RTE-MOTHER-A

USER'S MANUAL (Rev. 1.20)

Midas lab

REVISION HISTORY

Date	Revision	Chapter	Explanation of revision
January 15, 1999	0.10		Preliminary version
May 31, 1999	0.20	6.1.5	• Added the restriction to the explanation of GCS7- in the table corresponding to when the size of the GCS7- space differs from that of the GCS5- space.
		6.2.2	• Added EXTBUS_CONTROL to the table.
		6.2.6.9	• Added Cautions 1 and 3.
		6.2.8.11	• Added section.
		6.3.3	• Added section.
		6.3.4.8	• Added section.
		9.4	• Changed T19 to DMAAK- → DMARQ-.
September 10, 1999	1.00	5.1	• Changed the character strings of the connector diagram.
		5.7	• Changed the explanation of JP6.
		6.2.7.3	• Added Caution 2.
		6.2.7.5	• Changed the name of each bit from xxxLOW to xxxHIGH.
		6.2.7.9	• Changed the name of each bit from xxxLOW to xxxHIGH.
		6.2.9.6	• Changed the [1,0,1] combination of [BREQ_EN,BREQ_NUM[1:0]] to a prohibited setting.
		6.3.4.9	• Added section.
		6.4.2	• Added the list of legacy devices.
		6.5.2	• Added the table related to power supplies.
		7	• Added a great deal of contents.
October 6, 1999	1.10	6.3.7.2	• Added the subsystem vendor/device ID.
		7.1.2	• Added lan_test.
		7.2	• Deleted that the serial EEPROM is not installed when the motherboard is shipped from the factory.
			• Added that the MAC address can be acquired from the serial EEPROM and added the serial EEPROM contents.
October 30,1999	1.20	6.3.9.1	• Added that If the CPU board cannot access to the motherboard by using a 32-bit cycle, this CPU board cannot use USB functions.
		7.1.2	• Added test that the reading signatures from connected USB device.
January 20,2000	1.20		First edition of English version

CONTENTS

1. INTRODUCTION

The **RTE-MOTHER-A** is a motherboard that is used by connecting it to a Midas lab CPU board in the RTE-CB series. It is connected to the RTE-CB series CPU board by the GBUS.

The **RTE-MOTHER-A** is equipped with various types of general-purpose buses and connectors for connecting to interfaces. You can use these functions in a great variety of ways such as developing or evaluating real-time operating systems, drivers, or middleware.

1.1. NOTATION USED IN THIS MANUAL

This manual represents numbers according to the notation described in the following table. Hexadecimal and binary numbers may be hyphenated at every four digits, if they are difficult to read because of too many digits being in each number.

Number	Notation rule	Example
Decimal number	Only numerals are indicated.	"10" represents number 10 in decimal.
Hexadecimal number	A number is suffixed with letter H.	"10H" represents number 16 in decimal.
Binary number	A number is suffixed with letter B.	"10B" represents number 2 in decimal.

Number Notation Rules

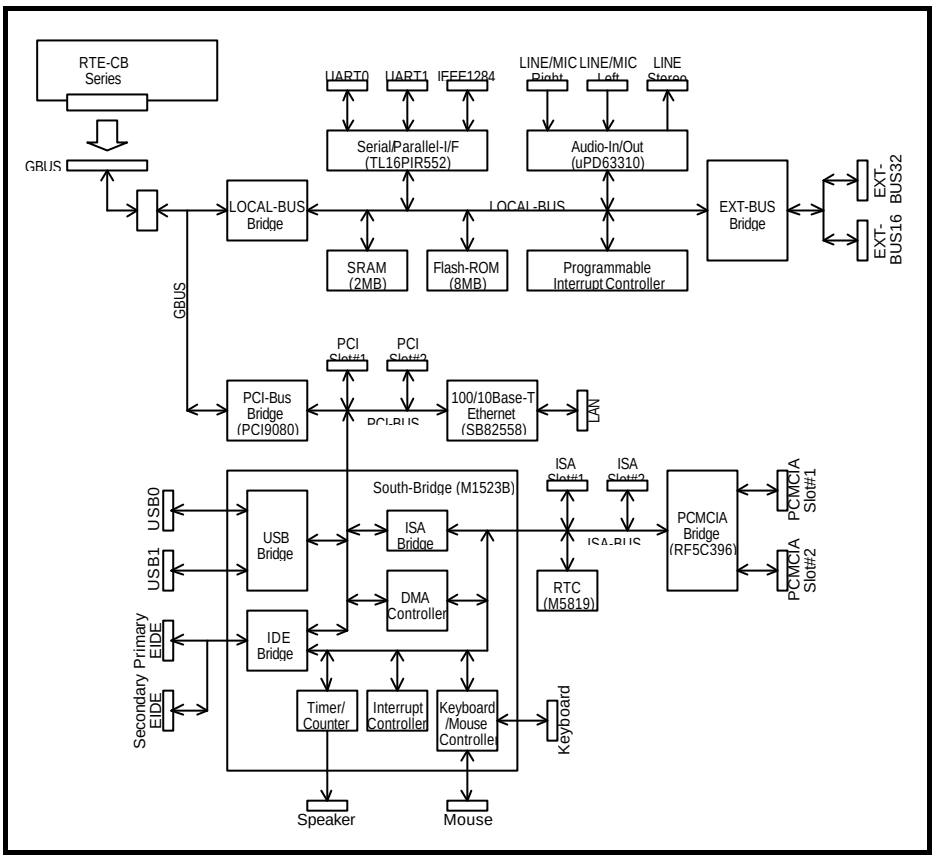
Brackets are used to indicate multiple signals. For example, if there is a bus named ADDR, the notation ADDR[19:16] indicates the four signals ADDR19, ADDR18, ADDR17, and ADDR16. In addition, for a negative logic signal such as BE-, the notation BE-[3:0] indicates the four signals BE3-, BE2-, BE1-, and BE0-. Even when there is only one target signal, this notation can be used such as in BE-[3], which indicates the same signal as BE3-.

1.2. TERMINOLOGY

Terms such as bus cycle, micro cycle, and burst cycle correspond to GBUS terminology. For details, see Section 8.1, "Terminology".

2. FUNCTIONS

The overview of each function block of the RTE-MOTHER-A is shown below.



RTE-MOTHER-A Block Diagram

3. MAJOR FEATURES

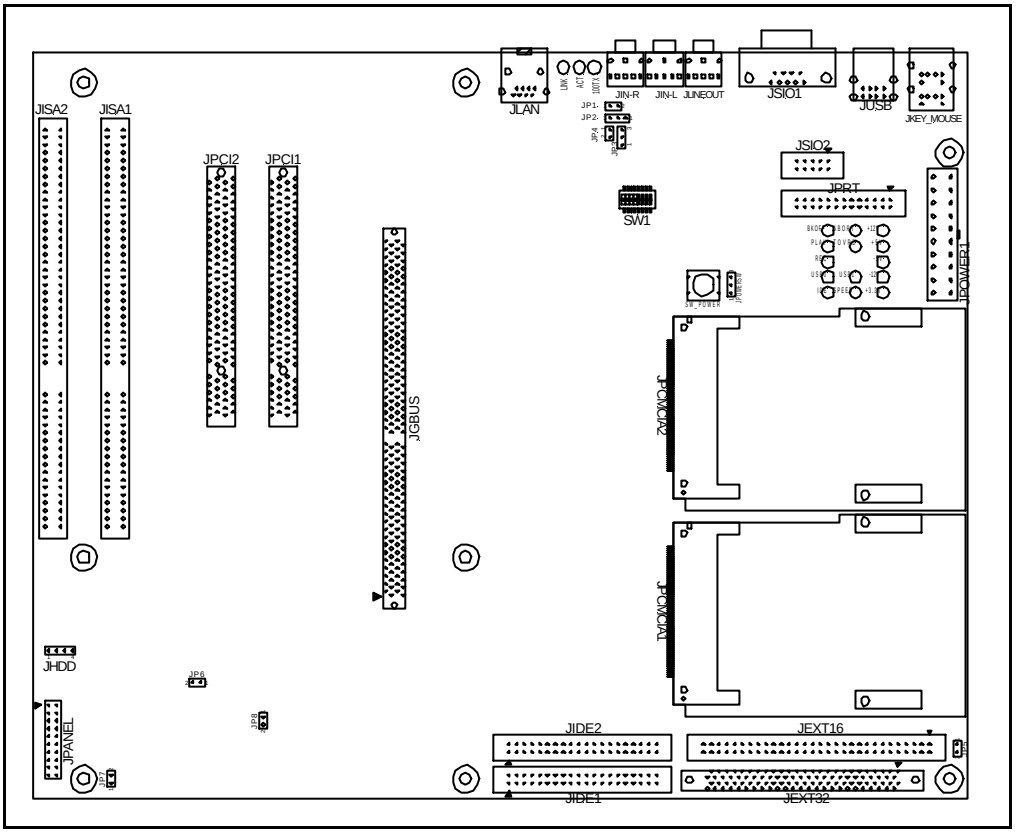
- Can be housed in a case that conforms to ATX standards, and an ATX-standard power supply can be used.
- Provides connectors for connecting to various types of standard buses (PCI×2, ISA×2, PCMCIA×2, USB×2, and E-IDE×2).
- Provides connectors for connecting to EXT-BUS (32-bit) and EXT-BUS (16-bit), which are compatible with the RTE-PC series.
- Provides 2M bytes of SRAM, which can be shared with the RTE-PC series and the PCI bus.
- Provides 8M bytes of flash ROM.
- Provides an interface for audio input/output.
- Provides various types of PC/AT (DOS/V)-compatible functions.
- Provides a 100Base-TX Ethernet interface.
- Provides a serial/parallel interface.

4. BASIC SPECIFICATIONS

Buses	
GBUS	CPU board (RTE-CB series) connection bus
	Data: 32 bits, address: 31 bits, 33 MHz, burst mode available
PCI-BUS	2 slots
	PCI 2.1-compliant, 33.33 MHz, 32 bits, +5 V
ISA-BUS	16-bit Type 2 slot
PCMCIA	2 slots
	PCMCIA2.1/JEIDA4.2-compliant, +5 V/+3.3 V
EXT-BUS	RTE-PC series-compatible EXT bus
	Either one of the 32-bit and 16-bit buses can be used
Interfaces	
Serial	2 channels
	DB9 connector, 10-pin pin header
Printer	1 channel
	IEEE1284-compliant, ECP/EPP-mode support, 26-pin pin header
USB	2 channels
	OpenHCI 1.0a-compliant
EIDE	2 channels
	PIO mode 0 to 4 and Multiword-DMA mode 0 to 2 support, ATA CD-ROM support
Keyboard	PS2-compatible
Mouse	PS2-compatible
Memory	
SRAM	2 MB (512K words × 8 bits × 4)
	Can be accessed from RTE-CB series and PCI bus
Flash ROM	8 MB (2M words × 8 bits × 4)
Others	
RTC	PC/AT-compatible real-time clock
AUDIO	Stereo input × 1 channel, stereo output × 1 channel
	AD/DA converter resolution: 16 bits/monaural, Maximum: 48 kHz sampling
LAN	10/100Base-T, 10Base-T/100Base-TX automatic negotiation, full duplex/half duplex

5. BOARD CONFIGURATION

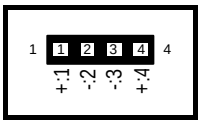
The physical layout of the major components on the RTE-MOTHER-A board is shown below. This chapter explains each component.



RTE-MOTHER-A Components Layout

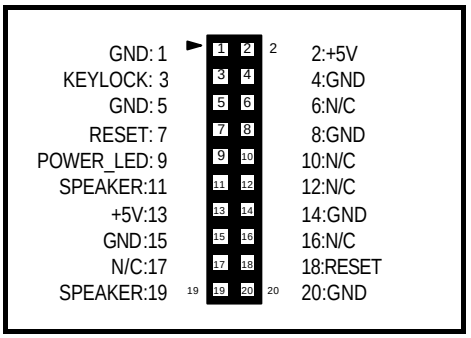
5.1. IDE ACCESS LED CONNECTOR (JHDD)

If an LED is connected to the JHDD connector, it will light up when a hard disk that is connected to the IDE bus is accessed or when a hard disk that is connected to the PCMCIA slot is accessed. The pin arrangement is shown below. Pin 1 and pin 4 are the same signal.



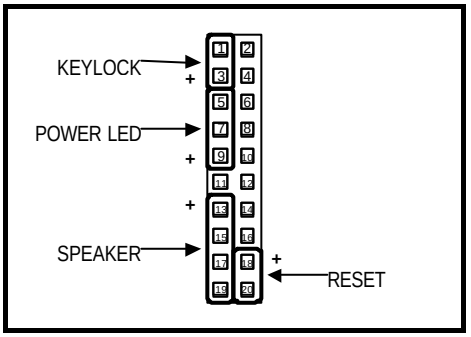
5.2. FRONT PANEL CONNECTOR (JPANEL)

The JPANEL connector is used for connecting to the front panel. If the case conforms to ATX standards, JPANEL enables the front panel's connectors to be connected.
The pin arrangement and functions are shown below. (N/C indicates that there is no connection.)



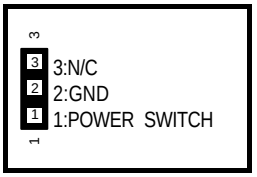
Pin name	Function
KEYLOCK	When this pin is Low, the keyboard is locked. Since this pin signal is pulled up, when nothing is connected, the keyboard will not be locked.
POWER_LED	A power supply LED is connected between this pin and GND. This pin is connected to +5 V via a 330Ω resistor.
SPEAKER	A speaker is connected to this pin. The speaker will emit a sound when a beep tone is generated by the SouthBridge chip or by the PCMCIA card.
RESET	When this pin is Low, the motherboard is reset. Since this pin signal is pulled up, do not connect anything unless a reset is required.

The cable connections for a typical ATX case are shown below.



5.3. POWER SWITCH CONNECTOR (JPOWERSW)

JPOWERSW is a connector for connecting the power switch. This switch is effective only when an ATX-standard power supply is used.
The pin arrangement is shown below. When pushed, the switch short circuits pin 1 and GND. The power supply toggles between ON and OFF each time the switch is pressed.



5.4. POWER SWITCH (SW_POWER)

The SW_POWER switch performs the same action as the switch connected to the JPOWERSW connector. When no switch is connected to the JPOWERSW connector, the power can be turned ON or OFF by using the SW_POWER switch. Like JPOWERSW, this switch is effective only when an ATX-standard power supply is used.

5.5. AUDIO INPUT SWITCHING JUMPERS (JP1, JP2, JP3, JP4)

JP1 to JP4 are jumpers for switching between whether audio is to be input as microphone input or as line input. (See Section 6.2.6, "Audio Circuits.")
The following table shows the JP1 to JP4 settings. The factory settings are for microphone input.

JIN-R input	JP1	JP2
Microphone input	Short	1-2 Short
Line input	Open	2-3 Short

JIN-L input	JP4	JP3
Microphone input	Short	1-2 Short
Line input	Open	2-3 Short

5.6. EXT-BUS Forced 16-Bit Jumper (JP5)

JP5 is a jumper for forcibly causing a 16-bit board that has been connected to the JEXT16 connector but has not been recognized as a 16-bit board to be recognized as a 16-bit board. Although the factory setting for this jumper is Open, JEXT-BUS can be forcibly set to 16-bit mode by setting this jumper to Short. (See Section 6.2.8.10, "EXT-BUS Status Register (EXTBUS_STATUS GCS2:0000-7080H) [Read Only].")

Do not connect a board to the JEXT32 connector when JP5 is set to Short. This may cause a failure to occur.

5.7. Battery Backup Memory Clearing Jumper (JP6)

JP6 is used to clear the battery backup memory within the RTC. This jumper should normally be used with 1-2 short-circuited. To clear the backup memory, shut off the motherboard's power and short circuit 2-3. Then return the jumper to the 1-2 short-circuited state and turn on the motherboard's power.

JP6	Use
1-2 short	Normal use
2-3 short	Clear battery backup memory

To set JP6 to the 2-3 short-circuited state, turn off the motherboard's power. If the motherboard's power is turned on when JP6 is set to the 2-3 short-circuited state, it may cause a failure to occur.

5.8. Front Panel Reset Switch Disabling Jumper (JP7)

To disable the front panel reset switch, set the JP7 to Open state.

5.9. Configuration ROM Switching Jumper (JP8)

JP8 switches the FPGA FLEX series configuration ROM that is installed on the motherboard. When JP8 is set to Short state, the EPC1441PC8 installed in the socket becomes effective. When JP8 is set to Open state, the EPC2TC32 becomes effective.

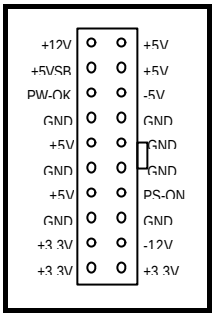
Use the factory setting for JP8.

5.10. Switch 1 (SW1)

SW1 is a general-purpose input register switch that is connected to the local bus. Its setting is read from the input register. (See Section 6.2.9.1, "SW1 Read Out Register (SW1_RDOUT GCS2:0000-8000H) [Read Only].") When read from the register, the value is 1 when the switch is OFF and 0 when the switch is ON. No specific uses have been determined for any of the bits.

5.11. Power Connector (JPOWER1)

An ATX-standard power supply can be connected. The pin arrangement is shown below.



If an ATX-standard power supply is connected, only +5 VSB is supplied when the power supply unit's AC power is first turned on. In this state, pushing the SW_POWER switch on the board or pushing the switch connected to JPOWERSW allows +5 V, +3.3 V, +12 V, -5 V, and -12 V to be supplied. Pushing the switch again stops the supply of power other than +5 VSB.

The +3.3 V supplied from the JPOWER1 connector is not used within this board. The +3.3 V within the board is created from the +5 V by the regulator on the board. This takes into consideration cases in which a non-ATX-standard power supply is connected, as described below.

If an ATX-standard power supply cannot be provided, refer to the following table and supply only the required power indicated there. The current consumption shown in the following table indicates only the current consumed by the motherboard. The current consumed by connected boards (such as RTE-CB series, PCI board, ISA board, or PCMCIA card) or equipment (such as a USB device, keyboard, or mouse) is not taken into consideration.

Pin	Use	Current
+5V	Always supply this power.	Max. 3.5A
+5VSB	Leave this unconnected.	--
+3.3V	Leave this unconnected.	--
+12V	This is supplied to JGBUS, JPCI1/2, and JISA1/2. This is required by PCMCIA1/2 when using a card for which +12 V is required as Vpp.	0A
-5V	This is supplied to JISA1/2.	0A
-12V	This is supplied to JPCI1/2 and JISA1/2.	0A
PW-OK	Leave this unconnected.	--
PS-ON	Leave this unconnected.	--

To connect only some of the power supplies, carefully confirm the amount of power required by the boards you intend to connect and make sure that the amount of power supplied is sufficient. If the amount of power supplied is insufficient, a board failure may occur.

When a non-ATX-standard power supply is used, the SW_POWER switch on the board and the switch connected to JPOWERSW cannot be used.

5.12. LED

The LEDs on this board are shown below.

LED name	Description
+5V	Lights when +5 V power is supplied.
+3.3V	Lights when +3.3 V power is supplied.
+12V	Lights when +12 V power is supplied.
-5V	Lights when -5 V power is supplied.
-12V	Lights when -12 V power is supplied.
USB0	Lights when +5 V power is supplied to the lower USB connector. (Reflects the control register contents. See Section 6.2.9.2, "POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write] ".)
USB1	Lights when +5 V power is supplied to the upper USB connector. (Reflects the control register contents. See Section 6.2.9.2, "POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write]".)
IDE	Lights when the hard disk installed on the IDE bus or in the PCMCIA slot is accessed.
SPEED	Lights when the SPEEDLED pin of the M1523B, which is a SouthBridge chip, is High.
PLAY, REC	Lights when the Audio function is operating. (For details, see Section 6.2.6.3, "Audio Status Register (AUDIO_STATUS GCS2:0000-5010H) [Read Only/Write Only]".)
TOVRDY	Lights when Time-Over Ready is generated. (See Section 6.1.4, "Time-Over Ready".)
ABORT	Lights when an Abort Termination is generated when accessing the PCI bus via the PCI9080. (See Section 6.3.4.6, "Abort Error".)
BRKOFF	Lights when the PCI9080 requests a back off when the GUSE_DIRECT_ACC- signal, which is a GBUS signal, is Low and the PCI bus is being accessed via the PCI9080. (See Section 6.3.4.7, "Back Off Error".)
LINK	Lights when the LAN is in LINK state.
ACT	Lights when packets are being exchanged with the LAN.
100TX	Lights when the LAN is operating in 100Base-TX mode.

5.13. JGBUS Connector (JGBUS)

JGBUS is a connector for connecting a Midas lab CPU board in the RTE-CB series. Either +5 V or +12 V is supplied as the power. However, this is limited to when the relevant power is supplied to JPOWER1. (See Section 6.1, "GBUS.")

5.14. PCI Slots (JPCI1, JPCI2)

These are connectors for inserting PCI cards. (See Section 6.3.6, "PCI Slots.") They are controlled by the PCI9080. PCI 2.1-compliant cards having 32-bit +5 V specifications can be used. +5 V, +12 V, and -12 V power are supplied (+3.3 V is not supplied). However, this is limited to when the relevant power is supplied to JPOWER1. JPCI1 uses AD19 as IDSEL, and JPCI2 uses AD20 as IDSEL.

5.15. ISA Slots (JISA1, JISA2)

These are connectors for inserting 16-bit or 8-bit ISA cards. They are controlled by M1523B, which is a SouthBridge chip. (See Section 6.4.4, "ISA Slots.") +5 V, +12 V, -5 V, and -12 V power are supplied. However, this is limited to when the relevant power is supplied to JPOWER1.

IRQ3 to IRQ7, IRQ9 to IRQ11, IRQ14, and IRQ15 can be used as interrupt lines. However, the interrupt lines are shared with the PCMCIA card.

DMA0 to DMA3, DMA5, and DMA6 can be used for DMA transfers.

5.16. PCMCIA Slots (JPCMCIA1, JPCMCIA2)

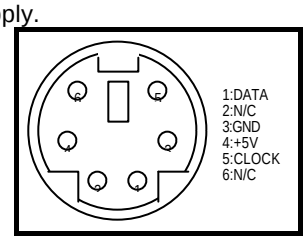
These are connectors for inserting PCMCIA cards. (See Section 6.5, "PCMCIA Bus.") They are

controlled by RFC5C396, which is on the ISA bus. PCMCIA2.1/JEIDA4.2-compliant cards having 16-bit +5 V and +3.3 V specifications can be used. To use a memory card for which +12 V is required as Vpp, +12 V must be supplied to JPOWER1.
The power supplied to a PCMCIA slot is turned ON and OFF by controlling the controller (RFC5C396). An over current state can be read by using a local bus register (see Section 6.2.9.3, "POWER Status Register (POWER_STATUS GCS2:0000-8030H) [Read Only]"), and an interrupt can be generated by generating an over current state (see Section 6.2.7.1, "Overview of Interrupt Resources").
JPCMCIA1 corresponds to SLOT#0 of the controller (RFC5C396) and JPCMCIA2 corresponds to SLOT#1.

The JPCMCIA1/2 connectors do no support the CardBus specifications.

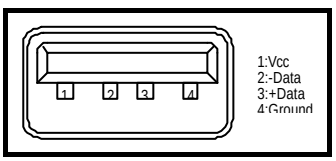
5.17. Keyboard/Mouse Connector (JKEY_MOUSE)

The JKEY_MOUSE connector is used for connecting a PS2-compatible keyboard and mouse. It is controlled by M1523B, which is a SouthBridge chip. (See Section 6.4.2, "M1523B On-chip Legacy Devices.")
The lower connector is for the keyboard, and the upper connector is for the mouse.
The pin arrangement of the connector is shown below. The +5 V on the motherboard is always supplied for the +5 V power supply.



5.18. USB Connector (JUSB)

JUSB is the Universal Serial Bus (USB) root HUB connector. It is controlled by M1523B, which is a SouthBridge chip. (See Section 6.3.9, "USB Controller (M1523B (SouthBridge) On-chip Implementation).")
The pin arrangement of the connector is shown below. Power is supplied to the USB connector by setting a local bus register. (See Section 6.2.9.2, "POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write].") An over current state can be read by using a local bus register (see Section 6.2.9.3, "POWER Status Register (POWER_STATUS GCS2:0000-8030H) [Read Only]"), and an interrupt can be generated by generating an over current state (see Section 6.2.7.1, "Overview of Interrupt Resources").



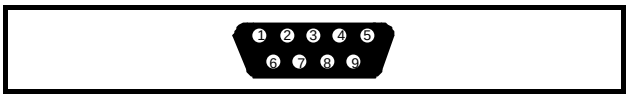
5.19. Serial Connectors (JSIO1, JSIO2)

JSIO1 and JSIO2 are RS-232C interface connectors controlled by a serial controller (TL16PIR552) connected to the local bus. (See Section 6.2.5, "UART/PRINTER (TL16PIR552) (GCS2:0000-0000H to GCS2:0000-302FH).")

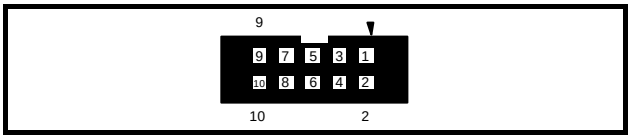
The shape of the JSIO1 connector is the general D-SUB 9-pin RS-232C connector used by PC/AT (DOS/V)-compatible devices. The shape of the JSIO2 connector is a 2.54mm-pitch pin header-type connector. For both connectors, all signals are converted to RS-232C levels. The following figures and table show the connector pin numbers and describe each pin.

The table shows the general cross-cable (reverse cable) wiring when the other side is a D-SUB 9-pin and D-SUB 25-pin connector.

The JSIO2 pin arrangement will be the same as the JSIO1 pin arrangement when a pressure welded-type connector is used for the ribbon cable.



JSIO1 Pin Arrangement (Male)



JSIO2 Pin Arrangement

JSIO1 pin No.	JSIO2 pin No.	Signal name	Input/Output	Destination pin No. (cross)	
				D-SUB9	D-SUB25
1	1	DCD	Input		
2	3	RxD(RD)	Input	3	2
3	5	TxD(SD)	Output	2	3
4	7	DTR(DR)	Output	1, 6	6, 8
5	9	GND		5	7
6	2	DSR(ER)	Input	4	20
7	4	RTS(RS)	Output	8	5
8	6	CTS(CS)	Input	7	4
9	8	RI	Input		
---	10	N/C			

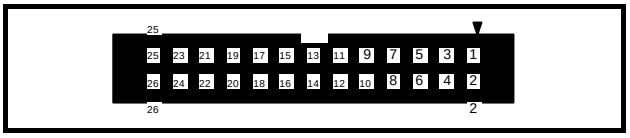
JSIO1 and JSIO2 Connector Signals

5.20. Parallel Connector (JPRT)

JPRT is a parallel (printer) connector controlled by a parallel controller (TL16PIR552) connected to the local bus. (See Section 6.2.5, "UART/PRINTER (TL16PIR552) (GCS2:0000-0000H to GCS2:0000-302FH)".) IEEE1284-compliant devices can be connected.

The shape of the JPRT connector is a 2.54 mm-pitch pin header-type connector. All signals are at the 5 V level. The following figure and table show the connector pin numbers and describe each pin.

The JPRT pin arrangement will be the same as the pin arrangement of a general D-SUB 25-pin connector used with a PC/AT-compatible device when a pressure welded-type connector is used for the ribbon cable.



JPRT Pin Arrangement

JPRT pin No.	Signal name	JPRT pin No.	Signal name
1	STB-	2	AUTO_FD-
3	D0	4	ERROR-
5	D1	6	INIT-
7	D2	8	SELECT_IN-
9	D3	10	GND
11	D4	12	GND
13	D5	14	GND
15	D6	16	GND
17	D7	18	GND
19	ACK-	20	GND
21	BUSY	22	GND
23	PE	24	GND
25	SELECT	26	NC

5.21. Audio Mini-Jack (JIN-R, JIN-L, JLINEOUT)

JIN-R, JIN-L, and JLINEOUT are audio input/output connectors. They are controlled by a μPD63310 connected to the local bus. (See Section 6.2.6, "Audio Circuits.")

JIN-R and JIN-L are audio input connectors that can be switched to either microphone input or line input according to JP1 to JP4. (See Section 5.5, "AUDIO INPUT SWITCHING JUMPERS (JP1, JP2, JP3, JP4).")

JLINEOUT is an audio output connector for line output.

Compatible connectors and signal levels are shown below.

JIN-R, JIN-L

Electrical input conditions

When MIC input is specified: 140m Vp-p (internal amplification: Approximately 20 db)

When LINE input is specified: 1.4 Vp-p

Compatible physical plug shape

Monaural mini-plug (diameter: 3.5)

JLINEOUT

Electrical output conditions

1.4 Vp-p

Compatible physical plug shape

Stereo mini-plug (diameter: 3.5)

5.22. LAN Connector (JLAN)

JLAN is a LAN interface connector controlled by a LAN controller (SB82558) connected to the PCI bus. (See Section 6.3.7, "LAN Controller (SB82558).") It supports 100Base-TX and 10Base-T.

5.23. IDE Connectors (JIDE1, JIDE2)

JIDE1 and JIDE2 are Enhanced IDE interface connectors controlled by M1523B, which is a SouthBridge chip. (See Section 6.3.10, "IDE Bus Master Controller (M1523B (SouthBridge) On-chip Implementation).") They support PIO modes 0 to 4 and MultiWord-DMA modes 0 to 3. JIDE1 corresponds to the primary IDE and JIDE2 corresponds to the secondary IDE. At most two hard disk drives or CD-ROM drives can be connected to each connector.

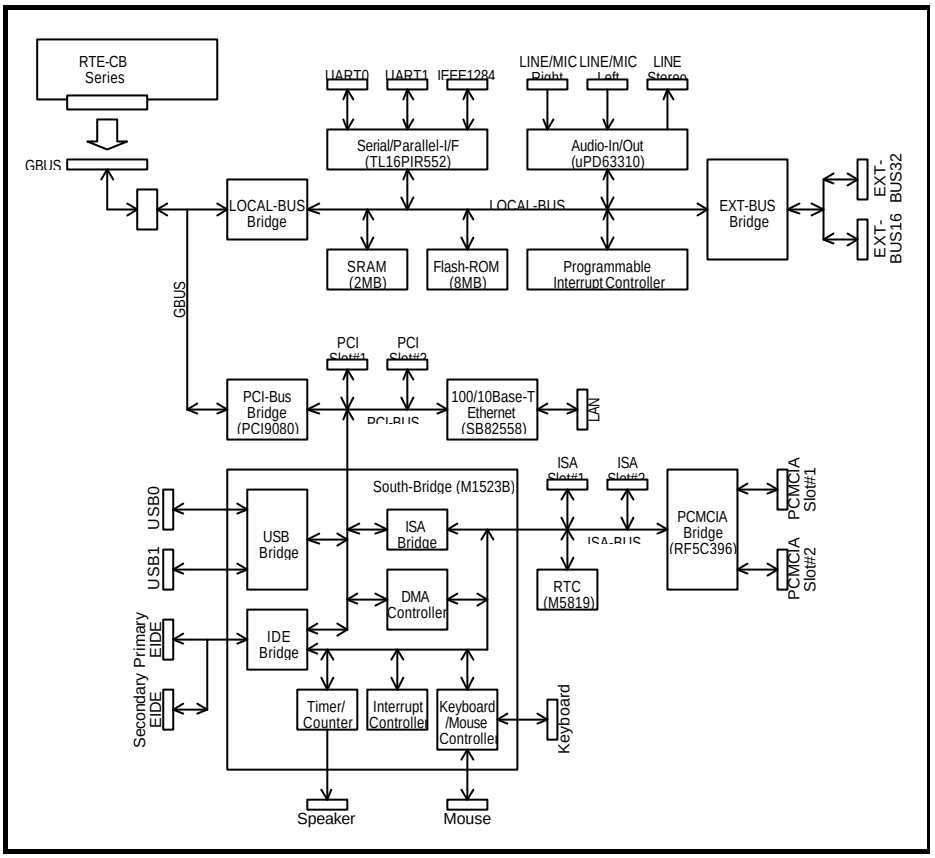
5.24. EXT-BUS Connectors (JEXT32, JEXT16)

JEXT32 and JEXT16 are EXT-BUS connectors that are compatible with the Midas lab expanded bus in the RTE-PC series. JEXT32 can connect a 32-bit EXT-BUS board, and JEXT16 can connect a 16-bit EXT-BUS board. For specifications of the JEXT32 and JEXT16 connectors, see Chapter 9, "APPENDIX.A 32-Bit EXT-BUS Specifications" and Chapter 10, "APPENDIX.B 16-Bit EXT-BUS Specifications," respectively.

Also, when a board is connected to the JEXT16 connector, EXT-BUS is automatically set to 16-bit mode. If the automatic switching to this mode does not function normally, you can forcibly switch it to 16-bit mode by using JP5. (See Section 6.2.8, "EXT-BUS Control Registers.")

JEXT32 and JEXT16 cannot be used at the same time. If boards are connected to both connectors, a failure may occur. Connect a board to only one of these connectors.

6. HARDWARE REFERENCES



This chapter describes the hardware of the RTE-MOTHER-A board. The following figure shows the configuration of the bus connections.

6.1. GBUS

GBUS is a bus that connects a bridge to the local bus and the PCI9080 (bridge to the PCI bus) on the RTE-MOTHER-A board. This bus is accessed from both the CPU board and from the bus master on the PCI bus via the PCI9080. For information about the GBUS bus cycle format, see Chapter 8, "GBUS SPECIFICATIONS."

6.1.1. GBUS Bus Mastership Arbitration

GBUS is accessed from both the CPU board and from the bus master on the PCI bus. Therefore, bus arbitration is required. The GBUS arbitration method differs according to the status of the GUSE_DIRECT_ACC- signal on the GBUS.

If the GUSE_DIRECT_ACC- signal is High, the bus master that accessed the bus last is made to wait. If the CPU board is made to wait, the wait is applied by the GREADY- signal of GBUS. If the PCI9080 is made to wait, it is made to wait without asserting an LHOLDA signal in response to the LHOLD signal asserted by the PCI9080. When the GUSE_DIRECT_ACC- signal is High, neither the CPU board nor the PCI9080 suffer much performance loss as long as accesses do not collide.

If the GUSE_DIRECT_ACC- signal is Low, HOLD-based bus arbitration is performed. If an LHOLD signal is asserted as a bus mastership request from the PCI9080, a GHOLD- signal is asserted on the GBUS. If GHLDA- is asserted from the CPU board in response to this, the PCI9080 begins GBUS access. When the GUSE_DIRECT_ACC- signal is Low, as long as the

CPU board has no special arbitration circuit, the CPU processing efficiency will be lower than when the GUSE_DIRECT_ACC- signal is High because the CPU will be stopped by a HOLD each time the PCI9080 accesses the bus. However, **if the CPU board has resources that can be accessed from the PCI bus and this access is permitted, the GUSE_DIRECT_ACC- signal must be set to Low**.

6.1.2. Temporary Release of GBUS Bus Mastership

The burst length (frequency) on the GBUS is not limited. Therefore, once the bus master on the PCI bus obtains the GBUS bus mastership, it may not relinquish that bus mastership for a long time. Taking this possibility into consideration, you can cause the bus mastership to be relinquished to the PCI9080 under fixed conditions. The conditions at this time can be selected under program control from among several choices. You can also cause the bus mastership to be relinquished to the PCI9080 according to a signal from the CPU board.
For details, see Section 6.2.9.6, "BREQ Control Register (BREQ_CONTROL GCS2:0000-8060H) [Read/Write]."

6.1.3. GBUS Bus Lock

Accesses to SRAM and the PCI bus support a bus lock function.
A bus lock can be applied due to an access to SRAM or the PCI bus from the CPU board by using the GBLOCK-[1:0] signal on the GBUS or control registers on the local bus. (See Section 6.2.9.8, "Bus Lock Control Register (BLOCK_CONTROL GCS2:0000-8080H) [Read/Write].") A bus lock can also be applied due to an access to SRAM from the bus master on the PCI bus via the PCI9080. This case is limited to when the bus lock is requested by the bus master on the PCI bus.

6.1.4. Time-Over Ready

If the GREADY- signal is not active for at least 10 milliseconds when EXT-BUS or the PCI bus is accessed from the CPU board, a time-over ready is generated and the bus cycle is forcibly terminated to avoid deadlock.
When a time-over ready is generated, the TOVRDY-LED on the motherboard lights up. This LED remains lit until it is cleared by software. (See Section 5.12, "LED" and Section 6.2.9.9, "TOVRDY LED Clear Register (TOVRDY_LED_CLR GCS2:0000-8090H) [Write Only].") In addition, the generation of a time-over ready can cause an interrupt to be generated. (See Section 6.2.7.1, "Overview of Interrupt Resources.")

A time-over ready that is generated by an access to EXT-BUS will be generated when one micro cycle exceeds 10 milliseconds.

A time-over ready that is generated by an access to the PCI bus is generated when the entire bus cycle exceeds 10 milliseconds, and the GREADY- signal remains asserted until that bus cycle ends.

6.1.5. GBUS Memory and I/O Map (Access from the CPU Board)

Memory and I/O are allocated according to the GCS-[7:0] signals from the CPU board. The following table shows the correspondence between the GCS-[7:0] signals and the resources that are accessed. If the CPU on the CPU board has an I/O space, you should allocate any space for which "I/O" appears in the recommended space column of the table to an I/O space.

Signal name	Recommended space	Minimum range	Maximum range	Accessed resources and remarks
GCS0-	Memory	2M bytes		• SRAM on the local bus • Can also be accessed from the bus master on the PCI bus. • A bus lock can be applied by the GLOCK0- signal or by control from I/O on the local bus.
GCS1-	Memory	8M bytes		• Flash ROM on the local bus • This space can be mapped to a Boot ROM space according to switch settings on the CPU board, and the system can be booted from flash ROM.
GCS2-	I/O	64K bytes		• Related to control registers on the local bus
GCS3-	Memory	64K bytes	16M bytes	• EXT-BUS memory space • When a board is connected to the JEXT32 connector, the memory space of the connected board can be accessed from this space. The maximum area size is 16M bytes. • When a board is connected to the JEXT16 connector, memory and I/O are not distinguished, and all spaces of the connected board can be accessed. The maximum area size is 1M byte. • If a control register on the local bus has an address expansion bank register and a space of at least 64K bytes has been allocated by using this bank register, access can be expanded to 16M bytes. (See Section 6.2.8, "EXT-BUS Control Registers.") • When a board is connected to the JEXT32 connector, two-cycle DMAs are supported for two channels, and an access expansion bank register is provided for each DMA channel. (See Section 6.2.8, "EXT-BUS Control Registers.")
GCS4-	I/O	64K bytes	16M bytes	• EXT-BUS I/O space • Can access the I/O space of JEXT32. The maximum area is 16M bytes. • JEXT16 cannot be accessed. • If a control register on the local bus has an address expansion bank register and a space of at least 64K bytes has been allocated by using this bank register, access can be expanded to 16M bytes. (See Section 6.2.8, "EXT-BUS Control Registers.") • Two-cycle DMAs are supported for two channels, and an access expansion bank register is provided for each DMA channel. (See Section 6.2.8, "EXT-BUS Control Registers.")
GCS5-	Memory	1M byte	2G bytes	• PCI bus memory space • Can access the memory or I/O space of the PCI bus via the PCI9080. • The allocated area should be as large as possible. • A bus lock can be applied by the GLOCK1- signal or by control from I/O on the local bus.
GCS6-	I/O	512 bytes		• PCI9080 control register space
GCS7-	I/O	64K bytes	2G bytes	• PCI bus I/O space • Accesses the I/O space of the PCI bus via the PCI9080. • An allocated area of 64K bytes is usually sufficient. • A bus lock can be applied by the GLOCK1- signal or by control from I/O on the local bus. • If the CPU on the CPU board has no I/O space, this space should be eliminated. In that case, I/O will be accessed from the GCS5- space. • If the size of the GCS5- space differs from the size of the GCS7- space, the address line of the portion corresponding to the difference in the sizes must be zeros. For example, if the GCS5- space is 1M byte and the GCS7- space is 64K bytes, GADDR[19:16] must be zeros when accessing the GCS7- space. This is a restriction related to the PCI9080.

6.1.6. GBUS Memory Map (Access from the PCI Bus)

The following two kinds of resources on the GBUS can be accessed from the bus master on the PCI bus.

One kind of resource is SRAM on the motherboard. All of the SRAM space can be accessed. To access this space, GADDR[25], which is a GBUS address, is set to Low. In other words, SRAM will be allocated to the range of GBUS addresses from 0000-0000H to 01FF-FFFFH. (However, the actual size of SRAM is 2M bytes.)

The other kind of resource is a resource on the CPU board. To access a resource on the CPU board, GADDR[25], which is a GBUS address, is set to High. In other words, the 32M-byte area of GBUS addresses from 0200-0000H to 03FF-FFFFH will be allocated as a resource on the CPU board.

However, to enable the bus master on the PCI bus to access a resource on the CPU board, GUSE_DIRECT_ACC-, which is a signal on the GBUS, must be Low. When GUSE_DIRECT_ACC- is High, the CPU board is indicating that it has no resources that can be accessed by the bus master on the PCI bus. The method of arbitrating access to SRAM differs significantly depending on whether GUSE_DIRECT_ACC- is High or Low. (See Section 6.1.1, "GBUS Bus Mastership Arbitration.")
Also, when the bus master on the PCI bus accesses a resource, a PCI9080 setting determines how the address on the PCI bus is converted to an address on the GBUS.

6.2. Local Bus

The local bus contains SRAM, flash ROM, Audio control registers, and other control registers such as interrupt controller control registers. Of these, SRAM can be accessed from both the CPU board and from the bus master on the PCI bus.

Also, access to the EXT-BUS can also be performed from the CPU board via a bridge connected to the local bus. The local bus resources are explained below. The notation "GCSx:yyyy-yyyH" indicates the address yyyy-yyyyH when the chip select GCSx- signal is active.

6.2.1. Number of Waits

The following table shows the number of waits when accessing a resource on the local bus. Depending on the resource, the number of waits may change according to the status of the GCLK_LOW- signal of the GBUS. Also, each cell in the following table shows two numbers of waits separated by "/". The first number indicates the number of waits for a single cycle or for the first micro cycle of a burst cycle. The second number indicates the number of waits for the second and subsequent micro cycles of a burst cycle.

When different resources are accessed consecutively, one additional wait than the number shown in the following table may be inserted for a single cycle or the first micro cycle of a burst cycle.

For the kinds of resources that correspond to each wait type, see the following section.

Wait type	Number of waits			
	GCLK_LOW-=High		GCLK_LOW-=Low	
	Write	Read	Write	Read
SRAM	0Wait/1Wait	0Wait/1Wait	0Wait/1Wait	0Wait/1Wait
FAST	4Wait/4Wait	3Wait/4Wait	3Wait/3Wait	2Wait/3Wait
MID	7Wait/7Wait	6Wait/7Wait	7Wait/7Wait	6Wait/7Wait
SLOW	8Wait/8Wait	7Wait/8Wait	5Wait/5Wait	4Wait/5Wait
PPCS	6Wait/6Wait	5Wait/6Wait	5Wait/5Wait	4Wait/5Wait

The numbers shown in the above table when the "Wait type" is PPCS are the minimum number of waits. Additional waits may be requested according to the resource to be accessed.

6.2.2. List of Resources

The following table lists the resources on the local bus. For information about the "Wait type" in the following table, see the previous section.

Name	Address	Wait type	Reference section
SRAM	GCS0:0000-0000H to GCS0:001F-FFFFH	SRAM	6.2.3
Flash ROM	GCS1:0000-0000H to GCS1:007F-FFFFH	FAST	6.2.4
UART0 (TL16PIR552)	GCS2:0000-0000H to GCS2:0000-007FH	FAST	6.2.5
UART1 (TL16PIR552)	GCS2:0000-1000H to GCS2:0000-107FH	FAST	6.2.5
PRINTER PPCS- (TL16PIR552)	GCS2:0000-2000H to GCS2:0000-207FH	PPCS	6.2.5
PRINTER ECPCS- (TL16PIR552)	GCS2:0000-3000H to GCS2:0000-302FH	FAST	6.2.5
μPD63310 register	GCS2:0000-4000H to GCS2:0000-401FH	SLOW	6.2.6.1
AUDIO_CONT	GCS2:0000-5000H	FAST	6.2.6.2
AUDIO_STATUS	GCS2:0000-5010H	FAST	6.2.6.3
AUDIO_MCLKDIV	GCS2:0000-5020H	FAST	6.2.6.4
AUDIO_FIFO	GCS2:0000-5030H	MID	6.2.6.5
AUDIO_FIFO_FULL_LEVEL	GCS2:0000-5040H	FAST	6.2.6.6
AUDIO_FIFO_HALF_LEVEL	GCS2:0000-5050H	FAST	6.2.6.7
AUDIO_FIFO_DEPTH	GCS2:0000-5060H	FAST	6.2.6.8
AUDIO_CONT2	GCS2:0000-5070H	FAST	6.2.6.9
AUDIO_STATUS2	GCS2:0000-5080H	FAST	6.2.6.10
INT_STATUS0	GCS2:0000-6000H	FAST	6.2.7.2
INT_CLEAR0	GCS2:0000-6010H	FAST	6.2.7.3
INT_EDGE0	GCS2:0000-6020H	FAST	6.2.7.4
INT_POLARITY0	GCS2:0000-6030H	FAST	6.2.7.5
INT_STATUS1	GCS2:0000-6040H	FAST	6.2.7.6
INT_CLEAR1	GCS2:0000-6050H	FAST	6.2.7.7
INT_EDGE1	GCS2:0000-6060H	FAST	6.2.7.8
INT_POLARITY1	GCS2:0000-6070H	FAST	6.2.7.9
GINTO0_INTEN0	GCS2:0000-6100H	FAST	6.2.7.10
GINTO0_INTEN1	GCS2:0000-6110H	FAST	6.2.7.14
GINTO1_INTEN0	GCS2:0000-6120H	FAST	6.2.7.11
GINTO1_INTEN1	GCS2:0000-6130H	FAST	6.2.7.15
GINTO2_INTEN0	GCS2:0000-6140H	FAST	6.2.7.12
GINTO2_INTEN1	GCS2:0000-6150H	FAST	6.2.7.16
GINTO3_INTEN0	GCS2:0000-6160H	FAST	6.2.7.13
GINTO3_INTEN1	GCS2:0000-6170H	FAST	6.2.7.17
EXTBUS_MEM_AMASK	GCS2:0000-7000H	FAST	6.2.8.2
EXTBUS_IO_AMASK	GCS2:0000-7010H	FAST	6.2.8.3
EXTBUS_CORE_MEM_BANK_ADDR	GCS2:0000-7020H	FAST	6.2.8.4
EXTBUS_CORE_IO_BANK_ADDR	GCS2:0000-7030H	FAST	6.2.8.5
EXTBUS_DMA0_MEM_BANK_ADDR	GCS2:0000-7040H	FAST	6.2.8.6
EXTBUS_DMA0_IO_BANK_ADDR	GCS2:0000-7050H	FAST	6.2.8.7
EXTBUS_DMA1_MEM_BANK_ADDR	GCS2:0000-7060H	FAST	6.2.8.8
EXTBUS_DMA1_IO_BANK_ADDR	GCS2:0000-7070H	FAST	6.2.8.9
EXTBUS_STATUS	GCS2:0000-7080H	FAST	6.2.8.10
EXTBUS_CONTROL	GCS2:0000-7090H	FAST	6.2.8.11
SW1_RDOUT	GCS2:0000-8000H	FAST	6.2.9.1
POWER_CONTROL	GCS2:0000-8020H	FAST	6.2.9.2
POWER_STATUS	GCS2:0000-8030H	FAST	6.2.9.3
ISA_INT_VECTOR	GCS2:0000-8040H	FAST	6.2.9.4
ISA_INT_STATUS	GCS2:0000-8050H	FAST	6.2.9.5
BREQ_CONTROL	GCS2:0000-8060H	FAST	6.2.9.6
FROM_CONTROL	GCS2:0000-8070H	FAST	6.2.9.7
BLOCK_CONTROL	GCS2:0000-8080H	FAST	6.2.9.8
TOVRDY_LED_CLR	GCS2:0000-8090H	FAST	6.2.9.9
ABORT_LED_CLR	GCS2:0000-80A0H	FAST	6.2.9.10
BACKOFF_LED_CLR	GCS2:0000-80B0H	FAST	6.2.9.11

6.2.3. SRAM (GCS0:0000-0000H to GCS0:001F-FFFFH)

This is the SRAM space that can be accessed from both the CPU board and from the PCI bus.
When it is accessed from the CPU board, it is allocated in the space shown above. However, when it is accessed from the bus master on the PCI bus, it is allocated in 0000-0000H to 01FF-FFFFH. (The actual SRAM size is 2M bytes.) The relationship between addresses on the PCI bus and addresses on the GBUS is determined by PCI9080 settings.

6.2.4. Flash ROM (GCS1:0000-0000H to GCS1:007F-FFFFH)

A +5 V single flash ROM is installed. Four MBM29F016 chips, each consisting of 2M words × 8 bits, are provided. The flash ROM can be overwritten in terms of individual bytes. For information about how to overwrite the flash ROM, refer to the MBM29F016 manual.
This area, which is the CPU board memory space, should be arranged in two areas. One of the two areas is a UV-EPROM area on the CPU board. A switch on the CPU board determines whether the UV-EPROM on the CPU board is accessed due to an access to this area or whether the flash ROM on the motherboard can be accessed. This enables the IPL to be written to this flash ROM area and the CPU board to be started up by using that IPL.
Also, functions for resetting the flash ROM and reading the BUSY status of the flash ROM are provided in a local bus register. (See Section 6.2.9.7, "Flash ROM Control Register (FROM_CONTROL GCS2:0000-8070H) [Read/Write].")

6.2.5. UART/PRINTER (TL16PIR552) (GCS2:0000-0000H to GCS2:0000-302FH)

The Texas Instruments TL16PIR552 (dual UART with 1284 parallel port) LSI is used as the UART/PRINTER controller. The TL16PIR552 has a 2-channel UART and 1-channel IEEE1284-compliant bi-directional printer port. It also has a 16-character FIFO buffer in the transmission/reception block of the UART, and a function for automatically controlling RTS/CTS flow. Therefore, an overrun error can be suppressed by the minimum interrupt.

Each register of the TL16PIR552 is assigned as listed below. For an explanation of the function of each register, refer to the TL16PIR552 manual. (The TL16PIR552 manual is available from the TI&ME of the Texas Instruments home page (http://www.ti.com/).

Address	Function	Read	Write
GCS2:0000-0000H	UART-CH#0	RBR/DLL	THR/DLL
GCS2:0000-0010H		IER/DLM	IER/DLM
GCS2:0000-0020H		IIR	FCR
GCS2:0000-0030H		LCR	LCR
GCS2:0000-0040H		MCR	MCR
GCS2:0000-0050H		LSR	LSR
GCS2:0000-0060H		MSR	MSR
GCS2:0000-0070H		SCR	SCR
GCS2:0000-1000H	UART-CH#1	RBR/DLL	THR/DLL
GCS2:0000-1010H		IER/DLM	IER/DLM
GCS2:0000-1020H		IIR	FCR
GCS2:0000-1030H		LCR	LCR
GCS2:0000-1040H		MCR	MCR
GCS2:0000-1050H		LSR	LSR
GCS2:0000-1060H		MSR	MSR
GCS2:0000-1070H		SCR	SCR
GCS2:0000-2000H	PRINTER(PPCS-)	DATA	DATA/ECPAFIFO
GCS2:0000-2010H		DSR	-----
GCS2:0000-2020H		DCR	DCR
GCS2:0000-2030H		EPPADDR	EPPADDR
GCS2:0000-2040H to GCS2:0000-2070H		EPPDATA	EPPDATA
GCS2:0000-3000H	PRINTER(ECPCS-)	PPDATAFIFO/ TESTFIFO/CNFGA	PPDATAFIFO/ TESTFIFO
GCS2:0000-3010H		CNFGB	-----
GCS2:0000-3020H		ECR	ECR

TL16PIR552 Register Arrangement

The XIN input of the TL16PIR552 is connected to the 22.1184 MHz clock.

The UART-CH#0, UART-CH#1, and PRINTER interrupts can be output to GINTO-[3:0] via the interrupt controller. (See Section 6.2.7, "Interrupt Control Circuits.")

UART-CH#0 is connected to the JSIO1 connector, UART-CH#1 is connected to the JSIO2 connector, and PRINTER is connected to JPRT.

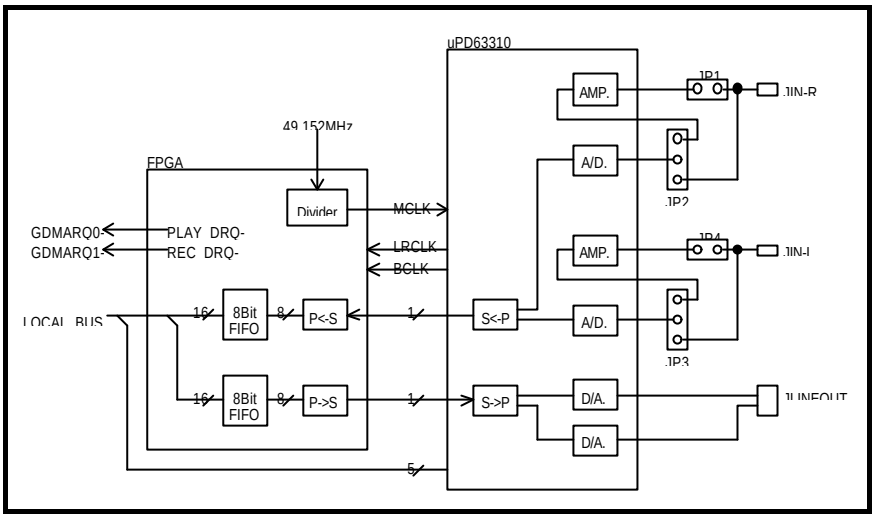
The TL16PIR552 is reset when the system is reset.

6.2.6. Audio Circuits

The audio circuits perform stereo audio input/output. An NEC *m*PD63310 is used for the A/D and D/A converter, the resolution is 16 bits per channel, and the maximum sampling rate is 48 kHz. These circuits have been designed to maintain upward compatibility with the audio features installed in the conventional Midas lab RTE-V831-PC and RTE-V832-PC.

In the following explanation, audio input is called "recording" and audio output is called "playing."

The following figure shows a block diagram of the audio circuits.



The *m*PD63310 has amplifiers in its input circuits, which are used for microphone (MIC) input. You can switch whether the input is MIC input or LINE input by using the JP1 to JP4 jumper switches. (See Section 5.5, "AUDIO INPUT SWITCHING JUMPERS (JP1, JP2, JP3, JP4).")

During recording, A/D-converted data is sent as serial data to the FPGA and converted to 8-bit data. The 8-bit data is stored in queuable, programmable 8-bit FIFOs up to a maximum of 254 bytes, and when data is read from the local bus, 2 bytes of data are read at one time.

During playing, data written from the local bus is stored in queuable, programmable 8-bit FIFOs up to a maximum of 254 bytes, converted sequentially to serial data, and sent to the μPD63310. After the μPD63310 converts the received data to parallel data, it is D/A-converted and output.

Since the maximum number of bytes can be set under program control separately for the play and record 8-bit FIFOs, you can evaluate the FIFO requirements by balancing them with CPU performance.

Although audio data can be transferred under software control according to FIFO-related information in the AUDIO_STATUS register, it can also be transferred by using DMA transfers. A DMA transfer uses channel 0 for GBUS playing and channel 1 for recording.

Although a FIFO itself has an 8-bit configuration, writing and reading is performed in 16-bit units. The 16-bit/8-bit conversion is performed by an internal circuit. Also, a DMA request is generated during playing when the FIFO has at least two bytes of free space, and a DMA request is generated during recording when the FIFO has at least two bytes of data.

A DMA request to the GBUS becomes inactive either when the corresponding GDMAAK- signal becomes active or after the next rising edge of GCLK for which the GWAITI- signal of the write cycle to the FIFO is sampled High. A DMA cycle ends following eight consecutive GCLK pulses after the GWAITI- signal is sampled High, and GDMARQ- will not become active again within four GCLK pulses after the DMA cycle ends. This is done to prevent a DMA overrun.

6.2.6.1 μPD63310 Register (GCS2:0000-40000H to GCS2:0000-401FH)

The μPD63310 register is allocated as follows. For details, see the μPD63310 data sheet.

Address	Function	D5	D4	D3	D2	D1	D0
GCS2:0000-4000H	Address register	Register number					
GCS2:0000-4010H	Data register	Gain control					

6.2.6.2.Audio Control Register (AUDIO_CONT GCS2:0000-5000H) [Read/Write]

Bit	Signal name	RST	Function
0	PLAY*1	0	0: Stop playing audio. 1: Play audio.
1	PLAY_UNDF_INTEN*2	0	0: Do not request interrupt when playing FIFO underflow occurs. 1: Request interrupt when playing FIFO underflow occurs.
2	PLAY_OVRF_INTEN*2	0	0: Do not request interrupt when play ing FIFO overflow occurs. 1: Request interrupt when playing FIFO overflow occurs.
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	
8	REC*3	0	0: Stop recording audio. 1: Record audio.
9	REC_UNDF_INTEN*2	0	0: Do not request interrupt when recording FIFO underflow occurs. 1: Request interrupt when recording FIFO underflow occurs.
10	REC_OVRF_INTEN*2	0	0: Do not request interrupt when recording FIFO overflow occurs. 1: Request interrupt when recording FIFO overflow occurs.
11	Unused	x	
12	Unused	x	
13	Unused	x	
14	Unused	x	
15	AUDIO-RESET*4	0	0: Do not reset audio circuits. 1: Reset all audio circuits.

<<Cautions>>

1. When PLAY is set to "1," a DMA request is immediately asserted (when it is not masked), and the PLAY_DOING bit of the AUDIO_STATUS register becomes "1."
   When PLAY is set to "0," a DMA request is immediately de-asserted. However, if the FIFO contains data, playing will continue, and the "1" state of the PLAY_DOING bit of the AUDIO_STATUS register will be maintained. When the playing of all data within the FIFO ends, the PLAY_DOING bit becomes "0."
   Also, after PLAY has been set to "1," if PLAY is set to "0" before playing begins, the PLAY_DOING bit immediately becomes "0." However, in this case, if CPU board DMA transfers are enabled, playing data may end up remaining in the FIFO, depending on the timing.
2. All enabled interrupt requests due to the occurrence of an overflow or underflow are combined by a logical OR operation and sent to the interrupt controller. These interrupts can be allocated to GINTO-[3:0] by setting the interrupt controller. (See Section 6.2.7, "Interrupt Control Circuits.")
3. When REC is set to "1," the REC_DOING bit of the AUDIO_STATUS register immediately becomes "1," and the fetching of data to the FIFO begins. In this state, when data is written to the FIFO, a DMA request is asserted.
   When REC is set to "0," a DMA request is immediately de-asserted. Therefore, if any data had remained in the FIFO at that time, it will not be transferred and will remain in the FIFO. Also, the fetching of audio data continues until a (16-bit) data division, and when that data fetching ends and the data is written to the FIFO, the REC_DOING bit of the AUDIO_STATUS register becomes "0."
   After REC has been set to "1," if REC is set to "0" before recording begins, the REC_DOING bit immediately becomes "0." However, in this case, recording data may end up remaining in the FIFO, depending on the timing.
4. The μPD63310, playing circuits, and recording circuits are all reset by AUDIO_RESET. The FIFOs are also flushed. Besides the AUDIO_RESET signal, there are functions for resetting the playing circuits and recording circuits separately. (See Section 6.2.6.9, "Audio Control Register 2 (AUDIO_CONT2 GCS2:0000-5070H) [Read/Write].")

6.2.6.3.Audio Status Register (AUDIO_STATUS GCS2:0000-5010H) [Read Only/Write Only]

Bit	Signal name	RST	Function
0	PLAY_DOING[*1]	0	0: No audio is being played. 1: Audio is being played.
1	PLAY_UNDF[*2*4*5]	0	0: No playing FIFO underflow occurred. 1: Playing FIFO underflow occurred.
2	PLAY_OVRF[*3*4*5]	0	0: No playing FIFO overflow occurred. 1: Playing FIFO overflow occurred.
3	Unused	x	
4	PLAY_FIFO_EMPTY[*7]	1	0: Playing FIFO is not empty. 1: Playing FIFO is empty.
5	PLAY_FIFO_ HFULL[*7]	0	0: Playing FIFO is not half full. 1: Playing FIFO is half full.
6	PLAY_FIFO_ FULL[*7]	0	0: Playing FIFO is not full. 1: Playing FIFO is full.
7	Unused	x	
8	REC_DOING[*1]	0	0: No audio is being recorded. 1: Audio is being recorded.
9	REC_UNDF[*2*4*6]	0	0: No recording FIFO underflow occurred. 1: Recording FIFO underflow occurred.
10	REC_ORVF[*3*4*6]	0	0: No recording FIFO overflow occurred. 1: Recording FIFO overflow occurred.
11	Unused	x	
12	REC_FIFO_EMPTY[*8]	1	0: Recording FIFO is not empty. 1: Recording FIFO is empty.
13	REC_FIFO_HFULL[*8]	0	0: Recording FIFO is not half full. 1: Recording FIFO is half full.
14	REC_FIFO_FULL[*8]	0	0: Recording FIFO is not full. 1: Recording FIFO is full.
15	Unused	x	

<<Cautions>>

1. For information about the actions of the PLAY_DOING and REC_DOING bits, refer to the Cautions concerning PLAY and REC in Section 6.2.6.2, "Audio Control Register (AUDIO_CONT GCS2:0000-5000H) [Read/Write]."
2. A playing FIFO underflow occurs when the playing FIFO contains no data to be sent to the μPD63310. A recording FIFO underflow occurs when the recording FIFO is read even though the recording FIFO contains no data.
3. A playing FIFO overflow occurs when data is written to the playing FIFO even though the playing FIFO is full. A recording FIFO overflow occurs when data sent from the μPD63310 is written to the recording FIFO even though the recording FIFO is full.
4. Since a FIFO has an 8-bit configuration, the occurrence of an underflow or overflow is detected in terms of byte units.
5. PLAY_UNDF or PLAY_ORVF is cleared when "1" is written to the relevant bit of this register. Also, the signal is cleared when the playing circuits are reset. (See Section 6.2.6.9, "Audio Control Register 2 (AUDIO_CONT2 GCS2:0000-5070H) [Read/Write].")
   The PLAY-LED lights as follows according to the states of PLAY_DOING, PLAY_UNDF, and PLAY_ORVF.

PLAY_DOING	PLAY_UNDF	PLAY_ORVF	PLAY-LED
0	0	0	Turns off
1	0	0	Green
x	1	0	Red
x	x	1	Orange

6. REC_UNDF or REC_ORVF is cleared when "1" is written to the relevant bit of this register. Also, the signal is cleared when the recording circuits are reset. (See Section 6.2.6.9, "Audio Control Register 2 (AUDIO_CONT2 GCS2:0000-5070H) [Read/Write].")
   The REC-LED lights as follows according to the states of REC_DOING, REC_UNDF, and REC_ORVF.

REC_DOING	REC_UNDF	REC_ORVF	REC-LED
0	0	0	Turns off
1	0	0	Green
x	1	0	Red
x	x	1	Orange

7. PLAY_FIFO_EMPTY becomes "1" when the FIFO does not contain at least one byte of data.
   PLAY_FIFO_FULL becomes "1" when the amount of empty space in the FIFO is less than two bytes relative to the maximum size of the playing FIFO, which was set for AUDIO_FIFO_FULL_LEVEL. (See Section 6.2.6.6, "Audio FIFO_FULL_LEVEL Setting Register (AUDIO_FIFO_FULL_LEVEL GCS2:0000-5040H) [Read/Write].")
   PLAY_FIFO_HFULL becomes active when the number of data items in the FIFO is greater than or equal to the playing FIFO's Half-Full value, which was set for AUDIO_FIFO_HALF_LEVEL. (See Section 6.2.6.7, "Audio FIFO_HALF_LEVEL Setting Register (AUDIO_FIFO_HALF_LEVEL GCS2:0000-5050H) [Read/Write].")
   PLAY_FIFO_HFULL is used as a criterion for transferring multiple data in a batch such as for a block transfer when data is transferred under software control without using a DMA transfer.
8. REC_FIFO_EMPTY becomes "1" when the FIFO contains less than two bytes of data.
   REC_FIFO_FULL becomes "1" when the amount of empty space in the FIFO is less than one byte relative to the maximum size of the recording FIFO, which was set for AUDIO_FIFO_FULL_LEVEL. (See Section 6.2.6.6, "Audio FIFO_FULL_LEVEL Setting Register (AUDIO_FIFO_FULL_LEVEL GCS2:0000-5040H) [Read/Write].")
   REC_FIFO_HFULL becomes active when the number of data items in the FIFO is greater than or equal to the recording FIFO's Half-Full value, which was set for AUDIO_FIFO_HALF_LEVEL. (See Section 6.2.6.7, "Audio FIFO_HALF_LEVEL Setting Register (AUDIO_FIFO_HALF_LEVEL GCS2:0000-5050H) [Read/Write].")
   REC_FIFO_HFULL is used as a criterion for transferring multiple data in a batch such as for a block transfer when data is transferred under software control without using a DMA transfer.

6.2.6.4.Audio MCLKDIV Setting Register (AUDIO_MCLKDIV GCS2:0000-5020H) [Read/Write]

Bit	Signal name	RST	Function
0	MCLK_DIV0	0	The MCLK frequency to be input to the *m*PD63310 is determined by MCLK_DIV[4:0].*1
1	MCLK_DIV1	1	
2	MCLK_DIV2	0	
3	MCLK_DIV3	0	
4	MCLK_DIV4	0	
5	Unused	x	
6	Unused	x	
7	Unused	x	
8	Unused	x	
9	Unused	x	
10	Unused	x	
11	Unused	x	
12	Unused	x	
13	Unused	x	
14	Unused	x	
15	Unused	x	

<<Cautions>>

1. The following table shows the relationship of the values set for MCLK_DIV[4:0] and the MCLK frequency and sampling rate.

MCLK_DIV[4:0]	MCLK (MHz) 49.152MHz / (DIV+2)	Sampling cycle (KHz) fs = MCLK/256	Transferred bytes/second fs*4
[0,0,0,0,0]	24.576MHz		
[0,0,0,1,0]	16.384MHz		
[0,0,0,1,0]	12.288MHz	48.0KHz	192.0KB/S
[0,0,0,1,1]	9.830MHz	38.4KHz	153.6KB/S
[0,0,1,0,0]	8.192MHz	32.0KHz	128.0KB/S
[0,0,1,0,1]	7.022MHz	27.5KHz	109.7KB/S
[0,0,1,1,0]	6.144MHz	24.0KHz	96.0KB/S
[0,0,1,1,1]	5.461MHz	21.3KHz	85.3KB/S
[0,1,0,0,0]	4.915MHz	19.2KHz	76.8KB/S
[0,1,0,0,1]	4.468MHz	17.5KHz	69.8KB/S
[0,1,0,1,0]	4.096MHz	16.0KHz	64.0KB/S
[0,1,0,1,1]	3.780MHz	14.8KHz	59.1KB/S
[0,1,1,0,0]	3.511MHz	13.7KHz	54.9KB/S
[0,1,1,0,1]	3.277MHz	12.8KHz	51.2KB/S
[0,1,1,1,0]	3.072MHz	12.0KHz	48.0KB/S
[0,1,1,1,1]	2.891MHz	11.3KHz	45.2KB/S
[1,0,0,0,0]	2.731MHz	10.7KHz	42.7KB/S
[1,0,0,0,1]	2.587MHz	10.1KHz	40.4KB/S
[1,0,0,1,0]	2.458MHz	9.6KHz	38.4KB/S
[1,0,0,1,1]	2.341MHz	9.1KHz	36.6KB/S
[1,0,1,0,0]	2.234MHz	8.7KHz	34.9KB/S
[1,0,1,0,1]	2.137MHz	8.3KHz	33.4KB/S
[1,0,1,1,0]	2.048MHz	8.0KHz	32.0KB/S
[1,0,1,1,1]	1.966MHz	7.7KHz	30.7KB/S
[1,1,0,0,0]	1.890MHz	7.4KHz	29.5KB/S
[1,1,0,0,1]	1.820MHz	7.1KHz	28.4KB/S
[1,1,0,1,0]	1.755MHz	6.9KHz	27.4KB/S
[1,1,0,1,1]	1.695MHz	6.6KHz	26.5KB/S
[1,1,1,0,0]	1.638MHz	6.4KHz	25.6KB/S
[1,1,1,0,1]	1.586MHz	6.2KHz	24.8KB/S
[1,1,1,1,0]	1.536MHz	6.0KHz	24.0KB/S
[1,1,1,1,1]	1.489MHz	5.8KHz	23.3KB/S

6.2.6.5.Audio FIFO (AUDIO_FIFO GCS2:0000-5030H) [Read/Write]

This is a register for accessing the playing and recording FIFOs. Writing to this register will be writing to the playing FIFO. Reading from this register will be reading from the recording FIFO. Reading and writing should be performed according to 16-bit accesses.
An odd-numbered access to this register will be data for the left channel, and an even-numbered access will be data for the right channel. Therefore, one set of (32-bit) stereo data is transferred by using two accesses.
Also, to use DMA to transfer data, specify this register as the address to be DMA transferred.
Data that is written during an overflow state is ignored.

6.2.6.6.Audio FIFO_FULL_LEVEL Setting Register (AUDIO_FIFO_FULL_LEVEL GCS2:0000-5040H) [Read/Write]

Bit	Signal name	RST	Function
0	PLAY_FIFO_FULL_LEVEL0	1	The size (maximum number of bytes) of the playing FIFO is set in PLAY_FIFO_FULL_LEVEL[7:0]. *1*2*3 [Playing FIFO size (bytes)] = PLAY_FIFO_FULL_LEVEL[7:0] + 1
1	PLAY_FIFO_FULL_LEVEL1	0	
2	PLAY_FIFO_FULL_LEVEL2	1	
3	PLAY_FIFO_FULL_LEVEL3	1	
4	PLAY_FIFO_FULL_LEVEL4	1	
5	PLAY_FIFO_FULL_LEVEL5	1	
6	PLAY_FIFO_FULL_LEVEL6	1	
7	PLAY_FIFO_FULL_LEVEL7	1	
8	REC_FIFO_FULL_LEVEL0	1	The size (maximum number of bytes) of the recording FIFO is set in REC_FIFO_FULL_LEVEL[7:0]. *1*2*3 [Recording FIFO size (bytes)] = REC_FIFO_FULL_LEVEL[7:0] + 1
9	REC_FIFO_FULL_LEVEL1	0	
10	REC_FIFO_FULL_LEVEL2	1	
11	REC_FIFO_FULL_LEVEL3	1	
12	REC_FIFO_FULL_LEVEL4	1	
13	REC_FIFO_FULL_LEVEL5	1	
14	REC_FIFO_FULL_LEVEL6	1	
15	REC_FIFO_FULL_LEVEL7	1	

<<Cautions>>

1. The values set in PLAY_FIFO_FULL_LEVEL[7:0] and REC_FIFO_FULL_LEVEL[7:0] must be odd numbers.
2. The minimum setting value for PLAY_FIFO_FULL_LEVEL[7:0] and REC_FIFO_FULL_LEVEL[7:0] is 01H, and the maximum setting value is FDH. Also, the following equation shows the relationship between the setting value and FIFO size (bytes).
   [FIFO size (bytes)] = [Setting value in xxx_FIFO_FULL_LEVEL[7:0]] + 1
   Therefore, the minimum size of a FIFO is 2 bytes, and the maximum size is 254 bytes.
3. The PLAY_FIFO_FULL, REC_FIFO_FULL, PLAY_ORVF, and REC_ORVF signals of the AUDIO_STATUS register are generated according to the value set in this register.

6.2.6.7.Audio FIFO_HALF_LEVEL Setting Register (AUDIO_FIFO_HALF_LEVEL GCS2:0000-5050H) [Read/Write]

Bit	Signal name	RST	Function
0	PLAY_FIFO_HALF_LEVEL0	1	The Half-Full value (bytes) of the playing FIFO is set in PLAY_FIFO_HALF_LEVEL[7:0]. *1*2 [Playing FIFO Half-Full value (bytes)] = PLAY_FIFO_HALF_LEVEL[7:0] + 1
1	PLAY_FIFO_HALF_LEVEL1	1	
2	PLAY_FIFO_HALF_LEVEL2	1	
3	PLAY_FIFO_HALF_LEVEL3	1	
4	PLAY_FIFO_HALF_LEVEL4	1	
5	PLAY_FIFO_HALF_LEVEL5	1	
6	PLAY_FIFO_HALF_LEVEL6	1	
7	PLAY_FIFO_HALF_LEVEL7	0	
8	REC_FIFO_HALF_LEVEL0	1	The Half-Full value (bytes) of the recording FIFO is set in REC_FIFO_HALF_LEVEL[7:0]. *1*2 [Recording FIFO Half-Full (bytes)] = REC_FIFO_HALF_LEVEL[7:0] + 1
9	REC_FIFO_HALF_LEVEL1	1	
10	REC_FIFO_HALF_LEVEL2	1	
11	REC_FIFO_HALF_LEVEL3	1	
12	REC_FIFO_HALF_LEVEL4	1	
13	REC_FIFO_HALF_LEVEL5	1	
14	REC_FIFO_HALF_LEVEL6	1	
15	REC_FIFO_HALF_LEVEL7	0	

<<Cautions>>

1. The minimum setting value for PLAY_FIFO_HALF_LEVEL[7:0] and REC_FIFO_HALF_LEVEL[7:0] is 01H, and the maximum setting value is FDH. Also, the following equation shows the relationship between the setting value and FIFO Half_Full value (bytes).
   [FIFO Half-Full value (bytes)] = [Setting value in xxx_FIFO_HALF_LEVEL[7:0]] + 1
   Therefore, the minimum Half-Full value of a FIFO is 2 bytes, and the maximum Half-Full value is 254 bytes.
2. The PLAY_FIFO_HFULL and REC_FIFO_HFULL signals of the AUDIO_STATUS register are generated according to the value set in this register.

6.2.6.8.Audio FIFO_DEPTH Register (AUDIO_FIFO_DEPTH GCS2:0000-5060H) [Read Only]

Bit	Signal name	RST	Function
0	PLAY_FIFO_DEPTH0	1	This is the number of bytes of data that are entered in the playing FIFO. *1 [Number of remaining bytes in playing FIFO] = PLAY_FIFO_DEPTH[7:0] + 1
1	PLAY_FIFO_DEPTH1	1	
2	PLAY_FIFO_DEPTH2	1	
3	PLAY_FIFO_DEPTH3	1	
4	PLAY_FIFO_DEPTH4	1	
5	PLAY_FIFO_DEPTH5	1	
6	PLAY_FIFO_DEPTH6	1	
7	PLAY_FIFO_DEPTH7	1	
8	REC_FIFO_DEPTH0	1	This is the number of bytes of data that are entered in the recording FIFO. *1 [Number of remaining bytes in recording FIFO] = REC_FIFO_DEPTH[7:0] + 1
9	REC_FIFO_DEPTH1	1	
10	REC_FIFO_DEPTH2	1	
11	REC_FIFO_DEPTH3	1	
12	REC_FIFO_DEPTH4	1	
13	REC_FIFO_DEPTH5	1	
14	REC_FIFO_DEPTH6	1	
15	REC_FIFO_DEPTH7	1	

<<Cautions>>

1. The following equation shows the relationship between the number of bytes of data remaining in the FIFO and PLAY_FIFO_DEPTH[7:0] and REC_FIFO_DEPTH[7:0].
   [FIFO remaining bytes] = [Setting value in xxx_FIFO_DEPTH[7:0]] + 1
   However, when the value of xxx_FIFO_DEPTH[7:0] is FFH, it indicates that no data is remaining in the FIFO.

6.2.6.9.Audio Control Register 2 (AUDIO_CONT2 GCS2:0000-5070H) [Read/Write]

Bit	Signal name	RST	Function
0	PLAY_HFULL_INTEN*1	0	0: Do not request an interrupt when the number of data values in the playing FIFO is less than the Half-Full value. 1: Request an interrupt when the number of data values in the playing FIFO is less than the Half-Full value.
1	Unused	x	
2	Unused	x	
3	Unused	x	
4	PLAY_RESET*2	0	0: Do not reset the playing circuits. 1: Reset the playing circuits.
5	Unused	x	
6	Unused	x	
7	Unused	x	
8	REC_HFULL_INTEN*3	0	0: Do not request an interrupt when the number of data values in the recording FIFO is greater than or equal to the Half-Full value. 1: Request an interrupt when the number of data values in the recording FIFO is greater than or equal to the Half-Full value.
9	Unused	x	
10	Unused	x	
11	Unused	x	
12	REC_RESET*2	0	0: Do not reset the recording circuits. 1: Reset the recording circuits.
13	Unused	x	
14	Unused	x	
15	BUS8_16-*4	0	0: Access the FIFO by using a 16-bit width. 1: Access the FIFO by using an 8-bit width (setting prohibited).

<<Cautions>>

1. When PLAY_HFULL_INTEN is set to "1," an interrupt will be generated if the number of data items in the playing FIFO becomes less than the value set in PLAY_FIFO_HALF_LEVEL[7:0]. This interrupt is generated corresponding to the FIFO status when audio is not being played.
2. When PLAY_RESET is set to "1," only the playing circuits are reset under hardware control. Also, if REC_RESET is set to "1," only the recording circuits are reset under hardware control. The μPD63310 is not reset by PLAY_RESET and REC_RESET. The *m*PD63310 can be reset by the AUDIO_RESET signal of the AUDIO_CONT register. (See Section 6.2.6.2, "Audio Control Register (AUDIO_CONT GCS2:0000-5000H) [Read/Write]."
   If the setting of PLAY_RESET or REC_RESET is not returned to "0" after it has been set to "1," the reset state is not canceled.
   The following table shows the relationship between the various reset bits and the circuits that are reset.

Control bit	*m*PD63310	Playing circuits	Recording circuits
AUDIO_RESET of AUDIO_CONT	Reset	Reset	Reset
PLAY_RESET of AUDIO_CONT2	---	Reset	---
REC_RESET of AUDIO_CONT2	---	---	Reset

3. When REC_HFULL_INTEN is set to "1," an interrupt will be generated if the number of data items in the recording FIFO becomes greater than or equal to the value set in REC_FIFO_HALF_LEVEL[7:0]. This interrupt is generated corresponding to the FIFO status when audio is not being recorded.
4. BUS8_16- should not be set to "1."

6.2.6.10.Audio Status Register 2 (AUDIO_STATUS2 GCS2:0000-5080H) [Read Only]

Bit	Signal name	RST	Function
0	Unused	x	
1	PLAY_UNDF2*1	0	0: No playing FIFO underflow occurred. 1: Playing FIFO underflow occurred.
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	
8	Unused	x	
9	Unused	x	
10	REC_ORVF2*2	0	0: No recording FIFO overflow occurred. 1: Recording FIFO overflow occurred.
11	Unused	x	
12	Unused	x	
13	Unused	x	
14	Unused	x	
15	Unused	x	

<<Cautions>>

1. PLAY_UNDF2 and REC_ORVF2 are the same as PLAY_UNDF and REC_ORVF of the audio status register, respectively.

6.2.7. Interrupt Control Circuits

The interrupt control circuits combine all interrupts that are generated on the motherboard to finally consolidate them at GINTO-[3:0].

6.2.7.1.Overview of Interrupt Resources

The following table lists the types of interrupts on the motherboard and describes controls for the interrupt control circuits.
The "Polarity control" column in the table indicates function that can correspond to the interrupt line being either High active/Low active or rising edge/falling edge.
Also, "Level/edge selection" indicates function that can correspond to the interrupt line being either level sensitive or edge sensitive.

Interrupt resource	Polarity control	Level/edge selection	Remark
TL16PIR552 UART0	No	No (level)	Interrupt is generated when the INTRPT0 pin of the TL16PIR552 is High.
TL16PIR552 UART1	No	No (level)	Interrupt is generated when the INTRPT1 pin of the TL16PIR552 is High.
TL16PIR552 PRINTER	Yes	Yes	The PINTR- pin of the TL16PIR552 is directly connected.
AUDIO	No	No (level)	Logical OR operation of all interrupts of the audio circuits.
PCI9080 LINTo	No	No (level)	Interrupt is generated when the LINTo- pin of the PCI9080 is Low.
GBUS GINTI0-	Yes	Yes	The GINTI0- pin of the GBUS is directly connected (see below).
GBUS GINTI1-	Yes	Yes	The GINTI1- pin of the GBUS is directly connected (see below).
ISA INTR interrupt	No	No (level)	INTR interrupt of the ISA-BUS (see below).
ISA NMI interrupt	No	No (level)	Interrupt is generated when the NMI pin of the M1523B (SouthBridge) is High.
SB82558 (LAN) INTA	Yes	Yes	The INTA- pin of the SB82558 is directly connected (see below).
PCI9080 LSERR	No	No (level)	Interrupt is generated when the LSERR- pin of the PCI9080 is Low.
PCI bus Parity Error	No	No (edge)	Interrupt due to the occurrence of a parity error on the PCI bus.
Timeover Ready	No	No (edge)	Interrupt is generated when a time-over ready is generated.
PCI slot #1 INTA	Yes	Yes	INTA- of the PCI slot #1 is directly connected (see below).
PCI slot #1 INTB	Yes	Yes	INTB- of the PCI slot #1 is directly connected (see below).
PCI slot #1 INTC	Yes	Yes	INTC- of the PCI slot #1 is directly connected (see below).
PCI slot #1 INTD	Yes	Yes	INTD- of the PCI slot #1 is directly connected (see below).
PCI slot #2 INTA	Yes	Yes	INTA- of the PCI slot #2 is directly connected (see below).
PCI slot #2 INTA	Yes	Yes	INTA- of the PCI slot #2 is directly connected (see below).
PCI slot #2 INTB	Yes	Yes	INTB- of the PCI slot #2 is directly connected (see below).
PCI slot #2 INTC	Yes	Yes	INTC- of the PCI slot #2 is directly connected (see below).
PCI slot #2 INTD	Yes	Yes	INTD- of the PCI slot #2 is directly connected (see below).
EXT-BUS EXT_INT0	Yes	Yes	EXT_INT0- pin of the EXT-BUS is directly connected.
EXT-BUS EXT_INT1	Yes	Yes	EXT_INT1- pin of the EXT-BUS is directly connected.
EXT-BUS EXT_INT2	Yes	Yes	EXT_INT2- pin of the EXT-BUS is directly connected.
EXT-BUS EXT_INT3	Yes	Yes	EXT_INT3- pin of the EXT-BUS is directly connected.
USB0 Over Current	Yes	Yes	Pin is Low when a current overflow occurs for USB channel 0.
USB1 Over Current	Yes	Yes	Pin is Low when a current overflow occurs for USB channel 1.
PCMCIA Over Current	Yes	Yes	Pin is Low when a PCMCIA current overflow occurs.

OUT0 and OUT1 of the μPD71054 (timer), which is normally installed on the CPU board, are connected to the GINTI0- and GINTI1- pins of the GBUS.

The ISA INTR interrupt is supplied to the CPU board as follows.

1) When the internal interrupt controller (i8259-compatible) of the M1523B (SouthBridge) detects an interrupt, the INTR pin of the M1523B becomes active.
2) When the motherboard circuit detects that the INTR pin of the M1523B is active, it causes an interrupt acknowledge cycle to be generated on the PCI bus to obtain the interrupt vector from the M1523B.
3) When the motherboard circuit has obtained the interrupt vector, it requests an ISA_INTR interrupt for the interrupt controller of the motherboard.
4) When ISA_INTR is asserted, the interrupt controller of the motherboard asserts GINTOx- according to the interrupt controller settings.

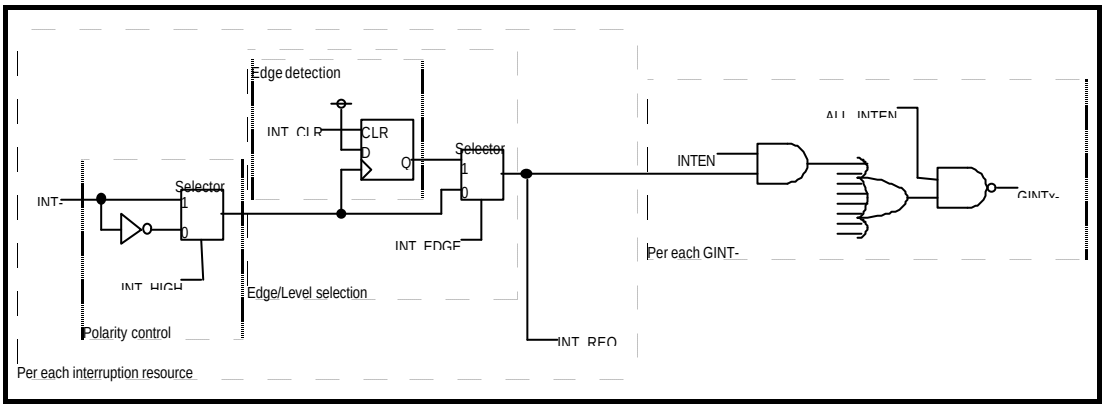
The CPU board performs the following processing when an ISA INTR interrupt is generated.

1] It reads the interrupt vector and performs the relevant interrupt handling.
2] It performs End Of Interrupt (EOI) processing for the interrupt controller of the M1523B.
3] It performs ISA_INTR clear processing for the interrupt controller of the motherboard.

If the M1523B is still asserting the INTR pin at the stage when the ISA_INTR clear processing is performed, processing returns to 2) and continues.

The SB82558 (LAN controller) interrupts and PCI bus slot interrupts are conventional level-sensitive interrupts, which are generated when the level is Low. Therefore, this mode normally is used. However, the mode can be switched to edge-sensitive mode or the polarity can be selected in order to take into consideration cases that cannot correspond to a level-sensitive interrupt.

The various interrupts are organized as shown in the following figure.



Interrupt requests from the various resources pass through the polarity control circuit, edge detection circuit, and edge/level selection circuit required for each interrupt resource. Then, a logical OR operation is performed on the interrupt requests of all interrupt resources. Finally, an ALL_MASK and logical AND operation are performed, and the result is connected to GINT-[3:0].
The ALL_MASK function is provided to support multiple interrupts when GINT-[3:0] are connected to edge-sensitive interrupt request pins of the CPU. An edge can be generated for GINT-[3:0] by masking ALL_MASK at the end of interrupt handling for these kinds of cases.

6.2.7.2.Interrupt Status Register 0 (INT_STATUS0 GCS2:0000-6000H) [Read Only]

Bit	Signal name	RST	Function
0	Unused	x	
1	UART0_INTRQ	0	0: There is no interrupt request from the TL16PRI552 UART0 pin. 1: There is an interrupt request from the TL16PRI552 UART0 pin.
2	UART1_INTRQ	0	0: There is no interrupt request from the TL16PRI552 UART1 pin. 1: There is an interrupt request from the TL16PRI552 UART1 pin.
3	PRT_INTRQ	0	0: There is no interrupt request from the TL16PRI552 PRINTER pin. 1: There is an interrupt request from the TL16PRI552 PRINTER pin.
4	AUDIO_INTRQ*2	0	0: There is no interrupt request from the AUDIO pin. 1: There is an interrupt request from the AUDIO pin.
5	P9_LINT_INTRQ*3	0	0: There is no interrupt request due to the PCI9080 LINTo- pin. 1: There is an interrupt request due to the PCI9080 LINTo- pin.
6	GINTI0_INTRQ	0	0: There is no interrupt request due to the GBUS GINTI0- pin. 1: There is an interrupt request due to the GBUS GINTI0- pin.
7	GINTI1_INTRQ	0	0: There is no interrupt request due to the GBUS GINTI1- pin. 1: There is an interrupt request due to the GBUS GINTI1- pin.
8	ISAINTR_INTRQ*4	0	0: There is no interrupt request due to the ISA bus INTR pin. 1: There is an interrupt request due to the ISA bus INTR pin.
9	ISANMI_INTRQ*5	0	0: There is no interrupt request due to the ISA bus NMI pin. 1: There is an interrupt request due to the ISA bus NMI pin.
10	LAN_INTRQ	0	0: There is no interrupt request due to the SB82558 INTA- pin. 1: There is an interrupt request due to the SB82558 INTA- pin.
11	P9_LSERR_INTRQ*6*7	0	0: There is no interrupt request due to the PCI9080 LSERR- pin. 1: There is an interrupt request due to the PCI9080 LSERR- pin.
12	PCI_PERR_INTRQ*7	0	0: There is no interrupt request due to the occurrence of a parity error on the PCI bus. 1: There is an interrupt request due to the occurrence of a parity error on the PCI bus.
13	TOVRDY_INTRQ	0	0: There is no interrupt request due to the occurrence of a time-over ready. 1: There is an interrupt request due to the occurrence of a time-over ready.
14	ABORT_ERR_INTRQ*8	0	0: There is no interrupt request due to the occurrence of an abort termination. 1: There is an interrupt request due to the occurrence of an abort termination.
15	BACKOFF_ERR_INTRQ*8	0	0: There is no interrupt request due to the occurrence of a back off. 1: There is an interrupt request due to the occurrence of a back off.

<<Cautions>>

1. All INTRQ signals are common to all GINTO-[3:0] pins. Also, the INTRQ signals that are read from this register are the signals before they are masked by the controller. Therefore, to specify the resource that is requesting the interrupt for GINTO-[3:0], a logical AND operation must be performed on the contents of the interrupt status registers and the contents of the interrupt enable registers that are set individually for the GINTO-[3:0] pins.
2. An interrupt request from the AUDIO pin may be due to multiple interrupt sources such as a FIFO overflow or underflow.
3. An interrupt request due to the PCI9080 LINTo- pin is the result of multiple interrupt sources combined within the PCI9080. (See Section 6.3.4, "PCI9080.")
4. An interrupt request due to the ISA bus INTR pin is the source of an interrupt that is requested for the CPU INTR pin in a PC/AT-compatible (DOS/V) machine. Interrupt sources are combined within the M1523B (SouthBridge) chip. (See Section 6.3.8, "M1523B (SouthBridge).")
5. An interrupt request due to the ISA bus NMI pin is the source of an interrupt that is requested for the CPU NMI pin in a PC/AT-compatible (DOS/V) machine. Interrupt sources are combined within the M1523B (SouthBridge) chip. (See Section 6.3.8, "M1523B (SouthBridge).")
6. An interrupt request due to the PCI9080 LSERR- pin is the result of multiple interrupt sources combined within the PCI9080. (See Section 6.3.4, "PCI9080.")
7. PCI_PERR_INTRQ is an interrupt request that is generated when a parity error is reported according to the PERR# signal on the PCI bus. Since the PCI bus master may replace a parity error with an SERR# signal on the PCI bus, PCI_PERR_INTRQ may be generated together with P9_LSERR_INTRQ.
8. ABORT_ERR_INTRQ and BACKOFF_ERR_INTRQ are errors that are generated in relation to

PCI bus access. For details, see Section 6.3.4.6, "Abort Error" and Section 6.3.4.7, "Back Off Error."

6.2.7.3.Interrupt Clear Register 0 (INT_CLEAR0 GCS2:0000-6010H) [Write Only]

Bit	Signal name	RST	Function
0	Unused	x	
1	Unused	x	
2	Unused	x	
3	PRT_INTCLR	x	0: Do not clear the interrupt request from the TL16PRI552 PRINTER pin. 1: Clear the interrupt request from the TL16PRI552 PRINTER pin.
4	Unused	x	
5	Unused	x	
6	GINTI0_INTCLR	x	0: Do not clear the interrupt request from the GBUS GINTI0- pin. 1: Clear the interrupt request from the GBUS GINTI0- pin.
7	GINTI1_INTCLR	x	0: Do not clear the interrupt request from the GBUS GINTI1- pin. 1: Clear the interrupt request from the GBUS GINTI1- pin.
8	ISAINTR_INTCLR*2	x	0: Do not clear the interrupt request due to the ISA bus INTR pin. 1: Clear the interrupt request due to the ISA bus INTR pin.
9	Unused	x	
10	LAN_INTCLR	x	0: Do not clear the interrupt request due to the SB82558 INTA- pin. 1: Clear the interrupt request due to the SB82558 INTA- pin.
11	Unused	x	
12	PCI_PERR_INTCLR	x	0: Do not clear the interrupt request due to the occurrence of a parity error on the PCI bus. 1: Clear the interrupt request due to the occurrence of a parity error on the PCI bus.
13	TOVRDY_INTCLR	x	0: Do not clear the interrupt request due to the occurrence of a time-over ready. 1: Clear the interrupt request due to the occurrence of a time-over ready.
14	ABORT_ERR_INTCLR	x	0: Do not clear the interrupt request due to the occurrence of an abort termination. 1: Clear the interrupt request due to the occurrence of an abort termination.
15	BACKOFF_ERR_INTCLR	x	0: Do not clear the interrupt request due to the occurrence of a back off. 1: Clear the interrupt request due to the occurrence of a back off.

<<Cautions>>

1. INTCLR is a bit for clearing an interrupt request that is being maintained by an edge detection circuit. When a level-mode interrupt is selected for an interrupt resource for which an edge-mode or level-mode interrupt can be selected, INTRQ is not cleared even if "1" is written to the relevant bit of this register.
2. End Of Interrupt (EOI) processing should be performed for the interrupt controller within the M1523B (SouthBridge) chip before "1" is written to ISAINTR_INTCLR. If an interrupt request from the M1523B is active when "1" is written to ISAINTR_INTCLR, the hardware will generate an interrupt acknowledge cycle on the PCI bus to obtain the vector. ISAINTR_INTRQ will become active again when the vector is obtained.
   When the EOI is issued for this M1523B, the resources on the PCI (such as the mask register of the interrupt controller within the M1523B) should be read while "1" is written to ISAINTR_INTCLR. This should be done to avoid the following kind of problem. It may take some time until the interrupt line that is output from the M1523B becomes inactive after the EOI is issued for the M1523B. Therefore, if "1" is written to ISAINTR_INTCLR immediately after the EOI is issued, the interrupt may end up being requested again due to the active state of the M1523B interrupt line before it becomes inactive due to the EOI.

6.2.7.4.Interrupt Edge Specification Register 0 (INT_EDGE0 GCS2:0000-6020H) [Read/Write]

Bit	Signal name	RST	Function
0	Unused	x	
1	Unused	x	
2	Unused	x	
3	PRT_INTEDGE*2	0	0: The interrupt request from the TL16PRI552 PRINTER pin is a level-mode request. 1: The interrupt request from the TL16PRI552 PRINTER pin is an edge-mode request.
4	Unused	x	
5	Unused	x	
6	GINTI0_INTEDGE	0	0: The interrupt request from the GBUS GINTI0- pin is a level-mode request. 1: The interrupt request from the GBUS GINTI0- pin is an edge-mode request.
7	GINTI1_INTEDGE	0	0: The interrupt request from the GBUS GINTI1- pin is a level-mode request. 1: The interrupt request from the GBUS GINTI1- pin is an edge-mode request.
8	Unused	x	
9	Unused	x	
10	LAN_INTEDGE*3	0	0: The interrupt request due to the SB82558 INTA- pin is a level-mode request. 1: The interrupt request due to the SB82558 INTA- pin is an edge-mode request.
11	Unused	x	
12	Unused	x	
13	Unused	x	
14	Unused	x	
15	Unused	x	

<<Cautions>>

1. INTEDGE is a register for setting whether an interrupt request from an interrupt resource is an edge-mode or level-mode request.
   The edge detection circuit maintains an edge detection result regardless of the setting of this register. When this register setting is switched from level mode to edge mode, the edge detection circuit should be cleared by the relevant INTCLR before the switch.
2. An interrupt from a TL16PIR552 printer uses edge mode or level mode according to the mode of the printer port.
3. Since an interrupt from the SB82558 (LAN controller) obeys PCI bus standards, level mode should be used.

6.2.7.5.Interrupt Polarity Specification Register 0 (INT_POLARITY0 GCS2:0000-6030H) [Read/Write]

Bit	Signal name	RST	Function
0	Unused	x	
1	Unused	x	
2	Unused	x	
3	PRT_INTHIGH	0	0: The interrupt request from the TL16PRI552 PRINTER pin is Low/falling edge. 1: The interrupt request from the TL16PRI552 PRINTER pin is High/rising edge.
4	Unused	x	
5	Unused	x	
6	GINTI0_INTHIGH	0	0: The interrupt request from the GBUS GINTI0- pin is Low/falling edge. 1: The interrupt request from the GBUS GINTI0- pin is High/rising edge.
7	GINTI1_INTHIGH	0	0: The interrupt request from the GBUS GINTI1- pin is Low/falling edge. 1: The interrupt request from the GBUS GINTI1- pin is High/rising edge.
8	Unused	x	
9	Unused	x	
10	LAN_INTHIGH *2	0	0: The interrupt request due to the SB82558 INTA- pin is Low/falling edge. 1: The interrupt request due to the SB82558 INTA- pin is High/rising edge.
11	Unused	x	
12	Unused	x	
13	Unused	x	
14	Unused	x	
15	Unused	x	

<<Cautions>>

1. INTHIGH sets the polarity of an interrupt request from an interrupt resource. When the INTEDGE setting is edge mode, either rising edge or falling edge can be selected according to the INTHIGH setting. When the INTEDGE setting is level mode, either Low active or High active can be selected according to the INTHIGH setting.
   When edge mode is set by INTEDGE, an edge detection circuit may end up detecting an edge according to the polarity setting changes. Therefore, when this register setting is switched, the edge detection circuit should be cleared by the relevant INTCLR after the switch.
2. Since an interrupt from the SB82558 (LAN controller) obeys PCI bus standards, Low level should be used.

6.2.7.6.Interrupt Status Register 1 (INT_STATUS1 GCS2:0000-6040H) [Read Only]

Bit	Signal name	RST	Function
0	PCI1_INTA_INTRQ	0	0: There is no interrupt request due to the PCI slot 1 INTA- pin. 1: There is an interrupt request due to the PCI slot 1 INTA- pin.
1	PCI1_INTB_INTRQ	0	0: There is no interrupt request due to the PCI slot 1 INTB- pin. 1: There is an interrupt request due to the PCI slot 1 INTB- pin.
2	PCI1_INTC_INTRQ	0	0: There is no interrupt request due to the PCI slot 1 INTC- pin. 1: There is an interrupt request due to the PCI slot 1 INTC- pin.
3	PCI1_INTD_INTRQ	0	0: There is no interrupt request due to the PCI slot 1 INTD- pin. 1: There is an interrupt request due to the PCI slot 1 INTD- pin.
4	PCI2_INTA_INTRQ	0	0: There is no interrupt request due to the PCI slot 2 INTA- pin. 1: There is an interrupt request due to the PCI slot 2 INTA- pin.
5	PCI2_INTB_INTRQ	0	0: There is no interrupt request due to the PCI slot 2 INTB- pin. 1: There is an interrupt request due to the PCI slot 2 INTB- pin.
6	PCI2_INTC_INTRQ	0	0: There is no interrupt request due to the PCI slot 2 INTC- pin. 1: There is an interrupt request due to the PCI slot 2 INTC- pin.
7	PCI2_INTD_INTRQ	0	0: There is no interrupt request due to the PCI slot 2 INTD- pin. 1: There is an interrupt request due to the PCI slot 2 INTD- pin.
8	EXT_INT0_INTRQ*2	0	0: There is no interrupt request due to the EXT bus EXT_INT0- pin. 1: There is an interrupt request due to the EXT bus EXT_INT0- pin.
9	EXT_INT1_INTRQ	0	0: There is no interrupt request due to the EXT bus EXT_INT1- pin. 1: There is an interrupt request due to the EXT bus EXT_INT1- pin.
10	EXT_INT2_INTRQ	0	0: There is no interrupt request due to the EXT bus EXT_INT2- pin. 1: There is an interrupt request due to the EXT bus EXT_INT2- pin.
11	EXT_INT3_INTRQ	0	0: There is no interrupt request due to the EXT bus EXT_INT3- pin. 1: There is an interrupt request due to the EXT bus EXT_INT3- pin.
12	USB0_OC_INTRQ	0	0: There is no interrupt request due to a USB0 over current. 1: There is an interrupt request due to a USB0 over current.
13	USB1_OC_INTRQ	0	0: There is no interrupt request due to a USB1 over current. 1: There is an interrupt request due to a USB1 over current.
14	PCMCIA_OC_INTRQ	0	0: There is no interrupt request due to a PCMCIA over current. 1: There is an interrupt request due to a PCMCIA over current.
15	Unused	x	

<<Cautions>>

1. All INTRQ signals are common to all GINTO-[3:0] pins. Also, the INTRQ signals that are read from this register are the signals before they are masked by the controller. Therefore, to specify the resource that is requesting the interrupt for GINTO-[3:0], a logical AND operation must be performed on the contents of the interrupt status registers and the contents of the interrupt enable registers that are set individually for the GINTO-[3:0] pins.
2. When the 16-bit EXT-BUS (JEXT16 connector) is used, the EXT-BUS interrupt is connected to EXT_INT0.

6.2.7.7. Interrupt Clear Register 1 (INT_CLEAR1 GCS2:0000-6050H) [Write Only]

Bit	Signal name	RST	Function
0	PCI1_INTA_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 1 INTA- pin. 1: Clear an interrupt request due to the PCI slot 1 INTA- pin.
1	PCI1_INTB_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 1 INTB- pin. 1: Clear an interrupt request due to the PCI slot 1 INTB- pin.
2	PCI1_INTC_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 1 INTC- pin. 1: Clear an interrupt request due to the PCI slot 1 INTC- pin.
3	PCI1_INTD_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 1 INTD- pin. 1: Clear an interrupt request due to the PCI slot 1 INTD- pin.
4	PCI2_INTA_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 2 INTA- pin. 1: Clear an interrupt request due to the PCI slot 2 INTA- pin.
5	PCI2_INTB_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 2 INTB- pin. 1: Clear an interrupt request due to the PCI slot 2 INTB- pin.
6	PCI2_INTC_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 2 INTC- pin. 1: Clear an interrupt request due to the PCI slot 2 INTC- pin.
7	PCI2_INTD_INTCLR	x	0: Do not clear an interrupt request due to the PCI slot 2 INTD- pin. 1: Clear an interrupt request due to the PCI slot 2 INTD- pin.
8	EXT_INT0_INTCLR *2	x	0: Do not clear an interrupt request due to the EXT bus EXT_INT0- pin. 1: Clear an interrupt request due to the EXT bus EXT_INT0- pin.
9	EXT_INT1_INTCLR	x	0: Do not clear an interrupt request due to the EXT bus EXT_INT1- pin. 1: Clear an interrupt request due to the EXT bus EXT_INT1- pin.
10	EXT_INT2_INTCLR	x	0: Do not clear an interrupt request due to the EXT bus EXT_INT2- pin. 1: Clear an interrupt request due to the EXT bus EXT_INT2- pin.
11	EXT_INT3_INTCLR	x	0: Do not clear an interrupt request due to the EXT bus EXT_INT3- pin. 1: Clear an interrupt request due to the EXT bus EXT_INT3- pin.
12	USB0_OC_INTCLR	x	0: Do not clear an interrupt request due to a USB0 over current. 1: Clear an interrupt request due to a USB0 over current.
13	USB1_OC_INTCLR	x	0: Do not clear an interrupt request due to a USB1 over current. 1: Clear an interrupt request due to a USB1 over current.
14	PCMCIA_OC_INTCLR	x	0: Do not clear an interrupt request due to a PCMCIA over current. 1: Clear an interrupt request due to a PCMCIA over current.
15	Unused	x	

<<Cautions>>

1. INTCLR is a bit for clearing an interrupt request that is being maintained by an edge detection circuit. When a level-mode interrupt is selected for an interrupt resource for which an edge-mode or level-mode interrupt can be selected, INTRQ is not cleared even if "1" is written to the relevant bit of this register.
2. When the 16-bit EXT-BUS (JEXT16 connector) is used, the EXT-BUS interrupt is connected to EXT_INT0.

6.2.7.8.Interrupt Edge Specification Register 1 (INT_EDGE1 GCS2:0000-6060H) [Read/Write]

Bit	Signal name	RST	Function
0	PCI1_INTA_INTEDGE*2	0	0: An interrupt request due to the PCI slot 1 INTA- pin is a level-mode request. 1: An interrupt request due to the PCI slot 1 INTA- pin is an edge-mode request.
1	PCI1_INTB_INTEDGE*2	0	0: An interrupt request due to the PCI slot 1 INTB- pin is a level-mode request. 1: An interrupt request due to the PCI slot 1 INTB- pin is an edge-mode request.
2	PCI1_INTC_INTEDGE*2	0	0: An interrupt request due to the PCI slot 1 INTC- pin is a level-mode request. 1: An interrupt request due to the PCI slot 1 INTC- pin is an edge-mode request.
3	PCI1_INTD_INTEDGE*2	0	0: An interrupt request due to the PCI slot 1 INTD- pin is a level-mode request. 1: An interrupt request due to the PCI slot 1 INTD- pin is an edge-mode request.
4	PCI2_INTA_INTEDGE*2	0	0: An interrupt request due to the PCI slot 2 INTA- pin is a level-mode request. 1: An interrupt request due to the PCI slot 2 INTA- pin is an edge-mode request.
5	PCI2_INTB_INTEDGE*2	0	0: An interrupt request due to the PCI slot 2 INTB- pin is a level-mode request. 1: An interrupt request due to the PCI slot 2 INTB- pin is an edge-mode request.
6	PCI2_INTC_INTEDGE*2	0	0: An interrupt request due to the PCI slot 2 INTC- pin is a level-mode request. 1: An interrupt request due to the PCI slot 2 INTC- pin is an edge-mode request.
7	PCI2_INTD_INTEDGE*2	0	0: An interrupt request due to the PCI slot 2 INTD- pin is a level-mode request. 1: An interrupt request due to the PCI slot 2 INTD- pin is an edge-mode request.
8	EXT_INT0_INTEDGE*3	0	0: An interrupt request due to the EXT bus EXT_INT0- pin is a level-mode request. 1: An interrupt request due to the EXT bus EXT_INT0- pin is an edge-mode request.
9	EXT_INT1_INTEDGE	0	0: An interrupt request due to the EXT bus EXT_INT1- pin is a level-mode request. 1: An interrupt request due to the EXT bus EXT_INT1- pin is an edge-mode request.
10	EXT_INT2_INTEDGE	0	0: An interrupt request due to the EXT bus EXT_INT2- pin is a level-mode request. 1: An interrupt request due to the EXT bus EXT_INT2- pin is an edge-mode request.
11	EXT_INT3_INTEDGE	0	0: An interrupt request due to the EXT bus EXT_INT3- pin is a level-mode request. 1: An interrupt request due to the EXT bus EXT_INT3- pin is an edge-mode request.
12	USB0_OC_INTEDGE	0	0: An interrupt request due to a USB0 over current is a level-mode request. 1: An interrupt request due to a USB0 over current is an edge-mode request.
13	USB1_OC_INTEDGE	0	0: An interrupt request due to a USB1 over current is a level-mode request. 1: An interrupt request due to a USB1 over current is an edge-mode request.
14	PCMCIA_OC_INTEDGE	0	0: An interrupt request due to a PCMCIA over current is a level-mode request. 1: An interrupt request due to a PCMCIA over current is an edge-mode request.
15	Unused	x	

<<Cautions>>

1. INTEDGE is a register for setting whether an interrupt request from an interrupt resource is an edge-mode or level-mode request.
   The edge detection circuit maintains an edge detection result regardless of the setting of this register. When this register setting is switched from level mode to edge mode, the edge detection circuit should be cleared by the relevant INTCLR before the switch.
2. A PCI bus slot interrupt is normally used in level mode.
3. When the 16-bit EXT-BUS (JEXT16 connector) is used, the EXT-BUS interrupt is connected to EXT_INT0.

6.2.7.9.Interrupt Polarity Specification Register 1 (INT_POLARITY1 GCS2:0000-6070H) [Read/Write]

Bit	Signal name	RST	Function
0	PCI1_INTA_INTHIGH *2	0	0: An interrupt request due to the PCI slot 1 INTA- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 1 INTA- pin is High/rising edge.
1	PCI1_INTB_INTHIGH *2	0	0: An interrupt request due to the PCI slot 1 INTB- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 1 INTB- pin is High/rising edge.
2	PCI1_INTC_INTHIGH *2	0	0: An interrupt request due to the PCI slot 1 INTC- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 1 INTC- pin is High/rising edge.
3	PCI1_INTD_INTHIGH *2	0	0: An interrupt request due to the PCI slot 1 INTD- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 1 INTD- pin is High/rising edge.
4	PCI2_INTA_INTHIGH *2	0	0: An interrupt request due to the PCI slot 2 INTA- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 2 INTA- pin is High/rising edge.
5	PCI2_INTB_INTHIGH *2	0	0: An interrupt request due to the PCI slot 2 INTB- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 2 INTB- pin is High/rising edge.
6	PCI2_INTC_INTHIGH *2	0	0: An interrupt request due to the PCI slot 2 INTC- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 2 INTC- pin is High/rising edge.
7	PCI2_INTD_INTHIGH *2	0	0: An interrupt request due to the PCI slot 2 INTD- pin is Low/falling edge. 1: An interrupt request due to the PCI slot 2 INTD- pin is High/rising edge.
8	EXT_INT0_INTHIGH *3	0	0: An interrupt request due to the EXT bus EXT_INT0- pin is Low/falling edge. 1: An interrupt request due to the EXT bus EXT_INT0- pin is High/rising edge.
9	EXT_INT1_INTHIGH	0	0: An interrupt request due to the EXT bus EXT_INT1- pin is Low/falling edge. 1: An interrupt request due to the EXT bus EXT_INT1- pin is High/rising edge.
10	EXT_INT2_INTHIGH	0	0: An interrupt request due to the EXT bus EXT_INT2- pin is Low/falling edge. 1: An interrupt request due to the EXT bus EXT_INT2- pin is High/rising edge.
11	EXT_INT3_INTHIGH	0	0: An interrupt request due to the EXT bus EXT_INT3- pin is Low/falling edge. 1: An interrupt request due to the EXT bus EXT_INT3- pin is High/rising edge.
12	USB0_OC_INTHIGH	0	0: An interrupt request due to a USB0 over current is Low/falling edge. 1: An interrupt request due to a USB0 over current is High/rising edge.
13	USB1_OC_INTHIGH	0	0: An interrupt request due to a USB1 over current is Low/falling edge. 1: An interrupt request due to a USB1 over current is High/rising edge.
14	PCMCIA_OC_INTHIGH	0	0: An interrupt request due to a PCMCIA over current is Low/falling edge. 1: An interrupt request due to a PCMCIA over current is High/rising edge.
15	Unused	x	

<<Cautions>>

1. INTHIGH sets the polarity of an interrupt request from an interrupt resource. When the INTEDGE setting is edge mode, either rising edge or falling edge can be selected according to the INTHIGH setting. When the INTEDGE setting is level mode, either Low active or High active can be selected according to the INTHIGH setting.
   When edge mode is set by INTEDGE, an edge detection circuit may end up detecting an edge according to the polarity setting changes. Therefore, when this register setting is switched, the edge detection circuit should be cleared by the relevant INTCLR after the switch.
2. A PCI bus slot interrupt is normally used with a Low level.
3. When the 16-bit EXT-BUS (JEXT16 connector) is used, the EXT-BUS interrupt is connected to EXT_INT0.

6.2.7.10.GINTO0 Interrupt Enable Register 0 (GINTO0_INTEN0 GCS2:0000-6100H) [Read/Write]

6.2.7.11.GINTO1 Interrupt Enable Register 0 (GINTO1_INTEN0 GCS2:0000-6120H) [Read/Write]

6.2.7.12.GINTO2 Interrupt Enable Register 0 (GINTO2_INTEN0 GCS2:0000-6140H) [Read/Write]

6.2.7.13.GINTO3 Interrupt Enable Register 0 (GINTO3_INTEN0 GCS2:0000-6160H) [Read/Write]

Bit	Signal name	RST	Function
0	GIn_ALL_INTEN*3	0	0: Disable interrupts from all resources. 1: Enable/disable interrupt according to the bit for the individual resource.
1	GIn_UART0_INTEN	0	0: Disable the interrupt from the TL16PRI552 UART0 pin. 1: Enable the interrupt from the TL16PRI552 UART0 pin.
2	GIn_UART1_INTEN	0	0: Disable the interrupt from the TL16PRI552 UART1 pin. 1: Enable the interrupt from the TL16PRI552 UART1 pin.
3	GIn_PRT_INTEN	0	0: Disable the interrupt from the TL16PRI552 PRINTER pin. 1: Enable the interrupt from the TL16PRI552 PRINTER pin.
4	GIn_AUDIO_INTEN	0	0: Disable the interrupt from the AUDIO pin. 1: Enable the interrupt from the AUDIO pin.
5	GIn_P9_LINT_INTEN	0	0: Disable the interrupt due to the PCI9080 LINTo- pin. 1: Enable the interrupt due to the PCI9080 LINTo- pin.
6	GIn_GINTI0_INTEN	0	0: Disable the interrupt due to the GBUS GINTI0- pin. 1: Enable the interrupt due to the GBUS GINTI0- pin.
7	GIn_GINTI1_INTEN	0	0: Disable the interrupt due to the GBUS GINTI1- pin. 1: Enable the interrupt due to the GBUS GINTI1- pin.
8	GIn_ISAINTR_INTEN	0	0: Disable the interrupt due to the ISA bus INTR pin. 1: Enable the interrupt due to the ISA bus INTR pin.
9	GIn_ISANMI_INTEN	0	0: Disable the interrupt due to the ISA bus NMI pin. 1: Enable the interrupt due to the ISA bus NMI pin.
10	GIn_LAN_INTEN	0	0: Disable the interrupt due to the SB82558 INTA- pin. 1: Enable the interrupt due to the SB82558 INTA- pin.
11	GIn_P9_LSERR_INTEN	0	0: Disable the interrupt due to the PCI9080 LSERR- pin. 1: Enable the interrupt due to the PCI9080 LSERR- pin.
12	GIn_PCI_PERR_INTEN	0	0: Disable the interrupt due to the occurrence of a parity error on the PCI bus. 1: Enable the interrupt due to the occurrence of a parity error on the PCI bus.
13	GIn_TOVRDY_INTEN	0	0: Disable the interrupt due to the occurrence of a time-over ready. 1: Enable the interrupt due to the occurrence of a time-over ready.
14	GIn_ABROT_ERR_INTEN	0	0: Disable the interrupt due to the occurrence of an abort termination. 1: Enable the interrupt due to the occurrence of an abort termination.
15	GIn_BACKOFF_ERR_INTEN	0	0: Disable the interrupt due to the occurrence of a back off. 1: Enable the interrupt due to the occurrence of a back off.

<<Cautions>>

1. One of these registers exists for each of GINTO-[3:0]. In the above table, GIn_xxxx_INTEN represents GI0_xxxx_INTEN to GI3_xxxx_INTEN.
2. Since INTEN is unrelated to the edge detection circuit, an interrupt request being maintained by an edge detection circuit will continue to be maintained even if INTEN is set to "0." (See Section 6.2.7.1, "Overview of Interrupt Resources").
3. Use GIn_ALL_INTEN when GINTOn- is connected to a CPU edge-sensitive interrupt on the CPU board. When the relevant interrupt handling routine ends, set GIn_ALL_INTEN to "0" and then return it to "1." If this is done, when interrupts arrive from multiple interrupt resources, an edge is generated on the interrupt line to the CPU so that the next interrupt will be generated after control leaves an interrupt handling routine. (See Section 6.2.7.1, "Overview of Interrupt Resources").

6.2.7.14.GINTO0 Interrupt Enable Register 1 (GINTO0_INTEN1 GCS2:0000-6110H) [Read/Write]

6.2.7.15.GINTO1 Interrupt Enable Register 1 (GINTO1_INTEN1 GCS2:0000-6130H) [Read/Write]

6.2.7.16.GINTO2 Interrupt Enable Register 1 (GINTO2_INTEN1 GCS2:0000-6150H) [Read/Write]

6.2.7.17.GINTO3 Interrupt Enable Register 1 (GINTO3_INTEN1 GCS2:0000-6170H) [Read/Write]

Bit	Signal name	RST	Function
0	GIn_PCI1_INTA_INTEN	0	0: Disable the interrupt due to the PCI slot 1 INTA- pin. 1: Enable the interrupt due to the PCI slot 1 INTA- pin.
1	GIn_PCI1_INTB_INTEN	0	0: Disable the interrupt due to the PCI slot 1 INTB- pin. 1: Enable the interrupt due to the PCI slot 1 INTB- pin.
2	GIn_PCI1_INTC_INTEN	0	0: Disable the interrupt due to the PCI slot 1 INTC- pin. 1: Enable the interrupt due to the PCI slot 1 INTC- pin.
3	GIn_PCI1_INTD_INTEN	0	0: Disable the interrupt due to the PCI slot 1 INTD- pin. 1: Enable the interrupt due to the PCI slot 1 INTD- pin.
4	GIn_PCI2_INTA_INTEN	0	0: Disable the interrupt due to the PCI slot 2 INTA- pin. 1: Enable the interrupt due to the PCI slot 2 INTA- pin.
5	GIn_PCI2_INTB_INTEN	0	0: Disable the interrupt due to the PCI slot 2 INTB- pin. 1: Enable the interrupt due to the PCI slot 2 INTB- pin.
6	GIn_PCI2_INTC_INTEN	0	0: Disable the interrupt due to the PCI slot 2 INTC- pin. 1: Enable the interrupt due to the PCI slot 2 INTC- pin.
7	GIn_PCI2_INTD_INTEN	0	0: Disable the interrupt due to the PCI slot 2 INTD- pin. 1: Enable the interrupt due to the PCI slot 2 INTD- pin.
8	GIn_EXT_INT0_INTEN*3	0	0: Disable the interrupt due to the EXT bus EXT_INT0- pin. 1: Enable the interrupt due to the EXT bus EXT_INT0- pin.
9	GIn_EXT_INT1_INTEN	0	0: Disable the interrupt due to the EXT bus EXT_INT1- pin. 1: Enable the interrupt due to the EXT bus EXT_INT1- pin.
10	GIn_EXT_INT2_INTEN	0	0: Disable the interrupt due to the EXT bus EXT_INT2- pin. 1: Enable the interrupt due to the EXT bus EXT_INT2- pin.
11	GIn_EXT_INT3_INTEN	0	0: Disable the interrupt due to the EXT bus EXT_INT3- pin. 1: Enable the interrupt due to the EXT bus EXT_INT3- pin.
12	GIn_USB0_OC_INTEN	0	0: Disable the interrupt due to a USB0 over current. 1: Enable the interrupt due to a USB0 over current.
13	GIn_USB1_OC_INTEN	0	0: Disable the interrupt due to a USB1 over current. 1: Enable the interrupt due to a USB1 over current.
14	GIn_PCMCIA_OC_INTEN	0	0: Disable the interrupt due to a PCMCIA over current. 1: Enable the interrupt due to a PCMCIA over current.
15	Unused	x	

<<Cautions>>

1. One of these registers exists for each of GINTO-[3:0]. In the above table, GIn_xxxx_INTEN represents GI0_xxxx_INTEN to GI3_xxxx_INTEN.
2. Since INTEN is unrelated to the edge detection circuit, an interrupt request being maintained by an edge detection circuit will continue to be maintained even if INTEN is set to "0." (See Section 6.2.7.1, "Overview of Interrupt Resources").
3. When the 16-bit EXT-BUS (JEXT16 connector) is used, the EXT-BUS interrupt is connected to EXT_INT0.

6.2.8. EXT-BUS Control Registers

The local bus registers include registers for controlling the EXT-BUS. Those registers are described below.

6.2.8.1.Bank Window

The EXT-BUS spaces include individual 16M-byte memory and I/O spaces for the 32-bit EXT-BUS (JEXT32 connector) and a 1M-byte space for the 16-bit EXT-BUS (JEXT16 connector). The bank method is used as the address expansion method to support cases in which space for the EXT-BUS cannot be allocated on the CPU board. This bank method allows the window size to be programmable (variable) to match the CPU board.
The addresses that are output to the EXT-BUS are determined by the following equation.

[EXT-BUS address bit23-16] = [GBUS address bit23-16]&~[address mask]
[bank address]&[address mask]

For bits corresponding to address mask bits that are "1," the value set as the bank address is output to the high-order 8 bits of the EXT-BUS address, and for bits corresponding to address mask bits that are "0," the GBUS address is output. Therefore, data having consecutive "1" high-order bits and consecutive "0" low-order bits, such as F0H, is set as the address mask. The bank window size is determined by the number of low-order bits that are "0." When FFH, which has no "0" bits, is set as the address mask, the bank window size will be 64K bytes. When C0H is set, the bank window size will be 4M bytes.
Also, since two address masks can be set, with one for the memory space and one for the I/O space, separate bank window sizes can be set for memory and I/O. However, the 16-bit EXT-BUS (JEXT16) only uses memory.
The bank address can be set separately when the CPU is accessed, when DMA channel 2 is accessed, and when DMA channel 3 is accessed. In addition, the memory and I/O spaces are set separately for each of these cases. However, since the GBUS GDMAAK2- signal and GBUS GDMAAK3- signal must be active to enable bank addresses for DMA accesses, DMA bank addresses cannot be used with a CPU board for which the DMAAK signal is not available.

6.2.8.2.EXT-BUS Memory Space Address Mask Register (EXTBUS_MEM_AMASK GCS2:0000-7000H)

[Read/Write]

Bit	Signal name	RST	Function
0	EXT_MEM_AMASK16	0	EXT_MEM_AMASK[23:16] specifies the address mask of the EXT-BUS memory space.
1	EXT_MEM_AMASK17	0	
2	EXT_MEM_AMASK18	0	
3	EXT_MEM_AMASK19	0	
4	EXT_MEM_AMASK20	0	
5	EXT_MEM_AMASK21	0	
6	EXT_MEM_AMASK22	0	
7	EXT_MEM_AMASK23	0	

<<Cautions>>

1. EXT_MEM_AMASK[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS memory space, that is, for an access when the GBUS GCS3- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, the setting of this register is always used for an access to the EXT-BUS, and the EXT_IO_AMASK[23:16] register is not used.

6.2.8.3.EXT-BUS I/O Space Address Mask Register (EXTBUS_IO_AMASK GCS2:0000-7010H)

[Read/Write]

Bit	Signal name	RST	Function
0	EXT_IO_AMASK16	0	EXT_IO_AMASK[23:16] specifies the address mask of the EXT-BUS I/O space.
1	EXT_IO_AMASK17	0	
2	EXT_IO_AMASK18	0	
3	EXT_IO_AMASK19	0	
4	EXT_IO_AMASK20	0	
5	EXT_IO_AMASK21	0	
6	EXT_IO_AMASK22	0	
7	EXT_IO_AMASK23	0	

<<Cautions>>

1. EXT_IO_AMASK[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS I/O space, that is, for an access when the GBUS GCS4- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, the setting of this register is not used for an access to the EXT-BUS, and the EXT_MEM_AMASK[23:16] register is used.

6.2.8.4.EXT-BUS CPU Memory Space Bank Address Register (EXTBUS_CORE_MEM_BANK_ADDR GCS2:0000-7020H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_CORE_MEM_BANKA16	0	EXT_CORE_MEM_BANKA[23:16] specifies the bank address of a non-DMA access to the EXT-BUS memory space.
1	EXT_CORE_MEM_BANKA17	0	
2	EXT_CORE_MEM_BANKA18	0	
3	EXT_CORE_MEM_BANKA19	0	
4	EXT_CORE_MEM_BANKA20	0	
5	EXT_CORE_MEM_BANKA21	0	
6	EXT_CORE_MEM_BANKA22	0	
7	EXT_CORE_MEM_BANKA23	0	

<<Cautions>>

1. EXT_CORE_MEM_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS memory space that is not a DMA access, that is, for an access when both the GBUS GDMAAK2- and GDMAAK3- signals are not active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, although no DMA cycle is defined for the 16-bit EXT-BUS, the setting of this register is not used and the EXT_DMA0_MEM_BANKA[23:16] register or EXT_DMA1_MEM_BANKA[23:16] register is used by cycles in which the GBUS GDMAAK2- and GDMAAK3- signals are active.

6.2.8.5.EXT-BUS CPU I/O Space Bank Address Register (EXTBUS_CORE_IO_BANK_ADDR GCS2:0000-7030H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_CORE_IO_BANKA16	0	EXT_CORE_IO_BANKA[23:16] specifies the bank address of a non-DMA access to the EXT-BUS I/O space.
1	EXT_CORE_IO_BANKA17	0	
2	EXT_CORE_IO_BANKA18	0	
3	EXT_CORE_IO_BANKA19	0	
4	EXT_CORE_IO_BANKA20	0	
5	EXT_CORE_IO_BANKA21	0	
6	EXT_CORE_IO_BANKA22	0	
7	EXT_CORE_IO_BANKA23	0	

<<Cautions>>

1. EXT_CORE_IO_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS I/O space that is not a DMA access, that is, for an access when both the GBUS GDMAAK2- and GDMAAK3- signals are not active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, the setting of this register is not used and the setting of EXT_CORE_MEM_BANKA[23:16] is used for an access to the EXT-BUS.

6.2.8.6.EXT-BUS DMA0 Memory Space Bank Address Register (EXTBUS_DMA0_MEM_BANK_ADDR GCS2:0000-7040H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_DMA0_MEM_BANKA16	0	EXT_DMA0_MEM_BANKA[23:16] specifies the bank address of an access to the EXT-BUS memory space according to DMA channel 2.
1	EXT_DMA0_MEM_BANKA17	0	
2	EXT_DMA0_MEM_BANKA18	0	
3	EXT_DMA0_MEM_BANKA19	0	
4	EXT_DMA0_MEM_BANKA20	0	
5	EXT_DMA0_MEM_BANKA21	0	
6	EXT_DMA0_MEM_BANKA22	0	
7	EXT_DMA0_MEM_BANKA23	0	

<<Cautions>>

1. EXT_DMA0_MEM_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS memory space when the EXT-BUS DMAAK0- signal is active, that is, when the GBUS GDMAAK2- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."

6.2.8.7.EXT-BUS DMA0 I/O Space Bank Address Register (EXTBUS_DMA0_IO_BANK_ADDR GCS2:0000-7050H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_DMA0_IO_BANKA16	0	EXT_DMA0_IO_BANKA[23:16] specifies the bank address of an access to the EXT-BUS I/O space according to DMA channel 2.
1	EXT_DMA0_IO_BANKA17	0	
2	EXT_DMA0_IO_BANKA18	0	
3	EXT_DMA0_IO_BANKA19	0	
4	EXT_DMA0_IO_BANKA20	0	
5	EXT_DMA0_IO_BANKA21	0	
6	EXT_DMA0_IO_BANKA22	0	
7	EXT_DMA0_IO_BANKA23	0	

<<Cautions>>

1. EXT_DMA0_IO_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS I/O space when the EXT-BUS DMAAK0- signal is active, that is, when the GBUS GDMAAK2- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, the setting of this register is not used and the setting of EXT_DMA0_MEM_BANKA[23:16] is used for an access to the EXT-BUS. However, since no DMA cycle is defined for the 16-bit EXT-BUS, there is no method for distinguishing whether or not a cycle is a DMA cycle on the EXT-BUS.

6.2.8.8.EXT-BUS DMA1 Memory Space Bank Address Register (EXTBUS_DMA1_MEM_BANK_ADDR GCS2:0000-7060H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_DMA1_MEM_BANKA16	0	EXT_DMA1_MEM_BANKA[23:16] specifies the bank address of an access to the EXT-BUS memory space according to DMA channel 3.
1	EXT_DMA1_MEM_BANKA17	0	
2	EXT_DMA1_MEM_BANKA18	0	
3	EXT_DMA1_MEM_BANKA19	0	
4	EXT_DMA1_MEM_BANKA20	0	
5	EXT_DMA1_MEM_BANKA21	0	
6	EXT_DMA1_MEM_BANKA22	0	
7	EXT_DMA1_MEM_BANKA23	0	

<<Cautions>>

1. EXT_DMA1_MEM_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS memory space when the EXT-BUS DMAAK1- signal is active, that is, when the GBUS GDMAAK3- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."

6.2.8.9.EXT-BUS DMA1 I/O Space Bank Address Register (EXTBUS_DMA1_IO_BANK_ADDR GCS2:0000-7070H) [Read/Write]

Bit	Signal name	RST	Function
0	EXT_DMA1_IO_BANKA16	0	EXT_DMA1_IO_BANKA[23:16] specifies the bank address of an access to the EXT-BUS I/O space according to DMA channel 3.
1	EXT_DMA1_IO_BANKA17	0	
2	EXT_DMA1_IO_BANKA18	0	
3	EXT_DMA1_IO_BANKA19	0	
4	EXT_DMA1_IO_BANKA20	0	
5	EXT_DMA1_IO_BANKA21	0	
6	EXT_DMA1_IO_BANKA22	0	
7	EXT_DMA1_IO_BANKA23	0	

<<Cautions>>

1. EXT_DMA1_IO_BANKA[23:16] determines the address to be output to the EXT-BUS for an access to the EXT-BUS I/O space when the EXT-BUS DMAAK1- signal is active, that is, when the GBUS GDMAAK3- signal is active. For the correspondence of the GBUS address and EXT-BUS address, see Section 6.2.8.1, "Bank Window."
2. When a 16-bit EXT-BUS (JEXT16 connector) is used, the setting of this register is not used and the setting of EXT_DMA1_MEM_BANKA[23:16] is used for an access to the EXT-BUS. However, since no DMA cycle is defined for the 16-bit EXT-BUS, there is no method for distinguishing whether or not a cycle is a DMA cycle on the EXT-BUS.

6.2.8.10.EXT-BUS Status Register (EXTBUS_STATUS GCS2:0000-7080H) [Read Only]

Bit	Signal name	RST	Function
0	JEXT32_16BIT*1	x	0: 32-bit EXT-BUS size is 32 bits. 1: 32-bit EXT-BUS size is 16 bits.
1	JEXT16_EN*1*2	x	0: No board is connected to the JEXT16 connector. 1: A board is connected to the JEXT16 connector.
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. The EXT-BUS has no dynamic bus sizing function. Therefore, when a board having a bus size of 16 bits is connected to the EXT-BUS, the EXT-BUS cannot be accessed using a data size exceeding 16 bits. When either JEXT32_16BIT or JEXT16_EN is "1," an access to the EXT-BUS is performed using a data size not exceeding 16 bits.
2. JEXT16_EN essentially is a signal that is generated according to the level of the pin that is connected to GND. If it cannot be detected normally by this method, JEXT16_EN can be set to "1" by short-circuiting JP5. (See Section 5.6, "EXT-BUS Forced 16-Bit Jumper (JP5)" and Section 10.2, "Signals.")

6.2.8.11.EXT-BUS Control Register (EXTBUS_CONTROL GCS2:0000-7090H) [Read/Write]

Bit	Signal name	RST	Function
0	DMAAK_THROUGH*1	1	0: The EXT-BUS DMAAK- signal becomes active only during a DMA cycle. 1: The EXT-BUS DMAAK- signal obeys the active timing of the DMAAK- signal of the CPU board.
1	Unused	x	
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. The DMAAK_THROUGH signal controls the active timing of the EXT-BUS DMAAK- signal. When DMAAK_THROUGH is set to "1," the EXT-BUS DMAAK- signal changes with almost the identical timing as the DMAAK- signal from the CPU board. (Actually, to guarantee the hold time for the RD-/WR- signal, the logical OR operation is performed with a signal that is latched within the motherboard.) Therefore, for a DMA transfer from the CPU board to the EXT-BUS, the EXT-BUS DMAAK- signal becomes active somewhat before the EXT-BUS DMA cycle begins (because the DMAAK- signal becomes active after the transfer cycle began on the CPU board). When DMAAK_THROUGH is set to "0," the EXT-BUS DMAAK- signal becomes active only during the DMA cycle.

Unless there is a special reason not to, the DMAAK_THROUGH signal should be used when set to "1." Since the timing according to which the EXT-BUS DMAAK- signal becomes active will be slower when DMAAK_THROUGH is set to "0," superfluous DMA cycles may be generated for some CPU boards when executing a DMA transfer from the CPU board to the EXT-BUS.

6.2.9. Other Control Registers

This section explains other registers on the local bus.

6.2.9.1.SW1 Read Out Register (SW1_RDOUT GCS2:0000-8000H) [Read Only]

This is a register for reading the SW1 status. The following table shows the data format.

Address	Data bus								Contents
	D7	D6	D5	D4	D3	D2	D1	D0	
GCS2:0000-8000H	SW1-8	SW1-7	SW1-6	SW1-5	SW1-4	SW1-3	SW1-2	SW1-1	0=ON , 1=OFF

SW1-1 corresponds to the SW1 "1" switch, and SW1-8 corresponds to the SW1 "8" switch. Also, 0 is read when the relevant bit's switch is ON, and 1 is read when it is OFF. SW1 is only used for software. No hardware control is switched according to the SW1 setting. No specific use has been fixed for SW1.

6.2.9.2.POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write]

Bit	Signal name	RST	Function
0	USB0_PWEN*1	0	0: Do not supply +5 V to USB channel 0. 1: Supply +5 V to USB channel 0.
1	USB1_PWEN*1	0	0: Do not supply +5 V to USB channel 1. 1: Supply +5 V to USB channel 1.
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. USB channel 0 is the lower JUSB connector. Channel 1 is the upper JUSB connector.

6.2.9.3.POWER Status Register (POWER_STATUS GCS2:0000-8030H) [Read Only]

Bit	Signal name	RST	Function
0	USB0_OVCURRENT*1*3	0	0: The +5 V power status of USB channel 0 is not over current. 1: The +5 V power status of USB channel 0 is over current.
1	USB1_OVCURRENT*1*3	0	0: The +5 V power status of USB channel 1 is not over current. 1: The +5 V power status of USB channel 1 is over current.
2	PCMCIA_OVCURRENT*2*3	0	0: The PCMCIA status is not over current. 1: The PCMCIA status is over current.
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. USB channel 0 is the lower JUSB connector. Channel 1 is the upper JUSB connector. For information about when the USB power is ON/OFF, see Section 6.2.9.2, 'POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write]."
2. PCMCIA_OVCURRENT becomes "1" when the +3.3 V, +5 V, or +12 V power state of either of the two PCMCIA slots is over current. The PCMCIA power is controlled according to a register within the RF5C396 (PCMCIA controller). (See Section 6.5, "PCMCIA Bus.")
3. Each over current state can cause an interrupt to be generated. (See Section 6.2.7.1, "Overview of Interrupt Resources").

6.2.9.4.ISA Interrupt Vector Register (ISA_INT_VECTOR GCS2:0000-8040H) [Read Only]

Bit	Signal name	RST	Function
0	ISA_INT_VECT0	0	ISA_INT_VECT[7:0] indicates the interrupt vector due to the ISA bus interrupt.
1	ISA_INT_VECT1	0	
2	ISA_INT_VECT2	0	
3	ISA_INT_VECT3	0	
4	ISA_INT_VECT4	0	
5	ISA_INT_VECT5	0	
6	ISA_INT_VECT6	0	
7	ISA_INT_VECT7	0	

<<Cautions>>

1. ISA_INT_VECT[7:0] indicates the interrupt vector that is obtained by generating an interrupt acknowledge cycle on the PCI bus when an interrupt is generated due to an ISA_INTR signal (M1523B INTR pin) request. The software can determine which interrupt was requested by looking at this vector.
2. The contents of this register are maintained until the interrupt is cleared by ISAINTR_CLR. If an interrupt is being held when the interrupt is cleared by ISAINTR_CLR, the interrupt acknowledge cycle is started again, and a new interrupt vector is set in this register. (See Section 6.2.7.3, "Interrupt Clear Register 0 (INT_CLEAR0 GCS2:0000-6010H) [Write Only].")

6.2.9.5.ISA Interrupt Status Register (ISA_INT_STATUS GCS2:0000-8050H) [Read Only]

Bit	Signal name	RST	Function
0	ISA_INTR*1	0	0: The M1523B INTR pin is inactive. 1: The M1523B INTR pin is active.
1	Unused	x	
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. ISA_INTR indicates whether the INTR pin, which is the ISA interrupt request pin of the M1523B (SouthBridge) chip, is active or inactive. This bit has no specific use, but is intended for hardware debugging.

6.2.9.6.BREQ Control Register (BREQ_CONTROL GCS2:0000-8060H) [Read/Write]

Bit	Signal name	RST	Function
0	BREQ_NUM0*1*2	0	BREQ_NUM[1:0] sets the timing for issuing a bus mastership request to the PCI9080.
1	BREQ_NUM1*1*2	0	
2	BREQ_EN*1*2	0	0: The bus mastership is not requested during the PCI9080 burst cycle. 1: The bus mastership is requested during the PCI9080 burst cycle.
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. The cycle that the PCI9080 generates for the GBUS is not limited to the burst count. Therefore, the CPU may be unable to obtain the GBUS bus mastership for a long period of time.
   BREQ_EN and BREQ_NUM[1:0] are provided to enable the CPU to always obtain the bus mastership within a certain desired interval. By setting BREQ_EN and BREQ_NUM[1:0], the PCI9080 BREQ pin can be set to active state according to the timing shown in the following table to direct the PCI9080 to relinquish the bus mastership. When the BREQ pin becomes active, the PCI9080 terminates the bus cycle within two micro cycles and relinquishes the bus mastership.

GBUS signal GUSE_DIRECT_ACC-	BREQ_EN	BREQ_NUM[1:0]	BREQ active timing
High	0	[x, x]	BREQ does not become active.
	1	[0, 0]	When the CPU begins the cycle and GADS- is set to active.
		[0, 1]	Setting prohibited.
		[1, 0]	The slowest timing among the following two. • The PCI9080 continued the cycle for at least six micro cycles. • The CPU begins the cycle and GADS- is set to active.
		[1, 1]	The slowest timing among the following two. • The PCI9080 continued the cycle for at least 14 micro cycles. • The CPU begins the cycle and GADS- is set to active.
Low	0	[x, x]	BREQ does not become active.
	1	[0, 0]	Immediately after the PCI9080 cycle begins.
		[0, 1]	Setting prohibited.
		[1, 0]	When the PCI9080 continued the cycle for at least six micro cycles.
		[1, 1]	When the PCI9080 continued the cycle for at least 14 micro cycles.

2. Another method for setting the PCI9080 BREQ pin to active besides using BREQ_EN and BREQ_NUM[1:0] is to set the GBUS GBREQ- signal to active. A logical OR operation is performed for the GBUS GBREQ- signal and the BREQ signal generated according to BREQ_EN and BREQ_NUM[1:0].

6.2.9.7.Flash ROM Control Register (FROM_CONTROL GCS2:0000-8070H) [Read/Write]

Bit	Signal name	RST	Function
0	FROM_RESET*1	0	0: The flash ROM RESET pin is set to inactive. 1: The flash ROM RESET pin is set to active.
1	FROM_BUSY*2	x	0: The flash ROM status is not BUSY. 1: The flash ROM status is BUSY.
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. FROM_RESET manipulates the flash ROM reset pin.
2. FROM_BUSY is a read-only bit. It becomes "1" when the status of any one of the four flash ROM is BUSY.

6.2.9.8.Bus Lock Control Register (BLOCK_CONTROL GCS2:0000-8080H) [Read/Write]

Bit	Signal name	RST	Function
0	SRAM_BLOCK*1	0	0: The bus lock for accessing SRAM is disabled. 1: The bus lock for accessing SRAM is enabled.
1	PCI_BLOCK*2	0	0: The bus lock for accessing the PCI bus is disabled. 1: The bus lock for accessing the PCI bus is enabled.
2	Unused	x	
3	Unused	x	
4	Unused	x	
5	Unused	x	
6	Unused	x	
7	Unused	x	

<<Cautions>>

1. The GBUS GBLOCK0- signal is the bus lock signal for SRAM. However, if no bus lock signal is output from the CPU, SRAM_BLOCK should be used. A logical OR operation is performed on the GBUS GBLOCK0- and SRAM_BLOCK signals.
   After "1" is set for SRAM_BLOCK, if the CPU attempts to access SRAM, the bus mastership is no longer passed to the PCI9080, and a bus lock state occurs for this and subsequent accesses. To escape from the bus lock state, set "0" for SRAM_BLOCK.
2. The GBUS GBLOCK1- signal is the bus lock signal for the PCI bus. However, if no bus lock signal is output from the CPU, PCI_BLOCK should be used. A logical OR operation is performed on the GBUS GBLOCK1- and PCI_BLOCK signals.
   When "1" is set for PCI_BLOCK, the PCI9080 LLOCK# pin becomes active. This causes a bus lock state to occur for subsequent accesses to the PCI bus. To escape from the bus lock state, set "0" for PCI_BLOCK.
   Since a bus lock cycle on the PCI bus is set as a read/modify/write cycle, the following sequence should be performed: Set "1" for PCI_BLOCK → Read cycle for the PCI bus → Write cycle for the PCI bus → Set "0" for PCI_BLOCK.

6.2.9.9.TOVRDY LED Clear Register (TOVRDY_LED_CLR GCS2:0000-8090H) [Write Only]

When data is written to this register, the TOVRDY-LED goes off. (See Section 6.1.4, "Time-Over Ready.")

6.2.9.10.ABORT LED Clear Register (ABORT_LED_CLR GCS2:0000-80A0H) [Write Only]

When data is written to this register, the ABORT-LED goes off. (See Section 6.3.4.6, "Abort Error.")

6.2.9.11.BACKOFF LED Clear Register (BAFCKOFF_LED_CLR GCS2:0000-80B0H) [Write Only]

When data is written to this register, the BACKOFF-LED goes off. (See Section 6.3.4.7, "Back Off Error.")

6.3. PCI Bus

The PCI bus of the RTE-MOTHER-A motherboard conforms to PCI standards 2.1. An access to the PCI bus from the CPU board is performed via the PCI9080.

6.3.1. Bus Arbitration

The round-robin method is used for bus arbitration of the PCI bus, and the priority rotates according to the following sequence.

1) PCI9080
2) Interrupt acknowledge cycle generation circuit
3) M1523B (SouthBridge)
4) SB82558 (LAN controller)
5) PCI slot 1
6) PCI slot 2

A bus mastership request from the M1523B (SouthBridge) chip is a request from the on-chip USB controller, IDE controller, or DMA controller installed on the M1523B.

6.3.2. Interrupt Acknowledge Cycle

When the M1523B (SouthBridge) requests an interrupt according to the INTR pin, a circuit on the motherboard generates an interrupt acknowledge cycle on the PCI bus and reads the interrupt vector from the M1523B. The interrupt request is issued to the CPU board when the interrupt vector has been read. (See Section 6.2.7.3, "Interrupt Clear Register 0 (INT_CLEAR0 GCS2:0000-6010H) [Write Only]" and Section 6.2.9.4, "ISA Interrupt Vector Register (ISA_INT_VECTOR GCS2:0000-8040H) [Read Only].")

6.3.3. Recommended Map

The following table shows the recommended memory map on the PCI bus.

Address range	Resources
0000-0000H to 0007-FFFFH	SRAM on the local bus. Same as the area that can be accessed from the range 0080-0000H to 0087-FFFFH. SRAM is also allocated to this range so that SRAM can be accessed from a resource on the ISA bus. The PCI9080 expanded ROM space is used.
007F-E000H to 007F-EFFFH	USB controller
007F-FF00H to 007F-FFFFH	PCI9080 registers
0080-0000H to 009F-FFFFH	SRAM on the local bus. The PCI9080 SP0 space is used.
00A0-0000H to 00FF-FFFFH	Reserved area
0200-0000H to 03FF-FFFFH	Resource on the CPU board (when the GUSE_DIRECT_ACC- signal is LOW). The PCI9080 SP1 space is used.

6.3.4. PCI9080

The PCI9080 is a PLX Technology PCI bus bridge chip. The PCI9080 data sheet is available from the PLX Technology home page (http://www.plxtech.com/).

6.3.4.1.PCI9080 Control Register

The PCI9080 control register can be accessed during cycles when the GBUS GCS6- pin is set to active. The control register on the PCI9080 for determining the action of the PCI9080, which is called the CFG register, can be accessed by using this space.
The CFG register can be used to make the following settings.

- Correspondence between a GBUS address and a memory space address on the PCI bus.
- Correspondence between a GBUS address and an I/O space address on the PCI bus.
- Size of the PCI bus space that can be accessed from the GBUS.
- Setting for generating a configuration cycle on the PCI bus.
- Detailed setting related to the PCI bus or GBUS cycle.
- DMA controller setting that can be transferred to the on-chip PCI bus/GBUS installed in the PCI9080

6.3.4.2.PCI Bus Memory Area

An access when the GBUS GCS5- signal is set to active will be a PCI bus-directed access to the PCI9080. (Since the settings will differ when the CPU board does not support the GBUS GCS7- signal, see Section 6.3.4.4, "When the GCS7- Signal Is Not Supported.")
A cycle at this time is controlled by a motherboard circuit so that bit 31 (LA31) of the PCI9080 local-side address is always Low. As a result, the PCI9080 setting for the GBUS-side (local bus-side for the PCI9080) address of an access to the memory space of the PCI bus will start from address 0000-0000H (PCI9080 control register A0H address: Local Bus Base Address Register for Direct Master to PCI Memory). Also, the GBUS-side address range will be the address range that the CPU board allocated to the GBUS GCS5- signal (PCI9080 control register 9CH address: Local Range Register for Direct Master to PCI). Although this address range setting is also valid for an access to the I/O space of the PCI bus, a memory range normally is set because the memory range is larger.
The high-order PCI addresses from this address range are set for the PCI control register A8H address: PCI Base Address (Remap) Register for Direct Master to PCI Memory.
When a multitasking application performs a PCI access for which the PCI Base Address (Remap) Register for Direct Master to PCI Memory must be overwritten, processing for saving this control register is required for each task.

6.3.4.3.PCI Bus I/O Area

An access when the GBUS GCS7- signal is set to active will be a PCI bus-directed access to the PCI9080. (Since the settings will differ when the CPU board does not support the GBUS GCS7- signal, see Section 6.3.4.4, "When the GCS7- Signal Is Not Supported.")
A cycle at this time is controlled by a motherboard circuit so that bit 31 (LA31) of the PCI9080 local-side address is always High. As a result, the PCI9080 setting for the GBUS-side (local bus-side for the PCI9080) address of an access to the I/O space of the PCI bus will start from address 8000-0000H (PCI9080 control register A4H address: Local Base Address Register for Direct Master to PCI IO/CFG). Also, the GBUS-side address range will be the address range allocated to the PCI bus memory area (PCI9080 control register 9CH address: Local Range Register for Direct Master to PCI). Since this address range setting is also valid for an access to the memory space of the PCI bus, a memory range is set because the memory range is larger.
By setting "1" for bit 13 (I/O Remap Select) of the PCI control register A8H address: PCI Base Address (Remap) Register for Direct Master to PCI Memory, the high-order 16 bits (bit 16 to bit 31) of the PCI address during an I/O cycle will be all "0" (as a result, the I/O space on the PCI bus will be 64K bytes in size).

6.3.4.4.When the GCS7- Signal Is Not Supported

If the CPU board does not support a space according to the GBUS GCS7- signal, an access to the PCI I/O space will also be performed by using the GCS5- space. In this case, the first half of the GCS5- space will be allocated for PCI memory space use and the second half will be allocated for PCI I/O space use.
The first half of the area that the CPU board allocates to the GCS5- space is set for the address range (PCI9080 control register 9CH address: Local Range Register for Direct Master to PCI).
The GBUS-side (local bus-side for the PCI9080) address for an access to the memory space of the PCI bus will start from address 0000-0000H (PCI9080 control register A0H address: Local Bus Base Address Register for Direct Master to PCI Memory).
The GBUS-side address for an access to the I/O space of the PCI bus will be the exact middle address of the area for the GCS5- signal (PCI9080 control register A4H address: Local Base Address Register for Direct Master to PCI IO/CFG).
By setting "1" for bit 13 (I/O Remap Select) of the PCI control register A8H address: PCI Base Address (Remap) Register for Direct Master to PCI Memory, the high-order 16 bits (bit 16 to bit 31) of the PCI address during an I/O cycle will be all "0" (as a result, the I/O space on the PCI bus will be 64K bytes in size).

6.3.4.5.PCI Bus Configuration Area

The area for I/O access is used to access the PCI bus configuration area. Configuration cycle address information is set in the PCI9080 control register ACH address: PCI Configuration Address Register for Direct Master to PCI IO/CFG. Also, bit 31 (Configuration Enable) of this control register must be set to "1."
After this information and bit are set, an access to the area for PCI bus I/O access can access the PCI bus configuration space.
Since the PCI9080 control register must be overwritten for a configuration space access, interrupts must be prohibited during a configuration space access when a multitasking application accesses the PCI bus.

6.3.4.6.Abort Error

When a master and target abort or a retry timeout is generated by a bus cycle generated on the PCI bus by the PCI9080, the PCI9080 asserts the BTERMo# pin (GBUS GBTERM- signal) to abort the bus cycle.
When this abort error status occurs, the ABORT-LED on the motherboard lights up. This LED stays lit until it is cleared under software control. (See Section 5.12, "LED" and Section 6.2.9.10, "ABORT LED Clear Register (ABORT_LED_CLR GCS2:0000-80A0H) [Write Only].") Also, an interrupt can be generated due to the occurrence of this abort error status. (See Section 6.2.7.1, "Overview of Interrupt Resources").
After the scanning of resources connected to the PCI bus (PCI bus configuration) is completed, a master and target abort essentially will not occur. Similarly, when the PCI bus is operating normally, a retry timeout will not occur. Therefore, a serious fault is possible when an abort error status occurs.

6.3.4.7.Back Off Error

If the PCI9080 requests a back off when the GBUS GUSE_DIRECT_ACC- signal is Low, a back off error status occurs.
When this back off error status occurs, the BRKOFF-LED on the motherboard lights up. This LED stays lit until it is cleared under software control. (See Section 5.12, "LED" and Section 6.2.9.11, "BACKOFF LED Clear Register (BAFCKOFF_LED_CLR GCS2:0000-80B0H) [Write Only].") Also, an interrupt can be generated due to the occurrence of this back off error status. (See Section 6.2.7.1, "Overview of Interrupt Resources").
A case in which back off occurs is described below.
Assume, for example, that a board referred to as A is inserted in the PCI bus and the internal configuration of this board consists of PCI9080 + SRAM + CPU just like the motherboard.

In this case, the PCI bus can become deadlocked according to the following scenario.

1) The CPU board begins an access to board A's SRAM.
2) At the same time, the board A's CPU board begins an access to the motherboard's SRAM.
3) The motherboard's PCI9080 obtains the PCI bus mastership and begins an access to board A's SRAM on the PCI bus.
4) Since board A's local bus is being used by the CPU, board A's PCI9080 requests a retry termination for a cycle from the PCI bus.
5) The motherboard's PCI9080 replies to the retry termination and relinquishes the bus mastership.
6) Board A obtains the bus mastership and begins an access to the motherboard's SRAM on the PCI bus.
7) The motherboard's PCI9080 requests a retry termination becomes its own local bus is being used by the CPU board.
8) Board A's PCI9080 replies to the retry termination and relinquishes the bus mastership.
9) Return to 3).

To avoid this status in the PCI9080, a back off request is issued to the CPU when the kind of situation described above is detected. If the CPU replies to this back off and relinquishes the bus mastership, the deadlock described above can be avoided.
If the GBUS GUSE_DIRECT_ACC- signal is High on the motherboard, a motherboard circuit implements support for this back off. Therefore, the kind of deadlock described above will not occur.
However, if the GUSE_DIRECT_ACC- signal is Low, back off cannot be supported and an error will occur.

6.3.4.8.Access to the GBUS from the PCI Bus

When the bus master on the PCI bus accesses a resource on the GBUS (SRAM on the local bus or resource on the CPU board), the PCI9080 must be configured so that an address on the local bus becomes a specific value. Also, when the bus master on the PCI bus accesses SRAM on the motherboard, the correspondence between an address on the PCI bus and a CPU address on the CPU board depends on a PCI9080 setting. (See Section 6.1.6, "GBUS Memory Map (Access from the PCI Bus).")

6.3.4.9.Initial Values

The PCI9080 control register is initialized as follows by a serial EEPROM that is installed when the motherboard is shipped from the factory. Since these initial values are set to suit a broad spectrum of typical devices, these values must be changed according to the type of connected CPU board or according to the contents of the applications to be run.

Register name	Address	Setting value
PCI Configuration ID Register	GCS6:000H	0030-1410H
PCI Revision ID Register	GCS6:008H	00H
PCI Class Code Register	GCS6:009H	FF-FFFFH
PCI Subsystem Vendor ID Register	GCS6:02CH	10B5H
PCI Subsytem ID Register	GCS6:02EH	9080H
PCI Expansion ROM Base Register	GCS6:030H	0000-0001H
PCI Interrupt Line Register	GCS6:03CH	00H
PCI Interrupt Pin Register	GCS6:03DH	00H
PCI Min_Gnt Register	GCS6:03EH	04H
PCI Max_Lat Register	GCS6:03FH	80H
Local Address Space 0 Range Register for PCI-to-Local Bus	GCS6:080H	FFE0-0000H
Local Address Space 0 Local Base Address (Remap) Register	GCS6:084H	0000-0001H
Mode/Arbitration Register	GCS6:088H	13E4-0000H
Big/Little Endian Descriptor Register	GCS6:08CH	0000-0000H
Expansion ROM Range Register	GCS6:090H	FFF8-0000H
Expansion ROM Local Base Address (Remap) Register and BREQo Control	GCS6:094H	0000-0012H
Local Address Space 0/Expansion ROM Bus Region Descriptor Register	GCS6:098H	47C3-24C3H
Local Range Register for Direct Master to PCI	GCS6:09CH	8000-0000H
Local Bus Base Address Register for Direct Master to PCI Memory	GCS6:0A0H	0000-0000H
Local Base Address Register for Direct Master to PCI IO/CFG	GCS6:0A4H	8000-0000H
PCI Base Address (Remap) Register for Direct Master to PCI Memory	GCS6:0A8H	0000-2807H
PCI Configuration Address Register for Direct Master to PCI IO/CFG	GCS6:0ACH	0000-0000H
Local Address Space 1 Range Register for PCI-to-Local Bus	GCS6:170H	FFF8-0000H
Local Address Space 1 Local Base Address (Remap) Register	GCS6:174H	0000-0000H
Local Address Space 1 Bus Region Descriptor Register	GCS6:178H	0000-02C3H

6.3.5. Device Numbers

PCI bus device numbers are assigned as shown in the following table.

Device	Device number
PCI slot 1	No.8 (AD19)
PCI slot 2	No.9 (AD20)
LAN controller (SB82558)	No.10 (AD21)
M1523B (SouthBridge) configuration register	No.7 (AD18)
USB controller (M1523B on-chip implementation)	Selected from among No.17 to No.20 (AD28 to AD31), Default: No. 20
IDE controller (M1523B on-chip implementation)	Selected from among No.13 to No.16 (AD24 to AD27), Default: No. 16

The device numbers of the USB controller and IDE controller can be selected by a setting in the M1523B configuration register.

6.3.6. PCI Slots

The motherboard has two slots for a PCI board.

6.3.6.1.Power supply

Although +5 V, +12 V, and -12 V power are supplied to each PCI slot, +3.3 V power is not supplied. Also, when +12 V or -12 V power is not supplied to the motherboard, it is also not supplied to the PCI slots.

6.3.6.2.Device Numbers

For device numbers that are assigned to each slot, see Section 6.3.5, "Device Numbers."

6.3.6.3.Interrupts

The interrupts of each PCI slot are directly connected to the interrupt control circuit of the local bus and do not pass through the SouthBridge chip. For details, see Section 6.2.7.1, "Overview of Interrupt Resources".

6.3.7. LAN Controller (SB82558)

6.3.7.1.Specifications

The SB82558 is an Intel LAN controller having the following features.

- Compatible with IEEE802.3/802.3u 10BASE-T and 100BASE-TX.
- Each transmission/reception FIFO is 3K bytes in size.
- Conforms to the Advanced Configuration and Power Interface (ACPI) specifications.
- Conforms to the PCI Power Management specifications Revision 1.0.
- Conforms to the Advanced Power Management (APM) specifications Revision 1.2.
- Supports ACPI Wake-up packets.
- Supports IEEE802.3u 100BASE-TX and 10BASE-T automatic negotiation.
- Supports half duplex/full duplex communication at 10 and 100 Mbps.

The RTE-MOTHER-A motherboard does not support the power management-related functions of the SB82558.

6.3.7.2.Device Numbers

For the device numbers that are assigned, see Section 6.3.5, "Device Numbers." The vendor ID of this device (configuration register) is 8086H, and the device ID is 1229H. Also, the subsystem vendor ID is 1410H, and the subsystem device ID is 0040H.

6.3.7.3.Addresses

Since this is a device that is connected to the PCI bus, the allocated addresses are determined by the value set in the register for the SB82558 configuration space.

6.3.7.4.Interrupts

The SB82558 INTA# pin is directly connected to the interrupt control circuit of the local bus and does not pass through the SouthBridge chip. For details, see Section 6.2.7.1, "Overview of Interrupt Resources".

6.3.8. M1523B (SouthBridge)

6.3.8.1.Specifications

The M1523B is a SouthBridge LSI chip for an ALI PC/AT motherboard. The following functions are implemented on the M1523B chip.

- PCI-ISA bridge function
- IDE controller
- SMM function
- Stop clock control
- APM function
- USB controller
- Distributed DMA function
- Interrupt controller function (i8259 subset × 2)
- Timer/counter function (i8254)
- PS/2 type keyboard/mouse controller

The RTE-MOTHER-A motherboard does not support the power management-related functions (SMM, Stop clock, APM).

6.3.8.2.Device Numbers

For the device numbers that are assigned, see Section 6.3.5, "Device Numbers." The vendor ID of this device (configuration register) is 10B9H, and the device ID is 1523H.

6.3.8.3.Configuration Functions

The following items can be configured (summary) according to the M1523B configuration space register on the PCI bus.

- Enabling/disabling of write post function or line buffer of PCI-ISA bridge function
- ISA bus clock
- ISA bus cycle timing
- I/O recovery time
- Enabling/disabling of on-chip IDE controller
- Enabling/disabling of on-chip keyboard interface
- Switching between PS/2 mouse and AT mouse
- Enabling/disabling of DMA controller high address
- Interrupt routing
- PCI bus interrupt line mode
- Power management control
- Distributed DMA functions
- IDE/USB controller IDSEL address
- Write-only register's write data reading function

6.3.8.4.Hardware-type Connections

The hardware-type connections to the M1523B chip are as follows.

- The on-chip keyboard and mouse interface function can be used.
- The on-chip USB interface function can be used.
- The on-chip IDE interface function can be used.
- For interrupts from the IDE, the primary IDE is connected to the SIRQII pin and the secondary IDE is connected to the SIRQI pin.
- Two slots, which are connected as ISA slots, and a PCMCIA controller (RF5C396) are connected to the ISA bus.
- No BIOS ROM is connected.
- The PCI interrupt input line is always inactive. PCI interrupts are directly connected to the interrupt controller of the motherboard and do not pass through the M1523B chip.
- The M1523B INTR output pin is connected to the ISA_INTR pin of the motherboard's interrupt controller.
- The M1523B NMI output pin is connected to the ISA_NMI pin of the motherboard's interrupt controller.
- The result of a logical OR operation performed on the output from the voltage monitoring circuit and the GBUS GRESETI- signal is connected to the M1523B PWG (power good) input pin.
- The M1523B RSTDRV output pin is connected to the GBUS GRESETO- signal and used as a motherboard system reset.
- The M1523B speaker output is output to the motherboard's JPANEL connector.
- The M1523B SPLED output pin is connected to the motherboard's SPEED-LED.
- The M1523B FERR-/IRQ13 input pin is fixed at Low level.
- A DMA request of the PCMCIA controller (RF5C396) is connected to the M1523B DRQ7 input pin and is not connected to an ISA slot.
- An interrupt request of the RTC (M5819P) is connected to the M1523B IRQ8- input pin.

6.3.9. USB Controller (M1523B (SouthBridge) On-chip Implementation)

6.3.9.1.Specifications

The USB controller that is implemented on the M1523B (SouthBridge) chip conforms to USB 1.0 specifications and OpenHCI1.0a specifications. It also has functions for handling a USB-compliant keyboard or mouse as a compatible legacy device.

If the CPU board cannot access to the motherboard by using a 32-bit cycle, this CPU board cannot use USB functions. Because OpenHCI specification requires a 32-bit access to USB registers.

6.3.9.2.Device Numbers

For the device numbers that are assigned, see Section 6.3.5, "Device Numbers." The vendor ID of this device (configuration register) is 10B9H, and the device ID is 5237H.

6.3.9.3.Addresses

Since this is a device that is connected to the PCI bus, the allocated addresses are determined by the value set in the register for the USB controller's configuration space.

6.3.9.4.Interrupts

The USB controller's INTA# pin is internally routed inside the M1523B. It can be routed to any interrupt input of the i8259 controller that is implemented on the M1523B chip according to an M1523B configuration register setting.

6.3.9.5.Power Supply

The ON/OFF state of the +5 V power that is supplied to the USB connector is controlled by the control register of the local bus. (See Section 6.2.9.2, "POWER Control Register (POWER_CONTROL GCS2:0000-8020H) [Read/Write].") Also, an over current state can be read (see Section 6.2.9.3, "POWER Status Register (POWER_STATUS GCS2:0000-8030H) [Read Only]"), and an interrupt can be generated by generating an over current state (see Section 6.2.7.1, "Overview of Interrupt Resources").

6.3.9.6.Configuration Register

The configuration space registers of the USB controller contain no USB function-related registers.

6.3.10. IDE Bus Master Controller (M1523B (SouthBridge) On-chip Implementation)

6.3.10.1.Specifications

This is a controller for performing bus master transfers to the IDE bus that is implemented on the M1523B (SouthBridge) chip. Besides this controller, the IDE interface has a separate register for supporting PIO mode.

6.3.10.2.Device Numbers

For the device numbers that are assigned, see Section 6.3.5, "Device Numbers." The vendor ID of this device (configuration register) is 10B9H, and the device ID is 5219H.

6.3.10.3.Addresses

Since this is a device that is connected to the PCI bus, the allocated addresses are determined by the value set in the register for the IDE controller's configuration space.

6.3.10.4.Interrupts

The IDE bus master controller's INTA# pin and INTB# pin are internally routed inside the M1523B. They can be routed to any interrupt input of the i8259 controller that is implemented on the M1523B chip according to an M1523B configuration register setting.

6.3.10.5.Configuration Register

The configuration space registers of the IDE bus master controller contain no IDE function-related registers.

6.4. ISA Bus

The ISA bus is bridged from the PCI bus inside the M1523B (SouthBridge) chip. This section describes the devices that are connected to the ISA bus.

6.4.1. ISA Bus-connected Devices

The following devices are connected to the ISA bus.

- IDE controller (PIO mode, M1523B on-chip implementation)
- DMA controller (M1523B on-chip implementation)
- Interrupt controller (i8259 subset × 2, M1523B on-chip implementation)
- Timer/counter (i8254, M1523B on-chip implementation)
- PS/2 type keyboard/mouse controller (M1523B on-chip implementation)
- RTC (Real Time Clock, M5819)
- ISA slot × 2
- PCMCIA controller (RF5C396)

6.4.2. M1523B On-chip Legacy Devices

The following table shows legacy devices that are implemented on the M1523B chip and their addresses. The I/O addresses in the table indicate I/O address on the PCI bus.

I/O address	Register name
0000H	DMA1 (slave) CH0 Base and Current Address
0001H	DMA1 (slave) CH0 Base and Current Count
0002H	DMA1 (slave) CH1 Base and Current Address
0003H	DMA1 (slave) CH1 Base and Current Count
0004H	DMA1 (slave) CH2 Base and Current Address
0005H	DMA1 (slave) CH2 Base and Current Count
0006H	DMA1 (slave) CH3 Base and Current Address
0007H	DMA1 (slave) CH3 Base and Current Count
0008H	DMA1 (slave) Status/Command
0009H	DMA1 (slave) Write Request
000AH	DMA1 (slave) Write Single Mask Bit
000BH	DMA1 (slave) Write Mode
000CH	DMA1 (slave) Clear Byte Pointer
000DH	DMA1 (slave) Master Clear
000EH	DMA1 (slave) Clear Mask
000FH	DMA1 (slave) Read/Write All Mask Register Bits
0020H	INT_1 (master) Control Register
0021H	INT_1 (master) Mask Register
0040H	Timer Counter – Channel 0 Count
0041H	Timer Counter – Channel 1 Count
0042H	Timer Counter – Channel 2 Count
0043H	Timer Counter Command Mode Register
0060H	Clear IRQ12 (for PS2), IRQ1 Latched Status / Keyboard Data Buffer
0061H	NMI and Speaker Status and Control (PORTB)
0064H	Keyboard Status/Command
0070H	CMOS RAM Address Port and NMI Mask Register
0071H	CMOS Data Register Port

I/O address	Register name
0081H	DMA Channel 2 Page Register
0082H	DMA Channel 3 Page Register
0083H	DMA Channel 1 Page Register
0087H	DMA Channel 0 Page Register
0089H	DMA Channel 6 Page Register
008AH	DMA Channel 7 Page Register
008BH	DMA Channel 5 Page Register
008FH	Refresh Address Register for Address 23 to 17
0092H	FAST RC/GATE-A20 Register
00A0H	INT_2 (slave) Control Register
00A1H	INT_2 (slave) Mask Register
00C0H	DMA2 (master) CH0 Base and Current Address
00C2H	DMA2 (master) CH0 Base and Current Count
00C4H	DMA2 (master) CH1 Base and Current Address
00C6H	DMA2 (master) CH1 Base and Current Count
00C8H	DMA2 (master) CH2 Base and Current Address
00CAH	DMA2 (master) CH2 Base and Current Count
00CCH	DMA2 (master) CH3 Base and Current Address
00CEH	DMA2 (master) CH3 Base and Current Count
00D0H	DMA2 (master) Status/Command
00D2H	DMA2 (master) Write Request
00D4H	DMA2 (master) Write Single Mask Bit
00D6H	DMA2 (master) Write Mode
00D8H	DMA2 (master) Clear Byte Pointer
00DAH	DMA2 (master) Master Clear
00DCH	DMA2 (master) Clear Mask
00DEH	DMA2 (master) Read/Write All Mask Register Bits
00F0H	Coprocessor Error Ignored Register
0170H	IDE2 (Secondary) Data Register
0171H	IDE2 (Secondary) Error Register
0172H	IDE2 (Secondary) Sector/Count Register
0173H	IDE2 (Secondary) Sector Number Register
0174H	IDE2 (Secondary) Cylinder Number Low Register
0175H	IDE2 (Secondary) Cylinder Number High Register
0176H	IDE2 (Secondary) Drive/Head Register
0177H	IDE2 (Secondary) Command/Status Register
01F0H	IDE1 (Primary) Data Register
01F1H	IDE1 (Primary) Error Register
01F2H	IDE1 (Primary) Sector/Count Register
01F3H	IDE1 (Primary) Sector Number Register
01F4H	IDE1 (Primary) Cylinder Number Low Register
01F5H	IDE1 (Primary) Cylinder Number High Register
01F6H	IDE1 (Primary) Drive/Head Register
01F7H	IDE1 (Primary) Command/Status Register
0376H	IDE2 (Secondary) Alternate Status Register
0377H	IDE2 (Secondary) Drive Address Register
03F6H	IDE1 (Primary) Alternate Status Register
03F7H	IDE1 (Primary) Drive Address Register
040BH	DMA1 Extended Mode Register
0481H	DMA CH2 High Page Register
0482H	DMA CH3 High Page Register
0483H	DMA CH1 High Page Register
0487H	DMA CH0 High Page Register
0489H	DMA CH6 High Page Register
048AH	DMA CH7 High Page Register
048BH	DMA CH5 High Page Register
04D0H	INT_1 (master) Edge/Level Control
04D1H	INT_2 (slave) Edge/Level Control
04D6H	DMA2 Extended Mode Register

6.4.3. RTC (Real Time Clock M5819)

The ALI M5819 is installed as the real time clock (RTC).

6.4.3.1.Addresses

The addresses shown in the following table are used to access the RTC.

Address	Function
PCI-I/O:0000-0070H	CMOS RAM Address Port
PCI-I/O:0000-0071H	CMOS Data Register Port

Address 70H is multiplexed with a register that is implemented within the M1523B chip.

6.4.4. ISA Slots

The motherboard has two slots for a 16-bit ISA board.

6.4.4.1.Power Supply

+5 V, -5 V, +12 V, and -12 V power are supplied to each ISA slot. Also, when -5 V, +12 V or -12 V power is not supplied to the motherboard, it is also not supplied to the ISA slots.

6.4.4.2.Hardware-type Connections

Both of the ISA slots are connected as follows.

- Although IRQ3 to IRQ7, IRQ9 to IRQ11, IRQ14, and IRQ15 can be used for interrupts, since IRQ3 to IRQ5, IRQ7, IRQ11, and IRQ14 are also connected to the PCMCIA controller (RF5C396), they must be shared.
- Although DRQ0 to DRQ3 and DRQ4 to DRQ7 can be used for DMA lines, since DRQ7 is also connected to the PCMCIA controller (RF5C396), it must be shared.
- The frequency of the OSC signal is 14.31 MHz.
- Either 1/2 the frequency of the OSC signal or 1/2, 1/3, 1/4, 1/5, or 1/6 the frequency of the PCI clock (33.33 MHz) can be selected for the CLK signal. This option is selected according to the M1523B configuration register.

6.4.5. PCMCIA Controller (RF5C396)

6.4.5.1.Specifications

The RF5C396 is a Ricoh PCMCIA controller LSI chip. It is PCMCIA2.1-compliant and JEDA4.2-compliant, and supports two PCMCIA slots. The data sheet for the RF5C396 is available from the Ricoh home page (http://www.ricoh.co.jp/).

6.4.5.2.Addresses

The addresses shown in the following table are used to access the RF5C396.

Address	Function
PCI-I/O:0000-03E0H	Index register
PCI-I/O:0000-03E1H	Data register

6.4.5.3.Hardware-type Connections

The various pins of the RF5C396 are connected as shown in the following table.

RF5C396 pin name	Connection
IRQ3	ISA bus IRQ3
IRQ4	ISA bus IRQ4
IRQ5	ISA bus IRQ5
IRQ7	ISA bus IRQ7
IRQ9/DACK-	ISA bus DACK7-
IRQ10/DREQ	ISA bus DRQ7
IRQ11/TC	ISA bus IRQ11
IRQ12/ATA_LED-	Turns IDE-LED on and off
IRQ14	ISA bus IRQ14
IRQ15/TC	ISA bus TC
SPKROUT-	JPANEL speaker output
RI_OUT-	Not connected
INTR-	Not connected

6.5. PCMCIA Bus

The PCMCIA bus is bridged as PCI bus → ISA bus → PCMCIA bus.
This section describes the PCMCIA bus.

6.5.1. Slots

PCMCIA slot 1 (JPCMCIA1) is the SLOT#0 side of the PCMCIA controller (RF5C396) and PCMCIA slot 2 (JPCMCIA2) is the SLOT#1 side.

6.5.2. Power Supply

For the power supplied to the PCMCIA slots, either +12 V or +5 V can be supplied as VPP and either +5 V or +3.3 V can be supplied as VCC. The ON/OFF state of the power supply and the voltage are selected according to a register in the RF5C396. The following table shows the settings for the RF5C396 and the power supplied to the PCMCIA slots.

Index:02H			Index:2FH	Voltage supplied to PCMCIA slot	
Bit4	Bit1	Bit0	Bit0	VCC pin	VPP pin
1	0	0	0	5V	0V
1	0	0	1	3V	0V
1	0	1	0	5V	5V
1	0	1	1	3V	3V
1	1	0	0	5V	12V
1	1	0	1	3V	12V
1	1	1	0	5V	OPEN
1	1	1	1	3V	OPEN
0	X	X	X	0V	0V

Although it cannot be detected individually for each slot, an over current state can be detected for the power supplied to the PCMCIA slots. (See Section 6.2.9.3, "POWER Status Register (POWER_STATUS GCS2:0000-8030H) [Read Only]".) An interrupt can be generated when an over current state is detected. (See Section 6.2.7.1, "Overview of Interrupt Resources.")

7. SOFTWARE

This chapter describes software for operating the hardware on the motherboard.

7.1. Sample Programs

The sample programs on the CD-ROM included with the RTE-MOTHER-A motherboard will operate various circuits on the motherboard and check the operating status. Some checks may require special-purpose jigs, and although the jigs cannot be operate directly, the source code will provide a reference for how to control them.

7.1.1. Precautions Concerning the Sample Programs

Note the following points when referencing the sample programs.

- The sample programs have been written for Green Hills Software's C compiler Ver. 1.8.9.
- Since the sample programs were written for checking the operation of hardware, the control procedures or hardware settings are not necessarily reasonable. For example, interrupts are not prohibited for portions in which interrupts must be prohibited according to the application characteristics, the setting sequence for I/O is inappropriate, or portions will not operate if an optimizing compilation mode is used.
- The sample programs have been written with an emphasis placed on reducing the work required for porting them to different CPU boards. Therefore, execution efficiency has not been taken into account.
- The sample programs use the safest possible contents for settings (configuration space) related to the MB1523B (SouthBridge) chip. Therefore, performance has been sacrificed somewhat for applications that frequently access the PCI bus, ISA bus, or PCMCIA bus.

7.1.2. Overview of Sample Programs

The processing or checks performed by using the sample programs' various files or directories are described below.

_CPU-name directory: Individual files for each CPU board are stored in a directory whose name begins with an underscore. The sample programs can be built by using the *.bld files that are in this directory.

_CPU-name¥src directory: This directory contains files (*.c and *.h) specific to each CPU board.

common directory: This directory contains common files that are not dependent on the contents of specific checks. It contains various types of header files and initialization/setting program files.

common¥pci.c: This file contains initialization routines for the configuration spaces of devices connected to the PCI bus. It contains initialization routines related to the ISA, IDE, or USB of the M1523B (SouthBridge) chip and an initialization routine for the SB82558 (LAN controller).

common¥pcmcia.c: This file contains an initialization routine for the RF5C396 (PCMCIA controller).

common¥rtev832_pc.c: This file contains an initialization routine that is used when a Midas lab RTE-V832-PC board is connected to the PCI slot. It also detects whether or not the board is installed in the PCI slot.

audio_dma_test directory: This directory contains a program for testing the motherboard audio recording/playing circuits. DMA transfers are used for transferring audio data. Two blocks are allocated as an audio data buffer. When one block of data has been recorded, playing begins while recording continues. When recording/playing is executed a fixed number of times, the program stops.

audio_soft_test directory: This directory contains a program for testing the motherboard audio recording/playing circuits. Although the contents are equivalent to those of audio_dma_test, DMA transfers are not used for transferring audio data. Instead, the audio data is transferred under software control by using a Half-Full interrupt.

beep_test directory: This directory contains a program for testing the Beep tone when the speaker is connected to the JPANEL connector of the motherboard.

extbus16_test directory: This directory contains a program for testing the functions of the JEXT16 connector on the motherboard. A special-purpose jig must be connected to the JEXT16 connector for this test. The checks performed are an EXT-BUS memory check and a check of interrupts from the EXT-BUS.

extbus32_test directory: This directory contains a program for testing the functions of the JEXT32 connector on the motherboard. A special-purpose jig must be connected to the JEXT32 connector for this test. The checks performed are an EXT-BUS memory and I/O check, an EXT-BUS 16-bit mode check, an EXT-BUS DMA transfer check, and a check of interrupts from the EXT-BUS.

flash_test directory: This directory contains a program for testing the erasing or writing to flash ROM on the motherboard. At the end of the test, the contents of the CPU board's ROM are written to the flash ROM.

ide_test directory: This directory contains a program for checking accesses to a hard disk that is connected to the JIDE connector on the motherboard. The checks performed are a hard disk reset, reading of hard disk information (displaying the model number and firmware version, which are part of the information), and reading sector 0 (partition information) by using a DMA transfer and confirming that the last data is 0x55aa.

isa_bm_sram_test_mb directory: This directory contains a program that is a companion to a program for checking access to common RAM (SRAM) on the motherboard from the bus master on the ISA bus. A special-purpose jig must be connected for this test.

key_mouse_test directory: This directory contains a program for checking the keyboard and mouse connectors on the motherboard. The keyboard check resets the keyboard and then makes the keyboard LED blink and confirms that "q" is entered. The mouse check also connects the keyboard and confirms that "q" is entered. Both checks are performed with the keyboard is connected to each connector after the user is prompted to do so by a message.

lan_test directory: This directory contains a program for writing the LAN's MAC address to the serial EEPROM that is connected to the LAN controller (SB82558) and then verifying that MAC address and a program for reading the LAN controller's vendor/device ID.

pci_sram_test_mb directory: This directory contains a program that is a companion to a program for checking access to common RAM (SRAM) on the motherboard from the bus master on the PCI bus.

pcmcia_test directory: This directory contains a program for testing the functions of the JPCMCIA1 and JPCMCIA2 connectors on the motherboard. The functions that are tested are the switching of the VCC and VPP power supplies, the generation of an over current interrupt, an interrupt due to a change in the card status, an access to the attribute space, and an access to the memory space. A special-purpose PCMCIA card (IO DATA's PCM-2M) is required to test the access to the attribute space and access to the memory space.

printer_uart_test directory: This directory contains a program for testing the functions of the JSIO1, JSIO2, and JPRT connectors on the motherboard. A special-purpose jig must be connected to these connectors for this test.

rte_timer_int_test directory: This directory contains a program for generating interrupts according to the GINTI-[1:0] signals that are output from the RTC and timer (8254) on the motherboard and from the CPU board (normally, TOUT0/TOUT1 of the μPD71054 on the CPU board is connected) and for testing the timer-related interrupts. For signals from the RTC, interrupts are generated at an interval of approximately 1 ms from updates (1 s), and for signals from the timer, interrupts are generated at an interval of approximately 55 ms. For signals from the μPD71054 on the CPU board, interrupts are generated at an interval of 60 Hz (16.667 ms) and 40 Hz (25 ms). RTC date information is read and displayed on the screen for every three update

interrupts. At this time, the accumulated number of seconds due to the various interrupts of 1 ms, 55 ms, 16.667 ms, and 25 ms are displayed together with the date information.

sram_test directory: This directory contains a program for memory testing all areas of common RAM (SRAM) on the motherboard.

usb_test directory: This directory contains a program for testing the functions of the USB connectors on the motherboard. The functions that are tested are the ON/OFF state of the power supply and the reading signatures from connected USB device and the generation of an over current interrupt. The reading signatures from connected USB device is not work correctly on the CPU board that cannot access to the motherboard by using a 32-bit cycle.

7.1.3. Sample Program Resource Allocation

The sample programs allocate the resources of the PCI bus or ISA bus as follows.

Memory:

PCI bus address	Allocated resource
0000-0000H to 0007-FFFFH	Common RAM (SRAM) on the motherboard (same area as 0080-0000H to 0087-FFFFH). Part of common RAM is also allocated here so that it can be accessed by the bus master of the ISA bus.
007F-E000H to 007F-EFFFH	USB controller
007F-FE00H to 007F-FEFFH	RTE-V832-PC that is installed in the PCI slot
007F-FF00H to 007F-FFFFH	PCI9080 register
0080-0000H to 009F-FFFFH	Common RAM (SRAM) on the motherboard
0200-0000H to 03FF-FFFFH	Area for accessing memory on the CPU board (when the CPU board supports direct access)
FFF8-0000 to FFFF-FFFFH	RTE-V832-PC that is installed in the PCI slot

I/O:

PCI bus address	Allocated resource
F000H to F007H	IDE Primary controller
F008H to F00FH	IDE Secondary controller
F800H to F81FH	i82258 (LAN controller)
FE00H to FEFFH	RTE-V832-PC that is installed in the PCI slot
FF00H to FFFFH	PCI9080 register

Interrup:

ISA interrupt number	Allocated resource
IRQ0	System Timer (SouthBridge on-chip 8254 channel 0)
IRQ1	PS/2 keyboard
IRQ2	Unused
IRQ3	Unused (With a PC/AT compatible machine, this is generally used by COM2.)
IRQ4	Unused (With a PC/AT compatible machine, this is generally used by COM1.)
IRQ5	Unused
IRQ6	Unused (With a PC/AT compatible machine, this is generally used by the floppy disk.)
IRQ7	Unused (With a PC/AT compatible machine, this is generally used by the printer.)
IRQ8	RTC
IRQ9	USB controller
IRQ10	Unused
IRQ11	PCMCIA controller
IRQ12	PS/2 mouse
IRQ13	Unused (With a PC/AT compatible machine, this is generally used by the FPU.)
IRQ14	Primary IDE controller
IRQ15	Secondary IDE controller

7.2. LAN Controller

To use the SB82558 (LAN controller), obtain the SB82558 driver from manufacturer of the real-time OS or middleware.
Note the following points when porting the obtained driver for use with the RTE-MOTHER-A motherboard.

- The MAC address can be obtained from the serial EEPROM that is connected to the LAN controller. (This method is described later.)
- The work area for providing LAN controller command packets will be a common RAM (SRAM) area on the motherboard. When setting the address of this work area for the LAN controller or when writing data in command packets, the work area address in the address space of the PCI bus will be specified. However, the address when this work area is viewed from the CPU board will differ from the PCI bus address.
  Therefore, the driver must properly use both the address on the PCI bus and the address viewed from the CPU.
  Note that a driver created for use in a PC/AT-compatible machine is generally written under the assumption that the address on the PCI bus and the address viewed from the CPU are equal. Also, note that the work area for command packets may be allocated in the stack for the same reason.

The data shown in the following table is entered in the serial EEPROM that is connected to the LAN controller. The notation MAC[x:y] in the table indicates MAC address bits, and the contents will be the same as the MAC address printed on the sticker affixed to the LAN controller.
The soundness of the related data within the serial EEPROM other than the MAC address and subsystem vendor/device ID should be determined according to the purpose of the driver or application.

Address	Data (D[15:0])
00H	MAC[39:32], MAC[47:40]
01H	MAC[23:16], MAC[31:24]
02H	MAC[7:0], MAC[15:8]
03H	0100H
04H	0000H
05H	0201H
06H	4701H
07H	0000H
08H	0000H
09H	0000H
0AH	4C01H
0BH	0040H (Subsystem Device ID)
0CH	1410H (Subsystem Vendor ID)
0DH to 3EH	0000H
3FH	CheckSum

7.3. General Precautions

General precautions that should be taken into account concerning programs are described below.

- Be careful concerning problems that may occur because the address on the PCI bus differs from the address viewed from the CPU, as described in the section concerning the LAN controller. A driver created for use in a PC/AT-compatible machine is often written under the assumption that the address on the PCI bus and the address viewed from the CPU are equal. Therefore, be careful concerning the address that is set when common RAM (SRAM) on the motherboard is accessed by the bus master on the PCI bus, ISA bus, or PCMCIA or by a DMA within the M1523B (SouthBridge) chip.
  A common RAM address must be set by using an address on the PCI bus in a setting for a DMA within the SouthBridge chip or for a board on the PCI/ISA bus. However, that address differs from the address used when the common RAM is accessed from the CPU.
  For standard resources that are installed on the motherboard, this precaution must be taken into account when the LAN controller, IDE controller, or USB controller bus master function is used.
- Some legacy devices will not operate normally when accessed by using a 16-bit read cycle even when consecutive addresses are used for I/O. For example, an access to the real-time clock (RTC) must be an 8-bit access. An 8-bit read cycle will not be generated by some CPUs (such as a CPU with no byte-enable signal or a CPU for which all byte enable signals are always active during a read access). For a Midas lab CPU board on which this kind of CPU is installed (for example, RTE-V850E/MA1-CB), a control port for forcing an 8-bit access is provided for a read cycle to the motherboard. This port should be used.
- When the CPU's data bus is 16 bits, a 32-bit access to the motherboard cannot be performed. For this kind of CPU, a board that requires a 32-bit access cannot be connected to the EXT-BUS or PCI bus.

8. GBUS SPECIFICATIONS

This section explains the GBUS specifications.

8.1. Terminology

The terminology used in this section is explained below.

8.1.1. CPU Board and Motherboard

A board in the RTE-CB series is called a CPU board, and the RTE-MOTHER-A is called a motherboard.

8.1.2. Bus Cycle and Micro Cycle

GBUS is a general bus that can be accessed in burst mode.
A bus cycle, including cases when the access is in burst mode, is a sequence of cycles that is completed (GADS- must be asserted once to end a bus cycle).
Bus cycles are classified into single cycles and burst cycles. A single cycle is a bus cycle in which data is transferred only once. A burst cycle is a bus cycle in which data is transferred multiple times. The cycle for each data transfer of a burst cycle is called a micro cycle.

8.2. Signals

The following table describes the GBUS signals. The input/output direction of each GBUS signal is described as viewed from the motherboard. Therefore, "input" means that a signal is output from the CPU board and input to the motherboard. (This criterion also applies to signal names.)
Signals described as "bidirectional" in the table change direction depending on the bus cycle status. Also, signals described as "input/output" change direction depending on whether the bus master is the CPU board or motherboard. The direction written first is the signal direction when the CPU board is the bus master, and the direction written second is the signal direction when the motherboard is the bus master.
A GBUS signal is a +5 V TTL level signal. The motherboard is always little endian.

Signal name	Input/output	Function
GCLK	Input	• GBUS synchronization clock. The maximum frequency is 33.33 MHz, and the minimum frequency is 10.0 MHz. The GBUS operates synchronized with the rising edge of this clock. • On the motherboard, this clock is terminated at 330 Ω with respect to +5 V and GND. Therefore, the circuit on the CPU board must be able to drive this resistance. • If the GCLK frequency is less than 16.67 MHz, GCLK_LOW- goes low. In this way, the motherboard can adjust the number of waits. • Since a Phase Lock Loop (PLL) zero delay buffer is used for this signal, when the GCLK frequency is changed, the motherboard must not be accessed for at least 1 ms after the frequency has been changed to allow the PLL to be locked.
GRESETI-	Input	• GBUS reset signal. When a reset occurs on the CPU board, this signal goes low. The motherboard is reset by this signal as well as by a power-on reset or a reset due to the reset switch.
GRESETO-	Output	• This signal goes low when a motherboard reset occurs. • The motherboard ORs the reset signal on the motherboard with GRESETI- as GRESETO-. Therefore, the CPU board resets the circuits on the CPU board by ORing GRESETI- and GRESETO-. (GRESETI- and GRESETO- are ORed because there is a possibility that the motherboard is not connected.)

Signal name	Input/output	Function
GADDR[31:2]	Input/output	• GBUS address signals. These signals are always driven by a valid value during a cycle. • GADDR[31] is ignored on the motherboard when the CPU is the bus master. A[31] in the motherboard becomes "1" for an access for which GCS7- is active, that is, for an access to the I/O space of the PCI bus. Otherwise, A[31] becomes "0." • The low-order addresses A1 and A0 use a byte enable signal. • The GAHI_EN- signal enables GADDR[31:26] from the CPU board to be treated as 0. • When the bus master is the motherboard, if GADDR[25] is 0, it indicates that the SRAM on the motherboard is selected. If GADDR[25] is 1, it indicates that a resource on the CPU board is selected.
GBEN-[3:0]	Input/output	• Byte enable signals of GBUS. These signals are always driven by a valid value during a cycle. • GBEN0-, GBEN1-, GBEN2-, and GBEN3- correspond to byte lanes GDATA[7:0], GDATA[15:8], GDATA[23:16], and GDATA[31:24], respectively, and the corresponding byte lane is valid when GBENx- is low.
GDATA[31:0]	Bidirectional	• GBUS bus data signals. • These signals are pulled up to 10 kΩ on the motherboard. • The direction of these signals is determined by GW/R-.
GADS-	Input/output	• GBUS address strobe signals. If this signal is sampled low on the rising edge of GCLK, it indicates the start of a bus cycle. • The motherboard ignores GADS- if none of the chip select signals (GCS-[7:0]) is active.
GREADY-	Output/input	• GBUS ready signal. If this signal is sampled low and GWAITI is sampled high on the rising edge of GCLK during a micro cycle, it indicates the end of the micro cycle. • The motherboard generates a time-over ready when the CPU board accesses the motherboard. This is done to avoid a collision with a GREADY- signal.
GWAITI-	Input	• Wait request signal. This signal is sampled on the rising edge of GCLK. • If the CPU board cannot support a cycle with few waits, the CPU board samples GWAITI- low according to the GREADY- sample timing so that the motherboard cannot handle GREADY- as a ready signal even though it is low at the time. Usually, this signal is used if the CPU board cannot support zero wait burst. (See Section 8.6.3, "GWAITI-.") • This signal is valid only in a cycle in which the CPU board is the bus master.
GBLAST-	Input/output	• Bus cycle completion notification signal. This signal is sampled on the rising edge of GCLK. • This signal is asserted low by the bus master when a micro cycle that completes the bus cycle starts. • The bus cycle is completed when the low level of GBLAST-, the low level of GREADY-, and the high level of GWAITI- are sampled on the rising edge of GCLK.
GBTERM-	Output/input	• Bus cycle completion request signal. This signal is sampled on the rising edge of GCLK. • When the accessed side requests completion of the bus cycle, the GREADY- and GBTERM- signals go low. If the bus master samples GBTERM- as low when it samples GREADY- as low, it must complete the bus cycle even though GBLAST- has not been asserted and start the bus cycle again by asserting GADS- again. GBTERM- must be asserted at the same time as GREADY-. • This signal is used to complete the bus cycle when the accessed side does not support burst cycles or when a burst cycle exceeding the supported number of bursts is requested.
GW/R-	Input/output	• Write/Read signal. This signal indicates the direction of the data bus. It is always driven by a valid value during the bus cycle. • This signal indicates the direction of the data bus for the bus master.

Signal name	Input/output	Function
GCS-[7:0]	Input	• Chip select signals. These signals are always driven by a valid value during the bus cycle. • When the CPU board is the bus master, it makes the corresponding chip select signal active to specify the resources on the motherboard. • Each chip select signal specifies the type of memory or I/O space and the size of each space. (See Section 8.5, "Allocating GCS-[7:0].")
GRD-	Input	• Read timing signal. This signal is asserted when the CPU board is the bus master. • This signal is not used by the motherboard. • When the CPU has an RD- command signal, that signal is usually connected.
GWR-	Input	• Write timing signal. This signal is asserted when the CPU board is the bus master. • This signal is not used by the motherboard. • When the CPU has a WR- command signal, that signal is usually connected.
GHOLD-	Output	• Bus hold request signal. • This signal is asserted low when the motherboard accesses the resources on the CPU board to acquire bus mastership. • When the GUSE_DIRECT_ACC- signal is high, the GHOLD- signal indicates to the CPU board that the motherboard has no resources that can be accessed. In this case, the CPU board does not have to support GHOLD-.
GHLDA-	Input	• Bus hold acknowledge signal. • This signal indicates that the CPU board releases bus mastership of GBUS to the motherboard. It is asserted low at that time. • A CPU board that asserts the GUSE_DIRECT_ACC- signal high can disconnect the GHLDA- signal.
GBREQ-	Input	• Bus mastership release request signal. This signal is connected to the PCI9080 BREQ pin. • This signal is set to active to temporarily relinquish the bus mastership to the PCI9080 when the PCI9080 is using the GBUS. • If the motherboard is in a bus cycle when GBREQ- is asserted low, GBLAST- is asserted in the next micro cycle, the bus cycle is completed in the next micro cycle, and then the bus mastership is relinquished. • GBREQ- is used to temporarily return the bus mastership to the CPU board when the number of bursts of a PCI9080 bus cycle is high or when a high priority bus cycle such as a refresh cycle is pending on the CPU board. • The RTE-MOTHER-A motherboard has a function for limiting the interval that the PCI9080 maintains the bus right according to local bus control register setting. (See Section 6.2.9.6, "BREQ Control Register (BREQ_CONTROL GCS2:0000-8060H) [Read/Write].")
GDMARQ-[3:0]	Output	• DMA request signals. Only two-cycle DMA transfers are supported. Fly-by DMA is not supported. • These signals are asserted low when a DMA request is generated on the motherboard. • The CPU board must support all four DMA signals. The number of DMA signals that can be asserted at the same time and can be supported by the GDMAAK- signal depends on the CPU board. • The CPU board uses the DMAAK signal in preference to DMAAK-[3:2] when the correspondence between all four GDMARQ- signals and GDMAAK- signals cannot be established. • The motherboard uses GDMARQ0- for Audio playing, GDMARQ1- for Audio recording, GDMARQ2- for the EXT-BUS DMARQ0-, and GDMARQ3- for the EXT-BUS DMARQ1-.

Signal name	Input/output	Function
GDMAAK-[3:0]	Input	• DMA acknowledge signals. • These signals are asserted low to acknowledge DMA requests from the motherboard. • The CPU board uses the DMAAK signal in preference to DMAAK-[3:2] when the correspondence between all four GDMARQ- signals and GDMAAK- signals cannot be established. • The motherboard uses GDMAAK0- for Audio playing, GDMAAK1- for Audio recording, GDMAAK2- for the EXT-BUS DMAAK0-, and GDMAAK3- for the EXT-BUS DMAAK1-. • An Audio DMA transfer does not require a DMAAK- signal. Therefore, the CPU board need not support GDMAAK0- and GDMAAK1-. However, when GDMAAK-[1:0] are not supported, the timing for de-asserting the GDMARQ- signal is slower than when they are supported. • When GDMAAK2- and GDMAAK3- are not supported, the corresponding EXT-BUS DMAAK- signals cannot be asserted.
GINTO-[3:0]	Output	• Interrupt request signals. • Both level-sensitive and edge-sensitive interrupts can be supported. • A low level or a falling edge indicates the occurrence of an interrupt.
GINTI-[1:0]	Input	• Interrupt request signals. • These interrupt signals are used to combine an interrupt on the CPU board with an interrupt on the other motherboard and return the combined signal to GINTO-[3:0]. • Usually, OUT0 and OUT1 of the TIC (*m*PD71054) on the CPU board are connected. The motherboard can select the type of sensitivity and the polarity for these interrupt signals under program control.
GETC[7:0]	—	• CPU board-dependent signals. These signals are not used by the RTE-MOTHER-A motherboard. • The CPU board determines the contents of the GETC[7:0] signals, including the signal directions and signal contents. The CPU board uses these signals to exchange signals having a special purpose with the motherboard.
GAHI_EN-	Input	• Upper address valid signal. • If the CPU board is the bus master when this signal is low, it indicates that the CPU board is driving a valid value on GADDR[31:26]. When this signal is high, it indicates that the CPU board is not driving a valid value on GADDR[31:26], and the circuits on the motherboard perform processes with all of the GADDR[31:26] signals low.
GMOTHER_ DETECT-	Output	• Motherboard detection signal. • This signal is pulled up on the CPU board and is connected to GND on the motherboard. The CPU board uses this signal when it must determine if the motherboard is connected. For example, this signal is used by a CPU board time-over ready generation circuit.
GUSE_ DIRECT_ACC-	Input	• When this signal is low, it indicates that the CPU board has resources that can be accessed from the motherboard. • When this signal is high, Ready-based bus arbitration is performed, and GHOLD- does not become active even when the PCI9080 requests the GBUS bus mastership. • When this signal is low, GHOLD- becomes active when the PCI9080 requests the GBUS bus mastership. That is, HOLD-based bus arbitration is performed.
GCLK_LOW-	Input	• When this signal is low, it indicates that the GCLK frequency does not exceed 16.67 MHz. When this signal is high, it indicates that the GCLK frequency is between 16.67 MHz and 33.33 MHz. • The circuits on the motherboard use this signal to determine the number of waits required to access resources on the motherboard.
GBLOCK-[1:0]	Input	• Bus lock signals. These signals must be valid during a bus cycle and between bus cycles to be locked. • When a bus lock signal is output from the CPU, the bus lock signal is connected to the motherboard by using these pins. • The GBLOCK0- signal is valid for the GCS0- (SRAM) space. The GBLOCK1- signal is valid for the GCS5- (PCI bus memory) and GCS7- (PCI bus I/O) spaces. • The RTE-MOTHER-A motherboard has a function for supplying the equivalent functions of GBLOCK-[1:0] according to a local bus control register setting. (See Section 6.2.9.8, "Bus Lock Control Register (BLOCK_CONTROL GCS2:0000-8080H) [Read/Write].")

Signal name	Input/output	Function
+5V	Output	• Power supply. Supplies +5 V ±5% from the motherboard to the CPU board.
+12V	Output	• Power supply. Supplies +12 V ±10% from the motherboard to the CPU board. However, if +12 V is not supplied to the motherboard, it is also not supplied to the CPU board.

8.3. Pin Assignments

The following table shows the GBUS pin assignments. "Reserve" indicates a reserved pin, and "N/C" indicates that a pin is not connected.

No.	Signal name	No.	Signal name	No.	Signal name	No.	Signal name
1	+12V	2	+12V	3	GND	4	+5V
5	GADDR2	6	GADDR3	7	GADDR4	8	GADDR5
9	GADDR6	10	GADDR7	11	GND	12	+5V
13	GADDR8	14	GADDR9	15	GADDR10	16	GADDR11
17	GADDR12	18	GADDR13	19	GADDR14	20	GADDR15
21	GND	22	+5V	23	GADDR16	24	GADDR17
25	GADDR18	26	GADDR19	27	GADDR20	28	GADDR21
29	GADDR22	30	GADDR23	31	GND	32	+5V
33	GADDR24	34	GADDR25	35	GADDR26	36	GADDR27
37	GADDR28	38	GADDR29	39	GADDR30	40	GADDR31
41	GND	42	+5V	43	GBEN3-	44	GBEN2-
45	GBEN1-	46	GBEN0-	47	GND	48	+5V
49	GDATA31	50	GDATA30	51	GDATA29	52	GDATA28
53	GDATA27	54	GDATA26	55	GDATA25	56	GDATA24
57	GND	58	+5V	59	GDATA23	60	GDATA22
61	GDATA21	62	GDATA20	63	GDATA19	64	GDATA18
65	GDATA17	66	GDATA16	67	GND	68	+5V
69	GDATA15	70	GDATA14	71	GDATA13	72	GDATA12
73	GDATA11	74	GDATA10	75	GDATA9	76	GDATA8
77	GND	78	+5V	79	GDATA7	80	GDATA6
81	GDATA5	82	GDATA4	83	GDATA3	84	GDATA2
85	GDATA1	86	GDATA0	87	GND	88	+5V
89	GND	90	GW/R-	91	GBTERM-	92	GREADY-
93	GRESETI-	94	GADS-	95	GBLAST-	96	GWAITI-
97	GND	98	GCLK	99	GND	100	+5V
101	GCS0-	102	GCS1-	103	GCS2-	104	GCS3-
105	GCS4-	106	GCS5-	107	GCS6-	108	GCS7-
109	Reserve	110	Reserve	111	Reserve	112	Reserve
113	GRD-	114	GWR-	115	GND	116	+5V
117	GHOLD-	118	GHLDA-	119	GBREQ-	120	N/C
121	GDMARQ0-	122	GDMARQ1-	123	GDMARQ2-	124	GDMARQ3-
125	GDMAAK0-	126	GDMAAK1-	127	GDMAAK2-	128	GDMAAK3-
129	Reserve	130	Reserve	131	Reserve	132	Reserve
133	GND	134	+5V	135	GINTO0-	136	GINTO1-
137	GINTO2-	138	GINTO3-	139	GINTI0-	140	GINTI1-
141	GETC0	142	GETC1	143	GETC2	144	GETC3
145	GETC4	146	GETC5	147	GETC6	148	GETC7
149	Reserve	150	Reserve	151	GAHI_EN-	152	GMOTHER_DETECT -
153	GND	154	+5V	155	GUSE_DIRECT_ACC-	156	GCLK_LOW-
157	GRESETO-	158	GBLOCK0-	159	GBLOCK1-	160	N/C
161	N/C	162	N/C	163	N/C	164	N/C
165	N/C	166	N/C	167	N/C	168	N/C
169	N/C	170	N/C	171	N/C	172	N/C
173	N/C	174	N/C	175	N/C	176	N/C
177	GND	178	+5V	179	+12V	180	+12V

The following connectors are used.

CPU board side connector (straight) → KEL 8817-180-170L
Motherboard side connector (straight) → KEL 8807-180-170S
Motherboard side connector (L angle) → KEL 8807-180-170L

8.4. Processing of Unused Pins

Signals that are not used as input signals to the GBUS motherboard can be unconnected on the CPU board because pull-up or pull-down processing is performed on the motherboard. The following table shows signals that can be unconnected and the processing performed on the motherboard because they are unconnected.

Signal name	Processing
GADDR[31:26]	• When GADDR[31:26] are not used, GADDR[31:26] can be unconnected by setting the GAHI_EN- signal high or by disconnecting it. In this case, when the CPU is the bus master, all bits of GADDR[31:26] are treated as 0 on the motherboard.
GWAITI-	• Pull-up processing is performed.
GBLAST-	• Pull-up processing is performed.
GBTERM-	• Pull-up processing is performed.
GCS-[7:0]	• Pull-up processing is performed.
GHLDA-	• Pull-up processing is performed.
GBREQ-	• Pull-up processing is performed.
GDMAAK-[3:0]	• Pull-up processing is performed.
GINTI-[1:0]	• Pull-up processing is performed.
GAHI_EN-	• Pull-up processing is performed.
GUSE_DIRECT_ACC-	• Pull-up processing is performed.
GCLK_LOW-	• Pull-up processing is performed.
GBLOCK-[1:0]	• Pull-up processing is performed.

8.5. Allocating GCS-[7:0]

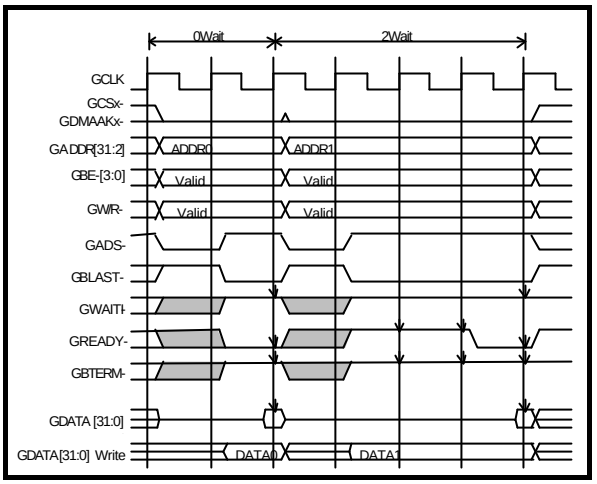
The following table shows the allocation of the chip select signals (GCS-[7:0]). All of the spaces can be accessed in a burst cycle. A space marked I/O under the heading "Recommended space" means that, if the CPU has an I/O space, it is recommended that the space be allocated as an I/O space. "Minimum range" indicates that the CPU board must allocate at least the indicated area for the corresponding chip select space. "Maximum range" indicates that, if the CPU board has an extra address range, addresses can be allocated for the indicated range. (See Section 6.1.5, "GBUS Memory and I/O Map (Access from the CPU Board).")

Signal name	Recommended space	Minimum range	Maximum range	Remarks
GCS0-	Memory	1M byte		• SRAM space • A bus lock can be applied by the GLOCK0- signal or by a local bus register setting.
GCS1-	Memory	2M bytes		• Flash ROM space • The program can be booted from this space instead of from UV-EPROM on the CPU board according to switch settings on the CPU board.
GCS2-	I/O	64K bytes		• Control register space on the local bus
GCS3-	Memory	64K bytes	16M bytes	• EXT-BUS memory space • When a board is connected to the 16-bit EXT-BUS (JEXT16 connector), it can be accessed by using this space.
GCS4-	I/O	64K bytes	16M bytes	• EXT-BUS I/O space
GCS5-	Memory	1M byte	2G bytes	• PCI bus memory space • A bus lock can be applied by the GLOCK1- signal or by a local bus register setting.
GCS6-	I/O	512 bytes		• PCI9080 control register space
GCS7-	I/O	64K bytes	2G bytes	• PCI bus I/O space • A bus lock can be applied by the GLOCK1- signal or by a local bus register setting. • If the CPU has no I/O space, no chip select is provided for this space, and the PCI bus I/O space can also be accessed by using the GCS5- space.

8.6. BUS CYCLE

8.6.1. Single Cycle

The following chart shows a single cycle when GBWAITI- and GBTERM- are always inactive and the CPU board is the bus master. When the motherboard is the bus master, the GCSx-, GDMAAK-, and GWAITI- signals are not used.

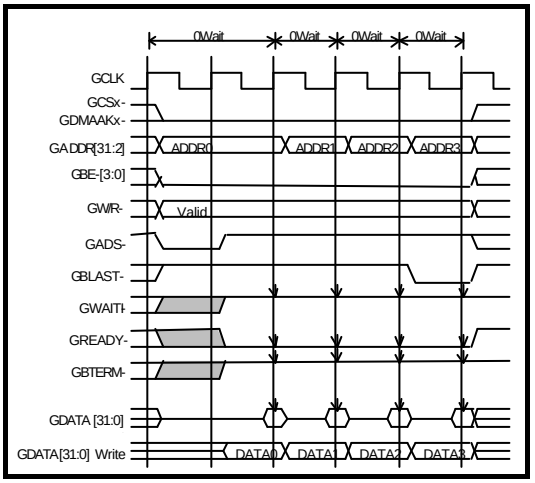


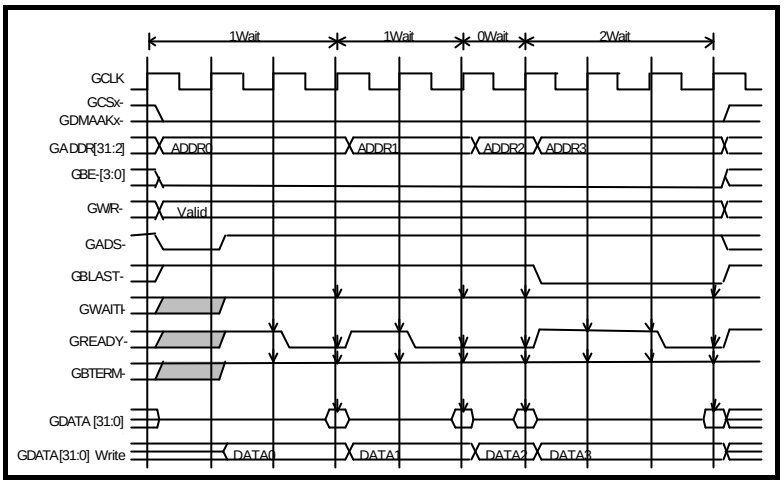
8.6.2. Burst Cycle

The following rules apply to a burst cycle.

- The addresses in a burst cycle can be in any sequence allowed by the GBUS specifications. However, the address sequence may be specified according to what is to be accessed. For the RTE-MOTHER-A motherboard, addresses must be in ascending order for an access to the PCI bus or an access to the PCI9080 control register.
- During a burst cycle, the GBE-[3:0] signals must all be active.
- The number of bursts (the number of micro cycles) is not limited. If the target of the access limits the number of bursts, use the GBTERM- signal to request canceling of the burst. (See Section 8.6.4, "GBTERM-.") The RTE-MOTHER-A motherboard has no resources that limit the number of bursts.

The following charts show a burst cycle when GBWAITI- and GBTERM- are always inactive and the CPU board is the bus master. When the motherboard is the bus master, the GCSx-, GDMAAK-, and GWAITI- signals are not used.



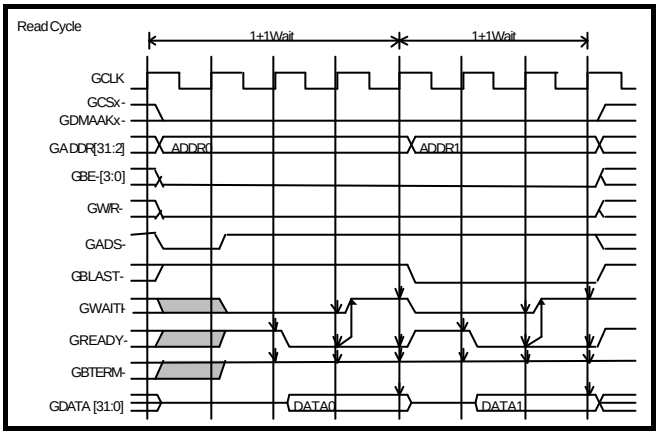


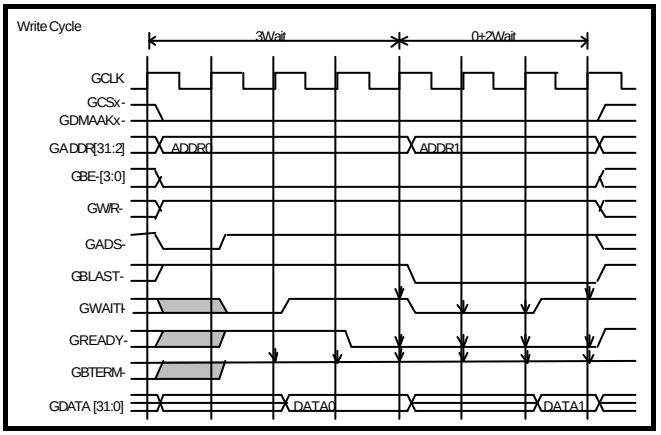
8.6.3. GWAITI-

The GWAITI- signal can be used as follows in a cycle in which the CPU board is the bus master.

- To delay sampling of data by a specific number of clocks because the data cannot be sampled in the read cycle due to a timing problem.
- To hold the target of an access by the specific number of clocks because data for the next micro cycle is not ready immediately after the completion of the first micro cycle in the burst cycle of a write cycle.

In other words, the roles of the read cycle and write cycle are switched, but GREADY- and GWAITI- serve as data transmission ready and data reception ready signals.
The following charts show that a wait cycle is inserted by the GWAITI- signal.



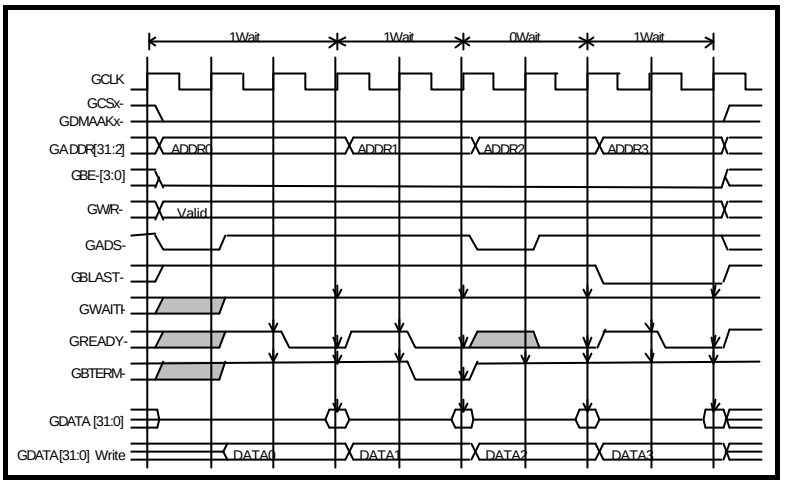


8.6.4. GBTERM-

If both the GBTERM- signal and GREADY- signal become active at the same time, the bus master completes the bus cycle after the current micro cycle ends, and then starts the burst cycle again by asserting GADS- active.

The GBTERM- signal is asserted active when the access target does not support burst cycles or when accesses are made more than the supported number of bursts. Asserting the GBTERM- signal only without asserting the GREADY- signal is not allowed.

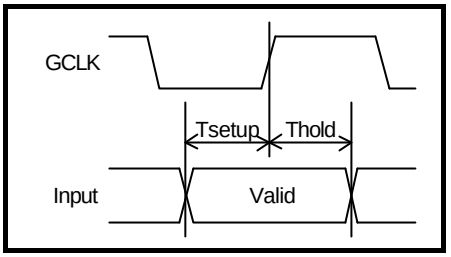
The following chart shows that the burst cycle is cancelled by the GBTERM- signal.



8.7. Timing

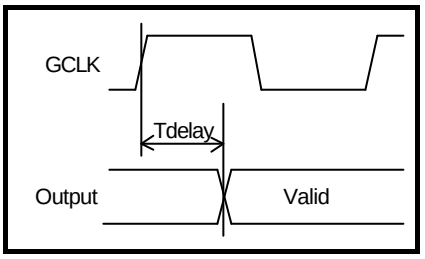
This section describes timing on the RTE-MOTHER-A. The CPU board is designed to satisfy this timing.

8.7.1. Setup Time



Signal name	Tsetup Min. (ns)	Thold Min. (ns)
GADDR[31:2]	12	0
GBEN-[3:0]	8	0
GDATA[31:0]	7	0
GADS-	14	0
GREADY-	9	1
GWAITI-	14	0
GBLAST-	8	0
GBTERM-	8	1
GW/R-	10	0
GCS-[7:0]	14	0
GBREQ-	15	0
GDMAAK-[3:0]	6	0
GLOCK-[1:0]	12	0

8.7.2. Delay Time



Signal name	Tdelay Max. (ns)
GADDR[31:2]	21
GBEN-[3:0]	17
GDATA[31:0]	21
GADS-	15
GREADY-	15
GBLAST-	17
GBTERM-	16
GW/R-	15

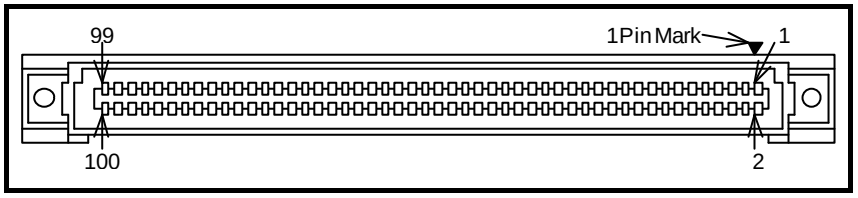
9. APPENDIX.A 32-Bit EXT-BUS Specifications

The JEXT32 connector is a 32-bit EXT-BUS connector that is provided so that memory or I/O spaces can be expanded. The local bus inside the RTE-MOTHER-A motherboard is connected to this connector.

9.1. Pin Assignments

No.	Signal name	No.	Signal name	No.	Signal name	No.	Signal name
1	GND	2	+5V	3	D0	4	D1
5	D2	6	D3	7	GND	8	D4
9	D5	10	D6	11	D7	12	GND
13	D8	14	D9	15	D10	16	D11
17	GND	18	D12	19	D13	20	D14
21	D15	22	GND	23	D16	24	D17
25	D18	26	D19	27	GND	28	D20
29	D21	30	D22	31	D23	32	GND
33	D24	34	D25	35	D26	36	D27
37	GND	38	D28	39	D29	40	D30
41	D31	42	GND	43	+5V	44	GND
45	Reserve	46	Reserve	47	(A1)	48	A2
49	A3	50	A4	51	GND	52	A5
53	A6	54	A7	55	A8	56	A9
57	A10	58	GND	59	A11	60	A12
61	A13	62	A14	63	A15	64	A16
65	GND	66	A17	67	A18	68	A19
69	A20	70	A21	71	A22	72	A23
73	GND	74	+5V	75	MRD-	76	Reserve
77	MWR0-	78	MWR1-	79	MWR2-	80	MWR3-
81	IORD-	82	IOWR-	83	GND	84	READY
85	GND	86	INT0-	87	INT1-	88	INT2-
89	INT3-	90	DMARQ0-	91	DMARQ1-	92	DMAAK0-
93	DMAAK1-	94	RESET-	95	32/16BIT-	96	N/C
97	+5V	98	GND	99	CLK	100	GND

JEXT32 Connector Pin Assignments



JEXT32 Pin Assignments

9.2. Signals

Signal name	Input/output	Function
D[0..31]	Input/output	Data bus signals. The CPU data bus signals are buffered and connected to these signals. These signals are pulled up to 10 kΩ on the board.
A[1..23]	Output	Address bus signals. The CPU address signals are buffered and connected to these signals.
MRD-	Output	Memory read cycle timing signal. This signal becomes active only for an access to the EXT-BUS space.
MWR-[0..3]	Output	Memory write cycle timing signals. MWR0-, MWR1-, MWR2-, and MWR3- correspond to D[0..7], D[8..15], D[16..23], and D[24..31], respectively. These signals become active only for an access to the EXT-BUS space.
IORD-	Output	I/O read cycle timing signal. This signal becomes active only for an access to the EXT-BUS space.
IOWR-	Output	I/O write cycle timing signal. This signal becomes active only for an access to the EXT-BUS space.
READY	Input	Signal for reporting the end of a cycle to the CPU. This signal is valid only for the EXT-BUS space. To ensure that the CPU recognizes the READY signal, the READY signal must be maintained active until MRD-, MWR-[0..3], IORD-, or IOWR- becomes inactive. This signal is pulled up to 10 kΩ on the board.
INT-[0..3]	Input	Low active interrupt request signals. These signals are connected to the interrupt controller on the board. These signals are pulled up to 10 kΩ on the board. (See Section 6.2.7.1., "Overview of Interrupt Resources.")
DMARQ-[0..1]	Input	Low active DMA request signals. These signals are buffered and then connected to the GBUS GDMARQ2- and GDMARQ3- signals. These signals are pulled up to 10 kΩ on the board.
DMAAK-[0..1]	Output	Low active DMA acknowledge signals. These signals are connected to the GBUS GDMAAK2- and GDMAAK3- signals.
RESET-	Output	Low active system reset signal.
32/16BIT-	Input	When this signal is asserted low, only D[15..0] of the data bus are used (16-bit bus mode). When this signal is asserted high, D[31..0] of the data bus are used (32-bit bus mode). This signal is pulled up to 10 kΩ on the board.
CLK	Output	Clock signal. The GBUS GCLK signal is buffered and then connected to this signal.
Reserve	—	Reserved signal. For a board that uses the EXT-BUS, this pin must not be connected to anything.

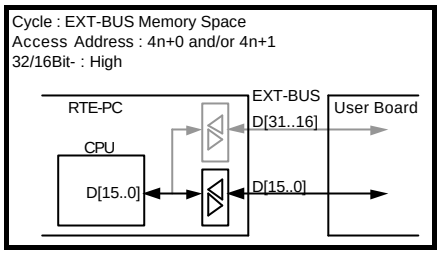
JEXT32 Connector Signals

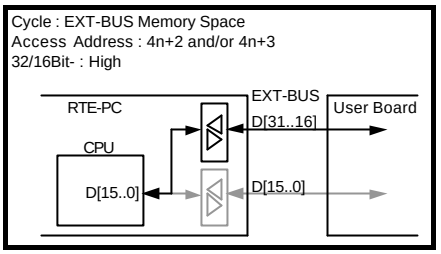
<<Cautions>>

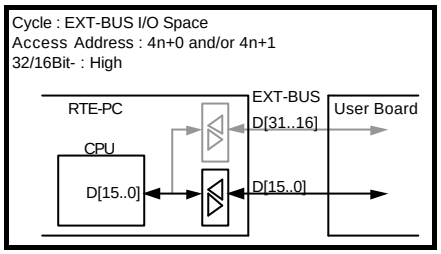
1. The 32/16BIT- signal will not necessarily be supported by all future boards in the RTE series. If you also plan to use a board that is connected to the EXT-BUS with future boards in the RTE series, you should design it so that it operates in 32-bit bus mode. When the 32/16BIT- signal is low, the MWR2- and MWR3- signals are not asserted. Instead, the MWR0- and MWR1- signals are asserted.
2. A1 is valid when the 32/16BIT- signal is low. Therefore, A1 may not be output by a future board in the RTE series for which the 32/16BIT- signal is not supported.
3. The maximum access bus width in a single cycle for the EXT-BUS depends on the bus width of the CPU's data bus. For example, for a CPU having a 16-bit data bus, up to 16 bits can be accessed for an access to I/O or an access due to a flyby DMA transfer. Therefore, for a register that is accessed by an I/O cycle or flyby DMA cycle of a board connected to the EXT-BUS, the data bus width must not exceed the data bus width of the CPU to which the board is to be connected. (The RTE-MOTHER-A motherboard has no flyby DMA feature.)

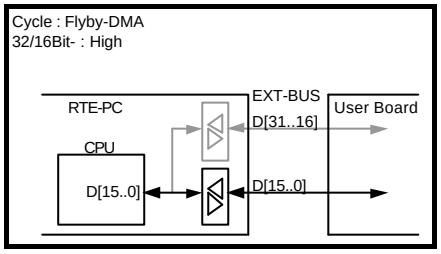
9.3. Data Bus Connection

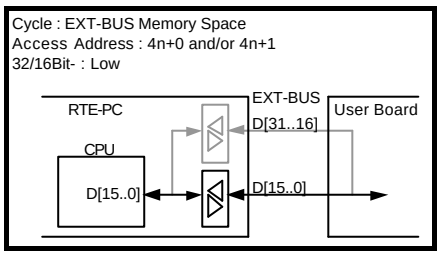
9.3.1. 16-Bit Data Bus CPU (Reference)

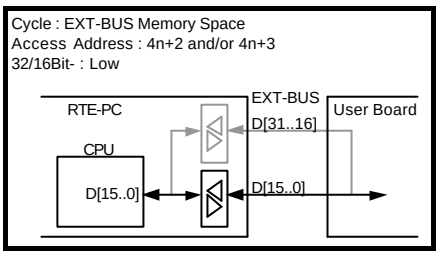


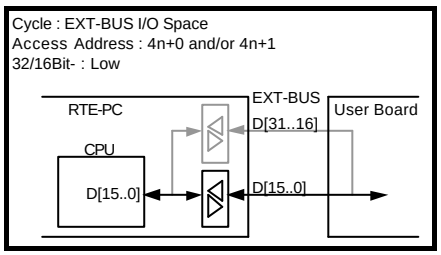


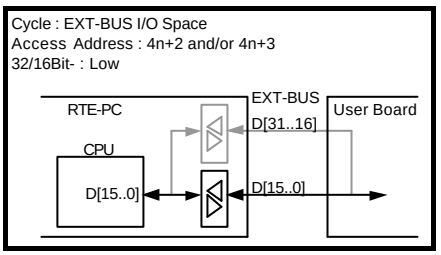


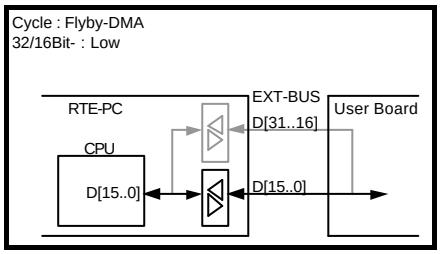




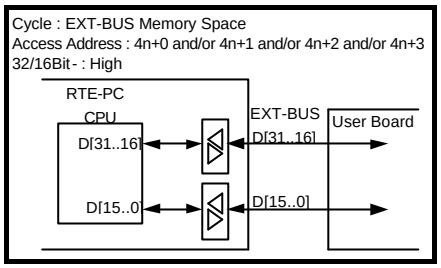


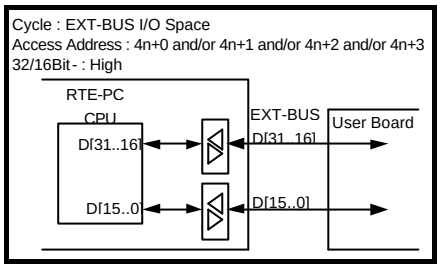


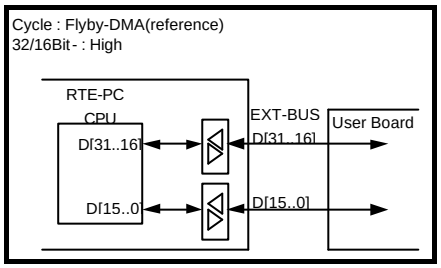


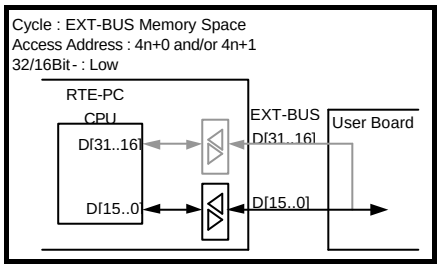


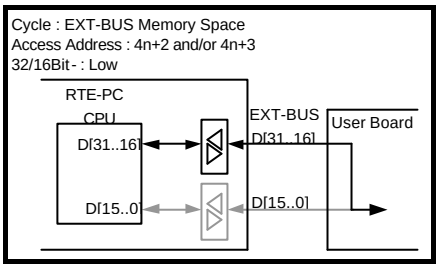
9.3.2. 32-Bit Data Bus CPU (for RTE-MOTHER-A Motherboard)

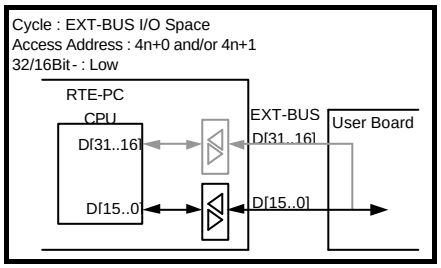


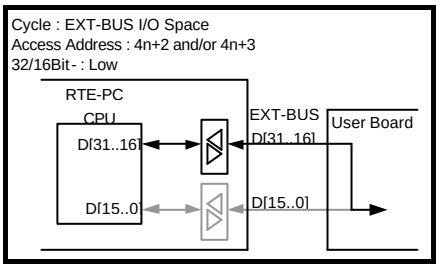


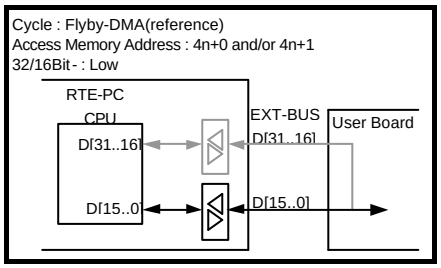


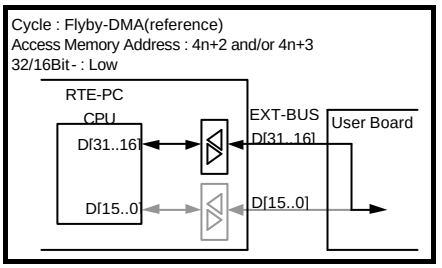




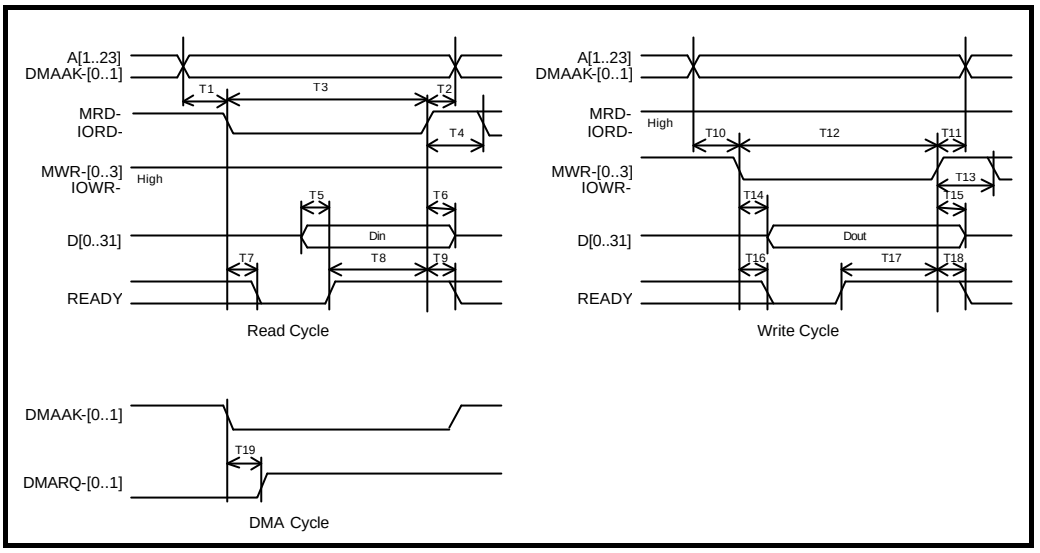








9.4. Timing



EXT-BUS Cycles

Signal	Description	MIN (ns)	MAX (ns)
T1	ADDR, DMAAK- → MRD-, IORD- setup time	10	
T2	MRD-, IORD- → ADDR, DMAAK- hold time	10	
T3	MRD-, IORD- cycle time	50	
T4	Interval between MRD- and IORD- cycles	20	
T5	RD DATA → RD READY setup time	0	
T6	MRD-, IORD- → RD DATA hold time	0	
T7	MRD-, IORD- → RD READY delay time		20
T8	RD READY → MRD-, IORD- delay time	15	
T9	MRD-, IORD- → RD READY hold time	0	
T10	ADDR, DMAAK- → MWR-, IOWR- setup time	10	
T11	MWR-, IOWR- → ADDR, DMAAK- hold time	10	
T12	MWR-, IOWR- cycle time	50	
T13	Interval between MWR- and IOWR- cycles	20	
T14	MWR-, IOWR- → WR DATA delay time		20
T15	MWR-, IOWR- → WR DATA hold time	10	
T16	MWR-, IOWR- → WR READY delay time		20
T17	WR READY → MWR-, IOWR- delay time	0	
T18	MWR-, IOWR- → WR READY hold time	0	
T19	DMAAK- → DMARQ- inactive delay time		20

EXT-BUS AC Specifications

9.5. Compatible Connectors

The connectors used on the EXT-BUS and the model numbers of compatible connectors that are certified for those connectors are shown below. When multiple boards are connected to the EXT-BUS, use cables to connect them in a daisy-chain configuration.

Connector used on EXT-BUS:	KEL 8830E-100-170S
Compatible connector (for circuit board):	KEL 8802-100-170S
Compatible connector (for cable):	KEL 8825E-100-1705
Paired cable right angle (for circuit board):	KEL 8830E-100-170L
	KEL 8831E-100-170L

9.6. Precautions

The following precautions concern the design of boards to be connected to the EXT-BUS.

1. When multiple boards are connected to the EXT-BUS, Hi-Z control must be performed so that the READY signal is driven only when a board has been selected.
2. T7 and T16 must be satisfied to insert waits in an EXT-BUS cycle.
3. When a DMA cycle is performed in single transfer mode, T19 in the timing diagram must be satisfied to ensure that the next DMA cycle does not occur. However, since this T19 greatly depends on the CPU capabilities, it may be changed in future boards of the RTE series.

10.APPENDIX.B 16-Bit EXT-BUS Specifications

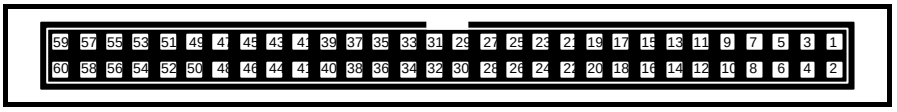
The JEXT16 connector is a 16-bit EXT-BUS connector that is provided so that memory or I/O spaces can be expanded. The local bus inside the RTE-MOTHER-A motherboard is connected to this connector.

10.1. Pin Assignments

The pin assignments of the JEXT16 connector are shown below.

No.	Signal name	No.	Signal name	No.	Signal name	No.	Signal name
1	+5V	2	+5V	31	GND	32	GND
3	D0	4	D1	33	A8	34	A9
5	D2	6	D3	35	A10	36	A11
7	D4	8	D5	37	A12	38	A13
9	D6	10	D7	39	A14	40	A15
11	GND	12	GND	41	+5V	42	+5V
13	D8	14	D9	43	A16	44	A17
15	D10	16	D11	45	A18	46	A19
17	D12	18	D13	47	BHE-	48	GND
19	D14	20	D15	49	GND*1	50	RD-
21	+5V	22	+5V	51	WR-	52	RESET-
23	A0	24	A1	53	GND	54	GND
25	A2	26	A3	55	READY	56	INT-
27	A4	28	A5	57	GND	58	GND
29	A6	30	A7	59	CPUCLK	60	GND

JEXT16 Connector Pin Assignments



JEXT16 Pin Assignments

10.2. Signals

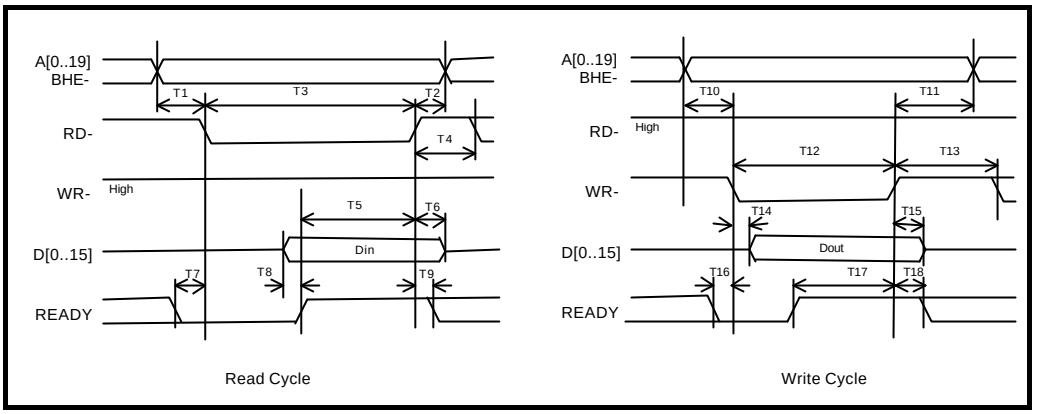
Signal name	Input/output	Function
A[0..19]	Output	Address bus signals. The CPU address signals are buffered and connected to these signals.
BHE-	Output	Byte high enable signal. The CPU UBE- signal is buffered and connected to this signal.
D[0..15]	Input/output	Data bus signals. The CPU data bus signals are buffered and connected to these signals. These signals are pulled up to 10 kΩ on the board.
RD-	Output	Read cycle timing signal. This signal becomes active only for an access to the JEXT space.
WR-	Output	Write cycle timing signal. This signal becomes active only for an access to the JEXT space.
READY	Input	Signal for reporting the end of a cycle to the CPU. This signal is valid only for the JEXT space. To ensure that the CPU recognizes the READY signal, the READY signal must be maintained active until RD- or WR- becomes inactive. This signal is pulled up to 10 kΩ on the board.
INT-	Input	Low active interrupt request signal. This signal is connected to the INT0- signal of the JEXT32 connector. This signal is pulled up to 10 kΩ on the board. (See Section 6.2.7.6, "Interrupt Status Register 1 (INT_STATUS1 GCS2:0000-6040H) [Read Only].")
RESET-	Output	Low active system reset signal.
CLK	Output	Clock signal. The GBUS GCLK pin is buffered and then connected to this signal.

JEXT16 Connector Signals

<<Cautions>>

1. Although 49Pin is essentially the GND pin, the RTE-MOTHER-A motherboard uses it to detect whether a board has been inserted in the JEXT16 connector. That is, if 49Pin is low, a board has been connected to the JEXT16 connector. To connect a board for which 49Pin is not connected to GND to the JEXT16 connector, you can forcibly connect 49Pin to GND by short-circuiting JP5. (See Section 5.6, "EXT-BUS Forced 16-Bit Jumper (JP5).")

10.3. Timing



JEXT16 Bus Cycles

Signal	Description	MIN (ns)	MAX (ns)
T1	RD address setup time	0	
T2	RD address hold time	0	
T3	RD cycle time	50	
T4	Interval between RD cycles	20	
T5	RD data setup time	15	
T6	RD data hold time	0	
T7	RD READY WAIT setup time	0	
T8	RD READY setup time	0	
T9	RD READY hold time	0	
T10	WR address setup time	0	
T11	WR address hold time	20	
T12	WR cycle time	50	
T13	Interval between WR cycles	20	
T14	WR data delay time		20
T15	WR data hold time	20	
T16	WR READY WAIT setup time	0	
T17	WR READY setup time	0	
T18	WR READY hold time	0	

JEXT16 Bus AC Specifications

- Memo -