RTE-V850E/MS1-PC

User’ s Manual (Rev. 1.01)

Midas lab

REVISION HISTORY

Date of enforcement	Revision	Chapter	Description
September 12, 1997	0.00		Preliminary edition
October 15, 1997	0.80		First edition
November 5, 1997	0.81	1.5.4	● Added the description about SIMM.
January 10, 1997	1.00	4.5.6 7	● 33MHz->40MHz.Added the detection of SIMM. ● Added the recommended settings at 40MHz.Changed the setting data.
January 16, 1997	1.01	8.2 8.3	● Changed the explanation of 32/16BIT- in the table. ● Changed the description of caution 1. ● Added the caution 4. ● Added the section.

CONTENTS

1. INTRODUCTION

This manual describes the **RTE-V850E/MS1-PC**, which is an evaluation board for the V850E/MS1, NEC's CPU. With the RTE-V850E/MS1-PC, it is possible to develop and debug programs, and evaluate the CPU performance, using the GreenHills Multi debugger. Communication with this debugger is carried out using the IBM-PC/AT ISA bus or RS-232C serial interface. It is also possible to expand memory and I/O units using local bus connectors provided on the evaluation board.

1.1. NUMERIC NOTATION

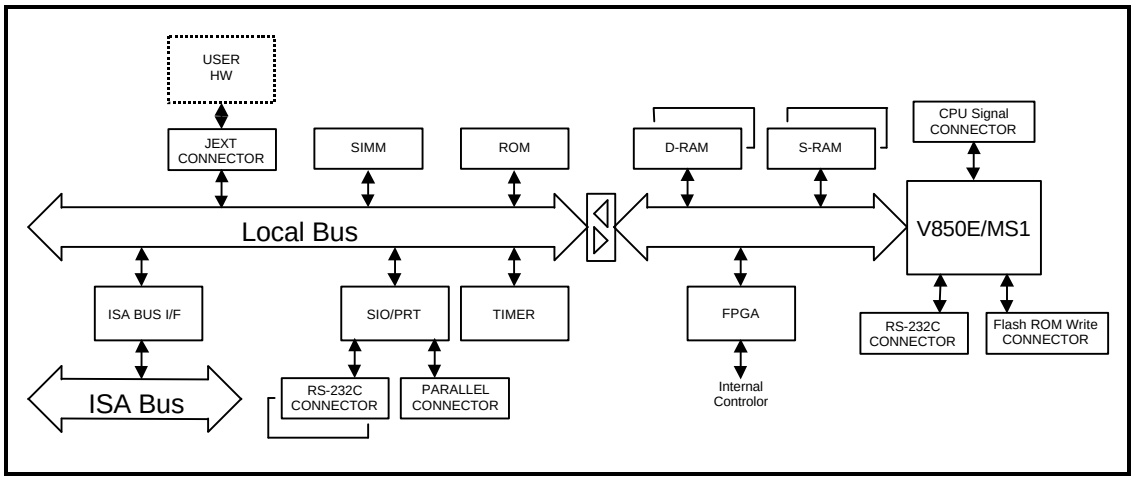
This manual represents numbers according to the notation described in the following table. Hexadecimal and binary numbers are hyphenated at every four digits, if they are difficult to read because of many digits being in each number.

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

2. FEATURES AND FUNCTIONS

The overview of each function block of the RTE-V850E/MS1-PC is shown below.



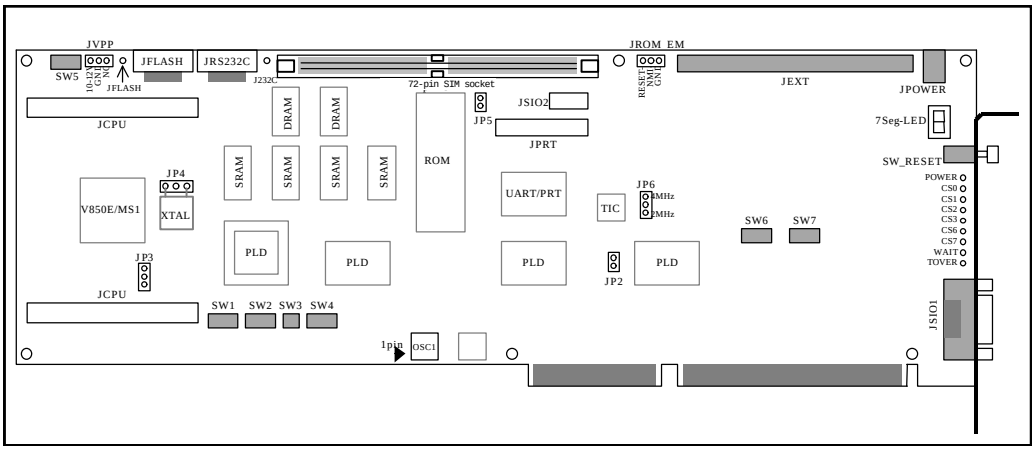
RTE-V850E/MS1-PC Block Diagram

Features

- ROM : Standard 128 Kbytes (64K × 16-bit EPROM × 1)
  Maximum 512 Kbytes (256K × 16-bit EPROM × 1)
- SRAM : 512 Kbytes (64K × 16-bit SRAM × 4)
- DRAM : 4 Mbytes (2M × 8-bit EDO-RAM × 2)
- SIMM : 4 or 8 Mbytes (SIMM is optional) installed in one 72-pin SIMM socket
  The EDO-type DRAM-SIMM can be used as well as the ordinary type DRAM-SIMM.
- RS-232C port (9-pin D-SUB connector × 1, 10-pin 2.54-mm pin header × 2)
- Parallel port (26-pin 2.54-mm pin header × 1)
- Communication function supported using the ISA bus of a PC/AT or compatible
- Local bus connector for user-installed expansion equipment
- Processor pin connector for measuring CPU signals
- External reset switch provided on the rear panel
- Connection pins for ROM in-circuit debugger
- Connector for writing to the built-in flash ROM of the CPU
- RS-232C connector using the built-in UART of the CPU

3. BOARD CONFIGURATION

The physical layout of the major components on the RTE-V850E/MS1-PC board is shown below. This chapter explains each component.



RTE-V850E/MS1-PC Board Top View

3.1. DIP SWITCHES

The RTE-V850E/MS1-PC features seven DIP switches, SW1 to SW7. The functions of these switches are explained below. The "Def." column in each of the following tables shows the factory-set positions.

3.1.1. DIP Switch 1 (SW1)

DIP SW1 is used to set up DMA transfer over EXT-BUS.

No.	Symbol	Def.	Description
1	EXDMA0*1	OFF	OFF : The INTP100/DMARQ0-/P04 and INTP110/DMAAK0-/P14 pins of the CPU can be used for purposes other than DMA transfer over EXT-BUS.
2	EXDMA0*1	OFF	ON : The INTP100/DMARQ0-/P04 and INTP110/DMAAK0-/P14 pins of the CPU are used for DMA transfer over EXT-BUS.
3	EXDMA1*2	OFF	OFF : The INTP101/DMARQ1-/P05 and INTP111/DMAAK1-/P15 pins of the CPU can be used for purposes other than DMA transfer over EXT-BUS.
4	EXDMA1*2	OFF	ON : The INTP101/DMARQ1-/P05 and INTP111/DMAAK1-/P15 pins of the CPU are used for DMA transfer over EXT-BUS.
5	PDMA	OFF	Reserved by the system. Must be set to OFF.
6	PDMA	OFF	
7	IORD-*3	OFF	OFF : The CS5-/RAS5-/IORD-/P85 pins of the CPU can be used for purposes other than DMA transfer over EXT-BUS. ON : The CS5-/RAS5-/IORD-/P85 pins of the CPU are used as IORD- for DMA transfer over EXT-BUS.
8	IOWR-*3	OFF	OFF : The CS4-/RAS4-/IOWR-/P84 pins of the CPU can be used for purposes other than DMA transfer over EXT-BUS. ON : The CS4-/RAS4-/IOWR-/P84 pins of the CPU can be used as IOWR- for DMA transfer over EXT-BUS.

*1: To perform DMA0 transfer over EXT-BUS, set switches 1 and 2 to ON.

*2: To perform DMA1 transfer over EXT-BUS, set switches 3 and 4 to ON.

*3: To perform DMA transfer over EXT-BUS, set switches 7 and 8 to ON.

3.1.2. DIP Switch 2 (SW2)

DIP SW2 is used to set interrupts.

No.	Symbol	Def.	Description
1	NMI*1	ON	OFF : The NMI/P20 pins of the CPU can be used for purposes other than the NMI from the NMI generator circuit. ON : The NMI/P20 pins of the CPU are connected to the NMI from the NMI generator circuit.
2	INTP130	OFF	OFF : The INTP130/P34 pins of the CPU can be used for purposes other than the INTP130 from the INTP130 generator circuit. ON : The INTP130/P34 pins of the CPU are connected to the INTP130 from the INTP130 generator circuit.
3	I_UART0	OFF	OFF : The INTP131/SO2/P35 pins of the CPU can be used for purposes other than the UART0 interrupt of the TL16PIR552. ON : The INTP131/SO2/P35 pins of the CPU are connected to the INTRPT0 pin of the TL16PIR552 (UART0 interrupt).
4	I_UART1	OFF	OFF The INTP132/SI2/P36 pins of the CPU can be used for purposes other than the UART1 interrupt of the TL16PIR552. ON : The INTP132/SI2/P36 pins of the CPU are connected to the INTRPT1 pin of the TL16PIR552 (UART1 interrupt).
5	I_PRT	OFF	OFF : The INTP133/SCK2-/P37 pins of the CPU can be used for purposes other than the PRINTER interrupt of the TL16PIR552. ON : The INTP133/SCK2-/P37 pins of the CPU are connected to the PINTR- pin of the TL16PIR552 (PRINTER interrupt).
6	I_TMR1	OFF	OFF : The INTP140/P114 pins of the CPU can be used for purposes other than the timer 1 interrupt of the TIC (μPD71054). ON : The INTP140/P114 pins of the CPU are connected to the TOUT1 pin of the TIC (μPD71054) (timer 1 interrupt).
7	I_ISA	OFF	Reserved by the system. Must be set to OFF.
8	Not used	OFF	

*1: To use the ROM monitor for the Multi debugger, set switch 1 to ON.

3.1.3. DIP Switch 3 (SW3)

DIP SW3 is used to set EXT-BUS interrupts.

No.	Symbol	Def.	Description
1	EXINT0	OFF	OFF : The INTP150/P124 pins of the CPU can be used for purposes other than interrupts from EXT-BUS. ON : The INTP150/P124 pins of the CPU are connected to INT0 of EXT-BUS.
2	EXINT1	OFF	OFF : The INTP151/P125 pins of the CPU can be used for purposes other than interrupts from EXT-BUS. ON : The INTP151/P125 pins of the CPU are connected to INT1 of EXT-BUS.
3	EXINT2	OFF	OFF : The INTP152/P126 pins of the CPU can be used for purposes other than interrupts from EXT-BUS. ON : The INTP152/P126 pins of the CPU are connected to INT2 of EXT-BUS.
4	EXINT3	OFF	OFF : The INTP153/P127 pins of the CPU can be used for purposes other than interrupts from EXT-BUS. ON : The INTP153/P127 pins of the CPU are connected to INT3 of EXT-BUS.

3.1.4. DIP Switch 4 (SW4)

DIP SW4 is used to set up the SIMM type and time-over ready interrupts.

No.	Symbol	Def.	Description
1	FLS_WP	OFF	Reserved by the system. Must be set to OFF.
2	FLS_VPP	OFF	Reserved by the system. Must be set to OFF.
3	TOVEN*1	ON	OFF : A time-over ready interrupt does not occur if a bus cycle does not end within a fixed period of time. OFF : A time-over ready interrupt occurs if a bus cycle does not end within a fixed period of time to make the bus cycle end.
4	SIMMEDO*2	OFF	OFF : Set the contact to OFF when the SIMM is not of EDO type. ON : Set the contact to ON when the SIMM is of EDO type.
5	NMI/130	OFF	Reserved by the system. Must be set to OFF.
6	Not used	OFF	
7	Not used	OFF	
8	Not used	OFF	

*1: For an explanation of time-over ready interrupts, see Section 12.1.

*2: Setting this switch does not cause any switching in the hardware. The settings of this switch are read by the ROM monitor for the Multi debugger and used to initialize the DRAM controller inside the CPU.

3.1.5. DIP Switch 5 (SW5)

DIP SW5 is used to set the states of CPU pins.

No.	Symbol	Def.	Description
1	MODE0	ON	OFF : Sets the MODE0 pin of the CPU to High. ON : Sets the MODE0 pin of the CPU to Low.
2	MODE1	ON	OFF : Sets the MODE1 pin of the CPU to High. ON : Sets the MODE1 pin of the CPU to Low.
3	MODE2	ON	OFF : Sets the MODE2 pin of the CPU to High. ON Sets the MODE2 pin of the CPU to Low.
4	CKSEL	ON	OFF : Sets the CKSEL pin of the CPU to High (direct mode). ON : Sets the CKSEL pin of the CPU to Low (PLL mode).
5	M3_NML	ON	Reserved by the system. Must be set to ON.
6	M3_HI	OFF	Reserved by the system. Must be set to OFF.
7	J232C	OFF	J232C and JFLASH are used in combination, as follows: [J232C, JFLASH] [OFF, OFF] : Pins P22 to P27 of the CPU can be used for purposes other than those described below. [ON, OFF] : Pins P22 to P27 of the CPU are connected to the JRS232C connector, thus enabling the JRS232C connector. (The J232C-LED lights.) [X, ON] : The system enters the mode in which writing to the built-in flash ROM of the CPU is possible using a writer, thus enabling the JFLASH connector. (The JFLASH-LED lights.)
8	JFLASH	OFF	

3.1.6. DIP Switch 6 (SW6)

DIP SW6 is used to select the I/O address of the ISA bus. Switch contacts 1 to 8 correspond to A4 to A11 of the ISA bus address. (A12 to A15 are fixed to 0.) Therefore, this switch can be used to select an I/O address in the range of 000xH to 0FFxH. When a switch contact is OFF, it represents 1. When it is ON, it represents 0. The default is 0200H.

No.	Symbol	Def.	Description
1	A4	ON	Select the I/O address of the ISA bus. (OFF=1, ON=0)
2	A5	ON	
3	A6	ON	
4	A7	ON	
5	A8	ON	
6	A9	OFF	
7	A10	ON	
8	A11	ON	

3.1.7. DIP Switch 7 (SW7)

The settings of this switch can be read by the CPU via a general-purpose port. When the ROM monitor for the Multi debugger is used, some of the bits are reserved. See Sections 4.3.7 and 6.5 for details.

3.2. JUMPER SWITCHES

The RTE-V850E/MS1-PC features four jumper switches: JP2, JP3, JP5, and JP6. The functions of these jumper switches are explained below.

3.2.1. JP2

JP2 is reserved by the system. Leave it open.

3.2.2. AVDD Power Switching Jumper Switch (JP3)

JP3 is used to select the power to be supplied to the AVDD pin of the CPU. (See Section 10.9.)

JP3 setting	Description
Jumper pins 1 and 2.	The same power (+5V) as that supplied to the HVDD pin of the CPU is supplied to the AVDD pin of the CPU. (Factory setting)
Jumper pins 2 and 3.	The AVDD signal from the JCPU connector is supplied to the AVDD pin of the CPU.

The AVSS pin of the CPU is connected to GND via resistor R1 (0 W) located at the rear of JP3. To isolate the AVSS pin from GND and supply AVSS from the JCPU connector, remove resistor R1.

3.2.3. ROM Size Switching Jumper Switch (JP5)

JP5 must be set according to the size of the mounted ROM.

JP5 setting	Description
Leave pins 1 and 2 open.	Leave the pins open when 128 Kbytes (64K × 16 bits) or 256 Kbytes (128K × 16 bits) of ROM is mounted. (Factory setting)
Jumper pins 1 and 2.	Jumper the pins when 512 Kbytes (256K × 16 bits) of ROM is mounted.

3.2.4. TIC Clock Switching Jumper Switch (JP6)

JP6 is used to select the clock pulses to be supplied to counters #1 and #2 of the TIC (μPD71054). (See Section 6.3.)

JP6 setting	Description
Jumper pins 1 and 2.	2-MHz clock pulses are supplied to counters #1 and #2 of the TIC. (Factory setting)
Jumper pins 2 and 3.	4-MHz clock pulses are supplied to counters #1 and #2 of the TIC.

3.3. SWITCH

The RTE-V850E/MS1-PC contains the SW_RESET switch.

3.3.1. Reset Switch (SW_RESET)

SW_RESET is a reset switch. Pressing this switch resets both the CPU and the board. (See Section 10.2.)

3.4. LEDs

The RTE-V850E/MS1-PC contains LEDs to indicate its status. The functions of the LEDs are explained below.

LED name	Description
POWER	Lights when power is supplied to the board.
CS0	Lights when the CS0- pin of the CPU goes Low.
CS1	Lights when the CS1- pin of the CPU goes Low.
CS2	Lights when the CS2- pin of the CPU goes Low.
CS3	Lights when the CS3- pin of the CPU goes Low.
CS4	Lights when the CS4- pin of the CPU goes Low.
CS5	Lights when the CS5- pin of the CPU goes Low.
CS6	Lights when the CS6- pin of the CPU goes Low.
CS7	Lights when the CS7- pin of the CPU goes Low.
WAIT	Lights when the WAIT- pin of the CPU goes Low.
TOVER	Lights when a time-over ready interrupt occurs.
JFLASH	Lights when the JFLASH connector is enabled.
J232C	Lights when the JRS232C connector is enabled.
7Seg-LED	Can be made to light depending on the contents of the output of a general-purpose port. (See Section 6.4.)

3.5. CONNECTORS AND SOCKETS

The RTE-V850E/MS1-PC features several connectors and sockets. These are explained below.

3.5.1. Power Supply Connector (JPOWER)

When this board is to be used as a standalone, that is, without being inserted in an ISA bus slot, the board should be supplied with power from an external power supply by connecting it to the JPOWER connector.

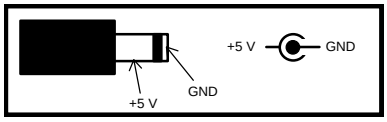
The external power should be one rated as listed below.

Voltage: 5 V

Current: Maximum of 2 A (excluding the current supplied to the JEXT connector)

Mating connector: Type A (5.5 mm in diameter)

Polarity:



When attaching an external power supply to the board, pay careful attention to its connector polarity. When inserting the board into the ISA bus slot, do not attach the JPOWER connector to an external power supply.

3.5.2. Crystal Socket (JP4)

JP4 has two roles. Namely, it is used to select the clock supplied to the CPU and also acts as the connector for the crystal oscillator. (See Section 10.7.)

JP4 setting	Description
Mount a crystal oscillator between pins X1 and X2.	The mounted crystal oscillator is connected to pins X1 and X2 of the CPU.
Jumper pin X1 and the center pin of JP4.	The oscillator mounted on the OSC1 socket is connected to pin X1 of the CPU.

3.5.3. Oscillator Socket (OSC1)

The OSC1 socket is used to mount the oscillator used to supply clock pulses to the CPU.

The crystal oscillator mounted on JP4 can also be used to supply clock pulses to the CPU, by using JP4. (See Section 10.7.)

On the OSC1 socket, a 5V oscillator of 8-pin DIP type (half type) must be mounted.

When mounting an oscillator, pay careful attention to the orientation of pin 1.
If the leads of the oscillator are short, the frame (outer coating) of the oscillator may be short-circuited with the socket pins.
5V power is supplied to the oscillator.

3.5.4. DRAM-SIMM Sockets

The DRAM-SIMM socket can hold a 72-pin SIMM (known as a module in the case of PC/AT machines. it's DRAM has Row-Address and Column-Address of same bit width, and It's has 4 separate RAS-signal lines.) with a capacity of 4 or 8 Mbytes, and can also accept an EDO type DRAM-SIMM in addition to normal DRAM-SIMMs. The capacity of the connected SIMM can be read through the PIO port. (See Section 6.6.)

A 16-/32-Mbyte SIMM cannot be used.

3.5.5. ROM Sockets

The RTE-V850E/MS1-PC has ROM sockets to hold 40-pin ROM chips to provide standard 128 Kbytes (64K × 16 bits). When the standard ROM is to be replaced to enable the use of the Multi debugger, it must be replaced by a ROM having an access time of less than 120 ns. When mounting ROM of a different capacity, it may be necessary to switch JP5 on the board. (See Section 3.2.3.)

3.5.6. Self-Writing Power Supply Connector (JVPP)

To enable self-writing to the built-in flash ROM of the CPU, 7.5V power is required. The JVPP connector is used to supply the power for generating the 7.5V power.

By supplying **10 to 12V** power to this connector and setting pin P21 of the CPU to Low, 7.5V power is supplied to the VPP pin of the CPU. (See Section 10.6.)

The pins of the JVPP connector are listed in the table below.

JVPP pin No.	Name	Input/ output	Description
1	10-12V	Input	To be connected to 10 to 12V power supply.
2	GND	Input	To be connected to GND of the power supply.
3	NC		Not to be connected.

3.5.7. Connector for ROM Emulation (JROM_EM)

JROM_EM are the test pins used to connect a ROM in-circuit type debugger. These test pins accept control signals from the ROM in-circuit debugger. The signal names and functions are listed in the table below.

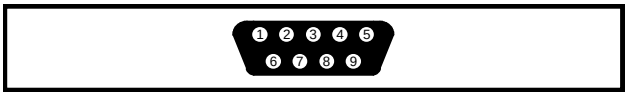
JROM_EM pin No.	Signal name	Input/output	Function
1	RESET-	Input	Connects the reset request signal from the ROM in-circuit debugger. The CPU is reset when a Low level signal is input. The input is pulled up by a 1-kΩ resistor on the board. (See Section 10.2.)
2	NMI-	Input	Connects the NMI request signal (break request) from the ROM in-circuit debugger. The NMI is input to the CPU when a Low level signal is input. The input is pulled up by a 1-kΩ resistor on the board. (See Section 10.3.)
3	GND	---	Ground pin to ground the ROM in-circuit debugger.

3.5.8. Serial Connectors (JSIO1, JSIO2)

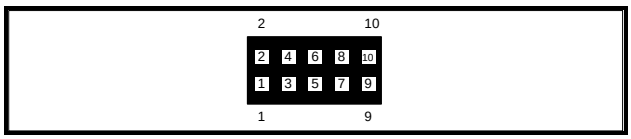
The JSIO1 and JSIO2 connectors are used for the RS-232C interface controlled by the serial controller (TL16PIR552). Regarding the connector shapes, JSIO1 is a 9-pin D-SUB RS-232C connector like that normally provided on the PC/AT, while JSIO2 is a pin header type connector with a pitch of 2.54 mm. All signals at both of these connectors are converted to RS-232C level. The pin arrangements of these connectors are shown below, after which the signal assignments are listed.

For the connection signals when the connectors are connected to the host, the table lists the wiring for both the D-SUB 9 pins and D-SUB 25 pins on the host side. (Regular cross-cable wiring is used for these connections.)

The pin arrangement of JSIO2 is identical to that of JSIO1 when a push-fit connector with a ribbon cable is connected to JSIO2.



JSIO1 Pin Arrangement



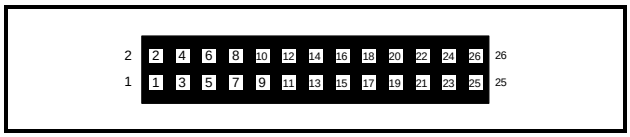
JSIO2 Pin Arrangement

JSIO1 pin No.	JSIO2 pin No.	Signal name	Input/output	Connector pin No. on the host side	
				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	NC			

3.5.9. Parallel Connector (JPRT)

The JPRT connector is used for the parallel interface controlled by the parallel (printer) controller (TL16PIR552). It is a pin header type connector with a pitch of 2.54 mm. All signals at this connector are at the RS-232C level. Its pin arrangement and signal assignment are shown and listed below.

The pin arrangement of JPRT is identical to the 25-pin D-SUB connector like that normally provided on the PC/AT when a push-fit connector with a ribbon cable is used.



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

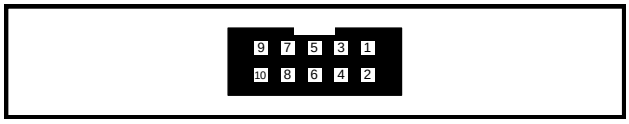
3.5.10. Serial Connector (JRS232C)

The JRS232C connector is an RS-232C connector controlled by the built-in UART of the CPU. (See Section 10.8.) This connector is a pin header type connector with a pitch of 2.54 mm. When a push-fit connector with a ribbon cable is connected to JRS232C, the pin arrangement of JRS232C is identical to that of a 9-pin D-SUB RS-232C connector like that normally provided on a PC/AT. All signals are converted to the RS-232C level.

To enable the use of JRS232C, SW5 must be set. (See Section 3.1.5.)

When JRS232C is enabled for use, the J232C-LED lights. (See Section 3.4.)

The pin arrangement of the JRS232C connector is shown below, after which the signal assignment is listed. For an explanation of the connections when the connector is connected to a personal computer (host), see the table given in Section 3.5.8.



JRS232C Pin Arrangement

JRS232C pin No.	Signal name	Input/output	Corresponding CPU pin
1	NC	Input	
3	RxD(RD)	Input	P23
5	TxD(SD)	Output	P22
7	DTR(DR)	Output	P27
9	GND		
2	DSR(ER)	Input	P24
4	RTS(RS)	Output	P25
6	CTS(CS)	Input	P26
8	NC		
10	NC		

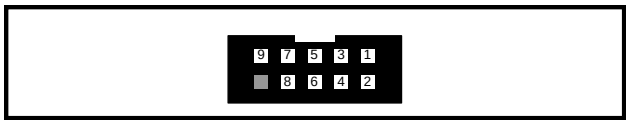
3.5.11. Flash Writing Connector (JFLASH)

The JFLASH connector is used to connect a writer when writing is to be performed to the built-in flash ROM of the CPU using that writer. **The NEC FP-100 flash writer can be connected directly to this connector.**

To enable the use of JFLASH, SW5 must be set. (See Section 3.1.5.)

When JFLASH is enabled, the JFLASH-LED lights. (See Section 3.4.)

For details of writing to flash ROM, see Chapter 11.



JFLASH Pin Arrangement

JFLASH pin No.	Signal name	Input/ output	Remarks
1	SO0	Output	Synchronous serial data output (CMOS level)
2	SI0	Input	Synchronous serial data input (CMOS level)
3	SCK0-	Input	Synchronous serial clock input (CMOS level)
4	RESET-	Input	Reset input
5	VPP	Input	VPP input
6	+3.3V	Output	CPU core power level output
7	+5V	Output	CPU-I/O power level output
8	GND	-	Ground
9	NC	-	Not connected
(10)	NC	-	Not connected (no pin)

3.5.12. Processor Pin Connector (JCPU)

The pins of the CPU are connected to the JCPU connector. For an explanation of the connections on the board, see Chapter 10.

JCPU pin No.	Signal name	JCPU pin No.	Signal name
1	GND	2	INTP142/SI3/P116
3	INTP141/SO3/P115	4	INTP140/P114
5	TI14/P113	6	TCLR14/P112
7	TO141/P111	8	TO140/P110
9	+3.3V	10	Not connected
11	+3.3V	12	JX1*1
13	GND	14	CKSEL
15	MODE0	16	MODE1
17	MODE2	18	MODE3/VPP
19	RESET-	20	INTP153/ADTRG/P127
21	+5V	22	INTP152/P126
23	INTP151/P125	24	INTP150/P124
25	TI15/P123	26	TCLR15/P122
27	TO151/P121	28	TO150/P120
29	CLKOUT/PX7	30	WAIT-/PX6
31	GND	32	REFRQ-/PX5
33	JCPU_WAIT-*2	34	HLDRQ-/P97
35	HLDAK-/P96	36	OE-/P95
37	ADV-/BCYST-/P94	38	WE-/P93
39	RD-/P92	40	UCAS-/UWR-/P91
41	+3.3V	42	LCAS-/LWR-/P90
43	CS7-/RAS7-/P87	44	CS6-/RAS6-/P86
45	CS5-/RAS5-/IORD-/P85	46	CS4-/RAS4-/IOWR-/P84
47	CS3-/RAS3-/P83	48	CS2-/RAS2-/P82
49	CS1-/RAS1-/P81	50	CS0-/RAS0-/P80
51	+5V	52	+5V
53	A23/P67	54	A22/P66
55	A21/P65	56	A20/P64
57	A19/P63	58	A18/P62
59	A17/P61	60	A16/P60
61	GND	62	A15/PB7
63	A14/PB6	64	A13/PB5
65	A12/PB4	66	A11/PB3
67	A10/PB2	68	A9/PB1
69	A8/PB0	70	GND
71	+3.3V	72	A7/PA7
73	A6/PA6	74	A5/PA5
75	A4/PA4	76	A3/PA3
77	A2/PA2	78	A1/PA1
79	A0/PA0	80	+5V

*1: The output of the oscillator mounted on the OSC1 socket is converted to the 3.3V level and output to this pin. (See Section 10.7.)

*2: WAIT- input from the JCPU connector. This is used to insert waits into a bus cycle via the JCPU connector.

JCPU pin No.	Signal name	JCPU pin No.	Signal name
81	GND	82	D15/P57
83	D14/P56	84	D13/P55
85	D12/P54	86	D11/P53
87	D10/P52	88	D9/P51
89	D8/P50	90	GND
91	+3.3V	92	D7/P47
93	D6/P46	94	D5/P45
95	D4/P44	96	D3/P43
97	D2/P42	98	D1/P41
99	D0/P40	100	+3.3V
101	+5V	102	INTP103/DMARQ3-/P07
103	INTP102/DMARQ2-/P06	104	INTP101/DMARQ1-/P05
105	INTP100/DMARQ0-/P04	106	TI10/P03
107	TCLR10/P02	108	TO101/P01
109	TO100/P00	110	GND
111	GND	112	INTP113/DMAAK3-/P17
113	INTP112/DMAAK2-/P16	114	INTP111/DMAAK1-/P15
115	INTP110/DMAAK0-/P14	116	TI11/P13
117	TCLR11/P12	118	TO111/P11
119	TO110/P10	120	INTP123/TC3-/P107
121	+3.3V	122	INTP122/TC2-/P106
123	INTP121/TC1-/P105	124	INTP120/TC0-/P104
125	TI12/P103	126	TCLR12/P102
127	TO121/P101	128	TO120/P100
129	ANI7/P77	130	ANI6/P76
131	+5V	132	ANI5/P75
133	ANI4/P74	134	ANI3/P73
135	ANI2/P72	136	ANI1/P71
137	ANI0/P70	138	AVDD*3
139	AVSS*3	140	AVREF
141	GND	142	NMI/P20
143	P21	144	TXD0/SO0/P22
145	RXD0/SI0/P23	146	SCK0-/P24
147	TXD1/SO1/P25	148	RXD1/SI1/P126
149	SCK1-/P27	150	+3.3V
151	+3.3V	152	INTP133/SCK2-/P37
153	INTP132/SI2/P36	154	INTP131/SO2/P35
155	INTP130/P34	156	TI13/P33
157	TCLR13/P32	158	TO131/P31
159	TO130/P30	160	INTP143/SCK3-/P117

*3: For an explanation of how to connect AVDD and AVSS to the CPU, see Section 10.9.

3.6. EXTENSION BUS CONNECTOR (JEXT)

The JEXT connector is provided to enable the extension of memory or I/O. This connector is internally connected to the local bus of the Base board. For details of this connector, see Chapter 8.

4. USING THE BOARD WITH THE MULTI DEBUGGER

This chapter explains how to set up the RTE-V850E/MS1-PC board when it is being used in combination with the Multi debugger of Green Hills Software, as the well as the cautions to be observed.

4.1. SOFTWARE

To use the RTE-V850E/MS1-PC with the Multi debugger, two software products must be installed, in addition to the Multi debugger.

One is the Midas server (rteserv). This software product controls RTE for WIN32 in accordance with the instructions received from the Multi debugger.

The other is RTE for WIN32. This software product is used for communication with the board.

For an explanation of how to install each software product, see the installation manual supplied with the product.

4.2. METHODS OF USE

The RTE-V850E/MS1-PC can be used in combination with the Multi debugger in either of two ways. In one case, the board is inserted into the ISA bus slot of a PC/AT or compatible, so that communication is performed via the ISA bus.

In the other case, the board is not inserted into the ISA bus, but is supplied with +5V power from JPOWER and connected to the host with an RS-232C cable, so that communication is performed via the RS-232C cable.

4.3. SETTING UP THE BOARD

The following explains how to set the switches on the board to enable use of the board with the Multi debugger.

4.3.1. DIP Switch 1 (SW1)

Usually, this switch need not be changed from the default settings. (See Section 3.1.1.)

4.3.2. DIP Switch 2 (SW2)

Set switch 1 of SW2 to the ON position to connect the NMI from the NMI generator circuit to CPU pins. The other switches usually need not be changed from their default positions. (See Section 3.1.2.)

4.3.3. DIP Switch 3 (SW3)

Usually, this switch need not be changed from the default settings. (See Section 3.1.3.)

4.3.4. DIP Switch 4 (SW4)

When using a SIMM, set contact 4 of SW4 according to whether the SIMM is of EDO or FastPage type. (If the user program is designed to set the SIMM-related SFRs, contact 4 of SW4 need not be set.) The other switches usually need not be changed from their default positions. (See Section 3.1.4.)

4.3.5. DIP Switch 5 (SW5)

Contacts 1 to 3, 5, and 6 of SW5 need not be changed from the default positions. The other switches usually need not be changed from their default positions, either. (See Section 3.1.5.)

4.3.6. DIP Switch 6 (SW6)

When inserting this board into an ISA bus slot, use this switch to set the I/O address on the host that is to be used by this board. The default I/O address is 020xH (0200H to 020FH). When setting the I/O address, make sure that the address is not the same as that of the mother board or any other board. (See Section 3.1.6.)

When not inserting this board into the ISA bus slot, this switch need not be changed from its default positions.

4.3.7. DIP Switch 7 (SW7)

DIP switch 7 is connected to the general-purpose input ports. For the ROM monitor for the Multi debugger, this switch is used to set the baud rate at which the board will communicate with the Multi debugger via the RS-232C interface and to set the profiler period. Contacts 5 to 8 of SW7 must be set to OFF.

SW7		Baud rate
1	2	
ON	ON	Not used
OFF	ON	38400 baud
ON	OFF	19200 baud
OFF	OFF	9600 baud (Factory setting)

Setting the Baud Rate

SW7		Profiler period
3	4	
ON	ON	The timer is not used.
OFF	ON	200 Hz 5 ms
ON	OFF	100 Hz 10 ms
OFF	OFF	60 Hz 16.67 ms (Factory setting)

Setting the Profiler Period

4.3.8. JP2

Leave JP2 open.

4.3.9. AVDD Power Switching Jumper Switch (JP3)

Usually, this switch need not be changed from the default setting. (See Section 3.2.2.)

4.3.10. ROM Capacity Switching Jumper Switch (JP5)

Set JP5 according to the size of the ROM containing the monitor for the Multi debugger. (See Section 3.2.3.)

4.3.11. TIC Clock Switching Jumper Switch (JP6)

Usually, this switch need not be changed from the default setting. (See Section 3.2.4.)

4.4. PROCEDURES FOR USE

The procedures for using the RTE-V850E/MS1-PC are explained below.

4.4.1. Procedure Involving the Insertion of the Board into ISA Bus Slot

When the board is inserted into an ISA bus slot in the PC, power (+5V) is supplied from the ISA bus to the board. In addition, the ISA bus can be used for communication with the debugger, so that programs are downloaded at high speed.

The board can be installed in the ISA bus slot according to the following procedure.

<1> Set the I/O address of the PC that is to be used by the RTE-V850E/MS1-PC, using the appropriate switch on the board. Make sure that the I/O address is not the same as that of any other board. For an explanation of how to set this switch, see Section 3.1.6.

<2> Turn off the power to the PC, open its housing, and select an ISA bus slot into which the board is to be inserted. If the slot is fitted with a rear panel, remove the panel.

<3> Insert the board into the ISA bus slot. Make sure that the board does not touch any adjacent board. Fasten the rear panel, attached to the board, to the housing of the PC, using a screw.

<4> Turn on the power to the PC, and check that the POWER-LED on the board lights. **If the LED does not light, turn off the PC power immediately, and check the installation.** If the system does not start normally (for example, if an error occurs during the installation of a device driver), it is likely that the set I/O address is the same as that of another board. Reconfirm the I/O address of the board by referring to the manuals supplied with the PC and other boards.

<5> When the system is found to be normal, turn off the PC power again, and replace its housing.

<6> Turn on the PC power, and check the installation using the Check RTE utility supplied with RTE for WIN32. (See the installation manual for RTE for WIN32.)

4.4.2. Procedure for Using the Board as a Standalone

When the board is used as a standalone rather than being installed in the PC, it requires an external power supply. In addition, communication with the debugger is supported only by the RS-232C interface. This configuration is useful when the Multi debugger is used on a host other than a PC/AT or compatible, as well as when the board is used for hardware confirmation and expansion.

The board can be used as a standalone according to the following procedure.

<1> Obtain an RS-232C cable (for connection with the host), an external power supply (+5V, 2A). In the case of the power supply, pay careful attention to its voltage and **connector polarity**. In addition, attach spacers to the four corners of the board, so that it will not pose any problem, regardless of where it is installed. See Sections 3.5.8 and 3.5.1 for an explanation of RS-232C cable connection and the power supply connector, respectively.

<2> Using the corresponding DIP switch on the board, set the RS-232C baud rate. For an explanation of how to set this switch, see Section 4.3.7.

<3> Connect the board to the host using the RS-232C cable. Also connect the external power supply to the JPOWER connector, then check that the POWER-LED on the board lights. **If the LED does not light, turn off the power immediately, then check the connection.**

<4> Check the connection using the Check RTE utility supplied with RTE for WIN32. (See the installation manual provided with RTE for WIN32.)

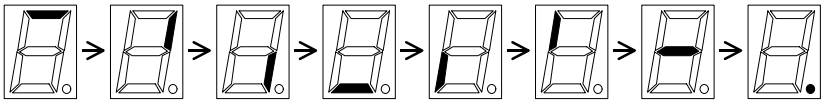
4.5. ROM MONITOR FOR THE MULTI DEBUGGER

To enable the use of the RTE-V850E/MS1-PC in combination with the Multi debugger, ROM that contains the ROM monitor for the Multi debugger (RTEMON) must be mounted on the board.
Cautions on the use of this ROM monitor are described below.

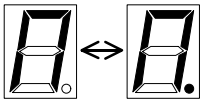
4.5.1. 7-Segment LED at Startup

When the ROM monitor for the Multi debugger is mounted, the 7-segment LED behaves as shown below when the power to the board is turned on. (The black portion is lit.)

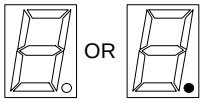
1) Checking of the 7-segment LED (see the figure below)



2) Counting of numbers due to a simple memory check on SRAM
3) rteserv connection wait state. (The dot does not blink if the profiler timer is stopped.)



4) rteserv connected state. (The dot is either on or off depending on whether it was on or off when rteserv was connected.)



4.5.2. ROM Monitor Work RAM

The ROM monitor uses the high-order 32 Kbytes (3E7-8000H to 3E7-FFFFH) of SRAM as its work RAM. User programs cannot, therefore, use this area and the image area of this area.

4.5.3. _INIT_SP Setting

_INIT_SP (stack pointer initial value) is set to 3E7-7FFCH (immediately before the monitor work RAM) by the monitor. (_INIT_SP can be changed in the Multi environment.)

4.5.4. Remote Connection

Either serial or ISA bus connection can be selected for operation with the Multi server (rteserv). To switch from serial connection to ISA bus connection or vice versa, it is necessary to reset the monitor (by pressing the reset switch on the rear panel) and run the Check RTE utility of RTE for WIN32.

4.5.5. Timer Interrupt

The profiler function of the Multi debugger cannot be used if timer interrupt is inhibited (see Section 4.3.7 for details of timer interrupt setting).

4.5.6. Hardware Initialization

Regarding the ROM, SRAM, and EDO-DRAM spaces, initializes the CPU (SFRs) internally as well as the board so that they are compatible with 40-MHz operation(See Section 7).

Regarding the SIMM space, only when a SIMM is mounted, the ROM monitor initializes the CPU (SFRs) internally so that it conforms to 40-MHz operation if a SIMM with an access time of 60 ns is mounted. Set the appropriate DIP switch according to whether the SIMM is of EDO or FastPage type. (See Section 3.1.4) At the time of a reset, the ROM monitor for the Multi debugger occurs 4 bytes writing and reading at the SIMM-Base-Address to detect a SIMM(Original data is write back).

4.6. APPLICATIONS USING MASKABLE INTERRUPTS

The following sections explain how to develop applications using maskable interrupts on the RTE-V850E/MS1-PC, as well as restrictions.

4.6.1. Interrupt Vectors

The interrupt vector area of the V850E/MS1, addresses 0000H to 07FFH, are fixed by the use of ROM, and cannot be changed. The ROM monitor for the Multi debugger provides an alternate vector area in SRAM, so that branch instructions for branching to the alternate vector area can be placed in the vector area of addresses 0000H to 07FFH.

If, for example, an interrupt with an exception code of 0080H is generated, the interrupt function of the CPU causes a branch to address 0080H, where there is a branch instruction for branching to offset address 0080H in the alternate vector area. Thus, by changing this alternate vector area in the same way as the original vector area, the user program can branch to an interrupt handling routine when an interrupt is generated.

A difference between an ordinary V850E/MS1 program and a program using Multi on the RTE-V850E/MS1-PC is that, for the former, the vector area is fixed at the time it is turned into ROM and the program need not establish (change) the area, while the latter must change a vector before enabling an interrupt.

The alternate vector area is located at addresses 3E7-8000H to 3E7-87FFH in SRAM. (Actually, reference/modification is possible from other addresses because an SRAM image is available.) For the above-mentioned interrupt with an exception code of 0080H, the instruction for branching to the desired interrupt handling routine must be written at address 3E7-8080H.

Actually, the interrupt vectors located at addresses 000-0000H to 000-07FFH and first referenced by the CPU, branch to FFE7-8000H to FFE7-87FFH. The CPU, however, masks bits 26 to 31 of the PC register to "0," so the vectors eventually branch to 3E7-8000H to 3E7-87FFH.

If the CPU has a built-in cache memory, a cache flush operation is required after a vector is changed. For the V850E/MS1, however, this is not required because it has no built-in cache memory.

A sample program for changing alternate vectors is shown on the next page. (This sample assumes

that the address of the interrupt handling routine relative to the alternate vector area is 22 bits or less.)

```
void SetAJump(int addr, int jmpdest)  /* ←Vector setting routine */
/* int addr;             address where we're storing the 'jr' */
/* int jmpdest;          address where the 'jr' jumps to */
{
    int offset;
    unsigned inst;
    unsigned int *p ;

     offset = jmpdest - addr;
     inst = 0x07800000 /* 'jr' opcode */ | (offset & 0x003fffff);
    *((UINT16 *)(addr    )) = (inst >> 16) & 0xffff ;
    *((UINT16 *)(addr + 2)) = (inst      ) & 0xffff ;
}
.....
void __interrupt IntEntry()  /* ←Interrupt handling routine */
{
    .......
}
.....
main()
{
    ........
           SetAJump((int)(0x080 + 0x3e78000) ,(int)IntEntry) ;
           /*               ↑Exception code of the target interrupt  */
```

4.6.2. General Restrictions and Cautions

The restrictions and cautions on debugging applications that use maskable interrupts are described below:

1) If an interrupt is generated before an alternate vector has been established, or if an interrupt is generated without establishing an alternate vector correctly, the program will break at the point where the interrupt is generated. The reason for this is that the initial value of an alternate vector is a branch instruction for branching to the break handling routine of the ROM monitor.
2) If the address of the interrupt handling routine relative to the alternate vector area exceeds 22 bits, either the value of at least one register must be destroyed or a relay point for branching must be established in order to branch to the interrupt handling routine.
3) The alternate vector area is protected as an area managed by the ROM monitor. It cannot, therefore, be changed by downloading a program.
4) All peripheral devices including those related to interrupts can be initialized only with the reset switch on the board. If, therefore, a program is reloaded and executed after it has been executed, the effects of the previous execution of the program remain on the peripheral devices. For a program using peripheral devices, the following procedure must be observed if the program is to be executed from the beginning after it has been executed: Disconnect rteserv, press the reset button on the RTE-V850E/MS1-PC, and then reconnect rteserv.
5) A program must be set to the DI (disable interrupt) state at the beginning. Then, after the necessary peripheral devices and vectors have been established, the program must be set to the EI (enable interrupt) state.

4.6.3. Restrictions and Cautions on Using Breakpoints

A breakpoint can be established within an interrupt handling routine to cause a break. After a break, the interrupt handling routine can be executed in single-step mode. In this case, note the following restrictions and cautions:

1) During a break, no maskable interrupts can be accepted.
2) The single-step function establishes a temporary break point for the next instruction. Consequently, if a user program in the EI (enable interrupts) state is executed in single-step mode, the program may accept an interrupt, branching to an interrupt handling routine, which, in turn, handles the interrupt, while a single instruction is executed in single-step mode. Even in single-step mode, therefore, pay careful attention to the cautions relating to breakpoints.
3) It is not possible to exit from an interrupt handling routine by using single-step execution. (To be more precise, single-step execution cannot be performed at the last "}" of an interrupt handling routine.) In addition, it is not possible to perform single-step execution of the reti instruction.
4) It is not possible to return from an interrupt handling routine to the previous routine by using the "Return" function of the debugger.

4.7. RTE COMMANDS

When using the board with the Multi debugger, connect the board to the Midas server (rteserv) to open the TARGET window. The RTE commands can be issued in this window. The following table lists the RTE commands.

Command	Description
HELP, ?	Displays help messages.
INIT	Initializes.
VER	Displays the version number.
SFR	Changes or displays the internal register (SFR).

RTE Commands

Some commands require parameters. All numeric parameters such as addresses and data are assumed to be hexadecimal numbers. The following numeric representations are invalid:

0x1234 1234H $1234

4.7.1. HELP (?)

<Format> HELP [command-name]

Displays a list of RTE commands and their formats. A question mark (?) can also be used in place of the character string HELP. If no command name is specified in the parameter part, the HELP command lists all usable commands.

<Example> HELP SFR

Displays help messages for the SFR command.

4.7.2. INIT

<Format> INIT

Initializes the RTE environment. Usually, this command should not be used.

4.7.3. VER

<Format> VER

Displays the version number of the current RTE environment.

4.7.4. SFR

<Format> SFR[register-name[data]]

Displays the data in the CPU's internal register (SFR) or writes data into it.

When the parameters are omitted, this command displays a list of register names which can be specified with the SFR command.

When only the register-name parameter is specified, this command reads and displays the data in the specified register.

When both the register-name and data parameters are specified, this command writes the data into the specified register.

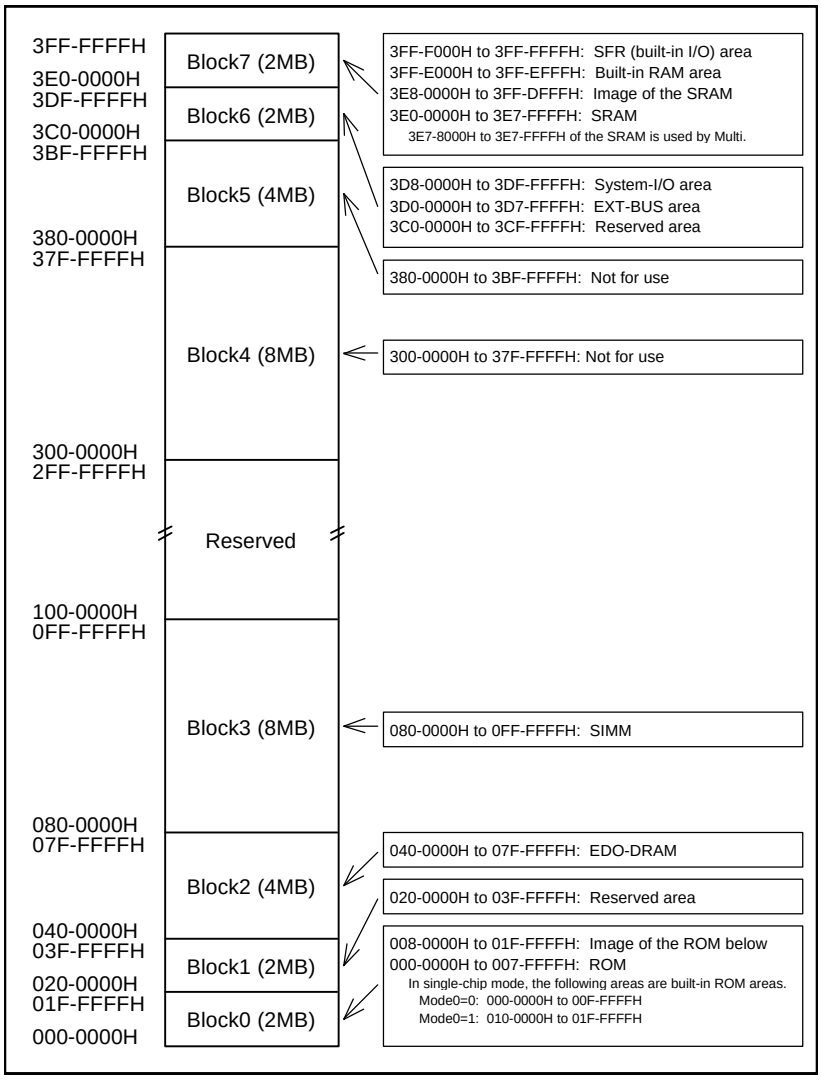
Specifying a display command for a read-inhibited register or a write command for a write-inhibited register results in an error. The access size for read or write is automatically set to the same size as that of the register.

5. HARDWARE REFERENCES

This chapter describes the hardware of the RTE-V850E/MS1-PC.

5.1. MEMORY MAP

The memory assignment of the board is shown below.



Memory Map

5.2. DETAILS OF THE MEMORY MAP

The memory map is explained in detail below. For an explanation of how to set the SFRs required to access these memory spaces, see Chapter 7.

CS0 space (Block0: 000-0000H to 01F-FFFFH, 16-bit ROM space)

This is the space for the ROM mounted on the board. A maximum of 512 Kbytes of ROM can be used. As standard, 46K × 16 bits (128 Kbytes) of ROM with an access time of 120 ns or less is mounted. Wait control uses the programmable wait function of the CPU. For an explanation of the required wait count, see Chapter 7.

The ROM mounted as standard incorporates the Multi debugger monitor.

When the CPU is in single-chip mode, the built-in flash ROM of the CPU appears as part of this area. When the MODE0 pin of the CPU is Low, the built-in flash ROM area appears between 000-0000H and 00F-FFFFH. When the MODE0 pin is High, it appears between 010-0000H and 01F-FFFFH. Except for these areas, external ROM can be accessed even in single-chip mode.

The built-in flash ROM is 1 Mbyte in size; however, the actual flash ROM is 128 Kbytes in size.

CS1 space (Block1: 020-0000H to 03F-FFFFH, 16-bit reserved area)

This area is reserved by the system. Do not attempt to access this area. Operation is not guaranteed if this area is accessed.

CS2 space (Block2: 040-0000H to 07F-FFFFH, 16-bit EDO-DRAM space)

This is the space for the EDO-DRAM (Hyper-Page) mounted on the board. Two 2M × 8-bit EDO-DRAM chips (4 Mbytes) are mounted. The width of the data bus is set to 16 bits.

CS3 space (Block3: 080-0000H to 0FF-FFFFH, 16-bit SIMM space)

This is the space for the SIMM mounted in the SIMM socket on the board. When a SIMM is not mounted, this space can be used as an extended space by using a CPU connector.

Either a 4-Mbyte or 8-Mbyte SIMM can be used. The SIMM may be of either Fast-Page or Hyper-Page type.

CS4 space (Block4: 300-0000H to 37F-FFFFH, not for use when DMA is used)

This space cannot be used when DMA is used, because the CS4-/RAS4-/IOWR-/P84 pins of the CPU are used for IOWR-. If DMA is never used, this space can be used as an extended space by using a CPU connector. To do this, the appropriate DIP switch must be set to disable the use of IORD- and IOWR-. (See Section 3.1.1.)

CS5 space (Block5: 380-0000H to 3BF-FFFFH, not for use when DMA is used)

This space cannot be used when DMA is used, because the CS5-/RAS5-/IORD-/P85 pins of the CPU are used for IORD-. If DMA is never used, this space can be used as an extended space by using a CPU connector. To do this, the appropriate DIP switch must be set to disable the use of IORD- and IOWR-. (See Section 3.1.1.)

CS6 space (Block6: 3C0-0000H to 3DF-FFFFH, 16-bit extended and I/O space)

This space is divided into three spaces. Wait control is performed by hardware using the WAIT- pin of the CPU.

Reserved area (3C0-0000H to 3CF-FFFFH):

Reserved by the system. Do not attempt to access this area. Operation is not guaranteed if this area is accessed.

EXT-BUS area (3D0-0000H to 3D7-FFFFH):

Allocated for EXT-BUS. Access to both the memory and I/O spaces of EXT-BUS is performed using this area. Because the size of this area is only 512 Kbytes as compared with the 16-Mbyte address space of EXT-BUS, the high-order bits of an address are specified using a bank port. The bank port is also used to specify which of the memory and I/O spaces of EXT-BUS are to be accessed. (See Sections 6.10, 6.11, and 6.12.)

System-I/O area (3D8-0000H to 3DF-FFFFH):

Allocated for the timers, serial/parallel ports, and other control ports mounted on the board. (See Section 6.1.)

CS7 space (Block7: 3E0-0000H to 3FF-FFFFH, 16-bit SRAM space)

This is the space for the SRAM mounted on the board. Four 128K × 8-bit SRAM chips (512 Kbytes) with an access time of 15 ns are mounted. Wait control uses the programmable wait function of the CPU. For an explanation of the required wait count, see Chapter 7.

Because this area contains the built-in RAM area and the SFR (built-in I/O) area of the CPU, the SRAM area between 3FF-E000H to 3FF-FFFFH cannot be accessed.

When ROM monitor for the Multi debugger is used, 3E7-8000H to 3E7-FFFF are used by that monitor. (See Section 4.5.2.)

6. SYSTEM-I/O

SYSTEM-I/O is an I/O device mapped in a memory space. The I/O devices include the UART/PRINTER, TIC, and ISA bus interface.

6.1. SYSTEM-I/O LIST

The following table lists the SYSTEM-I/O functions.

Address	Function	Note
3D8-0000H to 3D8-000EH	Sets/references UART-CH#0 (TL16PIR552)	
3D8-0010H to 3D8-001EH	Sets/references UART-CH#1 (TL16PIR552)	
3D8-0020H to 3D8-002EH	Sets/references PRINTER (PPCS-) (TL16PIR552)	
3D8-0030H to 3D8-003EH	Sets/references PRINTER (ECPCS-) (TL16PIR552)	
3D8-0040H to 3D8-0046H	Sets/references timer controller (μPD71054)	Recovery time is required.
3D8-0050H	Sets 7-segment LED display data	
3D8-0050H	References DIPS SW7	
3D8-0060H	References status (SIMM-PD, etc.)	
3D8-0070H to 3D8-007FH	Reserved by the system	Access prohibited.
3D8-0080H	Sets/references control ports (Mask, Tover, etc.)	
3D8-0090H	Sets/references NMI and interrupt selection	
3D8-00A0H	References interrupt status	
3D8-00B0H	Sets/references a bank port for EXT-BUS CPU-Core	
3D8-00C0H	Sets/references a bank port for EXT-BUS DMA0	
3D8-00D0H	Sets/references a bank port for EXT-BUS DMA1.	
3D8-00E0H to 3D8-00FFH	Reserved by the system	Access prohibited.

6.2. UART/PRINTER (TL16PIR552) (3D8-0000H TO 3D8-003EH)

The TL16PIR552 (dual UART with 1284 parallel ports) LSI chip produced by Texas Instruments is used as UART/PRINTER. The TL16PIR552 has two UART channels and one bidirectional printer port conforming to IEEE1284. It incorporates a 16-character FIFO buffer in the UART transmission/reception circuitry and is capable of automatically controlling the RTS/CTS flow, thereby preventing overrun errors with minimum interrupts.

Each register in the TL16PIR552 is assigned as listed below. Refer to the applicable TL16PIR552 manual for an explanation of the function of each register. (TL16PIR552 manuals can be obtained from the TI&ME page of the Texas Instruments US web site (http://www.ti.com/).)

Address	Function	Read	Write
3D8-0000H	UART-CH#0	RBR/DLL	THR/DLL
3D8-0002H		IER/DLM	IER/DLM
3D8-0004H		IIR	FCR
3D8-0006H		LCR	LCR
3D8-0008H		MCR	MCR
3D8-000AH		LSR	LSR
3D8-000CH		MSR	MSR
3D8-000EH		SCR	SCR
3D8-0010H	UART-CH#1	RBR/DLL	THR/DLL
3D8-0012H		IER/DLM	IER/DLM
3D8-0014H		IIR	FCR
3D8-0016H		LCR	LCR
3D8-0018H		MCR	MCR
3D8-001AH		LSR	LSR
3D8-001CH		MSR	MSR
3D8-001EH		SCR	SCR
3D8-0020H	PRINTER(PPCS-)	DATA	DATA/ECPAFIFO
3D8-0022H		DSR	-----
3D8-0024H		DCR	DCR
3D8-0026H		EPPADDR	EPPADDR
3D8-0028H to 3D8-002EH		EPPDATA	EPPDATA
3D8-0030H	PRINTER(ECPCS-)	PPDATAFIFO/ TESTFIFO/CNFGA	PPDATAFIFO/ TESTFIFO
3D8-0032H		CNFGB	-----
3D8-0034H		ECR	ECR

TL16PIR552 Register Arrangement

The XIN input of the TLC16PIR552 is connected to the 22.1184-MHz clock.

The UART-CH#0, UART-CH#1 and PRINTER interrupts can be connected to the CPU's interrupt ports as shown in the following table.

Interrupt source	Interrupt to connected CPU	Interrupt edge
UART-CH#0	NMI/P20, INTP130/P34 , INTP131/SO2/P35	Rising edge
UART-CH#1	INTP132/SI2/P36	Rising edge
PRINTER	INTP133/SCK2-/P37	Rising edge

Maskable interrupts pass through DIP SW2. See Section 10.3 for details of the NMI, and Sections 10.4 and 10.12 for details of the maskable interrupt.

UART-CH#0 is connected to the JSIO1 connector on the rear of the board, UART-CH#1 is connected to the JSIO2 connector and PRINTER is connected to JPRT. UART-CH#0 is used when using the Multi debugger for serial communication. In this time, NMI is used.

The TL16PIR552 is reset when the system is reset.

TIC (μPD71054) (3F-F040H TO 3F-F048H)

The μPD71054 produced by NEC is installed as a TIC. The μPD71054 is compatible with the i8254 produced by Intel. It has three timers/counters. These timers/counters are used to generate the timer interrupts of the ROM monitor for the Multi debugger.

Each register of the TIC is assigned as listed below.

Address	Read	Write
3D8-0040H	COUNTER#0	COUNTER#0
3D8-0042H	COUNTER#1	COUNTER#1
3D8-0044H	COUNTER#2	COUNTER#2
3D8-0046H	-----	Control Word

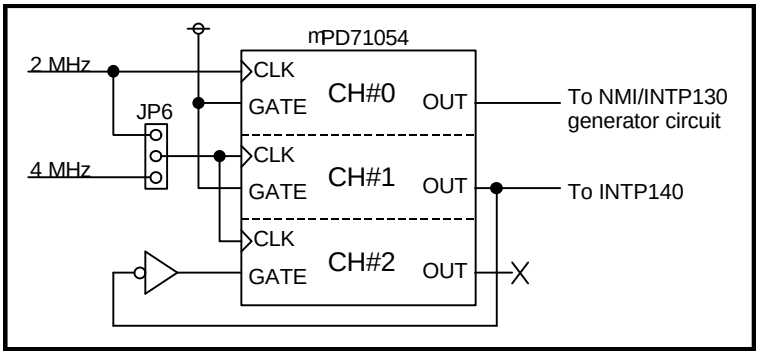
TIC Register Arrangement

The channels of the TIC are connected as shown in the figure below.

CH#0 is connected to the NMI and INTP130 generator circuit and is used as the interval timer of the ROM monitor program for the Multi debugger. A 2-MHz clock is connected to the clock input.

CH#1 can be used as required by the user program. It is connected to INTP140/P114 of the CPU. CH#1 also functions as the prescale counter for CH#2.

CH#2 can be used as required by the user program. It is not connected to interrupts. Either a 2-MHz or 4-MHz clock can be connected to the clock input of CH#1 and CH#2 by using JP6 on the board.



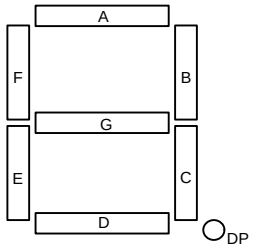
The μPD71054 requires 165 ns of command recovery time. See Chapter 7 for details of the recovery time.

The TIC is reset when the system is reset.

6.3. 7-SEGMENT LED DISPLAY DATA OUTPUT PORT (3D8-0050H [WRITE ONLY])

This port sets the data to be displayed on the 7-segment LED on the board. The data format is as shown in table below. Setting a bit to 0 causes the corresponding segment to light.
The initial value at power on is undefined.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DPseg	Gseg	Fseg	Eseg	Dseg	Cseg	Bseg	Aseg



6.4. DIP SW7 READ PORT (3D8-0050H [READ ONLY])

This port is used to read the status of DIP SW7 on the board. The data format is as shown in table below.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
SW7-8	SW7-7	SW7-6	SW7-5	SW7-4	SW7-3	SW7-2	SW7-1

SW7-[8..1]: The status of SW7 on the board can be read. SW7-1 corresponds to switch "1" of DIP SW7, while SW7-8 corresponds to switch "8" of DIP SW7. When a bit is ON, it represents 0. When a bit is OFF, it represents 1.
When the ROM monitor for the Multi debugger is used, some of the DIP SW7 bits are reserved by the monitor in the ROM mounted on the board. (See Section 4.3.7.) However, note that these functions are defined in terms of software. This means that, even when the switch setting is changed, these functions are not affected by this (hardware-based) action.

6.5. STATUS READ PORT (3D8-0060H [READ ONLY])

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
NMI/INTP-	SFLASH_BY-	SFLASH_WP-	SIMM_EDO-	PD4	PD3	PD2	PD1

PD[4..1]: The status of PD[4..1] of the DRAM (72-pin SIMM) mounted on the board can be read. The status of PD[2..1] indicates the size of the mounted DRAM. The relationship between PD[2..1] and the DRAM capacity is given in table below.

PD[2]	PD[1]	DRAM capacity
0	0	4 Mbytes
0	1	Reserved
1	0	16 Mbytes (Disabled)
1	1	8 Mbytes

PD[2..1] and DRAM Capacity

SIMM_EDO-: The status of DIP SW4-4 can be read. This switch is set by the user according to whether the SIMM mounted in the SIMM socket on the board is of EDO type. Note that this function is defined in terms of software. This means that, even when the switch setting is changed, this function is not affected by this (hardware-based) action. When DIP SW4-4 is OFF, it represents 1. When it is ON, it represents 0. According to the status of this switch, software can change the internal settings of the CPU (SFRs). Upon initialization, the ROM monitor for the Multi debugger sets SFRs according to the value of this bit. (See Section 4.5.6.)

SFLASH_WP-: The status of DIP SW4-1 can be read. This switch is reserved by the system and must be set to OFF. To allow for future expansion, programs must be created such that they are not affected by the value of this bit.

SFLASH_BY-: This switch is reserved by the system. To allow for future expansion, programs must be created such that they are not affected by the value of this bit.

NMI/INTP-: The status of DIP SW4-5 can be read. This switch is reserved by the system and must be set to OFF. To allow for future expansion, programs must be created such that they are not affected by the value of this bit.

6.6. CONTROL PORT (3D8-0080H [READ/WRITE])

This port has control bits related to interrupts. At power-on or reset, the initial value is 0x03.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Unused	Unused	Unused	SFLASH_RP-	TM0_INT_CLR	TOV_INT_CLR	INTP130_MASK	NMI_MASK

NMI_MASK: This bit controls the final masking of an NMI. For an explanation of the NMI signal generation logic, see Section 10.3.

INTP130_MASK: This bit controls the final masking of an INTP130. For an explanation of the INTP130 signal generation logic, see Section 10.4.

TOV_INT_CLR: The interrupt caused by a ready timeout of the CPU is retained by hardware and connected to the CPU. This bit clears the status of the retained interrupt. By setting this bit to "1" and then to "0", the status of the interrupt is cleared. Usually, this bit must be set to "0".

TM0_INT_CLR: The interrupt from channel 0 of the TIC (μPD71054) is retained by hardware and connected to the CPU. This bit clears the status of the retained interrupt. By setting this bit to "1" and then to "0", the status of the interrupt is cleared. Usually, this bit must be set to "0".

SFLASH_RP-: This bit is reserved by the system. To allow for future expansion, programs must be created so as not to change the value of this bit. If a program must change the setting, it must read this port and manipulate this bit only.

Unused bits: To allow for future expansion, programs must be created so as not to change the values of the unused bits. If a program must change the settings, it must read this port and manipulate the bits to be changed only.

6.7. NMI/INTP130 SELECT PORT (3D8-0090H [READ/WRITE])

This port controls the generation of NMI and INTP130 signals. For an explanation of the NMI and INTP130 signal generation logic, see Sections 10.3 and 10.4, respectively.
At power-on or a reset, the initial value is 0x00.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TOV_INTEN	TM0_INTEN	ISACOM_INTEN	UART0_INTEN	TOV_NMIEN	TM0_NMIEN	ISACOM_NMIEN	UART0_NMIEN

Set each bit to "1" when NMI or INTP130 is to be generated according to the corresponding interrupt request, or to "0" when it need not be generated.

UART0_NMIEN, UART0_INTEN : Interrupt request issued by UART-CH#0 of the TL16PIR552
ISACOM_NMIEN, ISACOM_INTEN : Interrupt request based on communication with the ISA bus
TM0_NMIEN, TM0_INTEN : Interrupt request issued by TOUT0 of the TIC (μPD71054)
TOV_NMIEN, TOV_INTEN : Interrupt request resulting from time-over ready occurrence

6.8. NMI/INTP130 STATUS PORT (3D8-00A0H [READ ONLY])

This port is used to identify the source of an NMI or INTP130 request. For an explanation of the NMI and INTP130 signal generation logic, see Sections 10.3 and 10.4, respectively.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
----	----	----	----	TOV_INTRQ	TM0_INTRQ	ISACOM_INTRQ	UART0_INTRQ

Each bit is set to "1" when NMI or INTP130 is generated by the corresponding interrupt request, or set to "0" when it is not generated. Each bit represents the interrupt request status of each interrupt request source and is not affected by the settings of the NMI/INTP130 select port. Therefore, the interrupt factor causing the NMI or INTP130 occurrence can be identified by ANDing the information in the NMI/INTP130 status port with that in the NMI/INTP130 select port.

UART0_INTRQ : Interrupt request issued by UART-CH#0 of the TL16PIR552
ISACOM_INTRQ : Interrupt request based on communication with the ISA bus
TM0_INTRQ : Interrupt request issued by TOUT0 of the TIC (μPD71054)
TOV_INTRQ : Interrupt request resulting from time-over ready occurrence

6.9. BANK PORT FOR EXT-BUS CPU-CORE (3D8-00B0H [READ/WRITE])

This port is used to set A[23:19] of the address to be output to EXT-BUS when EXT-BUS is to be accessed using an address between 3D0-0000H and 3D7-FFFFH in the memory map. It is also used to specify which of the memory and I/O spaces of EXT-BUS is to be accessed. The bank address set with this port is used when CPU-Core is to perform access (when the program causes the CPU to access).
At power-on or a reset, the initial value is 0x01.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CC_EXTA23	CC_EXTA22	CC_EXTA21	CC_EXTA20	CC_EXTA19	Unused	Unused	CC_MEMSEL

CC_EXTA[23:19]: A[23:19] of the address to be output to EXT-BUS can be set. A[18:2] of the address to be output to EXT-BUS is output from the address line of the CPU.

CC_MEMSEL: This bit is used to specify which of the memory and I/O spaces of EXT-BUS is to be accessed using an address between 3D0-0000H and 3D7-FFFFH. Set this bit to "1" to access the memory space and "0" to access the I/O space.

Unused bits: To allow for future expansion, programs must be created so as not to change the values of the unused bits. If a program must change the settings, it must read this port and manipulate the bits to be changed only.

6.10. BANK PORT FOR EXT-BUS DMA0 (3D8-00C0H [READ/WRITE])

This port is used to set A[23:19] of the address to be output to EXT-BUS when EXT-BUS is to be accessed using an address between 3D0-0000H and 3D7-FFFFH in the memory map. It is also used to specify which of the memory and I/O spaces of EXT-BUS is to be accessed. The bank address set with this port is used when DMA channel 0 is used for 2-cycle DMA (DMA cycle).

At power-on or a reset, the initial value is 0x01.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
D0_EXTA23	D0_EXTA22	D0_EXTA21	D0_EXTA20	D0_EXTA19	Unused	D0_2CYCLE	D0_MEMSEL

D0_EXTA[23:19]: A[23:19] of the address to be output to EXT-BUS can be set. A[18:2] of the address to be output to EXT-BUS is output from the address line of the CPU.

D0_2CYCLE: Set this bit to "1" to use DMA channel 0 of the CPU for 2-cycle DMA. Set it to "0" to use it for fly-by DMA.

D0_MEMSEL: This bit is used to specify which of the memory and I/O spaces of EXT-BUS is to be assessed using an address between 3D0-0000H and 3D7-FFFFH. Set this bit to "1" to access the memory space and "0" to access the I/O space.

Unused bit: To allow for future expansion, programs must be created so as not to change the value of the unused bit. If a program must change the setting, it must read this port and manipulate the bit only.

6.11. BANK PORT FOR EXT-BUS DMA1 (3D8-00D0H [READ/WRITE])

This port is used to set A[23:19] of the address to be output to EXT-BUS when EXT-BUS is to be accessed using an address between 3D0-0000H and 3D7-FFFFH in the memory map. It is also used to specify which of the memory and I/O spaces of EXT-BUS is to be accessed. The bank address set with this port is used when DMA channel 1 is used for 2-cycle DMA (DMA cycle).

At power-on or a reset, the initial value is 0x01.

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
D1_EXTA23	D1_EXTA22	D1_EXTA21	D1_EXTA20	D1_EXTA19	Unused	D1_2CYCLE	D1_MEMSEL

D1_EXTA[23:19]: A[23:19] of the address to be output to EXT-BUS can be set. A[18:2] of the address to be output to EXT-BUS is output from the address line of the CPU.

D1_2CYCLE: Set this bit to "1" to use DMA channel 1 of the CPU for 2-cycle DMA. Set it to "0" to use it for fly-by DMA.

D1_MEMSEL: This bit is used to specify which of the memory and I/O spaces of EXT-BUS is to be accessed using an address between 3D0-0000H and 3D7-FFFFH. Set this bit to "1" to access the memory space and "0" to access the I/O space.

Unused bit: To allow for future expansion, programs must be created so as not to change the value of the unused bit. If a program must change the setting, it must read this port and manipulate the bit only.

7. RECOMMENDED SETTINGS

This chapter specifies the recommended values for the parameters related to access to memory resources.

7.1. CPU SETTING

No restrictions are imposed on the settings of the bus control function built into the CPU. Therefore, to maximize the bus performance, set BCC (0xffff to f062) of SFR to 0x0000.

7.2. CS0/CS7 SPACES (SRAM/ROM)

For SRAM and ROM, waits are generated using the programmable wait function of the CPU (SFRs DWC1 and DWC2). The table below lists the recommended wait counts when ROM with an access of 120 ns/150 ns and SRAM are used.

Operating frequency	SRAM wait count	ROM (120 ns) wait count	ROM (150 ns) wait count
40 MHz	1Wait	5Wait	6Wait
33 MHz	1Wait	4Wait	5Wait
25 MHz	0Wait	3Wait	4Wait

7.3. CS2 SPACE (EDO-DRAM)

The table below lists the recommended settings for the EDO-DRAM in the CS2 space (values to be set in the SFR (DRC and RWC register)). The refresh cycle of EDO-DRAM is 1024cycles/128mS.

Operating frequency	DCR.PAE	DCR.RPC	DCR.RHC	DCR.DAC	DCR.CPC	DCR.RHD	DCR.DAW
40 MHz	10(EDO)	2	0	1	0	0	10(10Bit)
33 MHz	10(EDO)	1	0	0	0	0	10(10Bit)
25 MHz	10(EDO)	1	0	0	0	0	10(10Bit)

Operating frequency	RWC.RRW	RWC.RCW	RWC.SRW
40MHz	1	0	2
33MHz	0	0	2
25MHz	0	0	2

7.4. CS3 SPACE (SIMM)

The table below lists the recommended settings for the SIMM in the CS3 space (values to be set in the SFR (DRC and RWC register)). PAE must be set according to the type of the SIMM.

Operating frequency	SIMM	DCR.RPC	DCR.RHC	DCR.DAC	DCR.CPC	DCR.RHD	DCR.DAW
40 MHz	60 ns	2	1	2	0	0	10(10Bit)
	70 ns	2	1	2	0	0	10(10Bit)
33 MHz	60 ns	1	1	1	0	0	10(10Bit)
	70 ns	2	1	1	0	0	10(10Bit)
25 MHz	60 ns	1	0	1	0	0	10(10Bit)
	70 ns	1	0	1	0	0	10(10Bit)

Operating frequency	SIMM	RWC.RRW	RWC.RCW
33MHz	60nS	1	0
	70nS	1	1
33MHz	60nS	0	0
	70nS	1	0
25MHz	60nS	0	0
	70nS	0	0

7.5. CS6 SPACE WAITS

For the CS6 space, a circuit on the board generates a WAIT- signal; therefore, the programmable wait count of the CPU must be set to 0.

7.6. CS6 SPACE COMMAND RECOVERY TIME

Access to the mPD71054 must satisfy the command recovery time requirement. Therefore, when consecutive accesses to the mPD71054 are to be performed, the second access must be performed a certain length of time after the first access.

The recovery time is generated by performing read access to a memory resource other than the mPD71054. The memory resource which is recommended to be read-accessed for generating the recovery time is the DIP SW7 read port (3D8-0050H). After the mPD71054 is accessed, read access to DIP SW7 must always be performed once.

Remember that no write cycles must be generated for the CS6 space until the recovery time requirement has been satisfied after the mPD71054 is accessed. If, therefore, a subroutine is used to perform recovery time generation, make sure that the CS6 space does not have a stack.

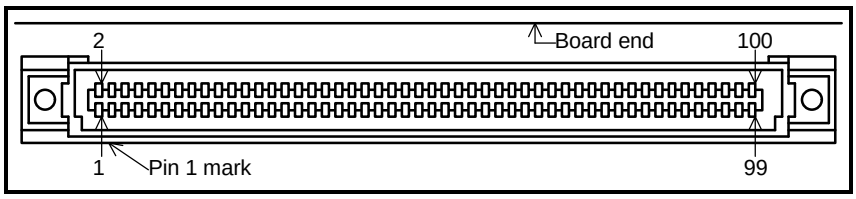
8. EXT-BUS SPECIFICATION

The JEXT is a connector for EXT-BUS which is used to expand memory and I/O units. The local bus on this board is connected to the JEXT connector.

8.1. PIN ARRANGEMENT

Number	Signal name	Number	Signal name	Number	Signal name	Number	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

JEXT Connector Pin Arrangement



JEXT Connector Pin Arrangement

8.2. SIGNALS

Signal name	Input/output	Function
D[0..31]	Input/output	Data bus signals, each of which is originally the CPU data bus signal received at a buffer. Each signal is pulled up with a 10-kΩ resistor on the board.
A[1..23]	Output	Address bus signals, each of which is originally the CPU address signal received at a buffer.
MRD-	Output	Memory read cycle timing signal, which becomes active only when the EXT-BUS space is accessed.
MWR-[0..3]	Output	Memory write cycle timing signals, each of which becomes active only when the EXT-BUS space is accessed. MWR0- corresponds to D[0..7]; MWR1- to D[8..15]; MWR2- to D[16..23]; and MWR3- to D[24..31].
IORD-	Output	I/O read cycle timing signal, which becomes active when the EXT-BUS space is accessed or in a fly-by DMA cycle.
IOWR-	Output	I/O write cycle timing signal, which becomes active when the EXT-BUS space is accessed or in a fly-by DMA cycle.
READY	Input	Signal notifying the CPU of the end of a cycle. It is valid only for the EXT-BUS space. To have the CPU recognize READY reliably, it is necessary to keep READY active until MRD-, MWR-[0..3], IORD-, or IOWR- becomes inactive. It is pulled up with a 10-kΩ resistor on the board.
INT-[0..3]	Input	Active-low interrupt request signals, which are connected to the INTP150, INTP151, INTP152, and INTP153 pins of the CPU, respectively, via a buffer and SW3. Each signal is pulled up with a 10-kΩ resistor on the board. (See Section 10.12.)
DMARQ-[0..1]	Input	Active-low DMA request signals, which are connected to the DMARQ0- and DMARQ1- pins of the CPU, respectively, via a buffer and SW1. Each signal is pulled up with a 10-kΩ resistor on the board. (See Section 10.11.)
DMAAK-[0..1]	Output	Active-low DMA response signals, to which the DMAAK0- and DMAAK1- pins of the CPU are connected, respectively, via SW1 and a buffer. (See Section 10.12.)
RESET-	Output	Active-low system reset signal
32/16BIT-	Input	When this signal is Low, in the case that CPU has 16bit data-bus, only D[15..0] of the data bus are used (16-bit bus mode).In the case that CPU has 32bit data-bus, any of D[15..0] or D[31..16] are used by an address. When it is High, D[31..0] of the data bus are used (32-bit bus mode). This signal cannot be change dynamically. It is pulled up with a 10-kΩ resistor on the board.
CLK	Output	Clock signal, to which the CLKOUT pin of the V850E/MS1 is connected via a buffer.
Reserve	-	Reserved signal. For a board using EXT-BUS, do not connect anything to this pin.

JEXT Connector Signals

Cautions:

1. The 32/16BIT- signal will not be supported by all future RTE-PC series CPUs. If a board connected to EXT-BUS is scheduled to be used with a future RTE-PC series CPU, the board must be designed to run in 32-bit bus mode.

   When 32/16BIT- is Low, MWR2- and MWR3- can never be asserted; MWR0- and MWR1- can be asserted.

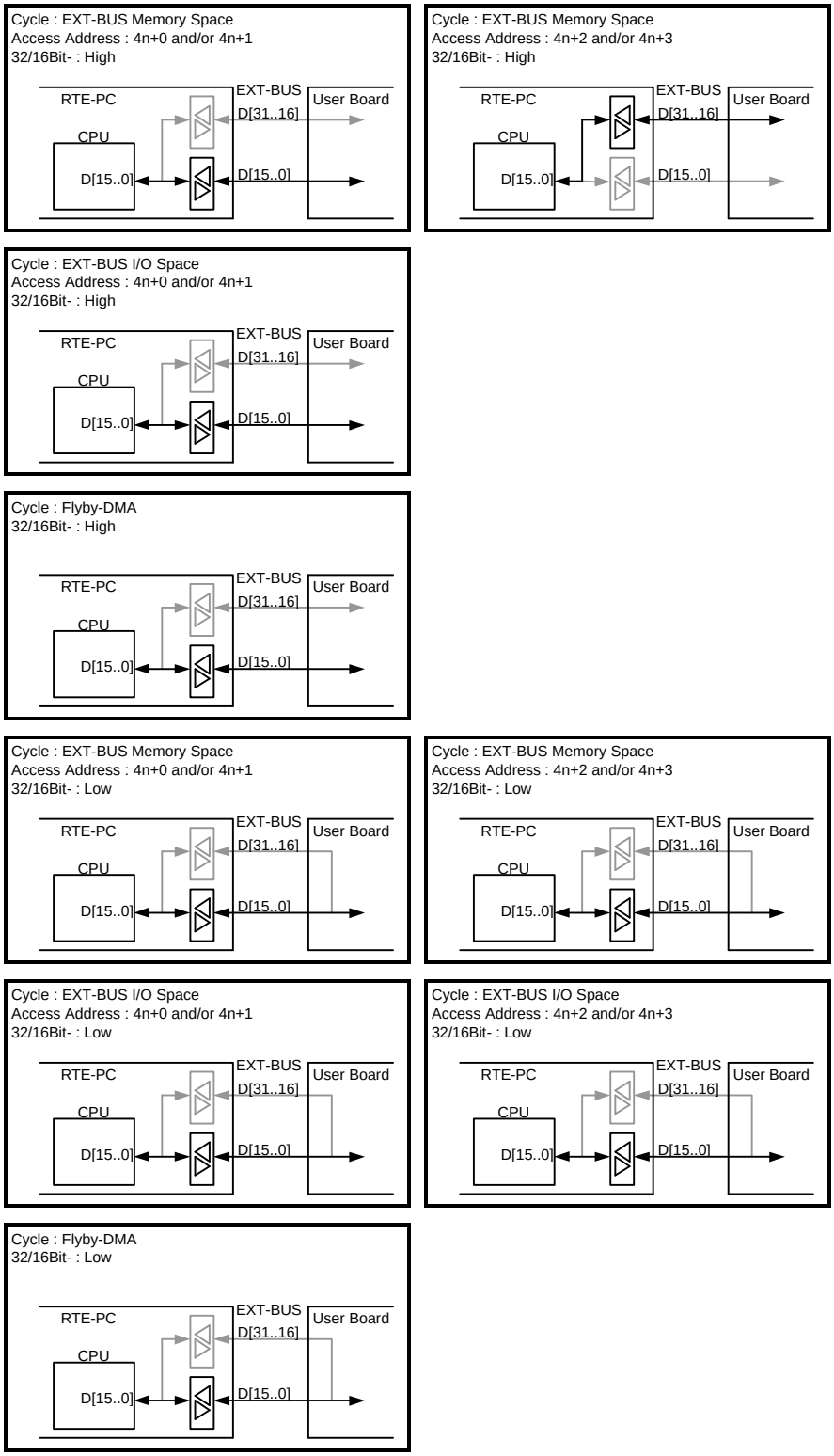
   Whien 32/16BIT- is Low, Cause D[15..0] and D[31..16] a short on a board connected to EXT-BUS (See Section 8.3).
2. A1 is effective when the 32/16BIT- signal is Low. Thus, A1 may not be output by future RTE-PC series CPUs not supporting the 32/16BIT- signal.
3. The maximum access bus width in one cycle of the EXT-BUS bus depends on the width of the data bus of the CPU. For a CPU with a 16-bit data bus, for example, the maximum width of I/O

access and fly-by DMA access is 16 bits. Therefore, the port to be accessed in an I/O cycle or fly-by DMA cycle of the board connected to EXT-BUS must have a data bus width equal to or less than that of the CPU to be connected, and its address must be 4n+0.

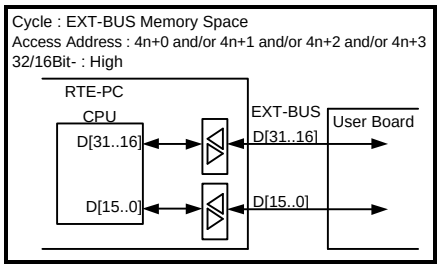
4. The DMA function will not be supported by all future RTE-PC series CPUs.

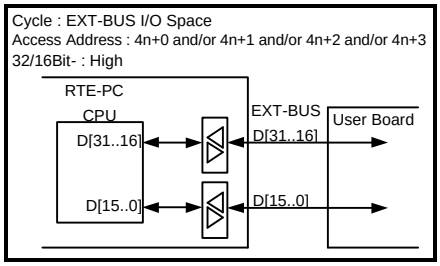
8.3. CONECTION OF DATA-BUS

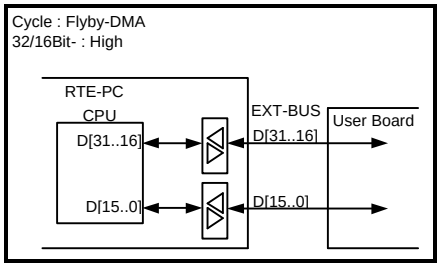
8.3.1. 16BIT DATA-BUS CPU (V850E/MS1)

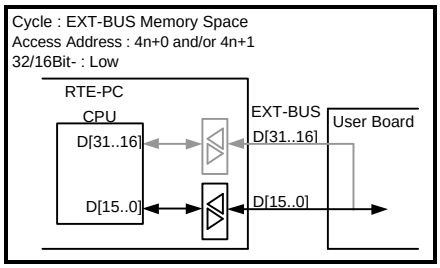


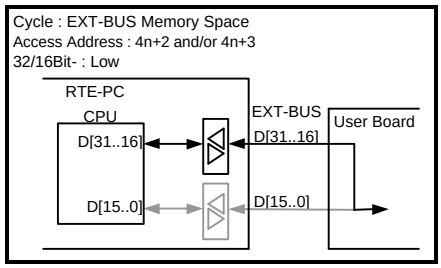
8.3.2. 32BIT DATA-BUS CPU (for a reference)

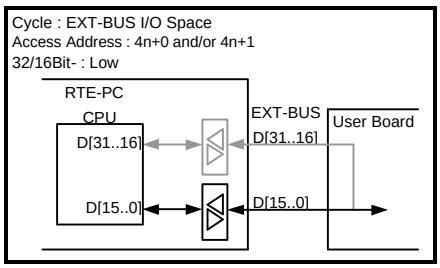


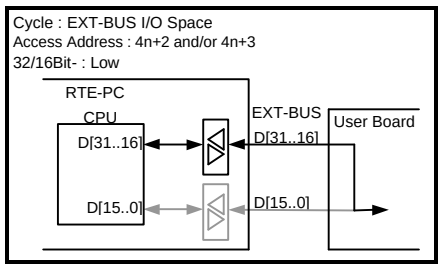


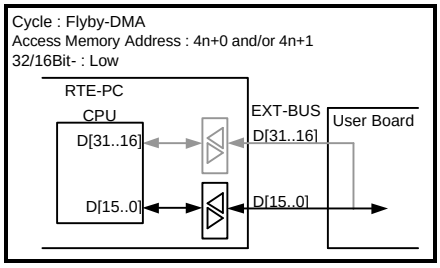


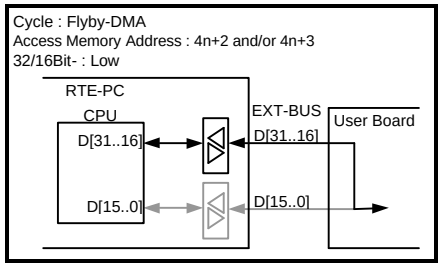




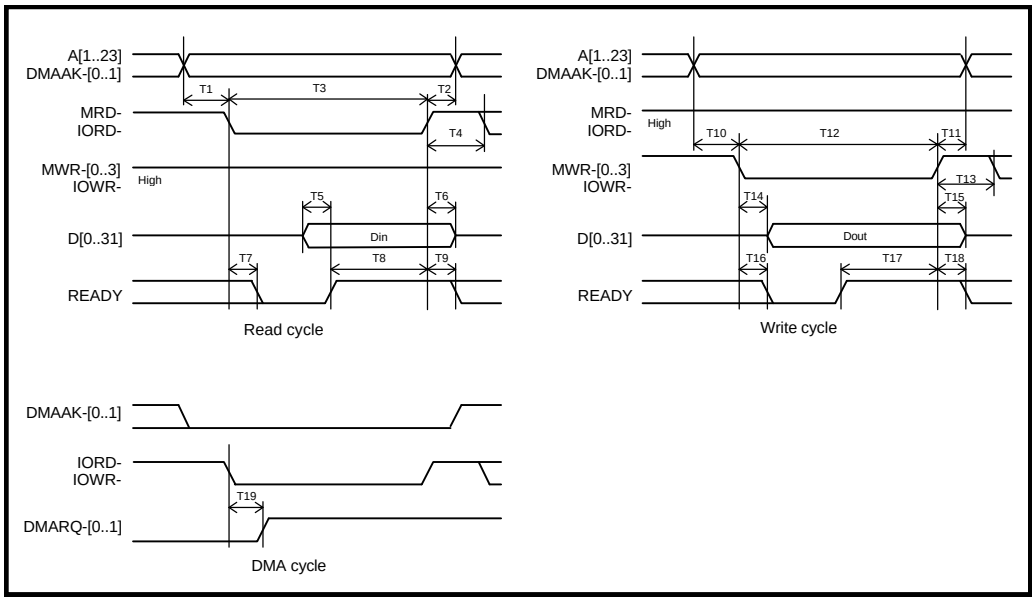








8.4. TIMING



EXT-BUS Cycles

Symbol	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	MRD-, IORD- cycle interval	20	
T5	RD DATA → RD READY setup time	0	
T6	MRD-, IORD- → RD DATA hold time	0	
T7	MRD-, IORD- → RD READY delay		20
T8	RD READY → MRD-, IORD- delay	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	MWR-, IOWR- cycle interval	20	
T14	MWR-, IOWR- → WR DATA delay		20
T15	MWR-, IOWR- → WR DATA hold time	10	
T16	MWR-, IOWR- → WR READY delay		20
T17	WR READY → MWR-, IOWR- delay	0	
T18	MWR-, IOWR- → WR READY hold time	0	
T19	IORD-, IOWR- → DMARQ- inactive delay		20

EXT-BUS AC Specifications

8.5. CONNECTORS

The model numbers of the connector used by EXT-BUS and the connectors that match this connector are listed below. To connect multiple boards to EXT-BUS, connect them in a daisy chain fashion using appropriate cables.

Connector used by EXT-BUS	:	KEL 8830E-100-170S
Matching connector (for a board)	:	KEL 8802-100-170S
Matching connector (for a cable)	:	KEL 8825E-100-1705
Right angle for a cable (for a board)	:	KEL 8830E-100-170L
	:	KEL 8831E-100-170L

8.6. CAUTIONS

The following cautions must be observed when designing a board to be connected to EXT-BUS:

1. To connect multiple boards to EXT-BUS, Hi-Z control must be applied so that the READY signal is driven only when a board is selected.
2. To insert waits into an EXT-BUS cycle, T7 and T16 must be satisfied.
3. When a DMA cycle is to start in single transfer mode, T19 in the timing chart must be satisfied to ensure that the next DMA cycle does not occur. T19 is highly dependent on the specifications of the DMA controller used and, therefore, may be changed in future RTE-PC series CPUs.
4. The CPU does not have an I/O space. For the RTE-V850E/MS1-PC, therefore, which of the I/O and memory spaces of EXT-BUS is to be accessed is determined by the setting of the appropriate port. (See Sections 6.10, 6.11, and 6.12.)
5. The DMA transfer supported by the RTE-V850E/MS1-PC allows transfer between specific spaces only. (See Chapter 9.)

9. DMA

The RTE-V850E/MS1-PC allows DMA transfer between the external board connected to EXT-BUS and the memory on the RTE-V850E/MS1-PC by means of the built-in DMA controller of the CPU.
This chapter explains DMA transfer.

9.1. SPACES BETWEEN WHICH DMA TRANSFER IS POSSIBLE

DMA transfer using EXT-BUS is possible between these spaces.

- For fly-by transfer
  EXT-BUS I/O space ↔ SRAM on the RTE-V850E/MS1-PC
  EXT-BUS I/O space ↔ DRAM on the RTE-V850E/MS1-PC
  EXT-BUS I/O space ↔ SIMM on the RTE-V850E/MS1-PC
- For 2-cycle transfer
  Any EXT-BUS space ↔ Any space on the RTE-V850E/MS1-PC

9.2. DMA CHANNELS

The table below lists the relations between the DMA signals of EXT-BUS and the DMA channels of the CPU.

EXT-BUS signal	CPU DMA channel
DMARQ0-, DMAAK0-	Channel 0
DMARQ1-, DMAAK1-	Channel 1

9.3. DIP SW SETTING

To enable DMA transfer using EXT-BUS, the following DIP SW switches must be set to ON. (See Section 3.1.1.)

DMA channel used	Setting of DIP SW1					
	1	2	3	4	7	8
Channel 0	ON	ON			ON	ON
Channel 1			ON	ON	ON	ON

9.4. CPU SETTING

To enable DMA transfer, the DMARQ- and DMAAK- pins of the channel to be used must be set up for use in DMA and the SFRs of the CPU must be set up so that CS4-/RAS4-/IOWR-/P84 and CS5-/RAS5-/IORD-/P85 are used as IOWR- and IORD-, respectively.

9.5. BANK PORT SETTING (2-CYCLE DMA)

To enable 2-cycle DMA transfer using EXT-BUS, an EXT-BUS DMA bank port must be set up. It is also necessary to specify which of the memory and I/O spaces of EXT-BUS is to be accessed with DMA. (See Sections 6.11 and 6.12.)

Three EXT-BUS DMA bank ports are available: that for CPU-Core, that for DMA channel 0, and that for DMA channel 1. Even during DMA transfer, therefore, a program can access any EXT-BUS address.

Transfer extending over multiple banks of EXT-BUS is not possible by performing DMA transfer once. Instead, DMA transfer must be performed for each bank.

9.6. BANK PORT SETTING (FLY-BY DMA)

For fly-by DMA transfer, no I/O address is required and, therefore, an address need not be set in a bank port. It is, however, necessary to specify in a bank port that fly-by DMA is to be performed. (See Sections 6.11 and 6.12.) Upon a reset, the bank port for each channel is cleared by hardware so that fly-by DMA can be performed. If, therefore, nothing has been written to a bank port since a reset, fly-by DMA transfer can be performed without setting the bank port.

9.7. WHEN USING MULTI

The ROM monitor for the Multi debugger uses NMIs for timer and communication interrupts. Therefore, each time an NMI is generated for a timer or communication interrupt, DMA stops. DMA is suspended while it is stopping, although the ROM monitor resumes DMA.

To perform DMA with critical timing, either stop the profile timer using the appropriate DIP switch (see Section 4.3.7) or mask NMIs temporarily during DMA transfer (see Section 6.7). Note, however, that while NMIs are being masked, no break is possible.

The profile function uses the profiler timer: An NMI interrupt is generated with this timer. By stopping the profile timer, therefore, an NMI caused by a timer interrupt is not generated.

Communication is used for an asynchronous break request from the debugger and for a report to the debugger of the occurrence of a break owing to a break point. This communication is handled with an NMI. When an NMI is disabled, therefore, a break is not possible.

10. CPU PIN CONNECTION

This chapter explains the uses of the CPU pins in the RTE-V850E/MS1-PC.

10.1. LIST

The table below lists the uses of the CPU pins. Details are given in the subsequent sections.

Pin name	Use	Section to be referenced
A0/PA0 to A7/PA7 A8/PB0 to A15/PB7 A16/P60 to A22/P66	Used as the address bus.	
A23/P67	Usable by the user.	10.10
ADV-/BCYST-/P94	Reserved by the system.	
ANI0/P70 to ANI7/P77 AVREF, AVSS, AVDD	Usable by the user.	10.9
CKSEL	Used as the CKSEL pin.	10.5
CS0-/RAS0-/P80	Used as the chip select for the CS0 space.	
CS1-/RAS1-/P81	Reserved by the system.	
CS2-/RAS2-/P82	Used as RAS- of EDO-DRAM.	
CS3-/RAS3-/P83	Used as RAS- of SIMM.	
CS4-/RAS4-/IOWR-/P84	Used as IOWR- if DMA is used. Usable by the user if DMA is not used.	10.11
CS5-/RAS5-/IORD-/P85	Used as IORD- if DMA is used. Usable by the user if DMA is not used.	10.11
CS6-/RAS6-/P86	Used as the chip select for the CS6 space.	
CS7-/RAS7-/P87	Used as the chip select for the CS7 space.	
D0/P40 to D7/P47 D8/P50 to D15/P57	Used as the data bus.	
HLDAK-/P96, HLDRQ-/P97	Usable by the user.	10.10
INTP100/DMARQ0-/P04 INTP101/DMARQ1-/P05	Used as DMARQ- if DMA of EXT-BUS is used. Usable by the user if DMA is not used.	10.11
INTP102/DMARQ2-/P06	The circuit is configured so that the pins can be used by the system. Currently, however, they are not used by the system and are usable by the user.	10.11
INTP103/DMARQ3-/P07	Usable by the user.	10.10
INTP110/DMAAK0-/P14 INTP111/DMAAK1-/P15	Used as DMAAK- if DMA of EXT-BUS is used. Usable by the user if DMA is not used.	10.12
INTP112/DMAAK2-/P16	The circuit is configured so that the pins can be used by the system. Currently, however, they are not used by the system and are usable by the user.	10.12
INTP113/DMAAK3-/P17	Usable by the user.	10.10
INTP130/P34	Used as INTP. If an interrupt is not used, usable by the user.	10.4
INTP131/SO2/P35 INTP132/SI2/P36, INTP133/SCK2-/P37	Used as INTP. If an interrupt is not used, usable by the user.	10.12
INTP140/P114, INTP141/SO3/P115	Used as INTP. If an interrupt is not used, usable by the user.	10.12
INTP142/SI3/P116, INTP143/SCK3-/P117	Usable by the user.	10.10
INTP150/P124 to INTP152/P126 INTP153/ADTRG/P127	Used as INTP if an EXT-BUS interrupt is used. Otherwise, usable by the user.	10.12
LCAS-/LWR-/P90 UCAS-/UWR-/P91 RD-/P92, WE-/P93, OE-/P95	Used as LCAS-/LWR-, UCAS-/UWR-, RD-, WE-, and OE-, respectively.	
MODE0 to MODE2	Used as the MODE pin.	10.5
MODE3/Vpp	Used as MODE3/Vpp.	10.6
NMI/P20	Used as NMI. If NMI is not used, usable by the user.	10.3
P21	Used to control the supply of VPP for self-writing to the built-in flash ROM of the CPU.	10.6
REFRQ-/PX5	Usable by the user.	10.10
RESET-	Used as RESET-.	10.2

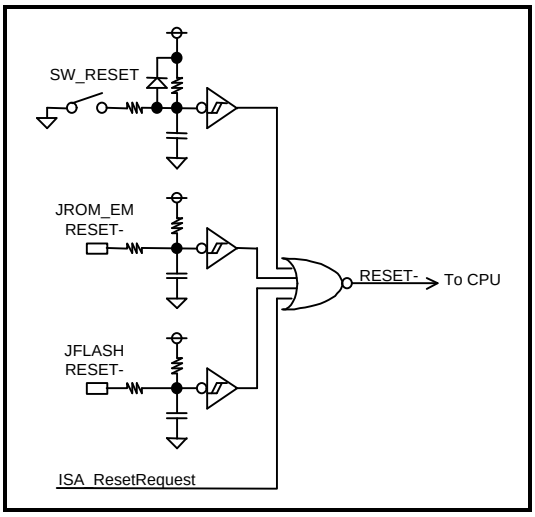
Pin name	Use	Section to be referenced
TO110/P00, TO101/P01 TCLR10/P02, TI10/P03	Usable by the user.	10.10
TO110/P10, TO111/P11 TCLR11/P12, TI11/P13	Usable by the user.	10.10
TO120/P100, TO121/P101 TCLR12/P102, TI12/P103 INTP120/TC0-/P104 to INTP123/TC3-/P107	Usable by the user.	10.10
TO130/P30, TO131/P31 TCLR13/P32, TI13/P33	Usable by the user.	10.10
TO140/P110, TO141/P111 TCLR14/P112, TI14/P113	Usable by the user.	10.10
TO150/P120, TO151/P121 TCLR15/P122, TI15/P123	Usable by the user.	10.10
TXD0/SO0/P22, TXD0/SI0/P23 SCK0-/P24, TXD1/SO1/P25 RXD1/SI1/P26, SCK1-/P27	Used for JFLASH and JRS232C. If JFLASH and JRS232C are not used, usable by the user.	10.8
WAIT-/PX6, CLKOUT/PX7	Used as WAIT- and CLKOUT, respectively.	
X1, X2	Used as X1 and X2 pins.	10.7
VDD	+3.3V	
VSS	GND	
HVDD	+5V	
HVSS	GND	
CVDD	+3.3V	
CVSS	GND	

10.2. RESET-

The factors listed below trigger a CPU reset. These factors reset the CPU. They also system-reset the board control circuit.

- **Power-on reset:** Occurs when the power to the board is switched on.
- **Reset request received from JROM_EM:** Input to the RESET- pin of the JROM_EM connector. (See Section 3.5.7.)
- **Reset by the SW_RESET:** Generated by the reset switch (SW_RESET) on the rear panel. (See Section 3.3.1.)
- **Reset request from JFLASH:** Sent from the JFLASH connector provided to enable writing to the built-in flash ROM of the CPU using a writer. (See Section 3.5.11 and Chapter 11.)
- **Reset request from the host:** Sent via the ISA bus.

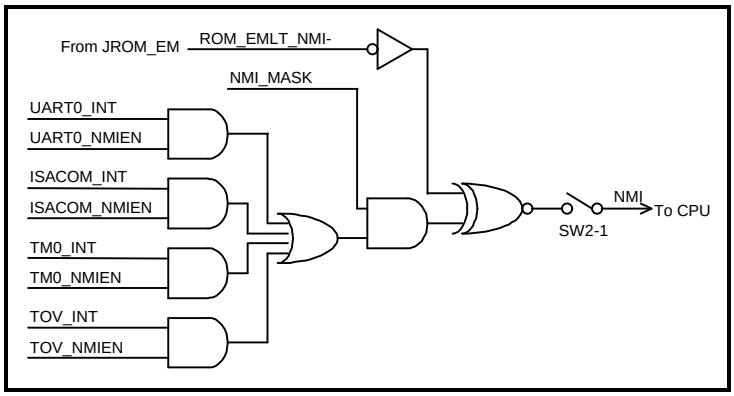
The figure below outlines the reset signal generation logic.



10.3. NMI

NMI is connected to the result of combining multiple interrupts by hardware. The following interrupts can be combined. For an explanation of interrupt selection, see Sections 6.8 and 6.9.

- **UART0_INT:** Interrupt issued by UART0 of the TL16PIR552. (See Section 6.2.)
- **ISACOM_INT:** Interrupt based on communication control with ISA bus. This interrupt is reserved by the system.
- **TM0_INT:** Interrupt issued by CH#0 of the TIC (μPD71054). (See Sections 6.3 and 6.7.)
- **TOV_INT:** Interrupt resulting from time-over ready occurrence. (See Section 12.1.)
- **ROM_EMLT_NMI-:** NMI request from the ROM emulator that is supplied from the JROM_EM connector. (See Section 3.5.7.)



The following procedure must be applied if an NMI occurs.

<1> Mask the NMI by means of hardware by setting NMI_MASK of the control port to "1". (In case multiple interrupts occurred, an edge is generated by masking.)

<2> Determine the NMI request source. This can be identified at the NMI/INTP130 status port. (See Section 6.9.)

<3> Clear the request by performing interrupt handling for the request source.

<4> Reset the mask by setting NMI_MASK of the control port to "0".

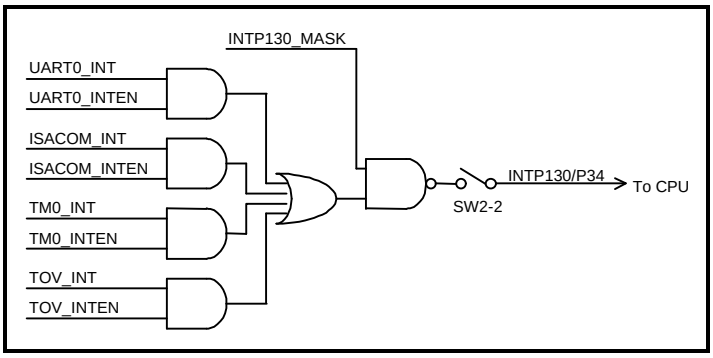
<5> Return from interrupt handling.

To enable the use of the ROM monitor for the Multi debugger, set contact 1 of SW2 to ON to enable the use of NMI.
This is also true when the debugger is to be used for the ROM emulator.

10.4. INTP130/P34

INTP130/P34 is connected to the result of combining multiple interrupts by hardware. The following interrupts can be combined. For an explanation of interrupt selection, see Sections 6.8 and 6.9.

- **UART0_INT:** Interrupt issued by UART0 of the TL16PIR552. (See Section 6.2.)
- **ISACOM_INT:** Interrupt based on communication control with ISA bus. This interrupt is reserved by the system.
- **TM0_INT:** Interrupt issued by CH#0 of the TIC (μPD71054). (See Sections 6.3 and 6.7.)
- **TOV_INT:** Interrupt resulting from time-over ready occurrence. (See Section 12.1.)



The following procedure must be applied when an INTP130/P34 occurs.

<1> Mask the INTP130/P34 by means of hardware by setting INTP130_MASK of the control port to "1". (In case that multiple interrupts occurred, an edge is generated by masking.)

<2> Determine the INTP130/P34 request source. This can be identified at the NMI/INTP130 status port. (See Section 6.9.)

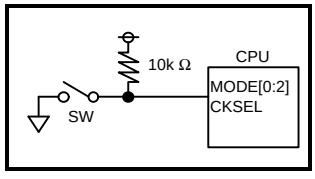
<3> Clear the request by performing interrupt handling for the request source.

<4> Reset the mask by setting INTP130_MASK of the control port to "0".

<5> Return from interrupt handling.

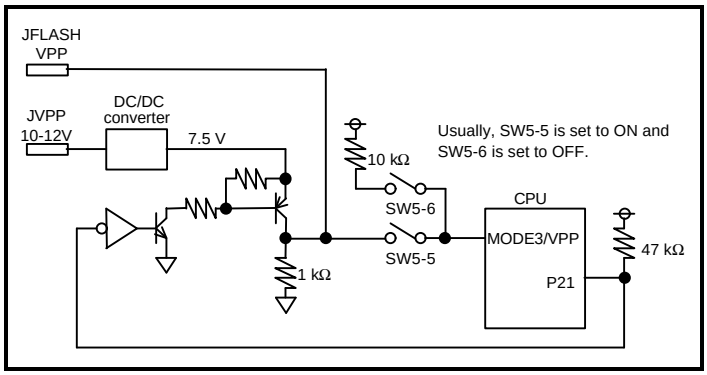
10.5. MODE0 TO MODE2 AND CKSEL

The levels of MODE0 to MODE2 and CKSEL can be changed using the corresponding contacts of DIP switch 5. (See Section 3.5.1.)



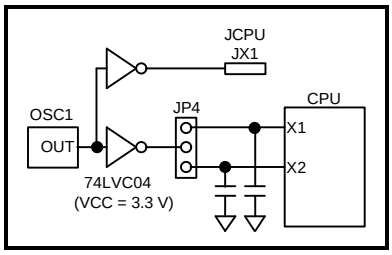
10.6. MODE3/Vpp AND P21

MODE3/Vpp is usually used at the Low level. It is used as Vpp to enable writing to the built-in flash ROM of the CPU. Vpp can be supplied from either of two sources: the JFLASH connector (see Section 3.5.11.) or the JVPP connector (see Section 3.5.6.).



10.7. X1 AND X2

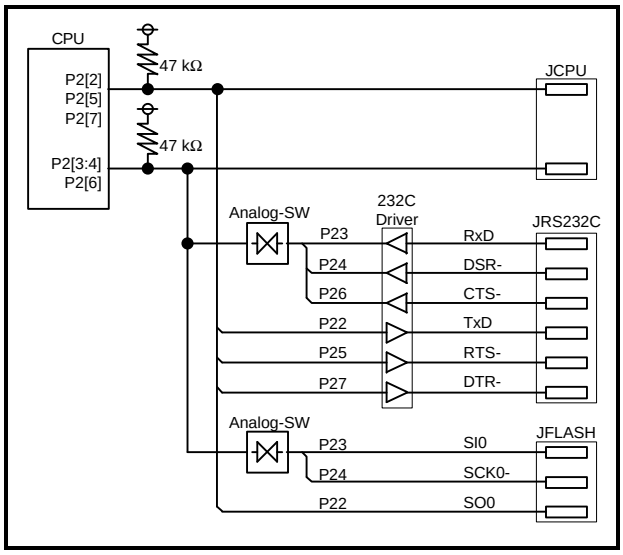
By mounting a crystal oscillator on JP4 and jumpering the middle pin of JP4 and the "X1" pin, the oscillator mounted on the OSC1 socket can be connected to pins X1 and X2. A 5V oscillator must be mounted in the OSC1 socket. (See Sections 3.5.2 and 3.5.3.)



10.8. P22 TO P27

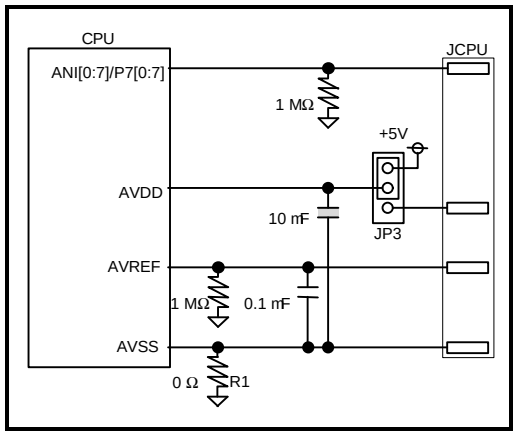
P22 to P27 can be used for serial communication via the JRS232C connector by using the built-in serial interface of the CPU. (See Section 3.5.10.) They are also used to write to the built-in flash ROM of the CPU using a writer. (See Chapter 11.) The use of the pins is determined by the combination of contacts 7 and 8 of DIP switch SW5.

SW5-7	SW5-8	Use of P22 to P27
OFF	OFF	Used by the user via the JCPU connector.
OFF	ON	Used by the user for serial communication via the JRS232C connector.
ON	----	Connected to the flash ROM writer via the JFLASH connector.



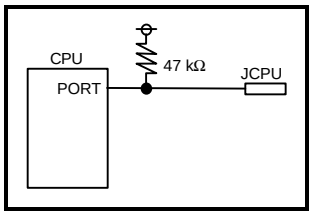
10.9. ANI0/P70 TO ANI7/P77, AVDD, AVSS, AND AVREF

ANI0/P70 to ANI7/P77, AVDD, AVSS, and AVREF can be used by the user via the JCPU connector.



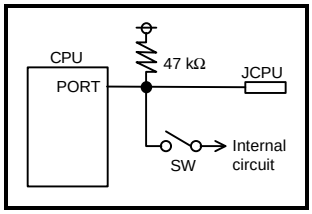
10.10. PORT (TYPE 1)

A port of this type can be used freely by the user via the JCPU connector.



10.11. PORT (TYPE 2)

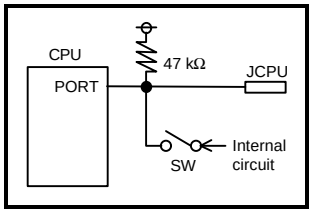
A port of this type is assigned a function within the board. If the function is not used, the port can be used freely by the user via the JCPU connector.



Pin name	Switch No.	Internal function	Section to reference
CS4-/RAS4-/IOWR-/P84	SW1-8	IOWR-	3.1.1
CS5-/RAS5-/IORD-/P85	SW1-7	IORD-	3.1.1
INTP110/DMAAK0-/P14	SW1-2	EXT-BUS DMAAK0-	3.1.1
INTP111/DMAAK1-/P15	SW1-4	EXT-BUS DMAAK1-	3.1.1
INTP112/DMAAK2-/P16	SW1-6	Reserved by the system.	3.1.1

10.12. PORT (TYPE 3)

A port of this type is assigned a function within the board. If the function is not used, the port can be used freely by the user via the JCPU connector.



Pin name	Switch No.	Internal function	Section to reference
INTP100/DMARQ0-/P04	SW1-1	EXT-BUS DMARQ0-	3.1.1
INTP101/DMARQ1-/P05	SW1-3	EXT-BUS DMAAK0-	3.1.1
INTP102/DMARQ2-/P06	SW1-5	Reserved by the system.	3.1.1
INTP131/SO2/P34	SW2-3	UART0 interrupt	3.1.2
INTP132/SI2/P36	SW2-4	UART1 interrupt	3.1.2
INTP133/SCK2-/P37	SW2-5	PRINTER interrupt	3.1.2
INTP140/P114	SW2-6	TIC CH#1 interrupt	3.1.2
INTP141/SO3/P115	SW2-7	Reserved by the system.	3.1.2
INTP150/P124	SW3-1	EXT-BUS INT0-	3.1.3
INTP151/P125	SW3-2	EXT-BUS INT1-	3.1.3
INTP152/P126	SW3-3	EXT-BUS INT2-	3.1.3
INTP153/P127	SW3-4	EXT-BUS INT3-	3.1.3

11. WRITING TO THE BUILT-IN FLASH ROM OF THE CPU

Writing to the built-in flash ROM of the CPU is possible using the dedicated writer, via the JFLASH connector. This chapter explains how to write to the built-in flash ROM of the CPU. (See Section 3.5.11.)

11.1. SWITCH SETTING

To enable writing to the built-in flash ROM of the CPU, the appropriate switch on the board must be set as follows:

Switch contact	Position	Note
SW5-1	OFF	Used to set the MODE0 to MODE3 pins of the CPU to flash memory programming mode.
SW5-2	OFF	
SW5-3	ON	
SW5-4	ON/OFF	Used to set CKSEL according to the clock supply state (OFF: direct mode, ON: PLL mode).
SW5-5	ON	Used to connect MODE3/VPP to the JFLASH connector.
SW5-6	OFF	
SW5-8	ON	Used to enable the JFLASH connector.

11.2. COMMUNICATION MODES

Communication between the writer and the CPU can be performed in either of two modes. The mode to be used is determined by the writer.

Communication mode
CMOS-level asynchronous serial (UART)
CMOS-level asynchronous serial (CSI)

11.3. WRITING PROCEDURE

The procedure for writing to the flash ROM is described below:

1. Turn off the power to the RTE-V850E/MS1-PC.
2. Set DIP switch SW5.
3. Connect the writer to the JFLASH connector.
4. Check that the writer is turned on. Turn on the power to the RTE-V850E/MS1-PC. The JFLASH-LED (green LED) on the socket board lights. If the writer has an LED indicating whether the target is turned on, check that the LED lights.
5. Operate the writer to perform writing.
6. After the completion of writing, turn off the power to the RTE-V850E/MS1-PC.
7. Disconnect the writer from the JFLASH connector. Set DIP switch SW5 appropriately. (To run the program in the built-in flash ROM of the CPU, the CPU must be placed in single-chip mode.)
8. Turn on the power to the RTE-V850E/MS1-PC.

11.4. CAUTION

- When data is written to or erased from the flash ROM, a high voltage (7.5 V) is applied. Caution must, therefore, be exercised during this procedure.

12. BUS CYCLE

12.1. TIME-OVER READY

On this board, when an external bus cycle (including a fly-by DMA cycle) generated for the CS6 space by the CPU has not been completed within a certain period, the time-over ready status is generated to forcibly complete the cycle. Time-over ready is generated after a bus cycle continues for 10 ms (with an error of about ±10%).

If time-over ready occurs, the TOVER-LED on the board lights and NMI and INTP130 interrupt factors are generated. By writing to the TOV_INT_CLR- control port, the TOVER-LED goes out, and the NMI and INTP130 interrupt factors are cleared. See Sections 5.2, 6.7, 6.8, 6.9, 10.3, and 10.4.

12.2. SIMM INTERFACE

12.2.1. Outline

CPU signals are not directly supplied as control signals for the SIMM, but they are processed within the board before being supplied. The following sections provide an overview of how the signals are processed.

12.2.2. Signal Description

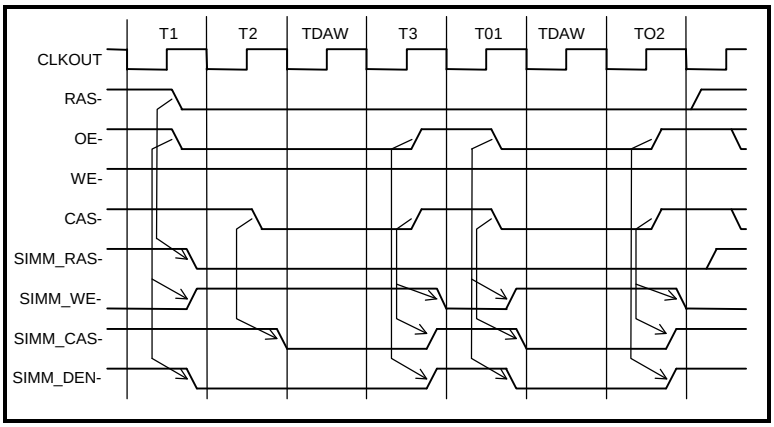
The signals used for waveforms described in this chapter are defined as follows:

CLKOUT	:	System clock output by the CPU.
RAS-	:	RAS- signal output by the CPU.
OE-	:	OE- signal output by the CPU.
WE-	:	WE- signal output by the CPU.
CAS-	:	CAS- signal output by the CPU.
SIMM_RAS-	:	RAS- signal input to SIMM.
SIMM_WE-	:	WE- signal input to SIMM.
SIMM_CAS-	:	CAS- signal input to SIMM.
SIMM_DEN-	:	Signal for opening and closing the data bus between the CPU and SIMM. The data bus between the CPU and SIMM is closed when the signal is Low.

12.2.3. Read Cycle

The figure below shows a read cycle.

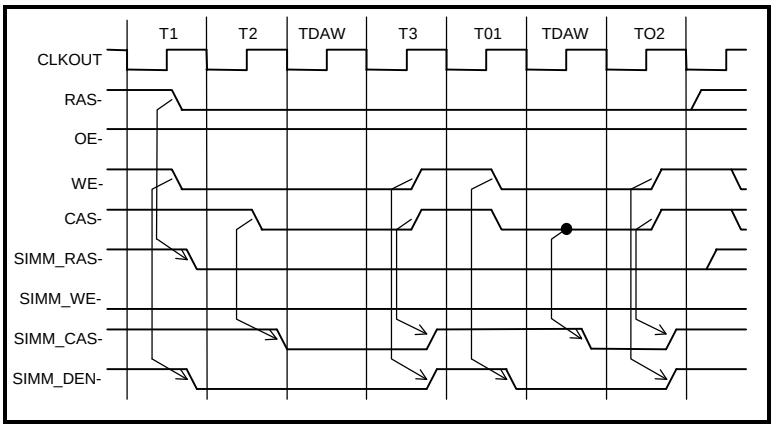
- The WE- signal for the SIMM is kept active except in refresh and read cycles.
- The RAS- signal for the SIMM is switched when a different bank of SIMM bank is accessed.
- At least one TDAW cycle is required in a write cycle.
- Regardless of whether the SIMM is of the Fast-Page or EDO type, the relationship between the CPU signals and the signals sent to the SIMM are the same; only the waveforms of the CPU signals differ.



12.2.4. Write Cycle

The figure below shows a write cycle.

- The WE- signal for the SIMM is kept active except in refresh and read cycles.
- The RAS- signal for the SIMM is switched when a different SIMM bank is accessed.
- When the WE- signal of the CPU and the CAS- signal of the CPU fall at the same time, the CAS- signal for the SIMM becomes one pulse later. Thus, at least one TDAW cycle is required.
- Regardless of whether the SIMM is of the Fast-Page or EDO type, the relationship between the CPU signals and the signals sent to the SIMM are the same; only the waveforms of the CPU signals differ.



- Memo -