NEC

User's Manual

VRC4173™

VR4100 Series™ Companion Chip

μPD31173

Document No. U14579EJ2V0UM00 (2nd edition)
Date Published February 2002 N CP(K)

[MEMO]

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note:

Strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note:

No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note:

Power-on does not necessarily define initial status of MOS device. Production process of MOS does not define the initial operation status of the device. Immediately after the power source is turned ON, the devices with reset function have not yet been initialized. Hence, power-on does not guarantee out-pin levels, I/O settings or contents of registers. Device is not initialized until the reset signal is received. Reset operation must be executed immediately after power-on for devices having reset function.

- **The information in this document is current as of November, 2001. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC's data sheets or data books, etc., for the most up-to-date specifications of NEC semiconductor products. Not all products and/or types are available in every country. Please check with an NEC sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without prior written consent of NEC. NEC assumes no responsibility for any errors that may appear in this document.
- NEC 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 NEC semiconductor products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of customer's equipment shall be done under the full responsibility of customer. NEC assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC endeavours to enhance the quality, reliability and safety of NEC semiconductor products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC semiconductor products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment, and anti-failure features.
- NEC semiconductor products are classified into the following three quality grades:
  "Standard", "Special" and "Specific". The "Specific" quality grade applies only to semiconductor products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of a semiconductor product depend on its quality grade, as indicated below. Customers must check the quality grade of each semiconductor product before using it in a particular application.
  "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
  "Special": 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, life support systems and medical equipment for life support, etc.
  The quality grade of NEC semiconductor products is "Standard" unless otherwise expressly specified in NEC's data sheets or data books, etc. If customers wish to use NEC semiconductor products in applications not intended by NEC, they must contact an NEC sales representative in advance to determine NEC's willingness to support a given application.
  (Note)
  (1) "NEC" as used in this statement means NEC Corporation and also includes its majority-owned subsidiaries.
  (2) "NEC semiconductor products" means any semiconductor product developed or manufactured by or for NEC (as defined above).

M8E 00.4

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Europe) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 01
Fax: 0211-65 03 327

• Branch The Netherlands
Eindhoven, The Netherlands
Tel: 040-244 58 45
Fax: 040-244 45 80

• Branch Sweden
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

NEC Electronics (France) S.A.
Vélizy-Villacoublay, France
Tel: 01-3067-58-00
Fax: 01-3067-58-99

NEC Electronics (France) S.A.
Representación en España
Madrid, Spain
Tel: 091-504-27-87
Fax: 091-504-28-60

NEC Electronics Italiana S.R.L.
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC Electronics (UK) Ltd.
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
Novena Square, Singapore
Tel: 253-8311
Fax: 250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

NEC do Brasil S.A.
Electron Devices Division
Guarulhos-SP, Brasil
Tel: 11-6462-6810
Fax: 11-6462-6829

J01.12

Major Revisions in This Edition (1/2)

Page	Description
Throughout	Deletion of descriptions related to 32-bit PC card (CardBus card)
p.25	Modification of description in **1.1 Features**
p.27	Modification of description in **1.3 (14) CARDU1, CARDU2 (PC Card Units)**
p.50	Modification of pin I/O direction in **Table 2-1 PCI Bus Interface Signals**
p.51	Modification of pin I/O direction in **Table 2-4 PC Card Interface Signals**
p.56	Modification of **Table 2-13 Test Interface Signals** and **Table 2-14 Test Modes**
pp.58 to 61	Modification of **Table 2-18 Pin Status and Recommended Connection Examples**
p.86	Modification of description in **6.1 General**
p.89	Addition of Note to **6.2.2 CMUSRST (base address + 0x042)**
p.90	Modification of register name in text of **7.1 General**
p.91	Modification of **Figure 7-1 Interrupt Control Outline Diagram**
p.96	Modification of description in **7.2.2 PIUINTREG (base address + 0x062)**
p.102	Modification of description in **7.2.8 MPIUINTREG (base address + 0x06E)**
p.111	Modification of description in **8.2.3 GIUPIODL (base address + 0x084)**
p.112	Modification of description in **8.2.4 GIUPIODH (base address + 0x086)**
p.127	Modification of description in **9.1.1 Block diagrams**
p.129	Modification of **Figure 9-4 Scan Sequencer State Transition Diagram**
p.129	Modification of description in **9.2 (3) ADPortScan state**
p.130	Modification of bit name in **9.2 (5) WaitPenTouch state**
p.134	Addition of Note to **Table 9-2 PIUCNTREG Register Bit Manipulation and States**
p.135	Modification of description in **9.3.2 PIUINTREG (base address + 0x0A4)**
p.145	Modification of bit name in **9.4 (2) Transfer flow for auto scan coordinate detection**
p.147	Modification of bit name in **9.4 (7) Transfer flow when returning from Suspend mode (Disable state)**
p.152	Addition of Caution to **10.1 General**
p.154	Modification of description in **10.2.3 SODATREG (base address + 0x0E6)**
p.158	Modification of bit name in **10.2.7 MCNTREG (base address + 0x0F2)**
p.161	Addition of Note to **10.2.10 SEQREG (base address + 0x0FA)**
p.164	Modification of description in **10.3.1 (2) When not using DMA transfer**
p.165	Modification of bit name in **10.3.2 Input (MIC)**
p.187	Modification of description in **13.1 General**
p.196	Addition of Caution to **13.2.11 CSRBADR (offset address: 0x10 to 0x13)**
pp.197 to 207	Addition of Caution to **13.2.13** to **13.2.25**
pp.209, 210	Addition of Caution to **13.2.28** to **13.2.30**
p.213	Addition of Caution to **13.2.32 SYSCNT (offset address: 0x80 to 0x83)**
p.216	Addition of Caution to **13.2.36** and **13.2.37**
pp.251, 252	Addition of Caution to **13.3.60** and **13.3.61**
p.253	Addition of Caution to **13.3.64 MEM0_CMD_TIM (PCI offset address: 0x885, ExCA extended offset address: 0x0A)**

Major Revisions in This Edition (2/2)

Page	Description
p.255	Addition of Caution to **13.3.67 MEM1_CMD_TIM (PCI offset address: 0x889, ExCA extended offset address: 0x0E)**
p.258	Modification of **Table 13-4 CardBus Socket Registers**
p.259	Modification of description in **13.4.1 SKT_EV (offset address: 0x000)**
pp.262, 263	Addition of Note to **13.4.3 SKT_PRE_STATE (offset address: 0x008)**
p.272	Modification of Caution in **13.5.3 Power supply interface**
p.276	Modification of reset value of device ID register in **Table 14-1 USB Host Control Configuration Registers**
pp.284, 285	Modification of bit name in **14.3.3 HcControl (offset address: 0x04)**
p.346	Modification of **Table 15-2 AC97U Operational Registers**
pp.354, 355	Modification of description and addition of Note in **15.3.7 CTRL (offset address: 0x18)**
p.381	Modification of **Figure 15-5 AC97U-Supported Slots**
p.385	Modification of Remark 2 in **15.5.6 (3) Filter function**
p.387	Modification of bit name in **15.6 (2) SDATAOUT slot 1: CMDADDR (Command Address Port)**
pp.394 to 396	Modification of description in **15.9 (1) Data output to the Codec (slot 3, 4, or 5)**
p.398	Modification of description in **15.9 (2) Data input from the Codec (slot 3, 4, 5, or 6)**
p.402	Addition of **APPENDIX A CAUTIONS**
pp.403 to 407	Addition of **APPENDIX B RESTRICTIONS**

The mark ★ shows major revised points.

INTRODUCTION

Target Readers

This manual is intended for users who understand the functions of the VRC4173 and develop application systems using them.

Purpose

This manual is designed to help users understand the architecture of the VRC4173, as described below.

Organization

This manual covers the following general topics.

- Overview
- Pin functions
- On-chip peripheral functions

How to Read This Manual

This manual assumes that the reader has general knowledge of electrical engineering, logic circuits, and microcomputers.

- To gain a general understanding of VRC4173 functions
  → Read this manual in the order of the contents.
- To learn about the electrical specifications of the VRC4173
  → Refer to the Data Sheet (separate document)

Conventions

Data significance:	Higher digits on the left and lower digits on the right
Active low representation:	XXX# (# after pin or signal name)
Note:	Footnote for item marked with **Note** in the text
Caution:	Information requiring particular attention
Remark:	Supplementary information
Numerical representation:	Binary or decimal ... XXXX
	Hexadecimal ... 0xXXXX

Prefix indicating power of 2 (address space, memory capacity):

K (kilo)	$2^{10} = 1,024$
M (mega)	$2^{20} = 1,024^2$
G (giga)	$2^{30} = 1,024^3$
T (tera)	$2^{40} = 1,024^4$
P (peta)	$2^{50} = 1,024^5$
E (exa)	$2^{60} = 1,024^6$

Related Documents

See the following documents when using this manual.

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

O Documents related to device

Document Name	Document Number
VRC4173 User's Manual	This manual
μPD31173 (VRC4173) Data Sheet	U15338E
VR4122™ User's Manual	U14327E
μPD30122 (VR4122) Data Sheet	To be prepared

CONTENTS

LIST OF FIGURES (1/2)

LIST OF FIGURES (2/2)

LIST OF TABLES (1/3)

LIST OF TABLES (2/3)

LIST OF TABLES (3/3)

CHAPTER 1 OVERVIEW

This chapter presents an overview of the VRC4173.

1.1 Features

The VRC4173 is a companion chip that is to be connected to a product having a PCI bus in the VR4100 Series such as NEC's 64-bit RISC processor VR4122.

The VRC4173 incorporates the I/O macros necessary for a handheld PC running Windows™ CE, and can also access design resources on a personal computer by means of the PCI bus interface.

The VRC4173 has the following features.

- ○ Processor interface
  - PCI bus: Compliant with PCI Local Bus Specification Revision 2.1
    33 MHz operation
  - CLKRUN signal support
- ○ On-chip USB host controller
  - Compliant with Open HCI Specification Release 1.0
  - USB ports: 2 ports
  - Full speed (12 Mbps) and low speed (1.5 Mbps) support
  - On-chip FIFO: 4 × 4 double word (PCI side), 64 × 1 byte (USB side)
- ○ On-chip PC card controller
- ★ • Compliant with 1997 PC Card standard (excluding 32-bit PC card)
  - Supports two card slots
  - Buffer with 5 V withstand voltage
  - On-chip interface for an external power supply control IC
- ○ On-chip AC-Link interface
  - Compliant with AC97 (Audio Codec '97) standard Rev 2.1
  - DMA support
- ○ On-chip PS/2 controller
- ○ On-chip keyboard controller
  - 96-key support (compatible with KIU of VR4121™)Note
- ○ On-chip audio controller
  - Playback (10-bit D/A converter), recording (12-bit A/D converter)
  - Compatible with AIU of VR4121
- ○ On-chip touch panel controller
  - Touch panel driver, coordinate detection (12-bit A/D)
  - General-purpose analog input: 1 port
  - Compatible with PIU function of VR4121
- ○ GPIO (general-purpose I/O pin)
  - Supports a total of 21 pins
  - Compatible with GIU of VR4121
- ○ On-chip 48 MHz oscillator
- ○ Power supply voltage: 3.3 V (some internals with 5 V withstand voltage)

Note This becomes 64-key support when the PS/2 is used with two channels because of exclusive use relative to the PS/2.

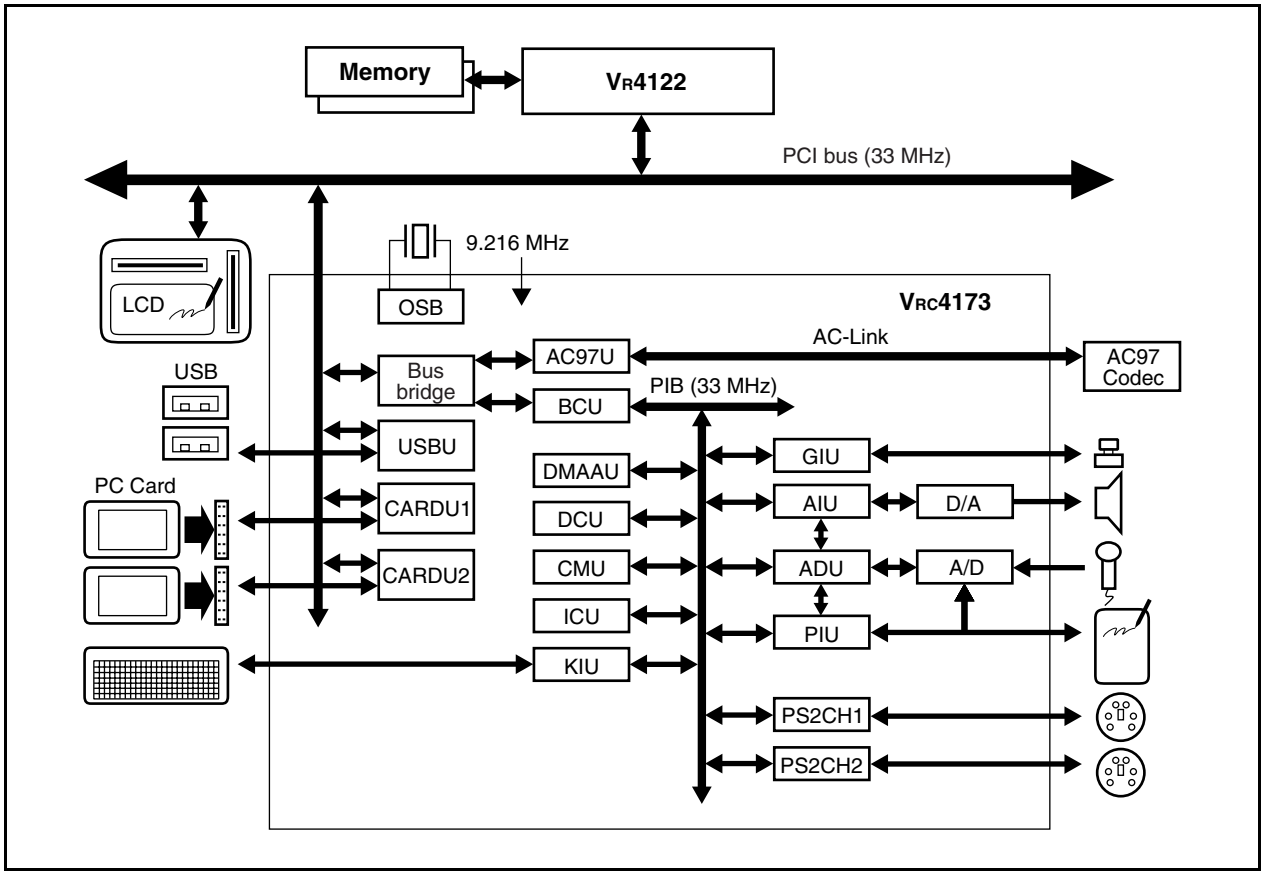
1.2 Ordering Information

Part Number	Package	Internal Maximum Operating Frequency
μPD31173F1-33-HN1	304-pin plastic FBGA (19 × 19)	33 MHz

1.3 Internal Block Configuration

Figure 1-1 shows an internal block diagram of the VRC4173 and connection example with external blocks.

Figure 1-1. Internal Block Diagram and Connection Example with External Blocks



The various peripheral units are briefly described below.

(1) PCI bridge

The PCI bridge controls the PCI bus for the BCU and AC97U units and controls the CLKRUN# signal for controlling the PCI bus clock.

(2) PIB bridge

The PIB bridge is a bus bridge between the internal local bus having a 32-bit width and the internal local bus having a 16-bit width (PIB: Peripheral Internal Bus).

(3) BCU (Bus Control Unit)

The BCU controls the PIB to which various units are connected.

(4) DMAAU (DMA Address Unit)

The DMAAU controls the addresses of DMA transfers that are used by the audio interface (MIC, speakers).

(5) DCU (DMA Control Unit)

The DCU controls the arbitration of DMA transfers that are used by the audio interface (MIC, speakers).

(6) CMU (Clock Mask Unit)

The CMU controls the supply of PCI clock (PCICLK), PIB bus clock (TClock, internal), and 48 MHz clock pulses to internal peripheral units. TClock is a clock having the same speed and timing as the PCI clock.

(7) ICU (Interrupt Control Unit)

The ICU controls interrupt requests that are generated due to external and internal sources and reports interrupt requests to the VR4122. Interrupts from the various on-chip macros of the VRC4173 are requested by a single level interrupt signal, and the source is determined according to the ICU register.

(8) GIU (General-purpose I/O Unit)

The GIU controls the 21 GPIO pins.

(9) PIU (Touch Panel Interface Unit)

The PIU controls an on-chip A/D converter and detects touches on the touch panel.

(10) AIU (Audio Interface Unit)

The AIU controls on-chip A/D and D/A converters and controls MIC sampling and audio output.

(11) KIU (Keyboard Interface Unit)

The KIU, which has 12 scan lines and 8 detection lines, can detect 64, 80, or 96 key inputs.

The scan lines are exclusively used relative to the PS2U. When the PS2U is used with 1 channel, the KIU supports 64 or 80 keys. When the PS2U is used with 2 channels, the KIU supports 64 keys.

(12) PS2U (PS/2 Unit)

The PS2U controls the PS/2 (Personal Computer Standard Keyboard) interface with two channels, PS2CH1 and PS2CH2.

The PS2U is exclusively controlled relative to the key scan lines of the KIU. When the PS/2 is used with 1 channel, the KIU supports 64 or 80 keys. When the PS/2 is used with 2 channels, the KIU supports 64 keys.

(13) ADU (A/D Converter Unit)

The ADU provides an interface with the on-chip A/D converter of the VRC4173. This A/D converter is a serial comparison type.

(14) CARDU1, CARDU2 (PC Card Units)

★ The CARDU1 and CARDU2 control the 16-bit PC card interface, which is compliant with the 1997 PC Card Standard. The 32-bit PC card (CardBus card) is not supported.

(15) USBU (Universal Serial Bus Unit)

The USBU controls the USB interface, which is compliant with Open HCI Specification Release 1.0.

(16) AC97U (AC97 Unit)

The AC97U controls the AC-Link, which is compliant with AC97 (Audio Codec '97) standard Rev 2.1.

1.4 PCI Device Configuration

The VRC4173 consists of a total of five PCI devices, which include three multifunction devices having BCU, AC97, and USB functions and two single function devices for PCMCIA channel 1 and channel 2. The following table shows the PCI devices and the corresponding units comprising each one.

Table 1-1. PCI Devices and Component Units

PCI Device Name		Component Unit	
Multifunction	BCU (Function 0)	DMAAU	Supports two channels for MIC and speaker
		DCU	Supports two channels for MIC and speaker
		CMU	Controls internal clock in low power consumption mode
		ICU	Integrates peripheral unit interrupts
		GIU	Manages 21 general-purpose I/O pins
		PIU	Incorporates A/D converter (12-bit conversion precision)
		AIU	Sampling rate: Maximum 44 ksps 8-bit/16-bit, two channels (D/A converter, A/D converter)
		KIU	Key scanner (12 × 8 line scan)
		PS2CH1	PS/2 control channel 1
		PS2CH2	PS/2 control channel 2
	AC97 (Function 1)	AC97U	Compliant with AC97 standard Rev 2.1
	USB (Function 2)	USBU	USB control
Single function	PCMCIA Ch1	CARDU1	PC card control channel 1
	PCMCIA Ch2	CARDU2	PC card control channel 2

1.5 Lists of Registers

The following tables list the registers of each unit.

Table 1-2. BCU Configuration Registers

Register Symbol	Function	Offset Address
VID	Vendor ID register	0x00 to 0x01
DID	Device ID register	0x02 to 0x03
PCICMD	PCI command register	0x04 to 0x05
PCISTS	PCI device status register	0x06 to 0x07
RID	Revision ID register	0x08
CLASSC	Class code register	0x09 to 0x0B
CACHELS	Cache line size register	0x0C
MLT	Master latency timer register	0x0D
HEDT	Header type register	0x0E
BIST	Built-in self-test register	0x0F
BADR	PIB I/O base address register	0x10 to 0x13
–	Reserved	0x14 to 0x2B
SUBVID	Subsystem vendor ID register	0x2C to 0x2D
SUBID	Subsystem ID register	0x2E to 0x2F
–	Reserved	0x30 to 0x3B
INTL	Interrupt line register	0x3C
INTP	Interrupt pin register	0x3D
MIN_GNT	Burst cycle minimum request time register	0x3E
MAX_LAT	Bus usage right request frequency register	0x3F
BUSCNT	PIB bus control register	0x40
IDSELNUM	PC card IDSEL selection register	0x41

Table 1-3. DMAAU Registers

Register Symbol	Function	Address
AIUIBALREG	AIU IN DMA base address lower register	0x000
AIUIBAHREG	AIU IN DMA base address higher register	0x002
AIUIALREG	AIU IN DMA address lower register	0x004
AIUIAHREG	AIU IN DMA address higher register	0x006
AIUOBALREG	AIU OUT DMA base address lower register	0x008
AIUOBAHREG	AIU OUT DMA base address higher register	0x00A
AIUOALREG	AIU OUT DMA address lower register	0x00C
AIUOAHREG	AIU OUT DMA address higher register	0x00E

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-4. DCU Registers

Register Symbol	Function	Address
DMARSTREG	DMA reset register	0x020
DMAIDLEREG	DMA sequencer status register	0x022
DMASENREG	DMA sequencer enable register	0x024
DMAMSKREG	DMA mask register	0x026
DMAREQREG	DMA request register	0x028

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-5. CMU Registers

Register Symbol	Function	Address
CMUCLKMSK	CMU clock mask register	0x040
CMUSRST	CMU soft reset register	0x042

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-6. ICU Registers

Register Symbol	Function	Address
SYSINT1REG	System interrupt register 1 (level 1)	0x060
PIUINTREG	PIU interrupt register (level 2)	0x062
AIUINTREG	AIU interrupt register (level 2)	0x064
KIUINTREG	KIU interrupt register (level 2)	0x066
GIULINTREG	GIUL interrupt register (level 2)	0x068
GIUHINTREG	GIUH interrupt register (level 2)	0x06A
MSYSINT1REG	System interrupt mask register 1 (level 1)	0x06C
MPIUINTREG	PIU interrupt mask register (level 2)	0x06E
MAIUINTREG	AIU interrupt mask register (level 2)	0x070
MKIUINTREG	KIU interrupt mask register (level 2)	0x072
MGIULINTREG	GIUL interrupt mask register (level 2)	0x074
MGIUHINTREG	GIUH interrupt mask register (level 2)	0x076

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-7. GIU Registers

Register Symbol	Function	Address
GIUDIRL	GPIO I/O select register L	0x080
GIUDIRH	GPIO I/O select register H	0x082
GIUPIODL	GPIO port I/O data register L	0x084
GIUPIODH	GPIO port I/O data register H	0x086
GIUINTSTATL	GPIO interrupt status register L	0x088
GIUINTSTATH	GPIO interrupt status register H	0x08A
GIUINTENL	GPIO interrupt enable register L	0x08C
GIUINTENH	GPIO interrupt enable register H	0x08E
GIUINTTYPL	GPIO interrupt type (edge or level) select register L	0x090
GIUINTTYPH	GPIO interrupt type (edge or level) select register H	0x092
GIUINTALSELL	GPIO interrupt active level select register L	0x094
GIUINTALSELH	GPIO interrupt active level select register H	0x096
GIUINTHTSELL	GPIO interrupt hold/through select register L	0x098
GIUINTHTSELH	GPIO interrupt hold/through select register H	0x09A
SELECTREG	Alternate function pin select register	0x09E

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-8. PIU Registers

Register Symbol	Function	Address
PIUCNTREG	PIU control register	0x0A2
PIUINTREG	PIU interrupt register	0x0A4
PIUSIVLREG	PIU data sampling period setting register	0x0A6
PIUSTBLREG	PIU A/D converter delay time setting register	0x0A8
PIUCMDREG	PIU A/D command register	0x0AA
PIUASCNREG	PIU A/D port scan register	0x0B0
PIUAMSKREG	PIU A/D scan mask register	0x0B2
PIUCIVLREG	PIU delay time count register	0x0BE
PIUPB00REG	PIU page 0 buffer 0 register	0x0C0
PIUPB01REG	PIU page 0 buffer 1 register	0x0C2
PIUPB02REG	PIU page 0 buffer 2 register	0x0C4
PIUPB03REG	PIU page 0 buffer 3 register	0x0C6
PIUPB10REG	PIU page 1 buffer 0 register	0x0C8
PIUPB11REG	PIU page 1 buffer 1 register	0x0CA
PIUPB12REG	PIU page 1 buffer 2 register	0x0CC
PIUPB13REG	PIU page 1 buffer 3 register	0x0CE
PIUAB0REG	PIU A/D scan buffer 0 register	0x0D0
PIUAB1REG	PIU A/D scan buffer 1 register	0x0D2
PIUPB04REG	PIU page 0 buffer 4 register	0x0DC
PIUPB14REG	PIU page 1 buffer 4 register	0x0DE

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-9. AIU Registers

Register Symbol	Function	Address
MDMADATREG	MIC DMA data register	0x0E0
SDMADATREG	Speaker DMA data register	0x0E2
SODATREG	Speaker output data register	0x0E6
SCNTREG	Speaker output control register	0x0E8
SCNVRREG	Speaker conversion rate register	0x0EA
MIDATREG	MIC input data register	0x0F0
MCNTREG	MIC input control register	0x0F2
MCNVRREG	MIC conversion rate register	0x0F4
DVALIDREG	Data valid register	0x0F8
SEQREG	Sequential register	0x0FA
INTREG	Interrupt register	0x0FC

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-10. KIU Registers

Register Symbol	Function	Address
KIUDAT0	KIU data0 register	0x100
KIUDAT1	KIU data1 register	0x102
KIUDAT2	KIU data2 register	0x104
KIUDAT3	KIU data3 register	0x106
KIUDAT4	KIU data4 register	0x108
KIUDAT5	KIU data5 register	0x10A
KIUSCANREP	KIU scan/repeat register	0x110
KIUSCANS	KIU scan status register	0x112
KIUWKS	KIU wait keyscan stable register	0x114
KIUWKI	KIU wait keyscan interval register	0x116
KIUINT	KIU interrupt register	0x118
KIURST	KIU reset register	0x11A
SCANLINE	KIU scan line register	0x11E

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-11. PS2CH1 Registers

Register Symbol	Function	Address
PS2CH1DATA	PS/2 channel 1 transmission/reception data register	0x120
PS2CH1CTRL	PS/2 channel 1 control register	0x122
PS2CH1RST	PS/2 channel 1 reset register	0x124

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-12. PS2CH2 Registers

Register Symbol	Function	Address
PS2CH2DATA	PS/2 channel 2 transmission/reception data register	0x140
PS2CH2CTRL	PS/2 channel 2 control register	0x142
PS2CH2RST	PS/2 channel 2 reset register	0x144

Remark The sum of a value in the Address column added to the base address that is set according to the BADR register of the BCU will be the physical address.

Table 1-13. CARDU Configuration Registers (1/2)

Register Symbol	Function	Offset Address
VID	Vendor ID register	0x00 to 0x01
DID	Device ID register	0x02 to 0x03
PCICMD	PCI command register	0x04 to 0x05
PCISTS	PCI device status register	0x06 to 0x07
RID	Revision ID register	0x08
CLASSC	Class code register	0x09 to 0x0B
CACHELS	Cache line size register	0x0C
MLT	Master latency timer register	0x0D
HEDT	Header type register	0x0E
BIST	Built-in self-test register	0x0F
CSRBADR	CardBus socket/ExCA base address register	0x10 to 0x13
CAP	PCI additional specifications code register	0x14
–	Reserved	0x15
SECSTS	Second status register	0x16 to 0x17
PCIBNUM	PCI bus number register	0x18
CARDNUM	Card number register	0x19
SUBBNUM	Subordinate bus number register	0x1A
CLT	CardBus latency timer register	0x1B
MEMB0	Memory base address register 0	0x1C to 0x1F
MEML0	Memory space boundary register 0	0x20 to 0x23
MEMB1	Memory base address register 1	0x24 to 0x27
MEML1	Memory space boundary register 1	0x28 to 0x2B
IOB0	I/O base address register 0	0x2C to 0x2F
IOL0	I/O space boundary register 0	0x30 to 0x33
IOB1	I/O base address register 1	0x34 to 0x37
IOL1	I/O space boundary register 1	0x38 to 0x3B

Table 1-13. CARDU Configuration Registers (2/2)

Register Symbol	Function	Offset Address
INTL	Interrupt line register	0x3C
INTP	Interrupt pin register	0x3D
BRGCNT	Bridge control register	0x3E to 0x3F
SUBVID	Subsystem vendor ID register	0x40 to 0x41
SUBID	Subsystem ID register	0x42 to 0x43
PC16BADR	PC card 16-bit interface legacy mode base address register	0x44 to 0x47
–	Reserved	0x48 to 0x7F
SYSCNT	System control register	0x80 to 0x83
–	Reserved	0x84 to 0x90
DEVCNT	Device control register	0x91
–	Reserved	0x92 to 0x93
SKDMA0	Socket DMA register 0	0x94 to 0x97
SKDMA1	Socket DMA register 1	0x98 to 0x9B
CHIPCNT	Chip control register	0x9C
–	Reserved	0x9D to 0x9E
SERRDIS	SERR# signal disable register	0x9F
CAPID	Capability ID register	0xA0
NIP	Power management additional function register	0xA1
PMC	Power management characteristic register	0xA2 to 0xA3
PMCSR	Power management control/status register	0xA4 to 0xA5
PMCSR_BSE	PMCSR bridge support extension register	0xA6
DATA	Data register	0xA7
–	Reserved	0xA8 to 0xFB
TEST	Test register	0xFC
–	Reserved	0xFD to 0xFFH

Table 1-14. ExCA Registers (1/2)

Register Symbol	Function	Offset Address	
		PCI Memory	ExCA
ID_REV	ID/revision register	0x800	0x00
IF_STATUS	Interface status register	0x801	0x01
PWR_CNT	Power control register	0x802	0x02
INT_GEN_CNT	Interrupt/general-purpose control register	0x803	0x03
CARD_SC	Card status change register	0x804	0x04
CARD_SCI	Card status change interrupt configuration register	0x805	0x05
ADR_WIN_EN	Address window enable register	0x806	0x06
IO_WIN_CNT	I/O window control register	0x807	0x07
IO_WIN0_SAL	I/O window 0 start address lower byte register	0x808	0x08
IO_WIN0_SAH	I/O window 0 start address higher byte register	0x809	0x09
IO_WIN0_EAL	I/O window 0 end address lower byte register	0x80A	0x0A
IO_WIN0_EAH	I/O window 0 end address higher byte register	0x80B	0x0B
IO_WIN1_SAL	I/O window 1 start address lower byte register	0x80C	0x0C
IO_WIN1_SAH	I/O window 1 start address higher byte register	0x80D	0x0D
IO_WIN1_EAL	I/O window 1 end address lower byte register	0x80E	0x0E
IO_WIN1_EAH	I/O window 1 end address higher byte register	0x80F	0x0F
MEM_WIN0_SAL	Memory window 0 start address lower byte register	0x810	0x10
MEM_WIN0_SAH	Memory window 0 start address higher byte register	0x811	0x11
MEM_WIN0_EAL	Memory window 0 end address lower byte register	0x812	0x12
MEM_WIN0_EAH	Memory window 0 end address higher byte register	0x813	0x13
MEM_WIN0_OAL	Memory window 0 offset address lower byte register	0x814	0x14
MEM_WIN0_OAH	Memory window 0 offset address higher byte register	0x815	0x15
GEN_CNT	General control register	0x816	0x16
–	Reserved	0x817	0x17
MEM_WIN1_SAL	Memory window 1 start address lower byte register	0x818	0x18
MEM_WIN1_SAH	Memory window 1 start address higher byte register	0x819	0x19
MEM_WIN1_EAL	Memory window 1 end address lower byte register	0x81A	0x1A
MEM_WIN1_EAH	Memory window 1 end address higher byte register	0x81B	0x1B
MEM_WIN1_OAL	Memory window 1 offset address lower byte register	0x81C	0x1C

Table 1-14. ExCA Registers (2/2)

Register Symbol	Function	Offset Address	
		PCI Memory	ExCA
MEM_WIN1_OAH	Memory window 1 offset address higher byte register	0x81D	0x1D
GLO_CNT	Global control register	0x81E	0x1E
–	Reserved	0x81F	0x1F
MEM_WIN2_SAL	Memory window 2 start address lower byte register	0x820	0x20
MEM_WIN2_SAH	Memory window 2 start address higher byte register	0x821	0x21
MEM_WIN2_EAL	Memory window 2 end address lower byte register	0x822	0x22
MEM_WIN2_EAH	Memory window 2 end address higher byte register	0x823	0x23
MEM_WIN2_OAL	Memory window 2 offset address lower byte register	0x824	0x24
MEM_WIN2_OAH	Memory window 2 offset address higher byte register	0x825	0x25
–	Reserved	0x826 to 0x827	0x26 to 0x27
MEM_WIN3_SAL	Memory window 3 start address lower byte register	0x828	0x28
MEM_WIN3_SAH	Memory window 3 start address higher byte register	0x829	0x29
MEM_WIN3_EAL	Memory window 3 end address lower byte register	0x82A	0x2A
MEM_WIN3_EAH	Memory window 3 end address higher byte register	0x82B	0x2B
MEM_WIN3_OAL	Memory window 3 offset address lower byte register	0x82C	0x2C
MEM_WIN3_OAH	Memory window 3 offset address higher byte register	0x82D	0x2D
EXT_INDX	Extended index register	–	0x2E
EXT_DATA	Extended data register	–	0x2F
MEM_WIN4_SAL	Memory window 4 start address lower byte register	0x830	0x30
MEM_WIN4_SAH	Memory window 4 start address higher byte register	0x831	0x31
MEM_WIN4_EAL	Memory window 4 end address lower byte register	0x832	0x32
MEM_WIN4_EAH	Memory window 4 end address higher byte register	0x833	0x33
MEM_WIN4_OAL	Memory window 4 offset address lower byte register	0x834	0x34
MEM_WIN4_OAH	Memory window 4 offset address higher byte register	0x835	0x35
IO_WIN0_OAL	I/O window 0 offset address lower byte register	0x836	0x36
IO_WIN0_OAH	I/O window 0 offset address higher byte register	0x837	0x37
IO_WIN1_OAL	I/O window 1 offset address lower byte register	0x838	0x38
IO_WIN1_OAH	I/O window 1 offset address higher byte register	0x839	0x39
–	Reserved	0x83A to 0x83F	0x3A to 0x3F

Table 1-15. ExCA Extended Registers

Register Symbol	Function	Offset Address	
		PCI Memory	ExCA Extension
MEM_WIN0_SAU	Memory window 0 start address higher byte register	0x840	0x00
MEM_WIN1_SAU	Memory window 1 start address higher byte register	0x841	0x01
MEM_WIN2_SAU	Memory window 2 start address higher byte register	0x842	0x02
MEM_WIN3_SAU	Memory window 3 start address higher byte register	0x843	0x03
MEM_WIN4_SAU	Memory window 4 start address higher byte register	0x844	0x04
IO_SETUP_TIM	I/O setup timing register	0x880	0x05
IO_CMD_TIM	I/O command timing register	0x881	0x06
IO_HOLD_TIM	I/O hold timing register	0x882	0x07
–	Reserved	0x883	0x08
MEM0_SETUP_TIM	Memory setup timing 0 register	0x884	0x09
MEM0_CMD_TIM	Memory command timing 0 register	0x885	0x0A
MEM0_HOLD_TIM	Memory hold timing 0 register	0x886	0x0B
–	Reserved	0x887	0x0C
MEM1_SETUP_TIM	Memory setup timing 1 register	0x888	0x0D
MEM1_CMD_TIM	Memory command timing 1 register	0x889	0x0E
MEM1_HOLD_TIM	Memory hold timing 1 register	0x88A	0x0F
–	Reserved	0x88B	0x10
MEM_TIM_SEL1	Memory timing selection 1 register	0x88C	0x11
MEM_TIM_SEL2	Memory timing selection 2 register	0x88D	0x12
–	Reserved	0x88E to 0x890	0x13 to 0x15
MEM_WIN_PWEN	Memory window post write enable register	0x891	0x16

Table 1-16. CardBus Socket Registers

Register Symbol	Function	Offset Address
SKT_EV	Socket event register	0x000
SKT_MASK	Socket mask register	0x004
SKT_PRE_STATE	Socket present state register	0x008
SKT_FORCE_EV	Socket force event register	0x00C
SKT_CNT	Socket control register	0x010
–	Reserved	0x014 to 0x7FF

Table 1-17. USB Host Control Configuration Registers

Name	Offset Address
Vendor ID register	0x00
Device ID register	0x02
Command register	0x04
Status register	0x06
Revision ID register	0x08
Class code base address register	0x09
Class code sub class register	
Class code programming interface register	
Cache line size register	0x0C
Latency timer register	0x0D
Header type register	0x0E
Built-in self-test register	0x0F
Base address register	0x10
Subsystem vendor ID register	0x2C
Subsystem ID register	0x2E
Interrupt line register	0x3C
Interrupt pin register	0x3D
Min_Gnt register (burst cycle minimum request time register)	0x3E
Max_lat register (bus usage right request frequency register)	0x3F
Power management register	0xE0

Table 1-18. Host Control Operational Registers

Register Symbol	Function	Offset Address
HcRevision	HC revision register	0x00
HcControl	HC control register	0x04
HcCommandStatus	HC command register	0x08
HcInterruptStatus	HC interrupt request detection register	0x0C
HcInterruptEnable	HC interrupt request enable register	0x10
HcInterruptDisable	HC interrupt request disable register	0x14
HcHCCA	HC base address register	0x18
HcPeriodCurrentED	HC period current ED register	0x1C
HcControlHeadED	HC control list 1st ED register	0x20
HcControlCurrentED	HC control list current ED register	0x24
HcBulkHeadED	HC bulk list 1st ED register	0x28
HcBulkCurrentED	HC bulk list current ED register	0x2C
HcDoneHead	HC last TD register	0x30
HcFmInterval	HC frame interval register	0x34
HcFmRemaining	HC frame bit time remaining register	0x38
HcFmNumber	HC frame counter register	0x3C
HcPeriodicStart	HC list processing start register	0x40
HcLSThreshold	HC low speed transfer diagnosis register	0x44
HcRhDescriptorA	HC power supply status register A	0x48
HcRhDescriptorB	HC power supply status register B	0x4C
HcRhStatus	HC status register	0x50
HcRhPortStatus1	HC port status register 1	0x54
HcRhPortStatus2	HC port status register 2	0x58

Table 1-19. AC97U PCI Configuration Registers

Register Symbol	Function	Offset Address
VID	Vendor ID register	0x00 to 0x01
DID	Device ID register	0x02 to 0x03
PCICMD	PCI command register	0x04 to 0x05
PCISTS	PCI device status register	0x06 to 0x07
RID	Revision ID register	0x08
CLASSC	Class code register	0x09 to 0x0B
CACHELS	Cache line size register	0x0C
MLT	Master latency timer register	0x0D
HEDT	Header type register	0x0E
BIST	Built-in self-test register	0x0F
BASEADR	Base address register	0x10 to 0x13
–	Reserved	0x14 to 0x2B
SVID	Subsystem vendor ID register	0x2C to 0x2D
SUBID	Subsystem ID register	0x2E to 0x2F
EXROMADR	Extended ROM base address register	0x30 to 0x33
–	Reserved	0x34 to 0x3B
INTL	Interrupt line register	0x3C
INTP	Interrupt pin register	0x3D
MIN_GNT	Burst cycle minimum request time register	0x3E
MAX_LAT	Bus usage right request frequency register	0x3F
–	Reserved	0x40 to 0xFF

Table 1-20. AC97U Operational Registers

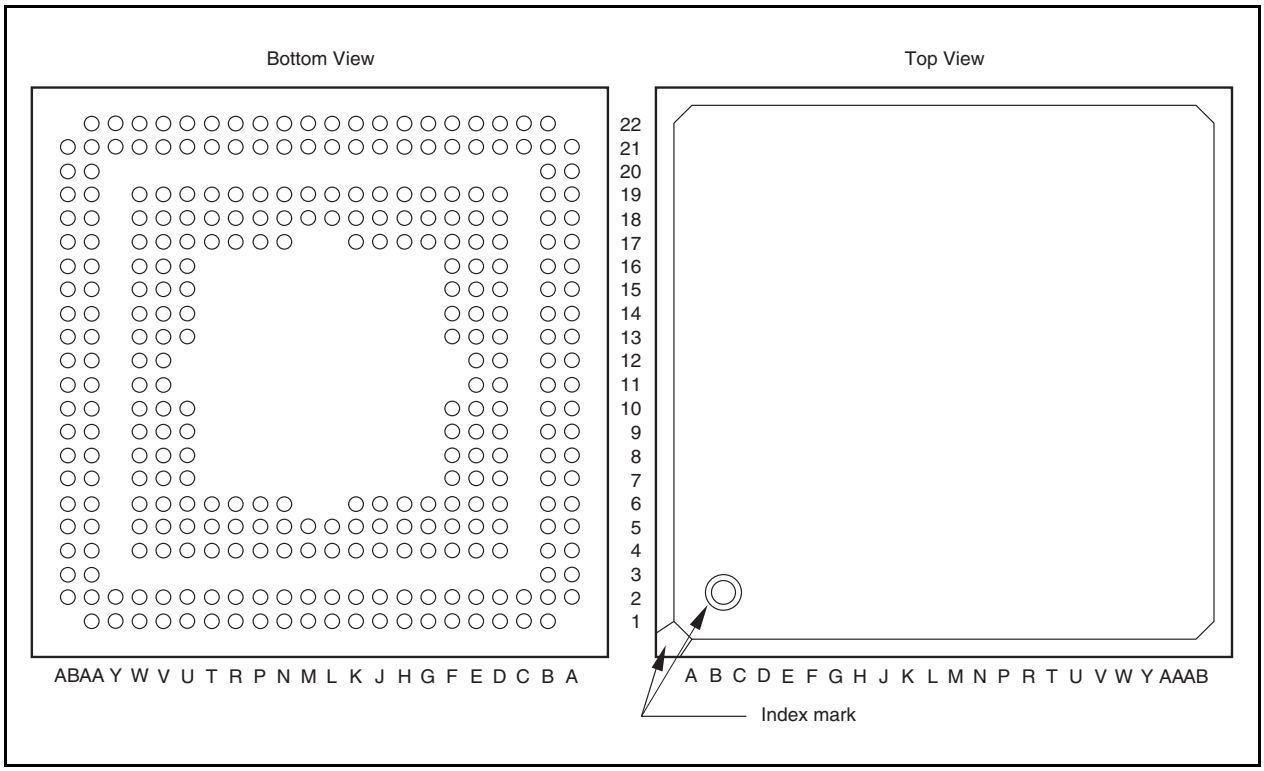
Register Symbol	Function	Offset Address
INT_CLR/INT_STATUS	Interrupt clear/status register	0x00
CODEC_WR	Codec write register	0x04
CODEC_RD	Codec read register	0x08
CODEC_REQ	Codec slot request register	0x0C
SLOT12_WR	Slot 12 write register	0x10
SLOT12_RD	Slot 12 read register	0x14
CTRL	Codec/SRC control register	0x18
ACLINK_CTRL	AC-Link control register	0x1C
SRC_RAM_DATA	Sample rate converter RAM data register	0x20
INT_MASK	Interrupt mask register	0x24
–	Reserved	0x28 to 0x2C
DAC1_CTRL	DAC1 DMA control register	0x30
DAC1L	DAC1 DMA length register	0x34
DAC1_BADDR	DAC1 DMA base address register	0x38
DAC2_CTRL	DAC2 DMA control register	0x3C
DAC2L	DAC2 DMA length register	0x40
DAC2_BADDR	DAC2 DMA base address register	0x44
DAC3_CTRL	DAC3 DMA control register	0x48
DAC3L	DAC3 DMA length register	0x4C
DAC3_BADDR	DAC3 DMA base address register	0x50
ADC1_CTRL	ADC1 DMA control register	0x54
ADC1L	ADC1 DMA length register	0x58
ADC1_BADDR	ADC1 DMA base address register	0x5C
ADC2_CTRL	ADC2 DMA control register	0x60
ADC2L	ADC2 DMA length register	0x64
ADC2_BADDR	ADC2 DMA base address register	0x68
ADC3_CTRL	ADC3 DMA control register	0x6C
ADC3L	ADC3 DMA length register	0x70
ADC3_BADDR	ADC3 DMA base address register	0x74

CHAPTER 2 PIN FUNCTIONS

This chapter describes the pin functions of the VRC4173.

2.1 Pin Configuration

- 304-pin plastic FBGA (19 × 19)
  μPD31173F1-33-HN1



Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
A2	$V_{DD}2$	B20	CD22#	E14	KSCAN2/GPIO2
A3	TEST2	B21	BVD22#	E15	$V_{DD}3$
A4	TEST0	B22	$V_{DD}2$	E16	KPORT3/GPIO11
A5	ADAGND	C1	PWCCLK	E17	GND2
A6	ADY	C2	PWCLATCH	E18	GND2
A7	$ADDV_{DD}$	C21	RCVBE	E19	GND2
A8	TESTC	C22	BVD21#	E21	IOWR2#
A9	TPX1/GPIO17	D1	VS11#	E22	IORD2#
A10	$V_{DD}3$	D2	VS12#	F1	CE11#
A11	SYNC	D4	GND2	F2	CE12#
A12	KSCAN8/PS2CLK2	D5	GND2	F4	OE1#
A13	KSCAN3/GPIO3	D6	AUDIOIN	F5	GND2
A14	GND3	D7	$ADAV_{REF}M$	F6	C1A0
A15	KPORT5/GPIO13	D8	AUDIOOUT	F7	ADX
A16	KPORT0/GPIO8	D9	TPY0/GPIO18	F8	$DAAV_{DD}$
A17	OCI2	D10	GND3	F9	TPY1/GPIO19
A18	READY2	D11	SDATAOUT	F10	TPX0/GPIO16
A19	WAIT2#	D12	KSCAN10/PS2CLK1	F13	KSCAN7/GPIO7
A20	CD21#	D13	KSCAN5/GPIO5	F14	$V_{DD}2$
A21	GND2	D14	GND2	F15	KSCAN0/GPIO0
B1	GND2	D15	KPORT7/GPIO15	F16	GND2
B2	PWCDATA	D16	KPORT2/GPIO10	F17	KPORT4/GPIO12
B3	TEST3	D17	PPON2	F18	GND2
B4	TEST1	D18	GND2	F19	VS22#
B5	ADDGND	D19	GND2	F21	VS21#
B6	ADIN	D21	RESET2	F22	WP2
B7	$ADAV_{REF}P$	D22	REG2#	G1	WE1#
B8	DAAGND	E1	RESET1	G2	WP1
B9	GND2	E2	REG1#	G4	IORD1#
B10	ACLINKRST#	E4	GND2	G5	IOWR1#
B11	BCLK	E5	GND2	G6	GND2
B12	KSCAN9/PS2DATA2	E6	GND2	G17	WE2#
B13	KSCAN4/GPIO4	E7	$ADAV_{DD}$	G18	OE2#
B14	KSCAN1/GPIO1	E8	$DAAV_{REF}$	G19	CE22#
B15	KPORT6/GPIO14	E9	$V_{DD}2$	G21	CE21#
B16	KPORT1/GPIO9	E10	TPEN/GPIO20	G22	C2D15
B17	PPON1	E11	SDATAIN	H1	C1A1
B18	OCI1	E12	KSCAN11/PS2DATA1	H2	$V_{DD}3$
B19	INPACK2#	E13	KSCAN6/GPIO6	H4	C1A2

Remark # indicates active low.

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
H5	GND3	M18	C2D3	T17	GND2
H6	C1A3	M19	C2D2	T18	C2A13
H17	C2D14	M21	C2D1	T19	C2A12
H18	C2D13	M22	C2D0	T21	C2A11
H19	C2D12	N1	C1A21	T22	C2A10
H21	C2D11	N2	C1A22	U1	C1D10
H22	GND3	N4	C1A23	U2	C1D11
J1	GND2	N5	C1A24	U4	C1D12
J2	C1A4	N6	GND2	U5	GND2
J4	$V_{DD}2$	N17	C2A25	U6	AD23
J5	C1A5	N18	C2A24	U7	GND2
J6	C1A6	N19	C2A23	U8	AD18
J17	C2D10	N21	C2A22	U9	GND2
J18	$V_{DD}3$	N22	C2A21	U10	AD12
J19	C2D9	P1	C1A16	U13	AD0
J21	$V_{DD}2$	P2	$V_{DD}2$	U14	CBE2#
J22	C2D8	P4	C1A25	U15	VRCINT
K1	C1A7	P5	$V_{DD}3$	U16	PAR
K2	C1A8	P6	C1D0	U17	C2A14
K4	C1A9	P17	$V_{DD}2$	U18	GND2
K5	C1A10	P18	C2A16	U19	C2A9
K6	C1A11	P19	GND2	U21	C2A8
K17	GND2	P21	C2A20	U22	C2A7
K18	DN2	P22	GND3	V1	C1D13
K19	DP2	R1	GND3	V2	C1D14
K21	DN1	R2	C1D1	V4	GND2
K22	DP1	R4	C1D2	V5	GND2
L1	C1A12	R5	C1D3	V6	GND2
L2	C1A13	R6	C1D4	V7	AD24
L4	C1A14	R17	C2A19	V8	AD19
L5	C1A15	R18	$V_{DD}3$	V9	AD16
L18	C2D7	R19	C2A18	V10	AD13
L19	C2D6	R21	C2A17	V11	AD9
L21	C2D5	R22	C2A15	V12	AD5
L22	C2D4	T1	C1D5	V13	AD1
M1	C1A17	T2	C1D6	V14	$V_{DD}2$
M2	C1A18	T4	C1D7	V15	CBE0#
M4	C1A19	T5	C1D8	V16	FRAME#
M5	C1A20	T6	C1D9	V17	GND2

Remark # indicates active low.

Pin No.	Pin Name	Pin No.	Pin Name	Pin No.	Pin Name
V18	GND2	Y1	BVD12#	AA21	C2A0
V19	GND2	Y2	INPACK1#	AA22	$V_{DD}2$
V21	C2A6	Y21	C2A2	AB2	$V_{DD}2$
V22	C2A5	Y22	C2A1	AB3	READY1
W1	C1D15	AA1	GND2	AB4	CD11#
W2	BVD11#	AA2	WAIT1#	AB5	PCLK
W4	GND2	AA3	SCLK	AB6	AD30
W5	GND2	AA4	CD12#	AB7	AD27
W6	AD28	AA5	AD31	AB8	AD22
W7	AD25	AA6	AD29	AB9	$V_{DD}3$
W8	AD20	AA7	AD26	AB10	AD15
W9	GND3	AA8	AD21	AB11	AD11
W10	AD14	AA9	AD17	AB12	AD8
W11	AD10	AA10	$V_{DD}2$	AB13	AD4
W12	AD6	AA11	IDSEL	AB14	PCIRST#
W13	AD2	AA12	AD7	AB15	$V_{DD}3$
W14	CBE3#	AA13	AD3	AB16	IRDY#
W15	GND3	AA14	GND2	AB17	PERR#
W16	DEVSEL#	AA15	CBE1#	AB18	REQ#
W17	CLKRUN#	AA16	TRDY#	AB19	SERR#
W18	GND2	AA17	STOP#	AB20	CLK48MX1
W19	GND2	AA18	GNT#	AB21	GND2
W21	C2A4	AA19	CLK48M		
W22	C2A3	AA20	CLK48MX2		

Remark # indicates active low.

Pin Identification

ACLINKRST#: AC-Link Reset
AD(31:0): Address/Data Bus
ADAGND: A/D Converter Analog Ground
$ADAV_{DD}$: A/D Converter Analog V_{DD}
$ADAV_{REF}M$: Analog Reference Minus Voltage for A/D Converter
$ADAV_{REF}P$: Analog Reference Plus Voltage for A/D Converter
ADDGND: Digital Ground for A/D Converter
$ADDV_{DD}$: Digital V_{DD} for A/D Converter
ADIN: A/D Converter General Input
ADX: A/D Converter Input for X Axis Port
ADY: A/D Converter Input for Y Axis Port
AUDIOIN: Audio Input
AUDIOOUT: Audio Output
BCLK: Bit Clock
BVD11#, BVD12#, BVD21#, BVD22#: Battery Voltage Detect
C1A(25:0): Card Slot 1 Address
C1D(15:0): Card Slot 1 Data
C2A(25:0): Card Slot 2 Address
C2D(15:0): Card Slot 2 Data
CBE(3:0)#: Chipset Byte Enable
CD11#, CD12#, CD21#, CD22#: Card Detect
CE11#, CE12#, CE21#, CE22#: Card Enable
CLK48M: 48 MHz Clock Output
CLK48MX1, CLK48MX2: 48 MHz Crystal Clock Input
CLKRUN#: Clock Run
DAAGND: Analog Ground for D/A Converter
$DAAV_{DD}$: Analog V_{DD} for D/A Converter
$DAAV_{REF}$: Analog Reference Voltage for D/A Converter
DEVSEL#: Device Select
DN1, DN2: Data Negative
DP1, DP2: Data Positive
FRAME#: Frame
GND2: Ground for I/O Buffer
GND3: Ground for Internal Circuit
GNT#: Grant
GPIO(20:0): General Purpose Input/Output
IDSEL: Identifier Select
INPACK1#, INPACK2#: Input Acknowledge
IORD1#, IORD2#: I/O Read
IOWR1#, IOWR2#: I/O Write
IRDY#: Initiator Ready
KPORT(7:0): Key Port
KSCAN(11:0): Key Scan
OCI1, OCI2: Over Current Indicator
OE1#, OE2#: Output Enable
PAR: Parity
PCIRST#: PCI Reset
PCLK: PCI Clock
PERR#: Parity Error
PPON1, PPON2: Port Power On
PS2CLK1, PS2CLK2: PS/2 Clock
PS2DATA1, PS2DATA2: PS/2 Data
PWCCLK: Power Control Clock
PWCDATA: Power Control Data
PWCLATCH: Power Control Latch
RCVBE: Receiver Buffer Enable
READY1, READY2: Ready
REG1#, REG2#: Attribute Memory Select
REQ#: Request
RESET1, RESET2: Reset
SCLK: Suspend Less Clock
SDATAIN: Serial Data Input
SDATAOUT: Serial Data Output

Remark # indicates active low.

SERR#:	System Error	TRDY#:	Target Ready
STOP#:	Stop	$V_{DD}2$:	Power Supply for I/O Buffer
SYNC:	Synchronization	$V_{DD}3$:	Power Supply Internal Circuit
TEST(3:0):	Test	VRCINT:	V_R Series Companion Chip
TESTC:	Test Clock		Interrupt
TPEN:	Touch Panel Interface Enable	VS11#, VS12#,	
TPX(1:0):	Touch Panel Interface	VS21#, VS22#:	Voltage Sense
	X Axis Port	WAIT1#, WAIT2#:	Wait
TPY(1:0):	Touch Panel Interface	WE1#, WE2#:	Write Enable
	Y Axis Port	WP1, WP2:	Write Protect

Remark # indicates active low.

2.2 Pin Function Lists

2.2.1 PCI bus interface signals

Table 2-1. PCI Bus Interface Signals

Signal Name	I/O	Function
AD(31:0)	I/O	This is a 32-bit bus that multiplexes the address bus and data bus.
CBE(3:0)#	I/O	This is a signal that multiplexes the bus command and byte enable signals.
IDSEL	I	This is an initialization device selection signal (for a PCI multifunction device).
FRAME#	I/O	This is a cycle frame signal.
DEVSEL#	I/O	This is a device selection signal.
IRDY#	I/O	This is an initiator ready signal.
TRDY#	I/O	This is a target ready signal.
STOP#	I/O	This is a stop signal.
REQ#	O	This is a PCI bus request signal.
GNT#	I	This is a PCI bus request acknowledge signal.
PCIRST#	I	This is a PCI reset signal.
CLKRUN#	I/O	This is a PCI clock run signal.
PAR	I/O	This is an even parity signal.
★ PERR#	I/O	This signal becomes active when a parity error occurs.
SERR#	O	This signal becomes active when a system error occurs.

2.2.2 USB interface signals

Table 2-2. USB Interface Signals

Signal Name	I/O	Function
OCI1, OCI2	I	Set these signals to active when an over current is detected.
DP1, DP2	I/O	These are USB serial data positive signals.
DN1, DN2	I/O	These are USB serial data negative signals.
PPON1, PPON2	O	These are USB root hub port power supply control signals.
RCVBE	I	This is a buffer enable signal. Set this to active to make the input signal to the USB port valid.

2.2.3 AC-Link interface signals

Table 2-3. AC-Link Interface Signals

Signal Name	I/O	Function
SDATAIN	I	This is the serial data signal from the AC97 Codec.
BCLK	I	This is the bit clock (28 MHz) signal from the AC97 Codec.
SDATAOUT	O	This is the serial data signal to the AC97 Codec.
SYNC	O	This is the SYNC output signal to the AC97 Codec.
ACLINKRST#	O	This is the AC97 Codec reset signal.

2.2.4 PC card interface signals

Table 2-4. PC Card Interface Signals

Signal Name	I/O	Function
C1A(25:0)	I/O	This is the PC card slot 1 address bus.
C1D(15:0)	I/O	This is the PC card slot 1 data bus.
CE1(2:1)#	I/O	This is the PC card slot 1 chip select signal.
CD1(2:1)#	I	This is the PC card slot 1 card detect signal.
OE1#	I/O	This is the PC card slot 1 output enable signal.
WE1#	O	This is the PC card slot 1 write enable signal.
READY1	I	This is the PC card slot 1 ready signal.
WP1	I/O	This is the PC card slot 1 write protect signal.
VS1(2:1)#	I/O	This is the PC card slot 1 card power supply identification signal.
IORD1#	I/O	This is the PC card slot 1 I/O read signal.
IOWR1#	I/O	This is the PC card slot 1 I/O write signal.
★ RESET1	O	This is the PC card slot 1 reset signal.
WAIT1#	I	This is the PC card slot 1 wait signal.
INPACK1#	I	This is the PC card slot 1 input acknowledge signal.
REG1#	I/O	This is the PC card slot 1 memory chip select signal.
BVD1(2:1)#	I	This is the PC card slot 1 battery voltage detect signal.
C2A(25:0)	I/O	This is the PC card slot 2 address bus.
C2D(15:0)	I/O	This is the PC card slot 2 data bus.
CE2(2:1)#	I/O	This is the PC card slot 2 chip select signal.
CD2(2:1)#	I	This is the PC card slot 2 card detect signal.
OE2#	I/O	This is the PC card slot 2 output enable signal.
WE2#	O	This is the PC card slot 2 write enable signal.
READY2	I	This is the PC card slot 2 ready signal.
WP2	I/O	This is the PC card slot 2 write protect signal.
VS2(2:1)#	I/O	This is the PC card slot 2 card power supply identification signal.
IORD2#	I/O	This is the PC card slot 2 I/O read signal.
IOWR2#	I/O	This is the PC card slot 2 I/O write signal.
★ RESET2	O	This is the PC card slot 2 reset signal.
WAIT2#	I	This is the PC card slot 2 wait signal.
INPACK2#	I	This is the PC card slot 2 input acknowledge signal.
REG2#	I/O	This is the PC card slot 2 memory chip select signal.
BVD2(2:1)#	I	This is the PC card slot 2 battery voltage detect signal.
PWCDATA	O	This is the serial data signal to the power supply control IC (TPS2202A compatible).
PWCCLK	O	This is the clock signal to the power supply control IC (TPS2202A compatible).
PWCLATCH	O	This is the data latch signal to the power supply control IC (TPS2202A compatible).

★ **Table 2-5. Correspondence of Signal Names for Each PC Card Interface Mode (1/2)**

VRC4173		16-Bit PC Card	
Slot 1	Slot 2	Memory Card	I/O Card
C1A25	C2A25	A25	A25
C1A24	C2A24	A24	A24
C1A23	C2A23	A23	A23
C1A22	C2A22	A22	A22
C1A21	C2A21	A21	A21
C1A20	C2A20	A20	A20
C1A19	C2A19	A19	A19
C1A18	C2A18	A18	A18
C1A17	C2A17	A17	A17
C1A16	C2A16	A16	A16
C1A15	C2A15	A15	A15
C1A14	C2A14	A14	A14
C1A13	C2A13	A13	A13
C1A12	C2A12	A12	A12
C1A11	C2A11	A11	A11
C1A10	C2A10	A10	A10
C1A9	C2A9	A9	A9
C1A8	C2A8	A8	A8
C1A7	C2A7	A7	A7
C1A6	C2A6	A6	A6
C1A5	C2A5	A5	A5
C1A4	C2A4	A4	A4
C1A3	C2A3	A3	A3
C1A2	C2A2	A2	A2
C1A1	C2A1	A1	A1
C1A0	C2A0	A0	A0
C1D15	C2D15	D15	D15
C1D14	C2D14	D14	D14
C1D13	C2D13	D13	D13
C1D12	C2D12	D12	D12
C1D11	C2D11	D11	D11
C1D10	C2D10	D10	D10
C1D9	C2D9	D9	D9
C1D8	C2D8	D8	D8
C1D7	C2D7	D7	D7
C1D6	C2D6	D6	D6

★ **Table 2-5. Correspondence of Signal Names for Each PC Card Interface Mode (2/2)**

VRC4173		16-Bit PC Card	
Slot 1	Slot 2	Memory Card	I/O Card
C1D5	C2D5	D5	D5
C1D4	C2D4	D4	D4
C1D3	C2D3	D3	D3
C1D2	C2D2	D2	D2
C1D1	C2D1	D1	D1
C1D0	C2D0	D0	D0
CE12#	CE22#	CE2#	CE2#
CE11#	CE21#	CE1#	CE1#
OE1#	OE2#	OE#	OE#
WE1#	WE2#	WE#	WE#
IORD1#	IORD2#	–	IORD0#
IOWR1#	IOWR2#	–	IOWR0#
WAIT1#	WAIT2#	WAIT#	WAIT#
REG1#	REG2#	REG#	REG#
INPACK1#	INPACK2#	–	INPACK#
WP1	WP2	WP	IOIS16#
BVD11#	BVD21#	BVD1	STSCHG#
BVD12#	BVD22#	BVD2	SPKR#
READY1	READY2	READY	IREQ#
CD12#	CD22#	CD2#	CD2#
CD11#	CD21#	CD1#	CD1#
VS12#	VS22#	VS2#	VS2#
VS11#	VS21#	VS1#	VS1#
RESET1	RESET2	RESET	RESET

2.2.5 Keyboard interface signals

Table 2-6. Keyboard Interface Signals

Signal Name	I/O	Function
KPORT(7:0)/GPIO(15:8)	I/O	These are keyboard scan data input signals. They are used to scan for pressed keys on the keyboard. When these are used as KPORT signals, external pull-down resistors are required. When these are not used as KPORT signals, they can be used as general-purpose I/O ports.
KSCAN(7:0)/GPIO(7:0)	I/O	These are keyboard scan data output signals. The scan line is set to active when scanning for pressed keys on the keyboard. When these are not used as KSCAN signals, they can be used as general-purpose I/O ports.
KSCAN8/PS2CLK2	I/O	This is a keyboard scan data output signal. The scan line is set to active when scanning for pressed keys on the keyboard. When this is not used as a KSCAN signal, it can be used as a PS2CLK2 signal.
KSCAN9/PS2DATA2	I/O	This is a keyboard scan data output signal. The scan line is set to active when scanning for pressed keys on the keyboard. When this is not used as a KSCAN signal, it can be used as a PS2DATA2 signal.
KSCAN10/PS2CLK1	I/O	This is a keyboard scan data output signal. The scan line is set to active when scanning for pressed keys on the keyboard. When this is not used as a KSCAN signal, it can be used as a PS2CLK1 signal.
KSCAN11/PS2DATA1	I/O	This is a keyboard scan data output signal. The scan line is set to active when scanning for pressed keys on the keyboard. When this is not used as a KSCAN signal, it can be used as a PS2DATA1 signal.

2.2.6 PS/2 interface signals

Table 2-7. PS/2 Interface Signals

Signal Name	I/O	Function
PS2CLK1/KSCAN10	I/O	This is a PS/2 port 1 clock signal. When this is not used as a PS2CLK1 signal, it can be used as a KSCAN10 signal.
PS2CLK2/KSCAN8	I/O	This is a PS/2 port 2 clock signal. When this is not used as a PS2CLK2 signal, it can be used as a KSCAN8 signal.
PS2DATA1/KSCAN11	I/O	This is a PS/2 port 1 serial data signal. When this is not used as a PS2DATA1 signal, it can be used as a KSCAN11 signal.
PS2DATA2/KSCAN9	I/O	This is a PS/2 port 2 serial data signal. When this is not used as a PS2DATA2 signal, it can be used as a KSCAN9 signal.

2.2.7 Touch panel interface signals

Table 2-8. Touch Panel Interface Signals

Signal Name	I/O	Function
TPX(1:0)/GPIO(17:16)	I/O	These are touch panel I/O signals. The coordinates at which the touch panel was pressed are detected by applying voltage to the X coordinate and inputting the voltage of the Y coordinate. The TPX1 signal should be connected to the ADX signal externally. When these are not used as TPX signals, they can be used as general-purpose I/O ports.
TPY(1:0)/GPIO(19:18)	I/O	These are touch panel I/O signals. The coordinates at which the touch panel was pressed are detected by applying voltage to the Y coordinate and inputting the voltage of the X coordinate. The TPY1 signal should be connected to the ADY signal externally. The TYP1 signal is also used as a touch panel touch status interrupt request input to the PIU (see **9.3.2 PIUINTREG (base address + 0x0A4)**). When these are not used as TPY signals, they can be used as general-purpose I/O ports.
TPEN/GPIO20	I/O	This is the touch panel pull-down resistor enable signal. When this is not used as a TPEN signal, it can be used as a general-purpose I/O port.
ADX	I	This is an analog input signal. It should be connected to the TPX1 signal externally when the touch panel is used.
ADY	I	This is an analog input signal. It should be connected to the TPY1 signal externally when the touch panel is used.
ADIN	I	This is a general-purpose analog input signal.
AUDIOIN	I	This is an audio analog input signal.

2.2.8 Audio interface signal

Table 2-9. Audio Interface Signal

Signal Name	I/O	Function
AUDIOOUT	O	This is an audio analog output signal.

2.2.9 General-purpose I/O signals

Table 2-10. General-Purpose I/O Signals

Signal Name	I/O	Function
GPIO(7:0)/KSCAN(7:0)	I/O	These are general-purpose I/O signals. See **2.2.5 Keyboard interface signals**.
GPIO(15:8)/KPORT(7:0)	I/O	These are general-purpose I/O signals. See **2.2.5 Keyboard interface signals**.
GPIO(17:16)/TPX(1:0)	I/O	These are general-purpose I/O signals. See **2.2.7 Touch panel interface signals**.
GPIO(19:18)/TPY(1:0)	I/O	These are general-purpose I/O signals. See **2.2.7 Touch panel interface signals**.
GPIO20/TPEN	I/O	This is a general-purpose I/O signal. See **2.2.7 Touch panel interface signals**.

2.2.10 Interrupt interface signal

Table 2-11. Interrupt Interface Signal

Signal Name	I/O	Function
VRCINT	O	This is an integrated interrupt request signal to the CPU.

2.2.11 Clock interface signals

Table 2-12. Clock Interface Signals

Signal Name	I/O	Function
PCLK	I	This is the 33 MHz clock input signal. It is used internally as the PCI clock (PCICLK).
SCLK	I	This is the 9.216 MHz clock input signal.
CLK48MX1 Note	I	This is the 48 MHz oscillator's USB clock input signal.
CLK48MX2 Note	O	This is the 48 MHz oscillator's USB clock output signal.
CLK48M	O	This is the 48 MHz clock output signal.

Note For information about how to connect the clock oscillator to CLK48MX1 and CLK48MX2, see **2.4 Clock Oscillator Connection**.

2.2.12 Test interface signals

★ **Table 2-13. Test Interface Signals**

Signal Name	I/O	Function
TESTC	I	These are LSI evaluation test signals. Set TESTC to 0 and TEST(3:0) to 1000.
TEST(3:0)	I	

Table 2-14 lists the test modes.

★ **Table 2-14. Test Modes**

TESTC Signal	TEST(3:0) Signals	Test Mode
0	1000	Normal operation mode
Other than above		LSI evaluation mode

Caution Normal operation mode is set when TESTC = 0, TEST(3:0) = 1000. The LSI evaluation mode is set for all other settings. Operations are not guaranteed in the LSI evaluation mode.

2.2.13 Power supplies and grounds

Table 2-15. A/D Converter Power Supplies and Grounds

Signal Name	Function
$ADAV_{DD}$	This signal is for the dedicated analog power supply for the A/D converter.
ADAGND	This signal is for the dedicated analog ground for the A/D converter.
$ADDV_{DD}$	This signal is for the dedicated digital power supply for the A/D converter.
ADDGND	This signal is for the dedicated digital ground for the A/D converter.
$ADAV_{REF}P$	This signal is for the A/D converter's positive pole reference voltage (connected to the $ADAV_{DD}$ pin).
$ADAV_{REF}M$	This signal is for the A/D converter's negative pole reference voltage (connected to the ADAGND pin).

Table 2-16. D/A Converter Power Supply and Ground

Signal Name	Function
$DAAV_{DD}$	This signal is for the dedicated analog power supply for the D/A converter.
DAAGND	This signal is for the dedicated analog ground for the D/A converter.
$DAAV_{REF}$	This signal is for the D/A converter's reference voltage (connected to the $DAAV_{DD}$ pin).

Table 2-17. Digital Power Supplies and Grounds

Signal Name	Function
$V_{DD}2$	This signal is for the internal digital power supply.
GND2	This signal is for the internal digital ground.
$V_{DD}3$	This signal is for the I/O buffer's digital power supply.
GND3	This signal is for the I/O buffer's digital ground.

2.3 Pin Status and Recommended Connection Examples

Table 2-18 shows the status of the pins when the VRC4173 is reset (when the PCIRST# signal is at low level) and examples of recommended, logically required pin processing.

Table 2-18. Pin Status and Recommended Connection Examples (1/4)

Pin Name		I/O	Drive Capacity (mA)	Withstand Voltage (V)	Internal Processing	External Processing	Status After Reset
PCI interface	AD(31:0)	I/O	12	3.3	–	–	Hi-Z
	CBE(3:0)#	I/O	12	3.3	–	–	Hi-Z
	FRAME#	I/O	12	3.3	–	Pull-up	Hi-Z
	IRDY#	I/O	12	3.3	–	Pull-up	Hi-Z
	TRDY#	I/O	12	3.3	–	Pull-up	Hi-Z
	STOP#	I/O	12	3.3	–	Pull-up	Hi-Z
	DEVSEL#	I/O	12	3.3	–	Pull-up	Hi-Z
	PAR	I/O	12	3.3	–	–	Hi-Z
	PERR#	I/O	12	3.3	–	Pull-up	Hi-Z
★	CLKRUN#	I/O	12	3.3	–	Pull-up	Hi-Z
	REQ#	O	18	3.3	–	–	Hi-Z
	GNT#	I	–	3.3	–	–	–
	SERR#	O	6	3.3	Open drain	Pull-up	Hi-Z
	IDSEL	I	–	3.3	–	–	–
	PCIRST#	I	–	3.3	–	–	–
USBU	DP1, DP2	I/O	–	5	–	–	Hi-Z
	DN1, DN2	I/O	–	5	–	–	Hi-Z
	OCI1, OCI2	I	–	3.3	–	**Note**	–
	PPON1, PPON2	O	3	3.3	–	–	Hi-Z
	RCVBE	I	–	3.3	–	**Note**	–
AC97U	SYNC	O	6	3.3	–	–	0
	BCLK	I	–	3.3	–	**Note**	–
	SDATAOUT	O	6	3.3	–	–	0
	SDATAIN	I	–	3.3	–	**Note**	–
	ACLINKRST#	O	6	3.3	–	–	0

Note This should be fixed at low level when the relevant unit is unused.

Remark 0: Low level, Hi-Z: High impedance

Table 2-18. Pin Status and Recommended Connection Examples (2/4)

Pin Name		I/O	Drive Capacity (mA)	Withstand Voltage (V)	Internal Processing[Note 1]	External Processing	Status After Reset
CARDU1	C1A(25:23)	I/O	3	5	–	–	Hi-Z
	C1A(22:20)	I/O	3	5	Pull-up	–	Hi-Z
	C1A(19:17)	I/O	3	5	–	–	Hi-Z
	C1A16	I/O	9	3.3	–	–	Hi-Z
	C1A(15:14)	I/O	3	5	Pull-up	–	Hi-Z
	C1A(13:0)	I/O	3	5	–	–	Hi-Z
	C1D(15:0)	I/O	3	5	–	–	Hi-Z
	CE11#	I/O	3	5	–	–	Hi-Z
	CE12#	I/O	3	5	–	–	Hi-Z
	OE1#	I/O	3	5	–	–	Hi-Z
	WE1#	O	3	5	–	–	Hi-Z
	WP1	I/O	3	5	Pull-up	–	Hi-Z
	VS11#	I/O	3	5	Pull-up	–	0
	VS12#	I/O	3	5	Pull-up	–	0
	IORD1#	I/O	3	5	–	–	Hi-Z
	IOWR1#	I/O	3	5	–	–	Hi-Z
	REG1#	I/O	3	5	–	–	Hi-Z
★	RESET1	O	3	5	–	–	0[Note 2]
★	CD11#	I	–	5	–	Pull-up	–
★	CD12#	I	–	5	–	Pull-up	–
	WAIT1#	I	–	5	–	Pull-up[Note 3]	–
	INPACK1#	I	–	5	–	Pull-up[Note 3]	–
	READY1	I	–	5	–	Pull-up[Note 3]	–
	BVD11#	I	–	5	Pull-up/ pull-down	–	–
	BVD12#	I	–	5	–	Pull-up[Note 3]	–

Notes 1. The switching of the pull-up/pull-down resistors and their on/off status are automatically switched by the internal sequencer.

★ **2.** Card resetting is controlled by writing to registers.

3. Set a pull-up resistor for the external PC card power supply.

Remark 0: Low level, Hi-Z: High impedance

Table 2-18. Pin Status and Recommended Connection Examples (3/4)

Pin Name		I/O	Drive Capacity (mA)	Withstand Voltage (V)	Internal Processing[Note 1]	External Processing	Status After Reset
CARDU2	C2A(15:23)	I/O	3	5	–	–	Hi-Z
	C2A(22:20)	I/O	3	5	Pull-up	–	Hi-Z
	C2A(19:17)	I/O	3	5	–	–	Hi-Z
	C2A16	I/O	9	3.3	–	–	Hi-Z
	C2A(15:14)	I/O	3	5	Pull-up	–	Hi-Z
	C2A(13:0)	I/O	3	5	–	–	Hi-Z
	C2D(15:0)	I/O	3	5	–	–	Hi-Z
	CE21#	I/O	3	5	–	–	Hi-Z
	CE22#	I/O	3	5	–	–	Hi-Z
	OE2#	I/O	3	5	–	–	Hi-Z
	WE2#	O	3	5	–	–	Hi-Z
	WP2	I/O	3	5	Pull-up	–	Hi-Z
	VS21#	I/O	3	5	Pull-up	–	0
	VS22#	I/O	3	5	Pull-up	–	0
	IORD2#	I/O	3	5	–	–	Hi-Z
	IOWR2#	I/O	3	5	–	–	Hi-Z
	REG2#	I/O	3	5	–	–	Hi-Z
★	RESET2	O	3	5	–	–	0[Note 2]
★	CD21#	I	–	5	–	Pull-up	–
★	CD22#	I	–	5	–	Pull-up	–
	WAIT2#	I	–	5	–	Pull-up[Note 3]	–
	INPACK2#	I	–	5	–	Pull-up[Note 3]	–
	READY2	I	–	5	–	Pull-up[Note 3]	–
	BVD21#	I	–	5	Pull-up/ pull-down	–	–
	BVD22#	I	–	5	–	Pull-up[Note 3]	–
Common to CARDU1 and CARDU2	PWCDATA	O	3	5	–	–	0
	PWCCLK	O	3	5	–	–	0
	PWCLATCH	O	3	5	–	–	0

Notes 1. The switching of the pull-up/pull-down resistors and their on/off status are automatically switched by the internal sequencer.

★ **2.** Card resetting is controlled by writing to registers.

3. Set a pull-up resistor for the external PC card power supply.

Remark 0: Low level, Hi-Z: High impedance

Table 2-18. Pin Status and Recommended Connection Examples (4/4)

Pin Name		I/O	Drive Capacity (mA)	Withstand Voltage (V)	Internal Processing	External Processing	Status After Reset
KIU/PS2U	KPORT(7:0)/ GPIO(15:8)	I/O	3	3.3	–	Pull-up[Note 1]	Hi-Z
	KSCAN11/PS2DATA1	I/O	3	5	–	–[Note 2]	Hi-Z
	KSCAN10/PS2CLK1	I/O	3	5	–	–[Note 2]	Hi-Z
	KSCAN9/PS2DATA2	I/O	3	5	–	–[Note 2]	Hi-Z
	KSCAN8/PS2CLK2	I/O	3	5	–	–[Note 2]	Hi-Z
	KSCAN(7:0)/ GPIO(7:0)	I/O	3	3.3	–	–	Hi-Z
PIU	TPX(1:0)/ GPIO(17:16)	I/O	18	3.3	Slew rate buffer	–	1
	TPY(1:0)/ GPIO(19:18)	I/O	18	3.3	Slew rate buffer	–	Hi-Z
	TPEN/GPIO20	I/O	3	3.3	–	Pull-down[Note 3]	0
	ADX	I	–	3.3	–	–	–
	ADY	I	–	3.3	–	–	–
	ADIN	I	–	3.3	–	–	–
	AUDIOIN	I	–	3.3	–	–	–
AIU	AUDIOOUT	O	–	3.3	–	**Note 4**	0
Interrupt	VRCINT	O	3	3.3	–	–	0
Clock	CLK48MX1	I	–	3.3	–	Resonator	–
	CLK48MX2	O	–	3.3	–	Resonator	–
	PCLK	I	–	3.3	–	–	–
	SCLK	I	–	3.3	–	9.126 MHz clock	–
	CLK48M	O	3	3.3	–	–	1
★ Test	TESTC	I	–	3.3	–	GND connection	–
★	TEST(3:0)	I	–	3.3	–	Fixed at 1000	–

Notes 1. When these pins are used as KPORT(7:0) signals, an external pull-down resistor is required.

★ **2.** When the PS2U is used, set a pull-up resistor.

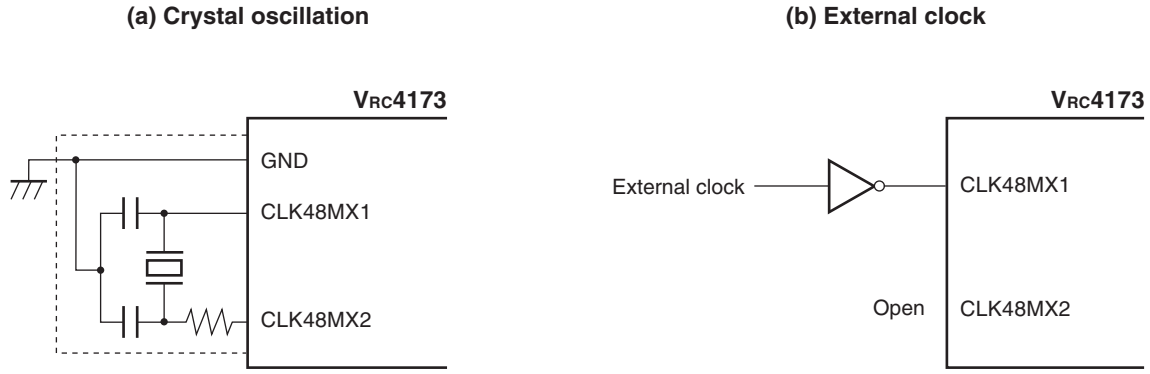
3. When the PIU is used, a switched pull-down resistor is required for the TPEN signal.

★ **4.** As the AUDIOOUT pin is a high-resistance output, power cannot be supplied. Connect a JFET input type operational amplifier (input bias below 100 nA).

Remark 0: Low level, 1: High level, Hi-Z: High impedance

2.4 Clock Oscillator Connection

Figure 2-1. External Circuit of Clock Oscillator



Cautions 1. When using a clock oscillator, wire as follows in the area enclosed by the broken lines in the above figure to avoid an adverse effect from wiring capacitance.

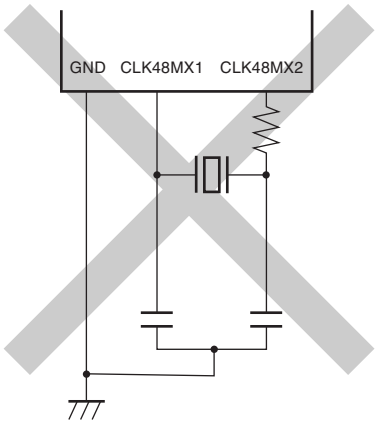
- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines. Also, do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as GND. Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

2. When inputting an external clock, make sure that no load such as the wiring capacitance is applied to the CLK48MX2 pin.

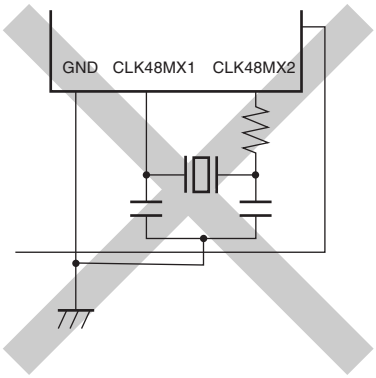
3. When using an external clock, the oscillator stop function cannot be used according to the CMUCLKMSK register of the CMU. If this function is used, operation cannot be guaranteed.

Figure 2-2. Examples of Improperly Connected Resonators

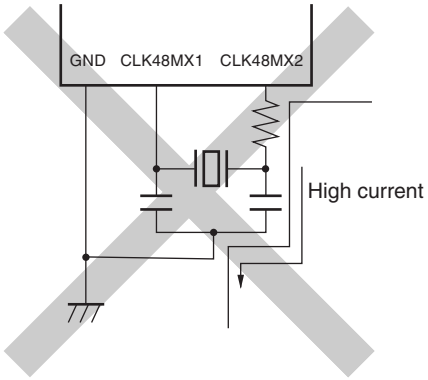
(a) Connection circuit wiring is too long



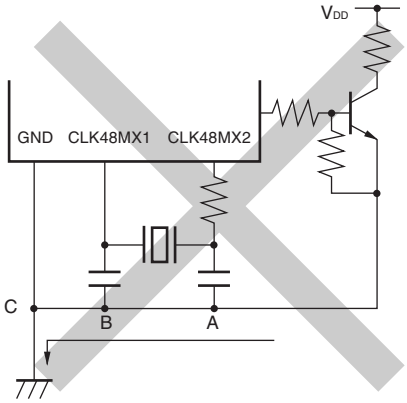
(b) Signal lines are crossed



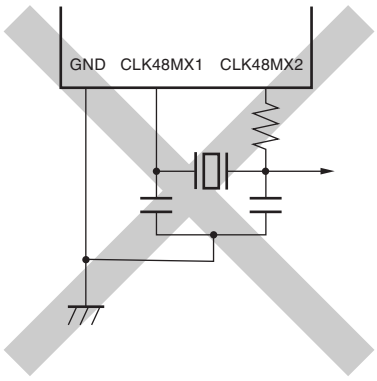
(c) A high fluctuating current flows near the signal line



(d) A current flows over the ground line of the oscillator (the potentials of points A, B, and C change)



(e) A signal is fetched



CHAPTER 3 BCU (BUS CONTROL UNIT)

3.1 General

The BCU controls the PIB (Peripheral Internal Bus), which is an internal bus.
The DMAAU, DCU, CMU, ICU, GIU, PIU, AIU, KIU, PS2CH1, PS2CH2, and ADU are connected to the PIB.

3.2 Register Set

Table 3-1 lists the BCU configuration registers.

Table 3-1. BCU Configuration Registers

Offset Address	R/W	Register Symbol	Function
0x00 to 0x01	R	VID	Vendor ID register
0x02 to 0x03	R	DID	Device ID register
0x04 to 0x05	R/W	PCICMD	PCI command register
0x06 to 0x07	R/W	PCISTS	PCI device status register
0x08	R	RID	Revision ID register
0x09 to 0x0B	R	CLASSC	Class code register
0x0C	R	CACHELS	Cache line size register
0x0D	R/W	MLT	Master latency timer register
0x0E	R	HEDT	Header type register
0x0F	R	BIST	Built-in self-test register
0x10 to 0x13	R/W	BADR	PIB I/O base address register
0x14 to 0x2B	–	–	Reserved
0x2C to 0x2D	R/W	SUBVID	Subsystem vendor ID register
0x2E to 0x2F	R/W	SUBID	Subsystem ID register
0x30 to 0x3B	–	–	Reserved
0x3C	R/W	INTL	Interrupt line register
0x3D	R	INTP	Interrupt pin register
0x3E	R	MIN_GNT	Burst cycle minimum request time register
0x3F	R	MAX_LAT	Bus usage right request frequency register
0x40	R/W	BUSCNT	PIB bus control register
0x41	R/W	IDSELNUM	PC card IDSEL selection register

These registers are described in detail below.

3.2.1 VID (offset address: 0x00 to 0x01)

Bit	15	14	13	12	11	10	9	8
Name	VID15	VID14	VID13	VID12	VID11	VID10	VID9	VID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	1	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	VID7	VID6	VID5	VID4	VID3	VID2	VID1	VID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	1	1	0	0	1	1

Bit	Name	Function
15:0	VID(15:0)	Vendor ID 0x1033: NEC

3.2.2 DID (offset address: 0x02 to 0x03)

Bit	15	14	13	12	11	10	9	8
Name	DID15	DID14	DID13	DID12	DID11	DID10	DID9	DID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DID7	DID6	DID5	DID4	DID3	DID2	DID1	DID0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	1	0	0	1	0	1

Bit	Name	Function
15:0	DID(15:0)	Device ID 0x00A5: BCU

3.2.3 PCICMD (offset address: 0x04 to 0x05)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	FBTB_EN	SERR_EN
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AD_STEP	PERR_EN	VGA_P_ SNOOP	MEMW_ INV_EN	SP_CYC	MASTER_ EN	MEM_EN	IO_EN
R/W	R	R/W	R	R	R	R/W	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	FBTB_EN	Enables/disables fast Back to Back. This function is not supported by the BCU.
8	SERR_EN	Enables/disables SERR# signal output. 1: Enable The SERR# signal is set to active if an address parity error is detected and the PERR_EN bit is 1. 0: Disable
7	AD_STEP	Enables/disables address/data stepping. This function is not supported by the BCU.
6	PERR_EN	Enables/disables parity error. 1: Enable output of the PERR# signal The PERR# signal is set to active if a data parity error is detected. The SERR# signal is set to active if an address parity error is detected and the SERR_EN bit is 1. 0: Disable output of the PERR# signal
5	VGA_P_SNOOP	VGA palette snoop. This function is not supported by the BCU.
4	MEMW_INV_EN	Enables/disables memory write and invalidate. This function is not supported by the BCU.
3	SP_CYC	Special cycle. This function is not supported by the BCU.
2	MASTER_EN	Controls bus master operation. 1: Operate as bus master on the PCI bus. 0: Do not operate as bus master on the PCI bus.
1	MEM_EN	Controls memory space. This function is not supported by the BCU.
0	IO_EN	Controls I/O space. 1: Respond to an I/O access to the PIB. 0: Do not respond to an I/O access to the PIB.

3.2.4 PCISTS (offset address: 0x06 to 0x07)

Bit	15	14	13	12	11	10	9	8
Name	DETECT_ PERR	SIG_SERR	RV_ MABORT	RV_ TABORT	SIG_ TABOT	DEVSEL1	DEVSEL0	DETECT_ D_PERR
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	FBTB_CAP	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	DETECT_PERR	Data and address parity error detection. Cleared to 0 when 1 is written. 1: Detected 0: Not detected
14	SIG_SERR	SERR# signal status. Cleared to 0 when 1 is written. 1: Active 0: Inactive
13	RV_MABORT	Master abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
12	RV_TABORT	Target abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
11	SIG_TABOT	Target abort reporting. Cleared to 0 when 1 is written. 1: Reported 0: Not reported
10:9	DEVSEL(1:0)	DEVSEL# timing 01: Medium speed
8	DETECT_D_PERR	Set to 1 when the following three conditions are satisfied. Cleared to 0 when 1 is written. • The BCU is the master of the bus cycle in which the data parity error occurred. • Either the BCU set the PERR# signal to active or the BCU detected that the PERR# signal became active due to the target. • The PERR_EN bit of the PCICMD register has been set to 1.
7	FBTB_CAP	Response to fast Back to Back. This is fixed at 0 (disabled).
6:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

3.2.5 RID (offset address: 0x08)

Bit	7	6	5	4	3	2	1	0
Name	RID7	RID6	RID5	RID4	RID3	RID2	RID1	RID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	RID(7:0)	Revision ID

3.2.6 CLASSC (offset address: 0x09 to 0x0B)

Bit	23	22	21	20	19	18	17	16
Name	CLASSC23	CLASSC22	CLASSC21	CLASSC20	CLASSC19	CLASSC18	CLASSC17	CLASSC16
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	1	1	0

Bit	15	14	13	12	11	10	9	8
Name	CLASSC15	CLASSC14	CLASSC13	CLASSC12	CLASSC11	CLASSC10	CLASSC9	CLASSC8
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CLASSC7	CLASSC6	CLASSC5	CLASSC4	CLASSC3	CLASSC2	CLASSC1	CLASSC0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
23:0	CLASSC(23:0)	Class code 0x068000: Bridge device

3.2.7 CACHELS (offset address: 0x0C)

Bit	7	6	5	4	3	2	1	0
Name	CACHELS7	CACHELS6	CACHELS5	CACHELS4	CACHELS3	CACHELS2	CACHELS1	CACHELS0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	CACHELS(7:0)	Sets the cache line size. This function is not supported by the BCU.

3.2.8 MLT (offset address: 0x0D)

Bit	7	6	5	4	3	2	1	0
Name	MLT7	MLT6	MLT5	MLT4	MLT3	MLT2	MLT1	MLT0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	MLT(7:4)	Sets the latency timer. 1111: 30 PCLK (900 ns) : 0010: 17 PCLK (510 ns) 0001: 16 PCLK (480 ns) 0000: 0 PCLK (0 ns)
3:0	MLT(3:0)	Write 0 to these bits. 0 is returned after a read.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

3.2.9 HEDT (offset address: 0x0E)

Bit	7	6	5	4	3	2	1	0
Name	HEDT7	HEDT6	HEDT5	HEDT4	HEDT3	HEDT2	HEDT1	HEDT 0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	0	0	0	0	0

Bit	Name	Function
7:0	HEDT(7:0)	Header type 0x80: This is a multifunction device and offset address 0x10 to 0x3F of the configuration register are default settings.

3.2.10 BIST (offset address: 0x0F)

Bit	7	6	5	4	3	2	1	0
Name	BIST7	BIST6	BIST5	BIST4	BIST3	BIST2	BIST1	BIST0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	BIST(7:0)	Built-in self-test. This function is not supported by the BCU.

3.2.11 BADR (offset address: 0x10 to 0x13)

Bit	31	30	29	28	27	26	25	24
Name	BADR31	BADR30	BADR29	BADR28	BADR27	BADR26	BADR25	BADR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	BADR23	BADR22	BADR21	BADR20	BADR19	BADR18	BADR17	BADR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	BADR15	BADR14	BADR13	BADR12	BADR11	BADR10	BADR9	BADR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	BADR7	BADR6	BADR5	BADR4	BADR3	BADR2	BADR1	BADR0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
31:9	BADR(31:9)	Sets the PIB I/O base address.
8:1	BADR(8:1)	Write 0 to these bits. 0 is returned after a read.
0	BADR0	Write 1 to this bit. 1 is returned after a read.

3.2.12 SUBVID (offset address: 0x2C to 0x2D)

Bit	15	14	13	12	11	10	9	8
Name	SUBVID15	SUBVID14	SUBVID13	SUBVID12	SUBVID11	SUBVID10	SUBVID9	SUBVID8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SUBVID7	SUBVID6	SUBVID5	SUBVID4	SUBVID3	SUBVID2	SUBVID1	SUBVID0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SUBVID(15:0)	Subsystem vendor ID This is a vendor identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

3.2.13 SUBID (offset address: 0x2E to 0x2F)

Bit	15	14	13	12	11	10	9	8
Name	SUBID15	SUBID14	SUBID13	SUBID12	SUBID11	SUBID10	SUBID9	SUBID8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SUBID7	SUBID6	SUBID5	SUBID4	SUBID3	SUBID2	SUBID1	SUBID0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SUBID(15:0)	Subsystem ID This is a controller identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

3.2.14 INTL (offset address: 0x3C)

Bit	7	6	5	4	3	2	1	0
Name	INTL7	INTL6	INTL5	INTL4	INTL3	INTL2	INTL1	INTL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	INTL(7:0)	Sets the interrupt request line. Since this function is not supported by the BCU, settings for these bits are invalid. Use the ICU to set the interrupt request line.

3.2.15 INTP (offset address: 0x3D)

Bit	7	6	5	4	3	2	1	0
Name	INTP7	INTP6	INTP5	INTP4	INTP3	INTP2	INTP1	INTP0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	INTP(7:0)	PCI interrupt pin 0x01: Serial

3.2.16 MIN_GNT (offset address: 0x3E)

Bit	7	6	5	4	3	2	1	0
Name	MIN_GNT7	MIN_GNT6	MIN_GNT5	MIN_GNT4	MIN_GNT3	MIN_GNT2	MIN_GNT1	MIN_GNT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MIN_GNT(7:0)	Burst cycle minimum request time. These bits are fixed at 0x00.

3.2.17 MAX_LAT (offset address: 0x3F)

Bit	7	6	5	4	3	2	1	0
Name	MAX_LAT7	MAX_LAT6	MAX_LAT5	MAX_LAT4	MAX_LAT3	MAX_LAT2	MAX_LAT1	MAX_LAT0
R/W	R	R	R	R	R	R	R	R
After reset	0	1	0	1	1	0	1	0

Bit	Name	Function
7:0	MAX_LAT(7:0)	Maximum delay time until a response is returned when the PCI bus usage right is requested. These bits are fixed at 0x56.

3.2.18 BUSCNT (offset address: 0x40)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	POSTON
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	POSTON	Enables/disables PCI I/O post write cycle. 1: Enable 0: Disable

3.2.19 IDSELNUM (offset address: 0x41)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	C2IDSEL1	C2IDSEL0	RFU	RFU	C1IDSEL1	C1IDSEL0
R/W	R	R	R/W	R/W	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5:4	C2IDSEL(1:0)	Selects the IDSEL signal of CARDU2 (PC card channel 2). 11: Reserved 10: Selects the AD25 signal as the IDSEL signal. 01: Selects the AD19 signal as the IDSEL signal. 00: Selects the AD13 signal as the IDSEL signal. When 11 is set, it is treated as if 00 were set (the AD13 signal is selected).
3:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	C1IDSEL(1:0)	Selects the IDSEL signal of CARDU1 (PC card channel 1). 11: Reserved 10: Selects the AD24 signal as the IDSEL signal. 01: Selects the AD18 signal as the IDSEL signal. 00: Selects the AD12 signal as the IDSEL signal. When 11 is set, it is treated as if 00 were set (the AD12 signal is selected).

The AD12 or AD13 signal is selected by default and connected for the IDSEL signal of CARDU1 or CARDU2, respectively.

When this address bit has been used for the IDSEL signal of another PCI device in the system, change it to a different address bit by setting the C1IDSEL(1:0) area or C2IDSEL(1:0) area.

CHAPTER 4 DMAAU (DMA ADDRESS UNIT)

4.1 General

The DMAAU register controls the DMA addresses for the AIU.

The DMA channel used for each unit can set a DMA start address as any half-word address in the physical address from 0x0000 0000 to 0xFFFF FFFE, and is retained in DRAM as a 2 KB block that starts at the address which is generated by masking the lower 10 bits of the DMA start address.

Caution DMA operations are not guaranteed if an address overlaps with another DMA buffer.

After a DMA start address is set to the DMA base address register, the VRC4173 performs DMA transfer using the registers of DMAAU as below.

(1) When the DMA start address is included in the first page of the DMA space

<1> The VRC4173 starts a DMA transfer after writing the start address to the DMA address register.

<2> When the DMA transfer reaches the first page boundary, the VRC4173 adds 1 KB to the contents of the DMA base address register, writes the value to the DMA address register, and continues the DMA transfer.

<3> When the DMA transfer reaches the second page boundary, the VRC4173 writes the contents of the DMA base address register to the DMA address register and continues the DMA transfer.

<4> The VRC4173 repeats <2> and <3> until all the data is transferred.

(2) When the DMA start address is included in the second page of the DMA space

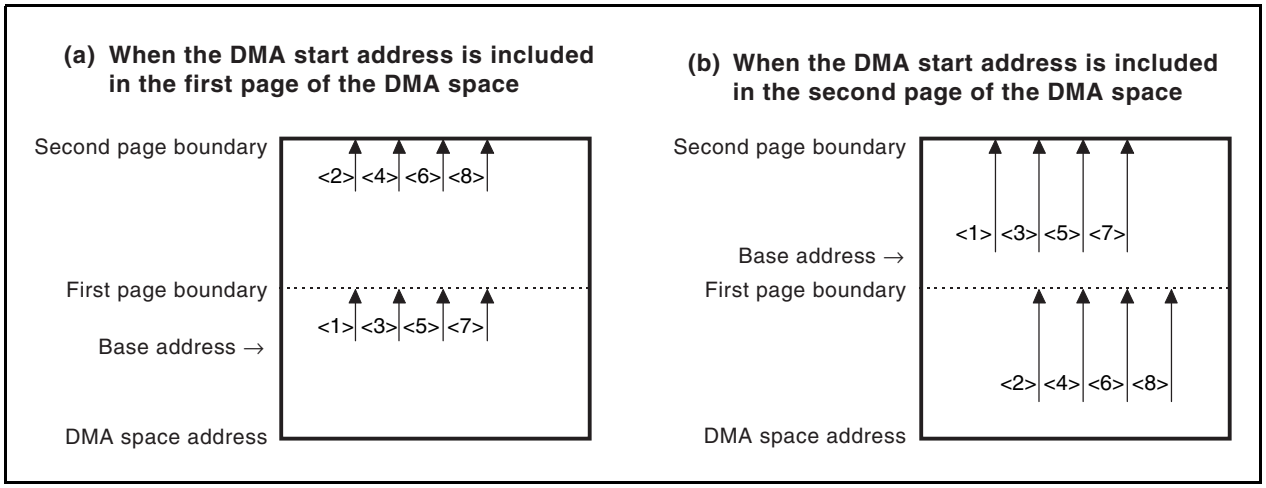
<1> The VRC4173 starts a DMA transfer after writing the start address to the DMA address register.

<2> When the DMA transfer reaches the second page boundary, the VRC4173 subtracts 1 KB from the contents of the DMA base address register, writes the value to the DMA address register, and continues the DMA transfer.

<3> When the DMA transfer reaches the first page boundary, the VRC4173 writes the contents of the DMA base address register to the DMA address register and continues the DMA transfer.

<4> The VRC4173 repeats <2> and <3> until all the data is transferred.

Figure 4-1. DMA Space Used in DMA Transfers



4.2 Register Set

Table 4-1 lists the DMAAU registers.

Table 4-1. DMAAU Registers

Address	R/W	Register Symbol	Function
BASE + 0x000	R/W	AIUIBALREG	AIU IN DMA Base Address Register Low
BASE + 0x002	R/W	AIUIBAHREG	AIU IN DMA Base Address Register High
BASE + 0x004	R/W	AIUIALREG	AIU IN DMA Address Register Low
BASE + 0x006	R/W	AIUIAHREG	AIU IN DMA Address Register High
BASE + 0x008	R/W	AIUOBALREG	AIU OUT DMA Base Address Register Low
BASE + 0x00A	R/W	AIUOBAHREG	AIU OUT DMA Base Address Register High
BASE + 0x00C	R/W	AIUOALREG	AIU OUT DMA Address Register Low
BASE + 0x00E	R/W	AIUOAHREG	AIU OUT DMA Address Register High

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

4.2.1 AIU IN DMA base address registers

(1) AIUIBALREG (base address + 0x000)

Bit	15	14	13	12	11	10	9	8
Name	AIUIBA15	AIUIBA14	AIUIBA13	AIUIBA12	AIUIBA11	AIUIBA10	AIUIBA9	AIUIBA8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AIUIBA7	AIUIBA6	AIUIBA5	AIUIBA4	AIUIBA3	AIUIBA2	AIUIBA1	AIUIBA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:1	AIUIBA(15:1)	DMA base address 15:1 for AIU input
0	AIUIBA0	DMA base address 0 for AIU input Write 0 to this bit. 0 is returned after a read.

(2) AIUIBAHREG (base address + 0x002)

Bit	15	14	13	12	11	10	9	8
Name	AIUIBA31	AIUIBA30	AIUIBA29	AIUIBA28	AIUIBA27	AIUIBA26	AIUIBA25	AIUIBA24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	AIUIBA23	AIUIBA22	AIUIBA21	AIUIBA20	AIUIBA19	AIUIBA18	AIUIBA17	AIUIBA16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15:0	AIUIBA(31:16)	DMA base address 31:16 for AIU input

AIUIBALREG and AIUIBAHREG registers are used to set the base addresses for the DMA channel used for audio input (recording). The addresses set to these registers become DMA transfer start addresses.

The DMA channel used for audio input is retained in DRAM as a 2 KB buffer that starts at the address which is generated by masking the lower 10 bits of the DMA start address.

4.2.2 AIU IN DMA address registers

(1) AIUIALREG (base address + 0x004)

Bit	15	14	13	12	11	10	9	8
Name	AIUIA15	AIUIA14	AIUIA13	AIUIA12	AIUIA11	AIUIA10	AIUIA9	AIUIA8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AIUIA7	AIUIA6	AIUIA5	AIUIA4	AIUIA3	AIUIA2	AIUIA1	AIUIA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	AIUIA(15:0)	Next DMA address 15:0 to be accessed for AIU input channel

(2) AIUIAHREG (base address + 0x006)

Bit	15	14	13	12	11	10	9	8
Name	AIUIA31	AIUIA30	AIUIA29	AIUIA28	AIUIA27	AIUIA26	AIUIA25	AIUIA24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	AIUIA23	AIUIA22	AIUIA21	AIUIA20	AIUIA19	AIUIA18	AIUIA17	AIUIA16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15:0	AIUIA(31:16)	Next DMA address 31:16 to be accessed for AIU input channel

4.2.3 AIU OUT DMA base address registers

(1) AIUOBALREG (base address + 0x008)

Bit	15	14	13	12	11	10	9	8
Name	AIUOBA15	AIUOBA14	AIUOBA13	AIUOBA12	AIUOBA11	AIUOBA10	AIUOBA9	AIUOBA8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AIUOBA7	AIUOBA6	AIUOBA5	AIUOBA4	AIUOBA3	AIUOBA2	AIUOBA1	AIUOBA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:1	AIUOBA(15:1)	DMA base address 15:1 for AIU output
0	AIUOBA0	DMA base address 0 for AIU output Write 0 to this bit. 0 is returned after a read.

(2) AIUOBAHREG (base address + 0x00A)

Bit	15	14	13	12	11	10	9	8
Name	AIUOBA31	AIUOBA30	AIUOBA29	AIUOBA28	AIUOBA27	AIUOBA26	AIUOBA25	AIUOBA24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	AIUOBA23	AIUOBA22	AIUOBA21	AIUOBA20	AIUOBA19	AIUOBA18	AIUOBA17	AIUOBA16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15:0	AIUOBA(31:16)	DMA base address 31:16 for AIU output

AIUOBALREG and AIUOBAHREG registers are used to set the base addresses for the DMA channel used for audio output (playback). The addresses set to these registers become DMA transfer start addresses.

The DMA channel used for audio output is retained in DRAM as a 2 KB buffer that starts at the address which is generated by masking the lower 10 bits of the DMA start address.

4.2.4 AIU OUT DMA address registers

(1) AIUOALREG (base address + 0x00C)

Bit	15	14	13	12	11	10	9	8
Name	AIUOA15	AIUOA14	AIUOA13	AIUOA12	AIUOA11	AIUOA10	AIUOA9	AIUOA8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AIUOA7	AIUOA6	AIUOA5	AIUOA4	AIUOA3	AIUOA2	AIUOA1	AIUOA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	AIUOA(15:0)	Next DMA address 15:0 to be accessed for AIU output channel

(2) AIUOAHREG (base address + 0x00E)

Bit	15	14	13	12	11	10	9	8
Name	AIUOA31	AIUOA30	AIUOA29	AIUOA28	AIUOA27	AIUOA26	AIUOA25	AIUOA24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	AIUOA23	AIUOA22	AIUOA21	AIUOA20	AIUOA19	AIUOA18	AIUOA17	AIUOA16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15:0	AIUOA(31:16)	Next DMA address 31:16 to be accessed for AIU output channel

CHAPTER 5 DCU (DMA CONTROL UNIT)

5.1 General

The DCU register is used for DMA control. Specifically, it controls acknowledgment from the BCU that handles bus arbitration and DMA requests from the on-chip peripheral I/O unit (AIU). It also controls DMA enable/disable settings.

5.2 DMA Priority Control

When a conflict occurs between DMA requests sent from on-chip peripheral I/O unit, the following priority levels are used to resolve the conflict. These priority levels cannot be changed.

Table 5-1. DMA Priority Levels

Priority Level	Type of DMA Operation
High	Audio input (recording)
Low	Audio output (playback)

5.3 Register Set

Table 5-2 lists the DCU registers.

Table 5-2. DCU Registers

Address	R/W	Register Symbol	Function
BASE + 0x020	R/W	DMARSTREG	DMA Reset Register
BASE + 0x022	R	DMAIDLEREG	DMA Sequencer Status Register
BASE + 0x024	R/W	DMASENREG	DMA Sequencer Enable Register
BASE + 0x026	R/W	DMAMSKREG	DMA Mask Register
BASE + 0x028	R	DMAREQREG	DMA Request Register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

5.3.1 DMARSTREG (base address + 0x020)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	DMARST
R/W	R	R	R	R	R	R	R	W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	DMARST	Reset DMA controller 1: Reset 0: Normal

This register is used to reset the DMA controller.

Reset the DMA controller after confirming that the DMA is not executed (idle status).

5.3.2 DMAIDLEREG (base address + 0x022)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	DMAISTAT
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
15:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	DMAISTAT	Display DMA sequencer status 1: Idle status 0: Sequencer busy

This register is used to display the DMA sequencer status.

5.3.3 DMASENREG (base address + 0x024)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	DMASEN
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	DMASEN	Enable DMA sequencer 1: Enable 0: Disable

This register is used to enable/disable the DMA sequencer.

5.3.4 DMAMSKREG (base address + 0x026)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	DMAMSKAIN	DMAMSKAOUT	RFU	RFU
R/W	R	R	R	R	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	DMAMSKAIN	Audio input DMA transfer enable/disable 1: Enable 0: Disable
2	DMAMSKAOUT	Audio output DMA transfer enable/disable 1: Enable 0: Disable
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

This register is used to enable/disable various types of DMA transfers.

The DMA transfer enable bits should be set when the units that receive DMA service have been stopped or when there are no pending DMA requests. If any of the above bits are set to a unit while a DMA request is pending for that unit, the operation of the VRC4173 will be undefined.

5.3.5 DMAREQREG (base address + 0x028)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	DRQAIN	DRQAOUT	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	DRQAIN	Audio input DMA transfer request 1: Request pending 0: No request
2	DRQAOUT	Audio output DMA transfer request 1: Request pending 0: No request
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

This register is used to indicate whether or not there are any DMA transfer requests.

CHAPTER 6 CMU (CLOCK MASK UNIT)

6.1 General

As various input clocks are supplied from the CPU to each unit, a masking method enables power consumption to be curtailed in units that are not used.

The units for which this masking method are used are the USBU, CARDU1, CARDU2, KIU, PIU, AIU, PS2CH1, PS2CH2, and AC97U units.

The basic functions are described below.

★ • Control of PCICLK (internal) supplied to USBU, CARDU1, CARDU2, and AC97U
★ • Control of TClock (internal clock synchronized with PCICLK) supplied to KIU, PIU, AIU, PS2CH1, and PS2CH2
• Control of 48 MHz clock supplied to USBU and CLK48M output pin
• Control of on-chip 48 MHz oscillator

The initial value is “0”, which specifies masking all supplied clocks. No clock is supplied unless the CPU writes “1” to the register.

6.2 Register Set

Table 6-1 lists the CMU registers.

Table 6-1. CMU Registers

Address	R/W	Register Symbol	Function
BASE + 0x040	R/W	CMUCLKMSK	CMU Clock Mask Register
BASE + 0x042	R/W	CMUSRST	CMU Soft Reset Register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

6.2.1 CMUCLKMSK (base address + 0x040)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	MSK 48MOSC	MSK 48MPIN	MSK 48MUSB	RFU	MSK AC97
R/W	R	R	R	R/W	R/W	R/W	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MSK CARD2	MSK CARD1	MSKUSB	MSK PS2CH2	MSK PS2CH1	MSKAIU	MSKKIU	MSKPIU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:13	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
12	MSK48MOSC	Control on-chip 48 MHz oscillator 1: Oscillation 0: Stop
11	MSK48MPIN	Supply/mask 48 MHz clock to external pin (CLK48M) 1: Supply 0: Mask
10	MSK48MUSB	Supply/mask 48 MHz clock to USBU unit 1: Supply 0: Mask
9	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
8	MSKAC97	Supply/mask PCICLK to AC97U unit 1: Supply 0: Mask
7	MSKCARD2	Supply/mask PCICLK to CARDU1 unit 1: Supply 0: Mask

(2/2)

Bit	Name	Function
6	MSKCARD1	Supply/mask PCICLK to CARDU2 unit 1: Supply 0: Mask
5	MSKUSB	Supply/mask PCICLK to USBU unit 1: Supply 0: Mask
4	MSKPS2CH2	Supply/mask TClock to PS2CH2 unit 1: Supply 0: Mask
3	MSKPS2CH1	Supply/mask TClock to PS2CH1 unit 1: Supply 0: Mask
2	MSKAIU	Supply/mask TClock to AIU unit 1: Supply 0: Mask
1	MSKKIU	Supply/mask TClock to KIU unit 1: Supply 0: Mask
0	MSKPIU	Supply/mask TClock to PIU unit 1: Supply 0: Mask

This register is used to mask the clocks that are supplied to the AC97U, CARDU1, CARDU2, USBU, KIU, PIU, AIU, PS2CH1, and PS2CH2 units.

Cautions 1. Set the clock supplied to USBU, CARDU1, CARDU2, and AC97U units during the soft reset period by the CMUSRST register (see 6.2.2).

2. Supply clock to the CLK48M pin after setting the MSK48MOSC bit to 1 (oscillator oscillation) and setting the MSK48MPIN bit to 1 (clock supply).

6.2.2 CMUSRST (base address + 0x042)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	AC97RST	CARD2RST[Note]	CARD1RST[Note]	USBRST
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	AC97RST	Soft reset to AC97U unit 1: Soft reset 0: Soft reset released
2	CARD2RST[Note]	Soft reset to CARDU2 unit 1: Soft reset 0: Soft reset released
1	CARD1RST[Note]	Soft reset to CARDU1 unit 1: Soft reset 0: Soft reset released
0	USBRST	Soft reset to USBU unit 1: Soft reset 0: Soft reset released

★ **Note** When either CARDU unit is reset via software, the power of the other CARDU unit will be initialized. Therefore, if a soft reset has been performed, set the registers of both units again.

CHAPTER 7 ICU (INTERRUPT CONTROL UNIT)

7.1 General

The ICU collects interrupt requests from the various on-chip peripheral units and transfers these interrupt request signals to the CPU.

The functions of the ICU's internal blocks are briefly described below.

- ADDECICU … Decodes read/write addresses from the CPU that are used for ICU registers.
- REGICU … This includes a register for interrupt masking. The initial value is "0", which specifies masking. No interrupt request signal is supplied to CPU unless the CPU writes "1" to this register.
- OUTICU … This block collects interrupt requests after masking them, and generates an interrupt request signal to output to the CPU.
  During Suspend mode, it also controls the masking of interrupt requests and output of the general interrupt source signal.

For details of the interrupt sources, see **7.2 Register Set**.

★ How an interrupt request is notified to the CPU core is shown below.

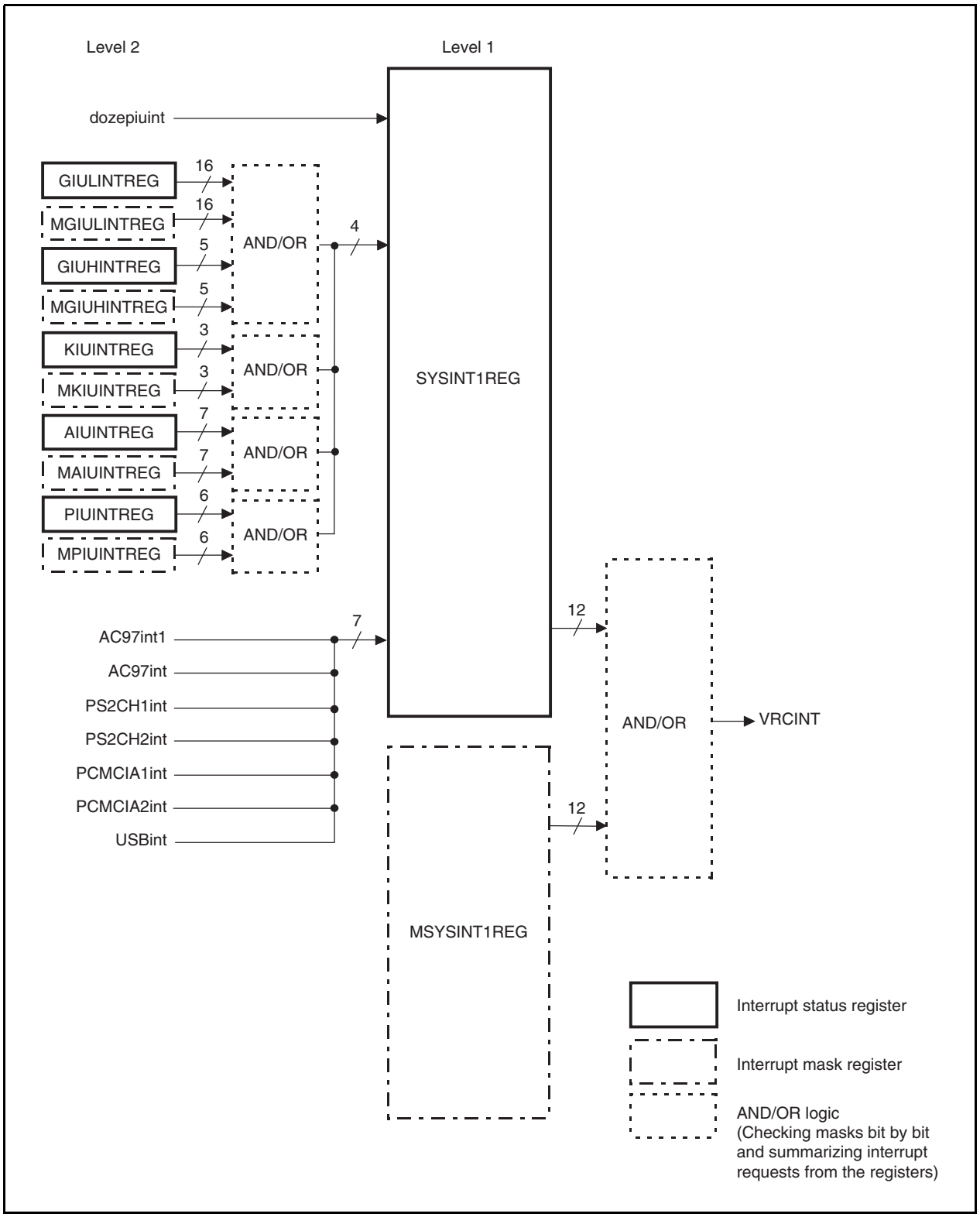
If an interrupt request occurs in the peripheral units, the corresponding bit in the interrupt status register of Level 2 (xxxINTREG register) is set to 1. The interrupt status register is ANDed bit-wise with the corresponding interrupt mask register of Level 2 (MxxxINTREG register). If the occurred interrupt request is enabled (set to 1) in the mask register, the interrupt request is notified to the interrupt status register of Level 1 (SYSINT1REG register) and the corresponding bit is set to 1. At this time, the interrupt requests from the same register of Level 2 are notified to the SYSINT1REG register as a single interrupt request.

Interrupt requests from some units directly set their corresponding bits in the SYSINT1REG register.

The SYSINT1REG register is ANDed bit-wise with the interrupt mask register of Level 1 (MSYSINT1REG register). If the interrupt request is enabled by MSYSINT1REG register (set to 1), a corresponding interrupt request signal is output from the ICU to the CPU.

Figure 7-1 shows an outline of interrupt control in the ICU.

★ **Figure 7-1. Interrupt Control Outline Diagram**



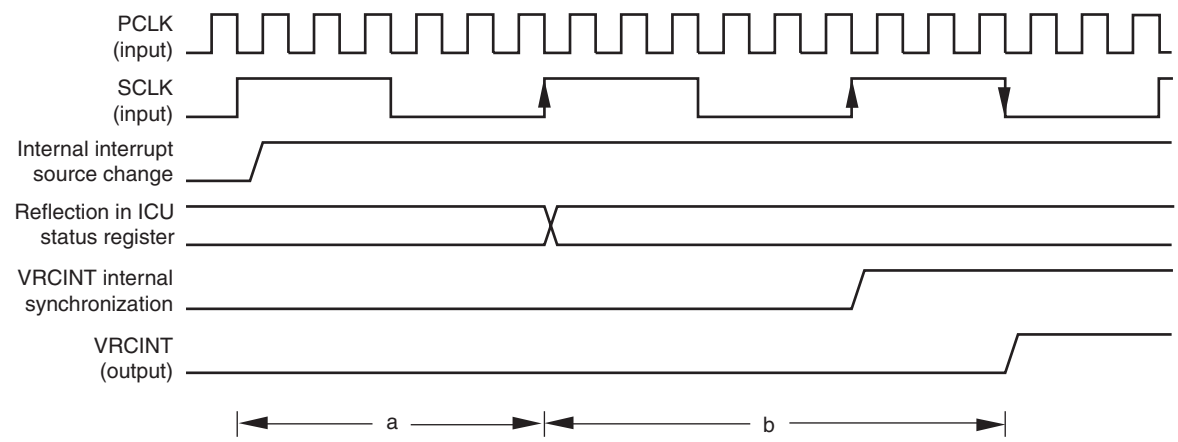
The VRCINT signal (interrupt request signal) output timing and the timing of the status change of each interrupt status register are described below. In the ICU, the sampling clock differs depending on the interrupt source. The assignment of sampling clock and interrupt sources are shown in the following table.

Table 7-1. Assignment of Sampling Clocks and Interrupt Sources

Sampling Clock	Interrupt Source
SCLK	DOZEPIUINTR, AC97INTR1, PS2CH1INTR, PS2CH2INTR, INTS(20:0) (GIUINTR), SCANINT (KIUINTR)
PCLK	Other than above

VRCINT signal is output in synchronization with the rising or falling edge of SCLK for all interrupt requests.
Due to differences in the sampling clocks and the synchronization of VRCINT output, there is a time lag between when the internal interrupt source changes and when this change is reflected in the interrupt status register and the VRCINT signal. These relationships are shown in Figures 7-2 and 7-3 below.

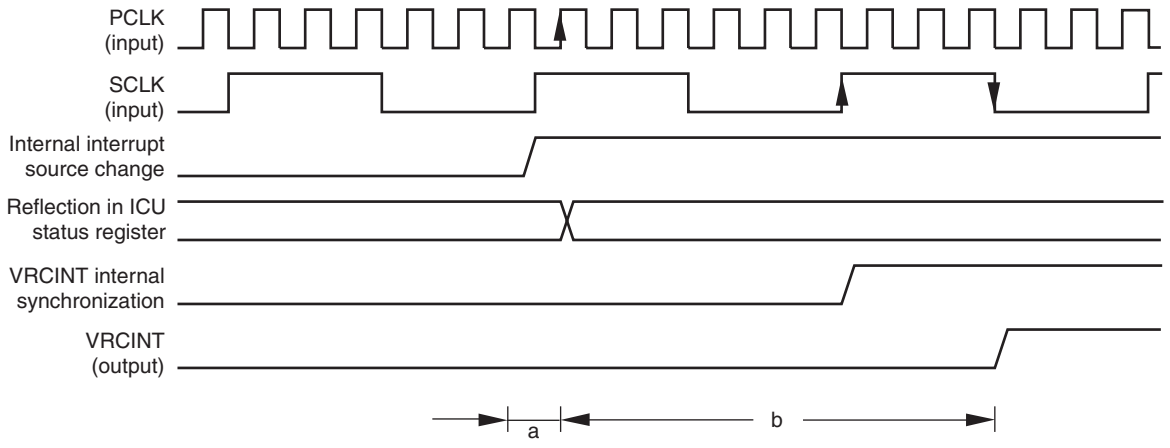
Figure 7-2. Time Lag Until Status Change Is Reflected in VRCINT Signal (When Sampling with SCLK)



Remark a: The time between when the internal interrupt source changes and when this change is reflected in the ICU status register (MAX. 1 × SCLK)

b: Time between when the change is reflected in the ICU status register and when it is output to VRCINT (1.5 × SCLK)

Figure 7-3. Time Lag Until Status Change Is Reflected in VRCINT Signal (When Sampling with PCLK)



Remark a: The time between when the internal interrupt source changes and when this change is reflected in the ICU status register (MAX. 1 × PCLK)

b: Time between when the change is reflected in the ICU status register and when it is output to VRCINT (MAX. 1.5 × SCLK)

7.2 Register Set

Table 7-2 lists the ICU registers.

Table 7-2. ICU Registers

Address	R/W	Register Symbol	Function
BASE + 0x060	R	SYSINT1REG	System interrupt register 1 (Level 1)
BASE + 0x062	R	PIUINTREG	PIU interrupt register (Level 2)
BASE + 0x064	R	AIUINTREG	AIU interrupt register (Level 2)
BASE + 0x066	R	KIUINTREG	KIU interrupt register (Level 2)
BASE + 0x068	R	GIULINTREG	GIUL interrupt register (Level 2)
BASE + 0x06A	R	GIUHINTREG	GIUH interrupt register (Level 2)
BASE + 0x06C	R/W	MSYSINT1REG	Mask system interrupt register 1 (Level 1)
BASE + 0x06E	R/W	MPIUINTREG	Mask PIU interrupt register (Level 2)
BASE + 0x070	R/W	MAIUINTREG	Mask AIU interrupt register (Level 2)
BASE + 0x072	R/W	MKIUINTREG	Mask KIU interrupt register (Level 2)
BASE + 0x074	R/W	MGIULINTREG	Mask GIUL interrupt register (Level 2)
BASE + 0x076	R/W	MGIUHINTREG	Mask GIUH interrupt register (Level 2)

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

7.2.1 SYSINT1REG (base address + 0x060)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	DOZE PIUINTR	RFU	RFU	AC97INTR1	AC97INTR	GIUINTR
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	KIUINTR	AIUINTR	PIUINTR	PS2CH1 INTR	PS2CH2 INTR	PCMCIA1 INTR	PCMCIA2 INTR	USBINTR
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:14	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
13	DOZEPIUINTR	PIU interrupt request during Suspend mode 1: Occurred 0: Normal
12:11	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
10	AC97INTR1	AC97int1 interrupt request 1: Occurred 0: Normal
9	AC97INTR	AC97 interrupt request 1: Occurred 0: Normal
8	GIUINTR	GIU interrupt request 1: Occurred 0: Normal
7	KIUINTR	KIU interrupt request 1: Occurred 0: Normal
6	AIUINTR	AIU interrupt request 1: Occurred 0: Normal
5	PIUINTR	PIU interrupt request 1: Occurred 0: Normal
4	PS2CH1INTR	PS2CH1 interrupt request 1: Occurred 0: Normal
3	PS2CH2INTR	PS2CH2 interrupt request 1: Occurred 0: Normal

(2/2)

Bit	Name	Function
2	PCMCIA1INTR	PCMCIA1 interrupt request 1: Occurred 0: Normal
1	PCMCIA2INTR	PCMCIA2 interrupt request 1: Occurred 0: Normal
0	USBINTR	USB interrupt request 1: Occurred 0: Normal

This register indicates when various interrupt requests occur in the VRC4173 system.

7.2.2 PIUINTREG (base address + 0x062)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	PADCMD INTR	PADADP INTR	PADPAGE1 INTR	PADPAGE0 INTR	PADDLO STINTR	RFU	PENCHG INTR
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:7	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
6	PADCMDINTR	PIU command scan interrupt request. This interrupt occurs when command scan found valid data. 1: Occurred 0: Normal
5	PADADPINTR	PIU A/D port scan interrupt request. This interrupt occurs when A/D port scan found a set of valid data. 1: Occurred 0: Normal
4	PADPAGE1INTR	PIU data buffer page 1 interrupt request. This interrupt occurs when a set of valid data is stored in page 1 of data buffer. 1: Occurred 0: Normal
3	PADPAGE0INTR	PIU data buffer page 0 interrupt request. This interrupt occurs when a set of valid data is stored in page 0 of data buffer. 1: Occurred 0: Normal
★ 2	PADDLOSTINTR	Data lost interrupt request. This interrupt occurs when a set of data did not found within specified time. 1: Occurred 0: Normal
1	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
0	PENCHGINTR	Touch panel contact status change interrupt request 1: Change has occurred 0: No change

This register indicates when various PIU-related interrupt requests occur.

7.2.3 AIUINTREG (base address + 0x064)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	INTMEND	INTM	INTMIDLE	INTMST
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	INTSEND	INTS	INTSIDLE	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11	INTMEND	Audio input (MIC) DMA buffer 2 page interrupt request 1: Occurred 0: Normal
10	INTM	Audio input (MIC) DMA buffer 1 page interrupt request 1: Occurred 0: Normal
9	INTMIDLE	Audio input (MIC) idle interrupt request (received data is lost). This interrupt occurs if valid data exists in MIDATREG register when data was received from A/D converter. 1: Occurred 0: Normal
8	INTMST	Audio input (MIC) receive completion interrupt request. This interrupt occurs when 12-bit converted data was received from the A/D converter. 1: Occurred 0: Normal
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	INTSEND	Audio output (speaker) DMA buffer 2 page interrupt request 1: Occurred 0: Normal
2	INTS	Audio output (speaker) DMA buffer 1 page interrupt request 1: Occurred 0: Normal
1	INTSIDLE	Audio output (speaker) idle interrupt request (mute). This interrupt occurs if there is no valid data in SODATREG register when data was transferred to D/A converter. 1: Occurred 0: Normal
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

This register indicates when various AIU-related interrupt requests occur.

7.2.4 KIUINTREG (base address + 0x066)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	KDATLOST	KDATRDY	SCANINT
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2	KDATLOST	Key scan data lost interrupt request 1: Occurred 0: Normal
1	KDATRDY	Key data scan complete interrupt request 1: Occurred 0: Normal
0	SCANINT	Key input detect interrupt request 1: Occurred 0: Normal

This register indicates when various KIU-related interrupt requests occur.

7.2.5 GIULINTREG (base address + 0x068)

Bit	15	14	13	12	11	10	9	8
Name	INTS15	INTS14	INTS13	INTS12	INTS11	INTS10	INTS9	INTS8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTS7	INTS6	INTS5	INTS4	INTS3	INTS2	INTS1	INTS0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTS(15:0)	Interrupt request input to GPIO(15:0) pin 1: Occurred 0: Normal

This register indicates when various GIU-related interrupt requests occur.

7.2.6 GIUHINTREG (base address + 0x06A)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTS20	INTS19	INTS18	INTS17	INTS16
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTS(20:16)	Interrupt request input to GPIO(20:16) pin 1: Occurred 0: Normal

This register indicates when various GIU-related interrupt requests occur.

7.2.7 MSYSINT1REG (base address + 0x06C)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	DOZE PIUINTR	RFU	RFU	AC97INTR1	AC97INTR	GIUINTR
R/W	R	R	R/W	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	KIUINTR	AIUINTR	PIUINTR	PS2CH1 INTR	PS2CH2 INTR	PCMCIA1 INTR	PCMCIA2 INTR	USBINTR
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:14	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
13	DOZEPIUINTR	PIU interrupt enable during Suspend mode 1: Enabled 0: Disabled
12:11	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
10	AC97INTR1	AC97int1 interrupt enable 1: Enabled 0: Disabled
9	AC97INTR	AC97 interrupt enable 1: Enabled 0: Disabled
8	GIUINTR	GIU interrupt enable 1: Enabled 0: Disabled
7	KIUINTR	KIU interrupt enable 1: Enabled 0: Disabled
6	AIUINTR	AIU interrupt enable 1: Enabled 0: Disabled
5	PIUINTR	PIU interrupt enable 1: Enabled 0: Disabled
4	PS2CH1INTR	PS2CH1 interrupt enable 1: Enabled 0: Disabled
3	PS2CH2INTR	PS2CH2 interrupt enable 1: Enabled 0: Disabled

(2/2)

Bit	Name	Function
2	PCMCIA1INTR	PCMCIA1 interrupt enable 1: Enabled 0: Disabled
1	PCMCIA2INTR	PCMCIA2 interrupt enable 1: Enabled 0: Disabled
0	USBINTR	USB interrupt enable 1: Enabled 0: Disabled

This register is used to mask various interrupt requests that occur in the VRC4173 system.

7.2.8 MPIUINTREG (base address + 0x06E)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	PADCMD INTR	PADADP INTR	PADPAGE1 INTR	PADPAGE0 INTR	PADDLO STINTR	RFU	PENCHG INTR
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:7	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
6	PADCMDINTR	PIU command scan interrupt enable 1: Enabled 0: Disabled
5	PADADPINTR	PIU A/D port scan interrupt enable 1: Enabled 0: Disabled
4	PADPAGE1INTR	PIU A/D data buffer page 1 interrupt enable 1: Enabled 0: Disabled
3	PADPAGE0INTR	PIU A/D data buffer page 0 interrupt enable 1: Enabled 0: Disabled
★ 2	PADDLOSTINTR	Data lost interrupt enable 1: Enabled 0: Disabled
1	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
0	PENCHGINTR	Touch panel contact status change interrupt enable 1: Enabled 0: Disabled

This register is used to mask various PIU-related interrupt requests.

7.2.9 MAIUINTREG (base address + 0x070)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	INTMEND	INTM	INTMIDLE	INTMST
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	INTSEND	INTS	INTSIDLE	RFU
R/W	R	R	R	R	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11	INTMEND	Audio input (MIC) DMA buffer 2 page interrupt enable 1: Enabled 0: Disabled
10	INTM	Audio input (MIC) DMA buffer 1 page interrupt enable 1: Enabled 0: Disabled
9	INTMIDLE	Audio input (MIC) idle interrupt (received data is lost) enable 1: Enabled 0: Disabled
8	INTMST	Audio input (MIC) receive complete interrupt enable 1: Enabled 0: Disabled
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	INTSEND	Audio output (speaker) DMA buffer 2 page interrupt enable 1: Enabled 0: Disabled
2	INTS	Audio output (speaker) DMA buffer 1 page interrupt enable 1: Enabled 0: Disabled
1	INTSIDLE	Audio output (speaker) idle interrupt (mute) enable 1: Enabled 0: Disabled
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

This register is used to mask various AIU-related interrupt requests.

7.2.10 MKIUINTREG (base address + 0x072)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	KDAT LOST	KDAT RDY	SCAN INT
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2	KDATLOST	Key scan data lost interrupt enable 1: Enabled 0: Disabled
1	KDATRDY	Key data scan complete interrupt enable 1: Enabled 0: Disabled
0	SCANINT	Key input detect interrupt enable 1: Enabled 0: Disabled

This register is used to mask various KIU-related interrupt requests.

7.2.11 MGIULINTREG (base address + 0x074)

Bit	15	14	13	12	11	10	9	8
Name	INTS15	INTS14	INTS13	INTS12	INTS11	INTS10	INTS9	INTS8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTS7	INTS6	INTS5	INTS4	INTS3	INTS2	INTS1	INTS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTS(15:0)	GPIO(15:0) pin interrupt input enable 1: Enabled 0: Disabled

This register is used to mask various GIU-related interrupt requests.

7.2.12 MGIUHINTREG (base address + 0x076)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTS20	INTS19	INTS18	INTS17	INTS16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTS(20:16)	GPIO(20:16) pin interrupt input enable 1: Enabled 0: Disabled

This register is used to mask various GIU-related interrupt requests.

7.3 Notes for Register Setting

There is no register setting flow in relation to the ICU.

With regard to the interrupt mask registers, the initial setting is “initial = 0 = mask” after setting. Therefore, enough masks must be cleared to provide sufficient interrupts for the CPU’s start-up processing.

CHAPTER 8 GIU (GENERAL-PURPOSE I/O UNIT)

8.1 General

The GIU controls the GPIO(20:0) pins. GPIO is a general-purpose port for which input and output are available. An interrupt request signal input function can be assigned to GPIO with input signal change (rising edge or falling edge of signal), low level, or high level used as the trigger.

Table 8-1 shows the clock to be used for interrupt request detection and the type of input buffer of the GPIO(20:0) pins.

Table 8-1. GPIO Pin Outline

Pin Name	Interrupt Request Detection Clock (Internal)	Input Buffer Type
GPIO(20:0)	SCLK	IO normal

When not used for an interrupt, the registers corresponding to these pins can be written to output a low-level or high-level signal. Each register can be read to check the state of the signal currently being input to the corresponding pin.

The GPIO pins can be used as transition factors from the Suspend or Standby mode to the Fullspeed mode.

8.2 Register Set

Table 8-2 lists the GIU registers.

Table 8-2. GIU Registers

Address	R/W	Register Symbol	Function
BASE + 0x080	R/W	GIUDIRL	GPIO I/O Select Register L
BASE + 0x082	R/W	GIUDIRH	GPIO I/O Select Register H
BASE + 0x084	R/W	GIUPIODL	GPIO Port I/O Data Register L
BASE + 0x086	R/W	GIUPIODH	GPIO Port I/O Data Register H
BASE + 0x088	R/W	GIUINTSTATL	GPIO Interrupt Status Register L
BASE + 0x08A	R/W	GIUINTSTATH	GPIO Interrupt Status Register H
BASE + 0x08C	R/W	GIUINTENL	GPIO Interrupt Enable Register L
BASE + 0x08E	R/W	GIUINTENH	GPIO Interrupt Enable Register H
BASE + 0x090	R/W	GIUINTTYPL	GPIO Interrupt Type (Edge or Level) Select Register L
BASE + 0x092	R/W	GIUINTTYPH	GPIO Interrupt Type (Edge or Level) Select Register H
BASE + 0x094	R/W	GIUINTALSELL	GPIO Interrupt Active Level Select Register L
BASE + 0x096	R/W	GIUINTALSELH	GPIO Interrupt Active Level Select Register H
BASE + 0x098	R/W	GIUINTHTSELL	GPIO Interrupt Hold/Through Select Register L
BASE + 0x09A	R/W	GIUINTHTSELH	GPIO Interrupt Hold/Through Select Register H
BASE + 0x09E	R/W	SELECTREG	Alternate Function Pin Select Register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

8.2.1 GIUDIRL (base address + 0x080)

Bit	15	14	13	12	11	10	9	8
Name	IOS15	IOS14	IOS13	IOS12	IOS11	IOS10	IOS9	IOS8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	IOS7	IOS6	IOS5	IOS4	IOS3	IOS2	IOS1	IOS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	IOS(15:0)	GPIO(15:0) pin I/O select 1: Output 0: Input

This register is used to set I/O modes for GPIO(15:0) pins. The IOS(15:0) bits correspond to the GPIO(15:0) pins.

When the IOS bit is set to 1, the corresponding GPIO pin is set for output and the value that has been written to the corresponding PIOD bit in the GIUPIODL register is output. When this bit is set to 0, the corresponding GPIO pin is set for input.

Caution The GPIO(15:0) pins are also used as KIU pins. When using as GPIO pins, setting is required in the SELECTREG register (see 8.2.15).

8.2.2 GIUDIRH (base address + 0x082)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	IOS20	IOS19	IOS18	IOS17	IOS16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	IOS(20:16)	GPIO(20:16) pin I/O select 1: Output 0: Input

This register is used to set I/O modes for GPIO(20:16) pins. The IOS(20:16) pins correspond to the GPIO(20:16) pins.

When the IOS bit is set to 1, the corresponding GPIO pin is set for output and the value that has been written to the corresponding PIOD bit in the GIUPIODH register is output. When this bit is set to 0, the corresponding GPIO pin is set for input.

Caution The GPIO(20:16) pins are also used as PIU pins. When using as GPIO pins, setting is required in the SELECTREG register (see 8.2.15).

8.2.3 GIUPIODL (base address + 0x084)

Bit	15	14	13	12	11	10	9	8
Name	PIOD15	PIOD14	PIOD13	PIOD12	PIOD11	PIOD10	PIOD9	PIOD8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PIOD7	PIOD6	PIOD5	PIOD4	PIOD3	PIOD2	PIOD1	PIOD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	PIOD(15:0)	GPIO(15:0) pin output data specification 1: High 0: Low

This register is used to read GPIO(15:0) pins and write data. The PIOD(15:0) bits correspond to the GPIO(15:0) pins.

When 1 is set to the corresponding IOS bit in the GIUDIRL register, the data written to the PIOD bit is output via the corresponding GPIO pin. When the value of the corresponding IOS bit in the GIUDIRL register is 0, writing a ★ value to the PIOD bit does not affect the GPIO pin (writing to the PIOD bit is performed normally).

When the value of the IOS bit in the GIUDIRL register is 0, reading the PIOD bit enables the corresponding GPIO pin's state to be read.

Caution The GPIO(15:0) pins are also used as KIU pins. When using as GPIO pins, setting is required in the SELECTREG register (see 8.2.15).

8.2.4 GIUPIODH (base address + 0x086)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	PIOD20	PIOD19	PIOD18	PIOD17	PIOD16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	PIOD(20:16)	GPIO(20:16) pin output data specification 1: High 0: Low

This register is used to read GPIO(20:16) pins and write data. The PIOD(20:16) bits correspond to the GPIO(20:16) pins.

When 1 is set to the corresponding IOS bit in the GIUDIRH register, the data written to the PIOD bit is output via the corresponding GPIO pin. When the value of the corresponding IOS bit in the GIUDIRH register is 0, writing a ★ value to the PIOD bit does not affect the GPIO pin (writing to the PIOD bit is performed normally).

When the value of the IOS bit in the GIUDIRH register is 0, reading the PIOD bit enables the corresponding GPIO pin's state to be read.

Caution The GPIO(20:16) pins are also used as PIU pins. When using as GPIO pins, setting is required in the SELECTREG register (see 8.2.15).

8.2.5 GIUINTSTATL (base address + 0x088)

Bit	15	14	13	12	11	10	9	8
Name	INTS15	INTS14	INTS13	INTS12	INTS11	INTS10	INTS9	INTS8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	INTS7	INTS6	INTS5	INTS4	INTS3	INTS2	INTS1	INTS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15:0	INTS(15:0)	Interrupt to GPIO(15:0) pins. Cleared to 0 when 1 is written. 1: Interrupt occurred 0: No interrupt

This register indicates the interrupt status of GPIO(15:0) pins. The INTS(15:0) bits correspond to the GPIO(15:0) pins.

The corresponding INTS bit is set to 1 when the signal input to the GPIO pin meets the condition set via the GIUINTTYPL register or the GIUINTALSELL register. Even if the corresponding bit is set to 1, however, no interrupt occurs when the GIUINTENL register is set to 0 (disable interrupt).

When the GPIO pin is not selected in the SELECTREG register, this register indicates 1, but the register value is invalid (interrupt status is not indicated).

When the GPIO pin is not selected in the SELECTREG register, disable the interrupt with the GIUINTENL register.

When using this register, it should be cleared to 0 once after the GIUINTTYPL and GIUINTALSELL registers are set to enable interrupt.

8.2.6 GIUINTSTATH (base address + 0x08A)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTS20	INTS19	INTS18	INTS17	INTS16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	1	1	1	1	1

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTS(20:16)	Interrupt to GPIO(20:16) pins. Cleared to 0 when 1 is written. 1: Interrupt occurred 0: No interrupt

This register indicates the interrupt status of GPIO(20:16) pins. The INTS(20:16) bits correspond to the GPIO(20:16) pins.

The corresponding INTS bit is set to 1 when the signal input to the GPIO pin meets the condition set via the GIUINTTYPH register or GIUINTALSELH register. Even if the corresponding bit is set to 1, however, no interrupt occurs when the GIUINTENH register is set to 0 (disable interrupt).

When the GPIO pin is not selected in the SELECTREG register, this register indicates 1, but the register value is invalid (interrupt status is not indicated).

When the GPIO pin is not selected in the SELECTREG register, disable the interrupt with the GIUINTENH register.

When using this register, it should be cleared to 0 once after the GIUINTTYPH and GIUINTALSELH registers are set to enable interrupt.

8.2.7 GIUINTENL (base address + 0x08C)

Bit	15	14	13	12	11	10	9	8
Name	INTE15	INTE14	INTE13	INTE12	INTE11	INTE10	INTE9	INTE8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTE7	INTE6	INTE5	INTE4	INTE3	INTE2	INTE1	INTE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTE(15:0)	Interrupt enable to GPIO(15:0) pins 1: Interrupt enable 0: Interrupt disable

This register is used to set interrupt enable status for GPIO(15:0) pins. The INTE(15:0) bits correspond to the GPIO(15:0) pins.

When 1 is set to the corresponding INTE bit, interrupts are enabled for the corresponding GPIO pins.

8.2.8 GIUINTENH (base address + 0x08E)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTE20	INTE19	INTE18	INTE17	INTE16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTE(20:16)	Interrupt enable to GPIO(20:16) pins 1: Interrupt enable 0: Interrupt disable

This register is used to set interrupt enable status for GPIO(20:16) pins. The INTE(20:16) bits correspond to the GPIO(20:16) pins.

When 1 is set to the corresponding INTE bit, interrupts are enabled for the corresponding GPIO pins.

8.2.9 GIUINTTYPL (base address + 0x090)

Bit	15	14	13	12	11	10	9	8
Name	INTT15	INTT14	INTT13	INTT12	INTT11	INTT10	INTT9	INTT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTT7	INTT6	INTT5	INTT4	INTT3	INTT2	INTT1	INTT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTT(15:0)	Interrupt request detection trigger 1: Edge 0: Level

This register is used to set the trigger to detect an interrupt request for GPIO(15:0) pins. The INTT(15:0) bits correspond to the GPIO(15:0) pins.

When 1 is set to the corresponding INTT bit, the edge detection method is used for the interrupt request signal at the corresponding GPIO pin (an interrupt request is triggered when the signal state changes from low to high or from high to low).

The level detection method is used when 0 is set, in which case the level set to the corresponding INTL bit in the GIUINTALSELL register is detected.

8.2.10 GIUINTTYPH (base address + 0x092)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTT20	INTT19	INTT18	INTT17	INTT16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTT(20:16)	Interrupt request detection trigger 1: Edge 0: Level

This register is used to set the trigger to detect an interrupt request for GPIO(20:16) pins. The INTT(20:16) bits correspond to the GPIO(20:16) pins.

When 1 is set to the corresponding INTT bit, the edge detection method is used for the interrupt request signal at the corresponding GPIO pin (an interrupt request is triggered when the signal state changes from low to high or from high to low).

The level detection method is used when 0 is set, in which case the level set to the corresponding INTL bit in the GIUINTALSELH register is detected.

8.2.11 GIUINTALSELL (base address + 0x094)

Bit	15	14	13	12	11	10	9	8
Name	INTL15	INTL14	INTL13	INTL12	INTL11	INTL10	INTL9	INTL8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTL7	INTL6	INTL5	INTL4	INTL3	INTL2	INTL1	INTL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTL(15:0)	Interrupt request detection level 1: High active 0: Low active

This register is used to set the active level when using the level detection method for interrupts to GPIO(15:0) pins. The INTL(15:0) bits correspond to the GPIO(15:0) pins.

The contents of this register are not reflected when the edge detection method is selected via the GIUINTTYPL register.

When using this register, be sure to set the level detection method via the GIUINTTYPL register.

8.2.12 GIUINTALSELH (base address + 0x096)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTL20	INTL19	INTL18	INTL17	INTL16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTL(20:16)	Interrupt request detection level 1: High active 0: Low active

This register is used to set the active level when using the level detection method for interrupts to GPIO(20:16) pins. The INTL(20:16) bits correspond to the GPIO(20:16) pins.

The contents of this register are not reflected when the edge detection method is selected via the GIUINTTYPH register.

When using this register, be sure to set the level detection method via the GIUINTTYPH register.

8.2.13 GIUINTHTSELL (base address + 0x098)

Bit	15	14	13	12	11	10	9	8
Name	INTH15	INTH14	INTH13	INTH12	INTH11	INTH10	INTH9	INTH8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	INTH7	INTH6	INTH5	INTH4	INTH3	INTH2	INTH1	INTH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	INTH(15:0)	GPIO(15:0) pin interrupt signal hold/through 1: Hold 0: Through

This register is used to set whether or not interrupt signals to the GPIO(15:0) pins should be held. The INTH(15:0) bits correspond to the GPIO(15:0) pins.

When 1 is set to the corresponding INTH bit, any interrupt signal input to the corresponding GPIO pin is held. When 0 is set to this bit, any interrupt signal input to the corresponding GPIO pin is not held and is instead allowed to pass through. Any held interrupt signal is cleared when 1 is set to the corresponding INTS bit in the GIUINTSTATL register.

INTH bits are not affected by GIUINTENL register.

If 1 (hold) is set to the INTH bit while the INTE bit in the GIUINTENL register is set to 0 (disable interrupts), any change in the pin state is retained as change data. Therefore, an interrupt still occurs when the INTE bit is again set to 1 (enable interrupts).

8.2.14 GIUINTHTSELH (base address + 0x09A)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	INTH20	INTH19	INTH18	INTH17	INTH16
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:0	INTH(20:16)	GPIO(20:16) pin interrupt signal hold/through 1: Hold 0: Through

This register is used to set whether or not interrupt signals to the GPIO(20:16) pins should be held. The INTH(20:16) bits correspond to the GPIO(20:16) pins.

When 1 is set to the corresponding INTH bit, any interrupt signal input to the corresponding GPIO pin is held. When 0 is set to this bit, any interrupt signal input to the corresponding GPIO pin is not held and is instead allowed to pass through. Any held interrupt signal is cleared when 1 is set to the corresponding INTS bit in the GIUINTSTATH register.

INTH bits are not affected by GIUINTENH register.

If 1 (hold) is set to the INTH bit while the INTE bit in the GIUINTENH register is set to 0 (disable interrupts), any change in the pin state is retained as change data. Therefore, an interrupt still occurs when the INTE bit is again set to 1 (enable interrupts).

The relationship between settings of GPIO interrupts enable/disable and hold/through is shown in Table 8-3.

Table 8-3. Correspondences Between Interrupt Mask and Interrupt Hold

Interrupt Trigger	Setting of GIUINTHTSEL Register	Setting of GIUINTEN Register	Hold in GIU	Notation to ICU
Level	Hold	Masked	Held	Not noticed
		Not masked	Held	Noticed
		Masked → canceled	Held	Noticed
	Through	Masked	Through	Not noticed
		Not masked	Through	Noticed
		Masked → canceled	Through	Not noticed
Edge	Hold	Masked	Held	Not noticed
		Not masked	Held	Noticed
		Masked → canceled	Held	Noticed
	Through	Masked	Through	Not noticed
		Not masked	Prohibited	Prohibited
		Masked → canceled	Through	Not noticed

8.2.15 SELECTREG (base address + 0x09E)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	SEL3	SEL2	SEL1	SEL0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	SEL3	Function selection of TPEN/GPIO20, TPY(1:0)/GPIO(19:18), TPX(1:0)/GPIO(17:16) pins 1: Used as GPIO(20:16) pins 0: Used as TPEN, TPY(1:0), TPX(1:0) pins
2	SEL2	Function selection of KSCAN11/PS2DATA1, KSCAN10/PS2CLK1 pins 1: Used as PS2DATA1, PS2CLK1 pins 0: Used as KSCAN(11:10) pins
1	SEL1	Function selection of KSCAN9/PS2DATA2, KSCAN8/PS2CLK2 pins 1: Used as PS2DATA2, PS2CLK2 pins 0: Used as KSCAN(9:8) pins
0	SEL0	Function selection of KPORT(7:0)/GPIO(15:8), KSCAN(7:0)/GPIO(7:0) pins 1: Used as GPIO(15:0) pins 0: Used as KPORT(7:0), KSCAN(7:0) pins

This register is used to select the alternate function pins of the VRC4173.

GPIO(20:0) pins, KIU, PIU, and PS2U pins are used exclusively from other function pins. Therefore, when setting the GPIO(20:0) pins to be used by using this register, set the enable bit to prohibit in the corresponding unit.

The correspondences of the alternate function pins are listed in the table on the next page.

Table 8-4. Alternate Function Correspondence Table of VRC4173

VRC4173 Pin	GPIO	PIU	PS2CH1	PS2CH2	KIU
TPEN/GPIO20	GPIO20	TPEN	–	–	–
TPY1/GPIO19	GPIO19	TPY1	–	–	–
TPY0/GPIO18	GPIO18	TPY0	–	–	–
TPX1/GPIO17	GPIO17	TPX1	–	–	–
TPX0/GPIO16	GPIO16	TPX0	–	–	–
KPORT7/GPIO15	GPIO15	–	–	–	KPORT7
KPORT6/GPIO14	GPIO14	–	–	–	KPORT6
KPORT5/GPIO13	GPIO13	–	–	–	KPORT5
KPORT4/GPIO12	GPIO12	–	–	–	KPORT4
KPORT3/GPIO11	GPIO11	–	–	–	KPORT3
KPORT2/GPIO10	GPIO10	–	–	–	KPORT2
KPORT1/GPIO9	GPIO9	–	–	–	KPORT1
KPORT0/GPIO8	GPIO8	–	–	–	KPORT0
KSCAN11/PS2DATA1	–	–	PS2DATA1	–	KSCAN11
KSCAN10/PS2CLK1	–	–	PS2CLK1	–	KSCAN10
KSCAN9/PS2DATA2	–	–	–	PS2DATA2	KSCAN9
KSCAN8/PS2CLK2	–	–	–	PS2CLK2	KSCAN8
KSCAN7/GPIO7	GPIO7	–	–	–	KSCAN7
KSCAN6/GPIO6	GPIO6	–	–	–	KSCAN6
KSCAN5/GPIO5	GPIO5	–	–	–	KSCAN5
KSCAN4/GPIO4	GPIO4	–	–	–	KSCAN4
KSCAN3/GPIO3	GPIO3	–	–	–	KSCAN3
KSCAN2/GPIO2	GPIO2	–	–	–	KSCAN2
KSCAN1/GPIO1	GPIO1	–	–	–	KSCAN1
KSCAN0/GPIO0	GPIO0	–	–	–	KSCAN0

Caution When using PS/2 for 1 channel, the KIU supports 64/80 keys.
When using PS/2 for 2 channels, the KIU supports only 64 keys.

CHAPTER 9 PIU (TOUCH PANEL INTERFACE UNIT)

9.1 General

The PIU uses an on-chip A/D converter and detects the X and Y coordinates of pen contact locations on the touch panel and scans the general-purpose A/D input port. Since the touch panel control circuit and the A/D converter (conversion precision: 12 bits) are both on-chip, the touch panel is connected directly to the VRC4173.

The PIU's function, namely the detection of X and Y coordinates, is performed partly by hardware and partly by software.

Hardware tasks:
- Touch panel applied voltage control
- Reception of coordinate data

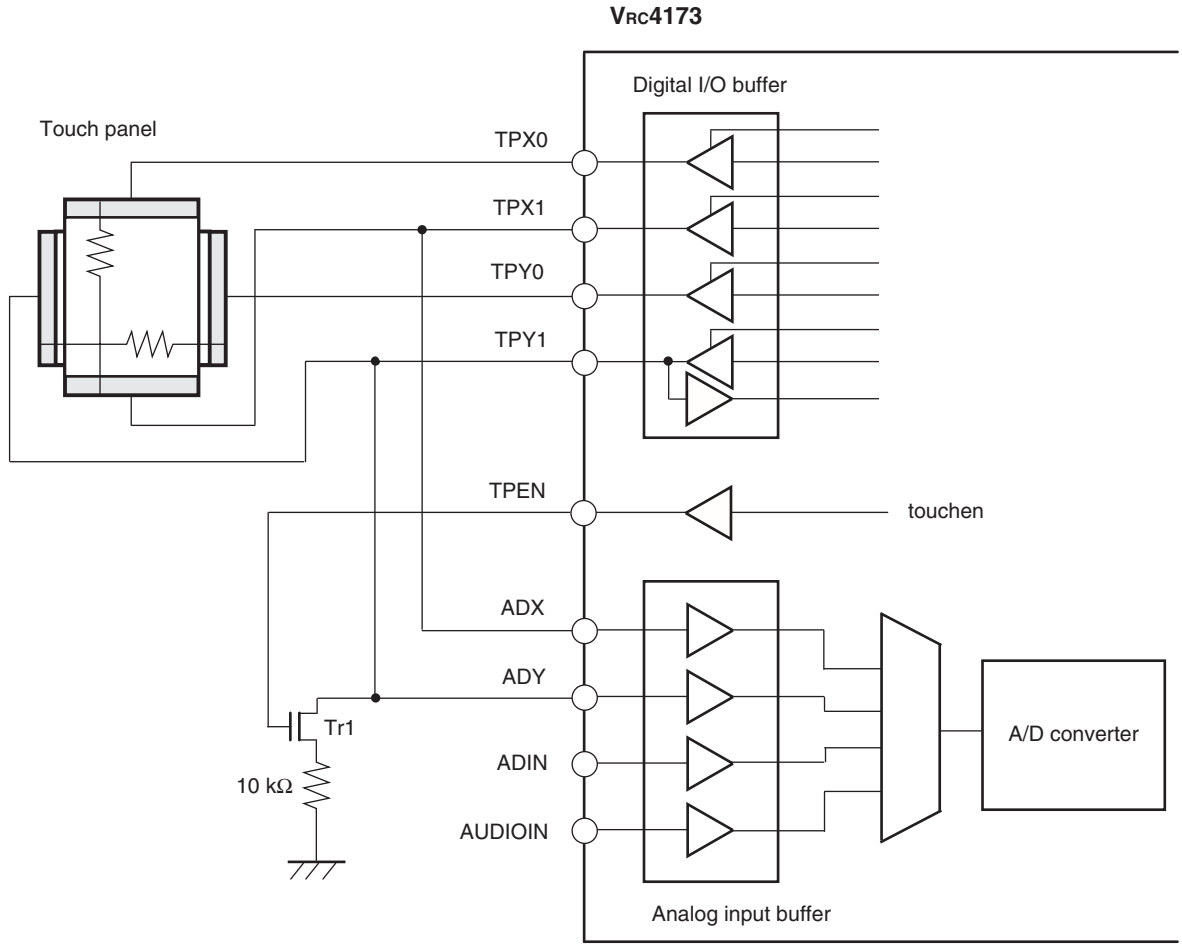
Software task:
- Processing of coordinate data based on data sampled by hardware

Features of the PIU's hardware tasks are described below.

- Can be directly connected to touch panel with four-pin resistance layers (on-chip touch panel driver)
- Interface for on-chip A/D converter
- Voltage detection at general-purpose A/D port and audio input port
- Operation of A/D converter based on various settings and control of voltage applied to touch panel
- Sampling of X-coordinate and Y-coordinate data
- Variable coordinate data sampling interval
- Interrupt request is triggered if pen touch occurs regardless of CPU operation mode (interrupt requests do not occur when in Hibernate mode)
- Four dedicated buffers for up to two pages each of coordinate data
- Two buffers for A/D port scan
- Auto/manual options for coordinate data sampling start/stop control

9.1.1 Block diagrams

Figure 9-1. PIU Peripheral Block Diagram



Remark When Tr1 is ON, pull down the TPY1 signal.
When Tr1 is OFF, leave the TPY1 signal open.

- **Touch panel**

  A set of four pins are located at the edges of the X-axis and Y-axis resistance layers, and the two layers have high resistance when there is no pen contact and low resistance when there is pen contact. The resistance between the two edges of the resistance layers is about 1 kΩ. When a voltage is applied to both edges of the Y-axis resistance layer, the voltage (V_{Y1} and V_{Y2} in Figure 9-2) is measured at the X-axis resistance layer's pins to determine the Y coordinate. Similarly, when a voltage is applied to both edges of the X-axis resistance layer, the voltage (V_{X1} and V_{X2} in Figure 9-2) is measured at the Y-axis resistance layer's pins to determine the X coordinate. For greater precision, voltage applied to individual resistance-layer pins can be measured to obtain X and Y coordinate data based on four voltage measurements. The obtained data is stored into the ★ PIUPBnmREG register (n = 0 or 1, m = 0 to 4).

Figure 9-2. Coordinate Detection Equivalent Circuits

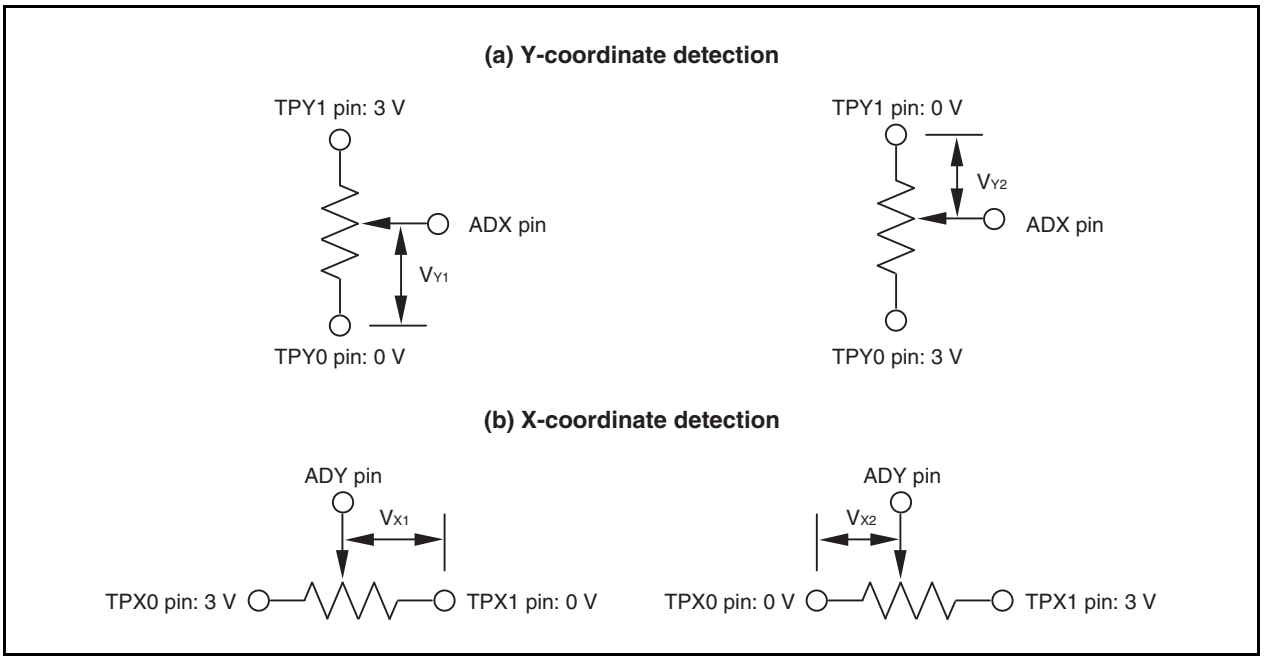
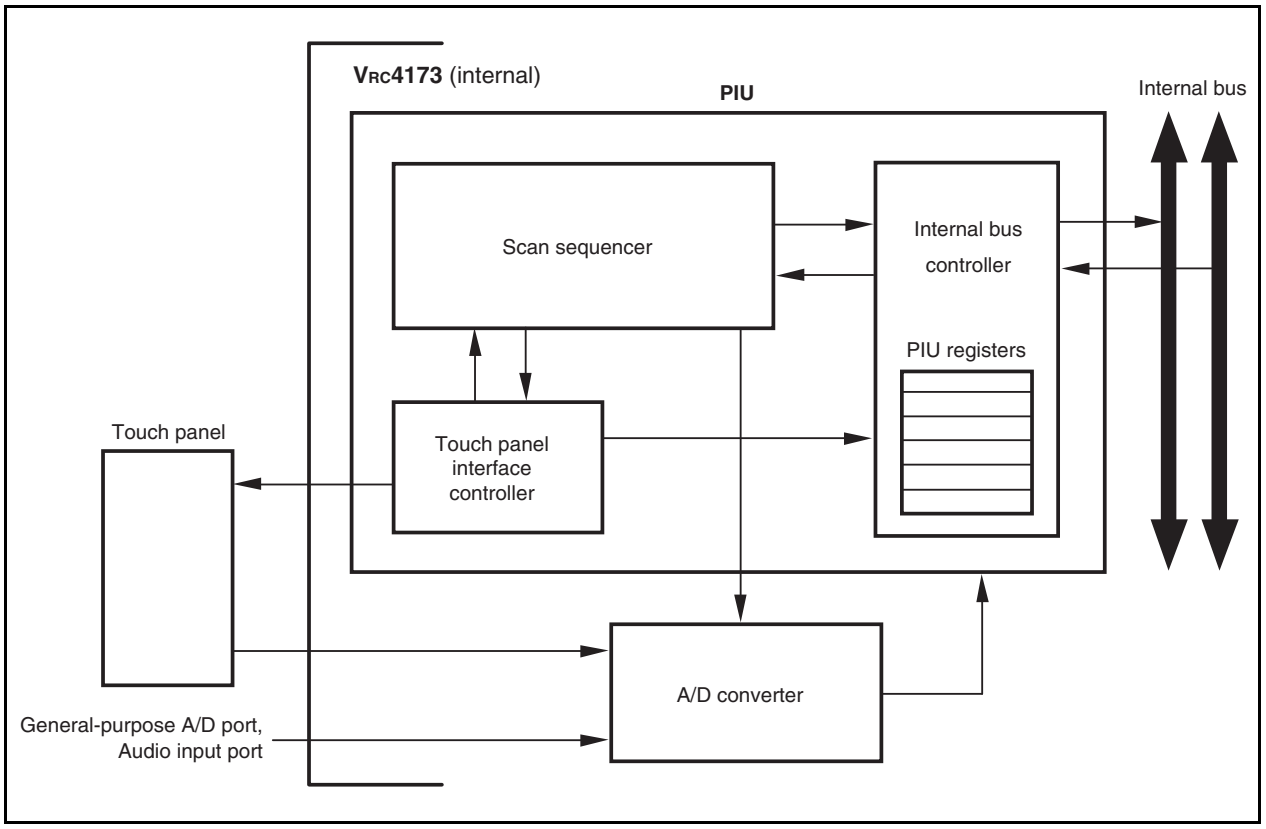


Figure 9-3. PIU Internal Block Diagram



The PIU includes three blocks: an internal bus controller, a scan sequencer, and a touch panel interface controller.

(1) Internal bus controller

The internal bus controller controls the internal bus, the PIU registers, and interrupts and performs serial/parallel conversion of data from the A/D converter.

(2) Scan sequencer

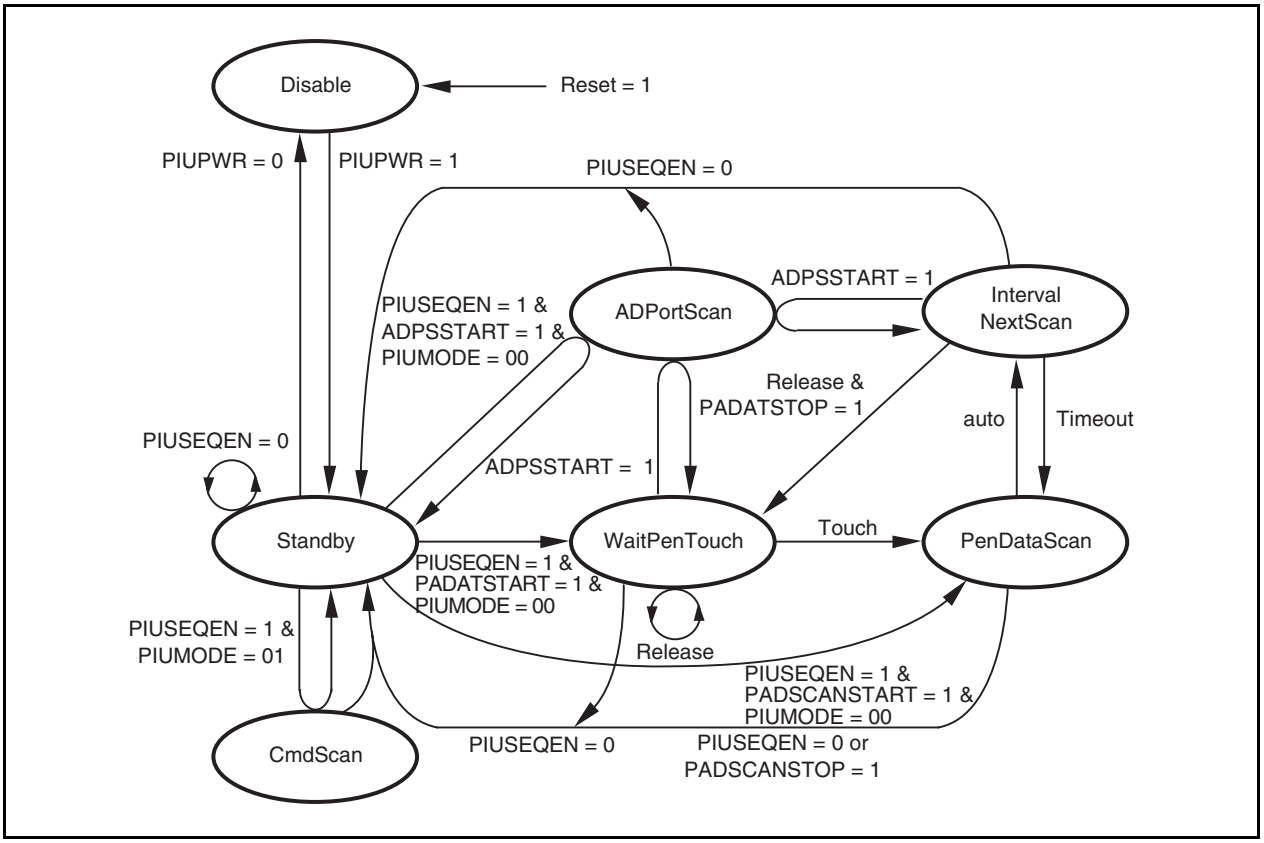
The scan sequencer is used for PIU state management.

(3) Touch panel interface controller

The touch panel interface controller is used to control the touch panel.

9.2 Scan Sequencer State Transition

★ **Figure 9-4. Scan Sequencer State Transition Diagram**



(1) Disable state

In this state, the A/D converter is in standby mode, the output pins are in touch detection mode and the input pins are in mask mode (to prevent misoperation when an undefined input is applied).

State transition to Suspend mode is possible, however, it is necessary to wait for the time set by STABLE(5:0) area in the PIUSTBLREG register to ensure stabilization.

(2) Standby state

In this state, the unit is in scan idle mode. The touch panel is in low-power mode (0 V voltage is applied to the touch panel and the A/D converter is in disable mode). Normally, this is the state from which various mode settings are made.

Caution State transitions occur when the PIUSEQEN bit is active, so the PIUSEQEN bit must be set as active after each mode setting has been completed.

(3) ADPortScan state

This is the state in which voltage is measured at the A/D converter's general-purpose port and audio input port. After the A/D converter is activated and voltage data is obtained, the data is stored in the PIU's internal ★ data buffer (PIUABnREG register). After the two ports are scanned, an A/D port scan interrupt occurs inside the PIU. After this interrupt occurs, the ADPSSTART bit is automatically set as inactive and the state changes to the state in which the ADPSSTART bit was active.

(4) CmdScan state

When in this state, the A/D converter operates using various settings. Voltage data from one port only is fetched based on a combination of the touch panel I/O signal setting (TPX(1:0), TPY(1:0)) and the selection of an input port (ADX, ADY, AUDIOIN, ADIN) connected to the A/D converter. Use PIUCMDREG register to make the touch panel pin setting and to select the input port.

(5) WaitPenTouch state

This is the standby state that waits for a touch panel “Touch” state. When the PIU detects a touch panel ★ “Touch” state, a touch panel contact status change interrupt occurs inside the PIU. At this point, if the PADATSTART bit is active, the state changes to the PenDataScan state.

In the WaitPenTouch state, it is possible to change to Suspend mode, however, the PCICLK stops and panel status detection is not performed.

(6) PenDataScan state

This is the state in which touch panel coordinates are detected. The A/D converter is activated and the four sets of data for each coordinate are sampled.

Caution If one complete pair of coordinates is not obtained during the interval between one pair of coordinates and the next coordinate data, a data lost interrupt occurs inside the PIU.

(7) IntervalNextScan state

This is the standby state that waits for the next coordinate sampling period and the touch panel’s “Release” state. After the touch panel state is detected, the time period specified via PIUSIVLREG register elapses before the transition to the PenDataScan state. If the PIU detects the “Release” state within the specified time period, a touch panel contact status change interrupt occurs inside the PIU. At this point, the state changes to the WaitPenTouch state if the PADATSTOP bit is active. If the PADATSTOP bit is inactive, it changes to the PenDataScan state after the specified time period has elapsed.

9.3 Register Set

Table 9-1 lists the PIU registers.

Table 9-1. PIU Registers

Address	R/W	Register Symbol	Function
BASE + 0x0A2	R/W	PIUCNTREG	PIU Control register
BASE + 0x0A4	R/W	PIUINTREG	PIU Interrupt cause register
BASE + 0x0A6	R/W	PIUSIVLREG	PIU Data sampling interval register
BASE + 0x0A8	R/W	PIUSTBLREG	PIU A/D converter start delay register
BASE + 0x0AA	R/W	PIUCMDREG	PIU A/D command register
BASE + 0x0B0	R/W	PIUASCNREG	PIU A/D port scan register
BASE + 0x0B2	R/W	PIUAMSKREG	PIU A/D scan mask register
BASE + 0x0BE	R	PIUCIVLREG	PIU Check interval register
BASE + 0x0C0	R/W	PIUPB00REG	PIU Page 0 Buffer 0 register
BASE + 0x0C2	R/W	PIUPB01REG	PIU Page 0 Buffer 1 register
BASE + 0x0C4	R/W	PIUPB02REG	PIU Page 0 Buffer 2 register
BASE + 0x0C6	R/W	PIUPB03REG	PIU Page 0 Buffer 3 register
BASE + 0x0C8	R/W	PIUPB10REG	PIU Page 1 Buffer 0 register
BASE + 0x0CA	R/W	PIUPB11REG	PIU Page 1 Buffer 1 register
BASE + 0x0CC	R/W	PIUPB12REG	PIU Page 1 Buffer 2 register
BASE + 0x0CE	R/W	PIUPB13REG	PIU Page 1 Buffer 3 register
BASE + 0x0D0	R/W	PIUAB0REG	PIU A/D scan Buffer 0 register
BASE + 0x0D2	R/W	PIUAB1REG	PIU A/D scan Buffer 1 register
BASE + 0x0DC	R/W	PIUPB04REG	PIU Page 0 Buffer 4 register
BASE + 0x0DE	R/W	PIUPB14REG	PIU Page 1 Buffer 4 register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

9.3.1 PIUCNTREG (base address + 0x0A2)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	PENSTC	PADSTATE2	PADSTATE1	PADSTATE0	PADAT STOP	PADAT START
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PADSCAN STOP	PADSCAN START	PADSCAN TYPE	PIUMODE1	PIUMODE0	PIUSEQEN	PIUPWR	PADRST
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:14	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
13	PENSTC	Touch/release when touch panel contact state changes 1: Touch 0: Release
12:10	PADSTATE(2:0)	Scan sequencer status 111: CmdScan 110: IntervalNextScan 101: PenDataScan 100: WaitPenTouch 011: Reserved 010: ADPortScan 001: Standby 000: Disable
9	PADATSTOP	Sequencer auto stop setting during touch panel release state 1: Auto stop after sampling data for one set of coordinates during release state 0: No auto stop (even during release state)
8	PADATSTART	Sequencer auto start setting during touch panel touch state 1: Auto start during touch state 0: No auto start during touch state
7	PADSCANSTOP	Forced stop setting for touch panel sequencer 1: Forced stop after sampling data for one set of coordinates 0: Do not stop
6	PADSCANSTART	Start setting for touch panel sequencer 1: Forced start 0: Do not start
5	PADSCANTYPE	Touch pressure sampling enable 1: Enable 0: Disable

(2/2)

Bit	Name	Function
4:3	PIUMODE(1:0)	PIU mode setting 11: Reserved 10: Reserved 01: Operate A/D converter using any command 00: Sample coordinate data
2	PIUSEQEN	Scan sequencer operation enable 1: Enable 0: Disable
1	PIUPWR	PIU power mode setting 1: Set PIU output as active and change to standby mode 0: Set panel to touch detection state and shift to PIU operation stop enabled mode
0	PADRST	PIU reset. Once the PADRST bit is set to 1, it is automatically cleared to 0 after four PCICLK cycles. 1: Reset 0: Normal

This register is used to make various settings for the PIU.

The PENSTC bit indicates the touch panel contact state at the time when the PENCHGINTR bit of PIUINTREG register is set to 1. This bit's state remains as it is until PENCHGINTR bit is cleared to 0. Also, when PENCHGINTR bit is cleared to 0, PENSTC bit indicates the touch panel contact state. However, PENSTC bit does not change while PENCHGINTR bit is set to 1, even if the touch panel contact state changes between release and touch.

Some bits in this register cannot be set in a specific state of scan sequencer. The combination of the setting of this register and the sequencer state is as follows.

Table 9-2. PIUCNTREG Register Bit Manipulation and States

PIUCNTREG Bit Manipulation		Scan Sequencer's State			
		Disable	Standby	WaitPenTouch	PenDataScan
PADRST[Note 1]	0 → 1	–	Disable	Disable	Disable
PIUPWR	0 → 1	Standby	?	×	×
	1 → 0	?	Disable	×	×
PIUSEQEN	0 → 1	×	**Note 2**	?	?
	1 → 0	?	?	Standby	Standby
PADATSTART	0 → 1	×	–	PenDataScan[Note 3]	×
	1 → 0	×	–	–	×
PADATSTOP	0 → 1	×	–	×	×
	1 → 0	×	–	×	×
PADSCANSTART	0 → 1	×	PenDataScan[Note 4]	×	×
	1 → 0	×	–	×	×
PADSCANSTOP	0 → 1	×	–	×	Standby[Note 5]
	1 → 0	×	–	×	–

PIUCNTREG Bit Manipulation		Scan Sequencer's State		
		IntervalNextScan	ADPortScan	CmdScan
PADRST[Note 1]	0 → 1	Disable	Disable	Disable
PIUPWR	0 → 1	?	?	?
	1 → 0	×	×	×
PIUSEQEN	0 → 1	?	?	?
	1 → 0	Standby	Standby	Standby
PADATSTART	0 → 1	×	×	×
	1 → 0	×	×	×
PADATSTOP	0 → 1	×	×	×
	1 → 0	×	×	×
PADSCANSTART	0 → 1	×	×	×
	1 → 0	×	×	×
PADSCANSTOP	0 → 1	Standby[Note 5]	Standby[Note 5]	Standby[Note 5]
	1 → 0	?	–	–

Notes 1. After 1 is written, the PADRST bit is automatically cleared to 0 after four PCICLK cycles.

★ **2.** The transition to WaitPenTouch state occurs when the PIUMODE(1:0) area is 00, and the transition to CmdScan state occurs when the PIUMODE(1:0) area is 01.

3. State transition occurs during touch state

4. State transition occurs when PIUSEQEN = 1

5. State transition occurs after one set of data is sampled. The PADSCANSTOP bit is cleared to 0 after the state transition occurs.

Remark –: The bit change is retained but there is no state transition.

×: Setting prohibited (operation not guaranteed)

?: Combination of state and bit status before setting does not exist.

9.3.2 PIUINTREG (base address + 0x0A4)

Bit	15	14	13	12	11	10	9	8
Name	OVP	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	PADCMD INTR	PADADP INTR	PADPAGE1 INTER	PADPAGE0 INTER	PADDLOST INTR	RFU	PENCHG INTR
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	OVP	Valid page ID bit (older valid page) 1: Valid data older than page 1 buffer data is retained 0: Valid data older than page 0 buffer data is retained
14:7	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
6	PADCMDINTR	PIU command scan interrupt. Cleared to 0 when 1 is written. 1: Indicates that command scan found valid data 0: Indicates that command scan did not find valid data in buffer
5	PADADPINTR	PIU A/D port scan interrupt. Cleared to 0 when 1 is written. 1: Indicates that A/D port scan found valid data with 1 value in buffer 0: Indicates that A/D port scan did not find valid data with 1 value in buffer
4	PADPAGE1INTER	PIU data buffer page 1 interrupt. Cleared to 0 when 1 is written. 1: Valid data with 1 value is stored in page 1 of data buffer 0: No valid data with 1 value in page 1 of data buffer
3	PADPAGE0INTER	PIU data buffer page 0 interrupt. Cleared to 0 when 1 is written. 1: Valid data with 1 value is stored in page 0 of data buffer 0: No valid data with 1 value in page 0 of data buffer
★ 2	PADDLOSTINTR	Data lost interrupt. Cleared to 0 when 1 is written. 1: Not data with 1 value found within specified time 0: No timeout
1	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
0	PENCHGINTR	Change in touch panel contact state interrupt. Cleared to 0 when 1 is written. 1: Change has occurred 0: No change

This register sets and indicates the interrupt request generation of PIU.

When the TPY1 signal changes, the PENCHGINTR bit is set to 1.

When the PENCHGINTR bit is set to1, the PENSTC bit indicates the touch panel contact state (touch or release) when a contact state changes. The PENSTC bit's state remains until PENCHGINTR bit is cleared to 0. Also, when PENCHGINTR bit is cleared to 0, PENSTC bit indicates the touch panel contact state. However, PENSTC bit does not change while PENCHGINTR bit is set to 1, even if the touch panel contact state changes between release and touch.

Caution In the Suspend mode, the VRC4173 retains the touch panel state. Therefore, if the Suspend mode has been entered while the touch panel is touched, the contact state may be mistakenly recognized as having changed, when the Fullspeed mode returns.
This may result in PENCHGINTR bit being set to 1, when a touch panel state change interrupt occurs immediately after the Fullspeed mode returns from the Suspend mode. Similarly, other bits of PIUINTREG register may be set to 1 on returning from the Suspend mode. Therefore, set each bit of PIUINTREG register to 1 to clear an interrupt request, immediately after the Fullspeed mode returns from the Suspend mode.

9.3.3 PIUSIVLREG (base address + 0x0A6)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	SCAN INTVAL10	SCAN INTVAL9	SCAN INTVAL8
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

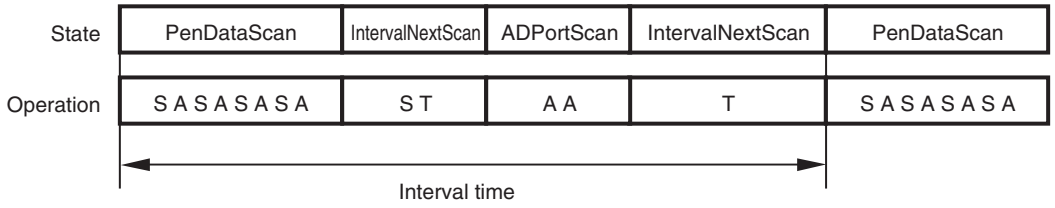
Bit	7	6	5	4	3	2	1	0
Name	SCAN INTVAL7	SCAN INTVAL6	SCAN INTVAL5	SCAN INTVAL4	SCAN INTVAL3	SCAN INTVAL2	SCAN INTVAL1	SCAN INTVAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	0	1	0	0	1	1	1

Bit	Name	Function
15:11	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
10:0	SCANINTVAL(10:0)	Coordinate data scan interval time setting Sampling interval = SCANINTVAL(10:0) × 30 μs

This register sets the interval time (sampling interval) for coordinate data scan.

The interval time for one pair of coordinate data is the value set via SCANINTVAL(10:0) multiplied by 30 μs. Accordingly, the logical range of interval times that can be set in 30 μs units is from 0 ms to about 60 ms. Actually, if the interval time setting is shorter than the time required for obtaining a pair of coordinate data or ADPScan data, a data lost interrupt will occur. If data lost interrupts occur frequently, set a longer interval time.

Figure 9-5. Interval Times and States



Remark S: Voltage stabilization standby time (STABLE(5:0) bits in PIUSTBLREG register)
A: A/D converter conversion time (about 10 μs)
T: Touch/release detection

9.3.4 PIUSTBLREG (base address + 0x0A8)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	STABLE5	STABLE4	STABLE3	STABLE2	STABLE1	STABLE0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	1	1

Bit	Name	Function
15:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5:0	STABLE(5:0)	Panel applied voltage stabilization standby time (PenDataScan, CmdScan state) A/D scan timeout time (ADPortScan state) Touch detection start standby time (Disable, WaitPenTouch, IntervalNextScan state) Standby time = STABLE(5:0) × 30 μs

The voltage stabilization standby time for the voltage applied to the touch panel can be set via STABLE(5:0) in 30 μs units between 0 μs and 1,890 μs.

9.3.5 PIUCMDREG (base address + 0x0AA)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	STABLEON	TPYEN1	TPYEN0	TPXEN1	TPXEN0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	TPYD1	TPYD0	TPXD1	TPXD0	ADCMD3	ADCMD2	ADCMD1	ADCMD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	1	1	1	1

Bit	Name	Function
15:13	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
12	STABLEON	Touch panel applied voltage stabilization time set during command scan (STABLE(5:0) area of PIUSTBLREG register) enable 1: Retain panel voltage stabilization time 0: Ignore panel voltage stabilization time (voltage stabilization standby time = 0)
11:10	TPYEN(1:0)	TPY port output enable switching during command scan 11: TPY1 output, TPY0 output 10: TPY1 output, TPY0 OFF (Hi-Z) 01: TPY1 OFF (Hi-Z), TPY0 output 00: TPY1 OFF (Hi-Z), TPY0 OFF (Hi-Z)
9:8	TPXEN(1:0)	TPX port output enable switching during command scan 11: TPX1 output, TPX0 output 10: TPX1 output, TPX0 OFF (Hi-Z) 01: TPX1 OFF (Hi-Z), TPX0 output 00: TPX1 OFF (Hi-Z), TPX0 OFF (Hi-Z)
7:6	TPYD(1:0)	TPY output level during command scan 11: TPY1 = High, TPY0 = High 10: TPY1 = High, TPY0 = Low 01: TPY1 = Low, TPY0 = High 00: TPY1 = Low, TPY0 = Low
5:4	TPXD(1:0)	TPX output level during command scan 11: TPX1 = High, TPX0 = High 10: TPX1 = High, TPX0 = Low 01: TPX1 = Low, TPX0 = High 00: TPX1 = Low, TPX0 = Low
3:0	ADCMD(3:0)	A/D converter input port selection for command scan 1111: A/D converter standby mode request 1110: Reserved : 0100: Reserved 0011: AUDIOIN port 0010: ADIN port 0001: ADY port 0000: ADX port

This register switches input/output and sets output level for each port during a command scanning operation.

Setting of the TPYD bit is invalid when the port output is set to OFF by the TPYEN bit.

Setting of the TPXD bit is invalid when the port output is set to OFF by the TPXEN bit.

9.3.6 PIUASCNREG (base address + 0x0B0)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	TPPSCAN	ADPS START
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1	TPPSCAN	Port selection for ADPortScan 1: Select ADX, ADY (for touch panel) as A/D port 0: Select ADIN (general-purpose) as A/D port and AUDIOIN as audio input port
0	ADPSSTART	ADPortScan start 1: Start ADPortScan 0: Do not perform ADPortScan

This register is used for ADPortScan setting.

The ADPortScan begins when the ADPSSTART bit is set. After the ADPortScan is completed, the state returns to the state when ADPortScan was started. The ADPSSTART bit is automatically cleared to 0.

If the ADPortScan is not completed within the time period set via PIUSTBLREG register's STABLE(5:0) area, a data lost interrupt occurs as a timeout interrupt.

Caution TPPSCAN bit operation is only valid during Standby state. The operation is not guaranteed during other states.

Some bits in this register cannot be set in a specific state of scan sequencer. The combination of the setting of this register and the sequencer state is as follows.

Table 9-3. PIUASCNREG Register Bit Manipulation and States

PIUASCNREG Bit Manipulation		Scan Sequencer's State			
		Disable	Standby	WaitPenTouch	PenDataScan
ADPSSTART	0 → 1	×	ADPortScan[Note]	×	×
	1 → 0	×	Disable	×	×
TPPSCAN	0 → 1	–	–	–	–
	1 → 0	–	–	–	–

PIUASCNREG Bit Manipulation		Scan Sequencer's State		
		IntervalNextScan	ADPortScan	CmdScan
ADPSSTART	0 → 1	×	ADPortScan[Note]	×
	1 → 0	×	Disable	×
TPPSCAN	0 → 1	×	WaitPenTouch	?
	1 → 0	?	?	Standby

Note After ADPortScan is completed, the bit is automatically cleared to 0.

Remark –: The bit change is retained but there is no state transition.
×: Setting prohibited (operation not guaranteed)
?: Combination of state and bit status before setting does not exist.

9.3.7 PIUAMSKREG (base address + 0x0B2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AUDINM	RFU	ADINM	RFU	ADYM	RFU	ADXM	RFU
R/W	R/W	R	R/W	R	R/W	R	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:8	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
7	AUDINM	Audio input port mask 1: Mask 0: Normal
6	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
5	ADINM	General-purpose A/D port mask 1: Mask 0: Normal
4	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
3	ADYM	Touch panel A/D port (ADY) mask 1: Mask 0: Normal
2	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
1	ADXM	Touch panel A/D port (ADX) mask 1: Mask 0: Normal
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

This register is used to set masking each A/D port. One bit corresponds to one port. When a port is masked (1), the analog data of that port is not converted into digital data.

The setting of this register is valid only in the ADPortScan state.

9.3.8 PIUCIVLREG (base address + 0x0BE)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	CHECK INTVAL10	CHECK INTVAL9	CHECK INTVAL8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CHECK INTVAL7	CHECK INTVAL6	CHECK INTVAL5	CHECK INTVAL4	CHECK INTVAL3	CHECK INTVAL2	CHECK INTVAL1	CHECK INTVAL0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	1	0	0	1	1	1

Bit	Name	Function
15:11	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
10:0	CHECKINTVAL(10:0)	Interval count value

This register is used for real-time reading of internal register values being counted down based on the PIUSIVLREG register setting.

9.3.9 PIUPBnmREG (base address + 0x0C0 to base address + 0x0CE, base address + 0x0DC to base address + 0x0DE)

Remark n = 0, 1, m = 0 to 4

PIUPB00REG	(base address + 0x0C0)	PIUPB10REG	(base address + 0x0C8)
PIUPB01REG	(base address + 0x0C2)	PIUPB11REG	(base address + 0x0CA)
PIUPB02REG	(base address + 0x0C4)	PIUPB12REG	(base address + 0x0CC)
PIUPB03REG	(base address + 0x0C6)	PIUPB13REG	(base address + 0x0CE)
PIUPB04REG	(base address + 0x0DC)	PIUPB14REG	(base address + 0x0DE)

Bit	15	14	13	12	11	10	9	8
Name	VALID	RFU	RFU	RFU	PADDATA11	PADDATA10	PADDATA9	PADDATA8
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PADDATA7	PADDATA6	PADDATA5	PADDATA4	PADDATA3	PADDATA2	PADDATA1	PADDATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	VALID	Indicates validity of data in page buffer 1: Valid 0: Invalid
14:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11:0	PADDATA(11:0)	A/D converter's sampling data

These registers are used to store coordinate data or touch pressure data. There are four coordinate data buffers and one touch pressure data buffer, each of which holds two pages of coordinate data or pressure data, and the addresses (register addresses) where the coordinate data or the pressure data is stored are fixed. Read coordinate data from the corresponding register in a valid page.

The VALID bit, which indicates when the data is valid, is automatically rendered invalid when the page buffer interrupt source (PADPAGE0INTR or PADPAGE1INTR bit in PIUINTREG register) is cleared to 0.

Table 9-4 shows correspondences between the sampled data and the register in which the sampled data is stored.

Table 9-4. Detected Data and Page Buffers

Detected Data	Page 0 Buffer	Page 1 Buffer
X–	PIUPB00REG	PIUPB10REG
X+	PIUPB01REG	PIUPB11REG
Y–	PIUPB02REG	PIUPB12REG
Y+	PIUPB03REG	PIUPB13REG
Z (Touch pressure)	PIUPB04REG	PIUPB14REG

9.3.10 PIUABnREG (base address + 0x0D0 to base address + 0x0D2)

Remark n = 0, 1
PIUAB0REG (base address + 0x0D0)
PIUAB1REG (base address + 0x0D2)

Bit	15	14	13	12	11	10	9	8
Name	VALID	RFU	RFU	RFU	PADDATA11	PADDATA10	PADDATA9	PADDATA8
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PADDATA7	PADDATA6	PADDATA5	PADDATA4	PADDATA3	PADDATA2	PADDATA1	PADDATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	VALID	Indicates validity of data in buffer 1: Valid 0: Invalid
14:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11:0	PADDATA(11:0)	A/D converter's sampling data

These registers are used to store general-purpose A/D port/audio input port sampling data or command scan data. There are two data buffers and the addresses (register address) where the data is stored are fixed.

The VALID bit, which indicates when the data is valid, is automatically rendered invalid when the page buffer interrupt source (PADADPINTR bit in PIUINTREG register) is cleared.

Table 9-5 shows correspondences between the sampled data and the register in which the sampled data is stored.

Table 9-5. A/D Ports and Data Buffers

Register	During ADPortScan		During CmdScan
	TPPSCAN = 0	TPPSCAN = 1	
PIUAB0REG	ADIN	ADX	CMDScanDATA
PIUAB1REG	AUDIOIN	ADY	–

9.4 Status Transfer Flow

Be sure to reset the PIU before operating the scan sequencer. Setting initial values via a reset sets particular values for the sequence interval, etc., that are required.

The following registers require initial settings.

SCANINTVAL(10:0) area in PIUSITVLREG register
STABLE(5:0) area in PIUSTBLREG register

Interrupt mask cancellation settings are required for registers other than the PIU registers.

Table 9-6. Mask Clear During Scan Sequencer Operation

Setting	Unit	Register	Bit	Value
Interrupt mask clear	ICU	MSYSINT1REG	PIUINTR	1
	ICU	MPIUINTREG	Bits 6:0	0x7F
Clock mask clear	CMU	CMUCLKMSK	MSKPIU	1

(1) Transfer flow for voltage detection at A/D general-purpose ports and audio input port

Standby, WaitPenTouch, or IntervalNextScan state
<1> PIUAMSKREG register Mask setting for A/D port and audio input port
<2> PIUASCNREG register ADPSSTART = 1
↓
ADPortScan state
<3> PIUASCNREG register ADPSSTART = 0
↓
Standby, WaitPenTouch, or IntervalNextScan state

(2) Transfer flow for auto scan coordinate detection

Standby state
<1> PIUCNTREG register PIUMODE(1:0) = 00
★ PADATSTART = 1
PADATSTOP = 1
<2> PIUCNTREG register PIUSEQEN = 1
↓
WaitPenTouch state

(3) Transfer flow for manual scan coordinate detection

Disable state
<1> PIUCNTREG register PIUPWR = 1
↓
Standby state
<2> PIUCNTREG register PIUMODE(1:0) = 00
PADSCANSTART = 1
<3> PIUCNTREG register PIUSEQEN = 1
↓
PenDataScan state

(4) Transfer flow during Suspend mode transition (WaitPenTouch state)

WaitPenTouch state
<1> Only waiting for the time set in PIUSTBLREG register's STABLE(5:0) area
<2> Execution of the SUSPEND instruction (Touch panel contact status change interrupt request does not occur)

(5) Transfer flow when returning from Suspend mode (WaitPenTouch state)

WaitPenTouch state (Register setting and stabilization wait are not needed.)
Touch detected
↓
PenDataScan state

(6) Transfer flow during Suspend mode transition (Disable state)

Standby, WaitPenTouch, or IntervalNextScan state
<1> PIUCNTREG register PIUSEQEN = 0
↓
Standby state
<2> PIUCNTREG register PIUPWR = 1
↓
Disable state
<3> Only waiting for the time set in PIUSTBLREG register's STABLE(5:0) area
<4> Execution of the SUSPEND instruction

(7) Transfer flow when returning from Suspend mode (Disable state)

Disable state
<1> PIUCNTREG register PIUPWR = 1
↓
Standby state
<2> PIUCNTREG register PIUMODE(1:0) = 00
★ PADATSTART = 1
PADATSTOP = 1
<3> PIUCNTREG register PIUSEQEN = 1
↓
WaitPenTouch state
Touch detected
↓
PenDataScan state

(8) Transfer flow of command scan

Disable state
<1> PIUCNTREG register PIUPWR = 1
↓
Standby state
<2> PIUCNTREG register PIUMODE(1:0) = 01
<3> PIUCNTREG register Touch panel pin setup and input port selection
<4> PIUCNTREG register PIUSEQEN = 1
↓
CmdScan state

9.5 Relationships Among TPX, TPY, ADX, ADY, TPEN, ADIN, and AUDIOIN Pins and States

Table 9-7. Relationships Among TPX, TPY, ADX, ADY, TPEN, ADIN, and AUDIOIN Pins and States

State	PADSTATE(2:0)	TPX1(ADX), TPX0	TPY1(ADY), TPY0	TPEN	AUDIOIN, ADIN
PIU disable (pen status detection)	Disable[Note]	HH	D–	H	––
Low-power standby	Standby	00	00	L	––
Pen status detection	WaitPenTouch/ IntervalNextScan	HH	D–	H	––
Voltage detection at general-purpose AD port	ADPortScan	00	00	L	–I
Voltage detection at audio input port	ADPortScan	00	00	L	I–
TPY1 = H, TPY0 = L, ADX = samp (X+)	PadDataScan	I–	HL	L	––
TPY1 = L, TPY0 = H, ADX = samp (X–)	PadDataScan	I–	LH	L	––
TPX1 = H, TPX0 = L, ADY = samp (Y+)	PadDataScan	HL	I–	L	––
TPX1 = L, TPX0 = H, ADY = samp (Y–)	PadDataScan	LH	I–	L	––
Touch pressure detection (Z)	PadDataScan	HH	d–	H	––

Note The states of pins are not guaranteed when the PADSTATE(2:0) area that precedes the CPU's SUSPEND instruction execution is in a state other then the Disable state.

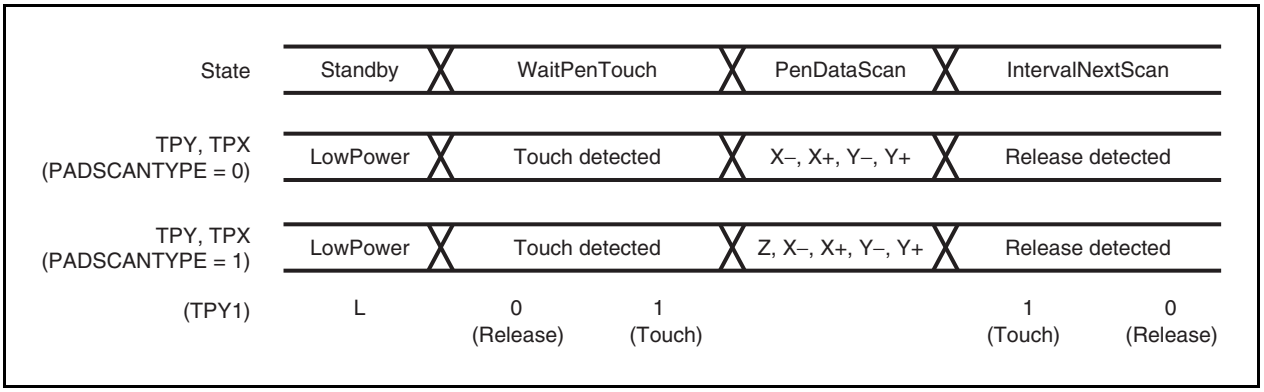
Remark 0: Low-level input
1: High-level input
L: Low-level output
H: High-level output
I: A/D converter input
D: Touch interrupt input (with a pull-down resistor)
d: No touch interrupt input (with a pull-down resistor)
–: don't care

9.6 Timing

9.6.1 Touch/release detection timing

Touch/release detection does not use the A/D converter but instead uses the voltage level of the TPY1 pin to determine the panel's touch/release state. The following figure shows a touch/release detection timing diagram.

Figure 9-6. Touch/Release Detection Timing

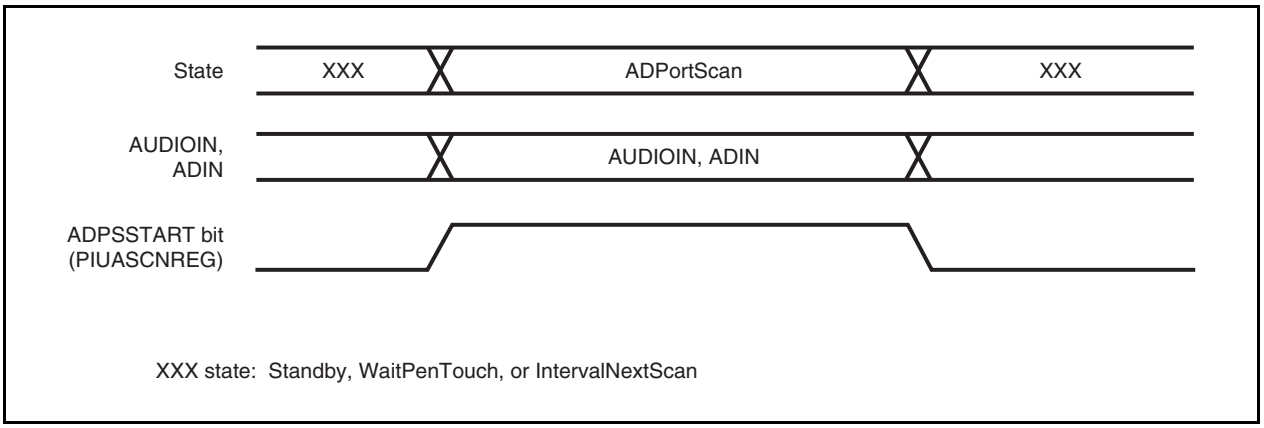


9.6.2 A/D port scan timing

The A/D port scan function sequentially scans the A/D converter's two input channel port pins and stores the data in the data buffer used for A/D port scanning.

The following figure shows an A/D port scan timing diagram.

Figure 9-7. A/D Port Scan Timing



9.7 Data Lost Generation Conditions

The PIU issues a data lost interrupt when any of the following four conditions exist.

1. Data for one coordinate has not been obtained within the interval period
2. The A/D port scan has not been completed within the time set via PIUSTBLREG register
3. Transfer of the next coordinate data has begun while valid data for both pages remains in the buffer
4. The next data transfer starts while there is valid data in the ADPortScan buffer

Once a data lost interrupt occurs, the sequencer is forcibly changed to the Standby state.
The causes and responses are shown below for the four conditions.

(1) When data for one coordinate has not been obtained within the interval period

(a) Cause

This condition occurs when the AIU has exclusive use of the A/D converter and the PIU is therefore unable to use the A/D converter.

If a data lost interrupt occurs frequently, implement a countermeasure that temporarily prohibits the AIU's use of the A/D converter.

(b) Response

After clearing the data lost interrupt by writing 1 to the PADDLOSTINTR bit of the PIUINTREG register, set the PIUCNTREG register's PADATSTART bit or PADSCANSTART bit to restart the coordinate detection operation. Once the data lost interrupt is cleared, the page in which the loss occurred becomes invalid. If the valid data prior to the data loss is needed, be sure to save the data that is being stored in the page buffer before clearing the data lost interrupt.

(2) When the A/D port scan has not been completed within the time set via PIUSTBLREG register

(a) Cause

Same as cause of condition (1)

(b) Response

After clearing the data lost interrupt by writing 1 to the PADDLOSTINTR bit of the PIUINTREG register, set the PIUASCNREG register's ADPSSTART bit to restart the A/D port scan operation. Once the data lost interrupt is cleared, the page in which the loss occurred becomes invalid. If the valid data prior to the data loss is needed, be sure to save the data that is being stored in the page buffer before clearing the data lost interrupt.

(3) When transfer of the next coordinate data has begun while valid data for both pages remains in the buffer

(a) Cause

This condition is caused when the data buffer contains two pages of valid data (both the data buffer page 1 and data buffer page 0 interrupts have occurred) but the valid data has not been processed. If the A/D converter is used frequently, this may shorten the time that would normally be required from when both pages become full until when the data loss occurs.

(b) Response

In condition (3), valid data contained in the pages when the interrupt occurs is never overwritten.

After two pages of valid data are processed, write 1 to the PADPAGE0INTR, PADPAGE1INTR, and PADDLOSTINTR bits of the PIUINTREG register to clear the three interrupts.

After clearing these interrupts, set the PADATSTART bit or PADSCANSTART bit of PIUCNTREG register to restart the coordinate detection operation.

(4) When the next data transfer starts while there is valid data in the ADPortScan buffer

(a) Cause

This condition is caused when valid data is not processed even while the ADPortScan buffer holds valid data (A/D port scan interrupt occurrence).

(b) Response

In condition (4), valid data contained in the buffer when the interrupt occurs is never overwritten.

After valid data in the buffer is processed, write 1 to the PADDLOSTINTR and PADADPINTR bits of the PIUINTREG register to clear the two interrupts.

After clearing these interrupts, set the ADPSSTART bit of PIUASCNREG register to restart the general-purpose A/D port scan.

CHAPTER 10 AIU (AUDIO INTERFACE UNIT)

10.1 General

The AIU supports speaker output and MIC input. The settings related to A/D converter and D/A converter are also performed by AIU. The resolution of the D/A converter used for a speaker is 10 bits, and the resolution of the A/D converter used for a microphone is 12 bits.

★ **Caution As the A/D converter and D/A converter are exclusively controlled, recording and playback by AIU cannot be performed simultaneously.**

10.2 Register Set

Table 10-1 lists the AIU registers.

Table 10-1. AIU Registers

Address	R/W	Register Symbol	Function
BASE + 0x0E0	R/W	MDMADATREG	MIC DMA Data Register
BASE + 0x0E2	R/W	SDMADATREG	Speaker DMA Data Register
BASE + 0x0E6	R/W	SODATREG	Speaker Output Data Register
BASE + 0x0E8	R/W	SCNTREG	Speaker Output Control Register
BASE + 0x0EA	R/W	SCNVRREG	Speaker Conversion Rate Register
BASE + 0x0F0	R/W	MIDATREG	MIC Input Data Register
BASE + 0x0F2	R/W	MCNTREG	MIC Input Control Register
BASE + 0x0F4	R/W	MCNVRREG	MIC Conversion Rate Register
BASE + 0x0F8	R/W	DVALIDREG	Data Valid Register
BASE + 0x0FA	R/W	SEQREG	Sequential Register
BASE + 0x0FC	R/W	INTREG	Interrupt Register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

10.2.1 MDMADATREG (base address + 0x0E0)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	MDMA11	MDMA10	MDMA9	MDMA8
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MDMA7	MDMA6	MDMA5	MDMA4	MDMA3	MDMA2	MDMA1	MDMA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11:0	MDMA(11:0)	MIC input DMA data

This register is used prior to DMA transfer to store 12-bit data that has been converted by the A/D converter and stored in MIDATREG register.

Write is used for debugging and is enabled when AIUMEN bit of SEQREG register is set to 1.

This register is initialized (0x0800) by resetting AIUMEN bit of SEQREG register to 0. Therefore, if the AIUMEN bit is set to 0 during DMA transfer, invalid data may be transferred.

10.2.2 SDMADATREG (base address + 0x0E2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	SDMA9	SDMA8
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	SDMA7	SDMA6	SDMA5	SDMA4	SDMA3	SDMA2	SDMA1	SDMA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9:0	SDMA(9:0)	Speaker output DMA data

This register is used to store 10-bit DMA data for speaker output. When SODATREG register is empty, the data is transferred to SODATREG register.

Write is used for debugging and is enabled when AIUSEN bit of SEQREG register is set to 1.

This register is initialized (0x0200) by resetting AIUSEN bit of SEQREG register to 0.

10.2.3 SODATREG (base address + 0x0E6)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	SODAT9	SODAT8
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	SODAT7	SODAT6	SODAT5	SODAT4	SODAT3	SODAT2	SODAT1	SODAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9:0	SODAT(9:0)	Speaker output data

★ This register is used to store 10-bit data for speaker output. Data is received from SDMADATREG register and is sent to the D/A converter.

Write is used for debugging and is enabled when AIUSEN bit of SEQREG register is set to 1.

This register is initialized (0x0200) by resetting AIUSEN bit of SEQREG register to 0.

10.2.4 SCNTREG (base address + 0x0E8)

Bit	15	14	13	12	11	10	9	8
Name	DAENAIU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	SSTATE	RFU	SSTOPEN	RFU
R/W	R	R	R	R	R	R	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	DAENAIU	This is the speaker D/A ($DAAV_{REF}$ connection) enable bit. 1: ON 0: OFF
14:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	SSTATE	Indicates speaker operation state 1: In operation 0: Stopped
2	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
1	SSTOPEN	Speaker output DMA transfer page boundary interrupt stop 1: Stop DMA request at 1-page boundary 0: Stop DMA request at 2-page boundary
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

This register is used to control the AIU's speaker block.

The DAENAIU bit controls the connection of $DAAV_{DD}$ and $DAAV_{REF}$ input to ladder type resistors in the D/A converter. Setting this bit to 0 (OFF) allows low power consumption when not using the D/A converter. When using the D/A converter, this bit must be set to 1 following the sequence described in **10.3 Operation Sequence**.

The content of the SSTATE bit is valid only when the AIUSEN bit of SEQREG register is set to 1.

10.2.5 SCNVRREG (base address + 0x0EA)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	SCNVR2	SCNVR1	SCNVR0
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2:0	SCNVR(2:0)	D/A conversion rate 111: Reserved : 101: Reserved 100: 8 ksps 011: Reserved 010: 44.1 ksps 001: 22.05 ksps 000: 11.025 ksps

This register is used to select a conversion rate for the D/A converter.

10.2.6 MIDATREG (base address + 0x0F0)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	MIDAT11	MIDAT10	MIDAT9	MIDAT8
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	1	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MIDAT7	MIDAT6	MIDAT5	MIDAT4	MIDAT3	MIDAT2	MIDAT1	MIDAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11:0	MIDAT(11:0)	MIC input data

This register is used to store 12-bit MIC input data that has been converted by the A/D converter. Data is sent to MDMADATREG register and is received from the A/D converter.

Write is used for debugging and is enabled when AIUMEN bit of SEQREG register is set to 1.

This register is initialized (0x0800) by resetting AIUMEN bit of SEQREG register to 0.

10.2.7 MCNTREG (base address + 0x0F2)

Bit	15	14	13	12	11	10	9	8
Name	ADENAIU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MSTATE	RFU	MSTOPEN	ADREQAIU
R/W	R	R	R	R	R	R	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	ADENAIU	This is the MIC A/D ($ADAV_{REF}P$ connection) enable bit. 1: ON 0: OFF
14:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	MSTATE	Indicates MIC operation state 1: In operation 0: Stopped
2	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
1	MSTOPEN	MIC input DMA transfer page boundary interrupt stop 1: Stop DMA request at 1-page boundary 0: Stop DMA request at 2-page boundary
0	ADREQAIU	A/D use request bit 1: Request 0: Normal

This register is used to control the AIU's MIC block.

The ADENAIU bit controls the connection of $ADAV_{DD}$ and $ADAV_{REF}P$ input to ladder type resistors in the A/D converter. Setting this bit to 0 (OFF) allows low power consumption when not using the A/D converter. When using the A/D converter, this bit must be set to 1 following the sequence described in **10.3 Operation Sequence**.

★ The content of the MSTATE bit is valid only when the AIUMEN bit of SEQREG register is set to 1.

This unit has priority when a conflict occurs with the PIU in relation to A/D conversion requests.

10.2.8 MCNVRREG (base address + 0x0F4)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	MCNVR2	MCNVR1	MCNVR0
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2:0	MCNVR(2:0)	A/D conversion rate 111: Reserved : 101: Reserved 100: 8 ksps 011: Reserved 010: 44.1 ksps 001: 22.05 ksps 000: 11.025 ksps

This register is used to select a conversion rate for the A/D converter.

10.2.9 DVALIDREG (base address + 0x0F8)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	SODATV	SDMAV	MIDATV	MDMAV
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	SODATV	This indicates when valid data has been stored in SODATREG register. 1: Valid data exists 0: No valid data
2	SDMAV	This indicates when valid data has been stored in SDMADATREG register. 1: Valid data exists 0: No valid data
1	MIDATV	This indicates when valid data has been stored in MIDATREG register. 1: Valid data exists 0: No valid data
0	MDMAV	This indicates when valid data has been stored in MDMADATREG register. 1: Valid data exists 0: No valid data

This register indicates when valid data has been stored in SODATREG, SDMADATREG, MIDATREG, or MDMADATREG register.

If data has been written directly to SODATREG, SDMADATREG, MIDATREG, or MDMADATREG register via software, the bits in this register are not active, so write 1 via software.

Write is used for debugging and is enabled when AIUSEN or AIUMEN bit of SEQREG register is set to 1.

If AIUSEN bit = 0 or AIUMEN bit = 0 in SEQREG register, then SODATV bit = SDMAV bit = 0 or MIDATV bit = MDMAV bit = 0.

10.2.10 SEQREG (base address + 0x0FA)

Bit	15	14	13	12	11	10	9	8
Name	AIURST	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	AIUMEN[Note]	RFU	RFU	RFU	AIUSEN[Note]
R/W	R	R	R	R/W	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	AIURST	AIU reset via software 1: Reset 0: Normal
14:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4	AIUMEN[Note]	MIC block operation enable, DMA enable 1: Enable operation 0: Disable operation
3:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	AIUSEN[Note]	Speaker block operation enable, DMA enable 1: Enable operation 0: Disable operation

★ **Note** As the MIC block and speaker block cannot be operated simultaneously, do not set these bits to 1 at the same time. If both are set to 1, unintended data may be played back.

This register is used to enable/disable the AIU's operation.

10.2.11 INTREG (base address + 0x0FC)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	MENDINTR	MINTR	MIDLEINTR	MSTINTR
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	SENDINTR	SINTR	SIDLEINTR	RFU
R/W	R	R	R	R	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11	MENDINTR	MIC DMA 2 page interrupt. Cleared to 0 when 1 is written. 1: Occurred 0: Normal
10	MINTR	MIC DMA 1 page interrupt. Cleared to 0 when 1 is written. 1: Occurred 0: Normal
9	MIDLEINTR	MIC idle interrupt (receive data loss). Cleared to 0 when 1 is written. 1: Occurred 0: Normal
8	MSTINTR	MIC receive complete interrupt. Cleared to 0 when 1 is written. 1: Occurred 0: Normal
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	SENDINTR	SPEAKER DMA 2 page interrupt. Cleared to 0 when 1 is written. 1: Occurred 0: Normal
2	SINTR	SPEAKER DMA 1 page interrupt. Cleared to 0 when 1 is written. 1: Occurred 0: Normal
1	SIDLEINTR	SPEAKER idle interrupt (mute). Cleared to 0 when 1 is written. 1: Occurred 0: Normal
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

This register is used to set/indicate whether AIU interrupts have occurred or not.

When data is received from the A/D converter, MIDLEINTR bit is set if valid data still exists in MIDATREG register (MIDATV bit = 1). In this case, MIDATREG register is overwritten.

MSTINTR bit is set when data is received in MDMADATREG register.

When data is passed to the D/A converter, SIDLEINTR bit is set if there is no valid data in SODATREG register (SODATV bit = 0). However, this interrupt is valid only after AIUSEN bit = 1, after which SODATV bit = 1 in DVALIDREG register.

10.3 Operation Sequence

10.3.1 Output (Speaker)

(1) When using DMA transfer

<1> Set D/A conversion rate (SCNVR(2:0) area in SCNVRREG register = any value)

<2> Set output data area to DMAAU

<3> DMA enable in DCU

<4> Set D/A converter's $DAAV_{REF}$ to ON (DAENAIU bit of SCNTREG register = 1)

<5> Wait for $DAAV_{REF}$ resistor stabilization time (about 5 μs) (use the RTC counter)

Even if speaker power is set to ON and speaker operation is enabled (AIUSEN bit = 1) without waiting for $DAAV_{REF}$ resistor stabilization time, speaker output starts after the period calculated with the formula below.

5 + 1/conversion rate (44.1, 22.05, 11.025, or 8 ksps) (μs)

In this case, however, a noise may occur when speaker power is set to ON.

<6> Set speaker power to ON via GPIO

<7> Speaker operation enable (AIUSEN bit of SEQREG register = 1)

When the speaker operation is enabled, the following internal operations occur.

1. DMA request
2. Receive acknowledge and DMA data from DMA
   - DVALIDREG register's SDMAV bit = SODATV bit = 1
3. Output 10-bit data (SODAT(9:0) area in SODATREG register) to D/A converter
   - SODATV bit = 0, SDMAV bit = 1
   - Send SDMADATREG register data to SODATREG register.
   - SODATV bit = 1, SDMAV bit = 0
4. Output DMA request and store the data after the next into SDMADATREG register.
   - SODATV bit = 1, SDMAV bit = 1
5. Refresh data at each conversion timing interval
   - Becomes SIDLEINTR bit = 1 when DMA is slow and SODATV bit = 0 during conversion timing interval, and (mute) interrupt occurs
6. DMA page boundary interrupt occurs at page boundary
   - Clear the page interrupt request to continue output.

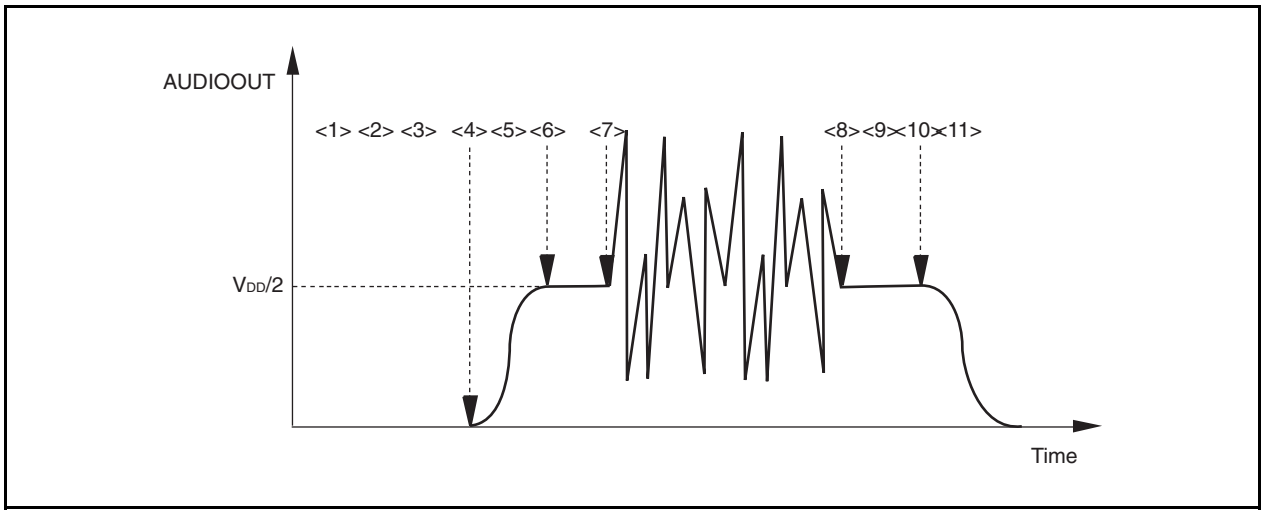
<8> Speaker operation disable (AIUSEN bit of SEQREG register = 0)

<9> Set speaker power to OFF via GPIO

<10> Set D/A converter's $DAAV_{REF}$ to OFF (DAENAIU bit of SCNTREG register = 0)

<11> DMA disable in DCU

Figure 10-1. Speaker Output and AUDIOOUT Pin



★ **(2) When not using DMA transfer**

<1> Enable clock supply to AIU in CMU
<2> Set D/A conversion rate (SCNVR(2:0) area in SCNVRREG register = any value)
<3> DMA disable in DCU (disable set as initial value)
<4> Set D/A converter's DAAVREF to ON (DAENAIU bit of SCNTREG register = 1)
<5> Wait for DAAVREF resistor stabilization time (about 5 μs)
<6> Set speaker power to ON via GPIO
<7> Speaker operation enable (AIUSEN bit of SEQREG register = 1)
Sampling counter begins to count up
<8> Set data to SODATREG register
<9> Speaker operation disable (AIUSEN bit of SEQREG register = 0)
<10> Set speaker power to OFF via GPIO
<11> Set D/A converter's DAAVREF to OFF (DAENAIU bit of SCNTREG register = 0)

Remark The interrupt request caused by mute is valid after setting AIUSEN bit = 1 and then SODATV bit = 1. However, SODATV bit does not become 1 until DVALIDREG register is written by the DMA or software. Therefore mute interrupt requests will not occur as long as DMA is disabled and SODATV bit is not converted by software.

10.3.2 Input (MIC)

<1> Set A/D conversion rate (MCNVR(2:0) area in MCNVRREG register = any value)

<2> Set input data area in DMAAU

<3> DMA enable in DCU

<4> Set A/D converter's $ADAV_{REF}P$ to ON (ADENAIU bit of MCNTREG register = 1)

★ MIC power can be set ON and MIC operation can be enabled (AIUMEN bit = 1) without waiting for $ADAV_{REF}P$ resistor stabilization time (about 5 μs). However, in such a case, sampling starts after the period calculated with the formula below.

5 + 1/conversion rate (44.1, 22.05, 11.025, or 8 ksps) (μs)

<5> Set MIC power to ON via GPIO.

<6> MIC operation enable (AIUMEN bit of SEQREG register = 1)

When the MIC operation is enabled, the following internal operations occur.

1. Output A/D conversion request to A/D converter
2. Return acknowledge and 12-bit conversion data from A/D converter.
3. Store data in MIDATREG register.
   - DVALIDREG register's MDMAV bit = 0, MIDATV bit = 1
4. Transfer data from MIDATREG register to MDMADATREG register.
   - MDMAV bit = 1, MIDATV bit = 0
   - The INTMST bit becomes 1 and an interrupt (receive complete) occurs.
5. Issue DMA request and store MIDMADATREG register data to memory.
   - MDMAV bit = 0, MIDATV bit = 0
6. An A/D request is issued once per conversion timing interval and 12-bit data is received
   - Becomes MIDLEINTR bit = 1 when DMA is slow and MIDATV bit = 1 during conversion timing interval, and (data loss) interrupt occurs
7. DMA page boundary interrupt occurs at page boundary
   - Clear the page interrupt request to continue output.

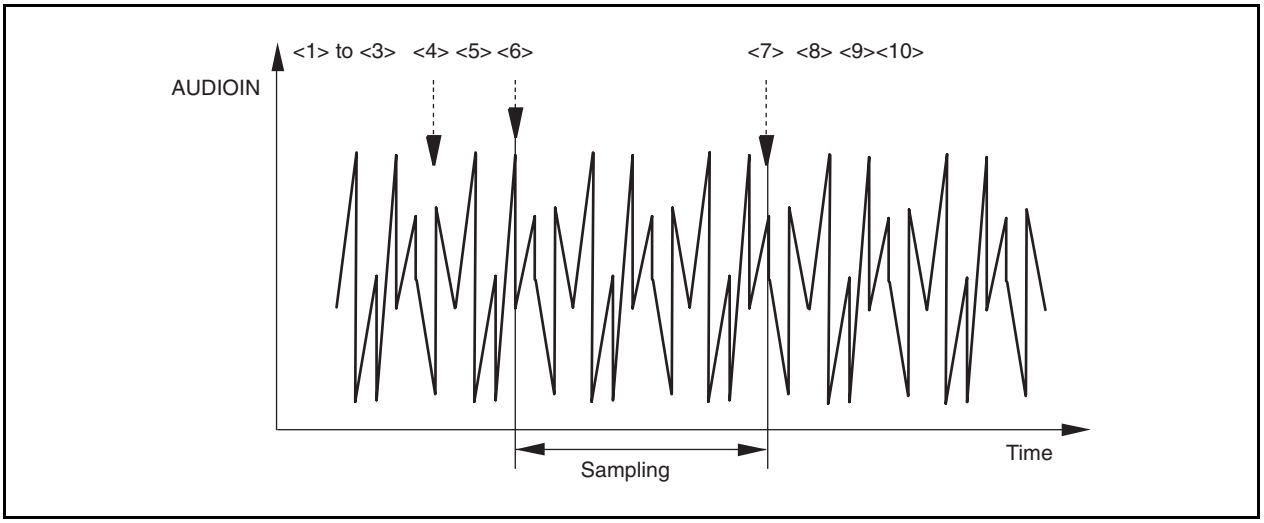
<7> MIC operation disable (AIUMEN bit of SEQREG register = 0)

<8> Set MIC power to OFF via GPIO.

<9> Set A/D converter's $ADAV_{REF}P$ to OFF (ADENAIU bit of MCNTREG register = 0)

<10> DMA disable in DCU

Figure 10-2. AUDIOIN Pin and MIC Operation



CHAPTER 11 KIU (KEYBOARD INTERFACE UNIT)

11.1 General

The KIU includes 12 scan lines and 8 detection lines. The number of key inputs to be detected can be selected from 96/80/64, by switching the number of scan lines from 12/10/8.

The register can be set to enable the 12 scan lines to be used as a general-purpose I/O port or PS/2 interface signals.

For details, see **CHAPTER 8 GIU (GENERAL-PURPOSE I/O UNIT)**.

11.2 Register Set

Table 11-1 lists the KIU registers.

Table 11-1. KIU Registers

Address	R/W	Register Symbol	Function
BASE + 0x100	R/W	KIUDAT0	KIU Data0 Register
BASE + 0x102	R/W	KIUDAT1	KIU Data1 Register
BASE + 0x104	R/W	KIUDAT2	KIU Data2 Register
BASE + 0x106	R/W	KIUDAT3	KIU Data3 Register
BASE + 0x108	R/W	KIUDAT4	KIU Data4 Register
BASE + 0x10A	R/W	KIUDAT5	KIU Data5 Register
BASE + 0x110	R/W	KIUSCANREP	KIU Scan/Repeat Register
BASE + 0x112	R	KIUSCANS	KIU Scan Status Register
BASE + 0x114	R/W	KIUWKS	KIU Wait Keyscan Stable Register
BASE + 0x116	R/W	KIUWKI	KIU Wait Keyscan Interval Register
BASE + 0x118	R/W	KIUINT	KIU Interrupt Register
BASE + 0x11A	W	KIURST	KIU Reset Register
BASE + 0x11E	R/W	SCANLINE	KIU Scan Line Register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

11.2.1 KIUDATn (base address + 0x100 to base address + 0x10A)

Remark n = 0 to 5
KIUDAT0 (base address + 0x100)
KIUDAT1 (base address + 0x102)
KIUDAT2 (base address + 0x104)
KIUDAT3 (base address + 0x106)
KIUDAT4 (base address + 0x108)
KIUDAT5 (base address + 0x10A)

Bit	15	14	13	12	11	10	9	8
Name	KEYDAT15	KEYDAT14	KEYDAT13	KEYDAT12	KEYDAT11	KEYDAT10	KEYDAT9	KEYDAT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	KEYDAT7	KEYDAT6	KEYDAT5	KEYDAT4	KEYDAT3	KEYDAT2	KEYDAT1	KEYDAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:8	KEYDAT(15:8)	Scan data from odd-numbered scans (scan by KSCAN1, 3, 5, 7, 9, or 11 pin)
7:0	KEYDAT(7:0)	Scan data from even-numbered scans (scan by KSCAN0, 2, 4, 6, 8, or 10 pin)

These registers are used to hold key scan data. Each KIU data register is able to hold the data from one scan operation.

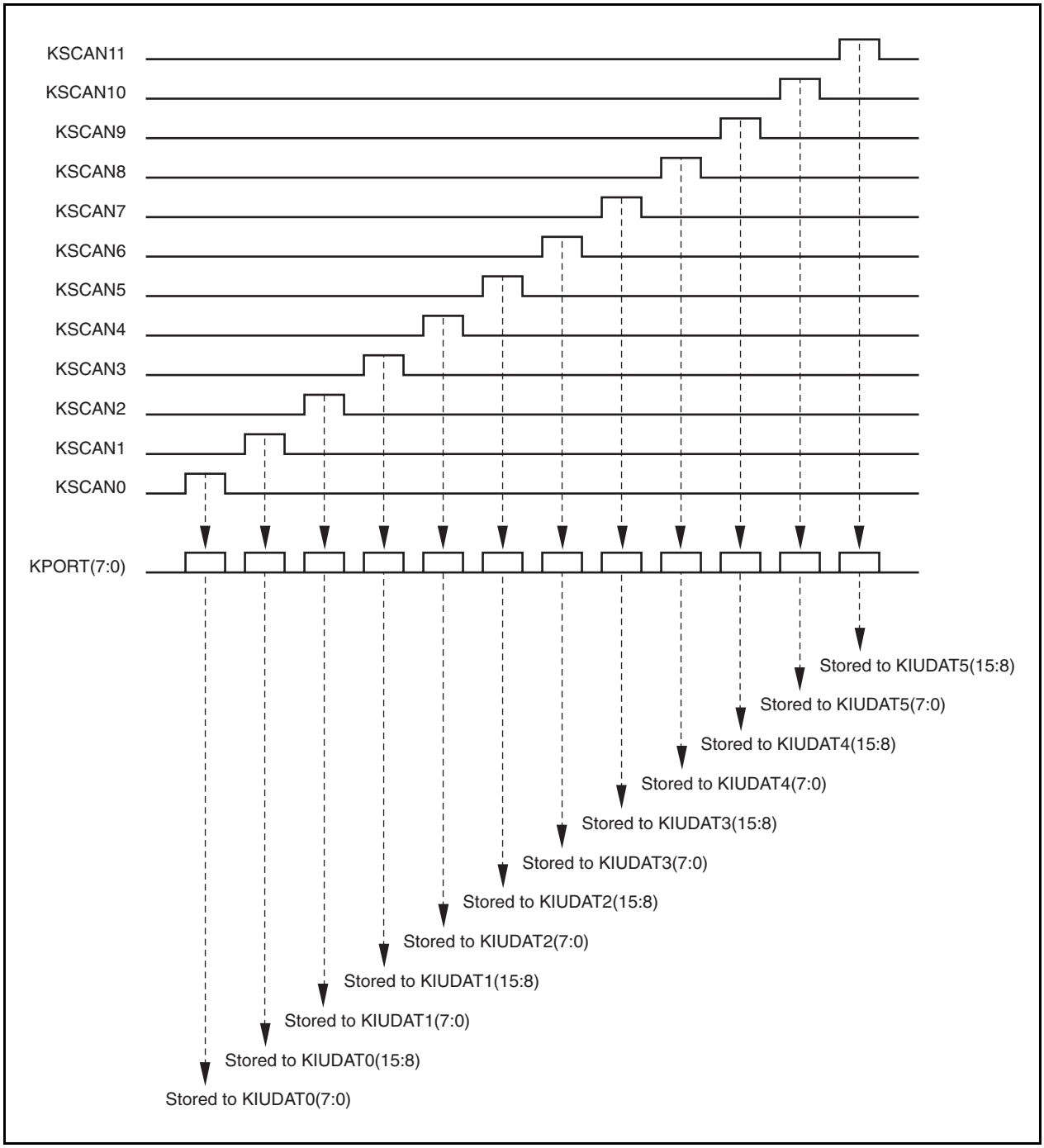
How scan data is input to the registers is as below. Figure 11-1 shows a scan operation and storing timing.

Register	Bits	Data
KIUDAT0	KEYDAT(7:0)	Stores the data scanned by the KSCAN0 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN1 pin.
KIUDAT1	KEYDAT(7:0)	Stores the data scanned by the KSCAN2 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN3 pin.
KIUDAT2	KEYDAT(7:0)	Stores the data scanned by the KSCAN4 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN5 pin.
KIUDAT3	KEYDAT(7:0)	Stores the data scanned by the KSCAN6 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN7 pin.
KIUDAT4	KEYDAT(7:0)	Stores the data scanned by the KSCAN8 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN9 pin.
KIUDAT5	KEYDAT(7:0)	Stores the data scanned by the KSCAN10 pin.
	KEYDAT(15:8)	Stores the data scanned by the KSCAN11 pin.

The data in the KIUDAT00 to KIUDAT05 registers should be read out in the interval time between two key scan operations. Scan interval is set by the KIUWKI register.

When data is not read before the next key scan operation starts, the key scan data lost interrupt occurs (see **11.2.6**). The data registers KIUDAT00 through KIUDAT05 overwrite the following scan data.

Figure 11-1. Scan Operation and Key Data Store Register



11.2.2 KIUSCANREP (base address + 0x110)

Bit	15	14	13	12	11	10	9	8
Name	KEYEN	RFU	RFU	RFU	RFU	RFU	STPREP5	STPREP4
R/W	R/W	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	STPREP3	STPREP2	STPREP1	STPREP0	SCANSTP	SCANSTART	ATSTP	ATSCAN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
15	KEYEN	Key scan enable 1: Enable 0: Disable
14:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9:4	STPREP(5:0)	KIU sequencer stop count setting 111111: 63 times : 000001: 1 time 000000: Reserved
3	SCANSTP	Key scan stop 1: Stop 0: Operate
2	SCANSTART	Key scan start 1: Start 0: Stop
1	ATSTP	Key auto stop setting 1: Auto stop 0: Not auto stop
0	ATSCAN	Key auto scan setting 1: Auto scan 0: Not auto scan

This register is used to enable operation of the key scan unit and to make settings for key scan and the KIU sequencer.

When the number of scan lines is set to 0 in the SCANLINE register, the KEYEN bit cannot be set to 1. Each mode is described in detail below.

- Key scan stop
  The SCANSTP bit should be set to 1 when the KIU sequencer stops the key scan operation in Scanning or IntervalNextScan mode.
  When this bit is set to 1, the key scan operation stops. However, if this bit is set to 1 during a key scan operation, the KIU sequencer stops after the current set of key data is received.
  This bit becomes 0 when the key scan operation stops.
  When the key scan operation is started by setting this bit to 1 during Stopped or WaitKeyIn state, the key scan operation stops immediately after a set of key scan operation is completed.

- Key scan start
  When the SCANSTART bit is set to 1, the KIU sequencer starts regardless of key contact detection.
  This bit becomes 0 when the key scan operation starts.
  This bit cannot be set while the KEYEN bit is 0.

- Key scan auto stop setting
  In the key scan auto stop mode, the key scan operation stops automatically when the data of all zeros is input to the KPORT(7:0) pins (no key contact is detected).
  The number of zeros is set by the STPREP(5:0) area.

- Key auto scan setting
  When the ATSCAN bit is set to 1, the key touch wait state is entered, and key scan operation starts automatically upon a key touch (1 is input to any of the KPORT(7:0) pins).
  When the KEYEN bit is 0, the key touch wait state is not entered even if this bit is set to 1. The key wait state is entered and the key auto scan mode is set from the point when the KEYEN bit is set to 1.

For details, see **Figure 11-4 Transition of Sequencer Status** and **Figure 11-5 Basic Operation Timing Chart**.

11.2.3 KIUSCANS (base address + 0x112)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	SSTAT1	SSTAT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	SSTAT(1:0)	KIU sequencer status 11: Scanning 10: IntervalNextScan 01: WaitKeyIn 00: Stopped

This register indicates the current KIU sequencer status.
Details of the status of the KIU sequencer are described below.

- Scanning: This is the state where the KIU sequencer performs key scan to load key data.
- IntervalNextScan: This is the state where the scan of a set of key data[Note] has completed and the start of the next key scan is being waited for. The interval time is set in the KIUWKI register.

Note The number of bits differs according to the number of KSCAN pins used. The number of KSCAN pins is set in the SCANLINE register.

KSCAN Pins	Data Bits
8	64 bits
10	80 bits
12	96 bits

- WaitKeyIn: This is the state of waiting for key input in the key auto scan mode. When the ATSCAN bit of the KIUSCANREP register is set to 1 and the KIU sequencer is enabled, key input is waited for. In this state, all the KSCAN pin[Note] outputs are high level. Prior to shifting the CPU into Suspend mode, KIU must always be set in auto scan mode and whether the state of the sequencer is WaitKeyIn must be confirmed.

Note The setting of the SCANLINE register's LINE(1:0) area determines the number of KSCAN pins used, as follows.

LINE(1:0)	KSCAN Pins
10	8
01	10
00	12

- Stopped: This is the state where the KIU sequencer is disabled.

11.2.4 KIUWKS (base address + 0x114)

Bit	15	14	13	12	11	10	9	8
Name	RFU	T3CNT4	T3CNT3	T3CNT2	T3CNT1	T3CNT0	T2CNT4	T2CNT3
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	1	1	1	1	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	T2CNT2	T2CNT1	T2CNT0	T1CNT4	T1CNT3	T1CNT2	T1CNT1	T1CNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
15	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
14:10	T3CNT(4:0)	Wait time setting ((T3CNT(4:0) + 1) × 30 μs) 11111: 960 μs : 00001: 60 μs 00000: Reserved
9:5	T2CNT(4:0)	Off time setting ((T2CNT(4:0) + 1) × 30 μs) 11111: 960 μs : 00001: 60 μs 00000: Reserved
4:0	T1CNT(4:0)	Stabilization time setting ((T1CNT(4:0) + 1) × 30 μs) 11111: 960 μs : 00001: 60 μs 00000: Reserved

This register is used to set the wait time between when the KIU sequencer sets the KSCAN signal active during a key matrix scan and when the status is read from the KPORT signal.

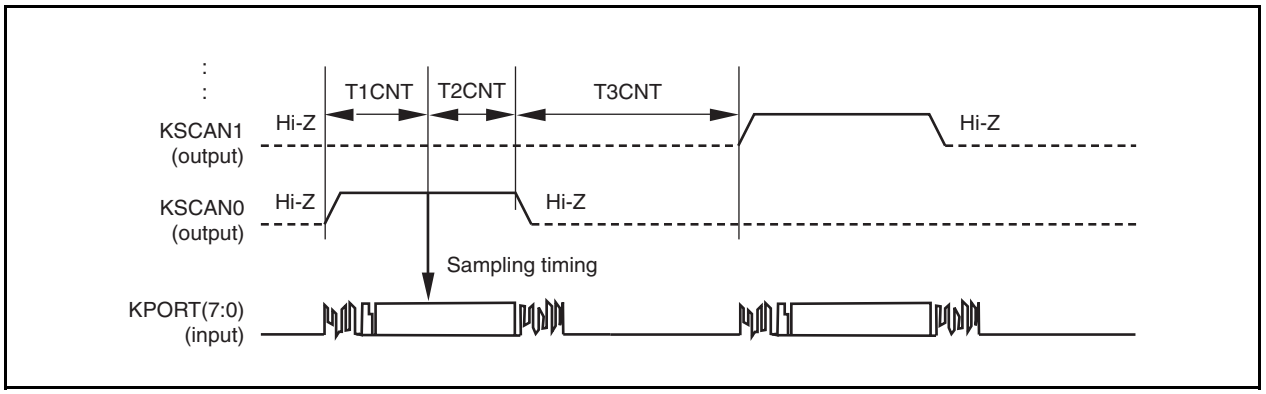
The T1CNT(4:0) area is used to set the stabilization time between when the KSCAN signal becomes high and when the key scan data is read.

The T2CNT(4:0) area is used to set the time between when the key data is read and when the KSCAN signal becomes high impedance.

The T3CNT(4:0) area is used to set the time between when the KSCAN signal becomes high impedance and when it becomes high again.

The status of output from the KSCAN signal and the timing of KPORT signal sampling are shown below.

Figure 11-2. KSCAN Signal Status and KPORT Signal Sampling Timing



11.2.5 KIUWKI (base address + 0x116)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	WINTVL9	WINTVL8
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

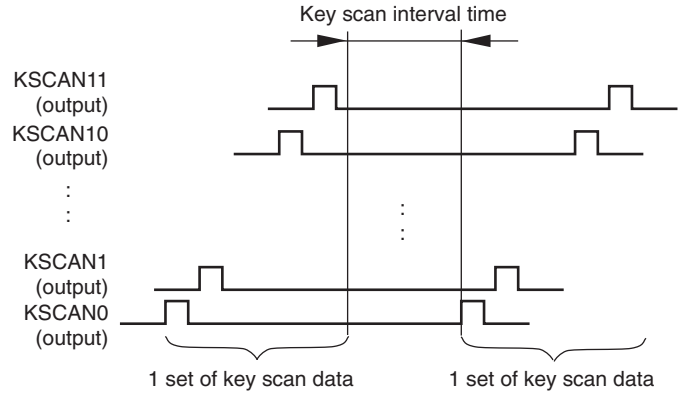
Bit	7	6	5	4	3	2	1	0
Name	WINTVL7	WINTVL6	WINTVL5	WINTVL4	WINTVL3	WINTVL2	WINTVL1	WINTVL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9:0	WINTVL(9:0)	Key scan interval time setting (WINTVL(9:0) × 30 μs) 1111111111: 30,690 μs : 0000000001: 30 μs 0000000000: No wait

This register is used to set the interval time between when one set of key data is obtained by the KIU sequencer and when the next set of key data is obtained.

The following figure shows the key scan interval time.

Figure 11-3. Key Scan Interval



11.2.6 KIUINT (base address + 0x118)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	KDATLOST	KDATRDY	SCANINT
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2	KDATLOST	Key scan data lost interrupt. Cleared to 0 when 1 is written. 1: Yes 0: No
1	KDATRDY	Key data scan complete interrupt. Cleared to 0 when 1 is written. 1: Yes 0: No
0	SCANINT	Key input detection interrupt. Cleared to 0 when 1 is written. 1: Yes 0: No

This register indicates the type of interrupt that has occurred in the KIU.

The key scan data lost interrupt occurs when data is not read out from the KIU data register (KIUDAT0 through KIUDAT5) between when data is input to the KIU data register after a key scan and when the next scan operation starts. The contents of the KIU data registers are overwritten to the new key scan data.

Key data scan complete interrupt occurs when all the key data is input after one scan operation is completed.

Key input detection interrupt occurs in the key auto scan mode when a key touch is detected (1 is detected from any of the KPORT(7:0) pins) in the key touch wait state, when a key scan operation starts after setting the start of key scan, or when a key scan operation starts after returning from the Suspend mode upon key touch detection.

11.2.7 KIURST (base address + 0x11A)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	KIURST
R/W	R	R	R	R	R	R	R	W
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0

Bit	Name	Function
15:1	RFU	Reserved. Write 0 to these bits. The value is undefined after a read.
0	KIURST	KIU reset. Cleared to 0 when 1 is written. 1: Reset 0: Normal operation

This register is used to reset the KIU registers.

11.2.8 SCANLINE (base address + 0x11E)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

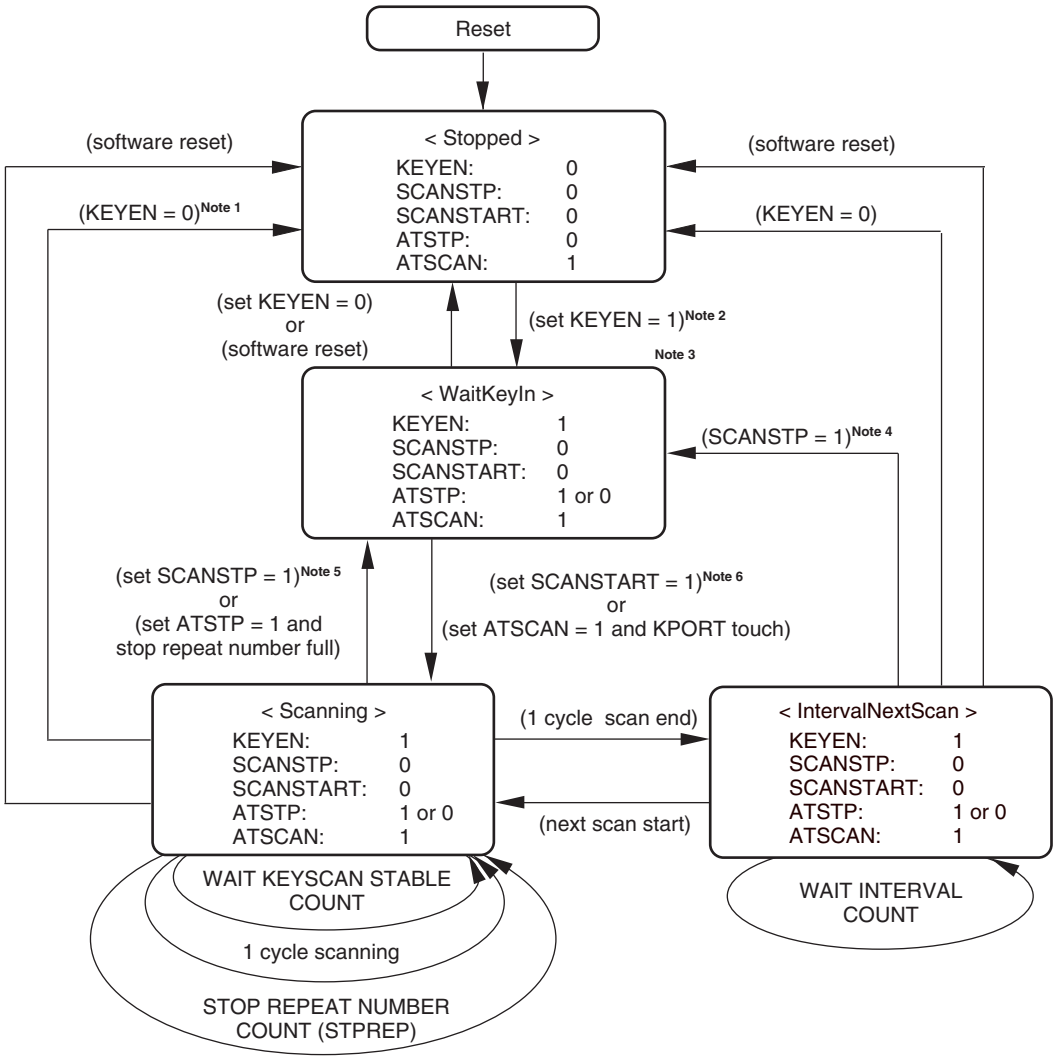
Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	LINE1	LINE0
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	LINE(1:0)	KSCAN pin use/do not use setting 11: Do not use KSCAN pins for key scan (All the KSCAN pins are used as general-purpose I/O ports or PS/2 ports) 10: Use eight key scan pins (KSCAN(7:0)) (Supports 64 keys, the remaining four pins can be used as PS/2 port) 01: Use ten key scan pins (KSCAN(9:0)) (Supports 80 keys, the remaining two pins can be used as PS2CH1 port) 00: Use twelve key scan pins (KSCAN(11:0)) (Supports 96 keys, no pins can be used as general-purpose I/O port and PS/2 port)

This register is used to switch the number of scan lines.

Select the alternate function pins with the SELECTREG register of the GIU (see **8.2.15**).

Figure 11-4. Transition of Sequencer Status



Remark

Stopped:	KIUSCANS register bit 1 = 0, bit 0 = 0
WaitKeyIn:	KIUSCANS register bit 1 = 0, bit 0 = 1
IntervalNextScan:	KIUSCANS register bit 1 = 1, bit 0 = 0
Scanning:	KIUSCANS register bit 1 = 1, bit 0 = 1
KEYEN:	KIUSCANREP register bit 15
STPREP:	KIUSCANREP register bits 9, 8, 7, 6, 5, 4
SCANSTP:	KIUSCANREP register bit 3
SCANSTART:	KIUSCANREP register bit 2
ATSTP:	KIUSCANREP register bit 1
ATSCAN:	KIUSCANREP register bit 0
software reset:	KIURST register bit 0 = 1 write
KPORT touch:	When any of KPORT(7:0) signals is 1
stop repeat number full:	When the scan data is 0 for the number of times specified by the STPREP register

Notes
1. When the KEYEN is set to 0 during a scanning operation, the status changes to the Stopped status after that scanning operation has completed.
2. The KEYEN bit cannot be set to 1 while both bits 1 and 0 of the SCANLINE register are 1.
3. When the status changes from the WaitKeyIn mode to the Scanning mode after the SCANSTP bit is set to 1, the status returns to the WaitKeyIn mode again after scanning a set of data.
4. When the SCANSTP bit is set to 1 in the IntervalNextScan mode, the status changes to the WaitKeyIn mode and the SCANSTP bit becomes 0 simultaneously.
5. If the SCANSTP bit is set to 1 during a scanning operation, that one set of data scanning is continued. After this scanning is completed, the status changes to the WaitKeyIn mode and the SCANSTP bit becomes 0 automatically.
6. The SCANSTART bit becomes automatically 0 when the status changes to the Scanning mode, except if the SCANSTART bit was set to 1 during the IntervalNextScan or Scanning mode.

Figure 11-5. Basic Operation Timing Chart (1/2)

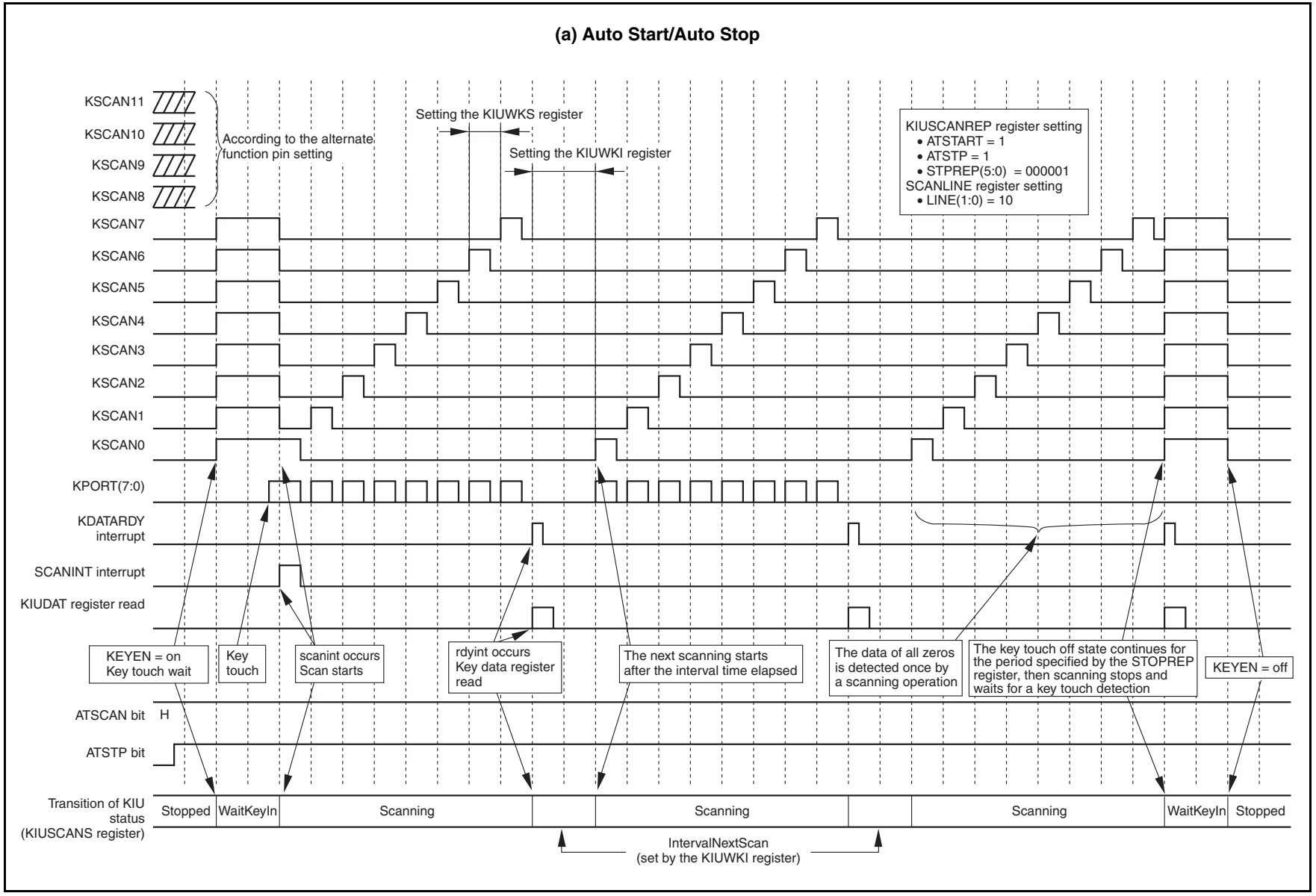
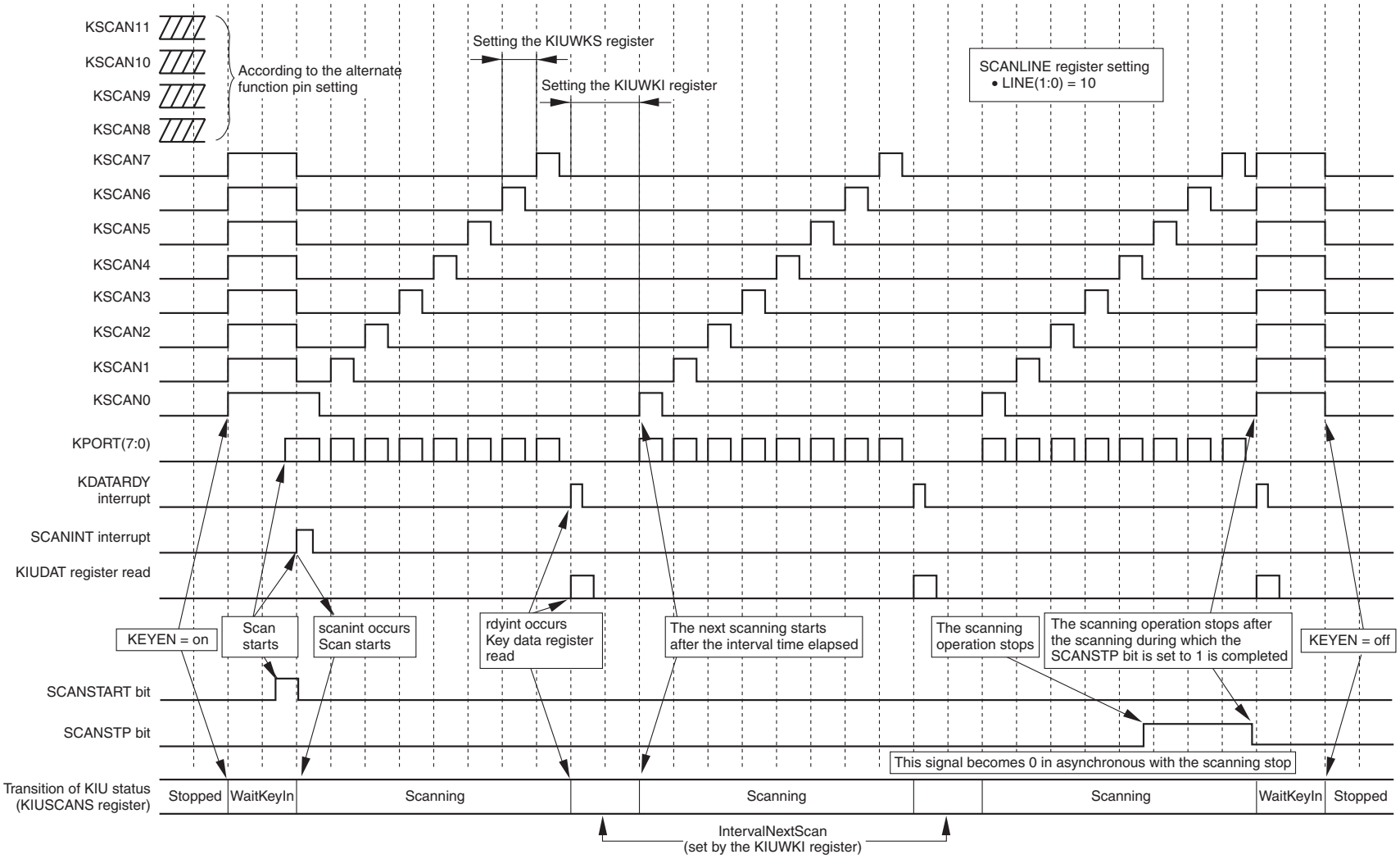


Figure 11-5. Basic Operation Timing Chart (2/2)

(b) Key Scan Start/Key Scan Stop



CHAPTER 12 PS2U (PS/2 UNIT)

12.1 General

The PS2U controls the PS/2 interface with two channels, PS2CH1 and PS2CH2.

The PS/2 interface performs bidirectional data transfers by using the PS2CLK1n and PS2DATA1n signals (n = 1, 2).

The PS2U pins are alternate function pins that are shared with the KIU pins. Use the SELECTREG register of the GIU (see **8.2.15**) to select the functions of the alternate function pins.

12.2 Register Set

Table 12-1 lists the PS2CH1 registers (for channel 1).

Table 12-2 lists the PS2CH2 registers (for channel 2).

Table 12-1. PS2CH1 Registers

Physical Address	R/W	Register Symbol	Function
BASE + 0x120	R/W	PS2CH1DATA	PS/2 channel 1 transmission/reception data register
BASE + 0x122	R/W	PS2CH1CTRL	PS/2 channel 1 control register
BASE + 0x124	W	PS2CH1RST	PS/2 channel 1 reset register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

Table 12-2. PS2CH2 Registers

Physical Address	R/W	Register Symbol	Function
BASE + 0x140	R/W	PS2CH2DATA	PS/2 channel 2 transmission/reception data register
BASE + 0x142	R/W	PS2CH2CTRL	PS/2 channel 2 control register
BASE + 0x144	W	PS2CH2RST	PS/2 channel 2 reset register

Remark BASE: Base address. This is set by using the BADR register of the BCU (see **3.2.11**).

These registers are described in detail below.

12.2.1 PS2CHnDATA (base address + 0x120, base address + 0x140)

Remark n = 1, 2
PS2CH1DATA (base address + 0x120)
PS2CH2DATA (base address + 0x140)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

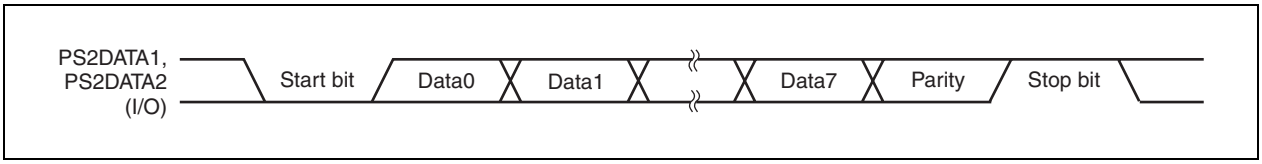
Bit	7	6	5	4	3	2	1	0
Name	PSDATA7	PSDATA6	PSDATA5	PSDATA4	PSDATA3	PSDATA2	PSDATA1	PSDATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:8	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
7:0	PSDATA(7:0)	PS/2 transmission/reception data When writing: Transmission data When reading: Reception data

This register stores transmission data that is output from the PS2DATAn pins or reception data that is input to the PS2DATAn pins (n = 1, 2). Reception data can be obtained by reading this register.

The PS2U can transmit/receive data in the following pattern.

Figure 12-1. Data Pattern



12.2.2 PS2CHnCTRL (base address + 0x122, base address + 0x142)

Remark n = 1, 2
PS2CH1CTRL (base address + 0x122)
PS2CH2CTRL (base address + 0x142)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	PERR	RVEN	INTEN	PS2EN	TEMT	REMT
R/W	R	R	R	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5	PERR	Reception data parity error detection (valid only when REMT bit = 1, even parity). 1: An error is detected 0: No error is detected
4	RVEN	Allows/denies use of reception FIFO. 1: Use 0: Do not use
3	INTEN	Enables/disables interrupt request detection. 1: Enable 0: Disable
2	PS2EN	Enables/disables use of PS/2 interface. 1: Disable 0: Enable
1	TEMT	Transmission data ready 1: Transmission data exists 0: No transmission data exists
0	REMT	Reception data ready 1: Reception data exists 0: No reception data exists

This register is used to set various types of controls and display status information for the PS/2 interface.

When the PS2EN bit is set to 0, data can be transmitted/received by using the PS/2 interface. When the RVEN bit is 1, the reception FIFO (8 bits × 8 stages) is used. When data is received from an external source, the REMT bit becomes 1, which indicates that reception data exists in the PS2CHnDATA register (n = 1, 2). This bit is cleared to 0 by reading the reception data. When the reception FIFO is used, this bit is cleared to 0 when all reception data is read and the FIFO is empty.

The INTEN bit controls whether detection of reception completion interrupt requests is enabled or disabled. If the INTEN bit is 1, when the reception of one group of data from the PS2DATAn pins is completed, the REMT bit is set to 1 and an interrupt request is reported to the CPU at the same time (n = 1, 2). When the CPU reads the reception data from the PS2CHnDATA register, it inactivates the interrupt request signal (n = 1, 2).

The PERR bit is set to 1 when a parity error is detected for the reception data. Be sure to always confirm the PERR bit before reading reception data.

Before setting this register, use the SELECTREG register of the GIU (see **8.2.15**) to set the alternate-function pins.

12.2.3 PS2CHnRST (base address + 0x124, base address + 0x144)

Remark n = 1, 2
PS2CH1RST (base address + 0x124)
PS2CH2RST (base address + 0x144)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	PS2RST
R/W	R	R	R	R	R	R	R	W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	PS2RST	PS/2 reset. Cleared to 0 when 1 is written. 1: Reset 0: Normal operation

This register is used when the PS/2U is internally reset.

12.3 Transmission Procedure

Use the following procedure to transmit data.

<1> Set the PS2EN bit of the PS2CHnCTRL register to 1 to disable reception.
<2> After waiting 100 μs, confirm whether any reception data exists.
<3> If reception data exists, read all of the reception data (until the REMT bit of the PS2CHnCTRL register becomes 0).
<4> Set transmission data in the PS2CHnDATA register (the TEMT bit of the PS2CHnCTRL register will become 1).
<5> After waiting 100 μs (Inhibit output time), set the PS2EN bit to 0 to begin transmission.
<6> Poll the TEMT bit to check for the completion of transmission.

Caution To transmit data continuously, after confirming that the TEMT bit has become 0 (completion of transmission), perform the next transmission operation according to the transmission procedure.

Remarks 1. n = 1, 2

2. Inhibit output time: Delay time from when the transmission data is set until transmission is enabled.

CHAPTER 13 CARDU1, CARDU2 (PC CARD UNITS)

★ 13.1 General

The VRC4173, which has two on-chip PC card unit (CARDU) channels for controlling PC cards that are compliant with the 1997 PC Card Standard, supports a total of two card slots.

Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card).

The main specifications are as follows.

- Compliant with 1997 PC Card Standard (excluding 32-bit PC card).
- Supports 5 V and 3 V cards.
- Supports PCI interrupts for interrupts to the system.
- Independent on-chip read/write buffers for each direction.
- Five memory windows and two I/O windows can be set.
- Supports the clock run protocol for the PCI.

However, the following functions are not supported.

- 32-bit PC card (CardBus card)
- Distributed DMA functions
- DMA between the PC card and PCI bus
- PCI power management functions (Revision 0.6)
- ZOOM Video mode
- Ring Indicate signal

13.2 Configuration Register Set

Table 13-1 lists the configuration registers.

CARDU1 and CARDU2 each have these registers.

Table 13-1. CARDU Configuration Registers (1/2)

Offset Address	R/W	Register Symbol	Function
0x00 to 0x01	R	VID	Vendor ID register
0x02 to 0x03	R	DID	Device ID register
0x04 to 0x05	R/W	PCICMD	PCI command register
0x06 to 0x07	R/W	PCISTS	PCI device status register
0x08	R	RID	Revision ID register
0x09 to 0x0B	R	CLASSC	Class code register
0x0C	R/W	CACHELS	Cache line size register
0x0D	R/W	MLT	Master latency timer register
0x0E	R	HEDT	Header type register
0x0F	R	BIST	Built-in self-test register
0x10 to 0x13	R/W	CSRBADR	CardBus socket/ExCA base address register
0x14	R	CAP	PCI additional specifications code register
0x15	–	–	Reserved
0x16 to 0x17	R/W	SECSTS	Second status register
0x18	R/W	PCIBNUM	PCI bus number register
0x19	R/W	CARDNUM	Card number register
0x1A	R/W	SUBBNUM	Subordinate bus number register
0x1B	R/W	CLT	CardBus latency timer register
0x1C to 0x1F	R/W	MEMB0	Memory base address register 0
0x20 to 0x23	R/W	MEML0	Memory space boundary register 0
0x24 to 0x27	R/W	MEMB1	Memory base address register 1
0x28 to 0x2B	R/W	MEML1	Memory space boundary register 1
0x2C to 0x2F	R/W	IOB0	I/O base address register 0
0x30 to 0x33	R/W	IOL0	I/O space boundary register 0
0x34 to 0x37	R/W	IOB1	I/O base address register 1
0x38 to 0x3B	R/W	IOL1	I/O space boundary register 1

Table 13-1. CARDU Configuration Registers (2/2)

Offset Address	R/W	Register Symbol	Function
0x3C	R/W	INTL	Interrupt line register
0x3D	R	INTP	Interrupt pin register
0x3E to 0x3F	R/W	BRGCNT	Bridge control register
0x40 to 0x41	R/W	SUBVID	Subsystem vendor ID register
0x42 to 0x43	R/W	SUBID	Subsystem ID register
0x44 to 0x47	R/W	PC16BADR	PC card 16-bit interface legacy mode base address register
0x48 to 0x7F	–	–	Reserved
0x80 to 0x83	R/W	SYSCNT	System control register
0x84 to 0x90	–	–	Reserved
0x91	R/W	DEVCNT	Device control register
0x92 to 0x93	–	–	Reserved
0x94 to 0x97	R/W	SKDMA0	Socket DMA register 0
0x98 to 0x9B	R/W	SKDMA1	Socket DMA register 1
0x9C	R/W	CHIPCNT	Chip control register
0x9D to 0x9E	–	–	Reserved
0x9F	R/W	SERRDIS	SERR# signal disable register
0xA0	R	CAPID	Capability ID register
0xA1	R	NIP	Power management additional function register
0xA2 to 0xA3	R/W	PMC	Power management characteristic register
0xA4 to 0xA5	R/W	PMCSR	Power management control/status register
0xA6	R	PMCSR_BSE	PMCSR bridge support extension register
0xA7	R	DATA	Data register
0xA8 to 0xFB	–	–	Reserved
0xFC	R/W	TEST	Test register
0xFD to 0xFF	–	–	Reserved

These registers are described in detail below.

13.2.1 VID (offset address: 0x00 to 0x01)

Bit	15	14	13	12	11	10	9	8
Name	VID15	VID14	VID13	VID12	VID11	VID10	VID9	VID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	1	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	VID7	VID6	VID5	VID4	VID3	VID2	VID1	VID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	1	1	0	0	1	1

Bit	Name	Function
15:0	VID(15:0)	Vendor ID 0x1033: NEC

13.2.2 DID (offset address: 0x02 to 0x03)

Bit	15	14	13	12	11	10	9	8
Name	DID15	DID14	DID13	DID12	DID11	DID10	DID9	DID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DID7	DID6	DID5	DID4	DID3	DID2	DID1	DID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	1	1	1	1	1	0

Bit	Name	Function
15:0	DID(15:0)	Device ID 0x003E: CARDU

13.2.3 PCICMD (offset address: 0x04 to 0x05)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	FBTB_EN	SERR_EN
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AD_STEP	PERR_EN	VGA_P_ SNOOP	MEMW_ INV_EN	SP_CYC	MASTER_ EN	MEM_EN	IO_EN
R/W	R	R/W	R/W	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	FBTB_EN	Enables/disables fast Back to Back. This function is not supported by the CARDU.
8	SERR_EN	Enables/disables system errors. 1: Enable The SERR# signal is set to active if an address parity error is detected and the PERR_EN bit is 1. 0: Disable
7	AD_STEP	Enables/disables address/data stepping. This function is not supported by the CARDU.
6	PERR_EN	Enables/disables parity error. 1: Enable output of the PERR# signal The PERR# signal is set to active if a data parity error is detected. The SERR# signal is set to active if an address parity error is detected and the SERR_EN bit is 1. 0: Disable output of the PERR# signal
5	VGA_P_SNOOP	VGA palette snoop. This bit setting is valid only when the VGA_EN bit of the BRGCNT register is 0. 1: React to a write access to a VGA palette address from the PCI bus side. Do not react to a read access to a VGA palette address or to an access to another VGA address. Do not react to a write access to a VGA palette address from the PC card side, and react to a read access to a VGA palette address or to an access to another VGA address. 0: Do not react to an access to a VGA address from the PCI bus side. React to an access to a VGA address from the PC card side.
4	MEMW_INV_EN	Enables/disables memory write and invalidate. This function is not supported by the CARDU.
3	SP_CYC	Special cycle. This function is not supported by the CARDU.

(2/2)

Bit	Name	Function
2	MASTER_EN	Controls bus master operation. 1: Operate as bus master on the PCI bus. 0: Do not operate as bus master on the PCI bus.
1	MEM_EN	Controls memory space. 1: Respond to a memory access to the PC card. 0: Do not respond to a memory access to the PC card.
0	IO_EN	Controls I/O space. 1: Respond to an I/O access to the PC card. 0: Do not respond to an I/O access to the PC card.

13.2.4 PCISTS (offset address: 0x06 to 0x07)

Bit	15	14	13	12	11	10	9	8
Name	DETECT_ PERR	SIG_SERR	RV_ MABORT	RV_ TABORT	SIG_ TABOT	DEVSEL1	DEVSEL0	DETECT_ D_PERR
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	FBTB_CAP	RFU	RFU	NEW_CAP	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	1	0	0	0	0

Bit	Name	Function
15	DETECT_PERR	Data and address parity error detection. Cleared to 0 when 1 is written. 1: Detected 0: Not detected
14	SIG_SERR	SERR# signal status. Cleared to 0 when 1 is written. 1: Active 0: Inactive
13	RV_MABORT	Master abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
12	RV_TABORT	Target abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
11	SIG_TABOT	Target abort reporting. Cleared to 0 when 1 is written. 1: Reported 0: Not reported
10:9	DEVSEL(1:0)	DEVSEL# timing 01: Medium speed
8	DETECT_D_PERR	Set to 1 when the following three conditions are satisfied. Cleared to 0 when 1 is written. • The CARDU is the master of the bus cycle in which the data parity error occurred. • Either the CARDU set the PERR# signal to active or the CARDU detected that the PERR# signal became active due to the target. • The PERR_EN bit of the PCICMD register has been set to 1.
7	FBTB_CAP	Response to fast Back to Back. This is fixed at 1 (enabled).
6:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4	NEW_CAP	Use of PCI power management. This is fixed at 1 (enabled).
3:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

This register shows the status of the PCI bus side.

13.2.5 RID (offset address: 0x08)

Bit	7	6	5	4	3	2	1	0
Name	RID7	RID6	RID5	RID4	RID3	RID2	RID1	RID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	RID(7:0)	Revision ID

13.2.6 CLASSC (offset address: 0x09 to 0x0B)

Bit	23	22	21	20	19	18	17	16
Name	CLASSC23	CLASSC22	CLASSC21	CLASSC20	CLASSC19	CLASSC18	CLASSC17	CLASSC16
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	1	1	0

Bit	15	14	13	12	11	10	9	8
Name	CLASSC15	CLASSC14	CLASSC13	CLASSC12	CLASSC11	CLASSC10	CLASSC9	CLASSC8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	1	1	1

Bit	7	6	5	4	3	2	1	0
Name	CLASSC7	CLASSC6	CLASSC5	CLASSC4	CLASSC3	CLASSC2	CLASSC1	CLASSC0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
23:0	CLASSC(23:0)	Class code 0x060700: Bridge device

13.2.7 CACHELS (offset address: 0x0C)

Bit	7	6	5	4	3	2	1	0
Name	CACHELS7	CACHELS6	CACHELS5	CACHELS4	CACHELS3	CACHELS2	CACHELS1	CACHELS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	CACHELS(7:0)	Sets the cache line size. The units are 32 bits (4 bytes).

13.2.8 MLT (offset address: 0x0D)

Bit	7	6	5	4	3	2	1	0
Name	MLT7	MLT6	MLT5	MLT4	MLT3	MLT2	MLT1	MLT0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:3	MLT(7:3)	Sets the latency timer. 11111: 38 PCLK (1140 ns) : 00010: 9 PCLK (270 ns) 00001: 8 PCLK (240 ns) 00000: 0 PCLK (0 ns)
2:0	MLT(2:0)	Write 0 to these bits. 0 is returned after a read.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.2.9 HEDT (offset address: 0x0E)

Bit	7	6	5	4	3	2	1	0
Name	HEDT7	HEDT6	HEDT5	HEDT4	HEDT3	HEDT2	HEDT1	HEDT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	1	0

Bit	Name	Function
7:0	HEDT(7:0)	Header type 0x02: This is a single function power management register definition.

13.2.10 BIST (offset address: 0x0F)

Bit	7	6	5	4	3	2	1	0
Name	BIST7	BIST6	BIST5	BIST4	BIST3	BIST2	BIST1	BIST0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	BIST(7:0)	Built-in self-test. This function is not supported by the CARDU.

13.2.11 CSRBADR (offset address: 0x10 to 0x13)

Bit	31	30	29	28	27	26	25	24
Name	CSRBADR31	CSRBADR30	CSRBADR29	CSRBADR28	CSRBADR27	CSRBADR26	CSRBADR25	CSRBADR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	CSRBADR23	CSRBADR22	CSRBADR21	CSRBADR20	CSRBADR19	CSRBADR18	CSRBADR17	CSRBADR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	CSRBADR15	CSRBADR14	CSRBADR13	CSRBADR12	CSRBADR11	CSRBADR10	CSRBADR9	CSRBADR8
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CSRBADR7	CSRBADR6	CSRBADR5	CSRBADR4	CSRBADR3	CSRBADR2	CSRBADR1	CSRBADR0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	CSRBADR(31:12)	Sets the CardBus socket register/ExCA base address.
11:0	CSRBADR(11:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.12 CAP (offset address: 0x14)

Bit	7	6	5	4	3	2	1	0
Name	CAP7	CAP6	CAP5	CAP4	CAP3	CAP2	CAP1	CAP 0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	1	0	0	0	0	0

Bit	Name	Function
7:0	CAP(7:0)	Capability pointer Indicates the offset address 0xA0 of the CAPID register (see **13.2.38**).

13.2.13 SECSTS (offset address: 0x16 to 0x17)

Bit	15	14	13	12	11	10	9	8
Name	S_DETECT_ PERR	S_SIG_ SERR	S_RV_ MABORT	S_RV_ TABORT	S_SIG_ TABOT	S_DEVSEL1	S_DEVSEL0	S_DETECT_ D_PERR
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	S_FBTB_ CAP	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	0	0	0	0	0

Bit	Name	Function
15	S_DETECT_PERR	Data and address parity error detection. Cleared to 0 when 1 is written. 1: Detected 0: Not detected
14	S_SIG_SERR	SERR# signal status. Cleared to 0 when 1 is written. 1: Active 0: Inactive
13	S_RV_MABORT	Master abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
12	S_RV_TABORT	Target abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
11	S_SIG_TABOT	Target abort reporting. Cleared to 0 when 1 is written. 1: Reported 0: Not reported
10:9	S_DEVSEL(1:0)	DEVSEL# timing 01: Medium speed
8	S_DETECT_D_PERR	Set to 1 when the following three conditions are satisfied. • The CardBus is the master of the bus cycle in which the data parity error occurred. • Either the CardBus set the PERR# signal to active or the CardBus detected that the PERR# signal became active due to the target. • The PERR_EN bit of the PCICMD register has been set to 1.
7	S_FBTB_CAP	Response to fast Back to Back. This is fixed at 1 (enabled).
6:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

This register shows the status of the CardBus side.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.14 PCIBNUM (offset address: 0x18)

Bit	7	6	5	4	3	2	1	0
Name	PCIBNUM7	PCIBNUM6	PCIBNUM5	PCIBNUM4	PCIBNUM3	PCIBNUM2	PCIBNUM1	PCIBNUM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	PCIBNUM(7:0)	PCI bus number The value of this register is set and managed by software.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.15 CARDNUM (offset address: 0x19)

Bit	7	6	5	4	3	2	1	0
Name	CARDNUM7	CARDNUM6	CARDNUM5	CARDNUM4	CARDNUM3	CARDNUM2	CARDNUM1	CARDNUM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	CARDNUM(7:0)	CardBus card number The value of this register is set and managed by software.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.16 SUBBNUM (offset address: 0x1A)

Bit	7	6	5	4	3	2	1	0
Name	SUBBNUM7	SUBBNUM6	SUBBNUM5	SUBBNUM4	SUBBNUM3	SUBBNUM2	SUBBNUM1	SUBBNUM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	SUBBNUM(7:0)	Subordinate bus number Write the maximum number among the bus numbers of busses to which the PC card is connected.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.17 CLT (offset address: 0x1B)

Bit	7	6	5	4	3	2	1	0
Name	CLT7	CLT6	CLT5	CLT4	CLT3	CLT2	CLT1	CLT0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:3	CLT(7:3)	Sets the CardBus latency timer. 11111: 38 PCLK (1140 ns) : 00010: 9 PCLK (270 ns) 00001: 8 PCLK (240 ns) 00000: 0 PCLK (0 ns)
2:0	CLT(2:0)	Write 0 to these bits. 0 is returned after a read.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.18 MEMB0 (offset address: 0x1C to 0x1F)

Bit	31	30	29	28	27	26	25	24
Name	MEMB031	MEMB030	MEMB029	MEMB028	MEMB027	MEMB026	MEMB025	MEMB024
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	MEMB023	MEMB022	MEMB021	MEMB020	MEMB019	MEMB018	MEMB017	MEMB016
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	MEMB015	MEMB014	MEMB013	MEMB012	MEMB011	MEMB010	MEMB09	MEMB08
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MEMB07	MEMB06	MEMB05	MEMB04	MEMB03	MEMB02	MEMB01	MEMB00
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	MEMB0(31:12)	Sets the memory base address 0.
11:0	MEMB0(11:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.19 MEML0 (offset address: 0x20 to 0x23)

Bit	31	30	29	28	27	26	25	24
Name	MEML031	MEML030	MEML029	MEML028	MEML027	MEML026	MEML025	MEML024
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	MEML023	MEML022	MEML021	MEML020	MEML019	MEML018	MEML017	MEML016
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	MEML015	MEML014	MEML013	MEML012	MEML011	MEML010	MEML09	MEML08
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MEML07	MEML06	MEML05	MEML04	MEML03	MEML02	MEML01	MEML00
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	MEML0(31:12)	Sets the memory space boundary 0.
11:0	MEML0(11:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.20 MEMB1 (offset address: 0x24 to 0x27)

Bit	31	30	29	28	27	26	25	24
Name	MEMB131	MEMB130	MEMB129	MEMB128	MEMB127	MEMB126	MEMB125	MEMB124
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	MEMB123	MEMB122	MEMB121	MEMB120	MEMB119	MEMB118	MEMB117	MEMB116
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	MEMB115	MEMB114	MEMB113	MEMB112	MEMB111	MEMB110	MEMB19	MEMB18
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MEMB17	MEMB16	MEMB15	MEMB14	MEMB13	MEMB12	MEMB11	MEMB10
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	MEMB1(31:12)	Sets the memory base address 1.
11:0	MEMB1(11:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.21 MEML1 (offset address: 0x28 to 0x2B)

Bit	31	30	29	28	27	26	25	24
Name	MEML131	MEML130	MEML129	MEML128	MEML127	MEML126	MEML125	MEML124
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	MEML123	MEML122	MEML121	MEML120	MEML119	MEML118	MEML117	MEML116
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	MEML115	MEML114	MEML113	MEML112	MEML111	MEML110	MEML19	MEML18
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MEML17	MEML16	MEML15	MEML14	MEML13	MEML12	MEML11	MEML10
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	MEML1(31:12)	Sets the memory space boundary 1.
11:0	MEML1(11:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.22 IOB0 (offset address: 0x2C to 0x2F)

Bit	31	30	29	28	27	26	25	24
Name	IOB031	IOB030	IOB029	IOB028	IOB027	IOB026	IOB025	IOB024
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	IOB023	IOB022	IOB021	IOB020	IOB019	IOB018	IOB017	IOB016
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	IOB015	IOB014	IOB013	IOB012	IOB011	IOB010	IOB09	IOB08
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	IOB07	IOB06	IOB05	IOB04	IOB03	IOB02	IOB01	IOB00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	IOB0(31:2)	Sets the I/O base address 0.
1:0	IOB0(1:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.23 IOL0 (offset address: 0x30 to 0x33)

Bit	31	30	29	28	27	26	25	24
Name	IOL031	IOL030	IOL029	IOL028	IOL027	IOL026	IOL025	IOL024
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	IOL023	IOL022	IOL021	IOL020	IOL019	IOL018	IOL017	IOL016
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	IOL015	IOL014	IOL013	IOL012	IOL011	IOL010	IOL09	IOL08
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	IOL07	IOL06	IOL05	IOL04	IOL03	IOL02	IOL01	IOL00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	IOL0(31:16)	Write 0 to these bits. 0 is returned after a read.
15:2	IOL0(15:2)	Sets the I/O space boundary 0.
1:0	IOL0(1:0)	Write 0 to these bits. 0 is returned after a read.

Address comparisons for the IOL0(31:16) area are performed using the values set in the IOB0 register.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.24 IOB1 (offset address: 0x34 to 0x37)

Bit	31	30	29	28	27	26	25	24
Name	IOB131	IOB130	IOB129	IOB128	IOB127	IOB126	IOB125	IOB124
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	IOB123	IOB122	IOB121	IOB120	IOB119	IOB118	IOB117	IOB116
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	IOB115	IOB114	IOB113	IOB112	IOB111	IOB110	IOB19	IOB18
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	IOB17	IOB16	IOB15	IOB14	IOB13	IOB12	IOB11	IOB10
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	IOB1(31:2)	Sets the I/O base address 1.
1:0	IOB1(1:0)	Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.25 IOL1 (offset address: 0x38 to 0x3B)

Bit	31	30	29	28	27	26	25	24
Name	IOL131	IOL130	IOL129	IOL128	IOL127	IOL126	IOL125	IOL124
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	IOL123	IOL122	IOL121	IOL120	IOL119	IOL118	IOL117	IOL116
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	IOL115	IOL114	IOL113	IOL112	IOL111	IOL110	IOL19	IOL18
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	IOL17	IOL16	IOL15	IOL14	IOL13	IOL12	IOL11	IOL10
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	IOL1(31:16)	Write 0 to these bits. 0 is returned after a read.
15:2	IOL1(15:2)	Sets the I/O space boundary 1.
1:0	IOL1(1:0)	Write 0 to these bits. 0 is returned after a read.

Address comparisons for the IOL1(31:16) area are performed using the values set in the IOB1 register.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.26 INTL (offset address: 0x3C)

Bit	7	6	5	4	3	2	1	0
Name	INTL7	INTL6	INTL5	INTL4	INTL3	INTL2	INTL1	INTL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
7:0	INTL(7:0)	Sets the interrupt request line. Since this function is not supported by the CARDU, settings for these bits are invalid.

13.2.27 INTP (offset address: 0x3D)

Bit	7	6	5	4	3	2	1	0
Name	INTP7	INTP6	INTP5	INTP4	INTP3	INTP2	INTP1	INTP0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	INTP(7:0)	PCI interrupt pin 0x01: Serial

13.2.28 BRGCNT (offset address: 0x3E to 0x3F)

(1/2)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	POST_ WR_EN	MEM1_ PREF_EN	MEM0_ PERF_EN
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	1	1

Bit	7	6	5	4	3	2	1	0
Name	IREQ_INT	CARD_RST	MABORT_ MODE	RFU	VGA_EN	ISA_EN	SERR_EN	PERR_EN
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
After reset	0	1	0	0	0	0	0	0

Bit	Name	Function
15:11	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
10	POST_WR_EN	Enables/disables posts for transactions to or from sockets. 1: Enable 0: Disable
9	MEM1_PREF_EN	Enables/disables prefetches in a memory window defined by the MEMB1 and MEML1 registers. 1: Enable 0: Disable
8	MEM0_PREF_EN	Enables/disables prefetches in a memory window defined by the MEMB0 and MEML0 registers. 1: Enable 0: Disable
7	IREQ_INT	Selects function interrupt mode from PC card. 1: Setting prohibited 0: Parallel The CARDU unit supports only parallel mode.
6	CARD_RST	Status of the CRST# signal (corresponds to the RESET1 or RESET2 signal of the V_{RC}4173) of the CardBus card. 1: Set the CRST# signal to active (initialize PC card) 0: Do not set the CRST# signal to active.

(2/2)

Bit	Name	Function
5	MABORT_MODE	Controls operation when a master abort occurs in the PCI bus and CardBus. 1: For a delayed transaction, return a target abort. For a post transaction, set the SERR# signal to active when the SERR_EN bit is 1. 0: When reading, return all 1's. When writing, discard the write data.
4	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
3	VGA_EN	Access to VGA address 1: React to an access to a VGA address from the PCI bus side. Do not react to an access to a VGA address from the CardBus card side. 0: Do not react to an access to a VGA address from the PCI bus side. However, when the VGA_P_SNOOP bit of the PCICMD register is 1, react to a write access to a VGA pallet address. React to an access to a VGA address from the CardBus card side. However, when the VGA_P_SNOOP bit is 1, do not react to a write access to a VGA pallet address.
2	ISA_EN	Reaction for an I/O window defined by the IOB1 or IOB0 register and IOL1 or IOL0 register 1: Setting prohibited 0: React according to IOB1 or IOB0 register and IOL1 or IOL0 register. The CARDU unit only supports the PCI bus.
1	SERR_EN	Enables/disables reporting of the CardBus CSERR# signal (corresponds to the WAIT1 or WAIT2 signal of the VRC4173) according to the PCI bus SERR# signal. 1: Enable 0: Disable
0	PERR_EN	Parity error processing on the CardBus 1: Check for and report parity errors 0: Ignore parity errors

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.29 SUBVID (offset address: 0x40 to 0x41)

Bit	15	14	13	12	11	10	9	8
Name	SUBVID15	SUBVID14	SUBVID13	SUBVID12	SUBVID11	SUBVID10	SUBVID9	SUBVID8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SUBVID7	SUBVID6	SUBVID5	SUBVID4	SUBVID3	SUBVID2	SUBVID1	SUBVID0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SUBVID(15:0)	Subsystem vendor ID This is a vendor identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

The read/write attribute for this register changes according to the SUB_ID_WR_EN bit of the SYSCNT register.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.30 SUBID (offset address: 0x42 to 0x43)

Bit	15	14	13	12	11	10	9	8
Name	SUBID15	SUBID14	SUBID13	SUBID12	SUBID11	SUBID10	SUBID9	SUBID8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SUBID7	SUBID6	SUBID5	SUBID4	SUBID3	SUBID2	SUBID1	SUBID0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SUBID(15:0)	Subsystem ID This is a controller identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

The read/write attribute for this register changes according to the SUB_ID_WR_EN bit of the SYSCNT register.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.31 PC16BADR (offset address: 0x44 to 0x47)

Bit	31	30	29	28	27	26	25	24
Name	PC16BADR 31	PC16BADR 30	PC16BADR 29	PC16BADR 28	PC16BADR 27	PC16BADR 26	PC16BADR 25	PC16BADR 24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	PC16BADR 23	PC16BADR 22	PC16BADR 21	PC16BADR 20	PC16BADR 19	PC16BADR 18	PC16BADR 17	PC16BADR 16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	PC16BADR 15	PC16BADR 14	PC16BADR 13	PC16BADR 12	PC16BADR 11	PC16BADR 10	PC16BADR 9	PC16BADR 8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PC16BADR 7	PC16BADR 6	PC16BADR 5	PC16BADR 4	PC16BADR 3	PC16BADR 2	PC16BADR 1	PC16BADR 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
31:2	PC16BADR(31:2)	Sets the PC card 16-bit interface legacy mode base address.
1:0	PC16BADR(1:0)	01 is returned after a read.

13.2.32 SYSCNT (offset address: 0x80 to 0x83)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	BAD_VCC_ REQ_DISB	RFU	PCPCI_EN	CH_ ASSIGN2	CH_ ASSIGN1	CH_ ASSIGN0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	SUB_ID_ WR_EN	ASYN_ INT_MODE	PCI_ CLK_RIN	RFU
R/W	R	R	R	R	R/W	R/W	R/W	R
After reset	0	0	0	0	0	1	0	0

Bit	Name	Function
31:22	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
21	BAD_VCC_REQ_DISB	Controls the BAD_VCC_REQ bit (see **13.4.3**). 1: Invalid 0: Valid
20	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
19	PCPCI_EN	Enables/disables DMA between the PC card and PCI bus. 1: Setting prohibited 0: Disable The CARDU unit does not support DMA.
18:16	CH_ASSIGN(2:0)	Sets DMA channel between the PC card and PCI bus. The CARDU unit does not support DMA. Set 100 for these bits.
15:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	SUB_ID_WR_EN	Sets read/write attribute of SUBVID and SUBID registers. 1: Read/writable 0: Read only

(2/2)

Bit	Name	Function
2	ASYN_INT_MODE	Synchronous/asynchronous setting for interrupt request samples from the PC card. 1: Sample interrupt requests other than for card insertion/removal asynchronously relative to the clock 0: Sample interrupt requests synchronously with the clock
1	PCI_CLK_RIN	PCI bus clock run control setting 1: Drive the primary CLKRUN# signal so that the PCICLK driven by CPU does not stop 0: Operate normally according to the clock run protocol
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.33 DEVCNT (offset address: 0x91)

Bit	7	6	5	4	3	2	1	0
Name	RFU	ZOOM_ VIDEO_EN	RFU	SR_PCI_ INT_SEL1	SR_PCI_ INT_SEL0	PCI_INT_ MODE	IRQ_MODE	IFG
R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
6	ZOOM_VIDEO_EN	Support for ZV (ZOOM Video) card 1: Setting prohibited 0: Disable support The CARDU unit does not support the ZV card.
5	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
4:3	SR_PCI_INT_SEL(1:0)	PCI serial interrupt request channel specification The CARDU unit does not support serial interrupts. Set 00 for these bits.
2	PCI_INT_MODE	PCI interrupt request output control 1: Setting prohibited 0: Output interrupt requests in parallel The CARDU unit supports only parallel output.
1	IRQ_MODE	IRQ interrupt request output control 1: Setting prohibited 0: Output interrupt requests in parallel The CARDU unit supports only parallel output.
0	IFG	PC card function interrupt request detection. Cleared to 0 when 1 is written. 1: Detected 0: Not detected

13.2.34 SKDMA0 (offset address: 0x94 to 0x97)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	DMA_PIN_CONFIG1	DMA_PIN_CONFIG0
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	DMA_PIN_CONFIG(1:0)	DMA request signal allocation control

Caution The CARDU of the VRC4173 does not support DMA. Therefore, the default values should be set for this register.

13.2.35 SKDMA1 (offset address: 0x98 to 0x9B)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DMA_ BADR11	DMA_ BADR10	DMA_ BADR9	DMA_ BADR8	DMA_ BADR7	DMA_ BADR6	DMA_ BADR5	DMA_ BADR4
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DMA_ BADR3	DMA_ BADR2	DMA_ BADR1	DMA_ BADR0	RFU	DMA_TRANS _SIZE1	DMA_TRANS _SIZE0	DMA_ DEC_EN
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:4	DMA_BADR(11:0)	Sets DMA register base address in the PCI I/O space.
3	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
2:1	DMA_TRANS_SIZE(1:0)	16-bit PC card DMA transfer data size
0	DMA_DEC_EN	Enables DMA base address decoding.

Caution The CARDU of the VRC4173 does not support DMA. Therefore, the default values should be set for this register.

13.2.36 CHIPCNT (offset address: 0x9C)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	S_PREF_DISB	RFU	RFU	RFU	RFU
R/W	R	R	R	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4	S_PREF_DISB	Enables/disables prefetch reads from CardBus. 1: Disable 0: Enable
3:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.37 SERRDIS (offset address: 0x9F)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	SERR_DIS_MAB	SERR_DIS_TAB	SERR_DIS_DT_PERR	RFU	RFU
R/W	R	R	R	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4	SERR_DIS_MAB	Enables/disables SERR# signal output when a master abort occurs during a post write operation. 1: Disable 0: Enable
3	SERR_DIS_TAB	Enables/disables SERR# signal output when a target abort occurs during a post write operation. 1: Disable 0: Enable
2	SERR_DIS_DT_PERR	Enables/disables SERR# signal output when a data parity error occurs during a post write operation. However, when the data parity error occurs on the PC card-side bus, this bit is also valid during operations other than post write operations. 1: Disable 0: Enable
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

★ **Caution The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card). Therefore, the default values should be set for this register.**

13.2.38 CAPID (offset address: 0xA0)

Bit	7	6	5	4	3	2	1	0
Name	CAPID7	CAPID6	CAPID5	CAPID4	CAPID3	CAPID2	CAPID1	CAPID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	CAPID(7:0)	Capability ID 0x01: Power management function

13.2.39 NIP (offset address: 0xA1)

Bit	7	6	5	4	3	2	1	0
Name	NIP7	NIP6	NIP5	NIP4	NIP3	NIP2	NIP1	NIP0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	NIP(7:0)	Power management additional function pointer. The CARDU unit does not have additional functions.

13.2.40 PMC (offset address: 0xA2 to 0xA3)

Bit	15	14	13	12	11	10	9	8
Name	PME_ SUPPORT4	PME_ SUPPORT3	PME_ SUPPORT2	PME_ SUPPORT1	PME_ SUPPORT0	D2_ SUPPORT	D1_ SUPPORT	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	1	1	1	1	1	1	1	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	DSI	AUX_PWR_ SOURCE	PME_CLK	VERSION2	VERSION1	VERSION0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	1	1	0	0	1

Bit	Name	Function
15	PME_SUPPORT4	Enables the PME# signal (internal signal) to be active during a D3Cold state.
14:11	PME_SUPPORT(3:0)	PME# signal active condition
10	D2_SUPPORT	Supports D2
9	D1_SUPPORT	Supports D1
8:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5	DSI	Necessity of initial settings other than PCI standard settings
4	AUX_PWR_SOURCE	Enables the PME# signal to be active during a D3Cold state when an auxiliary power supply is used.
3	PME_CLK	Necessity of the PCICLK when the PME# signal (internal signal) is generated
2:0	VERSION(2:0)	Supports the PCI Power Management Interface Specification 1.0.

Remark D3Cold: State name when switching from the D3 state to the D0 state due to a hardware reset.
D3Hot: State name when switching from the D3 state to the D0 state due to a software reset.

Caution Since the CARDU of the VRC4173 does not support power management functions, this register setting is disabled. Set the default values for this register.

13.2.41 PMCSR (offset address: 0xA4 to 0xA5)

Bit	15	14	13	12	11	10	9	8
Name	PME_ STATUS	DATA_ SCALE1	DATA_ SCALE0	DATA_ SEL3	DATA_ SEL2	DATA_ SEL1	DATA_ SEL0	PME_EN
R/W	R/W	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	PWR_ STATE1	PWR_ STATE0
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	PME_STATUS	PME# signal (internal signal) status. Cleared to 0 when 1 is written.
14:13	DATA_SCALE(1:0)	Sets the power management DATA register basic time
12:9	DATA_SEL(3:0)	Selects the power management DATA register
8	PME_EN	Enables PME# signal (internal signal) output
7:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	PWR_STATE(1:0)	Determines the power state

Caution Since the CARDU of the VRC4173 does not support power management functions, this register setting is disabled. Set the default values for this register.

13.2.42 PMCSR_BSE (offset address: 0xA6)

Bit	7	6	5	4	3	2	1	0
Name	BPCC_EN	B2_B3#	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	1	1	0	0	0	0	0	0

Bit	Name	Function
7	BPCC_EN	Enables bus power supply and clock control
6	B2_B3#	Supports B2 and B3 status during a D3Hot state.
5:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

Caution Since the CARDU of the VRC4173 does not support power management functions, this register setting is disabled. Set the default values for this register.

13.2.43 DATA (offset address: 0xA7)

Bit	7	6	5	4	3	2	1	0
Name	DATA7	DATA6	DATA5	DATA4	DATA3	DATA2	DATA1	DATA0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	DATA(7:0)	Write 0 to these bits. 0 is returned after a read. The CARDU unit does not support this function.

13.2.44 TEST (offset address: 0xFC)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	TEST1	TEST2
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1	TEST1	Sets the test mode (interlocation test when the PC card is inserted). 1: Test mode 0: Normal operation
0	TEST2	Sets the test mode (test when supplying power to PC card). 1: Test mode 0: Normal operation

13.3 ExCA Register Set

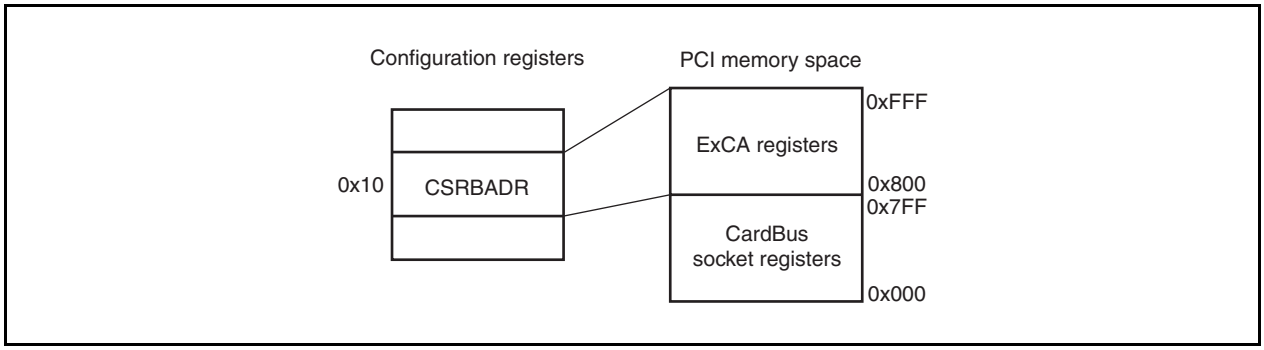
There are two methods of accessing these registers. Each access method is explained below.

(1) Access according to a memory access from the primary side

When all of the following conditions are satisfied, the ExCA registers can be accessed according to a memory access from the primary side.

- The higher 20 bits of the address match the higher 20 bits of the CSRBADR register within the configuration registers.
- The lower 12 bits of the address are in the range 0x800 to 0xFFF.
- The MEM_EN bit of the PCICMD register within the configuration registers is set to 1.

Figure 13-1. Access to ExCA Registers (Memory Access from Primary Side)

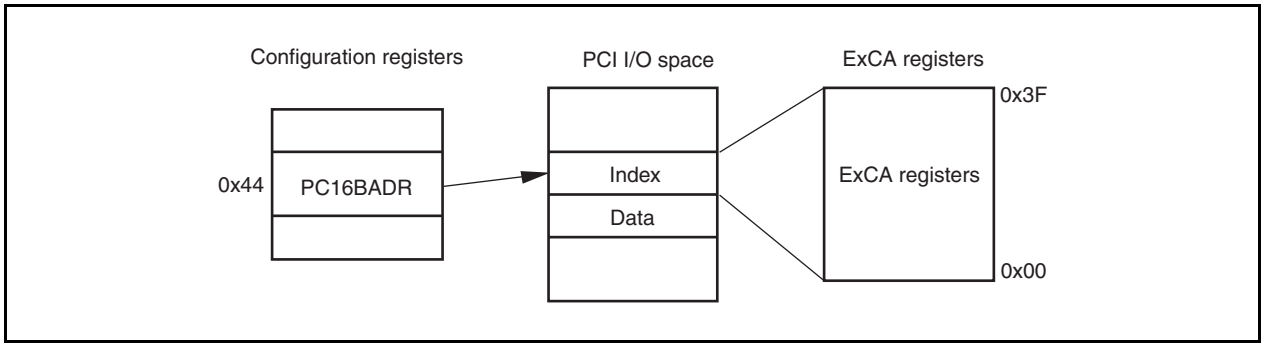


(2) Access according to an I/O access from the primary side (index method)

When all of the following conditions are satisfied, the ExCA registers can be accessed according to an I/O access from the primary side.

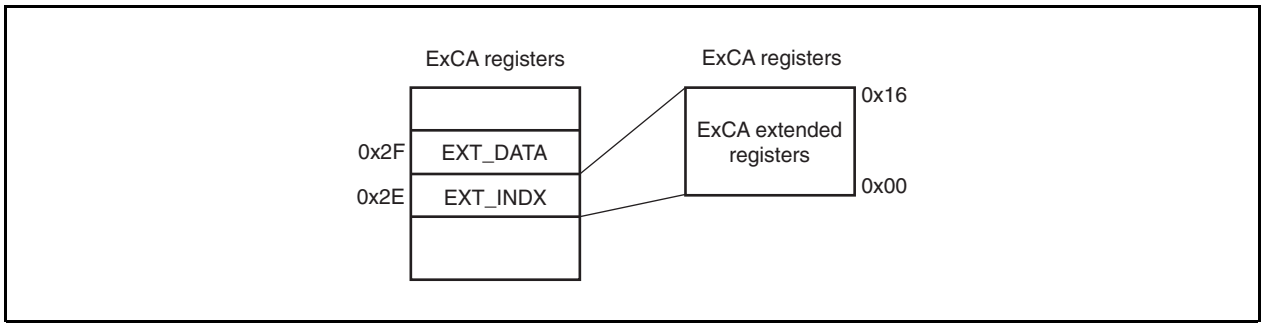
- The higher 30 bits of the address match the higher 30 bits of the PC16BADR register within the configuration registers.
- The IO_EN bit of the PCICMD register within the configuration registers is set to 1.

Figure 13-2. Access to ExCA Registers (I/O Access from Primary Side)



The ExCA registers also contain extended index and extended data registers.

Figure 13-3. ExCA Extended Registers



Tables 13-2 and 13-3 list the ExCA registers.

CARDU1 and CARDU2 each have the following ExCA registers.

Table 13-2. ExCA Registers (1/2)

Offset Address		R/W	Register Symbol	Function
PCI	ExCA			
0x800	0x00	R	ID_REV	ID/revision register
0x801	0x01	R	IF_STATUS	Interface status register
0x802	0x02	R/W	PWR_CNT	Power control register
0x803	0x03	R/W	INT_GEN_CNT	Interrupt/general-purpose control register
0x804	0x04	R/W	CARD_SC	Card status change register
0x805	0x05	R/W	CARD_SCI	Card status change interrupt configuration register
0x806	0x06	R/W	ADR_WIN_EN	Address window enable register
0x807	0x07	R/W	IO_WIN_CNT	I/O window control register
0x808	0x08	R/W	IO_WIN0_SAL	I/O window 0 start address lower byte register
0x809	0x09	R/W	IO_WIN0_SAH	I/O window 0 start address higher byte register
0x80A	0x0A	R/W	IO_WIN0_EAL	I/O window 0 end address lower byte register
0x80B	0x0B	R/W	IO_WIN0_EAH	I/O window 0 end address higher byte register
0x80C	0x0C	R/W	IO_WIN1_SAL	I/O window 1 start address lower byte register
0x80D	0x0D	R/W	IO_WIN1_SAH	I/O window 1 start address higher byte register
0x80E	0x0E	R/W	IO_WIN1_EAL	I/O window 1 end address lower byte register
0x80F	0x0F	R/W	IO_WIN1_EAH	I/O window 1 end address higher byte register
0x810	0x10	R/W	MEM_WIN0_SAL	Memory window 0 start address lower byte register
0x811	0x11	R/W	MEM_WIN0_SAH	Memory window 0 start address higher byte register
0x812	0x12	R/W	MEM_WIN0_EAL	Memory window 0 end address lower byte register
0x813	0x13	R/W	MEM_WIN0_EAH	Memory window 0 end address higher byte register
0x814	0x14	R/W	MEM_WIN0_OAL	Memory window 0 offset address lower byte register
0x815	0x15	R/W	MEM_WIN0_OAH	Memory window 0 offset address higher byte register
0x816	0x16	R/W	GEN_CNT	General control register
0x817	0x17	–	–	Reserved
0x818	0x18	R/W	MEM_WIN1_SAL	Memory window 1 start address lower byte register
0x819	0x19	R/W	MEM_WIN1_SAH	Memory window 1 start address higher byte register
0x81A	0x1A	R/W	MEM_WIN1_EAL	Memory window 1 end address lower byte register

Table 13-2. ExCA Registers (2/2)

Offset Address		R/W	Register Symbol	Function
PCI	ExCA			
0x81B	0x1B	R/W	MEM_WIN1_EAH	Memory window 1 end address higher byte register
0x81C	0x1C	R/W	MEM_WIN1_OAL	Memory window 1 offset address lower byte register
0x81D	0x1D	R/W	MEM_WIN1_OAH	Memory window 1 offset address higher byte register
0x81E	0x1E	R/W	GLO_CNT	Global control register
0x81F	0x1F	–	–	Reserved
0x820	0x20	R/W	MEM_WIN2_SAL	Memory window 2 start address lower byte register
0x821	0x21	R/W	MEM_WIN2_SAH	Memory window 2 start address higher byte register
0x822	0x22	R/W	MEM_WIN2_EAL	Memory window 2 end address lower byte register
0x823	0x23	R/W	MEM_WIN2_EAH	Memory window 2 end address higher byte register
0x824	0x24	R/W	MEM_WIN2_OAL	Memory window 2 offset address lower byte register
0x825	0x25	R/W	MEM_WIN2_OAH	Memory window 2 offset address higher byte register
0x826 to 0x827	0x26 to 0x27	–	–	Reserved
0x828	0x28	R/W	MEM_WIN3_SAL	Memory window 3 start address lower byte register
0x829	0x29	R/W	MEM_WIN3_SAH	Memory window 3 start address higher byte register
0x82A	0x2A	R/W	MEM_WIN3_EAL	Memory window 3 end address lower byte register
0x82B	0x2B	R/W	MEM_WIN3_EAH	Memory window 3 end address higher byte register
0x82C	0x2C	R/W	MEM_WIN3_OAL	Memory window 3 offset address lower byte register
0x82D	0x2D	R/W	MEM_WIN3_OAH	Memory window 3 offset address higher byte register
–	0x2E	R/W	EXT_INDX	Extended index register
–	0x2F	R/W	EXT_DATA	Extended data register
0x830	0x30	R/W	MEM_WIN4_SAL	Memory window 4 start address lower byte register
0x831	0x31	R/W	MEM_WIN4_SAH	Memory window 4 start address higher byte register
0x832	0x32	R/W	MEM_WIN4_EAL	Memory window 4 end address lower byte register
0x833	0x33	R/W	MEM_WIN4_EAH	Memory window 4 end address higher byte register
0x834	0x34	R/W	MEM_WIN4_OAL	Memory window 4 offset address lower byte register
0x835	0x35	R/W	MEM_WIN4_OAH	Memory window 4 offset address higher byte register
0x836	0x36	R/W	IO_WIN0_OAL	I/O window 0 offset address lower byte register
0x837	0x37	R/W	IO_WIN0_OAH	I/O window 0 offset address higher byte register
0x838	0x38	R/W	IO_WIN1_OAL	I/O window 1 offset address lower byte register
0x839	0x39	R/W	IO_WIN1_OAH	I/O window 1 offset address higher byte register
0x83A to 0x83F	0x3A to 0x3F	–	–	Reserved

Table 13-3. ExCA Extended Registers

Offset Address		R/W	Register Symbol	Function
PCI	ExCA Extension			
0x840	0x00	R/W	MEM_WIN0_SAU	Memory window 0 start address higher byte register
0x841	0x01	R/W	MEM_WIN1_SAU	Memory window 1 start address higher byte register
0x842	0x02	R/W	MEM_WIN2_SAU	Memory window 2 start address higher byte register
0x843	0x03	R/W	MEM_WIN3_SAU	Memory window 3 start address higher byte register
0x844	0x04	R/W	MEM_WIN4_SAU	Memory window 4 start address higher byte register
0x880	0x05	R/W	IO_SETUP_TIM	I/O setup timing register
0x881	0x06	R/W	IO_CMD_TIM	I/O command timing register
0x882	0x07	R/W	IO_HOLD_TIM	I/O hold timing register
0x883	0x08	–	–	Reserved
0x884	0x09	R/W	MEM0_SETUP_TIM	Memory setup timing 0 register
0x885	0x0A	R/W	MEM0_CMD_TIM	Memory command timing 0 register
0x886	0x0B	R/W	MEM0_HOLD_TIM	Memory hold timing 0 register
0x887	0x0C	–	–	Reserved
0x888	0x0D	R/W	MEM1_SETUP_TIM	Memory setup timing 1 register
0x889	0x0E	R/W	MEM1_CMD_TIM	Memory command timing 1 register
0x88A	0x0F	R/W	MEM1_HOLD_TIM	Memory hold timing 1 register
0x88B	0x10	–	–	Reserved
0x88C	0x11	R/W	MEM_TIM_SEL1	Memory timing selection 1 register
0x88D	0x12	R/W	MEM_TIM_SEL2	Memory timing selection 2 register
0x88E to 0x890	0x13 to 0x15	–	–	Reserved
0x891	0x16	R/W	MEM_WIN_PWEN	Memory window post write enable register

These registers are described in detail below.

13.3.1 ID_REV (PCI offset address: 0x800, ExCA offset address: 0x00)

Bit	7	6	5	4	3	2	1	0
Name	IF_TYPE1	IF_TYPE0	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	0	0	0	0	0

Bit	Name	Function
7:6	IF_TYPE(1:0)	Interface type 10: Supports 16-bit card
5:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

13.3.2 IF_STATUS (PCI offset address: 0x801, ExCA offset address: 0x01)

Bit	7	6	5	4	3	2	1	0
Name	RFU	CARD_ PWR	READY	CARD_WP	CARD_ DETECT2	CARD_ DETECT1	BV_ DETECT1	BV_ DETECT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Name	Function
7	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
6	CARD_PWR	Supply status of V_{CC} and V_{PP} to the card. 1: Supplied 0: Not supplied
5	READY	Status of slot 1 READY1 signal and slot 2 READY2 signal 1: High level 0: Low level
4	CARD_WP	PC card read/write attribute 1: Read only 0: Read/writable
3	CARD_DETECT2	Status of slot 1 CD12# signal and slot 2 CD22# signal 1: Low level 0: High level
2	CARD_DETECT1	Status of slot 1 CD11# signal and slot 2 CD21# signal 1: Low level 0: High level
1:0	BV_DETECT(1:0)	• For memory card Battery voltage status 11: Good 10: Falling 01: Cannot be supplied 00: Cannot be supplied • For I/O card BV_DETECT1: STSCHG# signal (corresponds to the BVD11# or BVD21# signal of the V_{RC}4173) status 1: Low level 0: High level BV_DETECT0: SPKR# signal (corresponds to the BVD12# or BVD22# signal of the V_{RC}4173) status 1: Low level 0: High level

13.3.3 PWR_CNT (PCI offset address: 0x802, ExCA offset address: 0x02)

Bit	7	6	5	4	3	2	1	0
Name	CARD_ OUT_EN	RFU	RFU	VCC1	VCC0	RFU	VPP1	VPP0
R/W	R/W	R	R	R/W	R/W	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	CARD_OUT_EN	Enables/disables output to a 16-bit card 1: Enable 0: Disable
6:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4:3	VCC(1:0)	Sets V_{CC} power supply level 11: 3.3 V 10: 5 V 01: Reserved 00: 0 V
2	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
1:0	VPP(1:0)	Sets V_{PP} power supply level 11: Reserved 10: 12 V 01: V_{CC} 00: 0 V

Caution When a read/write to the VCC(1:0) or VPP(1:0) area is performed, the VCC_CNT(2:0) or VPP_CNT(2:0) area of the SKT_CNT register within the CardBus socket registers is actually accessed.

13.3.4 INT_GEN_CNT (PCI offset address: 0x803, ExCA offset address: 0x03)

Bit	7	6	5	4	3	2	1	0
Name	RING_IND_EN	CARD_REST0	CARD_TYPE	RFU	FUC_INT_ROOT3	FUC_INT_ROOT2	FUC_INT_ROOT1	FUC_INT_ROOT0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	RING_IND_EN	Enables/disables Ring Indicate 1: Enable 0: Disable The CARDU unit does not support the Ring Indicate function. Set 0 for this bit.
6	CARD_REST0	16-bit PC card reset signal output status 1: Do not output reset 0: Output reset
5	CARD_TYPE	PC card type 1: I/O card 0: Memory card
4	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
3:0	FUC_INT_ROOT(3:0)	Function interrupt request routing 1111: IRQ15 1110: IRQ14 1101: IRQ13[Note] 1100: IRQ12 1011: IRQ11 1010: IRQ10 1001: IRQ9 1000: IRQ8[Note] 0111: IRQ7 0110: IRQ6[Note] 0101: IRQ5 0100: IRQ4 0011: IRQ3 0010: SMI[Note] 0001: IRQ1[Note] 0000: No routing Since the CARDU unit only supports parallel mode for PCI interrupts, the interrupt request routing setting is disabled. Set 0000 for these bits.

Note Only for Serialized interrupts

13.3.5 CARD_SC (PCI offset address: 0x804, ExCA offset address: 0x04)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	CARD_DT_CHG	RDY_CHG	BAT_WAR_CHG	BAT_DEAD_ST_CHG
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	CARD_DT_CHG	Detects change in the CDn1# signal and CDn2# signal (n = 1, 2) 1: Change occurred 0: No change occurred
2	RDY_CHG	• For memory card Detects change in the READY signal (corresponds to the READY1 or READY2 signal of the V_{RC}4173) from 0 to 1. 1: Change occurred 0: No change occurred • For I/O card Always fixed at 0.
1	BAT_WAR_CHG	• For memory card Detects change in the BVD2 signal (corresponds to the BVD12# or BVD22# signal of the V_{RC}4173) from 1 to 0. 1: Change occurred 0: No change occurred • For I/O card Always fixed at 0.
★ 0	BAT_DEAD_ST_CHG	• For memory card Detects change in the BVD1 signal (corresponds to the BVD11# or BVD21# signal of the V_{RC}4173) from 1 to 0. • For I/O card Detects change from 1 to 0 in the STSCHG# signal (corresponds to the BVD11# or BVD21# signal of the V_{RC}4173) or in the RI# signal (internal signal) when the RING_IND_EN bit is 1 (However, the CARDU unit does not support the Ring Indicate function). 1: Change occurred 0: No change occurred

Caution Each bit of this register can be cleared to 0 either by writing 1 to the corresponding bit or by reading the bit. The method that is used for clearing the bits is selected according to the INT_WB_CLR bit of the GLO_CNT register.

13.3.6 CARD_SCI (PCI offset address: 0x805, ExCA offset address: 0x05)

Bit	7	6	5	4	3	2	1	0
Name	CSC_INT_ ROOT3	CSC_INT_ ROOT2	CSC_INT_ ROOT1	CSC_INT_ ROOT0	CARD_DT_ EN	RDY_EN	BAT_WAR_ EN	BAT_DEAD_ EN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	CSC_INT_ROOT(3:0)	Status interrupt request routing 1111: IRQ15 1110: IRQ14 1101: IRQ13[Note] 1100: IRQ12 1011: IRQ11 1010: IRQ10 1001: IRQ9 1000: IRQ8[Note] 0111: IRQ7 0110: IRQ6[Note] 0101: IRQ5 0100: IRQ4 0011: IRQ3 0010: SMI[Note] 0001: IRQ1[Note] 0000: No routing Since the CARDU unit only supports parallel mode for PCI interrupts, the interrupt request routing setting is disabled. Set 0000 for these bits.
3	CARD_DT_EN	Enables/disables interrupt due to change in the CDn1# signal or CDn2# signal (n = 1, 2) 1: Enable 0: Disable
2	RDY_EN	Enables/disables interrupt due to change in the READYn signal from 0 to 1 (n = 1, 2) 1: Enable 0: Disable
1	BAT_WAR_EN	Enables/disables interrupt due to change in the BVDn2# signal from 1 to 0 (n = 1, 2) 1: Enable 0: Disable
0	BAT_DEAD_EN	Enables/disables interrupt due to change in the BVDn1# signal from 1 to 0 (n = 1, 2) 1: Enable 0: Disable

Note Only for Serialized interrupts

13.3.7 ADR_WIN_EN (PCI offset address: 0x806, ExCA offset address: 0x06)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_EN	IO_WIN0_EN	RFU	MEM_WIN4_EN	MEM_WIN3_EN	MEM_WIN2_EN	MEM_WIN1_EN	MEM_WIN0_EN
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	IO_WIN1_EN	Enables/disables I/O window 1 access. 1: Enable 0: Disable
6	IO_WIN0_EN	Enables/disables I/O window 0 access. 1: Enable 0: Disable
5	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
4	MEM_WIN4_EN	Enables/disables memory window 4 access. 1: Enable 0: Disable
3	MEM_WIN3_EN	Enables/disables memory window 3 access. 1: Enable 0: Disable
2	MEM_WIN2_EN	Enables/disables memory window 2 access. 1: Enable 0: Disable
1	MEM_WIN1_EN	Enables/disables memory window 1 access. 1: Enable 0: Disable
0	MEM_WIN0_EN	Enables/disables memory window 0 access. 1: Enable 0: Disable

13.3.8 IO_WIN_CNT (PCI offset address: 0x807, ExCA offset address: 0x07)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	IO_WIN1_ DATA_SEL	IO_WIN1_ DATA_SIZE	RFU	RFU	IO_WIN0_ DATA_SEL	IO_WIN0_ DATA_SIZE
R/W	R	R	R/W	R/W	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5	IO_WIN1_DATA_SEL	Determines the I/O window 1 data size 1: IOIS16# signal (corresponds to WP1 or WP2 signal of the V_{RC}4173) 0: IO_WIN1_DATA_SIZE bit
4	IO_WIN1_DATA_SIZE	Sets the I/O window 1 data size 1: 16 bits 0: 8 bits
3:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1	IO_WIN0_DATA_SEL	Determines the I/O window 0 data size 1: IOIS16# signal (corresponds to WP1 or WP2 signal of the V_{RC}4173) 0: IO_WIN0_DATA_SIZE bit
0	IO_WIN0_DATA_SIZE	Sets the I/O window 0 data size 1: 16 bits 0: 8 bits

13.3.9 IO_WIN0_SAL (PCI offset address: 0x808, ExCA offset address: 0x08)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_ SAL7	IO_WIN0_ SAL6	IO_WIN0_ SAL5	IO_WIN0_ SAL4	IO_WIN0_ SAL3	IO_WIN0_ SAL2	IO_WIN0_ SAL1	IO_WIN0_ SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_SAL(7:0)	Sets the I/O window 0 start address lower byte

13.3.10 IO_WIN0_SAH (PCI offset address: 0x809, ExCA offset address: 0x09)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_SAH7	IO_WIN0_SAH6	IO_WIN0_SAH5	IO_WIN0_SAH4	IO_WIN0_SAH3	IO_WIN0_SAH2	IO_WIN0_SAH1	IO_WIN0_SAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_SAH(7:0)	Sets the I/O window 0 start address higher byte

13.3.11 IO_WIN0_EAL (PCI offset address: 0x80A, ExCA offset address: 0x0A)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_EAL7	IO_WIN0_EAL6	IO_WIN0_EAL5	IO_WIN0_EAL4	IO_WIN0_EAL3	IO_WIN0_EAL2	IO_WIN0_EAL1	IO_WIN0_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_EAL(7:0)	Sets the I/O window 0 end address lower byte

13.3.12 IO_WIN0_EAH (PCI offset address: 0x80B, ExCA offset address: 0x0B)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_EAH7	IO_WIN0_EAH6	IO_WIN0_EAH5	IO_WIN0_EAH4	IO_WIN0_EAH3	IO_WIN0_EAH2	IO_WIN0_EAH1	IO_WIN0_EAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_EAH(7:0)	Sets the I/O window 0 end address higher byte

13.3.13 IO_WIN1_SAL (PCI offset address: 0x80C, ExCA offset address: 0x0C)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ SAL7	IO_WIN1_ SAL6	IO_WIN1_ SAL5	IO_WIN1_ SAL4	IO_WIN1_ SAL3	IO_WIN1_ SAL2	IO_WIN1_ SAL1	IO_WIN1_ SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_SAL(7:0)	Sets the I/O window 1 start address lower byte

13.3.14 IO_WIN1_SAH (PCI offset address: 0x80D, ExCA offset address: 0x0D)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ SAH7	IO_WIN1_ SAH6	IO_WIN1_ SAH5	IO_WIN1_ SAH4	IO_WIN1_ SAH3	IO_WIN1_ SAH2	IO_WIN1_ SAH1	IO_WIN1_ SAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_SAH(7:0)	Sets the I/O window 1 start address higher byte

13.3.15 IO_WIN1_EAL (PCI offset address: 0x80E, ExCA offset address: 0x0E)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ EAL7	IO_WIN1_ EAL6	IO_WIN1_ EAL5	IO_WIN1_ EAL4	IO_WIN1_ EAL3	IO_WIN1_ EAL2	IO_WIN1_ EAL1	IO_WIN1_ EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_EAL(7:0)	Sets the I/O window 1 end address lower byte

13.3.16 IO_WIN1_EAH (PCI offset address: 0x80F, ExCA offset address: 0x0F)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ EAH7	IO_WIN1_ EAH6	IO_WIN1_ EAH5	IO_WIN1_ EAH4	IO_WIN1_ EAH3	IO_WIN1_ EAH2	IO_WIN1_ EAH1	IO_WIN1_ EAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_EAH(7:0)	Sets the I/O window 1 end address higher byte

13.3.17 MEM_WIN0_SAL (PCI offset address: 0x810, ExCA offset address: 0x10)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN0_SAL7	MEM_ WIN0_SAL6	MEM_ WIN0_SAL5	MEM_ WIN0_SAL4	MEM_ WIN0_SAL3	MEM_ WIN0_SAL2	MEM_ WIN0_SAL1	MEM_ WIN0_SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN0_SAL(7:0)	Sets the memory window 0 start address lower byte

13.3.18 MEM_WIN0_SAH (PCI offset address: 0x811, ExCA offset address: 0x11)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN0 _DSIZE	RFU	RFU	RFU	MEM_ WIN0_SAH3	MEM_ WIN0_SAH2	MEM_ WIN0_SAH1	MEM_ WIN0_SAH0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN0_DSIZE	Sets the memory window 0 data size 1: 16 bits 0: 8 bits
6:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN0_SAH(3:0)	Sets the memory window 0 start address higher bits (A(23:20))

13.3.19 MEM_WIN0_EAL (PCI offset address: 0x812, ExCA offset address: 0x12)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN0_EAL7	MEM_ WIN0_EAL6	MEM_ WIN0_EAL5	MEM_ WIN0_EAL4	MEM_ WIN0_EAL3	MEM_ WIN0_EAL2	MEM_ WIN0_EAL1	MEM_ WIN0_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN0_EAL(7:0)	Sets the memory window 0 end address lower byte

13.3.20 MEM_WIN0_EAH (PCI offset address: 0x813, ExCA offset address: 0x13)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MEM_ WIN0_EAH3	MEM_ WIN0_EAH2	MEM_ WIN0_EAH1	MEM_ WIN0_EAH0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN0_EAH(3:0)	Sets the memory window 0 end address higher bits (A(23:20))

13.3.21 MEM_WIN0_OAL (PCI offset address: 0x814, ExCA offset address: 0x14)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN0_OAL7	MEM_ WIN0_OAL6	MEM_ WIN0_OAL5	MEM_ WIN0_OAL4	MEM_ WIN0_OAL3	MEM_ WIN0_OAL2	MEM_ WIN0_OAL1	MEM_ WIN0_OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN0_OAL(7:0)	Sets the memory window 0 offset address lower byte

13.3.22 MEM_WIN0_OAH (PCI offset address: 0x815, ExCA offset address: 0x15)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN0_WP	MEM_WIN0 _REGSET	MEM_WIN0 _OAH5	MEM_WIN0 _OAH4	MEM_WIN0 _OAH3	MEM_WIN0 _OAH2	MEM_WIN0 _OAH1	MEM_WIN0 _OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN0_WP	Enables/disables the memory window 0 write protect setting 1: Enable 0: Disable
6	MEM_WIN0_REGSET	Sets the memory window 0 mapping destination 1: Attribute memory 0: Common memory
5:0	MEM_WIN0_OAH(5:0)	Sets the memory window 0 offset address higher bits (A(25:20))

13.3.23 GEN_CNT (PCI offset address: 0x816, ExCA offset address: 0x16)

Bit	7	6	5	4	3	2	1	0
Name	VS2_ STATUS	VS1_ STATUS	RFU	RFU	RFU	RFU	ExCA_REG _RST_EN	RFU
R/W	R	R	R	R	R	R	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	VS2_STATUS	16-bit PC card interface VSn2# signal status (n = 1, 2) 1: High level 0: Low level
6	VS1_STATUS	16-bit PC card interface VSn1# signal status (n = 1, 2) 1: High level 0: Low level
5:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1	ExCA_REG_RST_EN	Enables/disables reset to the ExCA registers when the card is removed 1: Enable 0: Disable
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

13.3.24 MEM_WIN1_SAL (PCI offset address: 0x818, ExCA offset address: 0x18)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN1_SAL7	MEM_ WIN1_SAL6	MEM_ WIN1_SAL5	MEM_ WIN1_SAL4	MEM_ WIN1_SAL3	MEM_ WIN1_SAL2	MEM_ WIN1_SAL1	MEM_ WIN1_SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN1_SAL(7:0)	Sets the memory window 1 start address lower byte

13.3.25 MEM_WIN1_SAH (PCI offset address: 0x819, ExCA offset address: 0x19)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN1 _DSIZE	RFU	RFU	RFU	MEM_ WIN1_SAH3	MEM_ WIN1_SAH2	MEM_ WIN1_SAH1	MEM_ WIN1_SAH0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN1_DSIZE	Sets the memory window 1 data size 1: 16 bits 0: 8 bits
6:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN1_SAH(3:0)	Sets the memory window 1 start address higher bits (A(23:20))

13.3.26 MEM_WIN1_EAL (PCI offset address: 0x81A, ExCA offset address: 0x1A)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN1_EAL7	MEM_ WIN1_EAL6	MEM_ WIN1_EAL5	MEM_ WIN1_EAL4	MEM_ WIN1_EAL3	MEM_ WIN1_EAL2	MEM_ WIN1_EAL1	MEM_ WIN1_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN1_EAL(7:0)	Sets the memory window 1 end address lower byte

13.3.27 MEM_WIN1_EAH (PCI offset address: 0x81B, ExCA offset address: 0x1B)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MEM_ WIN1_EAH3	MEM_ WIN1_EAH2	MEM_ WIN1_EAH1	MEM_ WIN1_EAH0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN1_EAH(3:0)	Sets the memory window 1 end address higher bits (A(23:20))

13.3.28 MEM_WIN1_OAL (PCI offset address: 0x81C, ExCA offset address: 0x1C)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN1_OAL7	MEM_ WIN1_OAL6	MEM_ WIN1_OAL5	MEM_ WIN1_OAL4	MEM_ WIN1_OAL3	MEM_ WIN1_OAL2	MEM_ WIN1_OAL1	MEM_ WIN1_OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN1_OAL(7:0)	Sets the memory window 1 offset address lower byte

13.3.29 MEM_WIN1_OAH (PCI offset address: 0x81D, ExCA offset address: 0x1D)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN1_WP	MEM_WIN1 _REGSET	MEM_WIN1 _OAH5	MEM_WIN1 _OAH4	MEM_WIN1 _OAH3	MEM_WIN1 _OAH2	MEM_WIN1 _OAH1	MEM_WIN1 _OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN1_WP	Enables/disables the memory window 1 write protect setting 1: Enable 0: Disable
6	MEM_WIN1_REGSET	Sets the memory window 1 mapping destination 1: Attribute memory 0: Common memory
5:0	MEM_WIN1_OAH(5:0)	Sets the memory window 1 offset address higher bits (A(25:20))

13.3.30 GLO_CNT (PCI offset address: 0x81E, ExCA offset address: 0x1E)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	FUN_INT_LEV	INT_WB_CLR	CSC_INT_LEV	RFU
R/W	R	R	R	R	R/W	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	FUN_INT_LEV	Sets the trigger for function interrupt requests to the host 1: Level mode 0: Edge mode Since the CARDU unit only supports parallel mode for PCI interrupts, this bit setting is disabled. Set 0 for this bit.
2	INT_WB_CLR	Selects the method of clearing CSC interrupt flags 1: Cleared to 0 by writing 1 to the corresponding bit of the CARD_SC register 0: Cleared to 0 by reading the CARD_SC register
1	CSC_INT_LEV	Sets the trigger for CSC interrupt requests to the host 1: Level mode 0: Edge mode Since the CARDU unit only supports parallel mode for PCI interrupts, this bit setting is disabled. Set 0 for this bit.
0	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

13.3.31 MEM_WIN2_SAL (PCI offset address: 0x820, ExCA offset address: 0x20)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN2_SAL7	MEM_WIN2_SAL6	MEM_WIN2_SAL5	MEM_WIN2_SAL4	MEM_WIN2_SAL3	MEM_WIN2_SAL2	MEM_WIN2_SAL1	MEM_WIN2_SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN2_SAL(7:0)	Sets the memory window 2 start address lower byte

13.3.32 MEM_WIN2_SAH (PCI offset address: 0x821, ExCA offset address: 0x21)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN2_DSIZE	RFU	RFU	RFU	MEM_WIN2_SAH3	MEM_WIN2_SAH2	MEM_WIN2_SAH1	MEM_WIN2_SAH0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN2_DSIZE	Sets the memory window 2 data size 1: 16 bits 0: 8 bits
6:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN2_SAH(3:0)	Sets the memory window 2 start address higher bits (A(23:20))

13.3.33 MEM_WIN2_EAL (PCI offset address: 0x822, ExCA offset address: 0x22)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN2_EAL7	MEM_WIN2_EAL6	MEM_WIN2_EAL5	MEM_WIN2_EAL4	MEM_WIN2_EAL3	MEM_WIN2_EAL2	MEM_WIN2_EAL1	MEM_WIN2_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN2_EAL(7:0)	Sets the memory window 2 end address lower byte

13.3.34 MEM_WIN2_EAH (PCI offset address: 0x823, ExCA offset address: 0x23)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MEM_WIN2_EAH3	MEM_WIN2_EAH2	MEM_WIN2_EAH1	MEM_WIN2_EAH0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN2_EAH(3:0)	Sets the memory window 2 end address higher bits (A(23:20))

13.3.35 MEM_WIN2_OAL (PCI offset address: 0x824, ExCA offset address: 0x24)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN2_OAL7	MEM_ WIN2_OAL6	MEM_ WIN2_OAL5	MEM_ WIN2_OAL4	MEM_ WIN2_OAL3	MEM_ WIN2_OAL2	MEM_ WIN2_OAL1	MEM_ WIN2_OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN2_OAL(7:0)	Sets the memory window 2 offset address lower byte

13.3.36 MEM_WIN2_OAH (PCI offset address: 0x825, ExCA offset address: 0x25)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN2_WP	MEM_WIN2 _REGSET	MEM_WIN2 _OAH5	MEM_WIN2 _OAH4	MEM_WIN2 _OAH3	MEM_WIN2 _OAH2	MEM_WIN2 _OAH1	MEM_WIN2 _OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN2_WP	Enables/disables the memory window 2 write protect setting 1: Enable 0: Disable
6	MEM_WIN2_REGSET	Sets the memory window 2 mapping destination 1: Attribute memory 0: Common memory
5:0	MEM_WIN2_OAH(5:0)	Sets the memory window 2 offset address higher bits (A(25:20))

13.3.37 MEM_WIN3_SAL (PCI offset address: 0x828, ExCA offset address: 0x28)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN3_SAL7	MEM_ WIN3_SAL6	MEM_ WIN3_SAL5	MEM_ WIN3_SAL4	MEM_ WIN3_SAL3	MEM_ WIN3_SAL2	MEM_ WIN3_SAL1	MEM_ WIN3_SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN3_SAL(7:0)	Sets the memory window 3 start address lower byte

13.3.38 MEM_WIN3_SAH (PCI offset address: 0x829, ExCA offset address: 0x29)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN3_DSIZE	RFU	RFU	RFU	MEM_WIN3_SAH3	MEM_WIN3_SAH2	MEM_WIN3_SAH1	MEM_WIN3_SAH0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN3_DSIZE	Sets the memory window 3 data size 1: 16 bits 0: 8 bits
6:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN3_SAH(3:0)	Sets the memory window 3 start address higher bits (A(23:20))

13.3.39 MEM_WIN3_EAL (PCI offset address: 0x82A, ExCA offset address: 0x2A)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN3_EAL7	MEM_WIN3_EAL6	MEM_WIN3_EAL5	MEM_WIN3_EAL4	MEM_WIN3_EAL3	MEM_WIN3_EAL2	MEM_WIN3_EAL1	MEM_WIN3_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN3_EAL(7:0)	Sets the memory window 3 end address lower byte

13.3.40 MEM_WIN3_EAH (PCI offset address: 0x82B, ExCA offset address: 0x2B)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MEM_WIN3_EAH3	MEM_WIN3_EAH2	MEM_WIN3_EAH1	MEM_WIN3_EAH0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN3_EAH(3:0)	Sets the memory window 3 end address higher bits (A(23:20))

13.3.41 MEM_WIN3_OAL (PCI offset address: 0x82C, ExCA offset address: 0x2C)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN3_OAL7	MEM_WIN3_OAL6	MEM_WIN3_OAL5	MEM_WIN3_OAL4	MEM_WIN3_OAL3	MEM_WIN3_OAL2	MEM_WIN3_OAL1	MEM_WIN3_OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN3_OAL(7:0)	Sets the memory window 3 offset address lower byte

13.3.42 MEM_WIN3_OAH (PCI offset address: 0x82D, ExCA offset address: 0x2D)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN3_WP	MEM_WIN3_REGSET	MEM_WIN3_OAH5	MEM_WIN3_OAH4	MEM_WIN3_OAH3	MEM_WIN3_OAH2	MEM_WIN3_OAH1	MEM_WIN3_OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN3_WP	Enables/disables the memory window 3 write protect setting 1: Enable 0: Disable
6	MEM_WIN3_REGSET	Sets the memory window 3 mapping destination 1: Attribute memory 0: Common memory
5:0	MEM_WIN3_OAH(5:0)	Sets the memory window 3 offset address higher bits (A(25:20))

13.3.43 EXT_INDX (ExCA offset address: 0x2E)

Bit	7	6	5	4	3	2	1	0
Name	EXT_INDX7	EXT_INDX6	EXT_INDX5	EXT_INDX4	EXT_INDX3	EXT_INDX2	EXT_INDX1	EXT_INDX0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	EXT_INDX(7:0)	Sets the extended index

Caution Read/write according to a memory access from the primary side is not supported.

13.3.44 EXT_DATA (ExCA offset address: 0x2F)

Bit	7	6	5	4	3	2	1	0
Name	EXT_DATA7	EXT_DATA6	EXT_DATA5	EXT_DATA4	EXT_DATA3	EXT_DATA2	EXT_DATA1	EXT_DATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	EXT_DATA(7:0)	Sets the extended data

Caution Read/write according to a memory access from the primary side is not supported.

13.3.45 MEM_WIN4_SAL (PCI offset address: 0x830, ExCA offset address: 0x30)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN4 _SAL7	MEM_WIN4 _SAL6	MEM_WIN4 _SAL5	MEM_WIN4 _SAL4	MEM_WIN4 _SAL3	MEM_WIN4 _SAL2	MEM_WIN4 _SAL1	MEM_WIN4 _SAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN4_SAL(7:0)	Sets the memory window 4 start address lower byte

13.3.46 MEM_WIN4_SAH (PCI offset address: 0x831, ExCA offset address: 0x31)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN4 _DSIZE	RFU	RFU	RFU	MEM_WIN4 _SAH3	MEM_WIN4 _SAH2	MEM_WIN4 _SAH1	MEM_WIN4 _SAH0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN4_DSIZE	Sets the memory window 4 data size 1: 16 bits 0: 8 bits
6:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN4_SAH(3:0)	Sets the memory window 4 start address higher bits (A(23:20))

13.3.47 MEM_WIN4_EAL (PCI offset address: 0x832, ExCA offset address: 0x32)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN4_EAL7	MEM_ WIN4_EAL6	MEM_ WIN4_EAL5	MEM_ WIN4_EAL4	MEM_ WIN4_EAL3	MEM_ WIN4_EAL2	MEM_ WIN4_EAL1	MEM_ WIN4_EAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN4_EAL(7:0)	Sets the memory window 4 end address lower byte

13.3.48 MEM_WIN4_EAH (PCI offset address: 0x833, ExCA offset address: 0x33)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	MEM_ WIN4_EAH3	MEM_ WIN4_EAH2	MEM_ WIN4_EAH1	MEM_ WIN4_EAH0
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3:0	MEM_WIN4_EAH(3:0)	Sets the memory window 4 end address higher bits (A(23:20))

13.3.49 MEM_WIN4_OAL (PCI offset address: 0x834, ExCA offset address: 0x34)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN4_OAL7	MEM_ WIN4_OAL6	MEM_ WIN4_OAL5	MEM_ WIN4_OAL4	MEM_ WIN4_OAL3	MEM_ WIN4_OAL2	MEM_ WIN4_OAL1	MEM_ WIN4_OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN4_OAL(7:0)	Sets the memory window 4 offset address lower byte

13.3.50 MEM_WIN4_OAH (PCI offset address: 0x835, ExCA offset address: 0x35)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN4_WP	MEM_WIN4 _REGSET	MEM_WIN4 _OAH5	MEM_WIN4 _OAH4	MEM_WIN4 _OAH3	MEM_WIN4 _OAH2	MEM_WIN4 _OAH1	MEM_WIN4 _OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7	MEM_WIN4_WP	Enables/disables the memory window 4 write protect setting 1: Enable 0: Disable
6	MEM_WIN4_REGSET	Sets the memory window 4 mapping destination 1: Attribute memory 0: Common memory
5:0	MEM_WIN4_OAH(5:0)	Sets the memory window 4 offset address higher bits (A(25:20))

13.3.51 IO_WIN0_OAL (PCI offset address: 0x836, ExCA offset address: 0x36)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_ OAL7	IO_WIN0_ OAL6	IO_WIN0_ OAL5	IO_WIN0_ OAL4	IO_WIN0_ OAL3	IO_WIN0_ OAL2	IO_WIN0_ OAL1	IO_WIN0_ OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_OAL(7:0)	Sets the I/O window 0 offset address lower bytes (A(7:0))

13.3.52 IO_WIN0_OAH (PCI offset address: 0x837, ExCA offset address: 0x37)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN0_ OAH7	IO_WIN0_ OAH6	IO_WIN0_ OAH5	IO_WIN0_ OAH4	IO_WIN0_ OAH3	IO_WIN0_ OAH2	IO_WIN0_ OAH1	IO_WIN0_ OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN0_OAH(7:0)	Sets the I/O window 0 offset address higher bytes (A(15:8))

13.3.53 IO_WIN1_OAL (PCI offset address: 0x838, ExCA offset address: 0x38)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ OAL7	IO_WIN1_ OAL6	IO_WIN1_ OAL5	IO_WIN1_ OAL4	IO_WIN1_ OAL3	IO_WIN1_ OAL2	IO_WIN1_ OAL1	IO_WIN1_ OAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_OAL(7:0)	Sets the I/O window 1 offset address lower bytes (A(7:0))

13.3.54 IO_WIN1_OAH (PCI offset address: 0x839, ExCA offset address: 0x39)

Bit	7	6	5	4	3	2	1	0
Name	IO_WIN1_ OAH7	IO_WIN1_ OAH6	IO_WIN1_ OAH5	IO_WIN1_ OAH4	IO_WIN1_ OAH3	IO_WIN1_ OAH2	IO_WIN1_ OAH1	IO_WIN1_ OAH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	IO_WIN1_OAH(7:0)	Sets the I/O window 1 offset address higher bytes (A(15:8))

13.3.55 MEM_WIN0_SAU (PCI offset address: 0x840, ExCA extended offset address: 0x00)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN0_SAU7	MEM_ WIN0_SAU6	MEM_ WIN0_SAU5	MEM_ WIN0_SAU4	MEM_ WIN0_SAU3	MEM_ WIN0_SAU2	MEM_ WIN0_SAU1	MEM_ WIN0_SAU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN0_SAU(7:0)	Sets the memory window 0 start address higher bytes (A(31:24))

13.3.56 MEM_WIN1_SAU (PCI offset address: 0x841, ExCA extended offset address: 0x01)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN1_SAU7	MEM_ WIN1_SAU6	MEM_ WIN1_SAU5	MEM_ WIN1_SAU4	MEM_ WIN1_SAU3	MEM_ WIN1_SAU2	MEM_ WIN1_SAU1	MEM_ WIN1_SAU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN1_SAU(7:0)	Sets the memory window 1 start address higher bytes (A(31:24))

13.3.57 MEM_WIN2_SAU (PCI offset address: 0x842, ExCA extended offset address: 0x02)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN2_SAU7	MEM_ WIN2_SAU6	MEM_ WIN2_SAU5	MEM_ WIN2_SAU4	MEM_ WIN2_SAU3	MEM_ WIN2_SAU2	MEM_ WIN2_SAU1	MEM_ WIN2_SAU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN2_SAU(7:0)	Sets the memory window 2 start address higher bytes (A(31:24))

13.3.58 MEM_WIN3_SAU (PCI offset address: 0x843, ExCA extended offset address: 0x03)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN3_SAU7	MEM_ WIN3_SAU6	MEM_ WIN3_SAU5	MEM_ WIN3_SAU4	MEM_ WIN3_SAU3	MEM_ WIN3_SAU2	MEM_ WIN3_SAU1	MEM_ WIN3_SAU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN3_SAU(7:0)	Sets the memory window 3 start address higher bytes (A(31:24))

13.3.59 MEM_WIN4_SAU (PCI offset address: 0x844, ExCA extended offset address: 0x04)

Bit	7	6	5	4	3	2	1	0
Name	MEM_ WIN4_SAU7	MEM_ WIN4_SAU6	MEM_ WIN4_SAU5	MEM_ WIN4_SAU4	MEM_ WIN4_SAU3	MEM_ WIN4_SAU2	MEM_ WIN4_SAU1	MEM_ WIN4_SAU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	MEM_WIN4_SAU(7:0)	Sets the memory window 4 start address higher bytes (A(31:24))

13.3.60 IO_SETUP_TIM (PCI offset address: 0x880, ExCA extended offset address: 0x05)

Bit	7	6	5	4	3	2	1	0
Name	IO_SETUP_ TIM7	IO_SETUP_ TIM6	IO_SETUP_ TIM5	IO_SETUP_ TIM4	IO_SETUP_ TIM3	IO_SETUP_ TIM2	IO_SETUP_ TIM1	IO_SETUP_ TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	1	0

Bit	Name	Function
7:0	IO_SETUP_TIM(7:0)	Sets the I/O window setup timing 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x07: 8 clock pulses (240 ns) 0x06: 7 clock pulses (210 ns) 0x05 to 0x00: Setting prohibited

★ **Caution Set values that match the device specifications.**

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.61 IO_CMD_TIM (PCI offset address: 0x881, ExCA extended offset address: 0x06)

Bit	7	6	5	4	3	2	1	0
Name	IO_CMD_TIM7	IO_CMD_TIM6	IO_CMD_TIM5	IO_CMD_TIM4	IO_CMD_TIM3	IO_CMD_TIM2	IO_CMD_TIM1	IO_CMD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	0	1

Bit	Name	Function
7:0	IO_CMD_TIM(7:0)	Sets the I/O window command timing 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x04: 5 clock pulses (150 ns) 0x03: 4 clock pulses (120 ns) 0x02 to 0x00: Setting prohibited

★ **Caution Set values that match the device specifications.**

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.62 IO_HOLD_TIM (PCI offset address: 0x882, ExCA extended offset address: 0x07)

Bit	7	6	5	4	3	2	1	0
Name	IO_HOLD_TIM7	IO_HOLD_TIM6	IO_HOLD_TIM5	IO_HOLD_TIM4	IO_HOLD_TIM3	IO_HOLD_TIM2	IO_HOLD_TIM1	IO_HOLD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	1	0

Bit	Name	Function
7:0	IO_HOLD_TIM(7:0)	Sets the I/O window hold timing 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x01: 2 clock pulses (60 ns) 0x00: 1 clock pulse (30 ns)

Caution Set values that match the device specifications.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.63 MEM0_SETUP_TIM (PCI offset address: 0x884, ExCA extended offset address: 0x09)

Bit	7	6	5	4	3	2	1	0
Name	MEM0_SETUP_TIM7	MEM0_SETUP_TIM6	MEM0_SETUP_TIM5	MEM0_SETUP_TIM4	MEM0_SETUP_TIM3	MEM0_SETUP_TIM2	MEM0_SETUP_TIM1	MEM0_SETUP_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	1	0

Bit	Name	Function
7:0	MEM0_SETUP_TIM(7:0)	Sets the memory window setup timing 0 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x01: 2 clock pulses (60 ns) 0x00: 1 clock pulse (30 ns)

Caution Set values that match the device specifications.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.64 MEM0_CMD_TIM (PCI offset address: 0x885, ExCA extended offset address: 0x0A)

Bit	7	6	5	4	3	2	1	0
Name	MEM0_CMD_TIM7	MEM0_CMD_TIM6	MEM0_CMD_TIM5	MEM0_CMD_TIM4	MEM0_CMD_TIM3	MEM0_CMD_TIM2	MEM0_CMD_TIM1	MEM0_CMD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	1	0	0	0	0

Bit	Name	Function
7:0	MEM0_CMD_TIM(7:0)	Sets the memory window command timing 0 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x04: 5 clock pulses (150 ns) 0x03: 4 clock pulses (120 ns) 0x02 to 0x00: Setting prohibited

★ **Caution Set values that match the device specifications.**

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.65 MEM0_HOLD_TIM (PCI offset address: 0x886, ExCA extended offset address: 0x0B)

Bit	7	6	5	4	3	2	1	0
Name	MEM0_ HOLD_TIM7	MEM0_ HOLD_TIM6	MEM0_ HOLD_TIM5	MEM0_ HOLD_TIM4	MEM0_ HOLD_TIM3	MEM0_ HOLD_TIM2	MEM0_ HOLD_TIM1	MEM0_ HOLD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	1	0

Bit	Name	Function
7:0	MEM0_HOLD_TIM(7:0)	Sets the memory window hold timing 0 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x01: 2 clock pulses (60 ns) 0x00: 1 clock pulse (30 ns)

Caution Set values that match the device specifications.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.66 MEM1_SETUP_TIM (PCI offset address: 0x888, ExCA extended offset address: 0x0D)

Bit	7	6	5	4	3	2	1	0
Name	MEM1_SET UP_TIM7	MEM1_SET UP_TIM6	MEM1_SET UP_TIM5	MEM1_SET UP_TIM4	MEM1_SET UP_TIM3	MEM1_SET UP_TIM2	MEM1_SET UP_TIM1	MEM1_SET UP_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	1	0

Bit	Name	Function
7:0	MEM1_SETUP_TIM(7:0)	Sets the memory window setup timing 1 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x01: 2 clock pulses (60 ns) 0x00: 1 clock pulse (30 ns)

Caution Set values that match the device specifications.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.67 MEM1_CMD_TIM (PCI offset address: 0x889, ExCA extended offset address: 0x0E)

Bit	7	6	5	4	3	2	1	0
Name	MEM1_ CMD_TIM7	MEM1_ CMD_TIM6	MEM1_ CMD_TIM5	MEM1_ CMD_TIM4	MEM1_ CMD_TIM3	MEM1_ CMD_TIM2	MEM1_ CMD_TIM1	MEM1_ CMD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	1	1	0

Bit	Name	Function
7:0	MEM1_CMD_TIM(7:0)	Sets the memory window command timing 1 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x04: 5 clock pulses (150 ns) 0x03: 4 clock pulses (120 ns) 0x02 to 0x00: Setting prohibited

★ **Caution Set values that match the device specifications.**

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.68 MEM1_HOLD_TIM (PCI offset address: 0x88A, ExCA extended offset address: 0x0F)

Bit	7	6	5	4	3	2	1	0
Name	MEM1_ HOLD_TIM7	MEM1_ HOLD_TIM6	MEM1_ HOLD_TIM5	MEM1_ HOLD_TIM4	MEM1_ HOLD_TIM3	MEM1_ HOLD_TIM2	MEM1_ HOLD_TIM1	MEM1_ HOLD_TIM0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	MEM1_HOLD_TIM(7:0)	Sets the memory window hold timing 1 0xFF: 256 clock pulses (7680 ns) 0xFE: 255 clock pulses (7650 ns) : 0x01: 2 clock pulses (60 ns) 0x00: 1 clock pulse (30 ns)

Caution Set values that match the device specifications.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

13.3.69 MEM_TIM_SEL1 (PCI offset address: 0x88C, ExCA extended offset address: 0x11)

Bit	7	6	5	4	3	2	1	0
Name	MEM_WIN3 _TIMSEL1	MEM_WIN3 _TIMSEL0	MEM_WIN2 _TIMSEL1	MEM_WIN2 _TIMSEL0	MEM_WIN1 _TIMSEL1	MEM_WIN1 _TIMSEL0	MEM_WIN0 _TIMSEL1	MEM_WIN0 _TIMSEL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:6	MEM_WIN3_TIMSEL(1:0)	Selects the memory window 3 timing 00: Timing 0 Other: Timing 1
5:4	MEM_WIN2_TIMSEL(1:0)	Selects the memory window 2 timing 00: Timing 0 Other: Timing 1
3:2	MEM_WIN1_TIMSEL(1:0)	Selects the memory window 1 timing 00: Timing 0 Other: Timing 1
1:0	MEM_WIN0_TIMSEL(1:0)	Selects the memory window 0 timing 00: Timing 0 Other: Timing 1

13.3.70 MEM_TIM_SEL2 (PCI offset address: 0x88D, ExCA extended offset address: 0x12)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	MEM_WIN4 _TIMSEL1	MEM_WIN4 _TIMSEL0
R/W	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:2	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
1:0	MEM_WIN4_TIMSEL(1:0)	Selects the memory window 4 timing 00: Timing 0 Other: Timing 1

13.3.71 MEM_WIN_PWEN (PCI offset address: 0x891, ExCA extended offset address: 0x16)

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	POSTWEN
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:1	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
0	POSTWEN	Enables/disables the memory window post write cycle 1: Enable 0: Disable

Caution The setting of POSTWEN bit is common for all windows.

13.4 CardBus Socket Register Set

When all of the following conditions are satisfied, the CardBus socket registers can be accessed according to a memory access from the primary side.

- The higher 20 bits of the address match the higher 20 bits of the CSRBADR register.
- The lower 12 bits of the address are in the range 0x000 to 0x7FF.
- The MEM_EN bit of the PCICMD register is set to 1.

Figure 13-4. CardBus Socket Registers

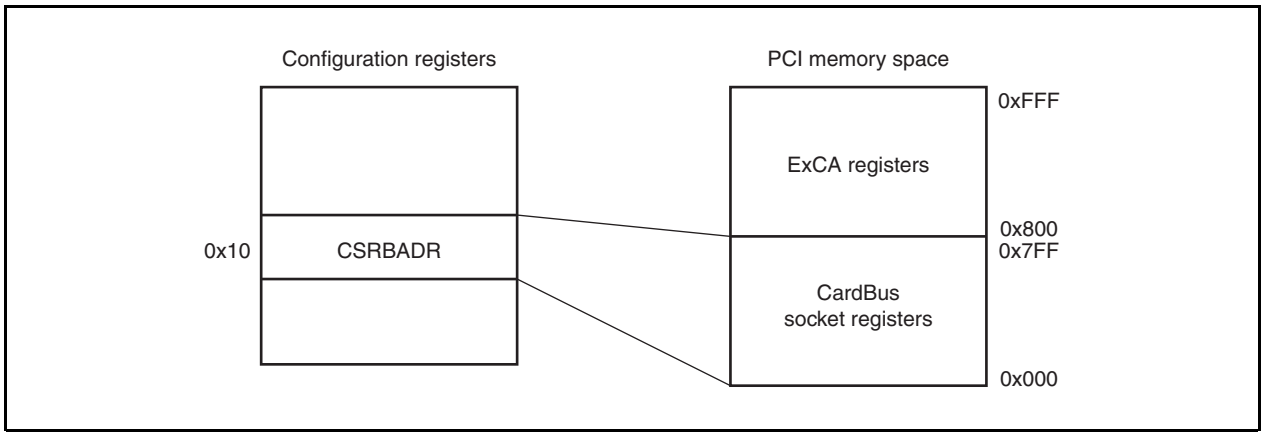


Table 13-4 lists the CardBus socket registers.

CARDU1 and CARDU2 each have the following registers.

Table 13-4. CardBus Socket Registers

Offset Address	R/W	Register Symbol	Function
0x000	R/W	SKT_EV	Socket event register
0x004	R/W	SKT_MASK	Socket mask register
★ 0x008	R	SKT_PRE_STATE	Socket present state register
0x00C	R/W	SKT_FORCE_EV	Socket force event register
0x010	R/W	SKT_CNT	Socket control register
0x014 to 0x7FF	–	–	Reserved

13.4.1 SKT_EV (offset address: 0x000)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	POW_ CYC_EV	CCD2_EV	CCD1_EV	CSTSCHG_ EV
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	POW_CYC_EV	Whether or not a change to 1 is to be detected in the POW_UP bit of the SKT_PRE_STATE register. Cleared to 0 when 1 is written. 1: Detected 0: Not detected
★ 2	CCD2_EV	Whether or not a change is to be detected in the CD12# or CD22# signal. Cleared to 0 when 1 is written. 1: Detected 0: Not detected
★ 1	CCD1_EV	Whether or not a change is to be detected in the CD11# or CD21# signal. Cleared to 0 when 1 is written. 1: Detected 0: Not detected

(2/2)

Bit	Name	Function
★ 0	CSTSCHG_EV	• For a 16-bit memory card Whether or not a change to 0 is to be detected in the BVD1 signal (corresponds to the BVD11# or BVD21# signal of the VRC4173) or BVD2 signal (corresponds to the BVD12# or BVD22# signal of the VRC4173) and a change to 1 is to be detected in the READY signal. • For a 16-bit I/O card Whether or not a change to 0 is to be detected in the STSCHG# signal (corresponds to the BVD11# or BVD21# signal of the VRC4173) and a change to 0 is to be detected in the RI# signal (internal signal) (However, the CARDU unit does not support the Ring Indicate function). 1: Detected 0: Not detected This bit can be cleared to 0 when 1 is written.

13.4.2 SKT_MASK (offset address: 0x004)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	POW_ CYC_MSK	CCD_MSK1	CCD_MSK0	CSC_MSK
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:4	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
3	POW_CYC_MSK	Controls interrupt requests according to the POW_CYC_EN bit of the SKT_EV register 1: Do not mask 0: Mask
2:1	CCD_MSK(1:0)	Controls interrupt requests according to the CCD1_EV and CCD2_EV bits of the SKT_EV register 11: Do not mask 00: Mask The settings 01 and 10 are prohibited. If they are set, they will be treated as if 00 were set (mask).
0	CSC_MSK	Controls interrupt requests according to the CSTSCHG_EV bit of the SKT_EV register 1: Do not mask 0: Mask

13.4.3 SKT_PRE_STATE (offset address: 0x008)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	YV_SKT	XV_SKT	3V_SKT	5V_SKT	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	1	1	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	YV_CARD_DT	XV_CARD_DT	3V_CARD_DT	5V_CARD_DT	BAD_VCC_REQ	DATA_LOST
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	NOT_A_CARD	READY	CB_CARD_DT[Note]	R2_CARD_DT	POW_UP	CCD20	CCD10	CSTSCHG[Note]
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	1	1	0

Bit	Name	Function
31	YV_SKT	Supplies YV (arbitrary) voltage. The CARDU unit does not support this function.
30	XV_SKT	Supplies XV (arbitrary) voltage. The CARDU unit does not support this function.
29	3V_SKT	Controls the 3 V power supply. This bit is fixed at 1 (enabled).
28	5V_SKT	Controls the 5 V power supply. This bit is fixed at 1 (enabled).
27:14	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
13	YV_CARD_DT	Detects a YV card 1: Detect a card that is operating at Y.Y V 0: Detect a card that is not operating at Y.Y V
12	XV_CARD_DT	Detects a XV card 1: Detect a card that is operating at X.X V 0: Detect a card that is not operating at X.X V

★ **Note** The CARDU of the VRC4173 does not support a 32-bit PC card (CardBus card).

(2/2)

Bit	Name	Function
11	3V_CARD_DT	Detects a 3 V card 1: Detect a card that is operating at 3 V 0: Detect a card that is not operating at 3 V
10	5V_CARD_DT	Detects a 5 V card 1: Detect a card that is operating at 5 V 0: Detect a card that is not operating at 5 V
9	BAD_VCC_REQ	Invalid V_{CC} request 1: Software requests an invalid V_{CC} 0: Software requests a valid V_{CC} However, if the BAD_VCC_REQ_DISB bit of the SYSCNT register within the configuration registers is 1, the request will be according to that bit.
8	DATA_LOST	Determines the possibility of data being lost by removing the card 1: Data may be lost 0: Data is not lost
7	NOT_A_CARD	Determines whether a card that can be recognized was inserted 1: A card that cannot be recognized was inserted 0: A card that can be recognized was inserted
★ 6	READY	Status of the READY signal of the memory card or IREQ# signal (corresponds to the READY1 or READY2 signal of the V_{RC}4173) of the I/O card 1: Low level 0: High level
5	CB_CARD_DT[Note]	Whether or not the CardBus card (32-bit PC card) is to be detected 1: Detected 0: Not detected
4	R2_CARD_DT	Whether or not the R2 PC card (16-bit PC card) is to be detected 1: Detected 0: Not detected
3	POW_UP	Status of power to the socket 1: Power up completed 0: Power down
2	CCD20	Whether or not the PC card is connected (status of CCD2# signal (corresponds to the CD12# or CD22# signal of the V_{RC}4173)) 1: Not connected (high level) 0: Connected (low level)
1	CCD10	Whether or not the PC card is connected (status of CCD1# signal (corresponds to the CD11# or CD21# signal of the V_{RC}4173)) 1: Not connected (high level) 0: Connected (low level)
0	CSTSCHG[Note]	Status of the CSTSCHG signal (corresponds to the BVD11# or BVD21# signal of the V_{RC}4173) of the CardBus card 1: High level 0: Low level

★ **Note** The CARDU of the V_{RC}4173 does not support a 32-bit PC card (CardBus card).

This register indicates the status of the power supply voltage that is supplied to a card connected to the PC card bus.

13.4.4 SKT_FORCE_EV (offset address: 0x00C)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	CVS_TEST	YV_CARD	XV_CARD	3V_CARD	5V_CARD	BAD_VCC_REQ	DATA_LOST
R/W	R	W	W	W	W	W	W	W
After reset	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	NOT_A_CARD	RFU	CB_CARD	R2_CARD	POW_UP	CCD20	CCD10	CSTSCHG
R/W	W	R	W	W	W	W	W	W
After reset	Undefined	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Name	Function
31:15	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
14	CVS_TEST	When 1 is written to this bit, interlocation is redone.
13	YV_CARD	The value written to this bit becomes the value of the YV_CARD_DT bit of the SKT_PRE_STATE register.
12	XV_CARD	The value written to this bit becomes the value of the XV_CARD_DT bit of the SKT_PRE_STATE register.
11	3V_CARD	The value written to this bit becomes the value of the 3V_CARD_DT bit of the SKT_PRE_STATE register.
10	5V_CARD	The value written to this bit becomes the value of the 5V_CARD_DT bit of the SKT_PRE_STATE register.
9	BAD_VCC_REQ	The value written to this bit becomes the value of the BAD_VCC_REQ bit of the SKT_PRE_STATE register.
8	DATA_LOST	The value written to this bit becomes the value of the DATA_LOST bit of the SKT_PRE_STATE register.
7	NOT_A_CARD	The value written to this bit becomes the value of the NOT_A_CARD bit of the SKT_PRE_STATE register.
6	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

(2/2)

Bit	Name	Function
5	CB_CARD	The value written to this bit becomes the value of the CB_CARD_DT bit of the SKT_PRE_STATE register. However, the value is ignored if there is a card.
4	R2_CARD	The value written to this bit becomes the value of the R2_CARD_DT bit of the SKT_PRE_STATE register. However, the value is ignored if there is a card.
3	POW_UP	When 1 is written to this bit, the POW_CYC_EV bit of the SKT_EV register is set. The POW_UP bit of the SKT_PRE_STATE register is not affected.
2	CCD20	When 1 is written to this bit, the CCD2_EV bit of the SKT_EV register is set. The CCD20 bit of the SKT_PRE_STATE register is not affected.
1	CCD10	When 1 is written to this bit, the CCD1_EV bit of the SKT_EV register is set. The CCD10 bit of the SKT_PRE_STATE register is not affected.
0	CSTSCHG	When 1 is written to this bit, the CSTSCHG_EV bit of the SKT_EV register is set. The CSTSCHG bit of the SKT_PRE_STATE register is not affected.

13.4.5 SKT_CNT (offset address: 0x010)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	STP_CLK_EN	VCC_CNT2	VCC_CNT1	VCC_CNT0	RFU	VPP_CNT2	VPP_CNT1	VPP_CNT0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:8	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
7	STP_CLK_EN	Enables/disables clock stopping according to the clock run protocol 1: Enable 0: Disable
6:4	VCC_CNT(2:0)	Controls the V_{CC} power supply 111: Reserved 110: Reserved 101: V_{CC} = Y.Y V 100: V_{CC} = X.X V 011: V_{CC} = 3.3 V 010: V_{CC} = 5 V 001: Reserved 000: V_{CC} = 0 V (power off)
3	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.

(2/2)

Bit	Name	Function
2:0	VPP_CNT(2:0)	Controls the V_{PP} power supply 111: Reserved 110: Reserved 101: V_{PP} = Y.Y V 100: V_{PP} = X.X V 011: V_{PP} = 3.3 V 010: V_{PP} = 5 V 001: V_{PP} = 12 V 000: V_{PP} = 0 V (power off)

Caution A read/write access to the VCC(1:0) or VPP(1:0) area of the PWR_CNT register within the ExCA registers will also be an access to this register.

13.5 PC Card Unit Operation

This section provides supplementary information about PC card unit operation.

13.5.1 16-bit PC card support

Each CARDU unit has five memory windows and two I/O windows for 16-bit PC cards.

(1) Memory window

Memory mapping can map from the 4 GB address space of a PCI system to the 64 MB address space within a card. The size of each memory window is 4 KB to 16 MB.

Table 13-5 shows the registers related to memory windows.

Table 13-5. Registers Related to Memory Windows

Register Name	Bit Name	Function
ADR_WIN_EN	MEM_WINm_EN	Enables each memory window
MEM_WINm_SAL	MEM_WINm_SAL(7:0)	Start address (A(23:12)) of each window
MEM_WINm_SAH	MEM_WINm_SAH(3:0)	
MEM_WINm_EAL	MEM_WINm_EAL(7:0)	End address (A(23:12)) of each window
MEM_WINm_EAH	MEM_WINm_EAH(3:0)	
MEM_WINm_SAU	MEM_WINm_SAU(7:0)	Higher address (A(31:24)) of each window Determines the 16 MB block of the PCI 4 GB space into which the card's memory area is to be placed.
MEM_WINm_OAL	MEM_WINm_OAL(7:0)	Offset address (A(25:12)) of each window This value is added to the PCI address to obtain the memory address on the card.
MEM_WINm_OAH	MEM_WINm_OAH(5:0)	
	MEM_WINm_REGSET	Common or attribute memory can be selected for each window.
	MEM_WINm_WP	Writing can be disabled/enabled for each window.
MEM_WINm_SAH	MEM_WINm_DSIZE	The data size of each window can be set to 8 or 16 bits.
MEM_TIM_SEL1	MEM_WINn_TIMSEL(2:0)	The access timing can be selected from two timings[Note] for each window.
MEM_TIM_SEL2	MEM_WIN4_TIMSEL(2:0)	
MEM_WIN_PWEN	POSTWEN	The card's memory area post write cycle can be enabled (common for all windows).

Note These are set by using three memory timing registers. The component registers are shown below.

- Timing 0: MEM0_SETUP_TIM register, MEM0_CMD_TIM register, and MEM0_HOLD_TIM register
- Timing 1: MEM1_SETUP_TIM register, MEM1_CMD_TIM register, and MEM1_HOLD_TIM register

Remark m = 0 to 4, n = 0 to 3

(2) I/O window

I/O mapping can map from the first 64 KB address space of the 4 GB address space of a PCI system to the 64 KB address space within a card. The size of each I/O window is 2 KB to 64 KB.

Table 13-6 shows the registers related to I/O windows.

Table 13-6. Registers Related to I/O Windows

Register Name	Bit Name	Function
ADR_WIN_EN	IO_WINn_EN	Enables each I/O window
IO_WINn_SAL	IO_WINn_SAL(7:0)	Start address (A(15:0)) of each window
IO_WINn_SAH	IO_WINn_SAH(7:0)	
IO_WINn_EAL	IO_WINn_EAL(7:0)	End address (A(15:0)) of each window
IO_WINn_EAH	IO_WINn_EAH(7:0)	
IO_WINn_OAL	IO_WINn_OAL(7:0)	Offset address (A(15:0)) of each window This value is added to the PCI address to obtain the I/O address on the card. Always enter 0 in the A0 bit.
IO_WINn_OAH	IO_WINn_OAH(7:0)	
IO_WIN_CNT	IO_WINn_DATA_SEL	Selects whether the data size of each window is to be fixed or determined according to the IOIS16# signal.
	IO_WINn_DATA_SIZE	The data size of each window can be set to 8 or 16 bits.

Remark n = 0, 1

13.5.2 Interrupts

Status interrupt requests of a PC card unit and function interrupt requests from a PC card are reported to the host by using PCI interrupts (parallel).

Table 13-7 shows the sources of interrupts and their masking methods.

★ **Table 13-7. Interrupt Sources and Corresponding Masks**

Card Type	Status/Function	Interrupt Source	Interrupt Mask
16-bit PC card	Status interrupt	Bit of the CARD_SC register within the ExCA registers is set	Bit of the CARD_SCI register within the ExCA registers (corresponding to the CARD_SC register bit) is set
	Function interrupt	The IREQ# signal is active	None

Caution An interrupt due to the CD1# or CD2# signal is masked according to hardware until interlocation ends.

The sources of status interrupts other than those due to the detection of a change in the CD1# or CD2# signal can be detected synchronously with the clock or asynchronously.

Set this detection mode by using the ASYN_INT_MODE bit of the SYSCNT register.

13.5.3 Power supply interface

The PC card unit has three serial signals, which are compatible with TI's TPS2202A, for controlling the socket power supplies (V_{PP} and V_{CC}). The VCC_CNT(2:0) and VPP_CNT(2:0) areas of the SKT_CNT register within the CardBus socket registers are monitored, and when there is a change in their values, the power supply control circuit operates and outputs serial signals (PWCDATA, PWCCLK, PWCLATCH).

Among the serial data values output from the PWCDATA signal, the higher four bits indicate the CARDU2 (slot 2) supply voltage setting and the lower four bits indicate the CARDU1 (slot 1) supply voltage setting.

Table 13-8 shows the relationships between the setting of the VPP_CNT(2:0) area of the CARDU1 (slot 1) SKT_CNT register, the PWCDATA signal, and the supply voltage.

Table 13-9 shows the relationships between the setting of the VCC_CNT(2:0) area of the CARDU1 (slot 1) SKT_CNT register, the PWCDATA signal, and the supply voltage.

Table 13-10 shows the relationships between the setting of the VPP_CNT(2:0) area of the CARDU2 (slot 2) SKT_CNT register, the PWCDATA signal, and the supply voltage.

Table 13-11 shows the relationships between the setting of the VCC_CNT(2:0) area of the CARDU2 (slot 2) SKT_CNT register, the PWCDATA signal, and the supply voltage.

Table 13-8. CARDU1 (Slot 1) V_{PP} Settings

VPP_CNT(2:0)	PWCDATA		Request Voltage (V)	Supply Voltage (V)
	Bit 0	Bit 1		
000	0	0	0	0
001	1	0	12	12
010	0	1	5.0	V_{CC} (slot 1)
011	0	1	3.3	V_{CC} (slot 1)
100	1	1	X.X	Hi-Z
101	1	1	Y.Y	Hi-Z
110	1	1	–	Hi-Z
111	1	1	–	Hi-Z

Remark Hi-Z: High impedance

Table 13-9. CARDU1 (Slot 1) V_{CC} Settings

VCC_CNT(2:0)	PWCDATA		Request Voltage (V)	Supply Voltage (V)
	Bit 2	Bit 3		
000	0	0	0	0
001	1	1	–	0
010	0	1	5.0	5.0
011	1	0	3.3	3.3
100	1	1	X.X	0
101	1	1	Y.Y	0
110	1	1	–	0
111	1	1	–	0

Table 13-10. CARDU2 (Slot 2) V_{PP} Settings

VPP_CNT(2:0)	PWCDATA		Request Voltage (V)	Supply Voltage (V)
	Bit 4	Bit 5		
000	0	0	0	0
001	1	0	12	12
010	0	1	5.0	V_{CC} (slot 2)
011	0	1	3.3	V_{CC} (slot 2)
100	1	1	X.X	Hi-Z
101	1	1	Y.Y	Hi-Z
110	1	1	–	Hi-Z
111	1	1	–	Hi-Z

Remark Hi-Z: High impedance

Table 13-11. CARDU2 (Slot 2) V_{CC} Settings

VCC_CNT(2:0)	PWCDATA		Request Voltage (V)	Supply Voltage (V)
	Bit 6	Bit 7		
000	0	0	0	0
001	1	1	–	0
010	0	1	5.0	5.0
011	1	0	3.3	3.3
100	1	1	X.X	0
101	1	1	Y.Y	0
110	1	1	–	0
111	1	1	–	0

The setting timing is explained next.

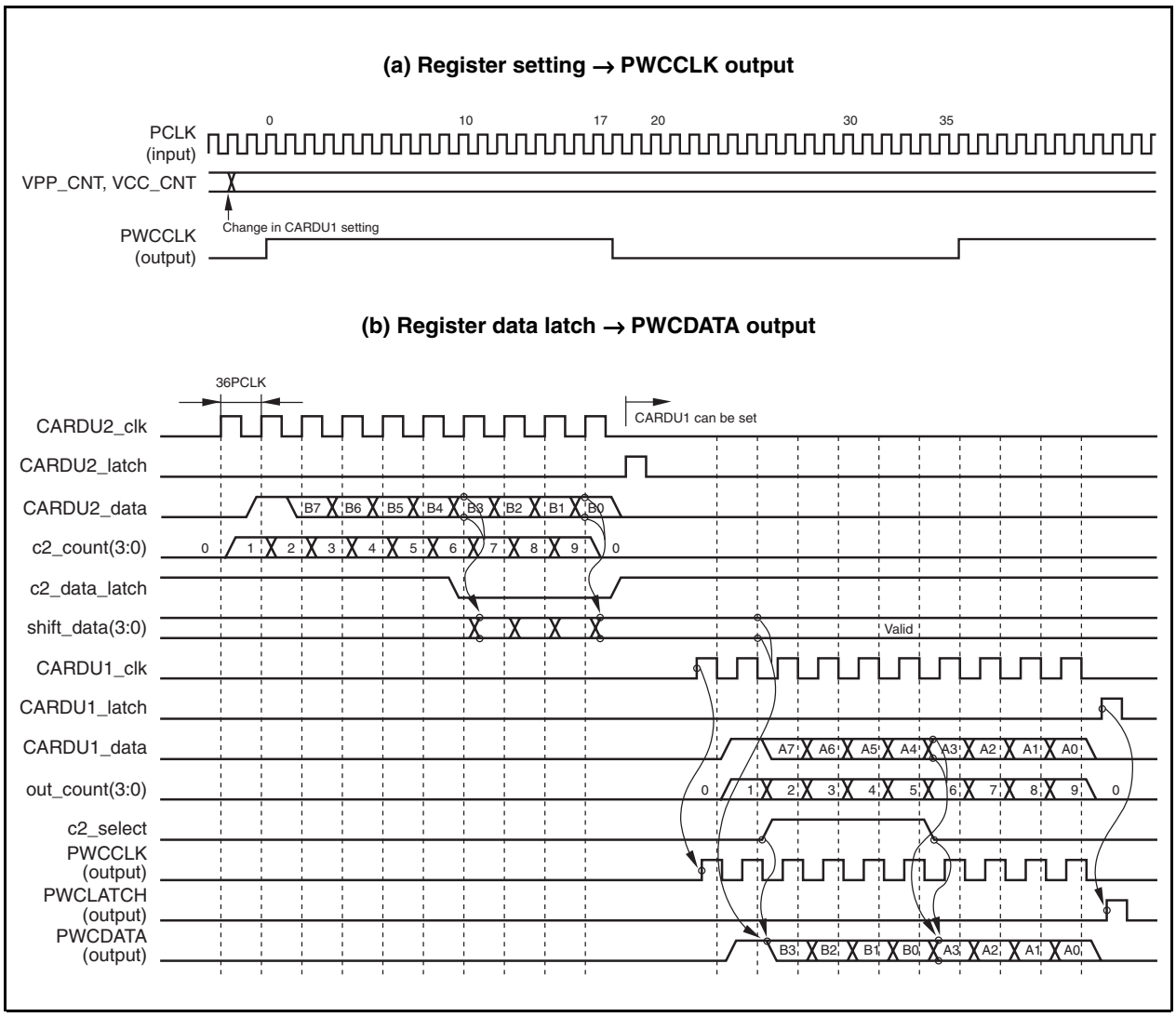
The V_{RC}4173 supports two PC card slots. The supply voltages of each slot are set by writing to the VCC_CNT(2:0) and VPP_CNT(2:0) areas of the SKT_CNT register of the CARDU1 and CARDU2.

Figure 13-5 shows the timing of the power supply control serial signals (PWCDATA, PWCCLK, and PWCLATCH).

Caution When writing to the VCC_CNT(2:0) and VPP_CNT(2:0) areas of the SKT_CNT register, obey the following rules.

- **When setting the VCC_CNT(2:0) and VPP_CNT(2:0) areas for CARDU1 and CARDU2, set the areas for CARDU2 first and then for CARDU1.**
- **Even if you want to set only one of CARDU1 and CARDU2, always set both units. If only one unit is set, the power supply voltage will not be set correctly.**
- ★ **Wait at least 12 μs (395 PCLK) after issuing the CARDU2 setting commands before issuing the CARDU1 supply voltage setting commands.**

Figure 13-5. Power Supply Control Serial Signal (PWCDATA, PWCCLK, PWCLATCH) Timing



CHAPTER 14 USBU (UNIVERSAL SERIAL BUS UNIT)

The USBU of the VRC4173, which is a USB host controller, is compliant with OPEN HCI Specification Release 1.0. The USBU supports power management functions such as PCI/USB-side clock stopping functions. It is also equipped with two downstream ports. The USBU does not support legacy functions.

14.1 Features

USBU features are shown below.

- ○ Functions
  - Compliant with OPEN HCI Specification Release 1.0
  - Communicates with a USB device asynchronously relative to the host CPU
  - Supports full-speed (12 Mbps) and low-speed (1.5 Mbps) USB devices
  - System clock: 48 MHz
- ○ Interface
  - USB interface transceiver
    - Compliant with the Universal Serial Bus Specification 1.0
    - Automatic switching between full speed (12 Mbps) and low speed (1.5 Mbps)
- ○ Communication with the host CPU
  - Via the operational registers within the USB host controller
  - Via the host controller communication area of the system memory space
- ○ Memory and I/O spaces
  - Mapping the operational registers to 4 KB blocks within the 4 GB system memory space
  - Locates a 256-byte host controller communication area within the system memory space
- ○ On-chip FIFO
  - PCI side: 16 bytes (4 × 4 double words)
  - USB side: 64 bytes (64 × 1 byte)
- ○ Root hub
  - Equipped with two downstream ports
- ○ Lower power consumption
  - Has functions for stopping the PCI clock and USB clock

14.2 USB Host Control Configuration Registers

To set the hardware resources to be used by a device, the device characteristics, and device operations, the PCI local bus (internal PCI bus of the VRC4173) accesses the USB host control configuration registers. Each register is accessed according to the PCI configuration cycle. For more detailed information, refer to the PCI Local bus Specification Revision 2.1.

Figure 14-1. USB Host Control Configuration Space

31 24	23 16	15 8	7 0	Offset
Device ID register		Vendor ID register		0x00
Status register		Command register		0x04
Class code register			Revision ID register	0x08
Built-in self-test register	Header type register	Latency timer register	Cache line size register	0x0C
Base address register				0x10
Reserved				0x14
Card bus CIS pointer				0x28
Subsystem ID register		Subsystem vendor ID register		0x2C
Extended ROM base address				0x30
Reserved				0x34
Reserved				0x38
Max_lat register	Min_Gnt register	Interrupt pin register	Interrupt line register	0x3C
Reserved				0x40
Power management register				0xE0

14.2.1 Register set

Table 14-1 lists the USB host control configuration registers.

Table 14-1. USB Host Control Configuration Registers

Offset Address	Register Name	Bits	R/W	Reset Value	Contents
0x00	Vendor ID register	15:0	R	0x1033	Vendor ID (NEC)
★ 0x02	Device ID register	31:16	R	0x0035	Device ID of this macro (USBU)
0x04	Command register	15:0	R/W	0x0000	See **14.2.2**.
0x06	Status register	31:16	R/W	0x0000	See **14.2.3**.
0x08	Revision ID register	7:0	R	0x01	Indicates that it is compliant with the PCI Local bus Specification Revision 2.1.
0x09	Class code base address register	31:24	R	0x0C	Indicates that it is a serial bus controller device.
	Class code sub class register	23:16	R	0x03	Indicates that it is a USB device.
	Class code programming interface register	15:8	R	0x10	Indicates that it is an OpenHCI host controller.
0x0C	Cache line size register	7:0	R	0x00	A cache cannot be used.
0x0D	Latency timer register	15:11	R/W	00000	Interval that the bus cycle continues to be executed.
		10:8	R	000	
0x0E	Header type register	23:16	R	0x80	It is not a PCI-to-PCI bridge.
0x0F	Built-in self-test register	31:24	R	0x00	BIST is not supported.
0x10	Base address register	31:0	R/W	0x0000 0000	See **14.2.4**.
0x2C	Subsystem vendor ID register	15:0	R(/W)	0x0000	(This can be written according to the setting of the ID Write Mask bit of the power management register.)
0x2E	Subsystem ID register	31:16	R(/W)	0x0000	(This can be written according to the setting of the ID Write Mask bit of the power management register.)
0x3C	Interrupt line register	7:0	R(/W)	0x00	Indicates the route of the interrupt request line (this can be written only when the power management register is used).
0x3D	Interrupt pin register	15:8	R	0x01	Indicates that it is equipped with the INTA# signal (internal PCI bus signal).
0x3E	Min_Gnt register (burst cycle minimum request time register)	23:16	R	0x01	Burst cycle minimum request time
0x3F	Max_lat register (bus usage right request frequency register)	31:24	R	0x2A	Maximum delay time until a response is returned when the PCI bus usage right is requested.
0xE0	Power management register	31:0	R/W	0x0000 0000	See **14.2.5**.

14.2.2 Command register (offset address: 0x04)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	Fast back-to-back enable	SERR# enable
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	Wait cycle control	Parity Error response	VGA palette snoop	Memory write and Invalidate enable	Special Cycles	Bus Master	Memory space	I/O space
R/W	R	R	R	R	R	R/W	R/W	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	Fast back-to-back enable	Fast back-to-back access is not supported.
8	SERR# enable	Control of responses to system errors. 1: Do not control 0: Control
7	Wait cycle control	Address/data stepping is not supported.
6	Parity Error response	Parity errors are not checked.
5	VGA palette snoop	The VGA palette snoop function is disabled.
4	Memory write and Invalidate enable	The memory write and invalidate function is disabled.
3	Special Cycles	Special cycles are ignored.
2	Bus Master	Control of bus master operation. 1: Do not control 0: Control
1	Memory space	Control of responses to memory accesses. 1: Do not control 0: Control
0	I/O space	The USBU does not react to I/O accesses.

14.2.3 Status register (offset address: 0x06)

Bit	15	14	13	12	11	10	9	8
Name	Detected parity error	Signaled system error	Received master abort	Received target abort	Signal target abort	DEVSEL timing	DEVSEL timing	Data Parity Error detected
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	Fast back-to-back capable	UDF support	66 MHz capable	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	Detected parity error	Data and address parity error detection 1: Detected 0: Not detected
14	Signaled system error	SERR# signal status 1: Active 0: Inactive
13	Received master abort	When a bus cycle that the USBU had been executing is terminated by a master abort, the master sets this bit to 1. When 0 is written, this bit is cleared.
12	Received target abort	When a bus cycle that the USBU had been executing is terminated by a target abort, the master sets this bit to 1. When 0 is written, this bit is cleared.
11	Signal target abort	When a bus cycle that the USBU accessed is terminated by a target abort, the target sets this bit to 1. When 0 is written, this bit is cleared.
10:9	DEVSEL timing	Active timing of DEVSEL# signal 01: Medium speed
8	Data Parity Error detected	This bit is set to 1 when the following three conditions are satisfied. • The USBU is the bus master of the bus cycle in which the data parity error occurred. • Either the USBU set the PERR# signal to active or the USBU detected that the PERR# signal became active due to the target. • The Parity Error response bit of the command register has been set to 1. Since the Parity Error response bit is fixed at 0 for the USBU unit, this bit will not be set to 1.
7	Fast back-to-back capable	Response to fast Back-to-Back. This is fixed at 0 (disabled).
6	UDF support	UDF is not supported.
5	66 MHz capable	33 MHz operation is set.
4:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

14.2.4 Base address register (offset address: 0x10)

Bit	31	30	29	28	27	26	25	24
Name	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (MSB)	Base address (LSB)	Base address (LSB)	Base address (LSB)	Base address (LSB)
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	Base address (LSB)	Base address (LSB)	Base address (LSB)	Base address (LSB)	Prefetchable	Type	Type	Memory space Indicator
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:12	Base address (MSB)	These are the higher 20 bits of the base address of the operational registers.
11:4	Base address (LSB)	This indicates that the operational registers have a 4 KB address space.
3	Prefetchable	Not prefetchable.
2:1	Type	This indicates that the operational registers can be located anywhere in the 4 GB main memory space.
0	Memory space Indicator	This indicates that the operational registers are mapped to the main memory space.

14.2.5 Power management register (offset address: 0xE0)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	Wakeup_ Enable
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	Wakeup_ Status
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ID Write Mask	PC_mode	REQ_ Enable	RFU	RFU	Status Change Standby	Power Status	Power Status
R/W	R/W	R/W	R/W	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:17	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
16	Wakeup_Enable	Controls WAKE signal (internal signal) output. 1: Enable WAKE signal (internal signal) 0: Disable WAKE signal (internal signal)
15:9	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
8	Wakeup_Status	Whether or not a Wakeup request was received 1: Wakeup request was received 0: No Wakeup request was received Cleared to 0 when 1 is written. Does not change when 0 is written.
7	ID Write Mask	Write protection of subsystem ID and subsystem vendor ID. 1: Write enabled 0: Write mask
6	PC_mode	Controls switching between PC/AT™-compatible and PC-9800 Series modes. 1: PC/AT compatible mode 0: PC-9800 Series mode

(2/2)

Bit	Name	Function
5	REQ_Enable	Controls REQ# signal output timing. 1: PCICLK (internal clock) asynchronous output 0: PCICLK synchronous output
4:3	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
2	Status Change Standby	Device status relative to power status transition control. 1: Can correspond 0: Cannot correspond
1:0	Power Status	Power status control 11: D3 (PCICLK stopped, device power off) 10: D2 (PCICLK stopped, device power on) 01: Reserved 00: D0 (PCICLK full mode)

Remarks 1. Always use the default setting (synchronous) for the REQ_Enable bit. If the asynchronous setting is used, the PCI specifications will be violated.

2. When the PC_mode bit = 0, the Power Status area is disabled. The Power Status area can be read or written from the system, but can only be read from the USB host controller (HC).

3. When the PC_mode bit = 0, the Wakeup_Status bit is disabled. When the PC_mode bit = 1 and the Power Status area = 10, the Wakeup_Status bit is set to 1 when a resume from the USB is detected. At this time, if the Wakeup_Enable bit is 1, set the WAKE signal (internal signal) to active. This bit is cleared when 1 is written to it and the WAKE signal is set to inactive at the same time. The above operations occur only when the RHSC bit of the HcInterruptEnable register is set.

4. When the PC_mode bit = 0, the Wakeup_Enable bit is disabled.

5. After 10 or 11 is set in the Power Status area, 0 is displayed in the Status Change Standby bit until the status change can actually occur, and 1 is displayed when the change can occur. Once the Status Change Standby bit becomes 1 after 10 or 11 is set in the Power Status area, it cannot return to 0.

14.3 Operational Registers

The USBU (USB host controller, HC) has on-chip operational registers, which are windows for communicating with the host CPU. These registers, which are mapped to a 4 KB range of the system's 4 GB main memory space, are used by the host controller driver (HCD). All of these registers are read/written in units of words. The CPU is accessed according to a PC memory cycle via the internal PCI bus. The base address is indicated by the base address register of the USB host control configuration space.

For more detailed information, refer to the **OPEN HCI Specification Release 1.0**.

14.3.1 Register set

Table 14-2 lists the host control operational registers.

Table 14-2. Host Control Operational Registers

Offset Address	R/W (HCD)	R/W (HC)	Register Symbol	Function
0x00	R	R	HcRevision	HC revision register
0x04	R/W	R/W	HcControl	HC control register
0x08	R/W	R/W	HcCommandStatus	HC command register
0x0C	R/W	R/W	HcInterruptStatus	HC interrupt request detection register
0x10	R/W	R	HcInterruptEnable	HC interrupt request enable register
0x14	R/W	R	HcInterruptDisable	HC interrupt request disable register
0x18	R/W	R	HcHCCA	HC base address register
0x1C	R	R/W	HcPeriodCurrentED	HC period current ED register
0x20	R/W	R	HcControlHeadED	HC control list 1st ED register
0x24	R/W	R/W	HcControlCurrentED	HC control list current ED register
0x28	R/W	R	HcBulkHeadED	HC bulk list 1st ED register
0x2C	R/W	R/W	HcBulkCurrentED	HC bulk list current ED register
0x30	R	R/W	HcDoneHead	HC last TD register
0x34	R/W	R	HcFmInterval	HC frame interval register
0x38	R/W	R/W	HcFmRemaining	HC frame bit time remaining register
0x3C	R/W	R/W	HcFmNumber	HC frame counter register
0x40	R/W	R	HcPeriodicStart	HC list processing start register
0x44	R/W	R	HcLSThreshold	HC low speed transfer diagnosis register
0x48	R/W	R	HcRhDescriptorA	HC power supply status register A
0x4C	R/W	R	HcRhDescriptorB	HC power supply status register B
0x50	R/W	R/W	HcRhStatus	HC status register
0x54	R/W	R/W	HcRhPortStatus1	HC port status register 1
0x58	R/W	R/W	HcRhPortStatus2	HC port status register 2

These registers are described in detail below.

14.3.2 HcRevision (offset address: 0x00)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	Revision	Revision	Revision	Revision	Revision	Revision	Revision	Revision
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	1	0	0	0	0

Bit	Name	Function
31:8	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
7:0	Revision	Indicates that the USBU is compliant with the OPEN HCI Specification Release 1.0.

14.3.3 HcControl (offset address: 0x04)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RWE	RWC	IR
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	0	0Note 1	0

Bit	7	6	5	4	3	2	1	0
★ Name	HCFS	HCFS	BLE	CLE	IE	PLE	CBSR	CBSR
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R	R	R	R	R	R
After reset	**Note 2**	**Note 2**	0	0	0	0	0	0

Bit	Name	Function
31:11	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
10	RWE	Remote Wakeup Enable Enables/disables remote wakeup when an upstream resume signal is detected. 1: Enable 0: Disable
9	RWC	Remote Wakeup Connected Remote wakeup signal support. 1: Supported 0: Not supported

Notes 1. Only a hardware reset is possible.

2. When a hardware reset occurs: 0, when a software reset occurs: 1

(2/2)

Bit	Name	Function
8	IR	Interrupt Routing Route of interrupt request generated due to an event registered in the HcInterruptStatus register 1: SMI# signal (internal signal) output 0: USBINT# signal (internal signal) output
7:6	HCFS	Host Controller Functional Status for USB USB operation mode 11: UsbSuspend 10: UsbOperational 01: UsbResume 00: UsbReset
5	BLE	Bulk List Enable Validity of bulk list processing of next frame 1: Valid 0: Invalid
4	CLE	Control List Enable Validity of control list processing of next frame 1: Valid 0: Invalid
3	IE	Isochronous Enable Validity of IsochronousED (Endpoint Descriptor) processing of next frame 1: Valid 0: Invalid
2	PLE	Periodic List Enable Validity of periodic list processing of next frame 1: Valid 0: Invalid
★ 1:0	CBSR	Control Bulk Service Ratio Control/bulk ED service ratio 11: 4:1 10: 3:1 01: 2:1 00: 1:1

14.3.4 HcCommandStatus (offset address: 0x08)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	SOC	SOC
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R	R
R/W (HC)	R	R	R	R	R	R	R/W	R/W
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	OCR	BLF	CLF	HCR
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R/W	R/W	R/W	R/W
After reset	Undefined	Undefined	Undefined	Undefined	0	0	0	0

Bit	Name	Function
31:18	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
17:16	SOC	Scheduling Overrun Count Scheduling overrun error count. Incremented as follows when an error occurs. 00 (initialization) → 01 → 10 → 11 → 00
15:4	RFU	Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
3	OCR	Ownership Change Request Whether or not an HC control change request was received. This bit is reset by the HCD. 1: Received 0: Not received
2	BLF	Bulk List Filled Whether or not a TD (Transfer Descriptor) exists within the bulk list. 1: Exists 0: Does not exist

(2/2)

Bit	Name	Function
1	CLF	Control List Filled Whether or not a TD exists within the control list. 1: Exists 0: Does not exist
0	HCR	Host Controller Reset HC software reset Set (1) by the HCD and cleared (0) by the HC.

14.3.5 HcInterruptStatus (offset address: 0x0C)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	RFU	OC	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R/W	R	R	R	R	R	R
After reset	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RHSC	FNO	UE	RD	SF	WDH	SO
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	Undefined	0	0	0	0	0	0	0

Bit	Name	Function
31	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
30	OC	Ownership Change This bit is set (1) by the HC when the HCD sets the OCR bit of the HcCommandStatus register. When it is not masked, this event immediately generates a system monitor interrupt request (SMI). This bit is cleared when 0 is written.
29:7	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
6	RHSC	Root Hub Status Change This bit is set (1) when the contents of the HcRhStatus register or HcRhPortStatusN (N = 1, 2) register are changed. This bit is cleared (0) when 0 is written.
5	FNO	Frame Number Overflow Change in the value of bit 15 of the HcFmNumber register 1: Change 0: No change

(2/2)

Bit	Name	Function
4	UE	Unrecoverable Error Detection of system error not related to the USB 1: Detected 0: Normal
3	RD	Resume Detected Detection of resume signal 1: Detected 0: Normal
2	SF	Start of Frame This bit is set at the start of a frame. This bit is cleared (0) when 0 is written.
1	WDH	Writeback Done Head This bit is set (1) when the contents of the HcDoneHead register are written to the HccaDoneHead area[Note]. This bit is cleared (0) by the HCD after the contents of the HccaDoneHead area are saved.
0	SO	Scheduling overrun Occurrence of an overrun due to the USB schedule of the current frame 1: Occurred 0: Normal

Note The HccaDoneHead area is in the HCCA (Host Controller Communication Area). The HCCA is a 256-byte system memory area that is used when the HCD and HC communicate. For details, see **14.4.11 HCCA (Host Controller Communication Area)**.

14.3.6 HcInterruptEnable (offset address: 0x10)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	MIE	OC	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RHSC	FNO	UE	RD	SF	WDH	SO
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	0	0	0	0	0	0	0

Bit	Name	Function
31	MIE	Master Interrupt Enable Validity of enabling of interrupts due to events indicated by other bits of this register 1: Valid 0: Invalid
30	OC	Ownership Change Interrupt request due to Ownership Change 1: Enable 0: Disable
29:7	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
6	RHSC	Root Hub Status Change Interrupt request due to Root Hub Status Change 1: Enable 0: Disable

(2/2)

Bit	Name	Function
5	FNO	Frame Number Overflow Interrupt request due to Frame Number Overflow 1: Enable 0: Disable
4	UE	Unrecoverable Error Interrupt request due to Unrecoverable Error 1: Enable 0: Disable
3	RD	Resume Detected Interrupt request due to Resume Detected 1: Enable 0: Disable
2	SF	Start of Frame Interrupt request due to Start of Frame 1: Enable 0: Disable
1	WDH	Writeback Done Head Interrupt request due to HcDoneHead Writeback 1: Enable 0: Disable
0	SO	Scheduling Overrun Interrupt request due to Scheduling Overrun 1: Enable 0: Disable

This register controls the sources that generate hardware interrupt requests.

Although a bit is set when 1 is written, even if 0 is written to a bit, it will be ignored. To clear (0) the value, write 1 to the corresponding bit of the HcInterruptDisable register.

Each bit of this register corresponds to a bit of the HcInterruptStatus register.

A hardware interrupt request is generated when all of the following conditions are satisfied.

- The interrupt source occurred and any bit of the HcInterruptStatus register was set.
- The corresponding bit of the HcInterruptEnable register is set.
- The MIE bit of the HcInterruptEnable register is set.

14.3.7 HcInterruptDisable (offset address: 0x14)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	MIE	OC	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RHSC	FNO	UE	RD	SF	WDH	SO
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	0	0	0	0	0	0	0

Bit	Name	Function
31	MIE	Master Interrupt Enable Validity of enabling of interrupts due to events indicated by other bits of this register 1: Invalid
30	OC	Ownership Change Interrupt request due to Ownership Change 1: Disable
29:7	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
6	RHSC	Root Hub Status Change Interrupt request due to Root Hub Status Change 1: Disable
5	FNO	Frame Number Overflow Interrupt request due to Frame Number Overflow 1: Disable

(2/2)

Bit	Name	Function
4	UE	Unrecoverable Error Interrupt request due to Unrecoverable Error 1: Disable
3	RD	Resume Detected Interrupt request due to Resume Detected 1: Disable
2	SF	Start of Frame Interrupt request due to Start of Frame 1: Disable
1	WDH	Writeback Done Head Interrupt request due to HcDoneHead Writeback 1: Disable
0	SO	Scheduling Overrun Interrupt request due to Scheduling Overrun 1: Disable

This register is used to clear (0) the corresponding bit of the HcInterruptEnable register.

When 1 is written to a bit of this register, the corresponding bit of the HcInterruptEnable register is cleared (0). Even if 0 is written to a bit of this register, it will be ignored.

When this register is read, the value of the HcInterruptEnable register is returned.

14.3.8 HcHCCA (offset address: 0x18)

Bit	31	30	29	28	27	26	25	24
Name	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA	HCCA
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	HCCA	Host Controller Communication Area Base address of the host controller communication area. Since this area is located in 256-byte units, bits 7 to 0 are fixed at 0.

14.3.9 HcPeriodCurrentED (offset address: 0x1C)

Bit	31	30	29	28	27	26	25	24
Name	PCED	PCED	PCED	PCED	PCED	PCED	PCED	PCED
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	PCED	PCED	PCED	PCED	PCED	PCED	PCED	PCED
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	PCED	PCED	PCED	PCED	PCED	PCED	PCED	PCED
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PCED	PCED	PCED	PCED	PCED	PCED	PCED	PCED
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	PCED	Period Current ED Physical address of the Isochronous/InterruptED of the periodic list processed by the current frame. Since the ED (Endpoint Descriptor) is located in 16-byte units, bits 3 to 0 are fixed at 0.

14.3.10 HcControlHeadED (offset address: 0x20)

Bit	31	30	29	28	27	26	25	24
Name	CHED	CHED	CHED	CHED	CHED	CHED	CHED	CHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	CHED	CHED	CHED	CHED	CHED	CHED	CHED	CHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	CHED	CHED	CHED	CHED	CHED	CHED	CHED	CHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CHED	CHED	CHED	CHED	CHED	CHED	CHED	CHED
R/W (HCD)	R/W	R/W	R/W	R/W	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	CHED	Control Head ED Physical address of the control list 1st ED. For details about the ED (Endpoint Descriptor), see **14.4.3 ED (Endpoint Descriptor)**.

14.3.11 HcControlCurrentED (offset address: 0x24)

Bit	31	30	29	28	27	26	25	24
Name	CCED	CCED	CCED	CCED	CCED	CCED	CCED	CCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	CCED	CCED	CCED	CCED	CCED	CCED	CCED	CCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	CCED	CCED	CCED	CCED	CCED	CCED	CCED	CCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CCED	CCED	CCED	CCED	CCED	CCED	CCED	CCED
R/W (HCD)	R/W	R/W	R/W	R/W	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	CCED	Control Current ED Physical address of the control list current ED. For details about the ED (Endpoint Descriptor), see **14.4.3 ED (Endpoint Descriptor)**.

14.3.12 HcBulkHeadED (offset address: 0x28)

Bit	31	30	29	28	27	26	25	24
Name	BHED	BHED	BHED	BHED	BHED	BHED	BHED	BHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	BHED	BHED	BHED	BHED	BHED	BHED	BHED	BHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	BHED	BHED	BHED	BHED	BHED	BHED	BHED	BHED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	BHED	BHED	BHED	BHED	BHED	BHED	BHED	BHED
R/W (HCD)	R/W	R/W	R/W	R/W	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	BHED	Bulk Head ED Physical address of the bulk list 1st ED. For details about the ED (Endpoint Descriptor), see **14.4.3 ED (Endpoint Descriptor)**.

14.3.13 HcBulkCurrentED (offset address: 0x2C)

Bit	31	30	29	28	27	26	25	24
Name	BCED	BCED	BCED	BCED	BCED	BCED	BCED	BCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	BCED	BCED	BCED	BCED	BCED	BCED	BCED	BCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	BCED	BCED	BCED	BCED	BCED	BCED	BCED	BCED
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	BCED	BCED	BCED	BCED	BCED	BCED	BCED	BCED
R/W (HCD)	R/W	R/W	R/W	R/W	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	BCED	Bulk Current ED Physical address of the bulk list current ED. For details about the ED (Endpoint Descriptor), see **14.4.3 ED (Endpoint Descriptor)**.

14.3.14 HcDoneHead (offset address: 0x30)

Bit	31	30	29	28	27	26	25	24
Name	DH	DH	DH	DH	DH	DH	DH	DH
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	DH	DH	DH	DH	DH	DH	DH	DH
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DH	DH	DH	DH	DH	DH	DH	DH
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DH	DH	DH	DH	DH	DH	DH	DH
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	DH	Done Head Physical address of the last TD that was added to the Done queue (transfer completed queue). For details about the TD (Transfer Descriptor), see **14.4.6 TD (Transfer Descriptor)**.

14.3.15 HcFmInterval (offset address: 0x34)

Bit	31	30	29	28	27	26	25	24
Name	FIT	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS	FSMPS
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	FI	FI	FI	FI	FI	FI
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	1	0	1	1	1	0

Bit	7	6	5	4	3	2	1	0
Name	FI	FI	FI	FI	FI	FI	FI	FI
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	1	0	1	1	1	1	1	1

Bit	Name	Function
31	FIT	Frame Interval Toggle This bit is inverted when a value is loaded into the FI area.
30:16	FSMPS	FS Large Data Packet This is the maximum number of data bits that can be transmitted/received by a single process.
15:14	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
13:0	FI	Frame Interval This is the bit time per frame.

14.3.16 HcFmRemaining (offset address: 0x38)

Bit	31	30	29	28	27	26	25	24
Name	FRT	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R/W	R	R	R	R	R	R	R
After reset	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	FR	FR	FR	FR	FR	FR
R/W (HCD)	R/W	R/W	R	R	R	R	R	R
R/W (HC)	R	R	R/W	R/W	R/W	R/W	R/W	R/W
After reset	Undefined	Undefined	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	FR	FR	FR	FR	FR	FR	FR	FR
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	FRT	Frame Remaining Toggle When the FR area becomes 0, the value of the FIT area of the HcFmInterval register is loaded in this bit.
30:14	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
13:0	FR	Frame Remaining This is a down counter that indicates the remaining bit time of the current frame.

14.3.17 HcFmNumber (offset address: 0x3C)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	FN	FN	FN	FN	FN	FN	FN	FN
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	FN	FN	FN	FN	FN	FN	FN	FN
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
15:0	FN	Frame Number This is a counter that is incremented when the HcFmRemaining register is reloaded.

14.3.18 HcPeriodicStart (offset address: 0x40)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	PS	PS	PS	PS	PS	PS
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	PS	PS	PS	PS	PS	PS	PS	PS
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:14	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
13:0	PS	Periodic Start This area indicates when periodic list processing started. The standard value is 0x3E67 counts (1 count is 0.1 ms).

14.3.19 HcLSThreshold (offset address: 0x44)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	LST	LST	LST	LST
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	0	1	1	0

Bit	7	6	5	4	3	2	1	0
Name	LST	LST	LST	LST	LST	LST	LST	LST
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	1	0	1	0	0	0

Bit	Name	Function
31:12	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
11:0	LST	Low-Speed Threshold This value is compared with the FR area of the HcFmRemaining register to determine whether a low-speed transfer can be performed before the EOF (End of Frame).

14.3.20 HcRhDescriptorA (offset address: 0x48)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	POTPGT	POTPGT	POTPGT	POTPGT	POTPGT	POTPGT	POTPGT	POTPGT
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	1[Note]	1[Note]	1[Note]	1[Note]	1[Note]	1[Note]	1[Note]	1[Note]

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	NOCP	OCPM	DT	NPS	PSM
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	Undefined	Undefined	Undefined	0[Note]	1[Note]	0[Note]	0[Note]	1[Note]

Bit	7	6	5	4	3	2	1	0
Name	NDP	NDP	NDP	NDP	NDP	NDP	NDP	NDP
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0[Note]	0[Note]	0[Note]	0[Note]	0[Note]	0[Note]	1[Note]	1[Note]

Bit	Name	Function
31:24	POTPGT	Poweron To Power Good Time Indicates the time that the HCD must support before accessing a port of the root hub to which power was applied.
23:13	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
12	NOCP	No Over Current Protection Whether or not to report an over current state 1: Over current protection is not supported 0: Report an over current state
11	OCPM	Over Current Protection Mode Over current state report mode 1: Reported for each port 0: Reported for all downstream ports in a batch

Note Only a hardware reset is possible for bits 31 to 24 and bits 12 to 0.

(2/2)

Bit	Name	Function
10	DT	Device Type Indicates that the root hub is not a compound device.
9	NPS	No Power Switching Denial of power application switching 1: If the HC is on, the port power is also always applied. 0: The port power can be on or off.
8	PSM	Power Switching Mode Power application mode 1: The power is applied separately at each port. 0: The power is applied at all ports simultaneously.
7:0	NDP	Number of Downstream Ports Number of downstream ports that are supported by the root hub

14.3.21 HcRhDescriptorB (offset address: 0x4C)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	1Note	1Note	1Note	1Note	1Note	1Note	1Note	1Note

Bit	23	22	21	20	19	18	17	16
Name	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM	PPCM
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	1Note	1Note	1Note	1Note	1Note	1Note	1Note	0Note

Bit	15	14	13	12	11	10	9	8
Name	DR	DR	DR	DR	DR	DR	DR	DR
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0Note	0Note	0Note	0Note	0Note	0Note	0Note	0Note

Bit	7	6	5	4	3	2	1	0
Name	DR	DR	DR	DR	DR	DR	DR	DR
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0Note	0Note	0Note	0Note	0Note	0Note	0Note	0Note

Bit	Name	Function
31:19	PPCM	Reserved. Write 1 to these bits. 1 is returned after a read.
18		Port Power Control Mask Masking of one set of power supplies of port 2 1: Mask 0: Do not mask
17		Port Power Control Mask Masking of one set of power supplies of port 1 1: Mask 0: Do not mask
16		Reserved. Write 0 to this bit. 0 is returned after a read.

Note Only a hardware reset is possible.

(2/2)

Bit	Name	Function
15:3	DR	Reserved. Write 0 to these bits. 0 is returned after a read.
2		Device Removable Connection of device to port 2 1: Connected 0: Not connected
1		Device Removable Connection of device to port 1 1: Connected 0: Not connected
0		Reserved. Write 0 to this bit. 0 is returned after a read.

14.3.22 HcRhStatus (offset address: 0x50)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	CRWE	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	1 Note 1	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	OCIC	**Note 2**
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R/W	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	1 Note 1	0 Note 1

Bit	15	14	13	12	11	10	9	8
Name	**Note 2**	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0 Note 1	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	OCI	**Note 2**
R/W (HCD)	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
R/W (HC)	R	R	R	R	R	R	R/W	R
After reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0 Note 1	0 Note 1

Notes 1. Only a hardware reset is possible for bit 31, bits 17 to 15, and bits 1 and 0.

2. The bit names for these bits differ for an HCD read and HCD write. The bit names in each case are as follows.

Bit	Name	
	For HCD Read	For HCD Write
16	LPSC	SGP
15	DRWE	SRWE
0	LPS	CGP

(2/2)

Bit	Name	Function
31	CRWE	Clean Remote Wakeup Enable 1: Clear the RWE bit of the HcControl register to 0. 0: No change
30:18	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
17	OCIC	Over Current Indicator Change The HC sets 1 in this bit when a change occurs in the OCI bit. This bit is cleared to 0 when 1 is written by the HCD. It does not change when 0 is written by the HCD.
16	SGP[Note]	<For HCD write> Set Global Power 1: When the PSM bit of the HcRhDescriptorA register is 1, set (1) only the PPS bit of the HcRhPortStatus register of the port that is not set in the PPCM area of the HcRhDescriptorB register. When the PSM bit of the HcRhDescriptorA register is 0, turn on the port of all ports. 0: No change
	LPSC	<For HCD read or HC read> Local Power Status Change The root hub does not support the local power status.
15	SRWE[Note]	<For HCD write> Set Remote Wakeup Enable 1: Set the RWE bit of the HcControl register 0: No change
	DRWE	<For HCD read or HC read> Device Remote Wakeup Enable 1: Indicates that the event is a remote wakeup event when the CSC bit of the HcRhPortStatus register is 1. 0: Indicates that the event is not a remote wakeup event when the CSC bit of the HcRhPortStatus register is 1.
14:2	RFU	Reserved. Write 0 to these bits. Since the values of these bits are undefined after a reset, initialize them by using software.
1	OCI	Over Current Indicator Existence of an over current state (when all downstream ports are reported in a batch) 1: An over current state exists 0: Current is operating normally When an over current state is reported for each port, clear this bit to 0.
0	CGP[Note]	<For HCD write> Clear Global Power 1: When the PSM bit of the HcRhDescriptorA register is 1, clear to 0 only the PPS bit of the HcRhPortStatus register of the port that is not set in the PPCM area of the HcRhDescriptorB register. When the PSM bit of the HcRhDescriptorA register is 0, turn off the power supplies of all ports. 0: No change
	LPS	<For HCD read or HC read> Local Power Status The root hub does not support the local power supply status.

Note These bits can be only written by the HCD. They cannot be read/written by the HC.

14.3.23 HcRhPortStatus1, 2 (offset address: 0x54, 0x58)

(1/4)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W (HCD)	R	R	R	R	R	R	R	R
R/W (HC)	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	PRSC	POCIC	PSSC	PESC	CSC
R/W (HCD)	R	R	R	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0Note 1	0Note 1	0Note 1	0Note 1	0Note 1

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	**Note 2**	**Note 2**
R/W (HCD)	R	R	R	R	R	R	R/W	R/W
R/W (HC)	R	R	R	R	R	R	R/W	R/W
After reset	0	0	0	0	0	0	UndefinedNote 1	0Note 1

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	**Note 2**	**Note 2**	**Note 2**	**Note 2**	**Note 2**
R/W (HCD)	R	R	R	R/W	R/W	R/W	R/W	R/W
R/W (HC)	R	R	R	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0Note 1	0Note 1	0Note 1	0Note 1	0Note 1

Notes 1. Only a hardware reset is possible for bits 20 to 16, bits 9 and 8, and bits 4 to 0.

2. The bit names for these bits differ for an HCD or HC read and HCD or HC write. The bit names in each case are as follows.

Bit	Name	
	For Reading	For Writing
9	LSDA	CPP
8	PPS	SPP
4	PRS	SPR
3	POCI	CSS
2	PSS	SPS
1	PES	SPE
0	CCS	CPE

(2/4)

Bit	Name	Function
31: 21	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
20	PRSC	Port Reset Status Change This bit is set (1) due to the completion of a reset. It is cleared (0) when 1 is written by the HCD.
19	POCIC	Port Over Current Indicator Change This bit is set (1) due to a change in the POCI bit. It is cleared (0) when 1 is written by the HCD.
18	PSSC	Port Suspend Status Change This bit is set (1) due to the completion of a resume. It is cleared (0) when either 1 is written by the HCD or the PRSC bit is set (1).
17	PESC	Port Enable Status Change This bit is set (1) due to the occurrence of an over current, a port power off state due to a device disconnection, or the occurrence of an operational error such as the bubble detection. It is cleared (0) when 1 is written by the HCD.
16	CSC	Connect Status Change This bit is set (1) due to a change in the CCS bit, the setting of the corresponding bit in the DR area of the HcRhDescriptorB register, or the writing of 1 to the SPR bit, SPE bit, or SPS bit when the CCS bit is 0. It is cleared (0) when 1 is written by the HCD.
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	CPP	<For writing> Clear Port Power This bit is set (1) due to the connection of a low-speed device. It is cleared (0) due to the connection of a full-speed device.
	LSDA	<For reading> Low-Speed Device Attached Speed of connected device 1: Low speed 0: Full speed

(3/4)

Bit	Name	Function
8	SPP	<For writing> Set Port Power This bit is set to 1 at the following times. • When the PSM bit of the HcRhDescriptorA register is 0 and the SGP bit of the HcRhStatus register becomes 1 • When the NPS bit of the HcRhDescriptorA register is 0, the PSM bit is 1, the corresponding bit of the PPCM area of the HcRhDescriptorB register is 0, and the SGP bit of the HcRhStatus register becomes 1 • When the NPS bit is 0, the PSM bit is 1, and the corresponding bit of the PPCM area is set to 1 This bit is cleared to 0 at the following times. • When the NPS bit is 0, the PSM bit is 0, and the CGP bit of the HcRhStatus register is set to 1 • When the NPS bit is 0, the PSM bit is 1, the corresponding bit of the PPCM area is 0, and the CGP bit is set to 1 • When the NPS bit is 0, the PSM bit is 1, the corresponding bit of the PPCM area is 1, and the CCP bit is set to 1 • When an over current occurs
	PPS	<For reading> Port Power Status Port power status 1: On 0: Off
7:5	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
4	SPR	<For writing> Set Port Reset This bit is set (1) when the CCS bit is 1 and 1 is written by the HCD. It is cleared (0) when the PRSC bit is set (1) and the HCFS area of the HcControl register is 01 or when the port power is off.
	PRS	<For reading> Port Reset Status Port reset signal status 1: Active 0: Inactive
3	CSS	<For writing> Clear Suspend Status This bit is set (1) when the NOCP bit of the HcRhDescriptorA register is 0, the OCPM bit is 1, and an over current occurs. It is cleared (0) when the conditions that caused it to be set are canceled.
	POCI	<For reading> Port Over Current Indicator Occurrence of an over current 1: Occurred 0: Normal

(4/4)

Bit	Name	Function
2	SPS	<For writing> Set Port Suspend This bit is set (1) when the CCS bit is 1 and 1 is written by the HCD. It is cleared (0) when the PSSC bit is set (1) or the PRSC bit is set (1), the HCFS area of the HcControl register is 01, or the port power is off.
	PSS	<For reading> Port Suspend Status Port suspend 1: Suspended 0: Not suspended
1	SPE	<For writing> Set Port Enable This bit is set (1) when the CCS bit is 1 and 1 is written by the HCD or when the PRSC bit is set (1) or the PSSC bit is set (1). It is cleared (0) due to the occurrence of an over current, a port power off state due to a device disconnection, the occurrence of an operational error such as the bubble detection, or the setting (1) of the CPE bit.
	PES	<For reading> Port Enable Status Port status 1: Enable 0: Disable
0	CPE	<For writing> Clear Port Enable This bit is set (1) when a device is connected. It is cleared (0) when the device is disconnected or the port power is off.
	CCS	<For reading> Current Connect Status Current status of downstream port 1: Device is connected 0: Device is not connected

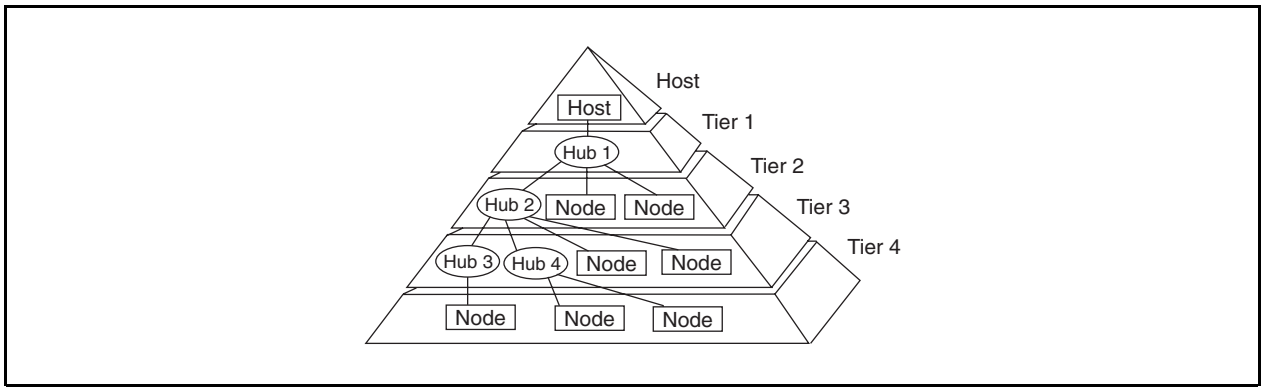
14.4 USB Specifications

This section describes communication functions and operations using the USB and the structure of the interface data that is used. For details, see Open HCI Specification Release 1.0.

14.4.1 General

The Universal Serial Bus (USB) is a serial bus for exchanging data between a host computer and various types of peripheral devices. The USB host and USB devices are connected by point-to-point connections using a scheme called a tiered star topology. A device called a hub is at the center of each star. Figure 14-2 shows the USB bus topology. At most 127 devices can be connected by using this tiered star topology on a USB. In addition, a device can be removed during operation.

Figure 14-2. Bus Topology



USB signals are sent point-to-point as differential signals using two signal lines. The two signal rates are full speed (12 Mbps) and low speed (1.5 Mbps). Low speed is used by a device that can have weak EMI protection, such as a mouse, in order to lower the cost of the device. Full-speed and low-speed devices are differentiated by the position of termination resistors, which are connected at both ends of the cable as shown in Figures 14-3 and 14-4. These termination resistors are also used for disconnection detection at each port.

Figure 14-3. Full-Speed Device Cable and Resistor Connections

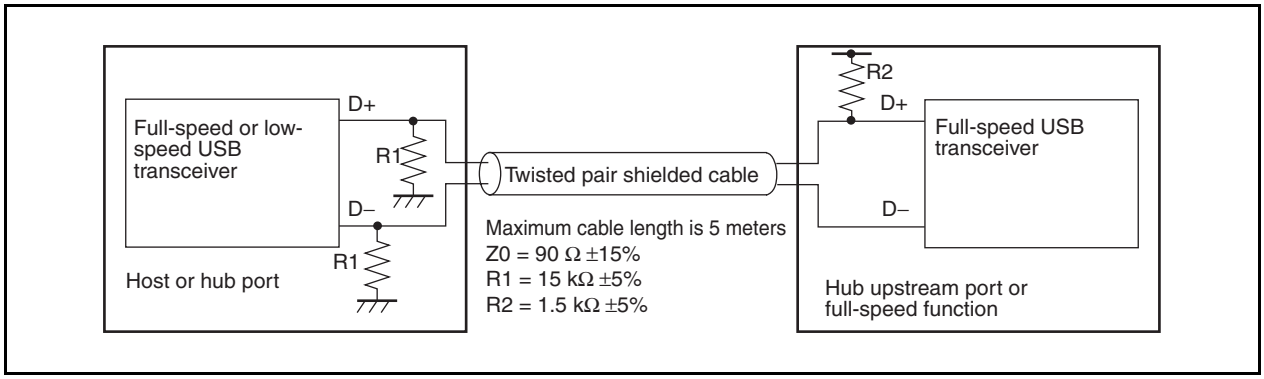
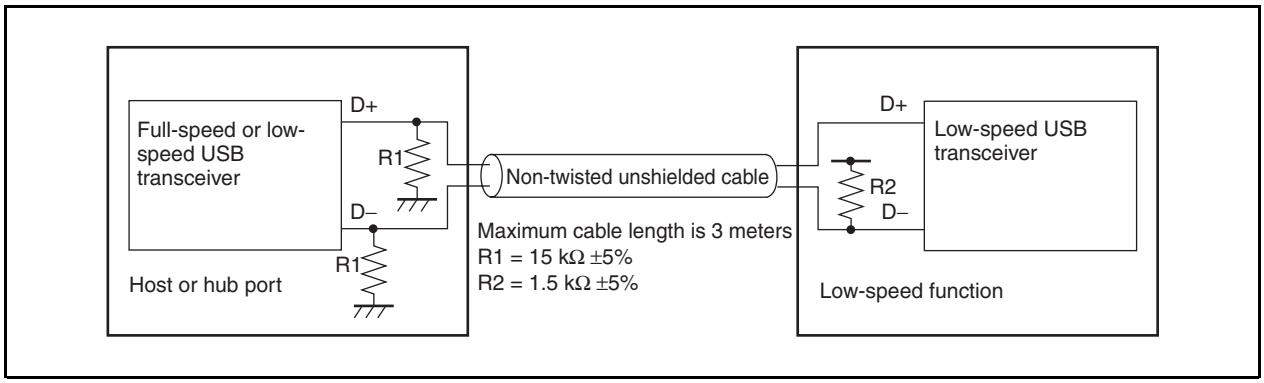


Figure 14-4. Low-Speed Device Cable and Resistor Connections



Data transfers are scheduled and managed by the host. Therefore, transfers are always initiated by the host. All transfers consist of at most three packets (token, data, and handshake). The token packet sends information such as the type and direction of processing, the address, and endpoint to the USB device. The USB device decodes the address field and determines whether or not it is being accessed itself. A data packet transmits data in the data transfer direction (either from the host to the device or from the device to the host) indicated by the token packet. Finally, the receiving side returns a handshake packet to the transmitting side to indicate whether the transfer was successful. The USB has the following four types of data transfers.

- Interrupt transfers: Small-scale data transfers used to convey information from a USB device to client software. The HCD executes USB data transfers by issuing tokens to devices periodically, with a period that satisfies device requests.
- Isochronous transfers: Periodic data transfers having a fixed data transfer rate.
- Control transfers: Asynchronous data transfers, which are used to convey configuration, command, or status information between client software and USB devices.
- Bulk transfers: Asynchronous data transfers, which are used to convey large quantities of information between client software and USB devices.

With OpenHCI, the data transfer types are further divided into the two categories of periodic and asynchronous. Interrupt and isochronous transfers are classified as periodic transfers, which are executed at a certain fixed period. Control and bulk transfers are classified as asynchronous transfers, which are not executed periodically.

To implement this kind of operation, the system must be equipped with a device called the USB host controller and software called the USB host controller driver (USBHCD). OpenHCI are specifications that define the relationship between the host controller and HCD. The USBU unit is compliant with the Open HCI Specification Release 1.0 and Open HCI Legacy Support Interface Specification Release Version 1.01.

14.4.2 Host controller communication methods

The host controller (HC) and host controller driver (HCD) communicate by using the following two routes.

1. Operational registers
2. HCCA (Host Controller Communication Area)

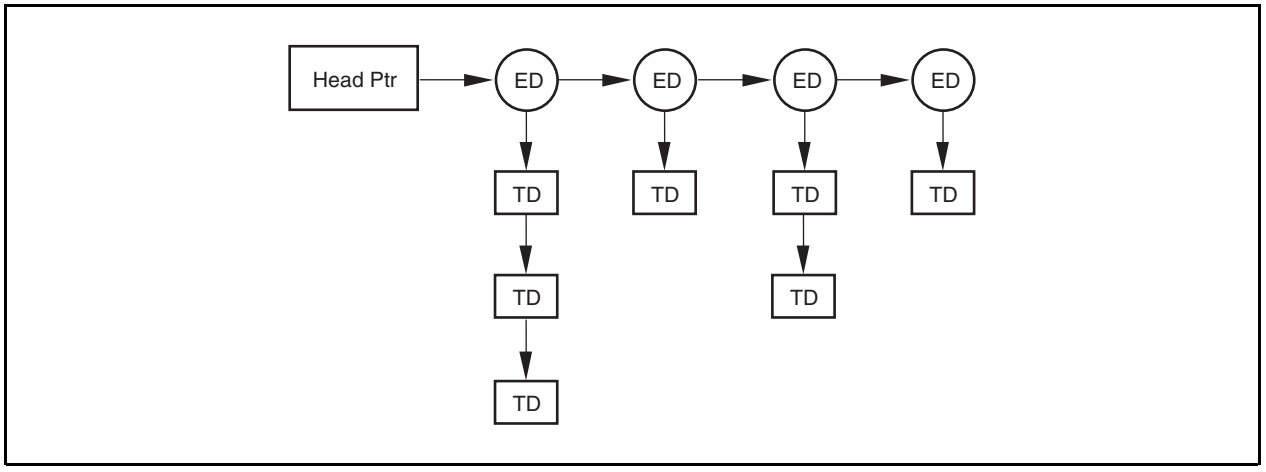
For communication that is performed by using the operational registers contained in the HC, the HC is the PCI target device. The operational registers, which are a set of control, status, list pointer, and other registers, also maintain a pointer that indicates the HCCA position within the system memory. For communication that is performed via the HCCA, the HC is the PCI device master. The HCCA is a 256-byte system memory area that maintains a header pointer for the InterruptED list, header pointer for the Done queue (transfer completed queue), and status information related to frames. By using this system memory, software can directly control HC functions without reading from the HC during normal conditions (for example, when there is no error). These two routes are used to control the HC and to exchange data transfer results on the USB.

Communication between the HC and a USB device is performed based on Endpoint Descriptors (ED) and Transfer Descriptors (TD), which are enqueued by the HCD. An ED maintains information (maximum packet size, endpoint address, endpoint speed, and data flow direction) that is required by the HC to communicate with an endpoint. An ED is also used as an anchor of the TD queue. The HCD generates EDs, assigns them to endpoints, and links them to the list.

A TD maintains information (data toggle information, buffer position in system memory, and completion status code) that is required for data packets that are transferred. Each TD stores information that is related to one or more data packets. The TD data buffer size is from 0 to 8192 bytes. However, only 1024 bytes can be transferred by a single data packet. TDs are processed sequentially beginning with the first one that was entered in the queue. A TD queue is linked with the ED of a given endpoint, and TDs are linked with the TD queue. The HCD creates the data of these structures and passes it to the HC for processing.

Figure 14-5 shows the relationship between EDs and TDs.

Figure 14-5. Relationship Between EDs and TDs



Although ED lists are classified into four types (bulk, control, interrupt, and isochronous), three ED list header pointers are maintained (the isochronous type is excluded). The IsochronousED list is simply linked after the InterruptEDs. The header pointers of the BulkED and ControlED lists are maintained in the operational registers, and the header pointer of the InterruptED list is maintained in the HCCA. There are 32 interrupt header pointers, and the header pointer that is used by a given frame is determined by using the lower 5 bits of the frame counter. The InterruptED list structure is a tree structure like the one shown in Figure 14-6. The execution interval is determined by the depth of the intersections of multiple paths. That is, an InterruptED that is linked to the root of the tree structure is executed at a rate of once per 1 ms.

Figure 14-6. InterruptED List

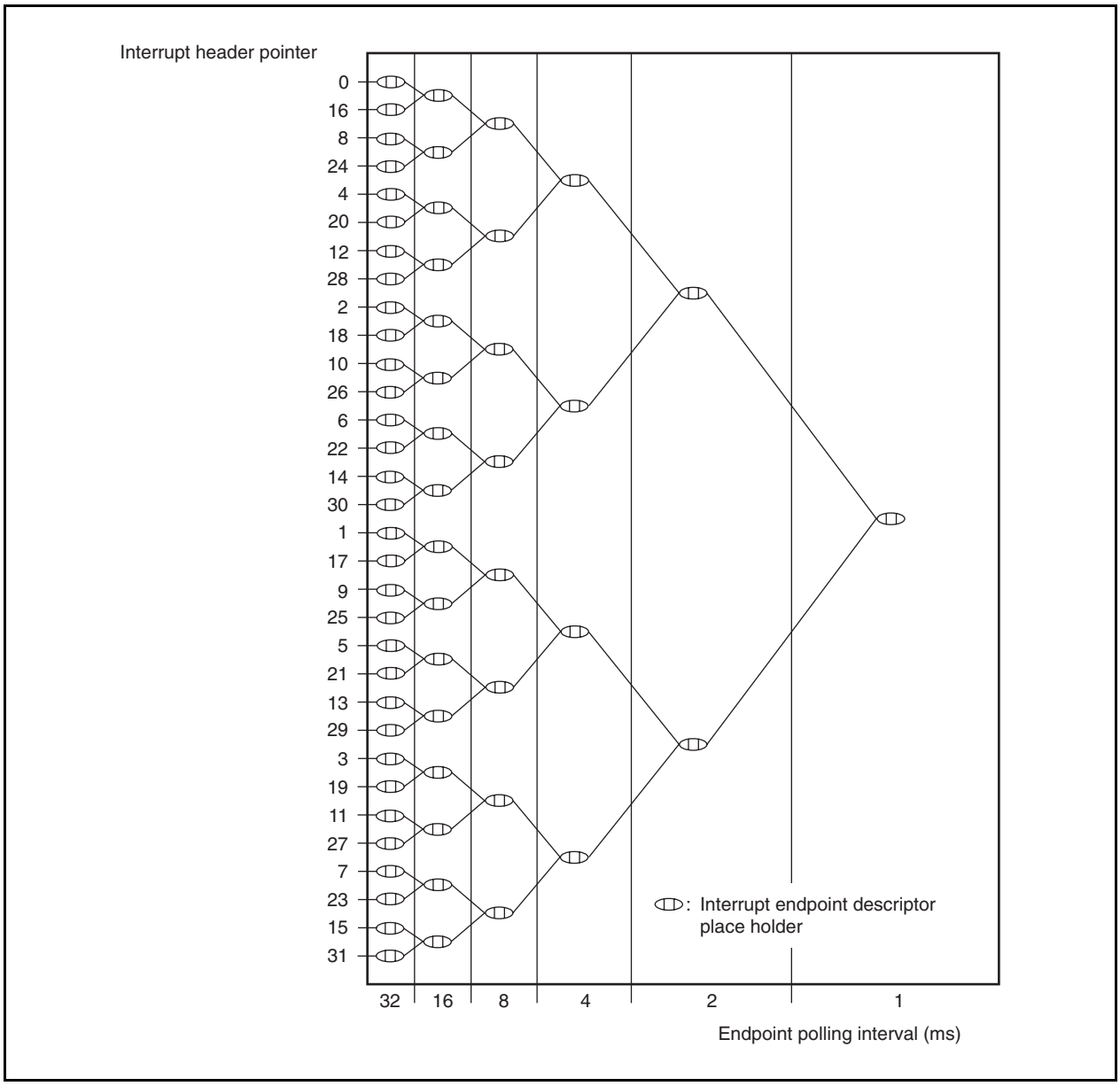
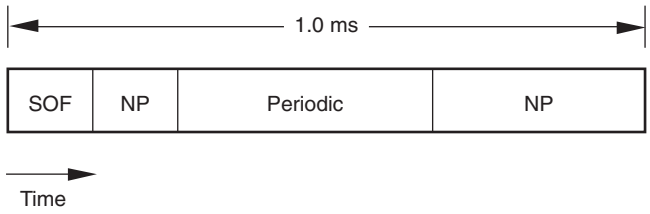


Figure 14-7 shows how OpenHCI allocates bandwidth. The HC selects the list to process based on a priority order algorithm. Control/bulk list processing has priority until the value of the FR area of the HcFmRemaining register from the beginning of the frame is the same as the value of the PS area of the HcPeriodicStart register.

When the value of the FR area is the same as the value of the PS area, periodic list processing has priority. The periodic list processing priority will be greater than or equal to the control/bulk list processing priority until the periodic list processing is completed or the frame time has elapsed. After the periodic list processing is completed, the control/bulk list processing is restarted.

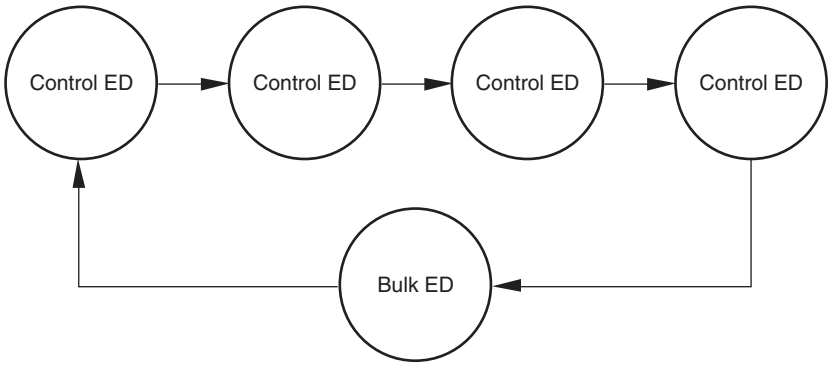
Figure 14-7. Bandwidth Allocation Method



Remark SOF: Start of Frame, NP: Non-Periodic Transfer

During interrupt/isochronous list processing, processing is performed from the InterruptED header pointer that is processed by the current frame. Since the isochronous list is linked after the interrupt list, the interrupt list always has a higher priority than the isochronous list. During bulk/control list processing, processing is restarted from the location where it was previously interrupted in each list. When the end of the list is reached, a value is loaded from the header pointer and processing is resumed. The control end pointer is compared with the bulk end pointer and an equivalent or higher access right to the bus is assigned. The ratio of the access rights is set in the CBSR area of the HcControl register. Figure 14-8 shows an example of a 4:1 control bulk service ratio. While control and bulk transfers have priority, the HC switches the ED processing of each list according to the value in the CBSR area.

Figure 14-8. 4:1 Control Bulk Service Ratio



The control bulk service ratio is maintained across multiple frames. When the processing of one data packet of a TD included in a given ED is serviced, the HC processes the next ED.

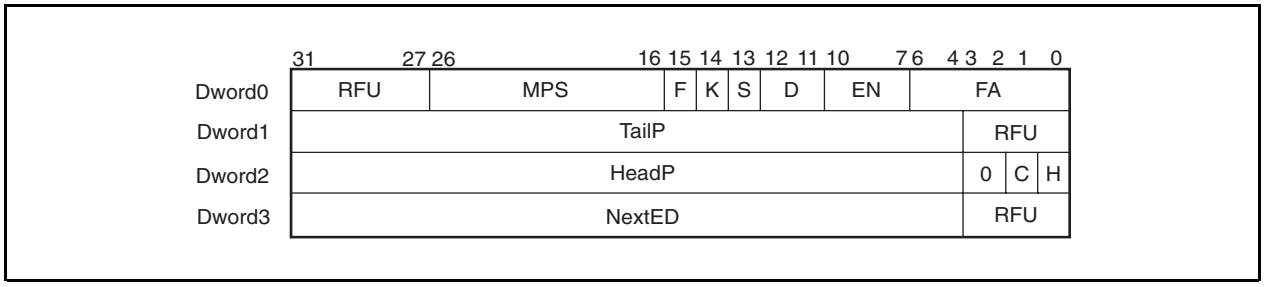
14.4.3 ED (Endpoint Descriptor)

An ED is always located in system memory in units of 16 bytes. When the HC checks an ED list and finds a linked TD, it executes the transfers indicated there. When the HCD must change the value of the HeadP area of an ED, the HCD sets (1) the K bit of the ED to disable all ED list processing having the same transfer type as the ED to be deleted so that the HC will not access the ED.

14.4.4 ED format

Figure 14-9 shows the ED format. For details about each field, see Table 14-3.

Figure 14-9. ED Format



14.4.5 ED fields

Table 14-3 shows details about the ED fields.

Table 14-3. ED Fields (1/2)

Field Name	R/W	Function
FA	R	Function Address USB address of the function that includes the endpoint controlled by this ED
EN	R	Endpoint Number Endpoint address within the function
D	R	Direction Indicates the data flow direction (in or out). If values other than 10 or 01 are set, the transfer direction is defined by the DP area of the TD. 11: Defined by the DP area of the TD 10: In 01: Out 00: Defined by the DP area of the TD
S	R	Speed Indicates the endpoint speed. 1: Low speed 0: Full speed
K	R	Skip When this bit is set, the HC skips to the next ED without accessing the TD queue and without issuing a USB token to the endpoint.
F	R	Format Indicates the format of the TDs linked to this ED. 1: IsochronousTD format (for an isochronous endpoint) 0: GeneralTD format (for a control, bulk, or interrupt endpoint)
MPS	R	Maximum Packet Size Indicates the maximum number of bytes (maximum: 1024 bytes) that can be transmitted to or received from the endpoint in one data packet. The data packet size that is sent to the endpoint by a write (OUT and SETUP) from the HC to the endpoint always is the smaller of the value of this area and the size of the data in the buffer. The data packet size sent by a read (IN) from the endpoint to the HC is determined by the endpoint.
TailP	R	TD Queue Tail Pointer When the values of this area and the HeadP area are the same, it indicates that the list contains no TDs that can be processed by the HC. When the values of this area and the HeadP area differ, the list contains TDs.
H	R/W	Halted This bit usually indicates that the endpoint TD queue processing was halted due to a TD processing error. This bit is set (1) by the HC.

Table 14-3. ED Fields (2/2)

Field Name	R/W	Function
C	R/W	Toggle Carry This bit is a data toggle carry bit. When a TD is retired, this bit is always written using the final data toggle value (LSB of the T area) that was used by the retired TD. This field is not used by an isochronous endpoint.
HeadP	R/W	TD Queue Head Pointer Indicates the TD that is to be processed next by this endpointer.
NextED	R	NextED Indicates the next ED (excluding 0x000 0000).
RFU	R	Reserved for Future Use Reserved. Write 0 to these bits. 0 is returned after a read.

14.4.6 TD (Transfer Descriptor)

TDs (Transfer Descriptors) are used by the HC to indicate a buffer for the data that is transmitted to or received from an endpoint. The two types of TDs are General and Isochronous. A GeneralTD is used by an interrupt, control, or bulk endpointer. An IsochronousTD is used by an isochronous transfer.

Both a GeneralTD and an IsochronousTD can indicate a buffer from 0 to 8192 bytes. In addition, the data buffer described by a single TD can be divided into two pages. This eliminates problems such as forcibly placing buffers so they are physically contiguous and moving excess data.

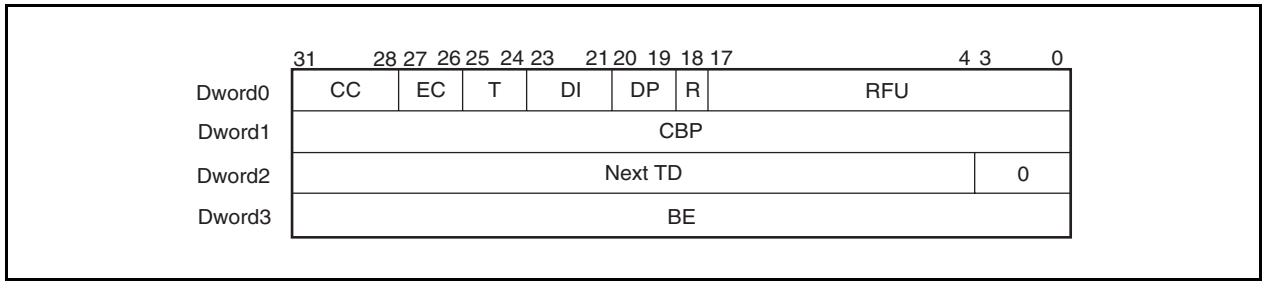
When the HCD adds a TD, it links the new TD to the TD indicated by the TailP area and updates the TailP area so that it points to the added TD. Therefore, the added TD must be added at the end of the TD queue. The HC processes TDs asynchronously relative to processing by the host CPU. Therefore, when the TD queue must be switched to another queue, the HC's endpoint TD queue processing must be halted so the queue can be changed. The HCD halts TD processing by setting (1) the K bit of the ED to be deleted.

14.4.7 GeneralTD format

A GeneralTD is a TD for control, bulk, or interrupt transfers. It is always located in system memory in units of 16 bytes.

Figure 14-10 shows the GeneralTD format. For details about each field, see Table 14-4.

Figure 14-10. GeneralTD Format



14.4.8 GeneralTD fields

Table 14-4 shows details about the GeneralTD fields.

Table 14-4. GeneralTD Fields (1/2)

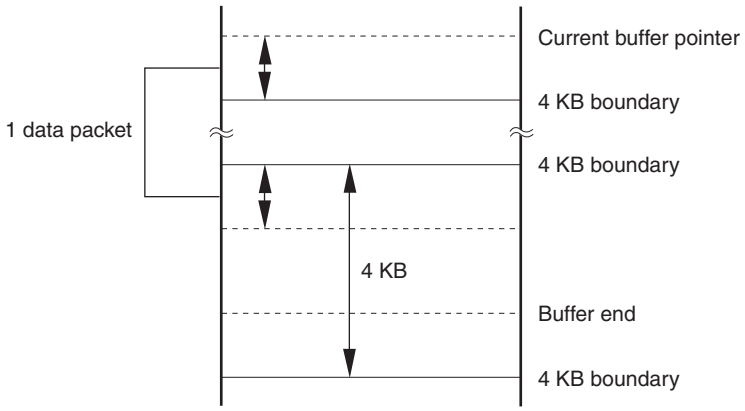
Field Name	R/W	Function
R	R	Buffer Rounding 1: The final data packet will not fill the defined buffer even if no error occurs. 0: The defined data buffer must be completely filled by the final data packet from the endpoint indicated by the TD.
DP	R	Direction/PID (Process ID) Indicates the data flow direction and PID used by the token. If the D area of the ED was set to 00 or 11, this area is meaningful for the HC. 11: Reserved 10: In (from endpoint) 01: Out (to endpoint) 00: Setup (to endpoint)
DI	R	Delay Interrupt Indicates the time until an interrupt request is generated for reporting that TD processing is completed. When TD processing is completed, the HC delays the generation of the interrupt request until the frame indicated by this area. When the value of this area is 111, no interrupt request related to the completion of this TD processing will be generated.
T	R/W	Data Toggle This area is used for comparing/generating the data PID value (DATA0 or DATA1). It is updated each time a data packet is successfully transmitted or received. When the MSB of this area is 0, the data toggle is obtained from the C bit of the ED and the LSB of this area is ignored. When the MSB of this area is 1, the LSB of this area indicates the data toggle.
EC	R/W	Error Count This area is incremented by each transmission error. When the value of this area is 2 and an error occurred, the error type is recorded in the CC area and the data is moved to the Done queue (transfer completed queue). When processing is completed without an error, this area is reset to 0.
CC	R/W	Condition Code This area is updated after processing is executed, regardless of whether or not processing is successful. When processing is successful, this area is set to NoError (0000). Otherwise, it is set according to the error type. 1111: NotAccessed 1110: NotAccessed 1101: BufferUnderrun 1100: BufferOverrun 1011: Reserved 1010: Reserved 1001: DataUnderrun 1000: DataOverrun 0111: UnexpectedPID 0110: PIDCheckFailure 0101: DeviceNotResponding 0100: STALL PID 0011: DataToggleMismatch 0010: BitStuffing Violation 0001: CRC Error 0000: NoError

Table 14-4. GeneralTD Fields (2/2)

Field Name	R/W	Function
CBP	R/W	Current Buffer Pointer Indicates the next physical address in memory to be accessed by a transmission/reception for the endpoint. When this area is 0, it indicates either a data packet of length zero or that all bytes were transferred.
NextTD	R/W	Next TD This area points to the next TD of the TD list that is linked with this endpointer.
BE	R	Buffer End Indicates the physical address of the final byte within this TD buffer.
RFU	R	Reserved for Future Use Reserved. Write 0 to these bits. 0 is returned after a read.

The CBP area of a GeneralTD indicates the address of the data packet that is used in a data packet transfer for the endpoint addressed by the ED. If the transfer is completed without any error occurring, the HC advances the value of the CBP area by the number of bytes that were transferred. If the buffer address indicated by the CBP area exceeds the 4 KB boundary during a data packet transfer, the higher 20 bits of the BE area are copied to the working value (software-determined location for temporarily saving the pointer value) of the CBP area, and the next buffer address will be byte 0 of the same 4 KB page space as the one where the last byte is maintained.

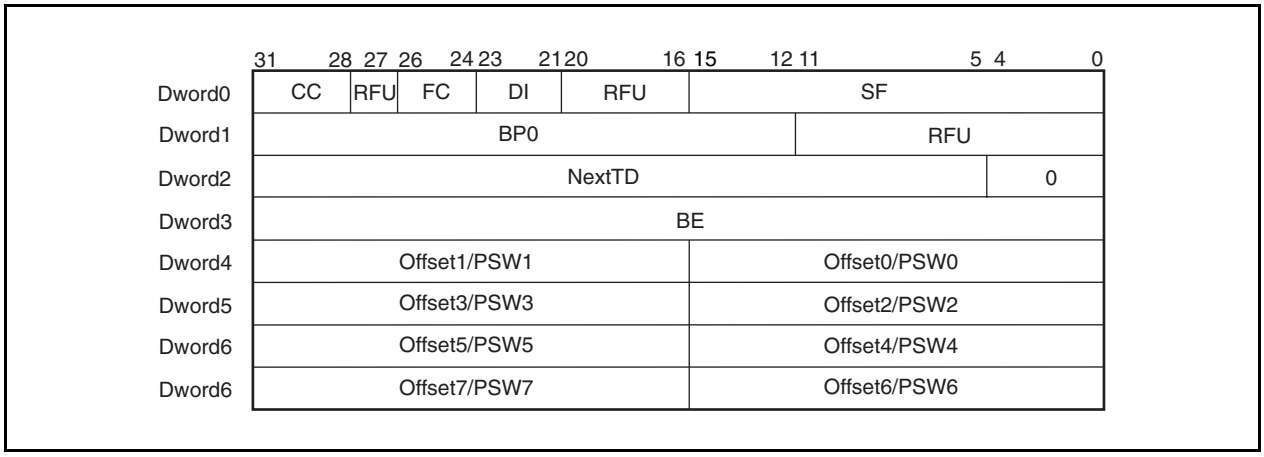
Figure 14-11. Current Buffer Pointer, Buffer End, and 4 KB Boundary



14.4.9 IsochronousTD format

An IsochronousTD is used only for isochronous endpoint. When the F bit of an ED is 1, all TDs that are linked to the ED always use this format, and the TDs are always located in system memory in units of 32 bytes.

Figure 14-12. IsochronousTD Format



14.4.10 IsochronousTD fields

Table 14-5 shows details about the IsochronousTD fields.

Table 14-5. IsochronousTD Fields

Field Name	R/W	Function
SF	R	Starting Frame Lower 16 bits of the frame number that is sent by the first data packet of the IsochronousTD
DI	R	Delay Interrupt Time until an interrupt request is issued after this IsochronousTD processing is completed
FC	R	Frame Count Number of data packets indicated by this IsochronousTD. When this area is 0, it indicates that one data packet is included. When it is 7, it indicates that eight data packets are included.
CC	R/W	Condition Code When an IsochronousTD is moved to the Done queue (transfer completed queue), this area contains the completion code.
BP0	R	Buffer Page 0 This area displays the physical page number of the first byte of the data buffer used by this IsochronousTD.
NextTD	R/W	Next TD This area indicates the next IsochronousTD in the IsochronousTD queue that is liked with the ED.
BE	R	Buffer End This area contains the physical address of the last byte of the buffer.
OffsetN	R	Offset N (N = 0 to 7) This area is used for determining the size and starting address of the isochronous data packet.
PSWN	W	Packet Status Word N (N = 0 to 7) This area contains the completion code and the data size that was received by the isochronous data packet.
RFU	R	Reserved for Future Use Reserved. Write 0 to these bits. 0 is returned after a read.

An IsochronousTD has (FC area value + 1) frame buffers, within the range from 1 to 8. The first data packet is sent by the frame for which the lower 16 bits of the HcFmNumber register matches the SF area of the IsochronousTD. If the buffer address exceeds the 4 KB boundary during a data packet transfer, the higher 20 bits of the BE area are used as the physical address of the next buffer. Therefore, the next buffer address will be byte 0 of the same 4 KB page space as the one where the last byte is maintained.

14.4.11 HCCA (Host Controller Communication Area)

The HCCA (Host Controller Communication Area) is a 256-byte area of system memory, which is used by system software for transmitting specific control/status information to or receiving this information from the HC. The system software always writes this area address in the HcHCCA register of the HC.

14.4.12 HCCA format

Table 14-6 shows the HCCA format.

Table 14-6. HCCA Format

Offset Address	Size (Bytes)	Field Name	R/W	Function
0	128	HccaInterruptTable	R	Pointer to an InterruptED
0x80	2	HccaFrameNumber	W	Displays the current frame number. This value is updated by the HC before periodic list processing begins in the frame.
0x82	2	HccaPad1	W	When the HC updates the HccaFrameNumber area, the HC sets this area to 0.
0x84	4	HccaDoneHead	W	When the HC reaches the end of frame and the decrement value of the value indicated by the DI area of the TD is 0, the HC writes the current value of HcDoneHead to this area. If interrupts are enabled, an interrupt request is generated. This area will not be written to again by the HC until the WDH bit of the HcInterruptStatus register is cleared by software. When this area is zero, an interrupt request is caused by a reason other than an update of this area, and the HcInterruptStatus register must be accessed to determine the source of the interrupt request. When this area is not zero, the interrupt request is a Done queue update interrupt request. When LSB of this area is not zero, an addition separate interrupt source has occurred. Therefore, check the HcInterruptStatus register to determine that source.
0x88	116	reserved	R/W	Reserved for use by the HC.

14.4.13 HCCA overview

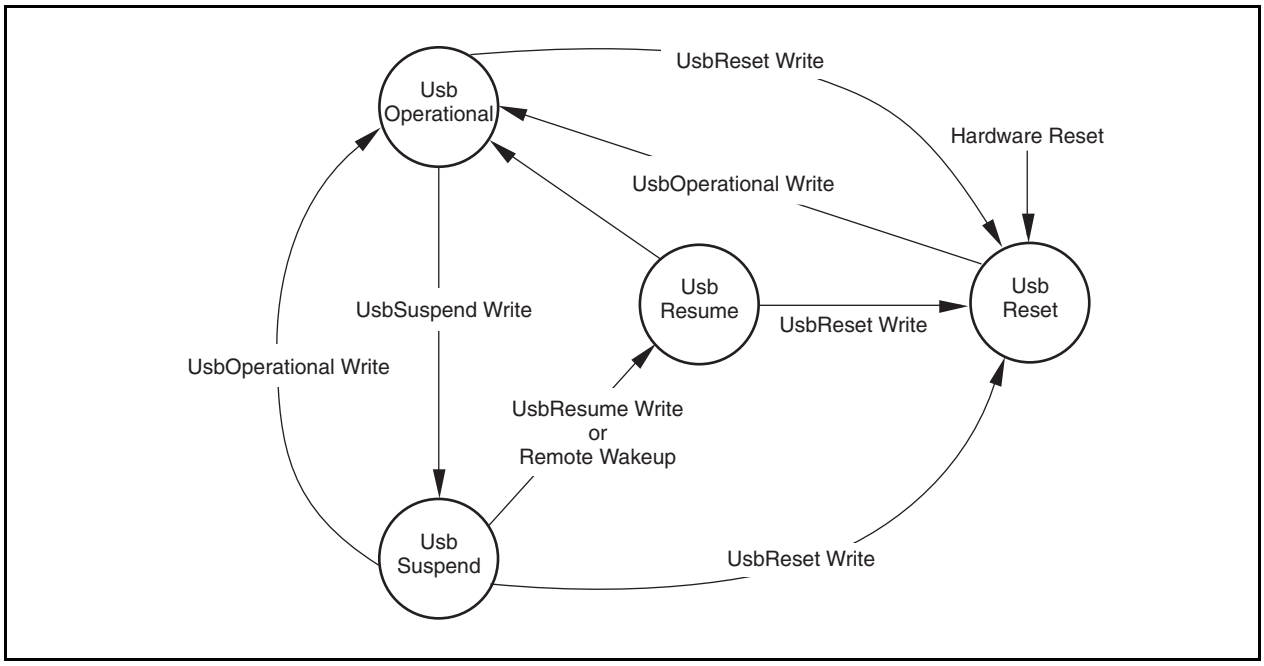
The HccaInterruptTable area, which consists of 32 Dword entry tables, points to pointers to each interrupt list of the ED list. The execution rate is higher for EDs having more linked lists. An ED that exists in only 1 list is executed every 32 ms, and an ED that exists in 2 lists is executed once every 6 ms. When an ED is linked to all 32 lists, it is executed once for each frame. The last entry of each of the 32 interrupt lists must point to an isochronous list.

After the SOF is sent, the HccaFrameNumber area is rewritten with the value of the FN area of the HcFmNumber register before the HC reads the ED to be processed in the new frame.

14.4.14 HC state transitions

The HC has four states. These states are UsbOperational, UsbReset, UsbSuspend, and UsbResume. The current state is indicated in the HCFS area of the HcControl register. The HCD can execute the transitions between the USB states shown in Figure 14-13. The HC can only execute the state transition from UsbSuspend to UsbResume during a remote wakeup event.

Figure 14-13. HC State Transitions



In the UsbOperational state, the HC processes a list and issues an SOF token. At the same time that the HC transitions to the UsbOperational state, the value of the FI area of the HcFmInterval register is loaded in the FR area of the HcFmRemaining register.

The first SOF token that is sent after the HC enters the UsbOperational state is sent at the frame boundary for which the FR area changed from 0 to the value of the FI area.

In the UsbReset state, the HC forcibly sends a reset signal to the bus. After a hardware reset, the HC is always in the UsbReset state.

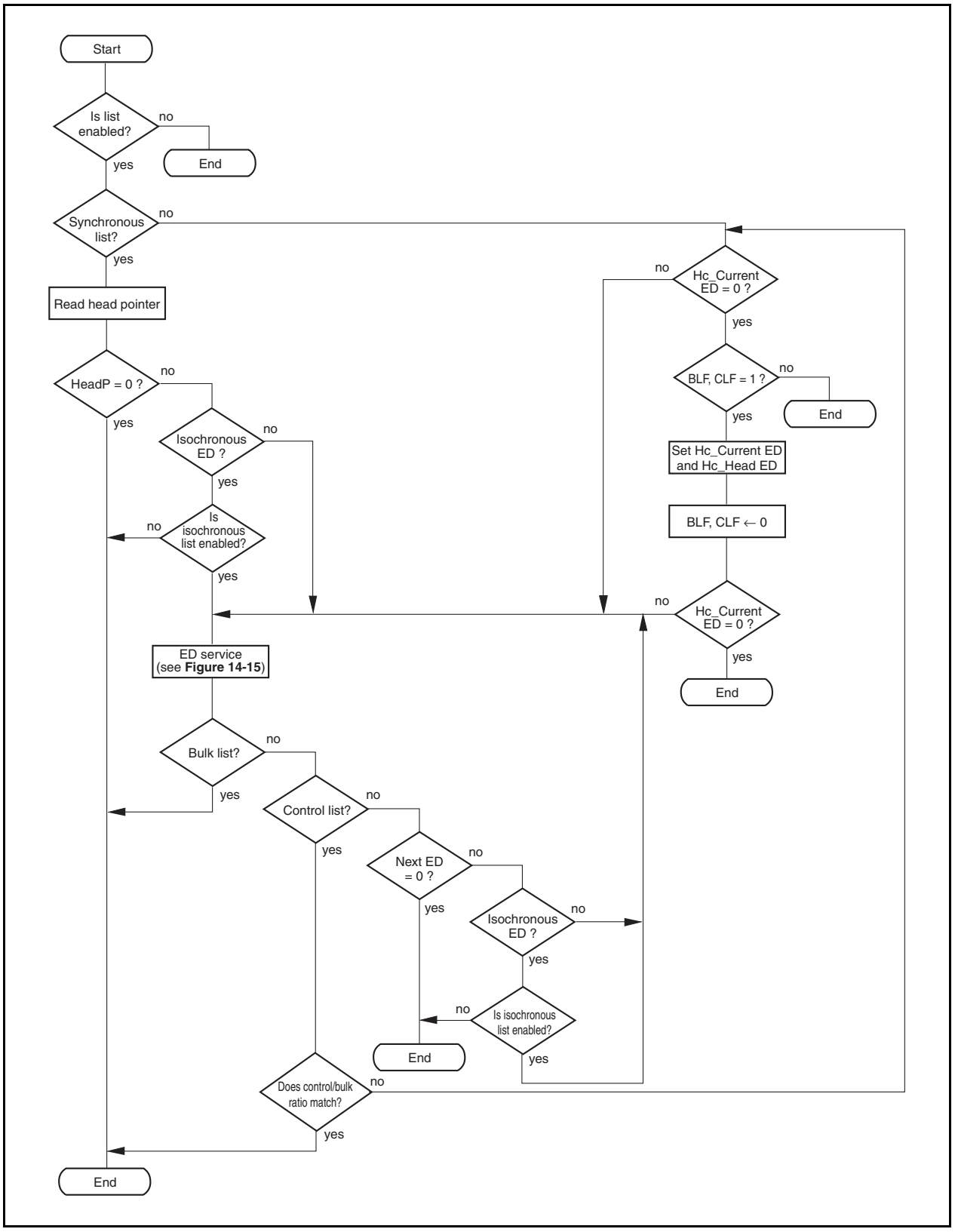
The UsbSuspend state indicates a state in which the USB is temporarily stopped. At this time, the HC monitors USB wakeup actions. The HC is forced to transition to the UsbResume state by a remote wakeup condition. This transition may conflict with a transition to the UsbReset state performed by the HCD. If this conflict occurs, the transition to the UsbReset state that was performed by the HCD has priority. The HC cannot transition to the UsbResume state for 5 ms after a transition to the UsbSuspend State.

In the UsbResume state, the HC forcibly sends a resume signal to the bus. During the UsbResume state, the root hub always transmits USB resume signals to the downstream ports. A transition to the UsbResume state is started by a remote wakeup signal from the HCD or root hub.

14.4.15 List service flow

Figure 14-14 shows the list service flow.

Figure 14-14. List Service Flow



The list service flow is executed after the HC determines the kind of list that must be serviced. A list is periodically disabled by the HCD to switch the ED. Therefore, when processing lists, the HC first checks whether or not the target list is enabled according to the BLE, CLE, and PLE bits of the HcControl register.

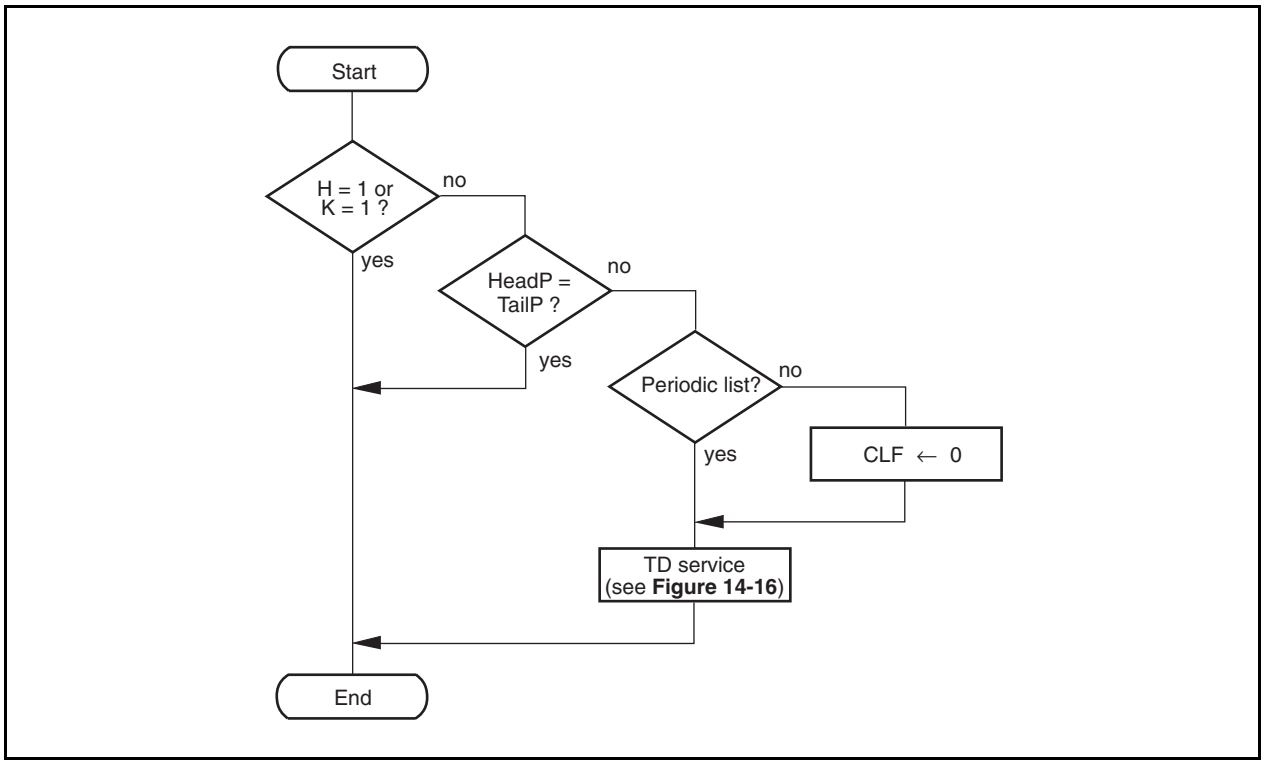
If the list is enabled, the HC services the list. If it is disabled, the HC skips that list and proceeds to the next list.

When a list is enabled, the HC confirms the position of the first ED for which service is requested. If the head pointer is 0 during periodic list processing, no ED exists in the list, and the HC proceeds to the next list. However during asynchronous list processing, if the CurrentED of a list is 0, the HC checks the BLF and CLF bits of the HcCommandStatus register. If these bits had been set to 1, at least one ED for which service is required exists in the target list. Therefore, the HC copies the HeadED to the CurrentED, clears the BLF and CLF bits to 0, and processes the ED that is indicated by the CurrentED. If the BLF and CLF bits are 0 when the HC checks them, the HC proceeds to the next list.

After ED service, for a periodic list, the HC checks the NextED area of the ED for which the service was just completed and continues processing the next ED. If the NextED area is 0x000 0000 at this time, the HC proceeds to an asynchronous list. For a bulk list, the HC only advances to the next list. For a control list, the next action differs according to whether or not the number of ControlEDs indicated by the control/bulk service ratio have been serviced.

Figure 14-15 shows the ED service flow.

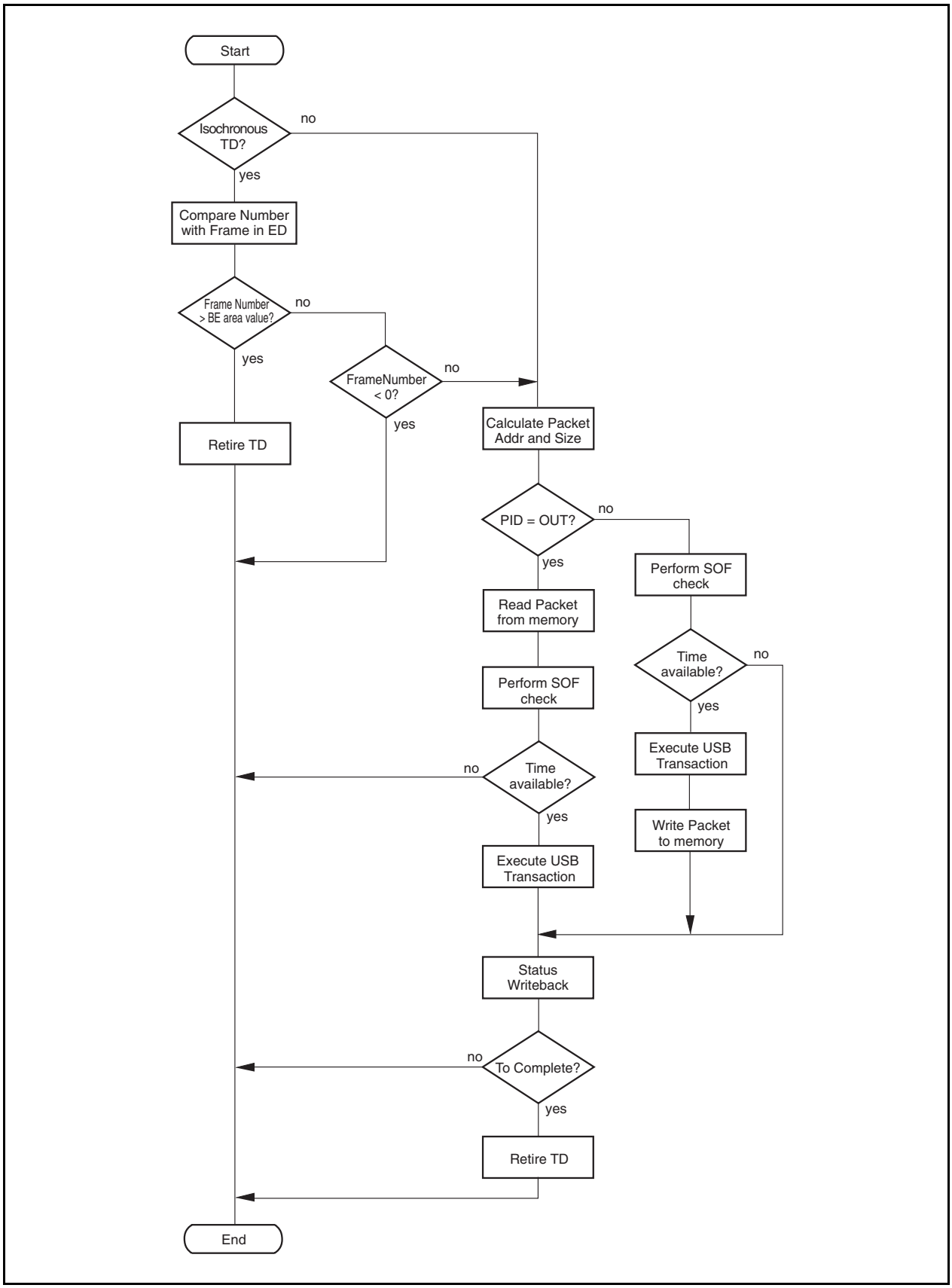
Figure 14-15. ED Service Flow



First, the HC reads the ED to be processed from system memory via the PCI bus. Next, the HC determines whether or not this ED should be processed. If either the K bit or the H bit of the ED is 1, the ED is skipped and the HC proceeds to the next list. When the HC determines that the ED should be processed, it determines whether or not any TDs that can be processed are in the queue. The HC compares the TailP area and HeadP area of the ED. When the values of these areas are equal, no enabled TD exists in the list, and the HC proceeds to the next ED or the next list. If the two values differ, the HC services that TD.

Figure 14-16 shows the TD service flow. When processing an IsochronousTD, the HC first calculates the relative frame number to decide whether to send a packet in the current frame. This relative frame number is used to select Offset(R) and Offset(R+1) (R = 0, 2, 4, 6). When the relative frame number is equal to the value of the FC area of the TD, Offset(R+1) becomes (BE area value + 1). The data buffer size for each transfer is calculated by subtracting Offset(R) from Offset(R+1), and its address is determined from Offset(R). When bit 12 of Offset(R) is 0, buffer page 0 of the IsochronousTD is used as the higher 20 bits of the address. When bit 12 of Offset(R) is 1, the higher 20 bits of the BR area are used as the higher 20 bits of the address.

Figure 14-16. TD Service Flow



When processing a GeneralTD, the HC obtains the next memory address from the CBP area. For a transmission to or reception from the CurrentBufferPointer address of the data, the data many not fit in a single physical page and may span multiple pages. In this case, the higher 20 bits of the BE area are used as the higher 20 bits of the address instead of the higher 20 bits of the CBP area. The maximum amount of data to be transmitted to or received from a device is the smaller of the value of the MPS area of the ED and the remaining buffer size.

After the HC decides the packet size, it always checks whether or not packets can be transferred until the end of frame. If the bit time request of the packets that are transferred is greater than the remaining bit time of the frame, processing is not performed.

After GeneralTD processing, the HC updates the CC, T, EC, and CBP areas of the GeneralTD. Also, after IsochronousTD processing, the HC updates Offset(R) to the value of the PSWN area (R = N = 0 to 7).

When a TD succeeded (all data was transmitted or received) or an error occurred, the HC moves the TD to the Done queue, updates the transfer completed queue interrupt counter (internal Done Queue Interrupt Counter), and updates the ED to change the HeadP, C, and H areas. To enqueue a TD in the Done queue, first the HC copies the value of the NextTD area of the current TD to the HeadP area of the ED. Next, it writes the value of the HcDoneHead register to the NextTD area of the TD that was enqueued. Finally, it writes the address of the TD that was enqueued in the HcDoneHead register.

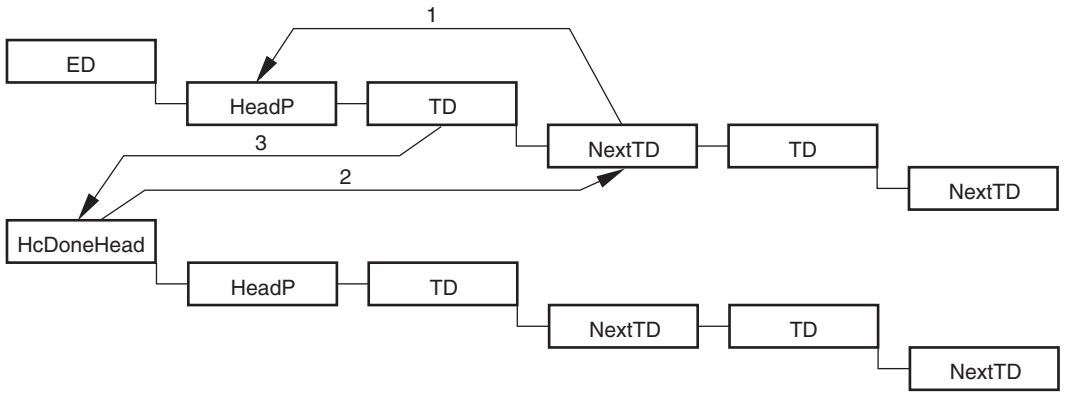
At this time, the HC uses the value of the C area of the ED and the value of the final T area of the TD for updating. When the TD is retired because of an error, the HC also updates the H area of the ED.

After performing these various kinds of processing, the HC writes the value of the HcDoneHead register to the HCCA and updates the transfer completed queue interrupt counter by using the DI area (base for number of SOFs issued), which defines the time until the interrupt request is generated. This counter is not updated if the value of the DI area of the TD is greater than the counter value.

The transfer completed queue interrupt counter is decremented by each SOF, and when it becomes 0, the HC immediately writes the current value of the HcDoneHead register to the HccaDoneHead area at the next frame boundary. After writing the value of the HcDoneHead register to the HCCA, the HC generates an interrupt request by resetting the HcDoneHead register to 0 and setting the WDH bit of the HcInterruptStatus register to 1. In this way, the transfer completed queue is transferred from the HC to the HCD via the HCCA. The HCD processes the Done queue and provides completion information to the software that requested the transfer. While the WDH bit of the HcInterruptStatus register is set, the HC does not write to the HCCA of the HcDoneHead register. In preparation for receiving another transfer completed queue from the HC, the WDH bit of the HcInterruptStatus register is cleared (0) by the HCD.

Figure 14-17 shows the transfer completed queue operation.

Figure 14-17. Transfer Completed Queue Operation



Remark 1 to 3: Operation sequence
→: Writing of data

CHAPTER 15 AC97U (AC97 UNIT)

15.1 General

The AC97U is a digital controller that is compliant with the Audio Codec ’97 Revision 2.1. It is used for connecting with an external Codec through an AC-Link.

15.2 Configuration Register Set

Table 15-1 lists the AC97U PCI configuration registers.

Table 15-1. AC97U PCI Configuration Registers

Offset Address	R/W	Register Symbol	Function
0x00 to 0x01	R	VID	Vendor ID register
0x02 to 0x03	R	DID	Device ID register
0x04 to 0x05	R/W	PCICMD	PCI command register
0x06 to 0x07	R/W	PCISTS	PCI device status register
0x08	R	RID	Revision ID register
0x09 to 0x0B	R	CLASSC	Class code register
0x0C	R	CACHELS	Cache line size register
0x0D	R/W	MLT	Master latency timer register
0x0E	R	HEDT	Header type register
0x0F	R	BIST	Built-in self-test register
0x10 to 0x13	R/W	BASEADR	Base address register
0x14 to 0x2B	–	–	Reserved
0x2C to 0x2D	R	SVID	Subsystem vendor ID register
0x2E to 0x2F	R	SUBID	Subsystem ID register
0x30 to 0x33	R	EXROMADR	Extended ROM base address register
0x34 to 0x3B	–	–	Reserved
0x3C	R/W	INTL	Interrupt line register
0x3D	R	INTP	Interrupt pin register
0x3E	R	MIN_GNT	Burst cycle minimum request time register
0x3F	R	MAX_LAT	Bus usage right request frequency register
0x40 to 0xFF	–	–	Reserved

These registers are described in detail below.

15.2.1 VID (offset address: 0x00 to 0x01)

Bit	15	14	13	12	11	10	9	8
Name	VID15	VID14	VID13	VID12	VID11	VID10	VID9	VID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	1	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	VID7	VID6	VID5	VID4	VID3	VID2	VID1	VID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	1	1	0	0	1	1

Bit	Name	Function
15:0	VID(15:0)	Vendor ID 0x1033: NEC

15.2.2 DID (offset address: 0x02 to 0x03)

Bit	15	14	13	12	11	10	9	8
Name	DID15	DID14	DID13	DID12	DID11	DID10	DID9	DID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DID7	DID6	DID5	DID4	DID3	DID2	DID1	DID0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	1	0	0	1	1	0

Bit	Name	Function
15:0	DID(15:0)	Device ID 0x00A6: AC97U

15.2.3 PCICMD (offset address: 0x04 to 0x05)

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	FBTB_EN	SERREN
R/W	R	R	R	R	R	R	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	AD_STEP	PERREN	VGA_P_ SNOOP	MEMW_ INV_EN	SP_CYC	MASTER_ EN	MEM_EN	IO_EN
R/W	R	R/W	R	R	R	R/W	R	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	FBTB_EN	Enables/disables fast Back to Back. This function is not supported by the AC97U.
8	SERREN	Enables/disables SERR# signal output. 1: Enable The SERR# signal is set to active if an address parity error is detected and the PERREN bit is 1. 0: Disable
7	AD_STEP	Enables/disables address/data stepping. This function is not supported by the AC97U.
6	PERREN	Enables/disables parity error. 1: Enable output of the PERR# signal The PERR# signal is set to active if a data parity error is detected. The SERR# signal is set to active if an address parity error is detected and the SERREN bit is 1. 0: Disable output of the PERR# signal
5	VGA_P_SNOOP	VGA palette snoop. This function is not supported by the AC97U.
4	MEMW_INV_EN	Enables/disables memory write and invalidate. This function is not supported by the AC97U.
3	SP_CYC	Special cycle. This function is not supported by the AC97U.
2	MASTER_EN	Controls bus master operation. 1: Operate as bus master on the PCI bus. 0: Do not operate as bus master on the PCI bus.
1	MEM_EN	Controls memory space. This function is not supported by the AC97U.
0	IO_EN	Controls the response to an I/O space access. 1: Respond to an I/O access to the AC97. 0: Do not respond to an I/O access to the AC97.

15.2.4 PCISTS (offset address: 0x06 to 0x07)

Bit	15	14	13	12	11	10	9	8
Name	DETECT_ PERR	SIG_SERR	RV_ MABORT	RV_ TABORT	SIG_TABOT	DEVSEL1	DEVSEL0	DETECT_ D_PERR
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W
After reset	0	0	0	0	0	0	1	0

Bit	7	6	5	4	3	2	1	0
Name	FBTB_CAP	UDF_SPT	66M_CAP	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15	DETECT_PERR	Data and address parity error detection. Cleared to 0 when 1 is written. 1: Detected 0: Normal
14	SIG_SERR	SERR# signal status. Cleared to 0 when 1 is written. 1: Active 0: Inactive
13	RV_MABORT	Master abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
12	RV_TABORT	Target abort reception. Cleared to 0 when 1 is written. 1: Received 0: Not received
11	SIG_TABOT	Target abort reporting. Cleared to 0 when 1 is written. 1: Reported 0: Not reported
10:9	DEVSEL(1:0)	DEVSEL# timing 01: Medium speed
8	DETECT_D_PERR	Set to 1 when the following three conditions are satisfied. Cleared to 0 when 1 is written. • The AC97U is the master of the bus cycle in which the data parity error occurred. • Either the AC97U set the PERR# signal to active or the AC97U detected that the PERR# signal became active due to the target. • The PERREN bit of the PCICMD register has been set to 1.
7	FBTB_CAP	Response to fast Back to Back. This is fixed at 0 (disabled).
6	UDF_SPT	Indicates that the AC97U does not support the UDF.
5	66M_CAP	Indicates 33 MHz operation.
4:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.2.5 RID (offset address: 0x08)

Bit	7	6	5	4	3	2	1	0
Name	RID7	RID6	RID5	RID4	RID3	RID2	RID1	RID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	RID(7:0)	Revision ID

15.2.6 CLASSC (offset address: 0x09 to 0x0B)

Bit	23	22	21	20	19	18	17	16
Name	CLASSC23	CLASSC22	CLASSC21	CLASSC20	CLASSC19	CLASSC18	CLASSC17	CLASSC16
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	1	0	0

Bit	15	14	13	12	11	10	9	8
Name	CLASSC15	CLASSC14	CLASSC13	CLASSC12	CLASSC11	CLASSC10	CLASSC9	CLASSC8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	7	6	5	4	3	2	1	0
Name	CLASSC7	CLASSC6	CLASSC5	CLASSC4	CLASSC3	CLASSC2	CLASSC1	CLASSC0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
23:0	CLASSC(23:0)	Class code 0x040100: Multimedia device

15.2.7 CACHELS (offset address: 0x0C)

Bit	7	6	5	4	3	2	1	0
Name	CACHELS7	CACHELS6	CACHELS5	CACHELS4	CACHELS3	CACHELS2	CACHELS1	CACHELS0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	CACHELS(7:0)	Sets the cache line size. This function is not supported by the AC97U.

15.2.8 MLT (offset address: 0x0D)

Bit	7	6	5	4	3	2	1	0
Name	MLT7	MLT6	MLT5	MLT4	MLT3	MLT2	MLT1	MLT0
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:4	MLT(7:4)	Sets the latency timer. 1111: 30 PCLK (900 ns) : 0010: 17 PCLK (510 ns) 0001: 16 PCLK (480 ns) 0000: 0 PCLK (0 ns)
3:0	MLT(3:0)	Write 0 to these bits. 0 is returned after a read.

Remark Values enclosed in parentheses are for PCICLK = 33 MHz.

15.2.9 HEDT (offset address: 0x0E)

Bit	7	6	5	4	3	2	1	0
Name	HEDT7	HEDT6	HEDT5	HEDT4	HEDT3	HEDT2	HEDT1	HEDT0
R/W	R	R	R	R	R	R	R	R
After reset	1	0	0	0	0	0	0	0

Bit	Name	Function
7:0	HEDT(7:0)	Header type 0x80: This is a multifunction device and is not a PCI-to-PCI bridge.

15.2.10 BIST (offset address: 0x0F)

Bit	7	6	5	4	3	2	1	0
Name	BIST7	BIST6	BIST5	BIST4	BIST3	BIST2	BIST1	BIST0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	BIST(7:0)	Built-in self-test. This function is not supported by the AC97U.

15.2.11 BASEADR (offset address: 0x10 to 0x13)

Bit	31	30	29	28	27	26	25	24
Name	BASEADR 31	BASEADR 30	BASEADR 29	BASEADR 28	BASEADR 27	BASEADR 26	BASEADR 25	BASEADR 24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	BASEADR 23	BASEADR 22	BASEADR 21	BASEADR 20	BASEADR 19	BASEADR 18	BASEADR 17	BASEADR 16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	BASEADR 15	BASEADR 14	BASEADR 13	BASEADR 12	BASEADR 11	BASEADR 10	BASEADR9	BASEADR8
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	BASEADR7	BASEADR6	BASEADR5	BASEADR4	BASEADR3	BASEADR2	RFU	I/OSpace
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
31:12	BASEADR(31:12)	Sets the higher 20 bits of the operational register base address.
11:2	BASEADR(11:2)	Write 0 to these bits. 0 is returned after a read.
1	RFU	Reserved. Write 0 to this bit. 0 is returned after a read.
0	I/OSpace	Indicates that the operational registers are mapped to the I/O space.

15.2.12 SVID (offset address: 0x2C to 0x2D)

Bit	15	14	13	12	11	10	9	8
Name	SVID15	SVID14	SVID13	SVID12	SVID11	SVID10	SVID9	SVID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SVID7	SVID6	SVID5	SVID4	SVID3	SVID2	SVID1	SVID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SVID(15:0)	Subsystem vendor ID This is a vendor identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

15.2.13 SUBID (offset address: 0x2E to 0x2F)

Bit	15	14	13	12	11	10	9	8
Name	SUBID15	SUBID14	SUBID13	SUBID12	SUBID11	SUBID10	SUBID9	SUBID8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SUBID7	SUBID6	SUBID5	SUBID4	SUBID3	SUBID2	SUBID1	SUBID0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
15:0	SUBID(15:0)	Subsystem ID This is a controller identification number to be used for recognizing the system or option card. The operating system writes and uses this ID.

15.2.14 EXROMADR (offset address: 0x30 to 0x33)

Bit	31	30	29	28	27	26	25	24
Name	EXROMADR 31	EXROMADR 30	EXROMADR 29	EXROMADR 28	EXROMADR 27	EXROMADR 26	EXROMADR 25	EXROMADR 24
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	EXROMADR 23	EXROMADR 22	EXROMADR 21	EXROMADR 20	EXROMADR 19	EXROMADR 18	EXROMADR 17	EXROMADR 16
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	EXROMADR 15	EXROMADR 14	EXROMADR 13	EXROMADR 12	EXROMADR 11	EXROMADR 10	EXROMADR 9	EXROMADR 8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	EXROMADR 7	EXROMADR 6	EXROMADR 5	EXROMADR 4	EXROMADR 3	EXROMADR 2	EXROMADR 1	EXROMADR 0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:0	EXROMADR(31:0)	Extended ROM base address Fixed at 0x0000 0000.

15.2.15 INTL (offset address: 0x3C)

Bit	7	6	5	4	3	2	1	0
Name	INTL7	INTL6	INTL5	INTL4	INTL3	INTL2	INTL1	INTL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
7:0	INTL(7:0)	Sets the interrupt request line. Since this function is not supported by the AC97U, settings for these bits are invalid.

15.2.16 INTP (offset address: 0x3D)

Bit	7	6	5	4	3	2	1	0
Name	INTP7	INTP6	INTP5	INTP4	INTP3	INTP2	INTP1	INTP0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	INTP(7:0)	PCI interrupt pin 0x01: Serial (equipped with INTA# signal)

15.2.17 MIN_GNT (offset address: 0x3E)

Bit	7	6	5	4	3	2	1	0
Name	MIN_GNT7	MIN_GNT6	MIN_GNT5	MIN_GNT4	MIN_GNT3	MIN_GNT2	MIN_GNT1	MIN_GNT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	1

Bit	Name	Function
7:0	MIN_GNT(7:0)	Burst cycle minimum request time. These bits are fixed at 0x01.

15.2.18 MAX_LAT (offset address: 0x3F)

Bit	7	6	5	4	3	2	1	0
Name	MAX_LAT7	MAX_LAT6	MAX_LAT5	MAX_LAT4	MAX_LAT3	MAX_LAT2	MAX_LAT1	MAX_LAT0
R/W	R	R	R	R	R	R	R	R
After reset	0	1	1	1	0	0	0	0

Bit	Name	Function
7:0	MAX_LAT(7:0)	Maximum delay time until a response is returned when the PCI bus usage right is requested. These bits are fixed at 0x70.

15.3 Operational Register Set

Table 15-2 lists the AC97U operational registers. The AC97U operational registers are mapped to the I/O space.

Table 15-2. AC97U Operational Registers

	Offset Address	R/W	Register Symbol	Function
	0x00	R/W	INT_CLR/INT_STATUS	Interrupt clear/status register
	0x04	R/W	CODEC_WR	Codec write register
★	0x08	R	CODEC_RD	Codec read register
★	0x0C	R	CODEC_REQ	Codec slot request register
	0x10	R/W	SLOT12_WR	Slot 12 write register
★	0x14	R	SLOT12_RD	Slot 12 read register
	0x18	R/W	CTRL	Codec/SRC control register
	0x1C	R/W	ACLINK_CTRL	AC-Link control register
	0x20	R/W	SRC_RAM_DATA	Sample rate converter RAM data register
	0x24	R/W	INT_MASK	Interrupt mask register
	0x28 to 0x2C	–	–	Reserved
	0x30	R/W	DAC1_CTRL	DAC1 DMA control register
	0x34	R/W	DAC1L	DAC1 DMA length register
	0x38	R/W	DAC1_BADDR	DAC1 DMA base address register
	0x3C	R/W	DAC2_CTRL	DAC2 DMA control register
	0x40	R/W	DAC2L	DAC2 DMA length register
	0x44	R/W	DAC2_BADDR	DAC2 DMA base address register
	0x48	R/W	DAC3_CTRL	DAC3 DMA control register
	0x4C	R/W	DAC3L	DAC3 DMA length register
	0x50	R/W	DAC3_BADDR	DAC3 DMA base address register
	0x54	R/W	ADC1_CTRL	ADC1 DMA control register
	0x58	R/W	ADC1L	ADC1 DMA length register
	0x5C	R/W	ADC1_BADDR	ADC1 DMA base address register
	0x60	R/W	ADC2_CTRL	ADC2 DMA control register
	0x64	R/W	ADC2L	ADC2 DMA length register
	0x68	R/W	ADC2_BADDR	ADC2 DMA base address register
	0x6C	R/W	ADC3_CTRL	ADC3 DMA control register
	0x70	R/W	ADC3L	ADC3 DMA length register
	0x74	R/W	ADC3_BADDR	ADC3 DMA base address register

15.3.1 INT_CLR/INT_STATUS (offset address: 0x00)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	INTR	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	IOSTS	STSDAT	STSADR	ACLINK_CK
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	CODECGPI	ACLINK	DAC1END	DAC2END	DAC3END	ADC1END	ADC2END	ADC3END
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	INTR	Master interrupt request status 1: Any of bits 11 to 9 and bits 7 to 0 of this register are 1 0: All of bits 11 to 9 and bits 7 to 0 of this register are 0
30:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11	IOSTS	AC97 input data slot 12 valid data input interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
10	STSDAT	AC97 input data slot 2 valid data input interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
9	STSADR	AC97 input data slot 1 valid data input interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.

(2/2)

Bit	Name	Function
8	ACLINK_CK	AC-Link clock request interrupt request, or clock request interrupt request from Codec side during a suspend state 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
7	CODECGPI	Interrupt request when 1 was set for the AC97 input data slot 12 bit 0 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
6	ACLINK	Interrupt request when a loopback transfer is performed and an error occurred 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
5	DAC1END	DAC1 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
4	DAC2END	DAC2 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
3	DAC3END	DAC3 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
2	ADC1END	ADC1 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
1	ADC2END	ADC2 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.
0	ADC3END	ADC3 DMA end interrupt request 1: Interrupt requested 0: No interrupt requested Writing 1 for this bit clears to 0 the interrupt request signal that was set.

15.3.2 CODEC_WR (offset address: 0x04)

Bit	31	30	29	28	27	26	25	24
Name	WRDY	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RWC	WADDR6	WADDR5	WADDR4	WADDR3	WADDR2	WADDR1	WADDR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	WDAT15	WDAT14	WDAT13	WDAT12	WDAT11	WDAT10	WDAT9	WDAT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	WDAT7	WDAT6	WDAT5	WDAT4	WDAT3	WDAT2	WDAT1	WDAT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	WRDY	CODEC register (register within external Codec) access status 1: Writing to Codec prohibited 0: Writing to Codec allowed
30:24	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
23	RWC	Sets a read/write command for the CODEC register 1: Read command 0: Write command
22:16	WADDR(6:0)	CODEC register access address For writing: Set the address when accessing the CODEC register. For reading: The address can be read according to the response from the Codec.
15:0	WDAT(15:0)	CODEC register access data For writing: Set the data when accessing the CODEC register. For reading: The data can be read according to the response from the Codec.

15.3.3 CODEC_RD (offset address: 0x08)

Bit	31	30	29	28	27	26	25	24
Name	RRDYA	RRDYD	WIP	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RADDR6	RADDR5	RADDR4	RADDR3	RADDR2	RADDR1	RADDR0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RDAT15	RDAT14	RDAT13	RDAT12	RDAT11	RDAT10	RDAT9	RDAT8
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RDAT7	RDAT6	RDAT5	RDAT4	RDAT3	RDAT2	RDAT1	RDAT0
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	RRDYA	CODEC register access status 1: The RADDR(6:0) area contains valid data from the Codec 0: The RADDR(6:0) area contains no valid data from the Codec This bit is cleared to 0 when read processing ends.
30	RRDYD	CODEC register access status 1: The RDAT(15:0) area contains valid data from the Codec 0: The RDAT(15:0) area contains no valid data from the Codec This bit is cleared to 0 when read processing ends.
29	WIP	Code read processing status 1: Read processing for the Codec remains 0: There is no read processing for the Codec
28:23	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
22:16	RADDR(6:0)	CODEC register access address The address can be read according to the response from the Codec.
15:0	RDAT(15:0)	CODEC register access data The data can be read according to the response from the Codec.

15.3.4 CODEC_REQ (offset address: 0x0C)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	SLOT3_REQ	SLOT4_REQ
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SLOT5_REQ	SLOT6_REQ	SLOT7_REQ	SLOT8_REQ	SLOT9_REQ	SLOT10_REQ	SLOT11_REQ	SLOT12_REQ
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:10	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
9	SLOT3_REQ	Codec input data slot 1 bit 11 (slot 3 request)
8	SLOT4_REQ	Codec input data slot 1 bit 10 (slot 4 request)
7	SLOT5_REQ	Codec input data slot 1 bit 9 (slot 5 request)
6	SLOT6_REQ	Codec input data slot 1 bit 8 (slot 6 request)
5	SLOT7_REQ	Codec input data slot 1 bit 7 (slot 7 request)
4	SLOT8_REQ	Codec input data slot 1 bit 6 (slot 8 request)
3	SLOT9_REQ	Codec input data slot 1 bit 5 (slot 9 request)
2	SLOT10_REQ	Codec input data slot 1 bit 4 (slot 10 request)
1	SLOT11_REQ	Codec input data slot 1 bit 3 (slot 11 request)
0	SLOT12_REQ	Codec input data slot 1 bit 2 (slot 12 request)

15.3.5 SLOT12_WR (offset address: 0x10)

Bit	31	30	29	28	27	26	25	24
Name	WRDY_SLOT	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	LOOP	RFU	RFU	RFU	WSLOT1219	WSLOT1218	WSLOT1217	WSLOT1216
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	WSLOT1215	WSLOT1214	WSLOT1213	WSLOT1212	WSLOT1211	WSLOT1210	WSLOT129	WSLOT128
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	WSLOT127	WSLOT126	WSLOT125	WSLOT124	WSLOT123	WSLOT122	WSLOT121	WSLOT120
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	WRDY_SLOT	Slot 12 write status 1: Writing to Codec prohibited 0: Writing to Codec allowed
30:24	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
23	LOOP	Slot 12 control 1: Loopback[Note] of SLOT12_WR and SLOT12_RD registers 0: Normal
22:20	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
19:0	WSLOT12(19:0)	Sets write data for slot 12

Note This is a VRC4173 standalone debug function. The value written to the SLOT12_WR register can be read by using the SLOT12_RD register.

15.3.6 SLOT12_RD (offset address: 0x14)

Bit	31	30	29	28	27	26	25	24
Name	RRDY_ SLOT	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RSLOT1219	RSLOT1218	RSLOT1217	RSLOT1216
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RSLOT1215	RSLOT1214	RSLOT1213	RSLOT1212	RSLOT1211	RSLOT1210	RSLOT129	RSLOT128
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RSLOT127	RSLOT126	RSLOT125	RSLOT124	RSLOT123	RSLOT122	RSLOT121	RSLOT120
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	RRDY_SLOT	Slot 12 read status 1: The RSLOT12(19:0) area contains valid data from the Codec 0: The RSLOT12(19:0) area contains no valid data from the Codec This bit is cleared to 0 when read processing ends.
30:20	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
19:0	RSLOT12(19:0)	Slot 12 data from the Codec can be read.

15.3.7 CTRL (offset address: 0x18)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	SRC_RAM_ADR	SRC_CNVT_ON	SRC_FILTER_ON	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R/W	R/W	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	MICENB[Note]	DAC3ENB	ADC3ENB
R/W	R	R	R	R	R	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC2ENB	ADC2ENB	DAC1ENB	ADC1ENB[Note]	RFU	RFU	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	DAC1FORM2	DAC1FORM1	DAC1FORM0	ADC1FORM2	ADC1FORM1	ADC1FORM0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	SRC_RAM_ADR	Selects sample rate converter RAM 1: ADC RAM or MIC RAM 0: DAC RAM
30	SRC_CNVT_ON	Controls sample rate converter 1: Operate converter 0: Stop converter
★ 29	SRC_FILTER_ON	Controls sample rate filter 1: Setting prohibited 0: Stop filter
28:19	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
18	MICENB[Note]	Enables/disables MIC slot 1: Enable 0: Disable
17	DAC3ENB	Enables/disables DAC3 (LINE1) slot 1: Enable 0: Disable

★ **Note** As the DMAs of the slots 3 (PCML) and 6 (MIC) cannot be operated simultaneously, do not set the MICENB bit and the ADC1ENB bit to 1 at the same time.

(2/2)

Bit	Name	Function
16	ADC3ENB	Enables/disables ADC3 (LINE1) slot 1: Enable 0: Disable
15	DAC2ENB	Enables/disables DAC2 (PCMR) slot 1: Enable 0: Disable
14	ADC2ENB	Enables/disables ADC2 (PCMR) slot 1: Enable 0: Disable
13	DAC1ENB	Enables/disables DAC1 (PCML) slot 1: Enable 0: Disable
12	ADC1ENB[Note]	Enables/disables ADC1 (PCML) slot 1: Enable 0: Disable
11:6	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
5:3	DAC1FORM(2:0)	DAC1 format 111: Setting prohibited 110: Input 44 Kss data from cache buffer 101: Input 22 Kss data from cache buffer 100: Input 11 Kss data from cache buffer 011: Input 32 Kss data from cache buffer 010: Input 16 Kss data from cache buffer 001: Input 8 Kss data from cache buffer ★ 000: Input 48 Kss data from cache buffer
2:0	ADC1FORM(2:0)	ADC1 format 111: Setting prohibited 110: Output 44 Kss data to cache buffer 101: Output 22 Kss data to cache buffer 100: Output 11 Kss data to cache buffer 011: Output 32 Kss data to cache buffer 010: Output 16 Kss data to cache buffer 001: Output 8 Kss data to cache buffer ★ 000: Output 48 Kss data to cache buffer

★ **Note** As the DMAs of the slots 3 (PCML) and 6 (MIC) cannot be operated simultaneously, do not set the MICENB bit and the ADC1ENB bit to 1 at the same time.

Remark Kss: Kilo Sampling per Second

15.3.8 ACLINK_CTRL (offset address: 0x1C)

Bit	31	30	29	28	27	26	25	24
Name	ck_stop_on	sync_on	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R/W	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	sync_time7	sync_time6	sync_time5	sync_time4	sync_time3	sync_time2	sync_time1	sync_time0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	15	14	13	12	11	10	9	8
Name	aclink_rst_on	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	aclink_rst_time7	aclink_rst_time6	aclink_rst_time5	aclink_rst_time4	aclink_rst_time3	aclink_rst_time2	aclink_rst_time1	aclink_rst_time0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	1	1	1	1	1	1	1	1

Bit	Name	Function
31	ck_stop_on	Set this bit to 1 when a power save mode or power down mode command is issued for the Codec. This bit is cleared (0) when a clock pulse is supplied from the Codec.
30	sync_on	Set this bit to 1 when you want to start the Codec again after the ck_stop_on bit was set to 1 or when a clock request interrupt request was issued from the Codec. A SYNC signal will be automatically output for the Codec. This bit is automatically cleared to 0 after the SYNC signal is output.
29:24	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
23:16	sync_time(7:0)	Sets the active period of the SYNC signal that is output when 1 was set for the sync_on bit (count clock = 33 MHz).
15	aclink_rst_on	Set this bit to 1 when you want to output a reset_b signal (internal signal) for the Codec. A reset_b signal will be automatically output for the Codec. This bit is automatically cleared to 0 after the reset_b signal is output.
7:0	aclink_rst_time(7:0)	Sets the active period of the reset_b signal that is output when 1 was set for the aclink_rst_on bit (count clock = 33 MHz).

Figure 15-1. SYNC Signal

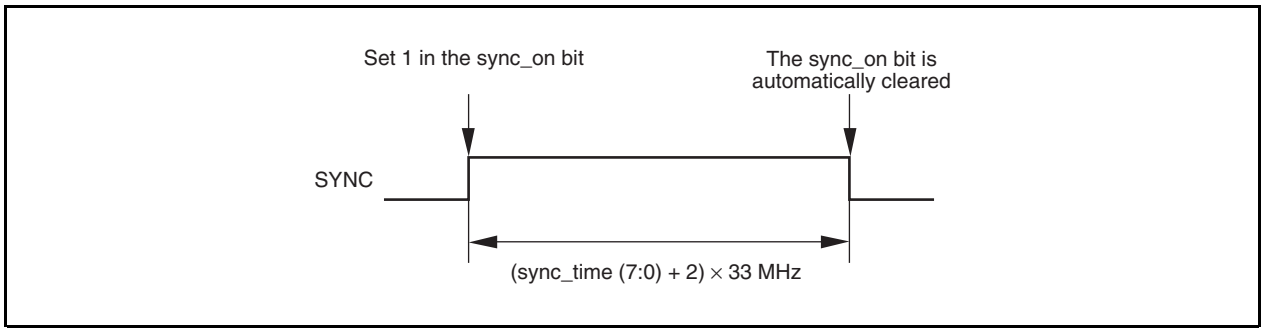
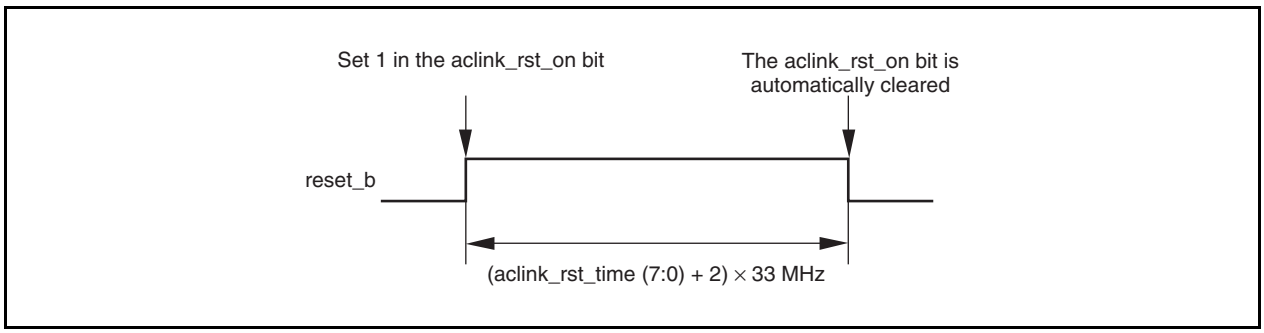


Figure 15-2. reset_b Signal (Internal Signal)



15.3.9 SRC_RAM_DATA (offset address: 0x20)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	SRC_RAM_DATA15	SRC_RAM_DATA14	SRC_RAM_DATA13	SRC_RAM_DATA12	SRC_RAM_DATA11	SRC_RAM_DATA10	SRC_RAM_DATA9	SRC_RAM_DATA8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	SRC_RAM_DATA7	SRC_RAM_DATA6	SRC_RAM_DATA5	SRC_RAM_DATA4	SRC_RAM_DATA3	SRC_RAM_DATA2	SRC_RAM_DATA1	SRC_RAM_DATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	SRC_RAM_DATA(15:0)	Sample rate converter RAM data

The RAM that was specified by the SRC_RAM_ADR bit of the CTRL register can be accessed by reading from or writing to this register.

However, this register cannot be written to during a DMA operation.

The RAM that is selected by the SRC_RAM_ADR bit is 16 bits × 32 levels. The 32-level RAM can be accessed by continuously reading from or writing to this register.

15.3.10 INT_MASK (offset address: 0x24)

(1/2)

Bit	31	30	29	28	27	26	25	24
Name	MMASK	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	MASK_ IOSTS	MASK_ STSDAT	MASK_ STSADR	MASK_ ACLINK_CK
R/W	R	R	R	R	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	MASK_ CODECGPI	MASK_ ACLINK	MASK_ DAC1END	MASK_ DAC2END	MASK_ DAC3END	MASK_ ADC1END	MASK_ ADC2END	MASK_ ADC3END
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	MMASK	Enables/disables mask interrupt 1: Enable 0: Disable
30:12	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
11	MASK_IOSTS	Enables/disables AC97 input data slot 12 valid data input interrupt 1: Enable 0: Disable
10	MASK_STSDAT	Enables/disables AC97 input data slot 2 valid data input interrupt 1: Enable 0: Disable
9	MASK_STSADR	Enables/disables AC97 input data slot 1 valid data input interrupt 1: Enable 0: Disable

(2/2)

Bit	Name	Function
8	MASK_ACLINK_CK	Enables/disables AC-Link clock request interrupt or enables/disables clock request interrupt from Codec side during a suspend state 1: Enable 0: Disable
7	MASK_CODECGPI	Enables/disables interrupt when 1 was set for the AC97 input data slot 12 bit 0 1: Enable 0: Disable
6	MASK_ACLINK	Enables/disables interrupt when a loopback transfer is performed and an error occurred 1: Enable 0: Disable
5	MASK_DAC1END	Enables/disables DAC1 DMA end interrupt 1: Enable 0: Disable
4	MASK_DAC2END	Enables/disables DAC2 DMA end interrupt 1: Enable 0: Disable
3	MASK_DAC3END	Enables/disables DAC3 DMA end interrupt 1: Enable 0: Disable
2	MASK_ADC1END	Enables/disables ADC1 DMA end interrupt 1: Enable 0: Disable
1	MASK_ADC2END	Enables/disables ADC2 DMA end interrupt 1: Enable 0: Disable
0	MASK_ADC3END	Enables/disables ADC3 DMA end interrupt 1: Enable 0: Disable

When an interrupt is set to disabled (the relevant bit is set to 0) in this register and an interrupt request is generated internally, the interrupt request is masked. When the relevant bit is set to 1, the interrupt request is reported to the external component.

15.3.11 DAC1_CTRL (offset address: 0x30)

Bit	31	30	29	28	27	26	25	24
Name	DAC1_ ENABLE	DAC1_ STATUS	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	DAC1_ENABLE	DAC1 DMA control 1: Enable 0: Disable
30	DAC1_STATUS	DAC1 AC-Link transfer status 1: Transfer in progress 0: Transfer ended
29:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.12 DAC1L (offset address: 0x34)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC1L15	DAC1L14	DAC1L13	DAC1L12	DAC1L11	DAC1L10	DAC1L9	DAC1L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC1L7	DAC1L6	DAC1L5	DAC1L4	DAC1L3	DAC1L2	DAC1L1	DAC1L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	DAC1L(15:0)	DAC1 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are input from memory.

  Also, the data size that is output at one time to the Codec is 16 bits of the data that was input from memory.

  Therefore, when 1 is set in the DAC1L(15:0) area, 32 bits × 4 data are input from memory and data is output 8 times to the Codec.

15.3.13 DAC1_BADDR (offset address: 0x38)

Bit	31	30	29	28	27	26	25	24
Name	DAC1_ BADDR31	DAC1_ BADDR30	DAC1_ BADDR29	DAC1_ BADDR28	DAC1_ BADDR27	DAC1_ BADDR26	DAC1_ BADDR25	DAC1_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	DAC1_ BADDR23	DAC1_ BADDR22	DAC1_ BADDR21	DAC1_ BADDR20	DAC1_ BADDR19	DAC1_ BADDR18	DAC1_ BADDR17	DAC1_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC1_ BADDR15	DAC1_ BADDR14	DAC1_ BADDR13	DAC1_ BADDR12	DAC1_ BADDR11	DAC1_ BADDR10	DAC1_ BADDR9	DAC1_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC1_ BADDR7	DAC1_ BADDR6	DAC1_ BADDR5	DAC1_ BADDR4	DAC1_ BADDR3	DAC1_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	DAC1_BADDR(31:2)	Sets the DAC1 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.14 DAC2_CTRL (offset address: 0x3C)

Bit	31	30	29	28	27	26	25	24
Name	DAC2_ ENABLE	DAC2_ STATUS	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	DAC2_ENABLE	DAC2 DMA control 1: Enable 0: Disable
30	DAC2_STATUS	DAC2 AC-Link transfer status 1: Transfer in progress 0: Transfer ended
29:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.15 DAC2L (offset address: 0x40)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC2L15	DAC2L14	DAC2L13	DAC2L12	DAC2L11	DAC2L10	DAC2L9	DAC2L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC2L7	DAC2L6	DAC2L5	DAC2L4	DAC2L3	DAC2L2	DAC2L1	DAC2L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	DAC2L(15:0)	DAC2 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are input from memory.

  Also, the data size that is output at one time to the Codec is 16 bits of the data that was input from memory.

  Therefore, when 1 is set in the DAC2L(15:0) area, 32 bits × 4 data are input from memory and data is output 8 times to the Codec.

15.3.16 DAC2_BADDR (offset address: 0x44)

Bit	31	30	29	28	27	26	25	24
Name	DAC2_ BADDR31	DAC2_ BADDR30	DAC2_ BADDR29	DAC2_ BADDR28	DAC2_ BADDR27	DAC2_ BADDR26	DAC2_ BADDR25	DAC2_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	DAC2_ BADDR23	DAC2_ BADDR22	DAC2_ BADDR21	DAC2_ BADDR20	DAC2_ BADDR19	DAC2_ BADDR18	DAC2_ BADDR17	DAC2_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC2_ BADDR15	DAC2_ BADDR14	DAC2_ BADDR13	DAC2_ BADDR12	DAC2_ BADDR11	DAC2_ BADDR10	DAC2_ BADDR9	DAC2_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC2_ BADDR7	DAC2_ BADDR6	DAC2_ BADDR5	DAC2_ BADDR4	DAC2_ BADDR3	DAC2_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	DAC2_BADDR(31:2)	Sets the DAC2 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.17 DAC3_CTRL (offset address: 0x48)

Bit	31	30	29	28	27	26	25	24
Name	DAC3_ ENABLE	DAC3_ STATUS	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	DAC3_ENABLE	DAC3 DMA control 1: Enable 0: Disable
30	DAC3_STATUS	DAC3 AC-Link transfer status 1: Transfer in progress 0: Transfer ended
29:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.18 DAC3L (offset address: 0x4C)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC3L15	DAC3L14	DAC3L13	DAC3L12	DAC3L11	DAC3L10	DAC3L9	DAC3L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC3L7	DAC3L6	DAC3L5	DAC3L4	DAC3L3	DAC3L2	DAC3L1	DAC3L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	DAC3L(15:0)	DAC3 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are input from memory.

  Also, the data size that is output at one time to the Codec is 16 bits of the data that was input from memory.

  Therefore, when 1 is set in the DAC3L(15:0) area, 32 bits × 4 data are input from memory and data is output 8 times to the Codec.

15.3.19 DAC3_BADDR (offset address: 0x50)

Bit	31	30	29	28	27	26	25	24
Name	DAC3_ BADDR31	DAC3_ BADDR30	DAC3_ BADDR29	DAC3_ BADDR28	DAC3_ BADDR27	DAC3_ BADDR26	DAC3_ BADDR25	DAC3_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	DAC3_ BADDR23	DAC3_ BADDR22	DAC3_ BADDR21	DAC3_ BADDR20	DAC3_ BADDR19	DAC3_ BADDR18	DAC3_ BADDR17	DAC3_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	DAC3_ BADDR15	DAC3_ BADDR14	DAC3_ BADDR13	DAC3_ BADDR12	DAC3_ BADDR11	DAC3_ BADDR10	DAC3_ BADDR9	DAC3_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	DAC3_ BADDR7	DAC3_ BADDR6	DAC3_ BADDR5	DAC3_ BADDR4	DAC3_ BADDR3	DAC3_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	DAC3_BADDR(31:2)	Sets the DAC3 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.20 ADC1_CTRL (offset address: 0x54)

Bit	31	30	29	28	27	26	25	24
Name	ADC1_ ENABLE	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	ADC1_ENABLE	ADC1 DMA control 1: Enable 0: Disable
30:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.21 ADC1L (offset address: 0x58)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC1L15	ADC1L14	ADC1L13	ADC1L12	ADC1L11	ADC1L10	ADC1L9	ADC1L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ADC1L7	ADC1L6	ADC1L5	ADC1L4	ADC1L3	ADC1L2	ADC1L1	ADC1L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	ADC1L(15:0)	ADC1 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are output to memory.

  Also, the data size that is input at one time from the Codec is 16 bits of the data that is output to memory.

  Therefore, when 1 is set in the ADC1L(15:0) area, data is input 8 times from the Codec and 32 bits × 4 data are output to memory.

15.3.22 ADC1_BADDR (offset address: 0x5C)

Bit	31	30	29	28	27	26	25	24
Name	ADC1_ BADDR31	ADC1_ BADDR30	ADC1_ BADDR29	ADC1_ BADDR28	ADC1_ BADDR27	ADC1_ BADDR26	ADC1_ BADDR25	ADC1_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	ADC1_ BADDR23	ADC1_ BADDR22	ADC1_ BADDR21	ADC1_ BADDR20	ADC1_ BADDR19	ADC1_ BADDR18	ADC1_ BADDR17	ADC1_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC1_ BADDR15	ADC1_ BADDR14	ADC1_ BADDR13	ADC1_ BADDR12	ADC1_ BADDR11	ADC1_ BADDR10	ADC1_ BADDR9	ADC1_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ADC1_ BADDR7	ADC1_ BADDR6	ADC1_ BADDR5	ADC1_ BADDR4	ADC1_ BADDR3	ADC1_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	ADC1_BADDR(31:2)	Sets the ADC1 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.23 ADC2_CTRL (offset address: 0x60)

Bit	31	30	29	28	27	26	25	24
Name	ADC2_ ENABLE	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	ADC2_ENABLE	ADC2 DMA control 1: Enable 0: Disable
30:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.24 ADC2L (offset address: 0x64)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC2L15	ADC2L14	ADC2L13	ADC2L12	ADC2L11	ADC2L10	ADC2L9	ADC2L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ADC2L7	ADC2L6	ADC2L5	ADC2L4	ADC2L3	ADC2L2	ADC2L1	ADC2L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	ADC2L(15:0)	ADC2 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are output to memory.

  Also, the data size that is input at one time from the Codec is 16 bits of the data that is output to memory.

  Therefore, when 1 is set in the ADC2L(15:0) area, data is input 8 times from the Codec and 32 bits × 4 data are output to memory.

15.3.25 ADC2_BADDR (offset address: 0x68)

Bit	31	30	29	28	27	26	25	24
Name	ADC2_ BADDR31	ADC2_ BADDR30	ADC2_ BADDR29	ADC2_ BADDR28	ADC2_ BADDR27	ADC2_ BADDR26	ADC2_ BADDR25	ADC2_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	ADC2_ BADDR23	ADC2_ BADDR22	ADC2_ BADDR21	ADC2_ BADDR20	ADC2_ BADDR19	ADC2_ BADDR18	ADC2_ BADDR17	ADC2_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC2_ BADDR15	ADC2_ BADDR14	ADC2_ BADDR13	ADC2_ BADDR12	ADC2_ BADDR11	ADC2_ BADDR10	ADC2_ BADDR9	ADC2_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ADC2_ BADDR7	ADC2_ BADDR6	ADC2_ BADDR5	ADC2_ BADDR4	ADC2_ BADDR3	ADC2_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	ADC2_BADDR(31:2)	Sets the ADC2 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.26 ADC3_CTRL (offset address: 0x6C)

Bit	31	30	29	28	27	26	25	24
Name	ADC3_ ENABLE	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R/W	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31	ADC3_ENABLE	ADC3 DMA control 1: Enable 0: Disable
30:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.3.27 ADC3L (offset address: 0x70)

Bit	31	30	29	28	27	26	25	24
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	RFU	RFU	RFU	RFU	RFU	RFU	RFU	RFU
R/W	R	R	R	R	R	R	R	R
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC3L15	ADC3L14	ADC3L13	ADC3L12	ADC3L11	ADC3L10	ADC3L9	ADC3L8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Name	ADC3L7	ADC3L6	ADC3L5	ADC3L4	ADC3L3	ADC3L2	ADC3L1	ADC3L0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:16	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.
15:0	ADC3L(15:0)	ADC3 DMA transfer count

- **DMA transfer count**

  For a single DMA transfer, 32 bits × 4 data are output to memory.

  Also, the data size that is input at one time from the Codec is 16 bits of the data that is output to memory.

  Therefore, when 1 is set in the ADC3L(15:0) area, data is input 8 times from the Codec and 32 bits × 4 data are output to memory.

15.3.28 ADC3_BADDR (offset address: 0x74)

Bit	31	30	29	28	27	26	25	24
Name	ADC3_ BADDR31	ADC3_ BADDR30	ADC3_ BADDR29	ADC3_ BADDR28	ADC3_ BADDR27	ADC3_ BADDR26	ADC3_ BADDR25	ADC3_ BADDR24
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	23	22	21	20	19	18	17	16
Name	ADC3_ BADDR23	ADC3_ BADDR22	ADC3_ BADDR21	ADC3_ BADDR20	ADC3_ BADDR19	ADC3_ BADDR18	ADC3_ BADDR17	ADC3_ BADDR16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

Bit	15	14	13	12	11	10	9	8
Name	ADC3_ BADDR15	ADC3_ BADDR14	ADC3_ BADDR13	ADC3_ BADDR12	ADC3_ BADDR11	ADC3_ BADDR10	ADC3_ BADDR9	ADC3_ BADDR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
After reset	0	0	0	0	0	0	0	0

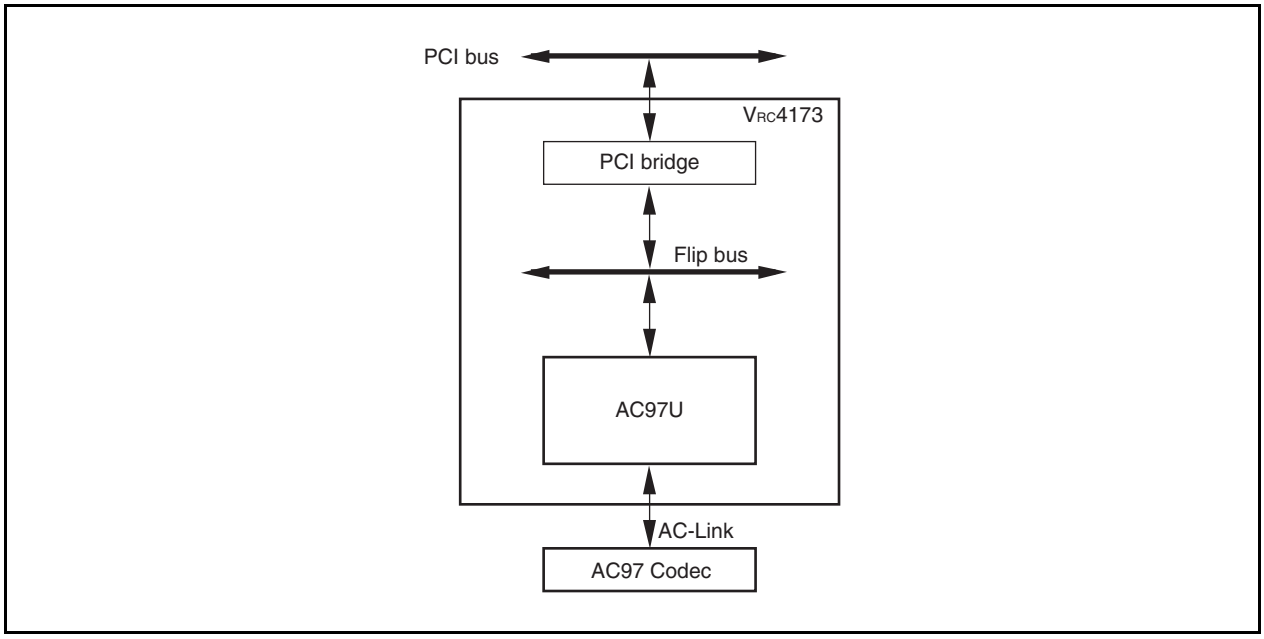
Bit	7	6	5	4	3	2	1	0
Name	ADC3_ BADDR7	ADC3_ BADDR6	ADC3_ BADDR5	ADC3_ BADDR4	ADC3_ BADDR3	ADC3_ BADDR2	RFU	RFU
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
After reset	0	0	0	0	0	0	0	0

Bit	Name	Function
31:2	ADC3_BADDR(31:2)	Sets the ADC3 DMA base address
1:0	RFU	Reserved. Write 0 to these bits. 0 is returned after a read.

15.4 AC97 Interface Configuration

Figure 15-3 shows the AC97 interface configuration.

Figure 15-3. AC97 Interface Configuration

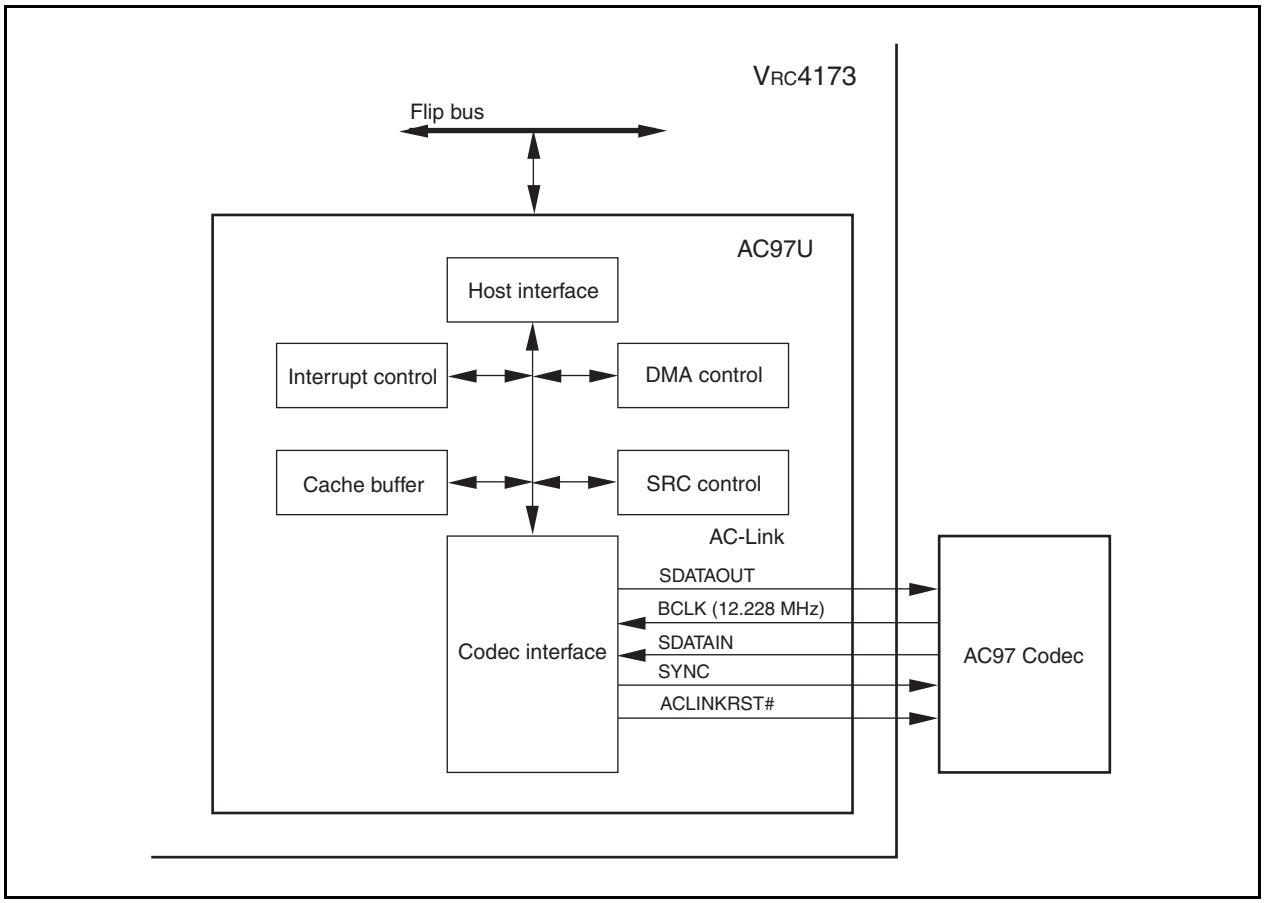


15.5 AC97U Function Overview

15.5.1 Block diagram

Figure 15-4 shows a block diagram of the AC97U.

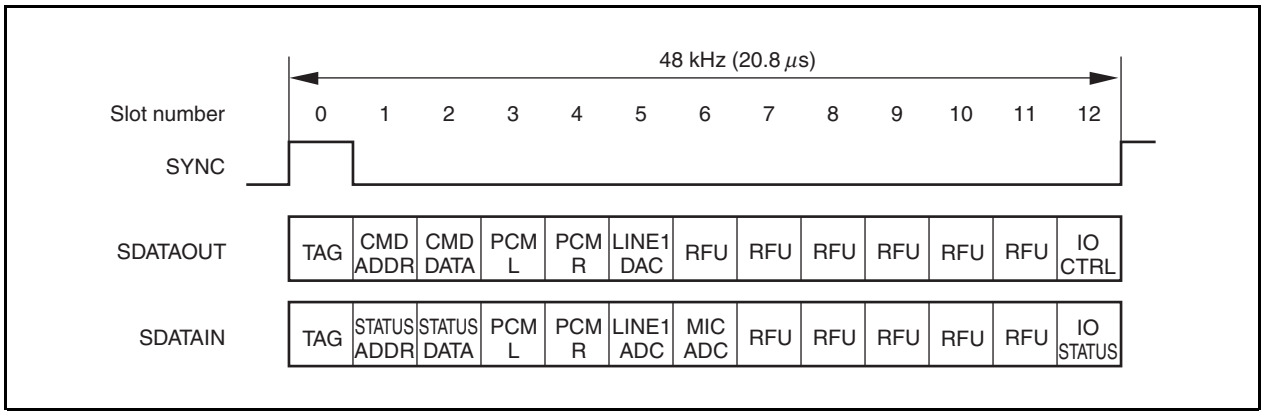
Figure 15-4. AC97U Block Diagram



15.5.2 AC-Link interface support format

The AC97U only supports the slots shown in Figure 15-5 among those in the AC97 guidelines.

★ **Figure 15-5. AC97U-Supported Slots**



For details about the data transfer format, see **15.6 AC-Link Interface Data Transfer Format**.

15.5.3 Cache buffer

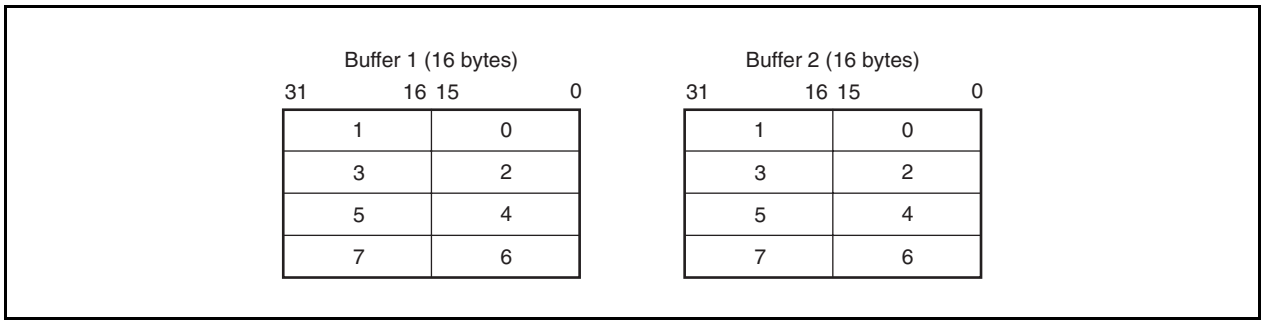
The AC97U has a cache buffer for transferring audio data or modem data. This buffer has a total of six blocks. The DAC1, DAC2, and DAC3 blocks are for output, and the ADC1, ADC2, and ADC3 blocks are for input.

Name	Function
DAC1	PCM L (16 bytes)
DAC2	PCM R (16 bytes)
DAC3	Line 1 (16 bytes)
ADC1	PCM L or MIC (16 bytes)
ADC2	PCM R (16 bytes)
ADC3	Line 1 (16 bytes)

(1) Buffer format

Two 16-byte buffers (buffer 1 and buffer 2) are allocated for each block.

Figure 15-6. Buffer Format



(2) Data transfer

On the PCI bus, 32-bit (word) data transfers are performed in units of 4 words. The first transfer after transfers are enabled is a 4-word burst transfer performed two times consecutively to fill the double buffer. Subsequently, the buffers are filled alternately by requesting the next data whenever one buffer becomes empty.

Figure 15-7. Data Transfer (Buffer → AC-Link)

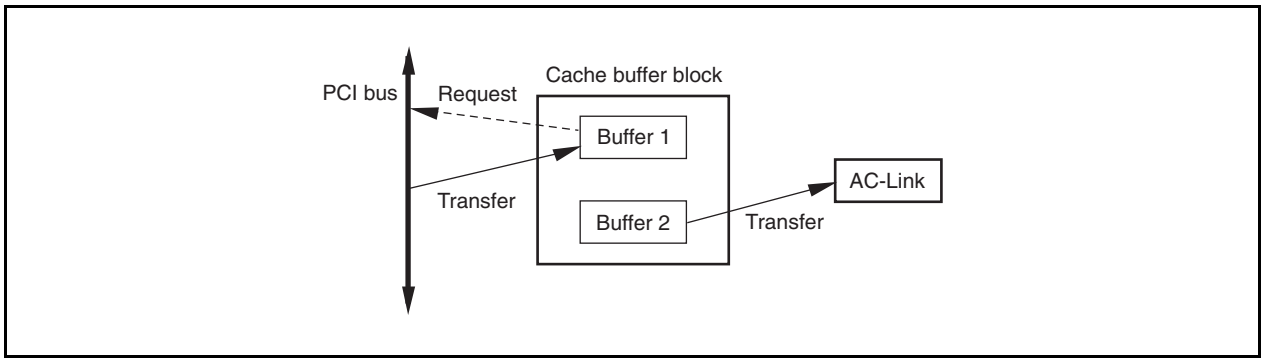
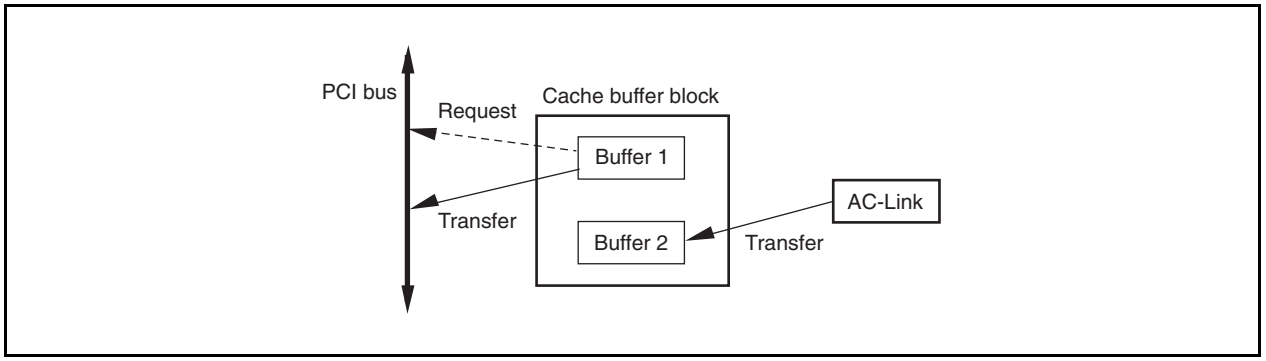


Figure 15-8. Data Transfer (AC-Link → Buffer)



15.5.4 DMA control

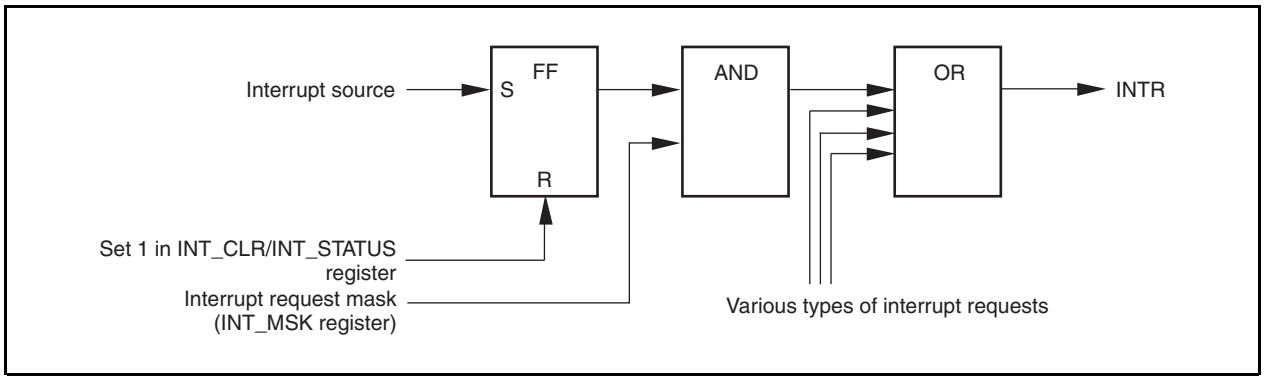
The following control registers are used for DMA transfers in terms of individual DAC1, DAC2, DAC3, ADC1, ADC2, and ADC3 cache buffers.

- Memory base address/data length register
- Transfer control register (enable, status)

15.5.5 Interrupt control

Figure 15-9 shows interrupt control.

Figure 15-9. Interrupt Control



15.5.6 SRC (sample rate converter)

This section explains the converter and filter functions.

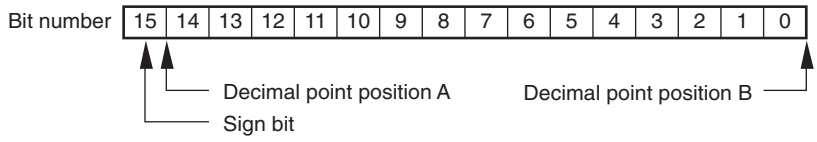
Since the AC97U handles all audio I/O data by using 48 kHz samples, rate conversion must be performed for data having various sample conditions. The converter function performs this rate conversion.

However, when a rate conversion is simply performed from PCM data, distortion occurs in the output analog waveform. Therefore, the filter function is used to restore the waveform.

(1) Input data format

- 16-bit data
- Bit 15 is a sign bit
- When the sign bit is 1, set two's-complement data for the data part (bits 14 to 0).
- The decimal point position can be selected. However, the same position is set for all data (fixed decimal point).

Figure 15-10. Input Data



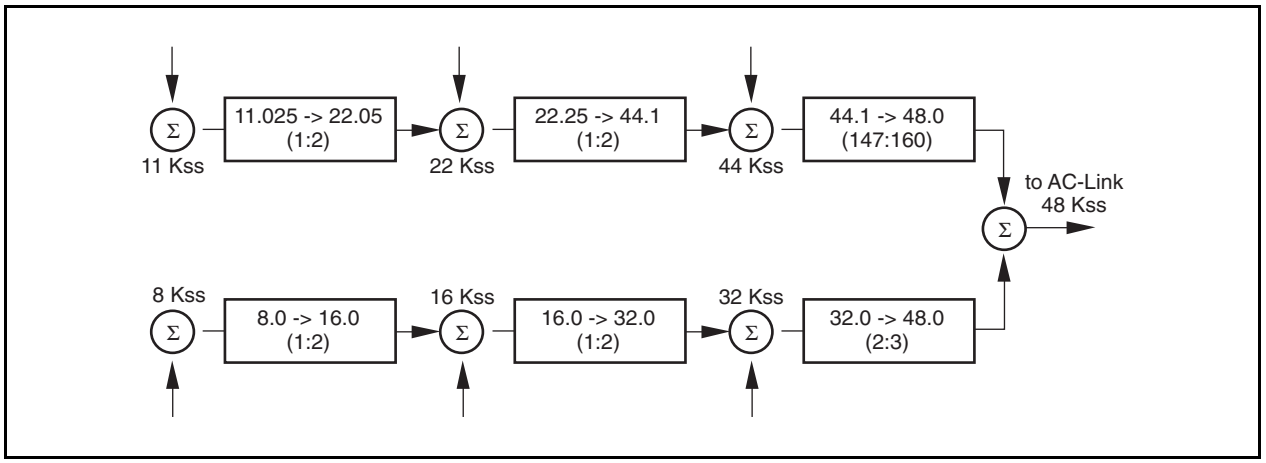
Remark Coefficient range when the decimal point position is A: $-1 < X < 1$
Coefficient range when the decimal point position is B: $-32767 < X < 32767$

(2) Converter function

Figure 15-11 shows the converter function when data is output.

When data is input, 48 Kss data is input from the AC-Link and converted to data of various rates in the reverse of the process shown for data output.

Figure 15-11. Converter Function (for Output)



(a) Output data conversion

- For 1:2 data
  Data B, A → B, (B+A)/2, A
- For 2:3 data
  Data C, B, A → C, (C+B)/2, (B+A)/2, A
- For 147:160 data
  Prepare 147 counters and insert on average 13 values by taking the averages of 13 pairs of consecutive data among them.

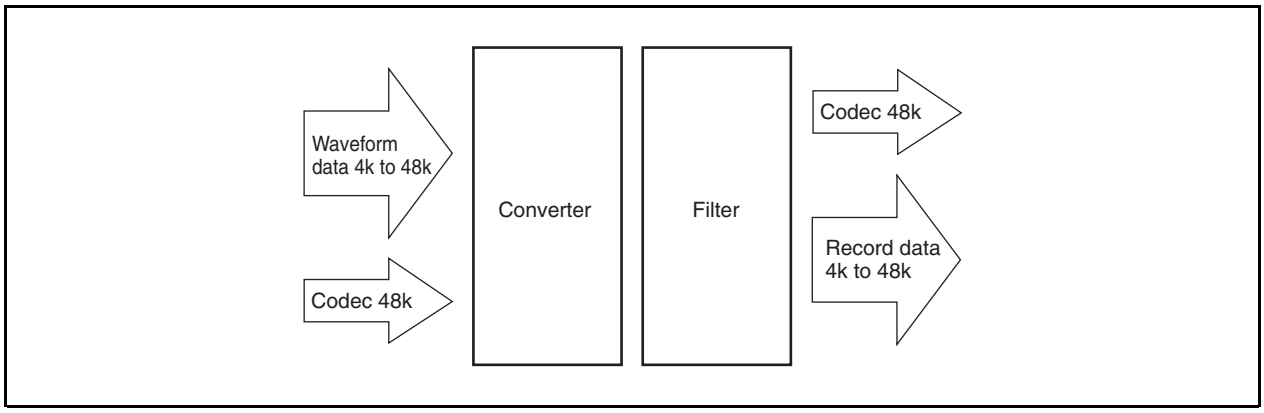
(b) Input data conversion

- For 1:2 data
  Data C, B, A → C, A
- For 2:3 data
  Data D, C, B, A → D, C, A
- For 147:160 data
  Prepare 160 counters and delete on average 13 values among them.

(3) Filter function

A 32-order FIR filter is used as the filter function.

Figure 15-12. Filter Function



The output equation is as follows.

$y(x\_0) = (a\_-31 \times x\_-31) + (a\_-30 \times x\_-30) +...+ (a\_0 \times x\_0)$

Remarks 1. y: Output signal
x: Input signal
a: Coefficient

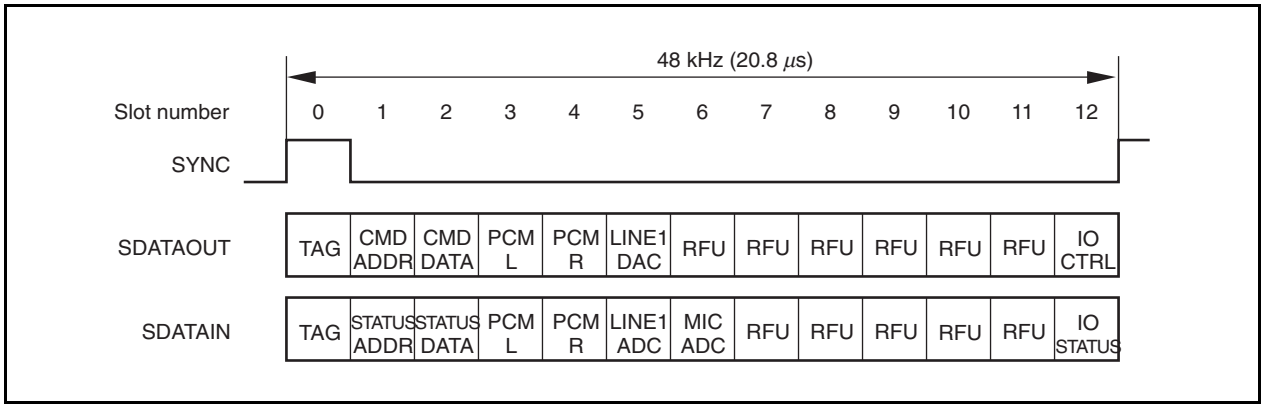
★ **2.** The "_–31" of x_–31 indicates the input signal that appeared 31 signals prior to the input signal (x_0) to be input next. The prior position (how many signals earlier) of the signal to be used differs according to the sample rate (8k, 16k, 32k, 11k, 22k, or 44k).

For input-signal 8 kHz sample data (system → Codec direction), to convert to a 48 kHz sample, the signal with an 8 kHz period (input signal) is converted to 6 times the sample rate (48 kHz) (signal expansion) and output.
For input-signal 48 kHz sample data (system ← Codec direction), to convert to an 8 kHz sample, the signal with a 48 kHz period (input signal) is converted to 1/6 times the sample rate (8 kHz) (signal compression) and output.
However, when the filter function is used together with the converter function, this processing is not performed because the input signal has been converted to the sample rate.
To use the filter function for a DMA transfer, the coefficients for the calculation must be set in the filter RAM in advance. For details about the filter RAM, see **15.12 Filter RAM**.

15.6 AC-Link Interface Data Transfer Format

Figure 15-13 shows the AC-Link interface data transfer format.

Figure 15-13. AC-Link Interface Data Transfer Format



- **SDATAOUT**

  For slots 1, 2, and 12, data is output by performing write processing in the operational registers.

  For slots 3, 4, and 5, data is output by starting DMAs for DAC1, DAC2, and DAC3, respectively.

- **SDATAIN**

  For slots 1, 2, and 12, data is input by generating an interrupt request when a valid data is input and performing read processing for the operational registers.

  For slots 3, 4, 5, and 6, data is input by starting DMAs for ADC1, ADC2, ADC3, and ADC1 respectively.

(1) SDATAOUT slot 0: TAG

Bit	Function
15	Valid Frame When any of bits 14 to 10 or bit 3 is 1, 1 is set for this bit.
14	Slot1 Valid bit When a write operation is performed for the CODEC_WR register, 1 is set for this bit.
13	Slot2 Valid bit When a write operation is performed for the CODEC_WR register, 1 is set for this bit.
12	Slot3 Valid bit When a DMA is started for DAC1, 1 is set for this bit.
11	Slot4 Valid bit When a DMA is started for DAC2, 1 is set for this bit.
10	Slot5 Valid bit When a DMA is started for DAC3, 1 is set for this bit.
9	Slot6 Valid bit (fixed at 0).
8	Slot7 Valid bit (fixed at 0).
7	Slot8 Valid bit (fixed at 0).
6	Slot9 Valid bit (fixed at 0).
5	Slot10 Valid bit (fixed at 0).
4	Slot11 Valid bit (fixed at 0).
3	Slot12 Valid bit When a write operation is performed for the CODEC_RD register, 1 is set for this bit.
2:0	These bits are fixed at 0.

(2) SDATAOUT slot 1: CMDADDR (Command Address Port)

Bit	Function
19	Read/Write Command (1: Read, 0: Write) Outputs the value that was set in the RWC bit of the CODEC_WR register.
18:12	Control Register Index Outputs the value that was set in the WADDR(6:0) area of the CODEC_WR register.
11:0	These bits are fixed at 0.

★

(3) SDATAOUT slot 2: CMDDATA (Command Data Port)

Bit	Function
19:4	Control Register Write Data Outputs the value that was set in the WDAT(15:0) area of the CODEC_WR register.
3:0	These bits are fixed at 0.

(4) SDATAOUT slot 3: PCML (PCM Playback Left Channel)

Bit	Function
19:4	PCM Playback Left Data Outputs the result of the calculation (filter or rate conversion) that was performed on the data that was input from memory due to a DMA for DAC1.
3:0	These bits are fixed at 0.

(5) SDATAOUT slot 4: PCMR (PCM Playback Right Channel)

Bit	Function
19:4	PCM Playback Right Data Outputs the result of the calculation (filter or rate conversion) that was performed on the data that was input from memory due to a DMA for DAC2.
3:0	These bits are fixed at 0.

(6) SDATAOUT slot 5: LINE1DAC (Optional Modem Line1 DAC)

Bit	Function
19:4	Optional Modem Line1 DAC Data Outputs the data that was input from memory due to a DMA for DAC3.
3:0	These bits are fixed at 0.

(7) SDATAOUT slot 12: IOCTRL (Optional Modem GPIO Control)

Bit	Function
19:0	up to 16 GPIO pins Outputs the value that was set in the WSLOT12(19:0) bits of the SLOT12_WR register.

(8) SDATAIN slot 0: TAG

Bit	Function
15	Valid Frame When any of bits 14 to 10 or bit 3 is 1, this bit becomes 1.
14	Slot1 Valid bit When 1 is entered for this bit, an interrupt request is generated.
13	Slot2 Valid bit When 1 is entered for this bit, an interrupt request is generated.
12	Slot3 Valid bit When 1 is entered for this bit, a DMA is started for ADC1.
11	Slot4 Valid bit When 1 is entered for this bit, a DMA is started for ADC2.
10	Slot5 Valid bit When 1 is entered for this bit, a DMA is started for ADC3.
9	Slot6 Valid bit When 1 is entered for this bit, a DMA is started for ADC1.
8:4	Unused
3	Slot12 Valid bit When 1 is entered for this bit, an interrupt request is generated.
2:0	Unused

(9) SDATAIN slot 1: STATUSADDR (Status Address Port)

Bit	Function
19	Unused
18:12	Control Register Index The value of this area is displayed in the RADDR(6:0) area of the CODEC_RD register.
11	Slot3 Request: PCM Left Channel The value of this bit is displayed in the SLOT3_REQ bit of the CODEC_REQ register.
10	Slot4 Request: PCM Right Channel The value of this bit is displayed in the SLOT4_REQ bit of the CODEC_REQ register.
9	Slot5 Request: Modem Line1 The value of this bit is displayed in the SLOT5_REQ bit of the CODEC_REQ register.
8	Slot6 Request: PCM Center The value of this bit is displayed in the SLOT6_REQ bit of the CODEC_REQ register.
7	Slot7 Request: PCM Left surround The value of this bit is displayed in the SLOT7_REQ bit of the CODEC_REQ register.
6	Slot8 Request: PCM Right surround The value of this bit is displayed in the SLOT8_REQ bit of the CODEC_REQ register.
5	Slot9 Request: PCM LFE The value of this bit is displayed in the SLOT9_REQ bit of the CODEC_REQ register.
4	Slot10 Request: Modem Line2 or PCM Left(n+1) The value of this bit is displayed in the SLOT10_REQ bit of the CODEC_REQ register.
3	Slot11 Request: Handset or PCM Right(n+1) The value of this bit is displayed in the SLOT11_REQ bit of the CODEC_REQ register.
2	Slot12 Request: PCM Center(n+1) The value of this bit is displayed in the SLOT12_REQ bit of the CODEC_REQ register.
1:0	Unused

(10) SDATAIN slot 2: STATUSDATA (Status Data Port)

Bit	Function
19:4	Control Register Read Data The value of this area is displayed in the RDAT(15:0) area of the CODEC_RD register.
3:0	Unused

(11) SDATAIN slot 3: PCML (PCM Record Left Channel)

Bit	Function
19:4	PCM Record Left Data This area is output to memory due to a DMA for ADC1.
3:0	Unused

(12) SDATAIN slot 4: PCMR (PCM Record Right Channel)

Bit	Function
19:4	PCM Record Right Data This area is output to memory due to a DMA for ADC2.
3:0	Unused

(13) SDATAIN slot 5: LINE1ADC (Optional Modem Line1 ADC)

Bit	Function
19:4	Optional Modem Line1 ADC Data This area is output to memory due to a DMA for ADC3.
3:0	Unused

(14) SDATAIN slot 6: MICADC (Optional Dedicated Microphone Record Data)

Bit	Function
19:4	Optional Dedicated Microphone Record Data This area is output to memory due to a DMA for ADC1.
3:0	Unused

(15) SDATAIN slot 12: IOSTATUS (Optional Modem GPIO Status)

Bit	Function
19:0	up to 16 GPIO pins The value of this area is displayed in the RSLOT12(19:0) area of the SLOT12_RD register. When 1 is entered in bit 0 (GPIO_INT enabled input pin event interrupt), an interrupt request is generated.

15.7 Data Output to Codec

(1) Data output to slot 1 or 2

The following steps <1> and <2> are repeated due to data output to slot 1 or 2.

<1> Confirm that the WRDY bit of the CODEC_WR register in the operational registers is 0 (When it is 1, writing is enabled for slots 1 and 2).

<2> Write the data to be output by using slot 1 or 2 to the WDAT(15:0) area of the CODEC_WR register in the operational registers.

(2) Data output to slot 12

The following steps <1> and <2> are repeated due to data output to slot 12.

<1> Confirm that the WRDY_SLOT bit of the SLOT12_WR register in the operational registers is 0 (When it is 1, writing is enabled for slot 12).

<2> Write the data to be output by using slot 12 to the WSLOT12(19:0) area of the SLOT12_WR register in the operational registers.

(3) Data output to slot 3, 4, or 5

A DMA to the following cache buffers is started for data output to slot 3, 4, or 5.

- Slot 3: DAC1
- Slot 4: DAC2
- Slot 5: DAC3

15.8 Data Input from Codec

(1) Slot 1 data input

Data is read due to slot 1 data input by repeating the following steps (a) and (b).

(a) Read data according to an interrupt request

<1> Valid data is input in slot 1 and an interrupt request is generated (the STSADR bit of the INT_CLR/INT_STATUS register in the operational registers becomes 1).

<2> Read the RADDR(6:0) area of the CODEC_RD register in the operational registers.

<3> Read bits 9 to 0 of the CODEC_REQ register in the operational registers.

(b) Read periodic data

<1> Periodically read the CODEC_RD register in the operational registers (when the RRDYA bit is 1, the RADDR(6:0) area is valid).

<2> Read bits 9 to 0 of the CODEC_REQ register in the operational registers.

(2) Slot 2 data input

Data is read due to slot 2 data input by repeating the following steps (a) and (b).

(a) Read data according to an interrupt request

<1> Valid data is input in slot 2 and an interrupt request is generated (the STSDAT bit of the INT_CLR/INT_STATUS register in the operational registers becomes 1).

<2> Read the RDAT(15:0) area of the CODEC_RD register in the operational registers.

(b) Read periodic data

<1> Periodically read the CODEC_RD register in the operational registers (when the RRDYD bit is 1, the RDAT(15:0) area is valid).

Remark The WIP bit of the CODEC_RD register in the operational registers counts the number of times 1 is set (status read) in the RWC bit of the CODEC_WR register in the operational registers. When valid data is input in the CODEC_RD register, this count value is set to −1. When this count value is not zero, the WIP bit becomes 1 (read processing for the CODEC remains). When the count value is 0, the WIP bit becomes 0 (no read processing for the CODEC remains).

(3) Slot 12 data input

Data is read due to slot 12 data input by repeating the following steps (a) and (b).

(a) Read data according to an interrupt request

<1> Valid data is input in slot 12 and an interrupt request is generated (the IOSTS bit of the INT_CLR/INT_STATUS register in the operational registers becomes 1).

<2> Read the RSLOT12(19:0) area of the SLOT12_RD register in the operational registers.

(b) Read periodic data

<1> Periodically read the SLOT12_RD register in the operational registers (when the RRDY_SLOT bit is 1, the RSLOT12(19:0) area is valid).

(4) Slot 3, 4, 5, or 6 data input

A DMA to the following cache buffers is started for slot 3, 4, 5, or 6 data input.

- Slot 3: ADC1[Note]
- Slot 4: ADC2
- Slot 5: ADC3
- Slot 6: ADC1[Note]

Note The DMAs of the slots 3 and 6 cannot be operated at the same time.

15.9 DMA Transfer

(1) Data output to the Codec (slot 3, 4, or 5)

The method of outputting data to the Codec (slot 3, 4, or 5) is shown below.

<1> Set the base address of the memory to be accessed and the transfer count ((32 bits × 4)/transfer) in the operational registers.

Slot	Base Address Setting Register	Transfer Count	
		Register	Area
3	DAC1_BADDR	DAC1L	DAC1L(15:0)
4	DAC2_BADDR	DAC2L	DAC2L(15:0)
5	DAC3_BADDR	DAC3L	DAC3L(15:0)

★

<2> The DMA operation is started by setting 1 in the DMA start bit in the operational registers.

Slot	Start Bit	
	Register	Bit
3	DAC1_CTRL	DAC1_ENABLE
4	DAC2_CTRL	DAC2_ENABLE
5	DAC3_CTRL	DAC3_ENABLE

<3> Set the following items in the CTRL register in the operational registers to start the transfers with the Codec.

- Whether or not to perform converter or filter operation
- Conversion rate of data to be output
- Enabling of transfers with the Codec

Slot	Converter Operation	Filter Operation	Conversion Rate	Enabling of Codec Transfer
3	SRC_CNVT_ON bit	SRC_FILTER_ON bit	DAC1FORM(2:0) area	DAC1ENB bit
4				DAC2ENB bit
5	–	–	–	DAC3ENB bit

<4> To continue to perform DMA operations, set the next transfer base address and transfer count.

★ <5> When the DMAs have executed the specified number of transfers, an interrupt request is generated and the DMA start bit is automatically cleared to 0.

Slot	Interrupt Status Bit	
	Register	Bit
3	INT_CLR/INT_STATUS	DAC1END
4		DAC2END
5		DAC3END

The DMA automatically loads the base address and transfer count and executes the transfer.
To end the DMA transfers forcibly, set 0 in the start bit in step <2>.
Also, if the next status bit becomes 0, set the Codec transfer enable bit in step <3> to 0.

Slot	Status Bit	
	Register	Bit
3	DAC1_CTRL	DAC1_STATUS
4	DAC2_CTRL	DAC2_STATUS
5	DAC3_CTRL	DAC3_STATUS

- The counters for the base address and transfer count are separate from the registers. The register values are loaded in the counters only when the start bit is set to 1 and the DMA end interrupt request is generated. Therefore, if the next base address and transfer count are set in advance after the start bit was set to 1 or the DMA end interrupt request was generated, DMA operations can be performed continuously by loading the setting values after the DMA that is currently being processed ends.

- The correspondence between the DMA transfer count and the Codec transfer count differs according to the following bit settings of the CTRL register in the operational registers.

  - SRC_CNVT_ON bit: Whether or not to perform converter operation
  - SRC_FILTER_ON bit: Whether or not to perform filter operation
  - DAC1FORM(2:0) area: Conversion rate of data to be output

The data length that is transferred to the buffer within the AC97U by a single DMA is 32 bits × 4 (= 16 bits × 8). The data that is used in a single transfer with the Codec is 16 bits among these bits.
The following table shows the relationship between the CTRL register settings and data transfer counts.

SRC_CNVT_ON bit, SRC_FILTER_ON bit	DAC1FORM (2:0)	DMA Transfer Count	DAC1 or DAC2 Codec Transfer Count	DAC3 Codec Transfer Count
SRC_CNVT_ON = 0 and SRC_FILTER_ON = 0	–	A (Arbitrary)	A × 8	A × 8
★ Other combination than above	000		A × 8	
	001		(A × 48) – 5	
	010		(A × 24) – 2	
	011		(A × 12) – 1	
	100		((A × 32) – 3) × 147/160	
	101		((A × 16) – 1) × 147/160	
	110		(A × 8) × 147/160	

- The address shown below is output as the DMA address.

  DMA address = (Address set in the operational register) + (0x10 × N)
  N: DMA transfer count (0, 1, 2, 3,...)

  The higher 12 bits (bits 31 to 20) are fixed, and the lower 18 bits (bits 19 to 2) vary. Therefore, if a carry occurs in the lower 18 bits, 1 is not added to the higher 12 bits.

(2) Data input from the Codec (slot 3, 4, 5, or 6)

The method of inputting data from the Codec (slot 3, 4, 5, or 6) is shown below.

<1> Set the base address of the memory to be accessed and the transfer count ((32 bits × 4)/transfer) in the operational registers.

Slot	Base Address Setting Register	Transfer Count	
		Register	Area
3, 6	ADC1_BADDR	ADC1L	ADC1L(15:0)
4	ADC2_BADDR	ADC2L	ADC2L(15:0)
5	ADC3_BADDR	ADC3L	ADC3L(15:0)

<2> The DMA operation is started by setting 1 in the DMA start bit in the operational registers.

Slot	Start Bit	
	Register	Bit
3, 6	ADC1_CTRL	ADC1_ENABLE
4	ADC2_CTRL	ADC2_ENABLE
5	ADC3_CTRL	ADC3_ENABLE

<3> Set the following items in the CTRL register in the operational registers.

- Whether or not to perform converter or filter operation
- Conversion rate of data to be input
- Enabling of transfers with the Codec

Slot	Converter Operation	Filter Operation	Conversion Rate	Enabling of Codec Transfer
3	SRC_CNVT_ON bit	SRC_FILTER_ON bit	ADC1FORM(2:0) area	ADC1ENB bit[Note]
4				ADC2ENB bit
5	–	–	–	ADC3ENB bit
6	SRC_CNVT_ON bit	SRC_FILTER_ON bit	ADC1FORM(2:0) area	MICENB bit[Note]

Note As the DMAs of the slots 3 and 6 cannot be operated simultaneously, do not set these bits to 1 at the same time.

<4> To continue to perform DMA operations, set the next transfer base address and transfer count.

<5> When the DMAs end, an interrupt request is generated and the DMA start bit is automatically cleared to 0.

Slot	Interrupt Bit	
	Register	Bit
3, 6	INT_CLR/INT_STATUS	ADC1END
4		ADC2END
5		ADC3END

The DMA automatically loads the base address and transfer count and executes the transfer.
To end the DMA transfers forcibly, set 0 in the start bit in step <2>.
Also, set the Codec transfer enable bit in step <3> to 0.

- The counters for the base address and transfer count are separate from the registers. The register values are loaded in the counters only when the start bit is set to 1 and the DMA end interrupt request is generated. Therefore, if the next base address and transfer count are set in advance after the start bit was set to 1 or the DMA end interrupt request was generated, DMA operations can be performed continuously by loading the setting values after the DMA that is currently being processed ends.

- The correspondence between the DMA transfer count and the Codec transfer count differs according to the following bit settings of the CTRL register in the operational registers.

  - SRC_CNVT_ON bit: Whether or not to perform converter operation
  - SRC_FILTER_ON bit: Whether or not to perform filter operation
  - ADC1FORM(2:0) area: Conversion rate of data to be input

  The data length that is transferred from the buffer within the AC97U by a single DMA is 32 bits × 4 (= 16 bits × 8). The data that is used in a single transfer with the Codec is 16 bits among these bits.
  The following table shows the relationship between the CTRL register settings and data transfer counts.

SRC_CNVT_ON bit, SRC_FILTER_ON bit	ADC1FORM (2:0)	DMA Transfer Count	ADC1 or ADC2 Codec Transfer Count	ADC3 Codec Transfer Count
SRC_CNVT_ON = 0 and SRC_FILTER_ON = 0	–	A (Arbitrary)	A × 8	A × 8
★ Other combination than above	000		A × 8	
	001		(A × 48) – 5	
	010		(A × 24) – 2	
	011		(A × 12) – 1	
	100		((A × 32) – 3) × 147/160	
	101		((A × 16) – 1) × 147/160	
	110		(A × 8) × 147/160	

- The address shown below is output as the DMA address.

  DMA address = (Address set in the operational register) + (0x10 × N)
  N: DMA transfer count (0, 1, 2, 3,...)

  The higher 12 bits (bits 31 to 20) are fixed, and the lower 18 bits (bits 19 to 2) vary. Therefore, if a carry occurs in the lower 18 bits, 1 is not added to the higher 12 bits.

15.10 Special Interrupts

This section explains bits 8 to 6 of the INT_CLR/INT_STATUS register in the operational registers.

(1) Bit 8: ACLINK_CK

An interrupt request is generated when there is a clock request from the Codec side during a suspend state.

(2) Bit 7: CODECGPI

An interrupt request is generated when valid data is input to AC97 input data slot 12 and bit 0 of slot 12 is 1.

(3) Bit 6: ACLINK

An interrupt request is generated when a loopback transfer is performed and an error occurred.

- Loopback transfer

  Set output data in the SLOT12_WR register in the operational registers and set the LOOP bit to 1 at the same time.

  The next valid data that is input to the SLOT12_RD register in the operational registers is compared with the output data and an error occurs if they differ.

15.11 AC97U Suspend Transition Procedure

The procedure for setting the AC97U to suspend mode is shown below.

<1> Issue a power down mode/power save mode command for the Codec.
<2> Set the ck_stop_on bit of the ACLINK_CTRL register in the operational registers to 1.
<3> Transition to suspend mode.

The procedure for canceling the AC97U suspend mode is shown below.
The AC97U suspend mode can be canceled from the CPU or canceled due to a request from the Codec.

(1) Cancellation from the CPU

<1> 1 is set in either the sync_on bit or aclink_rst_on bit of the ACLINK_CTRL register in the operational registers. The SYNC signal or reset_b signal (internal signal) is output and the suspend mode of the AC-Link interface is canceled.
<2> After confirming that the bit that was set in step <1> has become 0, the next processing is performed for the AC97U.

(2) Cancellation due to a request from the Codec

<1> An interrupt request is generated (the ACLINK_CK bit of the INT_CLR/INT_STATUS register in the operational registers is set).
<2> 1 is set in either the sync_on bit or aclink_rst_on bit of the ACLINK_CTRL register in the operational registers. The SYNC signal or reset_b signal is output and the suspend mode of the AC-Link interface is canceled.
<3> After confirming that the bit that was set in step <2> has become 0, the interrupt request of step <1> is cleared (0), and next processing is performed.

15.12 Filter RAM

To use the filter function with DMA transfers, the coefficients for the calculation must be set in advance in the filter RAM.

The method of setting this RAM is described below.

The RAM has two blocks. One is for data output (DAC) and the other is for data input (ADC).

One block is 16 bits × 32 levels.

(1) Method of setting the data output (DAC) RAM

<1> Confirm that the DAC1_ENABLE and DAC1_STATUS bits of the DAC1_CTRL register, the DAC2_ENABLE and DAC2_STATUS bits of the DAC2_CTRL register, and the DAC3_ENABLE and DAC3_STATUS bits of the DAC3_CTRL register in the operational registers are 0.

<2> Set 0 in the SRC_RAM_ADR bit of the CTRL register in the operational registers.

<3> Write data 32 times in the SRC_RAM_DATA register in the operational registers (internally, the address is automatically incremented and the data is expanded in the 32-level register).

(2) Method of setting the data input (ADC) RAM

<1> Confirm that the ADC1_ENABLE bit of the ADC1_CTRL register, the ADC2_ENABLE bit of the ADC2_CTRL register, and the ADC3_ENABLE bit of the ADC3_CTRL register in the operational registers are 0.

<2> Set 1 in the SRC_RAM_ADR bit of the CTRL register in the operational registers.

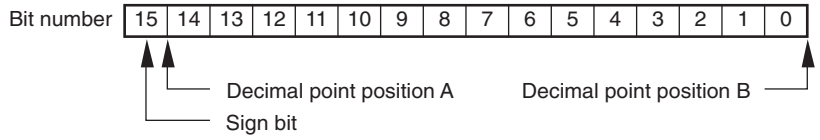
<3> Write data 32 times in the SRC_RAM_DATA register in the operational registers (internally, the address is automatically incremented and the data is expanded in the 32-level register).

(3) RAM data format

The RAM data format is shown below.

- Bit 15 is a sign bit
- The decimal point position is the same position that is used for audio data.

Figure 15-14. RAM Data Format



Remark Coefficient range when the decimal point position is A: –1 < X < 1
Coefficient range when the decimal point position is B: –32767 < X < 32767

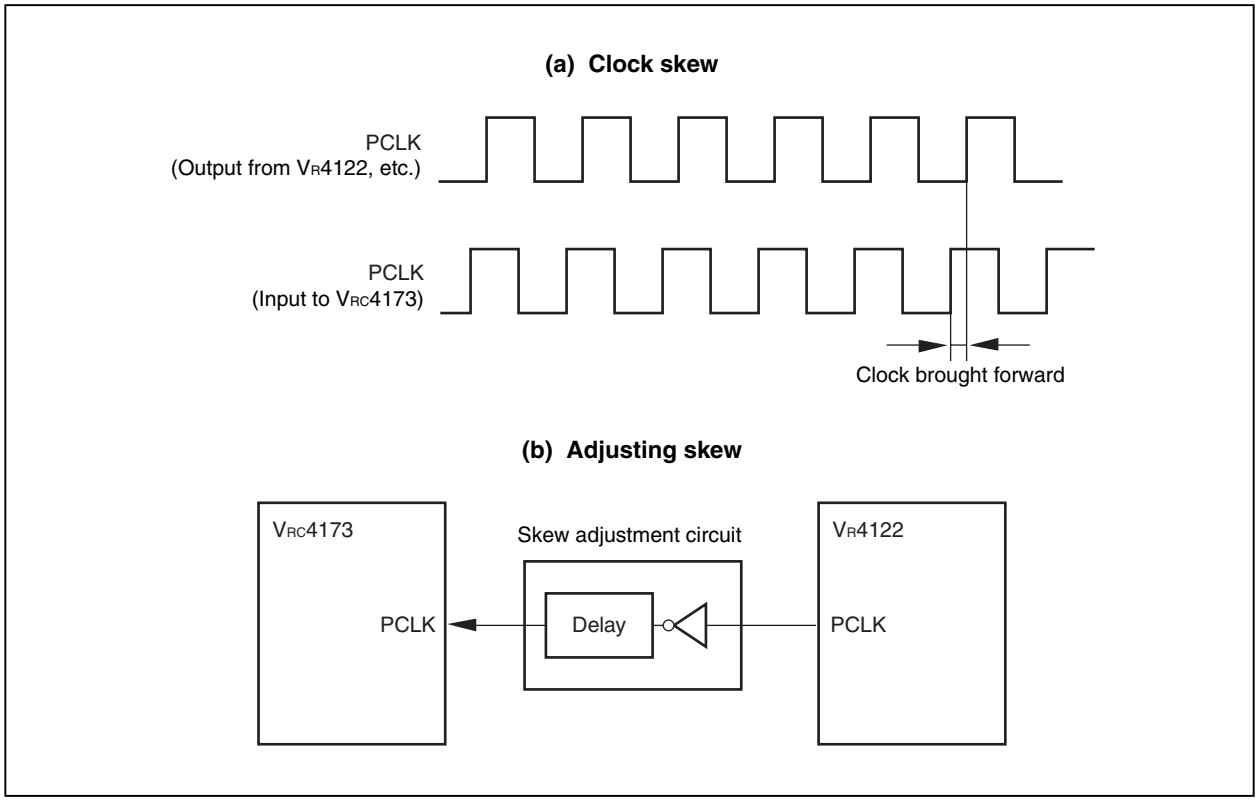
★

APPENDIX A CAUTIONS

A.1 Adjusting Skew of PCI Clock

If the hold time of the PCLK signal of the VRC4173 (4 ns, refer to **μPD31173 (VRC4173) Data Sheet**) cannot be satisfied, bring forward the rising edge of the PCI clock as illustrated in Figure A-1. Evaluate the PCI clock block. If necessary, insert a circuit that adjusts the skew of PCLK to the PCLK pin of the VRC4173.

Figure A-1. Adjusting Skew of PCI Clock



Version 3.1 of the VRC4173 has the following restrictions.

Consult NEC for the restrictions on products other than version 3.1.

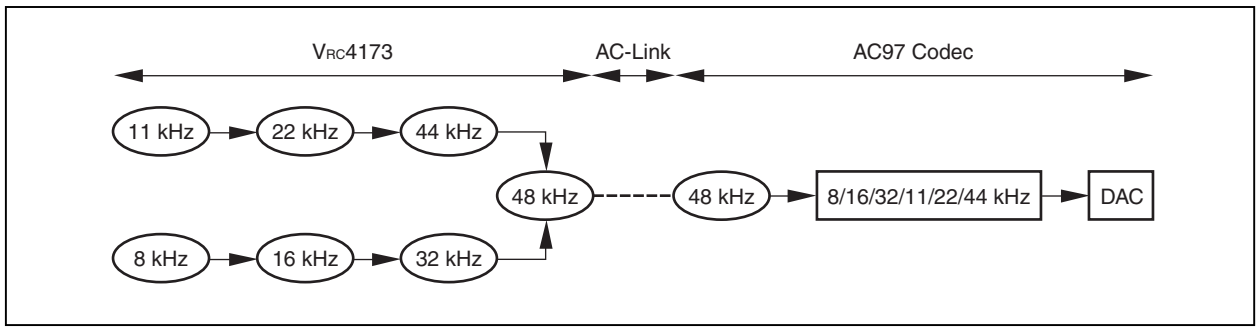
B.1 Noise During Operation of AC97

B.1.1 Phenomenon

If playback or recording is performed with the AC97U, noise is superimposed on the sound.

The AC97U uses an external AC97 Codec via an AC-Link. Because the AC-Link handles audio data at a rate of 48 kHz, the rate must be converted during recording or playback. In the case of playback, for example, the frequency of the AC-Link is 48 kHz where the sampling rate is 8, 16, 32, 11, 22, or 44 kHz, so the rate must be converted as shown in Figure B-1.

Figure B-1. Rate Conversion



The VRC4173 simply converts this rate (i.e., simply arranges 44 kHz data when the rate is converted into 48 kHz and interpolates the insufficient part with the adjacent data by means of linear approximation). If the frequency component of the data string changes, therefore, the waveform is distorted by the interpolating data newly created, and thus noise is generated.

B.1.2 Preventive measures

Modify the hardware and software as follows.

- Hardware: Slightly lower the frequency of the oscillator connected to the AC97 Codec so that the transfer frequency of the AC-Link is 44.1 kHz. By this method, however, the AC97 Codec may be used out of its specifications.
- Software: Modify the driver software of the AC97 block of the VRC4173 to implement rate conversion between the sampling frequency (8, 16, 32, 11, 22, or 44 kHz) and AC-Link transfer rate, and FIR filtering during recording, using software.

B.2 Erroneous Recognition of PC Card

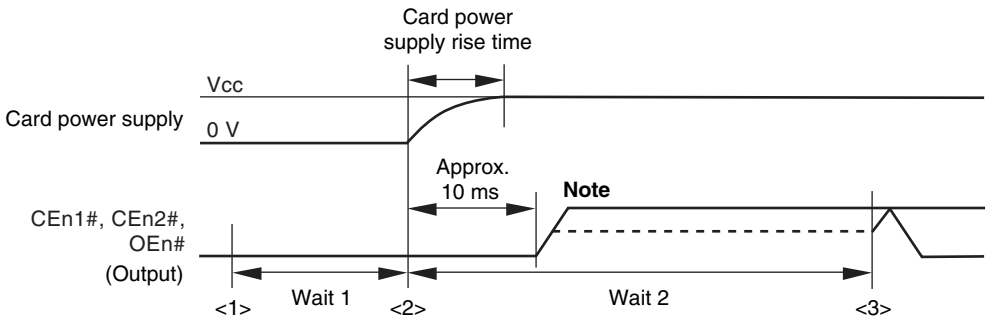
B.2.1 Phenomenon

The CEn1#, CEn2#, and OEn# pins output a low level for about 10 ms after the power supply to the card has been turned on (the supply voltage is set by the VCC(1:0) and VPP(1:0) areas of the PWR_CNT register). Some CF cards that quickly clear a power-on reset recognize this as a request for IDE mode by mistake, and enter the IDE mode. As a result, the VRC4173 cannot recognize the card.

Figure B-2 shows the statuses of the CEn1#, CEn2#, and OEn# signals when the PC card is initialized.

Remark n = 1, 2

Figure B-2. Initializing PC Card



Note The VRC4173 outputs a high-impedance state which is pulled up in the PC card to the high level.

Remarks 1. The numbers in the figure indicate the following operations.

<1> Clear the CARD_OUT_EN bit of the PWR_CNT register to 0.

<2> Set the VCC(1:0) area of the PWR_CNT register (on power application).

<3> Clear the CARD_REST0 bit of the INT_GEN_CNT register to 0 and set the CARD_OUT_EN bit of the PWR_CNT register to 1.

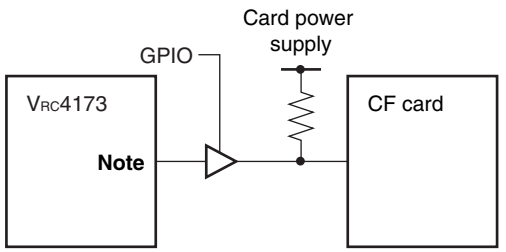
2. The dotted line indicates the high-impedance state.

3. n = 1, 2

B.2.2 Preventive measures

The CF card can be prevented from entering the IDE mode by adding the external circuit shown in Figure B-3.

Figure B-3. Example of Circuit Preventing Shift to IDE Mode



Note CEn1#, CEn2#, and OEn# pins (n = 1, 2)

Insert a buffer with a high-impedance control pin between the CEn1#, CEn2#, and OEn# pins, and the card socket, and pull up the buffer output to the card power supply (n = 1, 2). Connect the high-impedance control pin to the GPIO pin of the VR4122 or VRC4173. The GPIO pin controls the buffer so that it goes into a high-impedance state from when pull-out of the card is detected until driving the card control signal is started (by the CARD_OUT_EN bit of the PWR_CNT register).

B.3 Pulling up PC Card Pins

B.3.1 Phenomenon

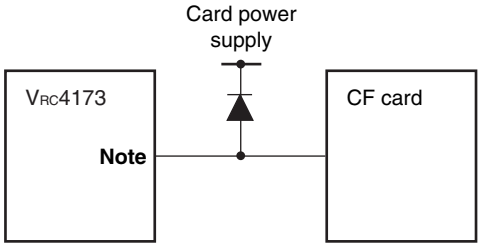
The C1A(22:20), C1A(15:14), WP1, C2A(22:20), C2A(15:14), and WP2 pins are always high when the power supply to the card is off. This is because the C1A(22:20), C1A(15:14), WP1, C2A(22:20), C2A(15:14), and WP2 pins are pulled up by an internal 50 kΩ pull-up resistor of the VRC4173.

B.3.2 Preventive measures

Although only a tiny amount of current flows into these pins when the power supply to the card is off, if any adverse influence is expected, insert a diode between these pins and the card power supply to drop the level of the card pins to close to 0 V.

Figure B-4 shows an example of inserting a diode.

Figure B-4. Example of Inserting Diode



Note C1A(22:20), C1A(15:14), WP1, C2A(22:20), C2A(15:14), and WP2 pins

B.4 Incorrect Playback with AIU

B.4.1 Phenomenon

If recording and playback are simultaneously performed with the AIU, illegal data may be output during playback and thus playback may not be correctly executed.

B.4.2 Preventive measures

(1) Do not set the AIUMEN and AIUSEN bits of the SEQREG register to 1 at the same time.

(2) Recording and playback can be performed simultaneously by using an interrupt and I/O read by the CPU as a substitute for DMA, without using recording DMA.
The sequence used to set the registers for performing recording and playback simultaneously is shown below.

<1> Set a DMA address.
<2> DMAMSKREG register of DCU = 0x0004
Recording DMA = Disabled, Playback DMA = Enabled
<3> Unmask interrupts.
<4> Set registers related to AIU recording.
<5> Set registers related to AIU playback.
<6> SEQREG register of AIU = 0x0011
Start the recording/playback sequencer.
<7> The MIC input receive complete interrupt request occurs (MSTINTR bit of the INTREG register of AIU = 1).
<8> Read the DVALIDREG register of AIU.
Check to see whether valid data is stored in the MDMADATREG register of AIU.
<9> Read the MDMADATREG register of AIU.
Receive the recording data.
<10> Write data to the DVALIDREG register and clear the MDMAV bit to 0.
<11> Clear the MIC input receive complete interrupt request.
Other processing can be performed until the next MIC input receive complete interrupt request occurs.

APPENDIX C INDEX

E

F

G

H

I

K

M

U

V

NEC

Facsimile Message

From:

Name

Company

Tel. FAX

Address

Although NEC has taken all possible steps to ensure that the documentation supplied to our customers is complete, bug free and up-to-date, we readily accept that errors may occur. Despite all the care and precautions we've taken, you may encounter problems in the documentation. Please complete this form whenever you'd like to report errors or suggest improvements to us.

Thank you for your kind support.

North America
NEC Electronics Inc.
Corporate Communications Dept.
Fax: +1-800-729-9288
+1-408-588-6130

Europe
NEC Electronics (Europe) GmbH
Market Communication Dept.
Fax: +49-211-6503-274

South America
NEC do Brasil S.A.
Fax: +55-11-6462-6829

Hong Kong, Philippines, Oceania
NEC Electronics Hong Kong Ltd.
Fax: +852-2886-9022/9044

Korea
NEC Electronics Hong Kong Ltd.
Seoul Branch
Fax: +82-2-528-4411

Taiwan
NEC Electronics Taiwan Ltd.
Fax: +886-2-2719-5951

Asian Nations except Philippines
NEC Electronics Singapore Pte. Ltd.
Fax: +65-250-3583

Japan
NEC Semiconductor Technical Hotline
Fax: +81- 44-435-9608

I would like to report the following error/make the following suggestion:

Document title: _____

Document number: _____ Page number: _____

If possible, please fax the referenced page or drawing.

Document Rating	Excellent	Good	Acceptable	Poor
Clarity	❑	❑	❑	❑
Technical Accuracy	❑	❑	❑	❑
Organization	❑	❑	❑	❑