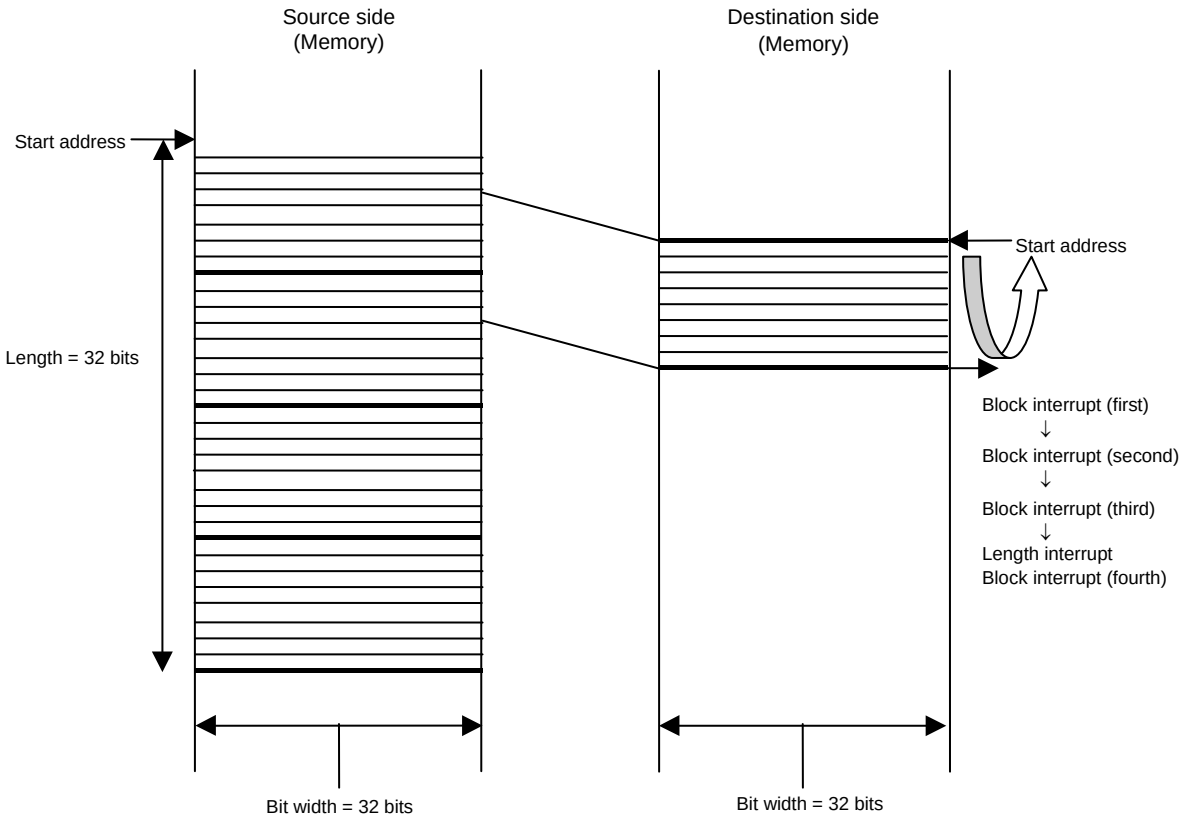
When the specified amount of data has been sent, an interrupt request signal is output. To perform a repeat transfer, the offset must be set to 0 (one-dimensional transfer).

Figure 3-5 shows an example when the source side does not run in the repeat transfer mode, but it is also possible to specify the repeat mode for the source side, as for the destination side.

If the length is set to 0 when the repeat transfer mode is specified, the length of data to transfer is infinite.

Parameters for repeat transfers can be specified only in words.

Figure 3-5. Example of Memory-to-Memory Repeat Transfer



(d) Memory-to-memory reverse transfer

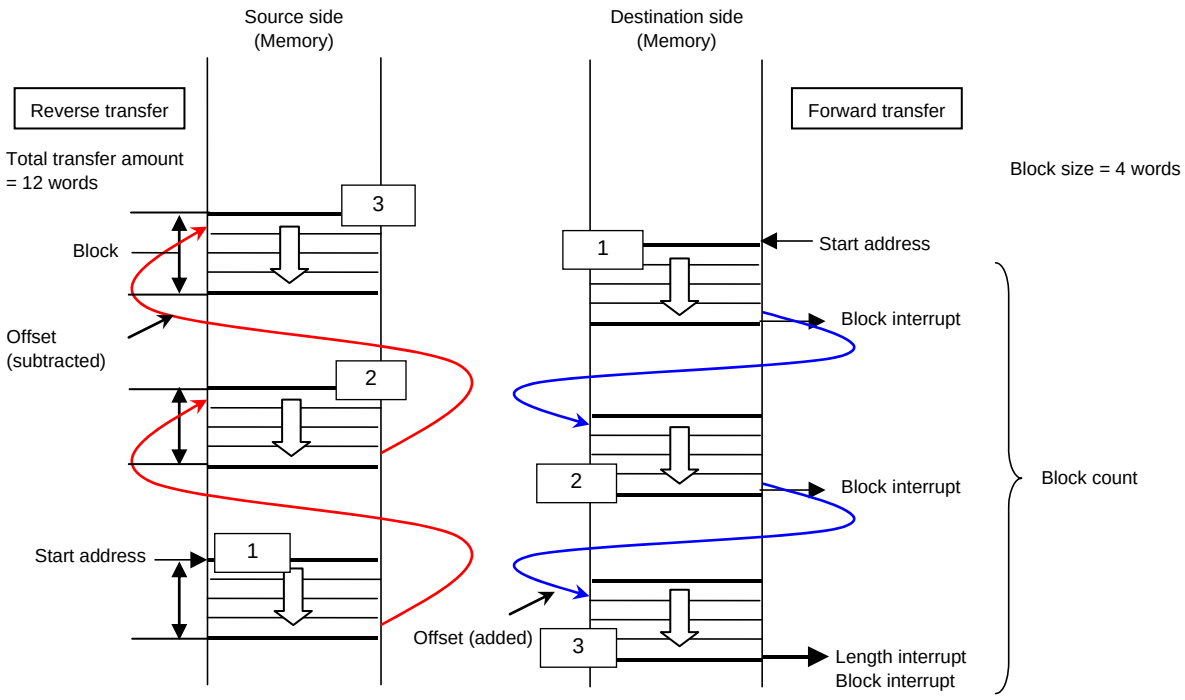
Memory-to-memory transfers support reverse transfer in block units, separately for the source and destination sides. (A reverse transfer is implemented by subtracting an offset from the addresses.)

Figure 3-6 shows an example of a transfer for which the source side performs a reverse transfer and the destination side performs a forward transfer.

The start address is incremented by the block size, the offset is subtracted from the address value, and then the specified number of data blocks are sent again. On the destination side, the offset is added to the address.

Figure 3-6 shows an example of a transfer for which the destination side performs a forward transfer, but a transfer for which the destination side performs a reverse transfer or both sides perform reverse transfers is also available.

Figure 3-6. Example of Memory-to-Memory Reverse Transfer



(4) Data control

(a) Read control block

Between the read control block and the write control block is a built-in four-entry FIFO buffer holding up to 256 bytes of data per entry.

Even if valid data is not in word units, the read control block reads the data, pads it with null data to change it to word units, and then stores the data in the FIFO buffer. To inform the write control block about the valid data bytes written to the FIFO buffer, the read control block outputs the value added to the FIFO read pointer to the write control block.

Table 3-5. Transfer Type and FIFO Read Pointer Addition Values

Transfer Start Address		Transfer Burst Size	FIFO Read Pointer Addition Value
64-byte boundary		INCR16	0
32-byte boundary		INCR8	0
Other	Word boundary	INCR8	0
	SA[1:0] = 3H	INCR8	+3
	SA[1:0] = 2H	INCR8	+2
	SA[1:0] = 1H	INCR8	+1

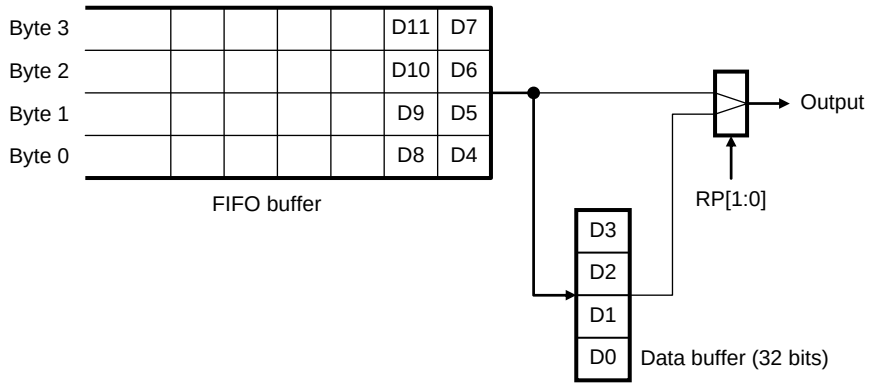
Remark SA: Source address

In addition, when the specified amount of data has been sent, the read control block outputs a transfer completion signal to the write control block. In response to this signal, the write control block writes all the data remaining in the FIFO buffer.

(b) Write control block

The write control block reads necessary data from the FIFO buffer starting from the address indicated by the read pointer of the logical channel entry selected by arbitration, and then writes it to the destination specified using a parameter register. To transfer data in bytes, data must be aligned.

Endian conversion is provided for both read and write control blocks.



WP[1:0]	RP[1:0]	Output [31:24]	Output [23:16]	Output [15:8]	Output [7:0]	Data Size
00	00	D3	D2	D1	D0	Word
00	01	D4	D3	D2	D1	Word
00	10	D5	D4	D3	D2	Word
00	11	D6	D5	D4	D3	Word
01	00	–	–	D0	–	Byte
		D2	D1	–	–	Halfword
01	01	–	–	D1	–	Byte
		D3	D2	–	–	Halfword
01	10	–	–	D2	–	Byte
		D4	D3	–	–	Halfword
01	11	–	–	D3	–	Byte
		D5	D4	–	–	Halfword
10	00	D1	D0	–	–	Halfword
10	01	D2	D1	–	–	Halfword
10	10	D3	D2	–	–	Halfword
10	11	D4	D3	–	–	Halfword
11	00	D0	–	–	–	Byte
11	01	D1	–	–	–	Byte
11	10	D2	–	–	–	Byte
11	11	D3	–	–	–	Byte

Remark WP: Write pointer, RP: FIFO read pointer

3.2.2 Memory-toperipheral transfer (on PCH2)

Memory-to-peripheral transfers send data in response to request signals sent from each module.
Physical channel 2 (PCH2) is a channel dedicated to memory-to-peripheral transfers. PCH2 has eight logical channels (LCH0 to LCH7). These logical channels are arbitrated on a burst transfer size basis in a round-robin fashion.

(1) LCH arbitration control

(a) Read control block

The same processing as for PCH0 is performed.

(b) Write control block

To select a logical channel subject to transfer, round-robin arbitration is performed among those logical channels for which starting a DMA transfer is directed by using the DMA start control register and for which the transfer request signal from a peripheral is active. Note that a logical channel can participate in arbitration if the FIFO buffer assigned to that channel has 4 bytes or more of valid data.

(2) AHB access control

(a) Read control block

The same processing as for PCH0 is performed. Note that only INCR8 is used as the transfer burst size.

(b) Write control block

The write control block only uses single transfers for AHB access. The transfer bit width (8 bits, 16 bits, or 32 bits) can be selected according to the specifications of the peripheral. Specify the transfer bit width for each logical channel using the mode register, according to the transfer bit width supported by each peripheral.

(3) Address control

(a) Read control block

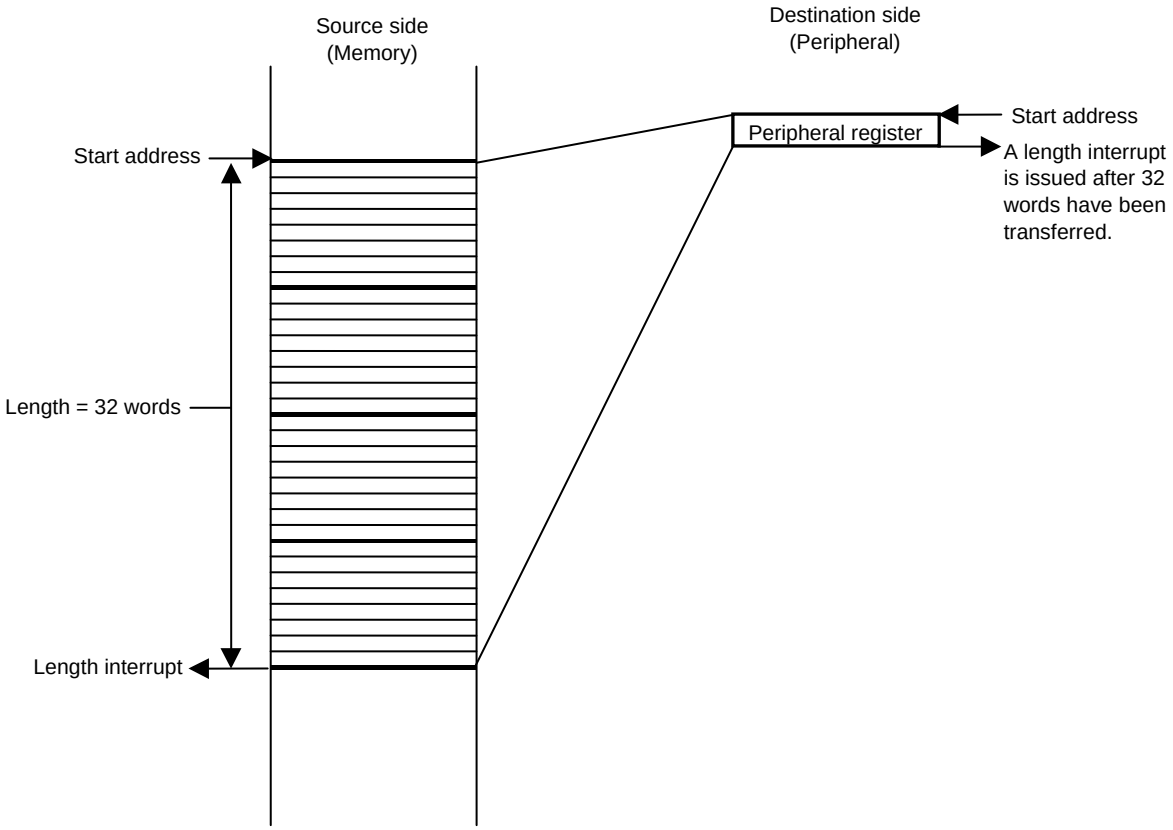
The same processing as for PCH0 is performed.

(b) Write control block

On the peripheral side, the value specified for the start address register is not updated; that is, the address is fixed. On the memory side, the value specified for the start address register is incremented. When the specified length of data has been sent, an interrupt request signal is output. It is also possible to specify a block size and output an interrupt on a per-block size basis, as shown in the examples of the memory-to-memory transfer.

Figure 3-7 shows an example of a memory-to-peripheral transfer.

Figure 3-7. Example of Memory-to-Peripheral Transfer



(4) Data control

(a) Read control block

Between the read control block and the write control block is a built-in 12-entry FIFO buffer holding up to 128 bytes of data per entry.

The basic control processing is the same as that for PCH0.

(b) Write control block

The write control block reads necessary data from the FIFO buffer and writes it to the destination specified by using a parameter register. FIFO read pointer control depends on the bus width on the peripheral side.

Table 3-6. Data Alignment for PCH2 Write Control Block

Peripheral Transfer Bus Width	RP	Byte Lane Selected	RP Addition Value
8 bits	00	n	+1
	01	n + 1	
	10	n + 2	
	11	n + 3	
16 bits	00	n	+2
	01	n + 1	
	10	n + 2	
	11	n + 3	
32 bits	00	n	+4
	01	n + 1	
	10	n + 2	
	11	n + 3	

(5) Timeout function

See **3.2.3 (5) Timeout function**.

3.2.3 Peripheral-to-memory transfer (on PCH3)

Peripheral-to-memory transfers send data in response to request signals sent from each module.
Physical channel 3 (PCH3) is a channel dedicated to peripheral-to-memory transfer. PCH3 has eight logical channels (LCH0 to LCH7). These logical channels are arbitrated on a single transfer size basis in a round-robin fashion.

(1) LCH arbitration control

(a) Read processing

To select a logical channel subject to transfer, round-robin arbitration is performed among those logical channels for which starting a DMA transfer is directed by using the DMA start control register and for which the transfer request signal from the peripheral is active. Note that a logical channel can participate in arbitration if the 16-entry entry buffer has one or more free areas.

(b) Write processing

No arbitration is performed, but entries are processed in the order they are registered in the entry buffer.

(2) AHB access control

(a) Read processing

The read control block uses single transfers for AHB access. The transfer bit width (8 bits, 16 bits, or 32 bits) can be selected according to the specifications of the peripheral. The transfer bit width can be specified using the mode register.

(b) Write processing

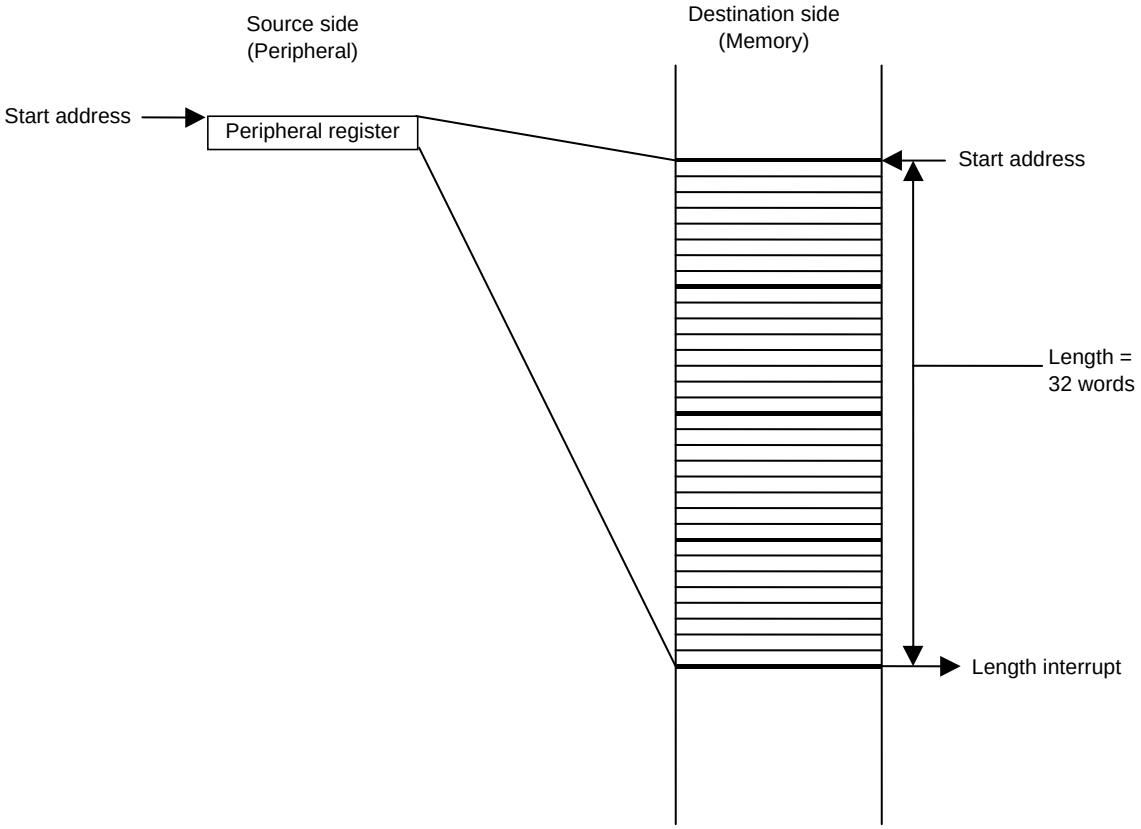
The write control block uses single transfers for AHB access. The transfer bit width applied when writing data to memory can be changed according to the entry in the entry buffer.

(3) Address control

On the peripheral side, the value specified for the start address register is not updated; that is, the address is fixed. On the memory side, the value specified for the start address register is incremented. When the specified length of data has been sent, an interrupt request signal is output. It is also possible to specify a block size and output an interrupt on a per-block size basis, as shown in the examples of the memory-to-memory transfer.

Figure 3-8 shows an example of a peripheral-to-memory transfer.

Figure 3-8. Example of Peripheral-to-Memory Transfer



(4) Data control

(a) Read control block

Between the read control block and the write control block is a built-in 16-entry entry buffer holding up to 4 bytes of data per entry.

The read control block consumes an entry per transaction to write the data read through a single transfer to the entry buffer. At the same time, the read control block saves the entry about the written data to the entry buffer. The items stored in the entry information register are the logical channel number and the number of valid bytes.

(b) Write control block

The write control block retrieves the transferred data in the order of entry and passes the LCH number and the number of valid bytes, stored in the entry information register, to the AHB access control block.

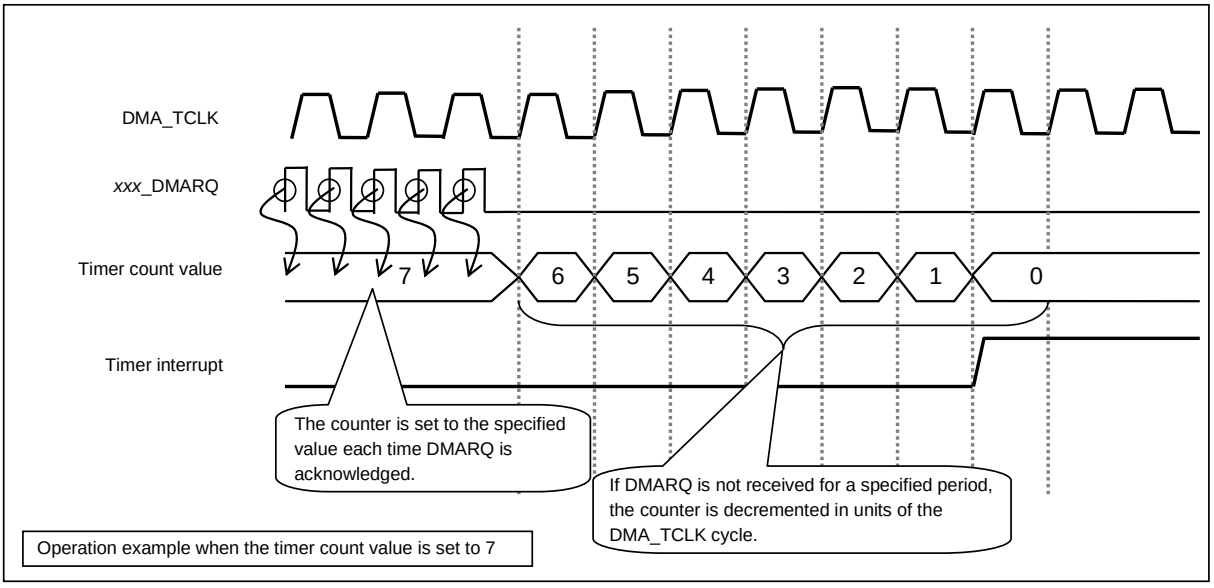
(5) Timeout function

A timeout function is provided for transfers between memory and UART interface, which is used when no DMA transfer request is sent from the peripheral interface for a specific period. To enable this function, use the mode register to enable the timer and specify the timeout time for the timer register. This timeout function is intended exclusively for UART0 to UART2.

The timer starts counting when the first DMA transfer request (DMARQ) is received after DMA starts.

When DMARQ is received again, the counter is set to the timer register value and decrementing starts.

Figure 3-9. Example of Timer Function



3.3 Endian Conversion

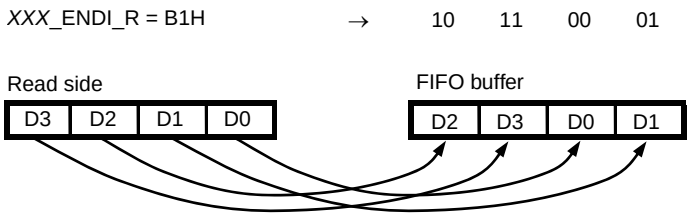
The DMA controller can perform byte endian conversion for the read and write control blocks.
Parameters for using endian conversion must be specified in words. If endian conversion is performed with parameters specified in other units, the first 4 bytes and the last 4 bytes of each transfer block might be garbled. Transferring under the conditions described in **3.3.3 Exception for endian conversion settings**, however, is possible as an exception.

3.3.1 Endian conversion on read control block

Byte lane of data to be captured into a data buffer (FIFO) can be selected for each byte.

- Data read from byte 0 → Selectable from byte 0, 1, 2 or 3 of AHB read data
- Data read from byte 1 → Selectable from byte 0, 1, 2 or 3 of AHB read data
- Data read from byte 2 → Selectable from byte 0, 1, 2 or 3 of AHB read data
- Data read from byte 3 → Selectable from byte 0, 1, 2 or 3 of AHB read data

Figure 3-10. Example of Basic Endian Conversion in Read Control Block



3.3.2 Endian conversion on write control block

Byte lane for peripherals to be written can be selected for each byte.

- Data written to byte 0 → Selectable from byte 0, 1, 2 or 3 of write data
- Data written to byte 1 → Selectable from byte 0, 1, 2 or 3 of write data
- Data written to byte 2 → Selectable from byte 0, 1, 2 or 3 of write data
- Data written to byte 3 → Selectable from byte 0, 1, 2 or 3 of write data

Figure 3-11. Example of Basic Endian Conversion in Write Control Block

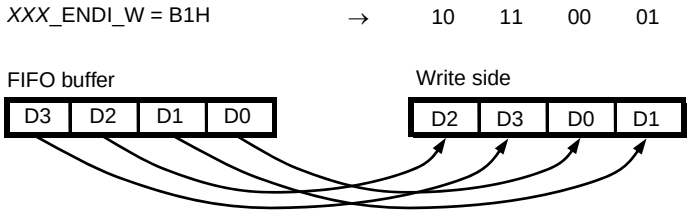
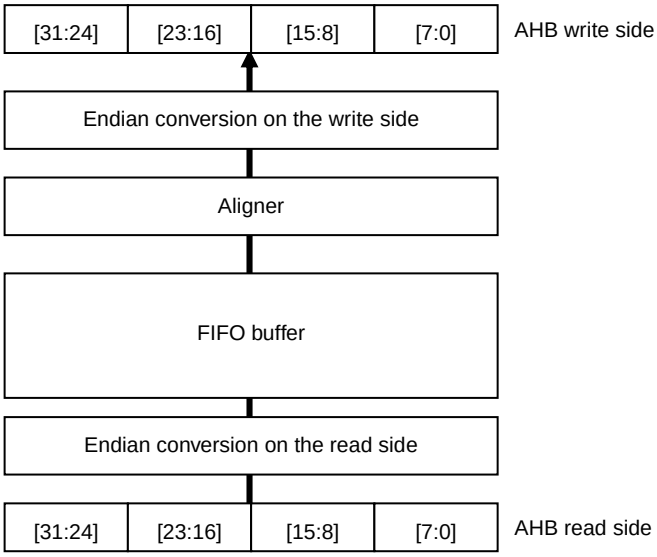


Figure 3-12. Timing for Performing Endian Conversion



3.3.3 Exception for endian conversion settings

Parameters for endian conversion must generally be specified in words. However, specification in other units is possible if any of the conditions below is satisfied.

The table below shows whether endian conversion is available according to the combination of data units used to specify the source and destination addresses.

Table 3-7. Settings That Enable Endian Conversion

Transfer Address Boundary		Endian Conversion	
SourceAddress[1:0]	**DestinationAddress[1:0]**	**Read Side**	**Write Side**
Word Lower address = 00	Word	○	○
	Halfword	○	△
	Byte	○	×
Halfword Lower address = 10	Word	△	○
	Halfword	△	△
	Byte	△	×
Byte Lower address = 01 10	Word	×	○
	Halfword	×	△
	Byte	×	×

Remark ○: Available
△: Available only for conversion in halfword units. Example: D3D2D1D0 will be D2D3D0D1.
×: Unavailable

3.4 DMA Transfer Start Sources

The following two types of sources trigger DMA transfer:

3.4.1 Software request (on PCH0)

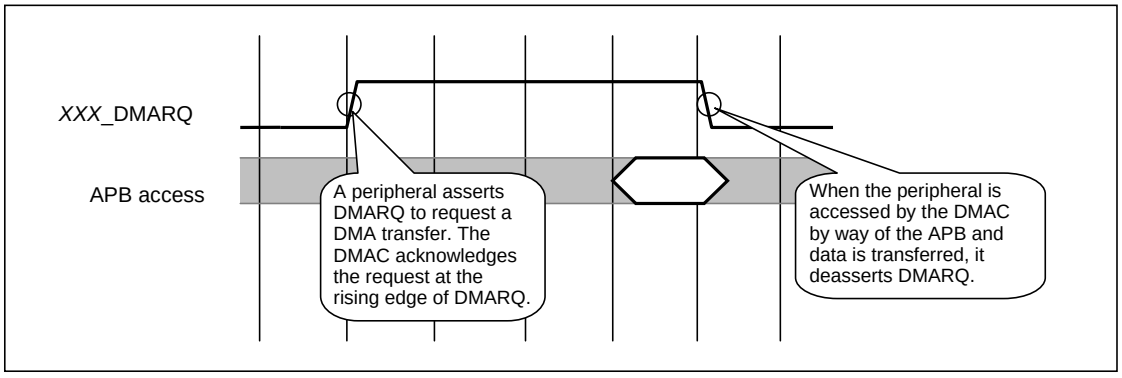
A DMA transfer starts when the DMA start control register of the DMA controller is set to 1.

3.4.2 Request from DMAC external pins (on PCH2 and PCH3)

A DMA transfer starts when a DMA request is issued from a peripheral while starting a DMA transfer is directed by the DMA start control register controlled via software.

A peripheral that requests a DMA transfer asserts the DMARQ signal, and then deasserts the signal when the peripheral receives data or data is read from the peripheral by way of the transfer.
The DMARQ signal is valid on the rising edge. (When the DMA controller detects a rising edge of the DMARQ signal, it acknowledges the signal as a DMA request signal.)

Figure 3-13. DMA Request Signal Timing



The DMARQ signal must be deasserted each time a data block is transferred. Be sure to supply the clock used to detect the signal edge for at least three cycles before and after the rising edge of the DMARQ signal. Otherwise, the DMA controller might fail to detect the DMARQ signal.

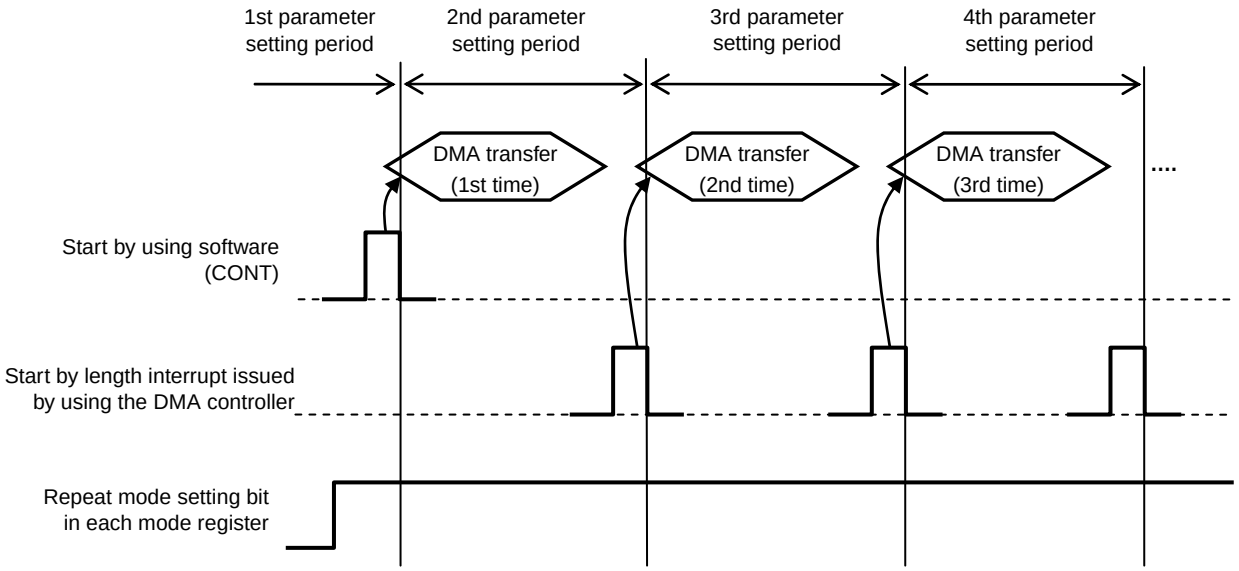
If the DMARQ signal is not in sync with the DMA clock (such as if the signal is output in sync with a serial clock), the period in which the DMA clock is not supplied must be considered when specifying the frequency for the peripheral macro that outputs the DMARQ signal.

3.4.3 Continuous transfer

Parameter setting registers have a reservation function, so the parameters for the subsequent transfer can be specified after a DMA transfer on a channel has started. The parameters must be reserved after one DMA transfer starts and before the next DMA transfer starts. In addition, the repeat mode setting bits corresponding to the transfer channel in each mode register must be set to 1.
A procedure for continuous transfer is shown below.

<1> Specify values for transfer parameters (1st time).
<2> Start DMA.
<3> Specify values for transfer parameters (2nd time).
<4> Wait for a length interrupt to occur. → A length interrupt occurs (1st time).
<5> Specify values for transfer parameters (3rd time).
<6> Wait for a length interrupt to occur. → A length interrupt occurs (2nd time).
<7> Specify values for transfer parameters (4th time).
<8> Wait for a length interrupt to occur. → A length interrupt occurs (3rd time).
... Repeat

Figure 3-14. Continuous Transfer



3.5 Forced Termination

The following three types of sources can be used to terminate a DMA transfer.

(1) Termination due to an error response

(2) Termination by using the DMA end control register

Ongoing DMA transfer can be terminated by setting the DMA end control register.

After the termination, the internal state is initialized.

Caution During a transfer between memory and peripheral, initialize the internal state on the peripheral side as well, after the transfer is terminated.

(3) Termination due to timeout (for UART only)

See 3.2.3 (5) Timeout function.

4. Usage

4.1 Cautions on Use

4.1.1 Requirements for parameter settings

The requirements below apply to the parameter settings for the DMA controller.
DMA operation is not guaranteed if these requirements are not satisfied.

- Repeat transfers must be enabled to perform infinite-length transfers. The offset must be set to 0.
- The offset for a repeat transfer must be set to 0 (one-dimensional transfer).
- Parameters for endian conversion must be specified in words (see the exception described in **3.3.3 Exception for endian conversion settings**).
- PCH1 is reserved.

(1) PCH0

	Register	Minimum Settable Unit
Source side	Address	Byte
	Offset	Byte
Destination side	Address	Byte
	Offset	Byte
Common	Block size	Byte
	Length	Byte

(2) PCH2

	Register	Minimum Settable Unit
Source side	Address	Byte
	Offset	Byte
	Block size	Byte
Destination side	Address	Byte
Common	Length	Byte

(3) PCH3

	Register	Minimum Settable Unit
Source side	Address	Byte
Destination side	Address	Byte
	Offset	Byte
	Block size	Byte
Common	Length	Byte

Remarks **1.** If words are specified as the transfer bit width by using a mode register, the minimum settable unit is a word.

2. If halfwords are specified as the transfer bit width by using a mode register, the minimum settable unit is a halfword.

REVISION HISTORY	EMMA Mobile EV2 User's Manual: DMA Controller

Rev.	Date	Description	
		Page	Summary
1.00	May 27, 2010	—	1st revision release
2.00	Sep 30, 2010	—	Incremental update from comments to the 1.0. (A change part fro the old revision is "★" marked in the page left end.)
3.00	May 31, 2011	—	Incremental update from comments to the 2.0.

EMMA Mobile EV2 User's Manual: DMA Controller

Publication Date: Rev.1.00 May 27, 2010
Rev.3.00 May 31, 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

DMA Controller

EMMA Mobile EV2

Renesas Electronics Corporation

R19UH0043EJ0300