

# M306NKT-EPB

User's Manual

Emulation Probe for M16C/6N Group M16C/6N4, 6N5, 6NK, 6NL, 6NM, 6NN

User's Manual

Rev.2.00  
Oct. 16, 2006

Renesas Technology  
[www.renesas.com](http://www.renesas.com)

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of non-flammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein. The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors. Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (<http://www.renesas.com>).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination. Any diversion or reexport contrary to the export control laws and regulations of Japan and/ or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

## Preface

The M306NKT-EPB is an emulation probe for The M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN. The M306NKT-EPB is used by connecting to the PC7501 emulator main unit.

This user's manual mainly describes specifications of the M306NKT-EPB emulation probe and how to setup it. For details on the emulator main unit and emulator debugger which are used with the M306NKT-EPB, refer to each product's user's manual.

All the components of this product are shown in "1.1 Package Components" (page 13). If there is any question or doubt about this product, contact your local distributor.

The related manuals for using this product are listed below. You can download the latest manuals from the Renesas Tools homepage (<http://www.renesas.com/en/tools>).

### Related manuals

Item	Manual
Emulator main unit	PC7501 User's Manual
Integrated development environment	High-performance Embedded Workshop User's Manual
Emulator debugger	M16C R8C PC7501 Emulator Debugger User's Manual
	M3T-PD30F User's Manual
C compiler	C compiler package for R8C/Tiny, M16C/60, 30, Tiny, 20 and 10 Series C Compiler User's Manual
Assembler	C compiler package for R8C/Tiny, M16C/60, 30, Tiny, 20 and 10 Series Assembler User's Manual

## Important

Before using this product, be sure to read this user's manual carefully.

Keep this user's manual, and refer to this when you have questions about this product.

### Emulator:

The emulator in this document refers to the following products that are manufactured by Renesas Technology Corp.:

- (1) PC7501 main unit
- (2) Emulation probe
- (3) Package converter board for connecting the user system

The emulator herein does not include the customer's user system and host machine.

### Purpose of use of the emulator:

This emulator is a device to support the development of a system that uses the M16C Family M16C/60 Series M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN of Renesas 16-bit single-chip MCUs. It provides support for system development in both software and hardware.

Be sure to use this emulator correctly according to said purpose of use. Please avoid using this emulator for other than its intended purpose of use.

### For those who use this emulator:

This emulator can only be used by those who have carefully read the user's manual and know how to use it.

Use of this emulator requires the basic knowledge of electric circuits, logical circuits, and MCUs.

### When using the emulator:

- (1) This product is a development supporting unit for use in your program development and evaluation stages. In mass-producing your program you have finished developing, be sure to make a judgment on your own risk that it can be put to practical use by performing integration test, evaluation, or some experiment else.
- (2) In no event shall Renesas Solutions Corp. be liable for any consequence arising from the use of this product.
- (3) Renesas Solutions Corp. strives to renovate or provide a workaround for product malfunction at some charge or without charge. However, this does not necessarily mean that Renesas Solutions Corp. guarantees the renovation or the provision under any circumstances.
- (4) This product has been developed by assuming its use for program development and evaluation in laboratories. Therefore, it does not fall under the application of Electrical Appliance and Material Safety Law and protection against electromagnetic interference when used in Japan.
- (5) Renesas Solutions Corp. cannot predict all possible situations or possible cases of misuse where a potential danger exists. Therefore, the warnings written in this user's manual and the warning labels attached to this emulator do not necessarily cover all of such possible situations or cases. Please be sure to use this emulator correctly and safely on your own responsibility.
- (6) This product is not qualified under UL or other safety standards and IEC or other industry standards. This fact must be taken into account when taking this product from Japan to some other country.

**Usage restrictions:**

This emulator has been developed as a means of supporting system development by users. Therefore, do not use it as a device used for equipment-embedded applications. Also, do not use it for developing the systems or equipment used for the following purposes either:

- (1) Transportation and vehicular
- (2) Medical (equipment where human life is concerned)
- (3) Aerospace
- (4) Nuclear power control
- (5) Undersea repeater

If you are considering the use of this emulator for one of the above purposes, please be sure to consult your local distributor.

**About product changes:**

We are constantly making efforts to improve the design and performance of this emulator. Therefore, the specification or design of this emulator or its user's manual may be changed without prior notice.

**About the rights:**

- (1) We assume no responsibility for any damage or infringement on patent rights or any other rights arising from the use of any information, products or circuits presented in this user's manual.
- (2) The information or data in this user's manual does not implicitly or otherwise grant a license for patent rights or any other rights belonging to us or third parties.
- (3) This user's manual and this emulator are copyrighted, with all rights reserved by us. This user's manual may not be copied, duplicated or reproduced, in whole or part, without prior written consent of us.

**About diagrams:**

The diagrams in this user's manual may not all represent exactly the actual object.

## Precautions for Safety

### Definitions of Signal Words

In both the user's manual and on the product itself, several icons are used to insure proper handling of this product and also to prevent injuries to you or other persons, or damage to your properties.

This chapter describes the precautions which should be taken in order to use this product safely and properly. Be sure to read this chapter before using this product.



This symbol represents a warning about safety. It is used to arouse caution about a potential danger that will possibly inflict an injury on persons. To avoid a possible injury or death, please be sure to observe the safety message that follows this symbol.



**DANGER** indicates an imminently dangerous situation that will cause death or heavy wound unless it is avoided. However, there are no instances of such danger for the product presented in this user's manual.



**WARNING** indicates a potentially dangerous situation that will cause death or heavy wound unless it is avoided.



**CAUTION** indicates a potentially dangerous situation that will cause a slight injury or a medium-degree injury unless it is avoided.



**CAUTION** with no safety warning symbols attached indicates a potentially dangerous situation that will cause property damage unless it is avoided.



This is used in operation procedures or explanatory descriptions to convey exceptional conditions or cautions to the user.

In addition to the five above, the following are also used as appropriate.

△ means WARNING or CAUTION.

Example:



**CAUTION AGAINST AN ELECTRIC SHOCK**

⊘ means PROHIBITION.

Example:



**DISASSEMBLY PROHIBITED**

● means A FORCIBLE ACTION.

Example:



**UNPLUG THE POWER CABLE FROM THE RECEPTACLE.**

## ⚠ WARNING

### Warnings for AC Power Supply:



- If the attached AC power cable does not fit the receptacle, do not alter the AC power cable and do not plug it forcibly. Failure to comply may cause electric shock and/or fire.

- Use an AC power cable which complies with the safety standard of the country.

- Do not touch the plug of the AC power cable when your hands are wet. This may cause electric shock.

- This product is connected signal ground with frame ground. If your developing product is transformless (not having isolation transformer of AC power), this may cause electric shock. Also, this may give an unreparable damage to this product and your developing one.

While developing, connect AC power of the product to commercial power through isolation transformer in order to avoid these dangers.

- If other equipment is connected to the same branch circuit, care should be taken not to overload the circuit.



- When installing this equipment, insure that a reliable ground connection is maintained.



- If you smell a strange odor, hear an unusual sound, or see smoke coming from this product, then disconnect power immediately by unplugging the AC power cable from the outlet.

Do not use this as it is because of the danger of electric shock and/or fire. In this case, contact your local distributor.

- Before setting up this emulator and connecting it to other devices, turn off power or remove a power cable to prevent injury or product damage.

### Warnings to Be Taken for This Product:



- Do not disassemble or modify this product. Personal injury due to electric shock may occur if this product is disassembled and modified. Disassembling and modifying the product will void your warranty.

- Make sure nothing falls into the cooling fan on the top panel, especially liquids, metal objects, or anything combustible.

### Warning for Installation:



- Do not set this product in water or areas of high humidity. Make sure that the product does not get wet. Spilling water or some other liquid into the product may cause unreparable damage.

### Warning for Use Environment:



- This equipment is to be used in an environment with a maximum ambient temperature of 35°C. Care should be taken that this temperature is not exceeded.

 **CAUTION****Cautions to Be Taken for Turning On the Power:**

- Turn ON/OFF the power of the emulator and user system as simultaneously as possible.
- Do not leave either the emulator or user system powered on, because of leakage current the internal circuits may be damaged.
- When turning on the power again after shutting off the power, wait about 10 seconds.

**Cautions to Be Taken for Handling This Product:**

- Use caution when handling the main unit. Be careful not to apply a mechanical shock.
- Do not touch the connector pins of the emulator main unit and the target MCU connector pins directly. Static electricity may damage the internal circuits.
- Do not pull this emulator by the communications interface cable or the flexible cable. And, excessive flexing or force may break conductors.
- Do not flex the flexible cable for connecting to the emulator main unit and emulation probe excessively. The cable may cause a break.
- Do not use inch-size screws for this equipment. The screws used in this equipment are all ISO (metric-size) type screws. When replacing screws, use same type screws as equipped before.

**Caution to Be Taken for System Malfunctions:**

- If the emulator malfunctions because of interference like external noise, do the following to remedy the trouble.
  - (1) Press the RESET button on the PC7501's front panel.
  - (2) If normal operation is not restored after step (1), shut OFF the emulator once and then reactivate it.

**Caution to Be Taken for External Trigger Input Voltage:**

- Keep an external trigger input voltage of the emulator main unit less than a Vcc2 voltage. Overvoltage may cause damage to the internal circuits.



# Contents

	Page
Preface.....	3
Important.....	4
Precautions for Safety .....	6
Contents.....	9
User Registration .....	11
Terminology .....	12
1. Outline.....	13
1.1 Package Components .....	13
1.2 Other Tool Products Required for Development .....	14
1.3 System Configuration .....	15
1.3.1 System Configuration .....	15
1.3.2 Names and Functions of the PC7501 Upper Panel LEDs .....	16
1.4 Specifications .....	18
1.5 Operating Environment.....	20
2. Setup.....	21
2.1 Flowchart of Starting Up the Emulator .....	21
2.2 Installing the Emulator Debugger (M16C R8C PC7501 Emulator Debugger) .....	22
2.2.1 Installing the Emulator Debugger.....	22
2.3 Connecting the Host Machine .....	23
2.4 Connecting the PC7501 .....	24
2.5 Connecting the Power Supply for the Emulator .....	25
2.6 Turning ON the Power.....	26
2.6.1 Checking the Connections of the Emulator System.....	26
2.6.2 Turning ON/OFF the Power .....	26
2.6.3 LED Display When the Emulator Starts Up Normally .....	27
2.7 Downloading Firmware.....	28
2.7.1 When It is Necessary to Download Firmware .....	28
2.7.2 Downloading Firmware in Maintenance Mode.....	28
2.8 Self-check .....	29
2.8.1 Self-check Procedure .....	29
2.8.2 If an Error is Detected in the Self-check.....	30
2.9 Connecting the User System.....	31
2.9.1 Connecting to a 100-pin LCC Socket.....	33
2.9.2 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 1).....	34
2.9.3 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 2).....	35
2.9.4 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 3).....	36
2.9.5 Connecting to a 100-pin 0.5mm Pitch Foot Pattern (Part 1).....	37
2.9.6 Connecting to a 100-pin 0.5mm pitch Foot Pattern (Part 2) .....	38
2.9.7 Connecting to a 128-pin 0.5mm pitch Foot Pattern .....	39
2.10 Switch Settings .....	40
2.10.1 A/D Conversion Bypass Capacitors .....	40
2.10.2 Pullup of Emulation Ports.....	41
2.11 Selecting Clock Supply.....	42
2.11.1 Clocks.....	42
2.11.2 Using an Internal Oscillator Circuit Board .....	42
2.11.3 Using the Oscillator Circuit on the User System .....	46
2.11.4 Using the Internal Generator Circuit.....	46

	Page
3. Usage (How to Use the Emulator Debugger).....	47
3.1 Starting Up the Emulator Debugger .....	47
3.1.1 INIT Dialog Box .....	47
3.1.2 MCU Setting Dialog Box .....	53
3.1.3 Checking Connections of the Emulator System.....	56
3.2 Editor (Source) Window.....	57
3.3 Program Execution .....	58
3.3 Hardware Breakpoint Setting Window.....	62
3.5 Trace Window.....	66
3.6 RAM Monitor Window.....	70
4. Hardware Specifications .....	71
4.1 Target MCU Specifications.....	71
4.2 Access Timing .....	72
4.2.1 Operation Timing of Memory Expansion Mode and Microprocessor Mode (Vcc1=5V).....	72
4.2.2 Operation Timing of Memory Expansion Mode and Microprocessor Mode (Vcc1=3V).....	77
4.3 Differences between the Actual MCU and Emulator.....	82
4.4 Connection Diagrams.....	85
4.5 External Dimensions.....	87
4.5.1 External Dimensions of the Emulation Probe .....	87
4.5.2 External Dimensions of the M30800T-PTC .....	88
4.5.3 External Dimensions of the Converter Board M3T-F160-100NRB .....	89
4.5.4 External Dimensions of the Converter Board M3T-F160-100NSD .....	89
4.5.5 External Dimensions of the Converter Board M3T-F160-128NRD .....	90
4.6 Notes on Using This Product.....	91
5. Troubleshooting .....	96
5.1 Flowchart to Remedy the Troubles .....	96
5.2 When the Emulator Debugger Does Not Start Up Properly .....	97
5.3 How to Request for Support .....	100
6. Maintenance and Guarantee .....	101
6.1 User Registration.....	101
6.2 Maintenance .....	101
6.3 Guarantee.....	101
6.4 Repair Provisions .....	101
6.5 How to Make Request for Repair .....	102

## User Registration

When you have purchased the emulator presented in this user's manual, please be sure to register it. As the H/W Tool Customer Registration Sheet is included with this manual, fill it in and FAX it to your local distributor or email the same contents to the following address. If you register it by email, you can use a text format for user registration created when installing the software in the following folder. Your registered information is used for only after-sale services, and not for any other purposes. Without user registration, you will not be able to receive maintenance services such as a notification of field changes or trouble information. So be sure to carry out the user registration.

For more information about user registration, please email to the following address.

[regist\\_tool@renesas.com](mailto:regist_tool@renesas.com)

## Terminology

Some specific words used in this user's manual are defined as follows:

### **Emulator system**

This means an emulator system built around the PC7501 emulator. The PC7501 emulator system is configured with an emulator main unit, emulation probe, host machine and the integrated development environment High-performance Embedded Workshop.

### **Emulator main unit (Hereafter PC7501)**

This means an emulator main unit for M16C Family.

### **Emulation probe**

This means the emulation probe (this product) for the M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN MCUs.

### **Emulator debugger (M16C R8C PC7501 Emulator Debugger)**

This means a software tool starting up from the High-performance Embedded Workshop to control the PC7501 and this product and enable debugging.

### **Renesas Integrated development environment High-performance Embedded Workshop**

This tool provides powerful support for the development of embedded applications for Renesas microcomputers. It has an emulator debugger function allowing for the emulator to be controlled from the host machine via an interface. Furthermore, it permits a range of operations from editing a project to building and debugging it to be performed within the same application. What's more, it supports version management.

### **Firmware**

Program that analyzes contents of communication with the emulator debugger and controls the emulator hardware. This program is installed in the flash memory in the emulator main unit. This program is downloadable from the emulator debugger to upgrade the firmware or to support other MCUs.

### **Host machine**

This means a personal computer used to control the PC7501 and emulation probe.

### **Software break**

A software break is a function to break the program before the system executes an instruction at the specified address. The instruction at the preset address will not be executed.

### **Hardware break**

A hardware break is a function to break the program when the system detects a write/read of data to/from memory or a leading/trailing edge of the signal entered from the external trace cable. The former break function is called address break; and the latter break function is called trigger break. While the instruction at the address where the software break is set is not executed, a hardware break is performed after the specified instruction is executed.

### **Target MCU**

This means the MCU you are going to debug.

### **User system**

This means a user's application system using the microcomputer to be debugged.

### **User program**

This means the program you are going to debug.

### **Evaluation MCU**

This means the MCU mounted on the emulation probe which is operated in the specific mode for tools.

### **#**

In this user's manual, this symbol is used to show active Low. (e.g. RESET#: Reset signal)

## 1. Outline

This chapter describes the package components, the system configuration, the specifications of the emulator functions and the operating environment.

### 1.1 Package Components

The M306NKT-EPB package consists of the following items. When unpacking it, check to see if your M306NKT-EPB contains all of these items.

Table 1.1 Package components

Item	Quantity
M306NKT-EPB emulation probe	1
M30800T-PTC (pre-mounted) converter board for 100-pin 0.65mm pitch LCC	1
IC61-1004-051 100-pin 0.65mm pitch LCC socket made by Yamaichi Electronics Co., Ltd.	1
OSC-3 (16MHz) oscillator circuit board	1
OSC-2 oscillator circuit bare board	1
Network registers for ports P0--P5 pullup (51kΩ x 8)	6
H/W Tool Customer Registration Sheet (English)	1
H/W Tool Customer Registration Sheet (Japanese)	1
Repair Request Sheet (English)	1
Repair Request Sheet (Japanese)	1
M306NKT-EPB User's Manual (this manual)	1
M306NKT-EPB User's Manual (Japanese)	1
M306NKT-EPB Supplementary Document (English)	1
M306NKT-EPB Supplementary Document (Japanese)	1

\* Please keep the M306NKT-EPB's packing box and cushion material in your place for reuse at a later time when sending your product for repair or other purposes. Always use these packing box and cushion material when transporting this product.

\* If there is any question or doubt about the packaged product, contact your local distributor.

\* For purchasing the IC61-1004-051 or for technical information, contact Yamaichi Electronics Co., Ltd.

## 1.2 Other Tool Products Required for Development

To bring forward programs development on an M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN MCUs, the products listed below are necessary in addition to those contained package. Get them separately.

Table 1.2 Other tool products required for development

Product		Product name	Notes	
Emulator main unit		PC7501	-	
Emulator debugger		M16C R8C PC7501 Emulator debugger	Included with the PC7501	
		M3T-PD30F		
Converter boards	M16C/6N4 M16C/6N5	100-pin 0.65mm pitch QFP (PRQP0100JB-A, Previous code: 100P6S-A)	M3T-F160-100NRB	
			M3T-100LCC-DMS +	
			M3T-DIRECT100S	
			M3T-100LCC-DMS +	
		M3T-DUMMY100S		
		100-pin 0.5mm pitch LQFP (PLQP0100KB-A, Previous code: 100P6Q-A)	M3T-F160-100NSD	Required according to a foot pattern of a user system Refer to “2.9 Connecting the User System” (page 31)
			M3T-100LCC-QSD	
M16C/6NK M16C/6NL M16C/6NM M16C/6NN	100-pin 0.5mm pitch LQFP (PLQP0100KB-A, Previous code: 100P6Q-A)	M3T-F160-100NSD		
		M3T-100LCC-QSD		
	128-pin 0.5mm pitch LQFP (PLQP0128KB-A, Previous code: 128P6Q-A)	M3T-F160-128NRD		

\* For purchasing these products, contact your local distributor.

## 1.3 System Configuration

### 1.3.1 System Configuration

Figure 1.1 shows a configuration of the M306NKT-EPB system.

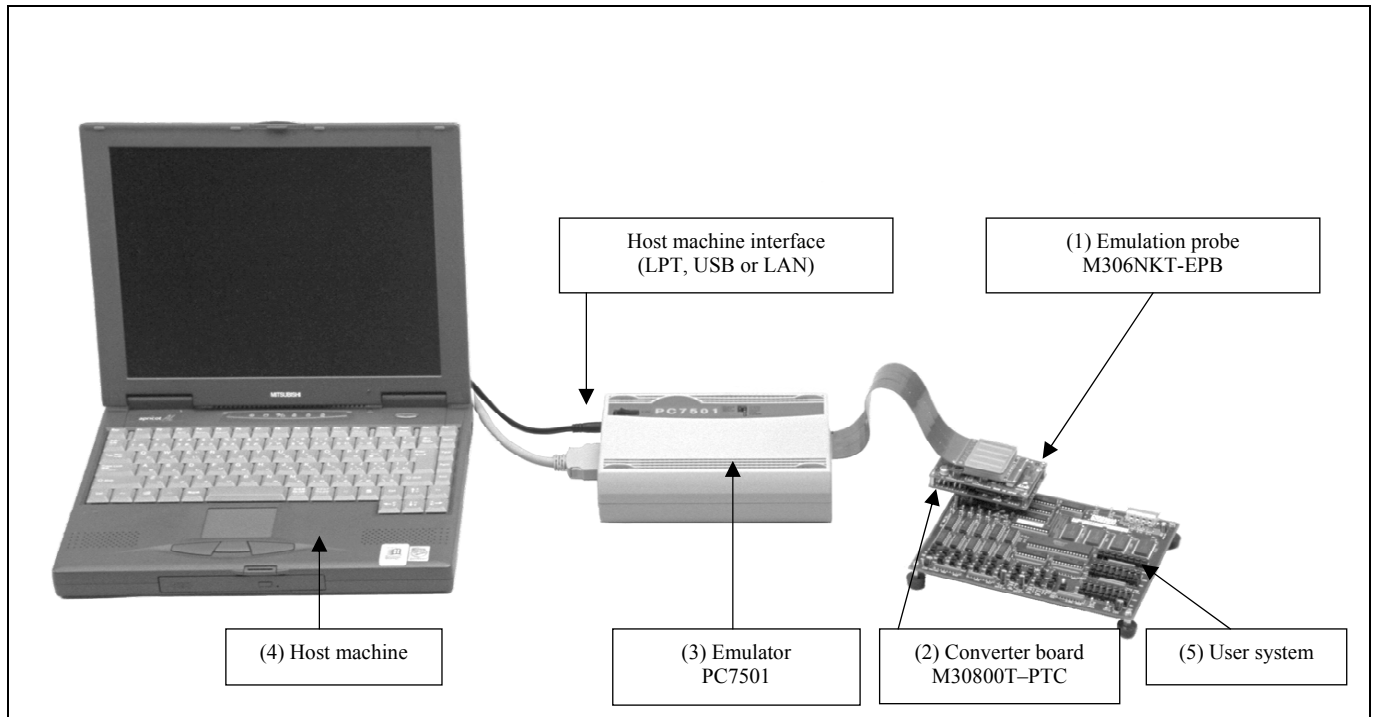


Figure 1.1 System configuration

(1) Emulation probe M306NKT-EPB (this product)

This is an emulation probe for the M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN and contains an evaluation MCU.

(2) Pitch converter board M30800T-PTC (included)

This is a pitch converter board for connecting to 100-pin 0.65mm pitch LCC socket IC61-1004-051 on the user system. For details, refer to "2.9 Connecting the User System" (page 31).

(3) Emulator main unit PC7501

The emulator main unit for the M16C Family to be used with this product.

(4) Host machine

A personal computer to control the emulator.

(5) User system

This is your application system.

This emulator cannot supply the power to the user system. Therefore design your system so that the user system is powered separately.

1.3.2 Names and Functions of the PC7501 Upper Panel LEDs

Figure 1.2 shows the names of the LEDs on the upper panel of the emulator.

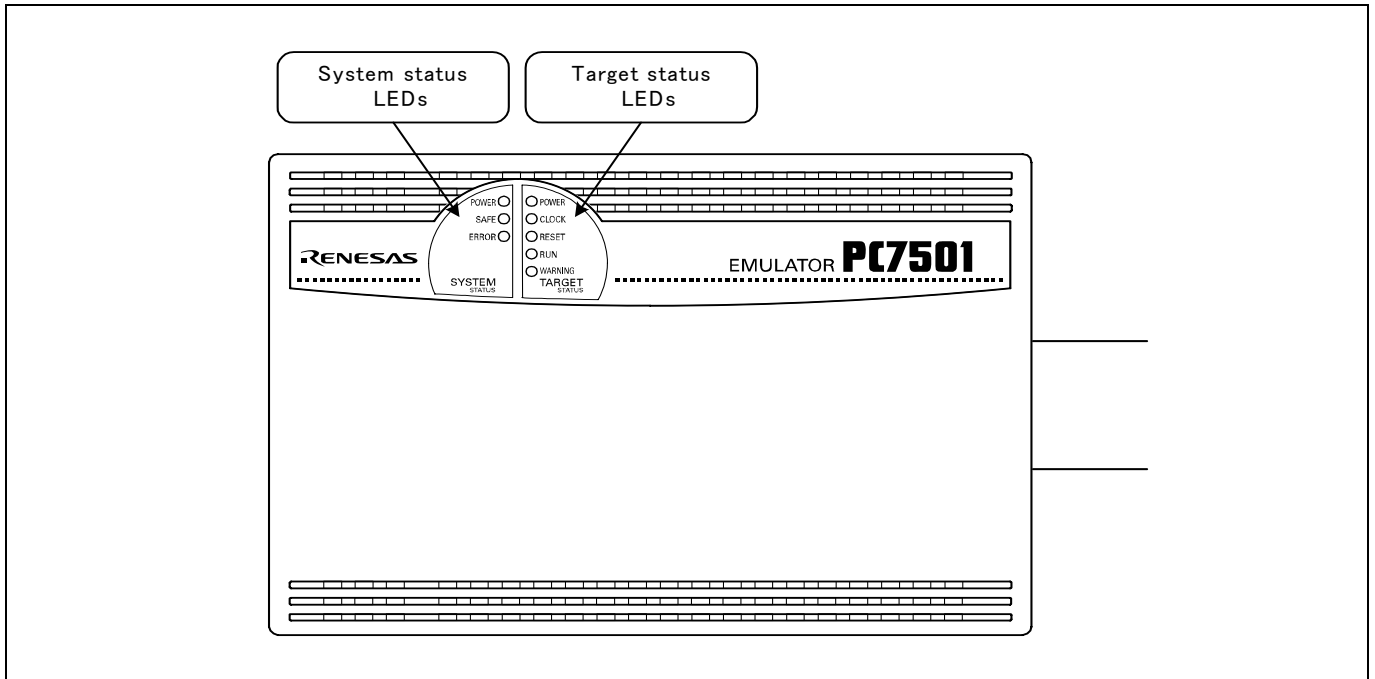


Figure 1.2 Names of the LEDs on the upper panel of the PC7501

(1) System Status LEDs

The system status LEDs indicate the emulator PC7501's power supply, firmware operating status, etc. Table 1.3 lists the definition of each system status LED.

Table 1.3 Definitions of the system status LEDs

Name	Status	Meaning
POWER	ON	Emulator system power supply is turned ON.
	OFF	Emulator system power supply is turned OFF.
SAFE	ON	Emulator system is operating normally.
	Flashing	1. The self-check is being executed. 2. Downloaded firmware is being written into memory. 3. If the ERROR LED is flashing on at the same time or alternately, it indicates that the self-check error occurred.
	OFF	PC7501 system is not operating normally.
ERROR	ON	PC7501 system is not operating normally (system status error).
	Flashing	1. Firmware is being downloaded. 2. If the SAFE LED is flashing at the same time or alternately, it indicates that the self-check error occurred.
	OFF	PC7501 system is operating normally.



## (2) Target Status LEDs

The target status LEDs indicate operating status of the target MCU and power supply of the user system. Table 1.4 lists the definition of each target status LED.

Table 1.4 Definition of the target status LEDs

Name	Status	Meaning
POWER	ON	Power is supplied to the user system.
	OFF	Power is not supplied to the user system.
CLOCK	ON	Clock is supplied to the target MCU.
	OFF	Clock is not supplied to the target MCU.
RESET	ON	Target MCU is being reset, or reset signal of the user system is held low.
	OFF	Target MCU is not being reset.
RUN	ON	User program is being executed.
	OFF	User program has been halted.
WARNING	ON	Addresses 00000h and 00001h have been read out wrongly.
	OFF	Addresses 00000h and 00001h have not been read out wrongly.

## IMPORTANT

### Note on Usage in Memory Expansion or Microprocessor Mode:

- When you use this product in memory expansion or microprocessor mode, be sure to design your system so that pins RDY# and HOLD# are not held low at startup. Otherwise, it cannot startup normally.

### Note on the Target Status POWER LED:

- If your MCU has two or more Vcc terminals, the LED does not light unless power is supplied to all the terminals.

### Note on the Target Status CLOCK LED:

- CLOCK LED is turned off when both main and sub clocks are not oscillating. If either clock is oscillating, the LED is turned on.
- If the LED is not turned on, check the following.
  - (1) After powering on the PC7501 (before starting up the emulator debugger):  
Make sure that the oscillator circuit board is properly installed in the PC7501 and it is oscillating normally.
  - (2) After the emulator debugger is started up (after the Init dialog box settings are completed):  
Make sure that the oscillator selected in the Init dialog box is oscillating normally.

### Note on Accessing Addresses 00000h and 00001h:

- With the M16C/60 Series MCUs, when a maskable interrupt is generated, the interrupt data (interrupt number and interrupt request level) stored in addresses 00000h and 00001h are read out. Also, the interrupt request bit is cleared when address 00000h or 00001h is read out. Consequently, when the address 00000h or 00001h readout instruction is executed or when address 00000h or 00001h is read out in the cause of a program runaway, a malfunction occurs in that the interrupt is not executed despite the interrupt request, because the request bit of the highest priority interrupt factor enabled is cleared.  
For this malfunction, when the reading out to address 00000h or 00001h is generated excluding the interrupt, the WARNING LED (yellow) of the PC7501's upper panel lights up to alarm. When this LED lights, there is a possibility of wrong read access, therefore check the user program.

## 1.4 Specifications

Tables 1.5 and 1.6 list the specifications of the M306NKT-EPB.

Table 1.5 Specifications of the M306NKT-EPB (1/2)

Applicable MCU	M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN
Applicable MCU mode	Single-chip mode Memory expansion mode Microprocessor mode
Maximum operating frequency	24MHz (when using PLL)
Applicable power supply	3.0--5.5V
Emulation memory	(1) 512KB + 4KB for internal ROM of MCU 0F000h--0FFFFh, 80000h--FFFFFh, maximum operating frequency: 24MHz, 0 wait * When "Disable the Internal Flash ROM" is checked in the MCU tab of the emulator debugger's MCU Setting dialog box, it is 10MHz. (2) 31KB for internal RAM 00400h--07FFFh, maximum operating frequency: 24MHz, 0 wait (3) CS3*--CS0* area (Possible to allocate by 4KB) - Maximum operating frequency (at 5.0V) 0 wait: 7.0MHz 1 wait: 20.0MHz 2 wait, 3 wait: 24.0MHz - Maximum operating frequency (at 3.0V) 0 wait: 6.0MHz 1 wait: 17.0MHz 2 wait, 3 wait: 24.0MHz
Clock oscillator	X <sub>IN</sub> -X <sub>OUT</sub> (1 to 16MHz): Internal oscillator circuit board Switchable to external oscillator input or internal generator circuit X <sub>CIN</sub> -X <sub>COU</sub> T (32.768kHz): Internal oscillator circuit board Switchable to external oscillator input
Basic debugging functions	- Download - Address match break (max. 8 points) - Software break (max. 64 points) - Program execution/stop (allows free-run execution supporting software breaks) - Memory reference/setting (reference/setting C-variables, run-time execution) - Register reference/setting - Disassemble display - C-level debugging, etc.
Real-time trace function	- 256K-cycle bus information recordable (Bus, external trigger, time stamp) - 5 trace modes supported (Break/Before/About/After/Full) - Can be recorded ON/OFF by events
Real-time RAM monitor function	- 4,096 bytes (256 bytes x 16) - Data/last access result
Hardware break function	8 points (Execution address, bus detection, interrupt, external trigger signal)
Execution time measurement function	Time between program start and stop Maximum/minimum/average execution time and pass count of specified four zones. Count clock: Equal to MCU Clock or 16MHz
C0 coverage	8,192KB (256KB x 32 blocks)
External trigger input/event output	External trigger input (MCU-dependent-voltage CMOS level x8) or event output (break x1, event x7)

Table 1.6 Specifications of the M306NKT-EPB (2/2)

Host machine interface	<ul style="list-style-type: none"> <li>- LPT parallel (ECP, EPP, Byte/compatibility and Nibble/compatibility modes)</li> <li>- USB (USB 1.1, full-speed) *1</li> <li>- LAN (10BASE-T)</li> </ul>	
Power supply to emulator	Supplied from included AC adapter (power supply voltage: 100--240V, 50/60Hz)	
Operating temperature	5 to 35°C (no dew)	
Storage temperature	-10 to 60°C (no dew)	
Power supply to emulation probe	Supplied from the DC power supply of the emulator main unit	
Connection to user system (see 2.9 "Connecting the User System" on page 31)	M16C/6N4 M16C/6N5	For 100-pin 0.65mm pitch LCC: M30800T-PTC (included)
		For 100-pin 0.65mm pitch QFP (PRQP0100JB-A): (1) M3T-F160-100NRB (not included) (2) M30800T-PTC (included) + M3T-100LCC-DMS (not included) + M3T-DIRECT100S (not included)
		(3) M30800T-PTC (included) + M3T-100LCC-DMS (not included) + M3T-DUMMY100S (not included)
	M16C/6NK M16C/6NL M16C/6NM M16C/6NN	For 100-pin 0.5mm pitch LQFP (PLQP0100KB-A): (1) M3T-F160-100NSD (not included) (2) M30800T-PTC (included) + M3T-100LCC-QSD (not included)
		For 100-pin 0.5mm pitch LQFP (PLQP0100KB-A): (1) M3T-F160-100NSD (not included) (2) M30800T-PTC (included) + M3T-100LCC-QSD (not included)
		For 128-pin 0.5mm pitch LQFP (PLQP0128KB-A): (1) M3T-FLX-128NRD (not included)

\*1 Available to connect the host machine that supports USB 2.0.

With the USB interface, not all hardware (such as host machine, USB devices, USB hub) combination will work and guaranteed.

## 1.5 Operating Environment

Be sure to use this emulator with the operating environmental of the emulator and host machine listed in Tables 1.7 and 1.8.

Table 1.7 Operating environmental conditions

Item	Description
Operating temperature	5 to 35°C (no dew)
Storage temperature	-10 to 60°C (no dew)

Table 1.8 Operating environment of the host machine

Item	Description
Host machine	IBM PC/AT compatibles
OS	Windows 98SE* Windows Me Windows XP Windows 2000
CPU	Pentium III 600 MHz or more recommended
Memory	128 MB or more recommended
Pointing device such as mouse	Mouse or any other pointing device usable with the above OS that can be connected to the main body of the host machine.
CD drive	Needed to install the emulator debugger and refer to the user's manual

\* Windows and Windows NT are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries.

## 2. Setup

This chapter describes the preparation for using this product, the procedure for starting up the emulator and how to change settings.

### 2.1 Flowchart of Starting Up the Emulator

The procedure for starting up the emulator is shown in Figure 2.1. For details, refer to each section hereafter. And, when the emulator does not start up normally, refer to “5. Troubleshooting” (page 96).

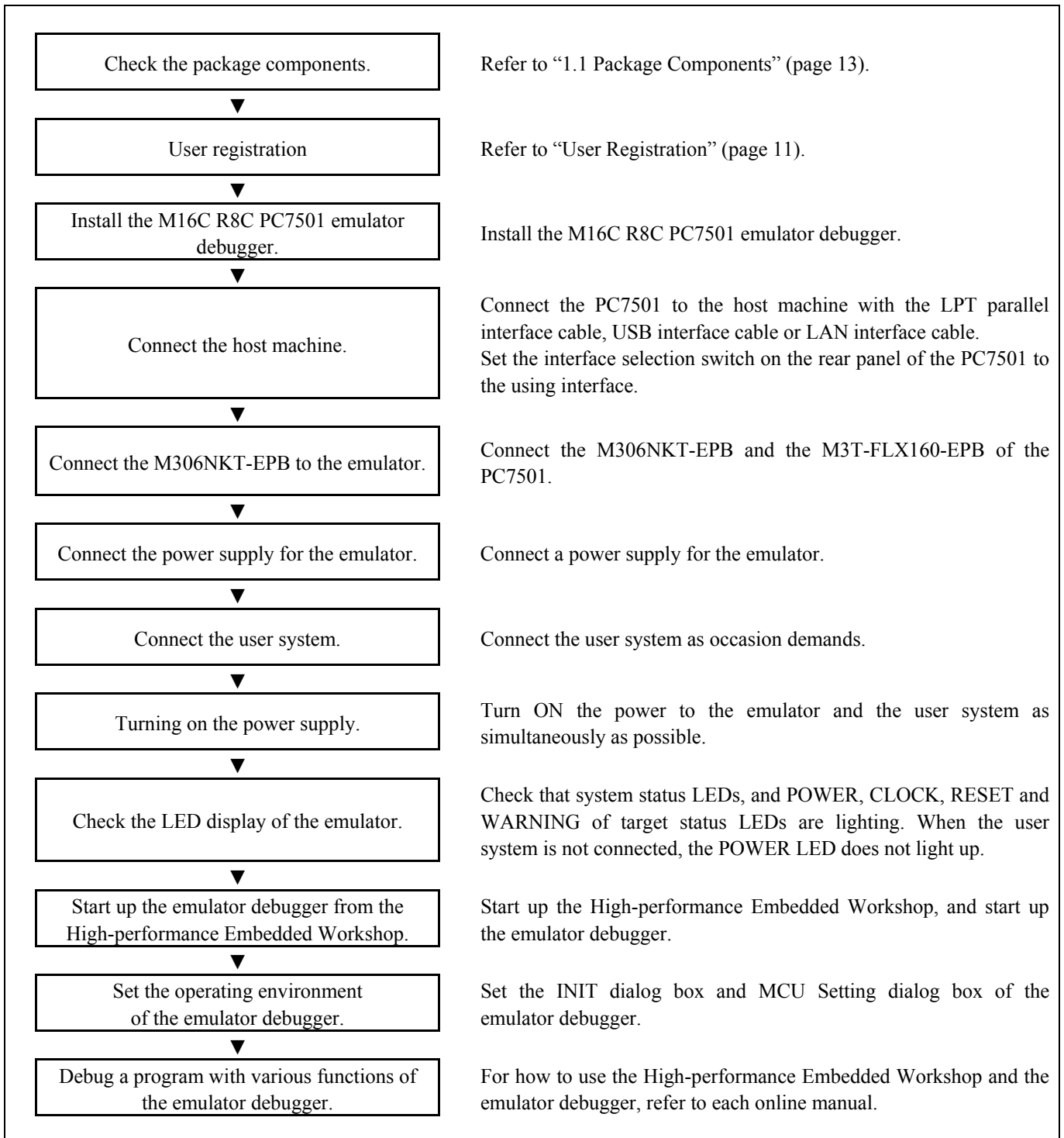


Figure 2.1 Flowchart of starting up the emulator

## 2.2 Installing the Emulator Debugger (M16C R8C PC7501 Emulator Debugger)

If the OS used in your host machine is Windows XP or 2000, this installation must be executed by a user with administrator rights. Be aware that users without administrator rights cannot complete the installation.

Install the M16C R8C PC7501 emulator debugger following the procedure described below.

### 2.2.1 Installing the Emulator Debugger

(1) Downloading the M16C R8C PC7501 emulator debugger

Download the M16C R8C PC7501 emulator debugger from the URL below.

[http://download.renesas.com/eng/mpumcu/upgrades/in\\_circuit\\_emulators/pc7501/index.html](http://download.renesas.com/eng/mpumcu/upgrades/in_circuit_emulators/pc7501/index.html)

(2) Launching the installer

Start the “setup.exe” program.

(3) Entering the user information

In the “user information” dialog box, enter the user information (contractor, section, contact address, and host machine). The supplied information will be turned into a format by which technical support will be provided by e-mail.

(4) Completing the installation

A dialog box will be displayed indicating that setup has been completed. It means that the installation you made is completed.

### 2.3 Connecting the Host Machine

When connecting the emulator PC7501 to a host machine, you can choose your desired interface from LPT parallel interface, USB interface and LAN interface. Use the interface selection switch on the emulator PC7501's rear panel to specify your desired interface. Figure 2.2 shows the outline to connect each interface cable.

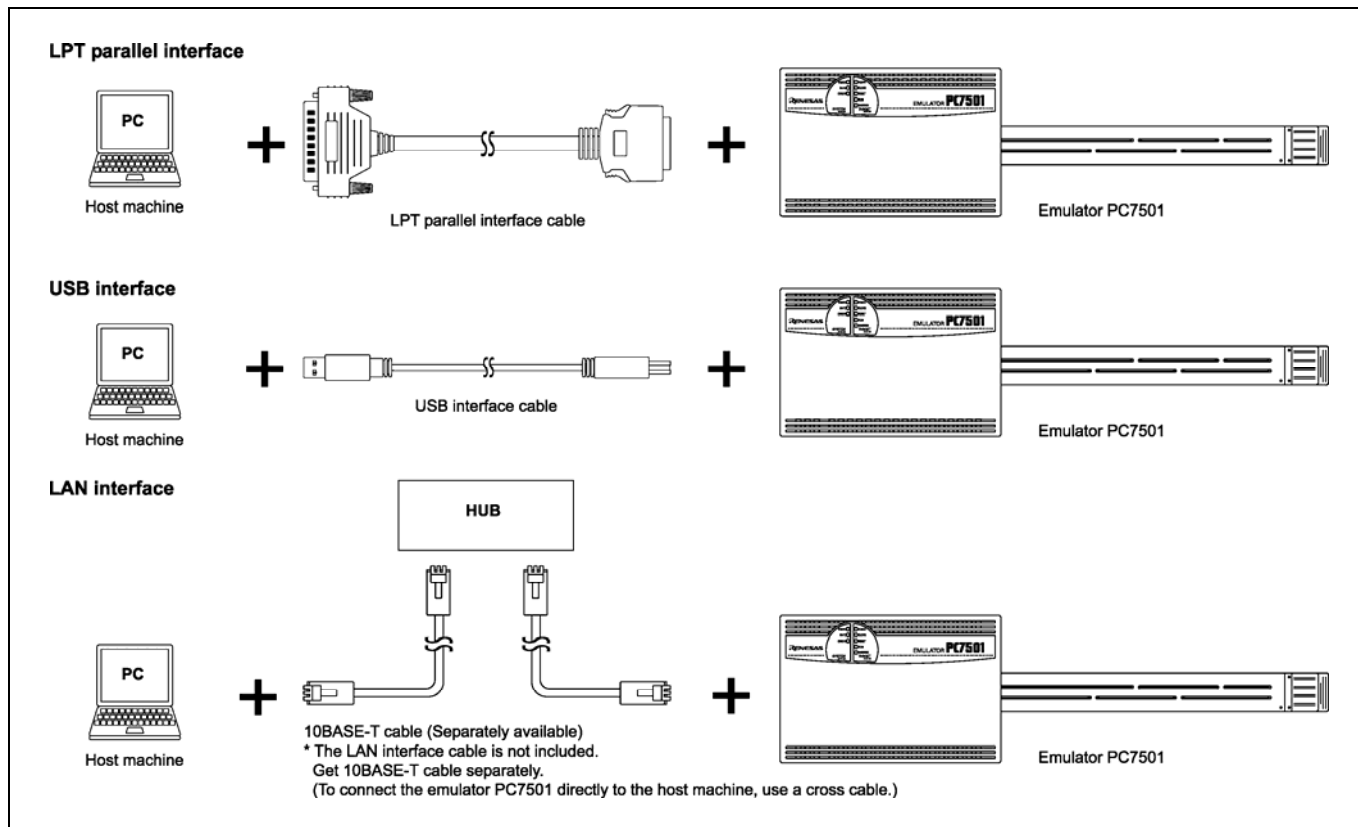


Figure 2.2 Outline for interface cable connections

## 2.4 Connecting the PC7501

Figure 2.3 shows how to connect the PC7501 and the emulation probe.

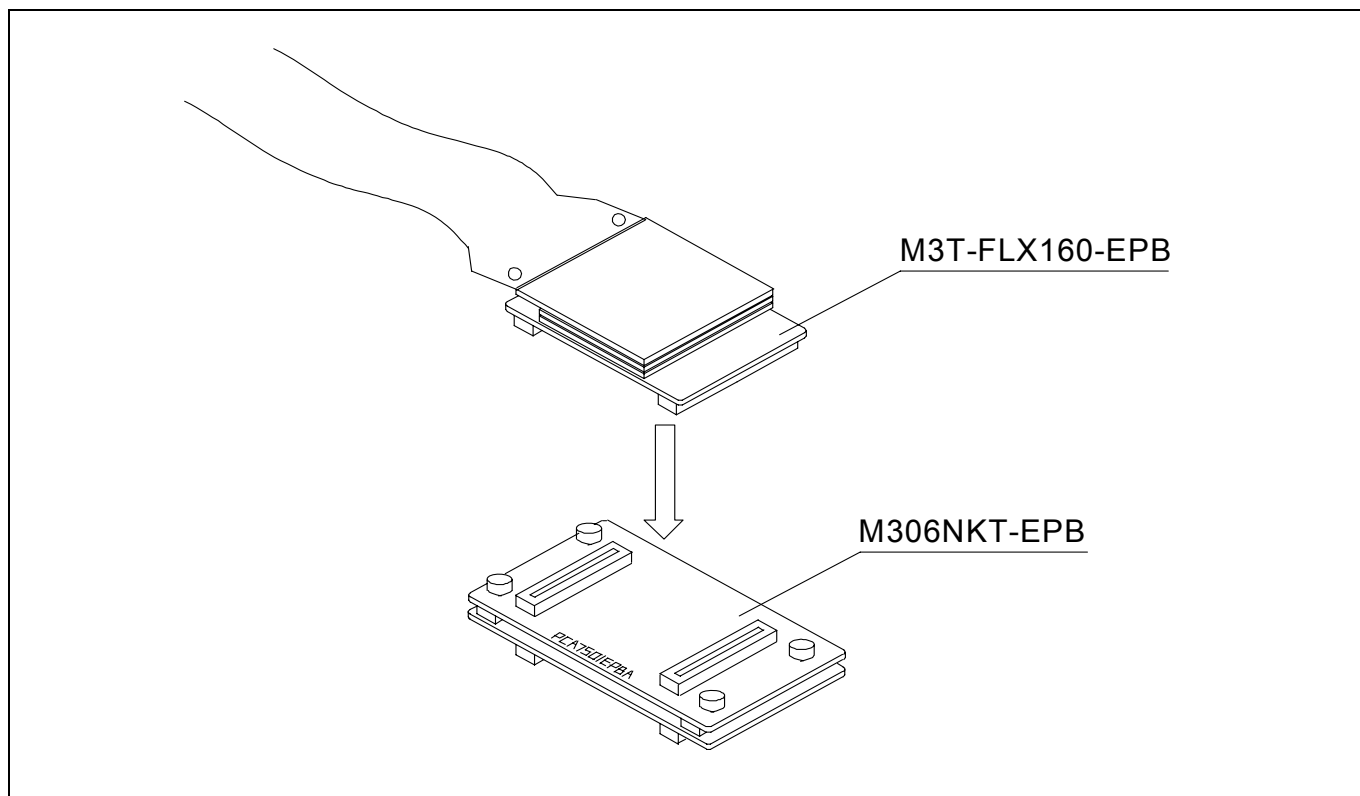


Figure 2.3 Connecting the PC7501 and emulation probe

### CAUTION

#### Note on Connecting the PC7501:



- Always shut OFF power before connecting the emulation probe. Otherwise, internal circuits may be damaged.

### IMPORTANT

#### Notes on Connecting the PC7501:

- When connecting the emulation probe, be sure to hold the both sides of the emulation probe horizontally and insert it directly.
- The connectors of the M306NKT-EPB are guaranteed for only 50 insertion/removal iterations.



## 2.5 Connecting the Power Supply for the Emulator

The power is supplied from AC adapter to the emulator PC7501. Here following explains how to connect the AC adapter.

- (1) Turn OFF the power to the PC7501.
- (2) Connect the DC cable of the AC adapter to the PC7501.
- (3) Connect the AC power cable to the AC adapter.
- (4) Connect the AC power cable to the outlet.

### CAUTION

#### Cautions for AC Adapter:



- Use only the AC adapter included in PC7501 package.
- The included AC adapter is for PC7501. Do not use it for other products.
- Before installing this equipment or connecting it to other equipment, disconnect the AC power cable from its outlet to prevent injury or accident.
- The DC plug on the included AC adapter has the below polarity.



- The included AC adapter has no power supply switch. The AC adapter is always active while connecting the AC power cable. Check if the power is supplied by the LED of the AC adapter.

## 2.6 Turning ON the Power

### 2.6.1 Checking the Connections of the Emulator System

Before turning the power ON, check the connection of the interface cable with host machine, PC7501, emulation probe, and user system.

### 2.6.2 Turning ON/OFF the Power

Turn ON the power of the emulator and user system as simultaneously as possible.

Turn OFF the power of the emulator and user system as simultaneously as possible.

Do not leave either the emulator or user system powered on, because of leakage current the internal circuits may be damaged.

When turning ON the power again after shutting OFF the power, wait for about 10 seconds.

## IMPORTANT

### Notes on Power Supply:

- The emulator's pin Vcc is connected to the user system in order to monitor user system voltage. For this reason, the emulator cannot supply power to the user system. Therefore, provide the user system with a separate power supply from that of the emulator.
- The voltage of the user system should be as follows.  
 $3.0\text{ V} \leq V_{cc} \leq 5.5\text{ V}$
- Do not change user system power supply voltage after power has been activated.

2.6.3 LED Display When the Emulator Starts Up Normally

Figure 2.4 shows upper panel LED lighting status when the emulator started up properly. Check it when starting up the emulator system.

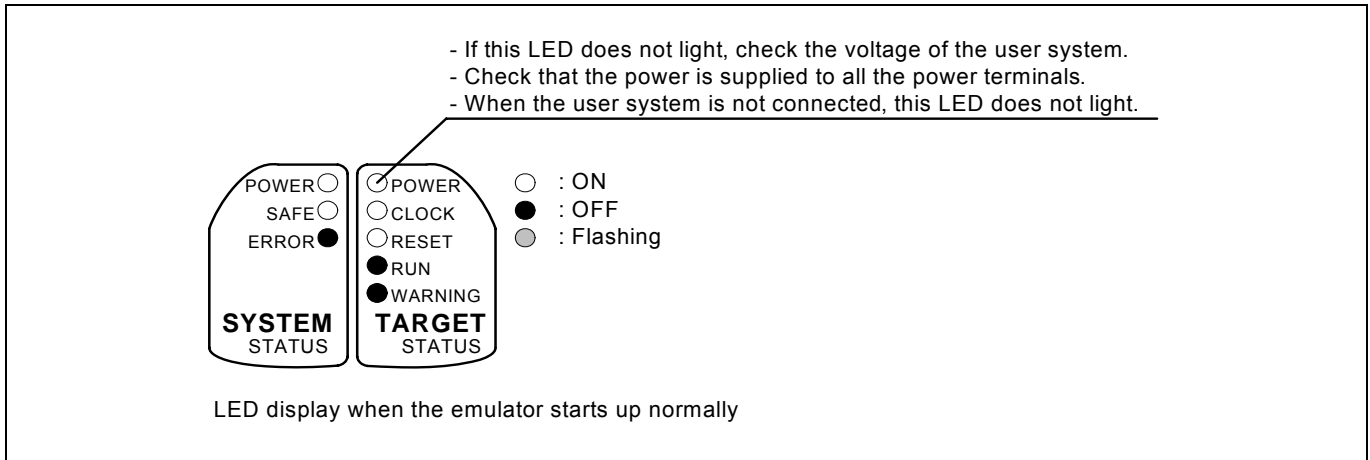


Figure 2.4 LED display when the power turned on PC7501

## IMPORTANT

**Note on Memory Expansion or Microprocessor Mode:**

- To use memory expansion or microprocessor mode, be sure to set pins RDY#, HOLD# and NMI# so that they are not held low at startup. Otherwise the emulator system will not start up correctly.

---

**Note on the Target Status POWER LED:**

- If your MCU has two or more Vcc terminals, the LED does not light unless power is supplied to all the terminals.

---

**Note on the Target Status CLOCK LED:**

- If the LED is not turned on, check the following.
  - (1) After powering on the PC7501 (before starting up the emulator debugger):  
Make sure that the oscillator circuit board is properly installed in the PC7501 and it is oscillating normally.
  - (2) After the emulator debugger is started up (after the Init dialog box settings are completed):  
Make sure that the oscillator selected in the Init dialog box is oscillating normally.

## 2.7 Downloading Firmware

### 2.7.1 When It is Necessary to Download Firmware

It is necessary to download the firmware in the cases listed below. Normally, the following are automatically detected when the emulator debugger is started up, and the firmware is downloaded.

- (1) When you use this product for the first time
- (2) When the firmware has been upgraded
- (3) When the emulator debugger has been upgraded
- (4) When you use this product with a PC7501 which was used with another emulation probe before

If downloading firmware is not completed in the cases below, redownload the firmware.

- When the power is unexpectedly shut down during a download from the emulator debugger
- When a communications interface cable is unexpectedly pulled out

### 2.7.2 Downloading Firmware in Maintenance Mode

Download firmware in the special mode called maintenance mode as explained here following.

**The user system must not be connected when you download the firmware.**

- (1) Set the interface select switch on the rear panel of the PC7501 to the LPT side and connect the LPT parallel interface cable to the PC7501 and the host machine.
- (2) Within 2 seconds of activating power to the emulator, press the system reset switch on the PC7501 front panel to switch to maintenance mode. When the emulator is switched to maintenance mode, the System Status SAFE LED begins to flash.
- (3) Start up the emulator debugger. When settings in the Init dialog box are complete, the dialog which urges to download the firmware will appear. Download firmware following messages. Required time for downloading the firmware is about 60 seconds.

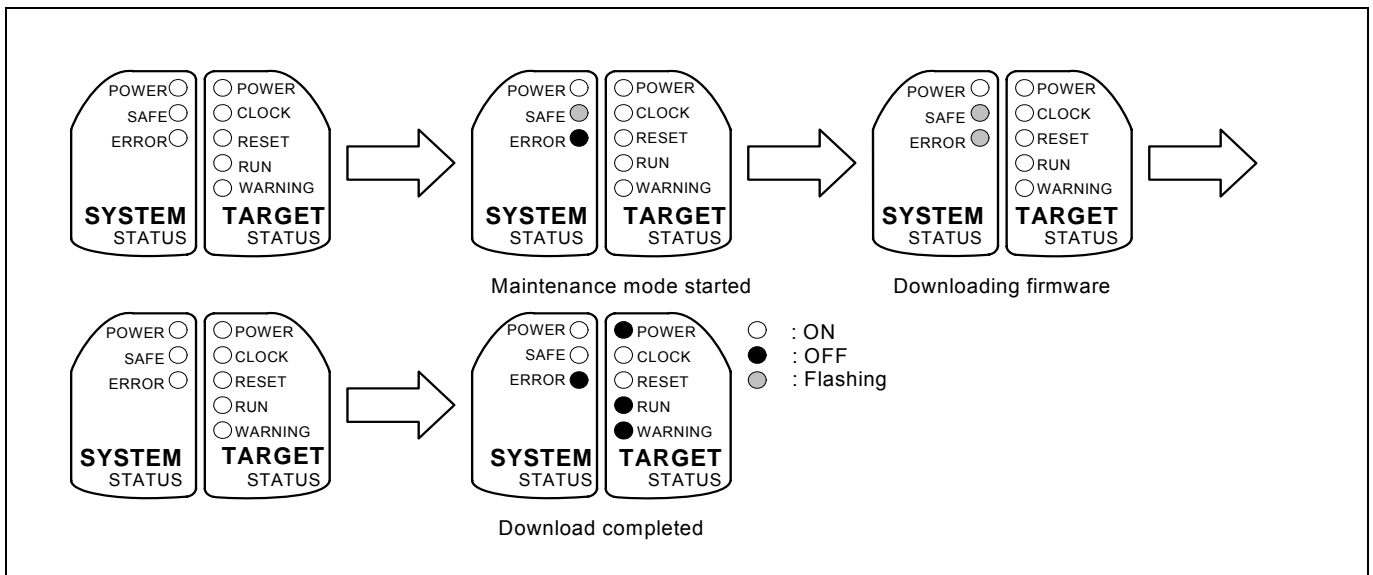


Figure 2.5 LEDs during firmware download in maintenance mode

## IMPORTANT

**Note on Downloading Firmware:**

- Do not shut OFF power while firmware is being downloaded. Doing so, the emulator will not start up properly. If power is shut off by mistake, redownload the firmware in maintenance mode.

## 2.8 Self-check

### 2.8.1 Self-check Procedure

The self-check is a function to check the memory etc. mounted in the emulator. To run the self-check of the emulator, do so as explained here below. While the self-check is in progress, the LEDs will change as shown in Figure 2.6. In case of ERRORS 1 to 4, because the target status LEDs will change depending on errors, check the system status LEDs.

- (1) If the user system is connected, disconnect it.
- (2) Set the switches as the factory-settings to execute the self-check (see Table 2.1).
- (3) Within 2 seconds of activating power to the emulator, press the system reset switch on the PC7501's front panel.
- (4) Check the "SAFE" LED starts flashing and then press the system reset switch again.
- (5) The self-check will start. If the normal result is displayed in about 30 seconds, the self-check terminated normally.

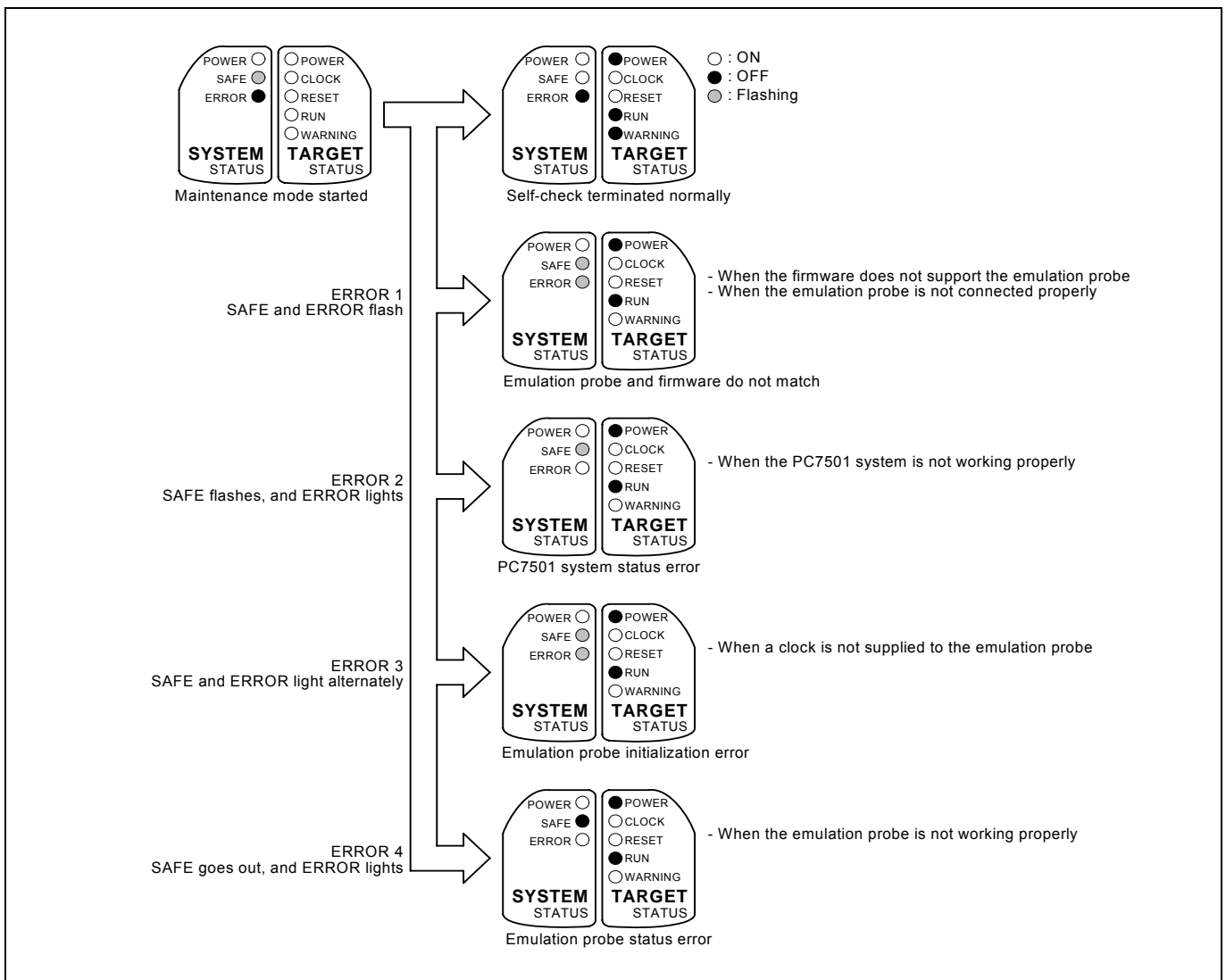


Figure 2.6 LED displays in the self-check

### 2.8.2 If an Error is Detected in the Self-check

If the self-check does not result normally (ERROR 1 to ERROR 4 in Figure 2.6), check the following.

- (1) Recheck the connection of the emulation probe and PC7501.
- (2) Redownload the proper firmware.

## IMPORTANT

### Note on the Self-check:

- Perform the self-check while not connecting the user system.
- If the self-check does not result normally (excluding user system errors), the emulation probe may be damaged. Then contact your local distributor.

### 2.9 Connecting the User System

Connect the emulation pod to the user system for M16C/6N4 or M16C/6N5, and for M16C/6NK, M16C/6NL, M16C/6NM and M16C/6NN as shown in Figures 2.7 and 2.8, respectively.

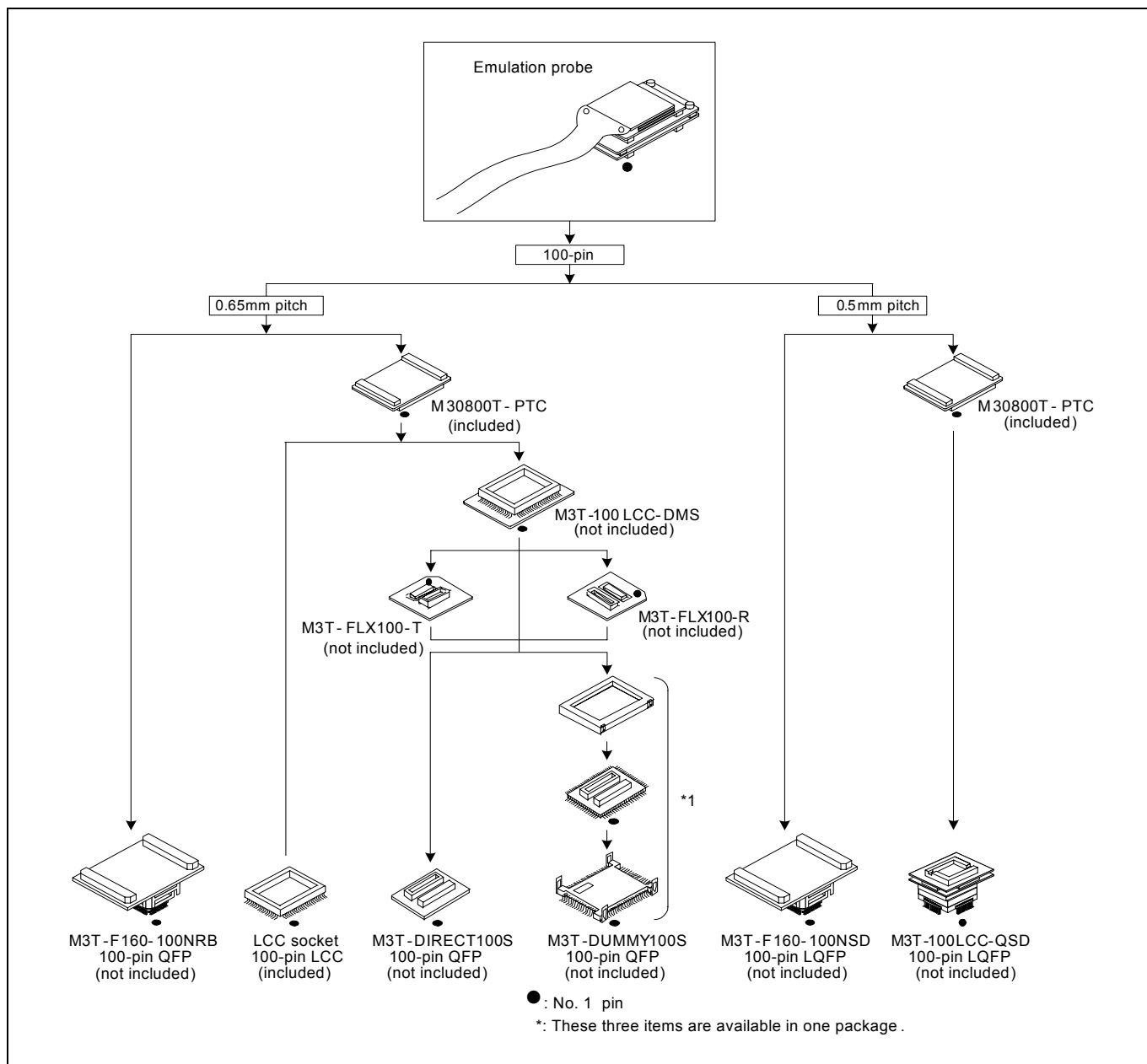


Figure 2.7 Connecting the user system according to user system (for M16C/6N4 and M16C/6N5)

## ⚠ CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

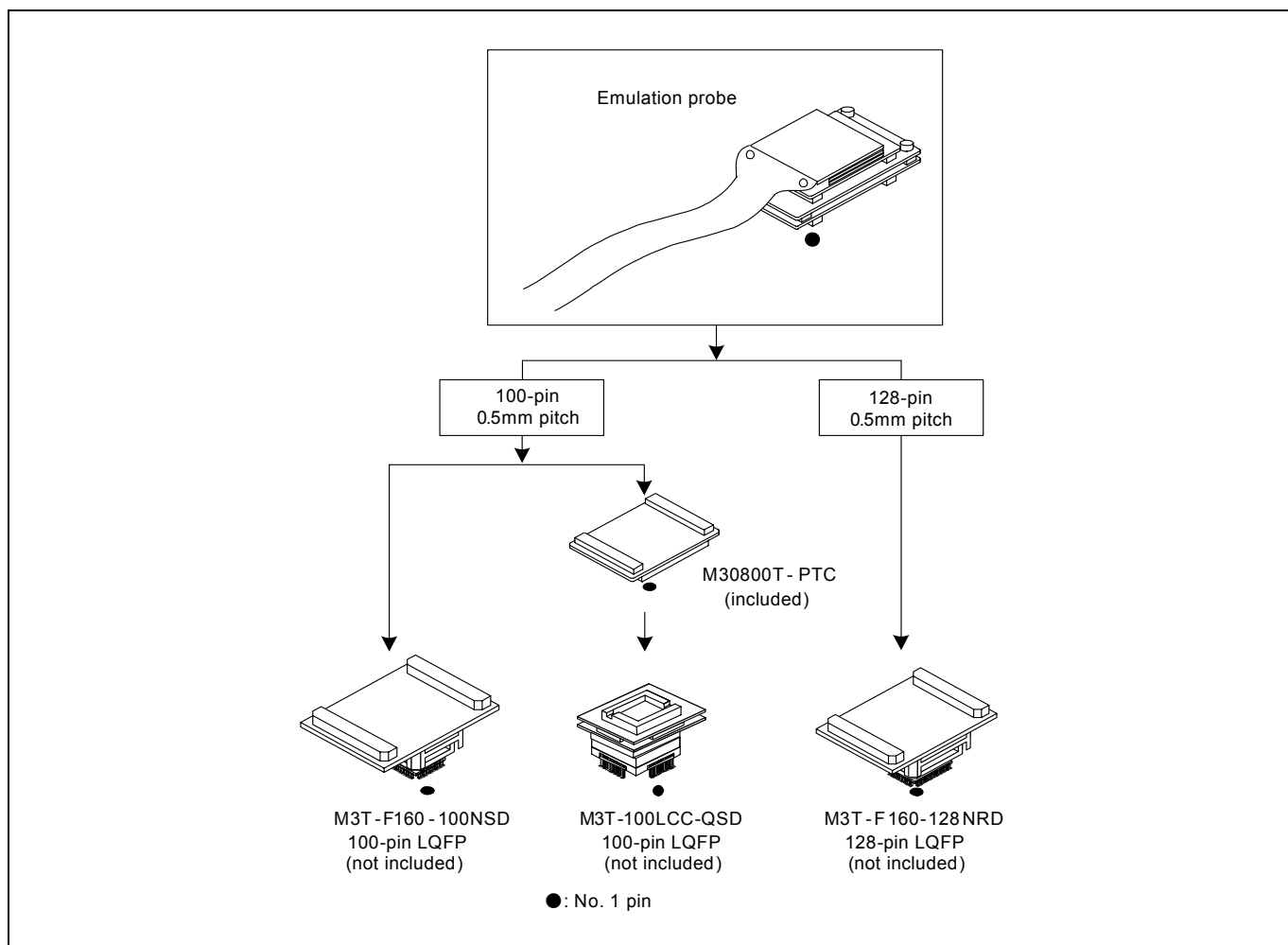


Figure 2.8 Connecting the user system according to user system (for M16C/6NK, M16C/6NL, M16C/6NM and M16C/6NN)

## CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.



### 2.9.1 Connecting to a 100-pin LCC Socket

Here following is a procedure of connecting to a 100-pin LCC socket (IC61-1004-051 of Yamaichi Electronics Co., Ltd.) on the user system using the included M30800T-PTC.

- (1) Attach the CN2 side of the M30800T-PTC to the J4 side of the M306NKT-EPB.
- (2) Attach the M30800T-PTC to the 100-pin LCC socket.

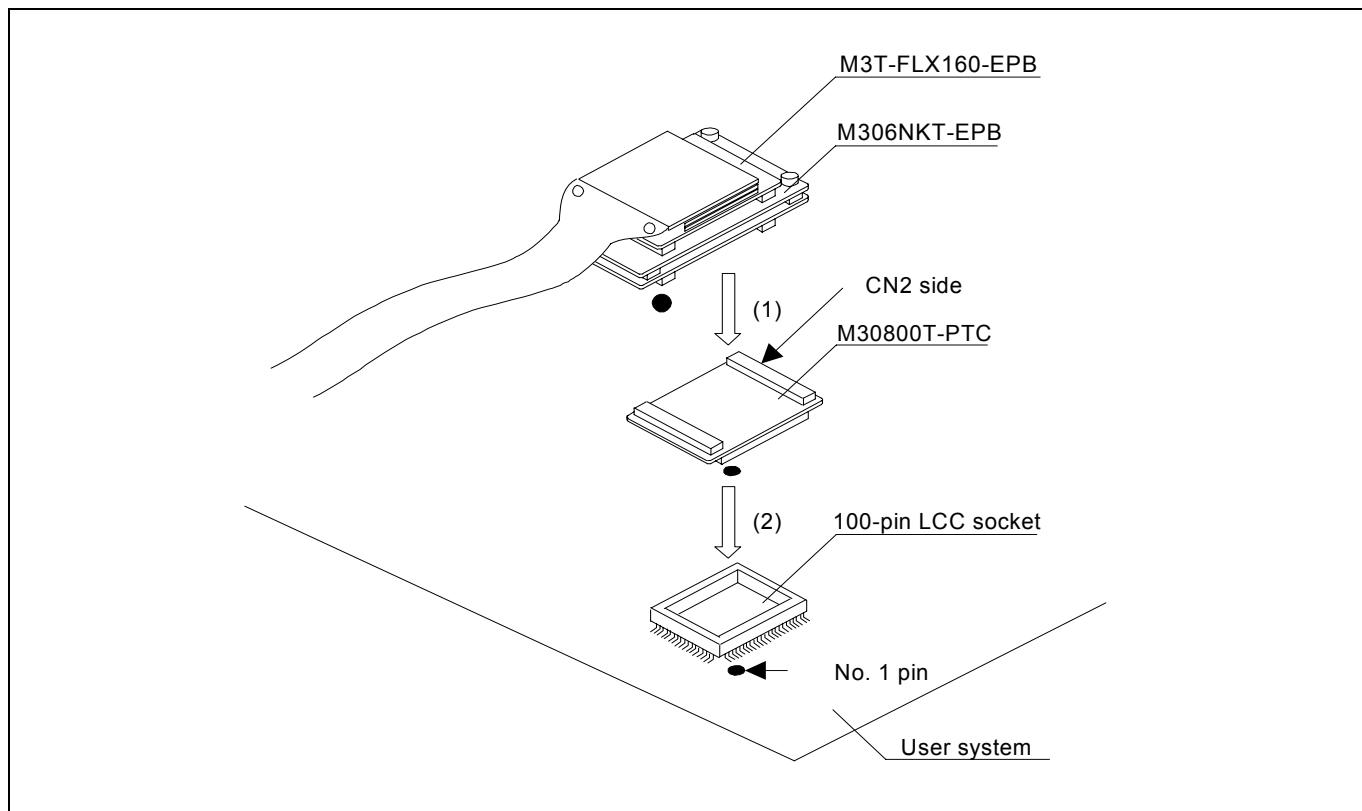


Figure 2.9 Connecting to a 100-pin LCC socket

## ⚠ CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

## IMPORTANT

### Notes on Connectors of the Converter board:

- The connectors of the M30800T-PTC are guaranteed for only 50 insertion/removal iterations.
- For purchasing the IC61-1004-051 or for technical information, contact Yamaichi Electronics Co., Ltd.

2.9.2 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 1)

Here following is a procedure of connecting to a 100-pin 0.65mm pitch foot pattern on the user system using the M3T-F160-100NRB (not included). For details on the M3T-F160-100NRB (not included), refer to its user's manual.

- (1) Attach the NQPACK100RB included with the M3T-F160-100NRB to the user system.
- (2) Attach the YQPACK100RB included with the M3T-F160-100NRB to the NQPACK100RB and secure it with the YQ-GUIDE's.
- (3) Attach the CN2 side of the M3T-F160-100NRB to the J4 side of the M306NKT-EPB.
- (4) Attach the M3T-F160-100NRB to the YQPACK100RB.

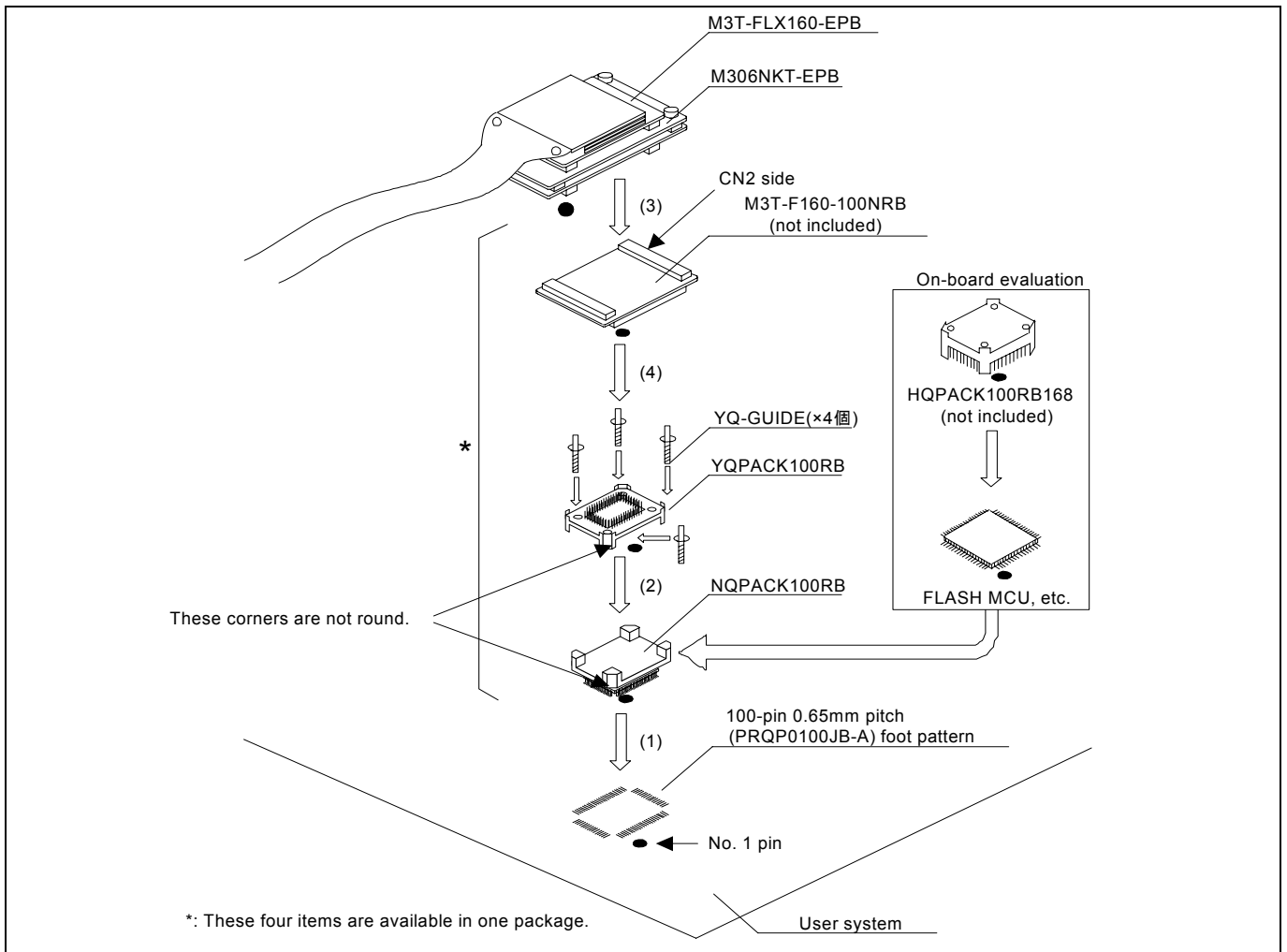


Figure 2.10 Connecting to a 100-pin 0.65mm pitch foot pattern (Part 1)

**CAUTION**

Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

**IMPORTANT**

Note on Connectors of the Converter board:

- The connectors of the M3T-F160-100NRB are guaranteed for only 50 insertion/removal iterations.

### 2.9.3 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 2)

Here following is a procedure of connecting to a 100-pin 0.65mm pitch foot pattern on the user system using the M3T-DIRECT100S (not included). For details on the M3T-100LCC-DMS (not included) and M3T-DIRECT100S (not included), refer to each user's manual.

- (1) Attach the M3T-DIRECT100S to the user system.
- (2) Attach the M3T-100LCC-DMS to the M3T-DIRECT100S.
- (3) Attach the CN2 side of the M30800T-PTC to the J4 side of the M306NKT-EPB.
- (4) Attach the M30800T-PTC to the M3T-100LCC-DMS.

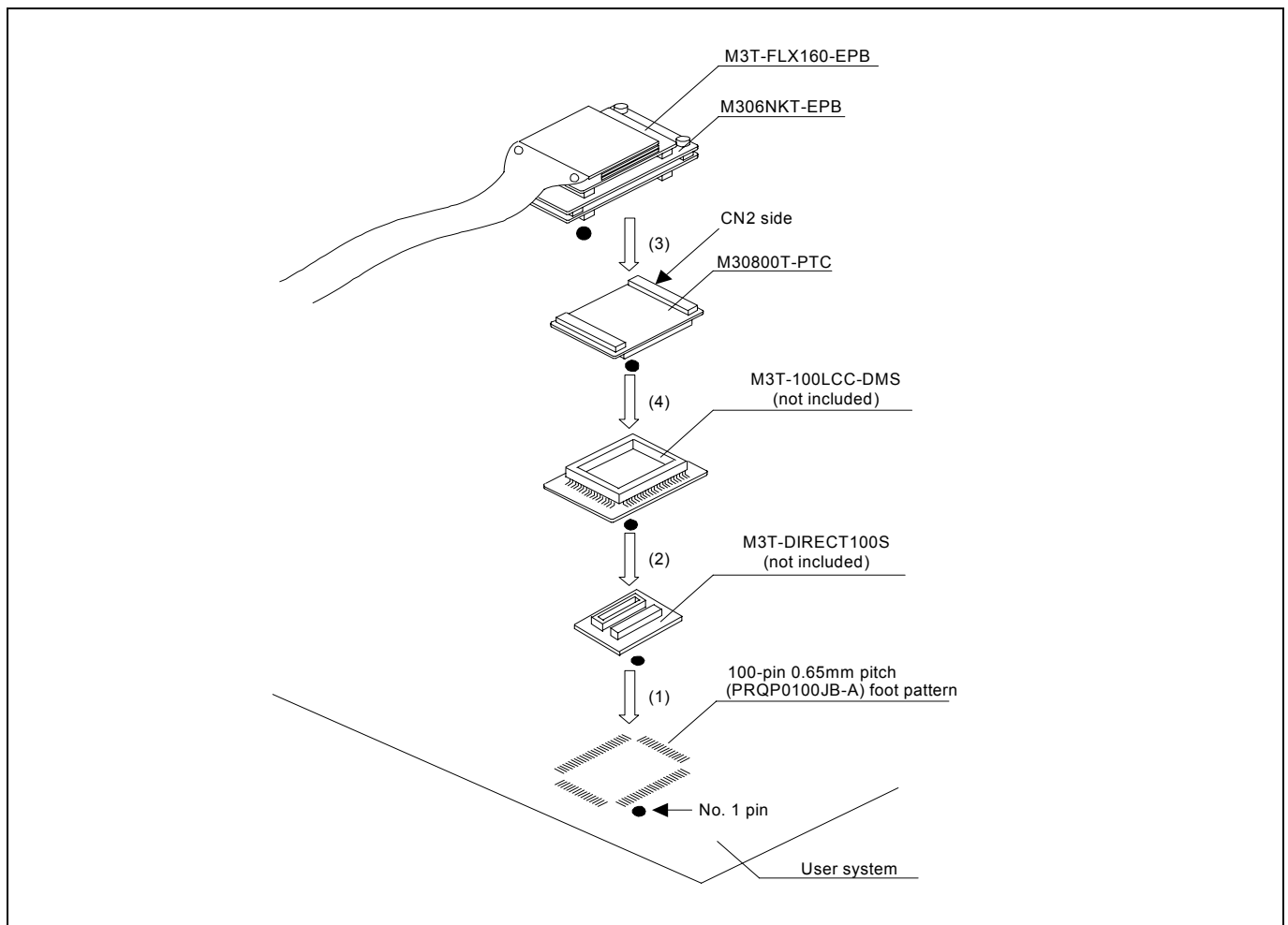


Figure 2.11 Connecting to a 100-pin 0.65mm pitch foot pattern (Part 2)

## ⚠ CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

## IMPORTANT

### Notes on Connectors of the Converter board:

- The connectors of the M30800T-PTC are guaranteed for only 50 insertion/removal iterations.
- The connectors of the M3T-100LCC-DMS and M3T-DIRECT100S are guaranteed for only 20 insertion/removal iterations.

2.9.4 Connecting to a 100-pin 0.65mm Pitch Foot Pattern (Part 3)

Here following is a procedure of connecting to a 100-pin 0.65mm pitch foot pattern on the user system using the M3T-DUMMY100S (not included). For details on the M3T-100LCC-DMS (not included) and M3T-DUMMY100S (not included), refer to each user's manual.

- (1) Attach the M3T-DUMMY100S to the user system.
- (2) Attach the M3T-100LCC-DMS to the M3T-DUMMY100S.
- (3) Attach the CN2 side of the M30800T-PTC to the J4 side of the M306NKT-EPB.
- (4) Attach the M30800T-PTC to the M3T-100LCC-DMS.

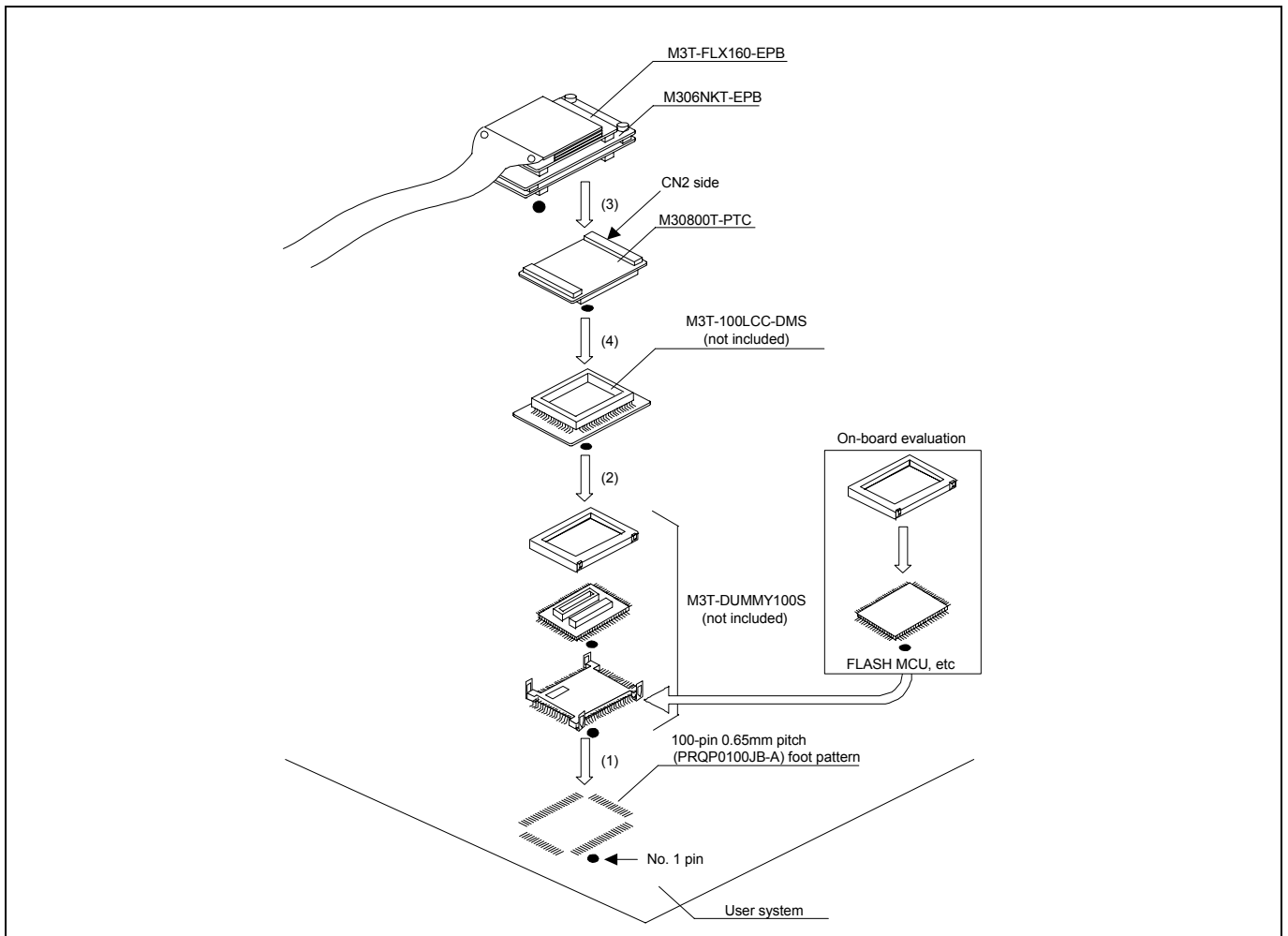


Figure 2.12 Connecting to a 100-pin 0.65mm pitch foot pattern (Part 3)

**CAUTION**

Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

**IMPORTANT**

Notes on Connectors of the Converter board:

- The connectors of the M30800T-PTC are guaranteed for only 50 insertion/removal iterations.
- The connectors of the M3T-100LCC-DMS and M3T-DUMMY100S are guaranteed for only 20 insertion/removal iterations.

2.9.5 Connecting to a 100-pin 0.5mm Pitch Foot Pattern (Part 1)

Here following is a procedure of connecting to a 100-pin 0.5mm pitch foot pattern on the user system using the M3T-F160-100NSD (not included). For details on the M3T-F160-100NSD (not included), refer to its user's manual.

- (1) Attach the NQPACK100SD-ND included with the M3T-F160-100NSD to the user system.
- (2) Attach the YQPACK100SD included with the M3T-F160-100NSD to the NQPACK100SD-ND and secure it with the YQ-GUIDE's.
- (3) Attach the CN2 side of the M3T-F160-100NSD to the J4 side of the M306NKT-EPB.
- (4) Attach the M3T-F160-100NSD to the YQPACK100SD.

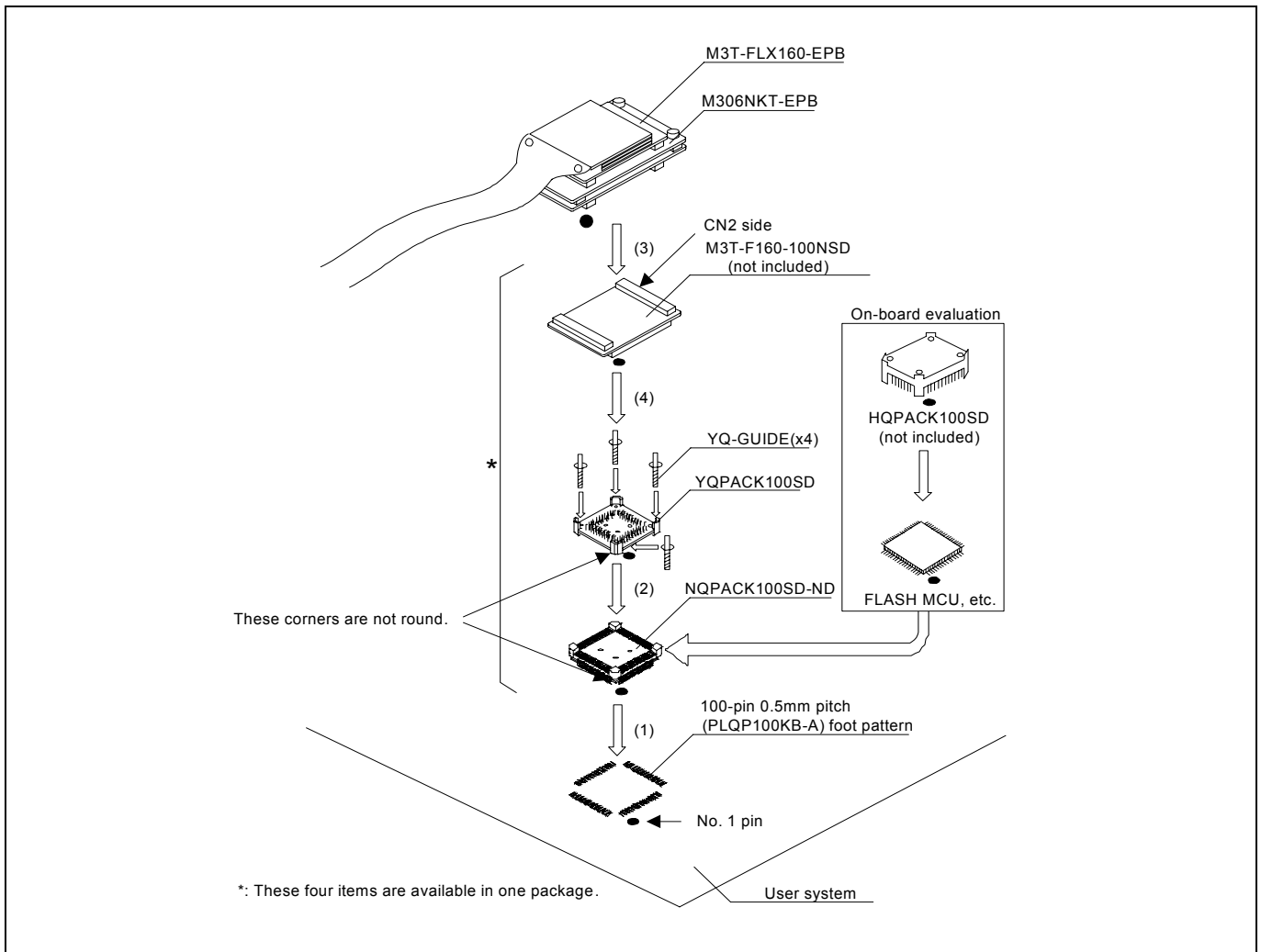


Figure 2.13 Connecting to a 100-pin 0.5mm pitch foot pattern (Part 1)

**CAUTION**

Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

**IMPORTANT**

Note on Connectors of the Converter board:

- The connectors of the M3T-F160-100NSD are guaranteed for only 50 insertion/removal iterations.

### 2.9.6 Connecting to a 100-pin 0.5mm pitch Foot Pattern (Part 2)

Here following is a procedure of connecting to a 100-pin 0.5mm pitch foot pattern on the user system using the M3T-100LCC-QSD (not included). For details on the M3T-100LCC-QSD (not included), refer to its user's manual.

- (1) Attach the TQPACK100SD to the user system.
- (2) Attach the TQSOCKET100SDG included with the M3T-100LCC-QSD to the TQPACK100SD.
- (3) Attach the M3T-100LCC-QSD to the TQSOCKET100SDG.
- (4) Attach the CN2 side of the M30800T-PTC to the J4 side of the M306NKT-EPB.
- (5) Attach the M30800T-PTC to the M3T-100LCC-QSD.

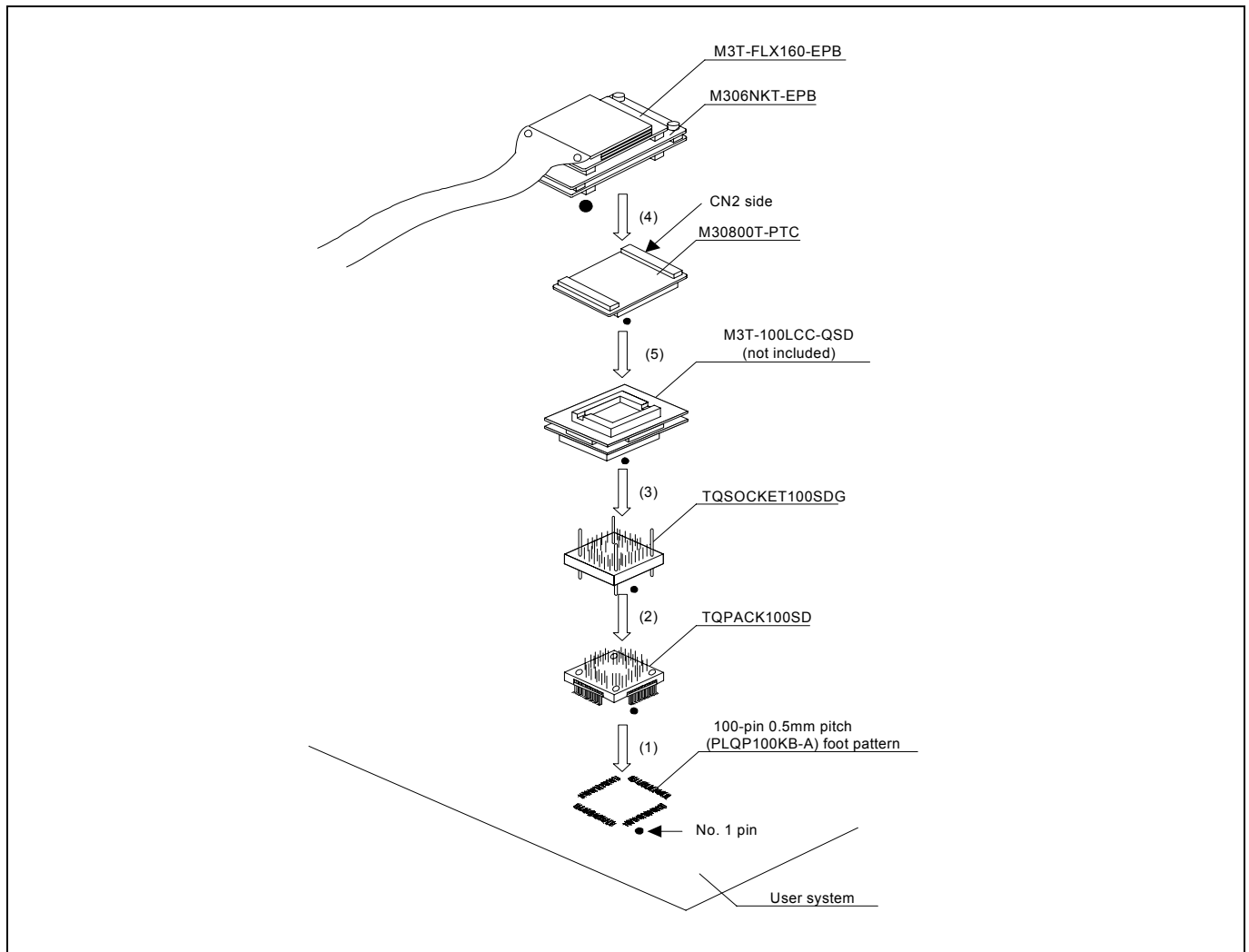


Figure 2.14 Connecting to a 100-pin 0.5mm pitch foot pattern (Part 2)

## CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

## IMPORTANT

### Note on Connectors of the Converter board:

- The connectors of the M30800T-PTC and M3T-100LCC-QSD are guaranteed for only 50 insertion/removal iterations.

### 2.9.7 Connecting to a 128-pin 0.5mm pitch Foot Pattern

Here following is a procedure of connecting to a 128-pin 0.5mm pitch foot pattern on the user system using the M3T-F160-128NRD (not included). For details on the M3T-F160-128NRD (not included), refer to its user's manual.

- (1) Attach the NQPACK128RD included with the M3T-F160-128NRD to the user system.
- (2) Attach the YQPACK128RD included with the M3T-F160-128NRD to the NQPACK128RD and secure it with the YQ-GUIDE's.
- (3) Attach the CN2 side of the M3T-F160-128NRD to the J4 side of the M306NKT-EPB.
- (4) Attach the M3T-F160-128NRD to the YQPACK128RD.

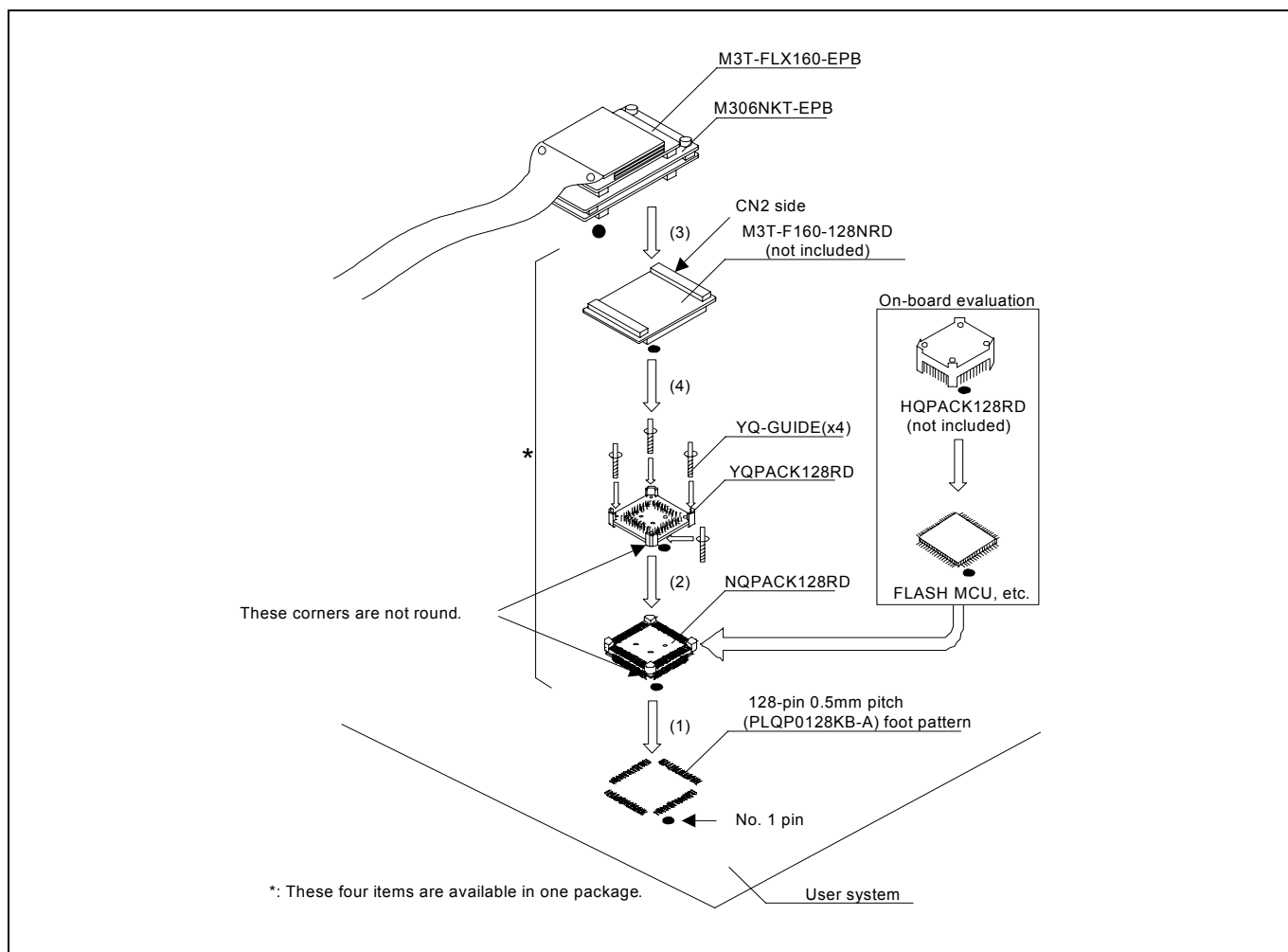


Figure 2.15 Connecting to a 128-pin 0.5mm pitch foot pattern

## CAUTION

### Note on Connecting the User System:



- Take care not to attach a converter board in a wrong direction. It may cause a fatal damage to the emulator and user system.

## IMPORTANT

### Note on Connectors of the Converter board:

- The connectors of the M3T-F160-128NRD are guaranteed for only 50 insertion/removal iterations.

## 2.10 Switch Settings

### 2.10.1 A/D Conversion Bypass Capacitors

There is a foot pattern on the M306NKT-EPBM board for mounting bypass capacitors for the A/D conversion circuit near the MCU. Mount suitable bypass capacitors as occasion demands. Figure 2.16 shows where they are installed and the configuration of this product.

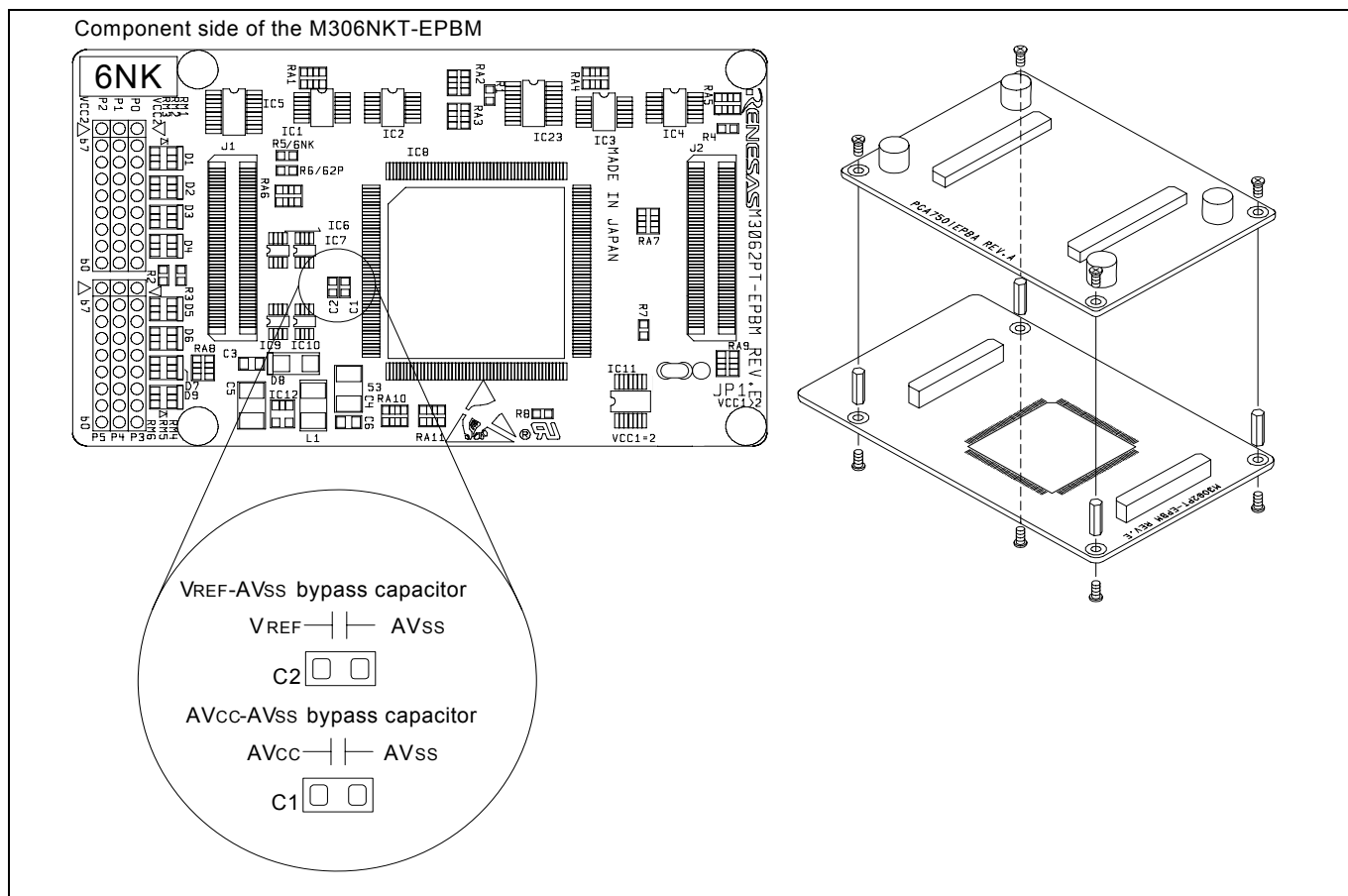


Figure 2.16 Foot pattern for A/D conversion bypass capacitors and the configuration of this product

## IMPORTANT

Note on the A/D Converter:

- Because a package converter board and other devices are used between the evaluation MCU and the user system, the A/D converter operates differently from that of an actual MCU. Make the final evaluation of the A/D converter with an actual MCU.



### 2.10.2 Pullup of Emulation Ports

Because this product emulates some I/O ports (P0 to P5, P10), you can not pullup ports P0 to P5 by the pullup control registers. Install the included resistor arrays (51 k $\Omega$ ) as necessary. Figure 2.17 shows the positions of the pullup resistors.

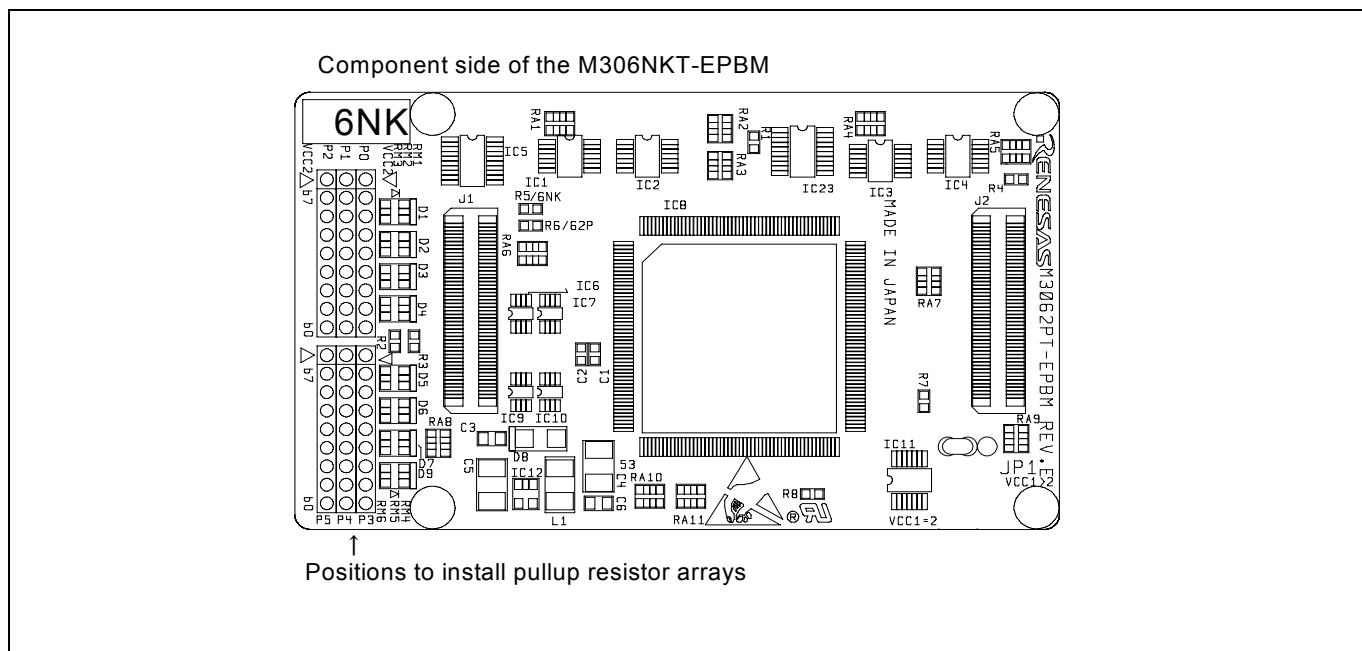


Figure 2.17 Positions of the pullup resistors

Table 2.1 Emulation ports and pullup resistor part numbers

Port	Resistor part
P00--P07	RM1 (51 k $\Omega$ pullup)
P10--P17	RM2 (51 k $\Omega$ pullup)
P20--P27	RM3 (51 k $\Omega$ pullup)
P30--P37	RM4 (51 k $\Omega$ pullup)
P40--P47	RM5 (51 k $\Omega$ pullup)
P50--P57	RM6 (51 k $\Omega$ pullup)

## 2.11 Selecting Clock Supply

### 2.11.1 Clocks

You can choose a clock supplied to the evaluation MCU by the Emulator tab in the Init dialog box of the emulator debugger. Table 2.2 shows the clocks and their initial settings.

Table 2.2 Clock supply to the MCU

Clock	Display of emulator debugger	Description	Default setting
Main (X <sub>IN</sub> -X <sub>OUT</sub> )	Internal	Internal oscillator circuit board (OSC-3 or OSC-2)	Yes
	External	Oscillator circuit on the user system	-
	Generated	Internal generator circuit (1.0--16.0MHz)	-
Sub (X <sub>CIN</sub> -X <sub>COU</sub> T)	Internal	Internal oscillator circuit (32.768kHz)	-
	External	Oscillator circuit on the user system	Yes

## IMPORTANT

### Note on Changing the Clock Supply:

- The clock supply can be set by the Init dialog box when starting up the emulator debugger or inputting CLK command on the script window.

### 2.11.2 Using an Internal Oscillator Circuit Board

#### (1) Kinds of Oscillator Circuit Boards

The PC7501 comes with an oscillator circuit board OSC-3 (30MHz). And an oscillator circuit board OSC-3 (16MHz) and an oscillator circuit board OSC-2 (bare board) are included with this product.

If you use the internal oscillator circuit board OSC-3 (16MHz) or OSC-2 of the PC7501 as a main clock, choose "Internal" in the emulator debugger after replacing oscillator circuit boards to change a clock supplied to an MCU.

## (2) Replacing an Oscillator Circuit Board

1. Remove the four screws of both sides of this product and lift off the upper cover (see Figure 2.18).

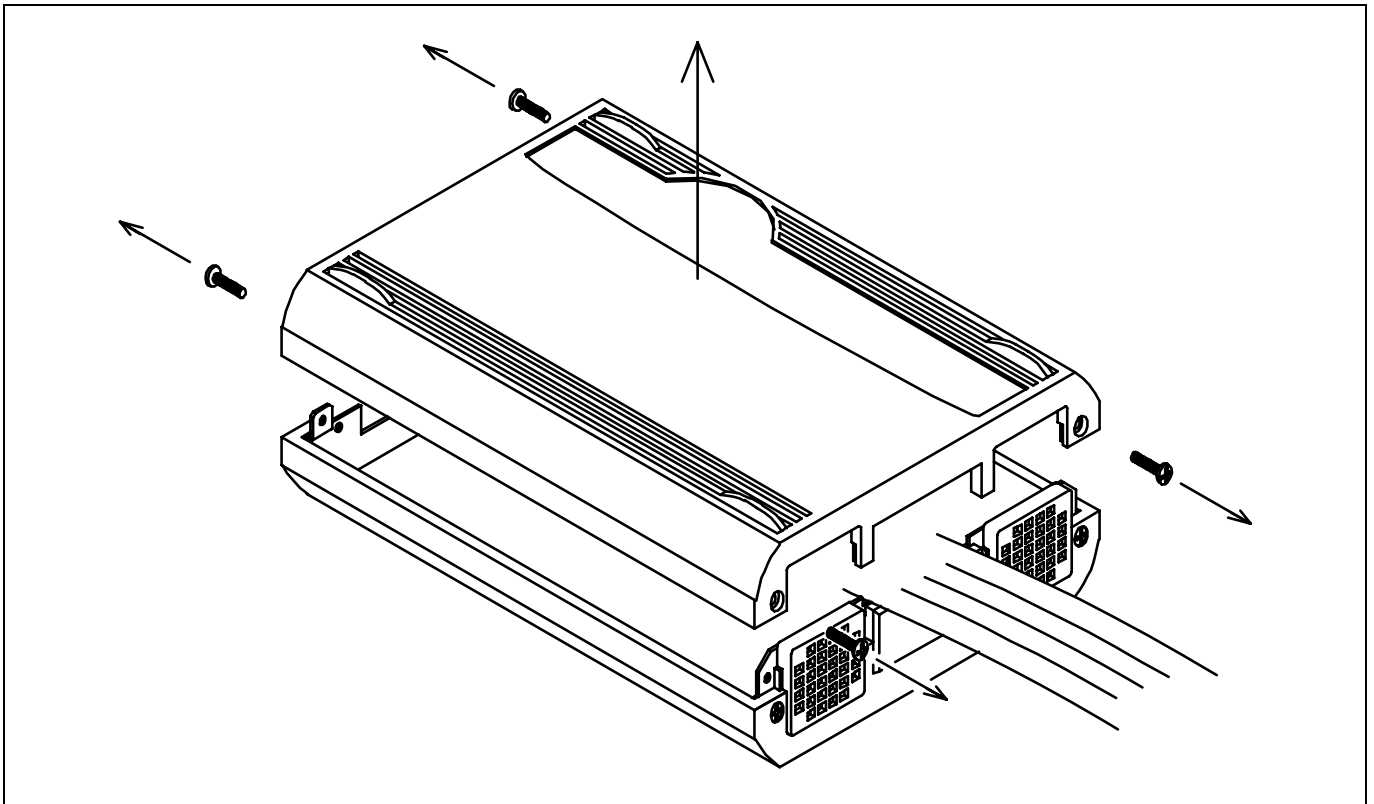


Figure 2.18 Removing the upper cover

2. Replace the oscillator circuit board (see Figure 2.19).

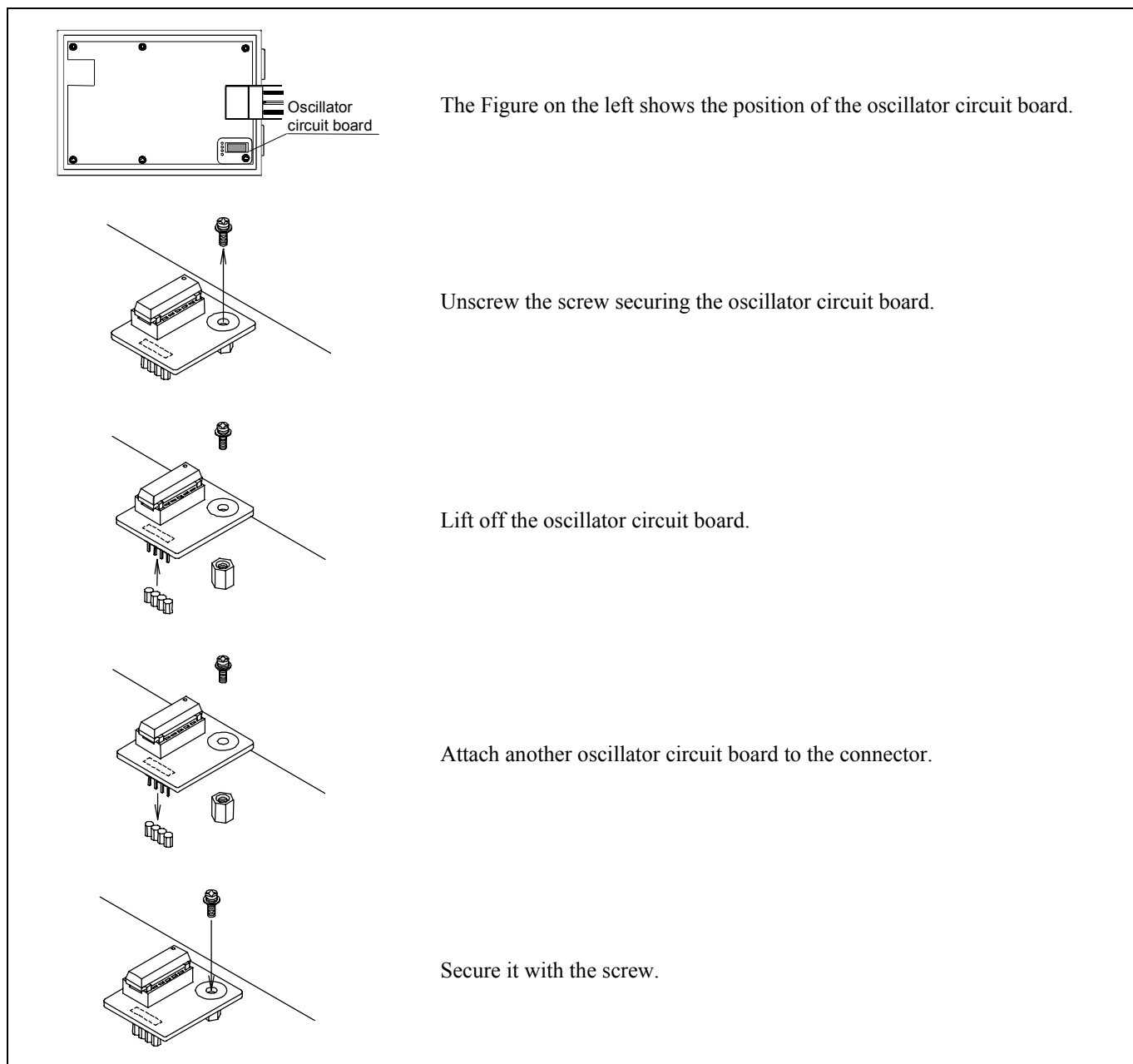


Figure 2.19 Replacing the oscillator circuit board

3. Reinstall the upper cover and secure the four screws of the PC7501.

## CAUTION

### Note on Replacing the Oscillator Circuit Board:



- Always shut OFF power when removing the upper cover or replacing oscillator circuit board. Otherwise, internal circuits may be damaged.

(3) Using the Internal Oscillator Circuit Bare Board

To use the emulation probe at a frequency you like, build a desired oscillator circuit on the included OSC-2 oscillator circuit bare board. Figure 2.20 shows an external view of the OSC-2 oscillator circuit bare board and where the connector pins are located. Figure 2.21 shows the circuitry of the oscillator circuit bare board OSC-2. Use the number of oscillator circuits recommended by the oscillator manufacturer.

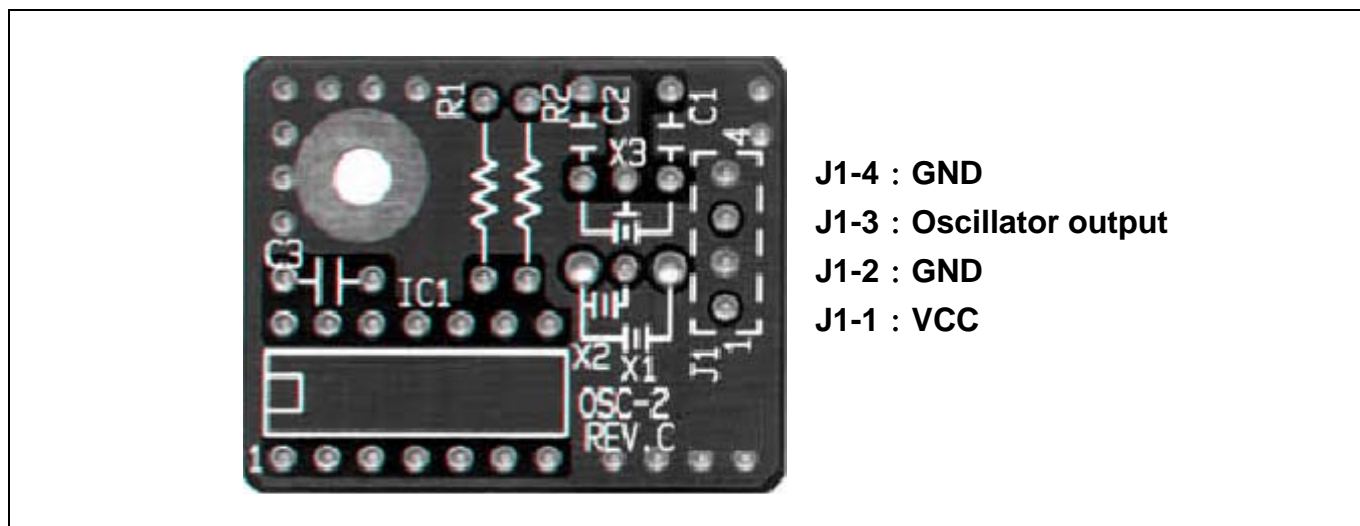


Figure 2.20 External view of the oscillator circuit board OSC-2 and its connector pin positions

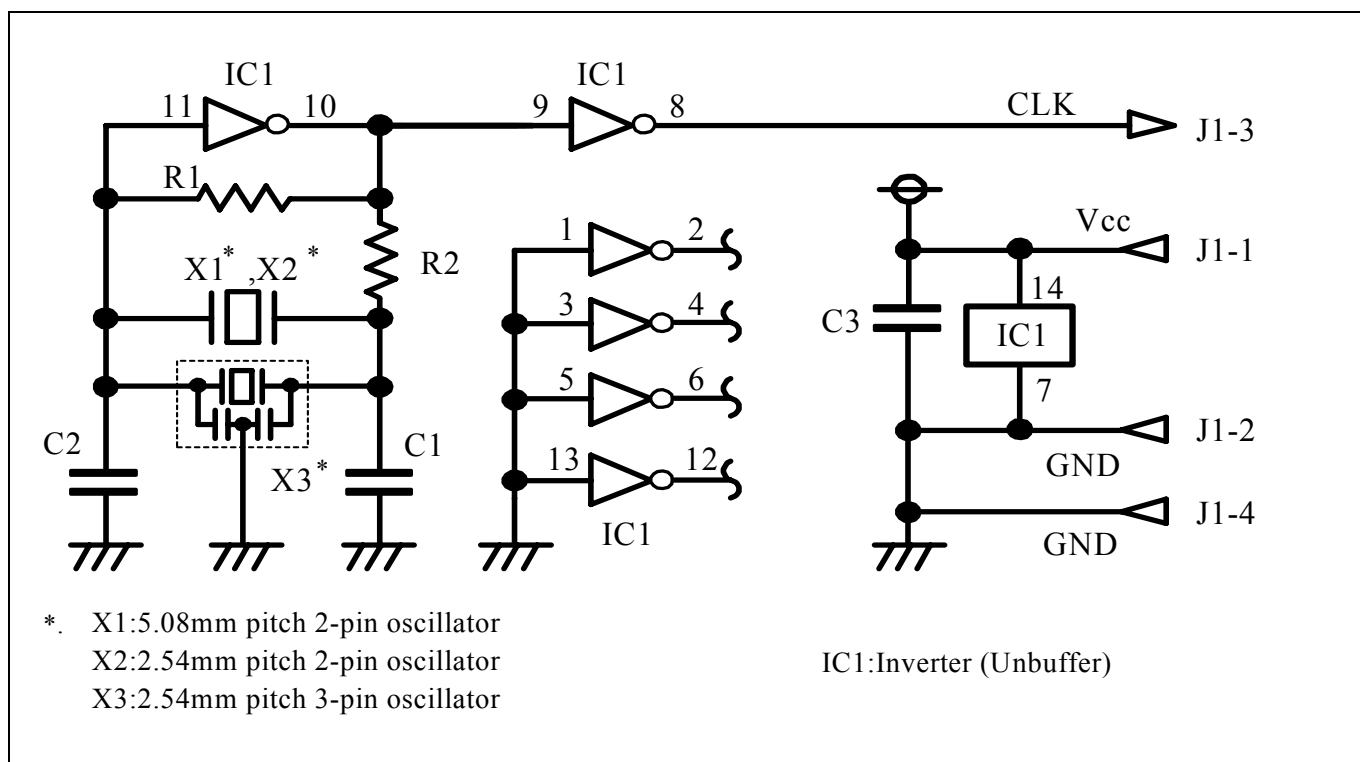


Figure 2.21 Circuits of the oscillator circuit bare board OSC-2

### 2.11.3 Using the Oscillator Circuit on the User System

To operate this product with an external clock, construct the oscillator circuit as shown in Figure 2.22 in the user system and input the oscillator output at 50% duty (within the operating range of the evaluation MCU) into pin  $X_{IN}$ . And pin  $X_{OUT}$  should be open. Choose "External" in the emulator debugger to use this clock.

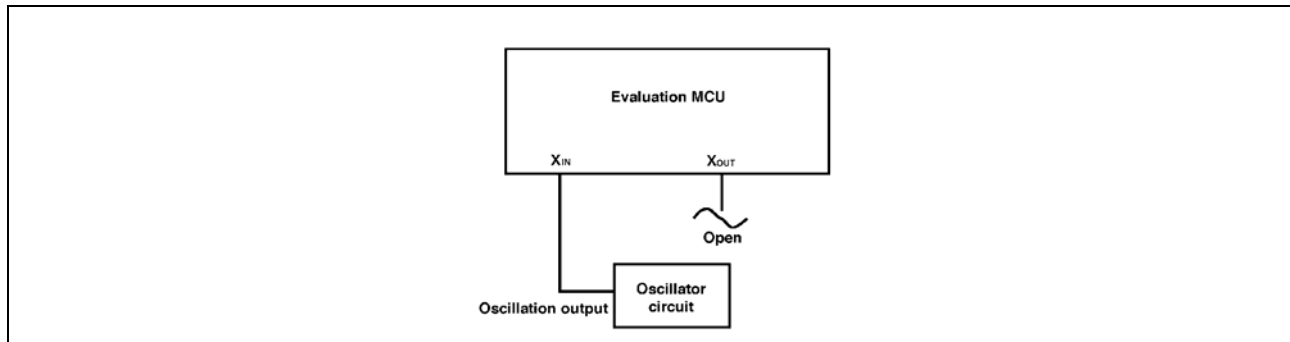


Figure 2.22 External oscillator circuit

Make note that in the oscillator circuit shown in Figure 2.23 where a resonator is connected between pins  $X_{IN}$  and  $X_{OUT}$ , oscillation does not occur because a converter board and other devices are used between the evaluation MCU and the user system. It is same for sub-clock oscillator circuits ( $X_{CIN}$  and  $X_{COUT}$ ).

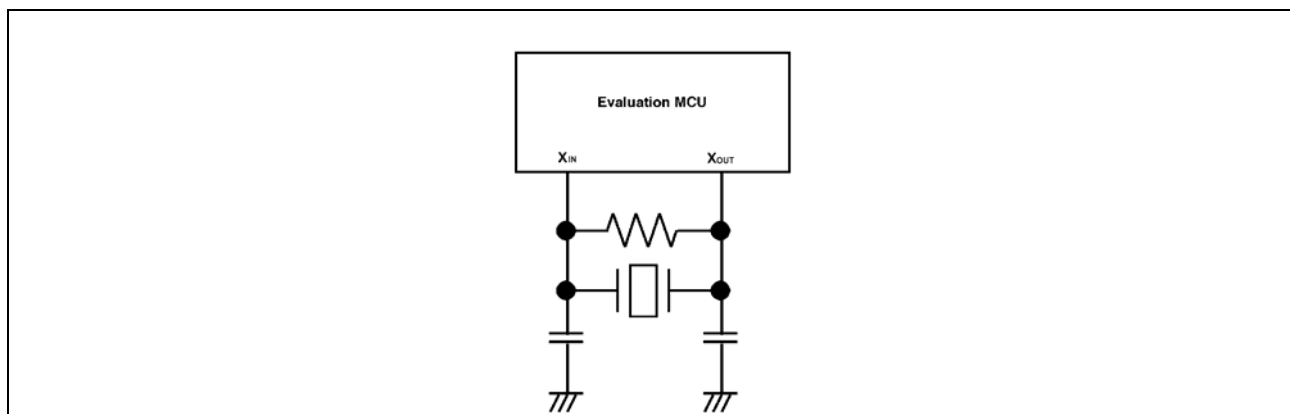


Figure 2.23 Circuit in which oscillation does not occur

### 2.11.4 Using the Internal Generator Circuit

The dedicated circuit in the PC7501 can generate any arbitrary frequency specified by the emulator debugger, and it can be supplied as a main clock. It does not depend on either the oscillator circuit board in the PC7501 or the oscillator circuit on the user system. If you want to debug programs without the user system or change a frequency temporarily, you can check its operation before purchasing an oscillator. If you want to use the internal generator circuit in the PC7501 as a main clock, choose "Generate" in the emulator debugger and specify a frequency you like to use this clock.

Although you can change a frequency between 1.0 and 99.9 MHz by 0.1 MHz for the PC7501, do not specify a value exceeding the maximum input frequency of the  $X_{IN}$  of an MCU.

## IMPORTANT

### Notes on Using the Internal Generator Circuit:

- The internal generator circuit is equipped for temporary debugging purposes. Temperature characteristics of frequencies are not guaranteed.
- Be sure to evaluate your system with an oscillator or oscillator module whose frequency is same as that of the internal oscillator circuit (internal clock) for final evaluation purposes.

## 3. Usage (How to Use the Emulator Debugger)

This chapter describes how to start up the emulator debugger from the High-performance Embedded Workshop.

### 3.1 Starting Up the Emulator Debugger

When debugging the completed programs, switch the session. The session can be changed by the drop down list of the tool bar shown below.



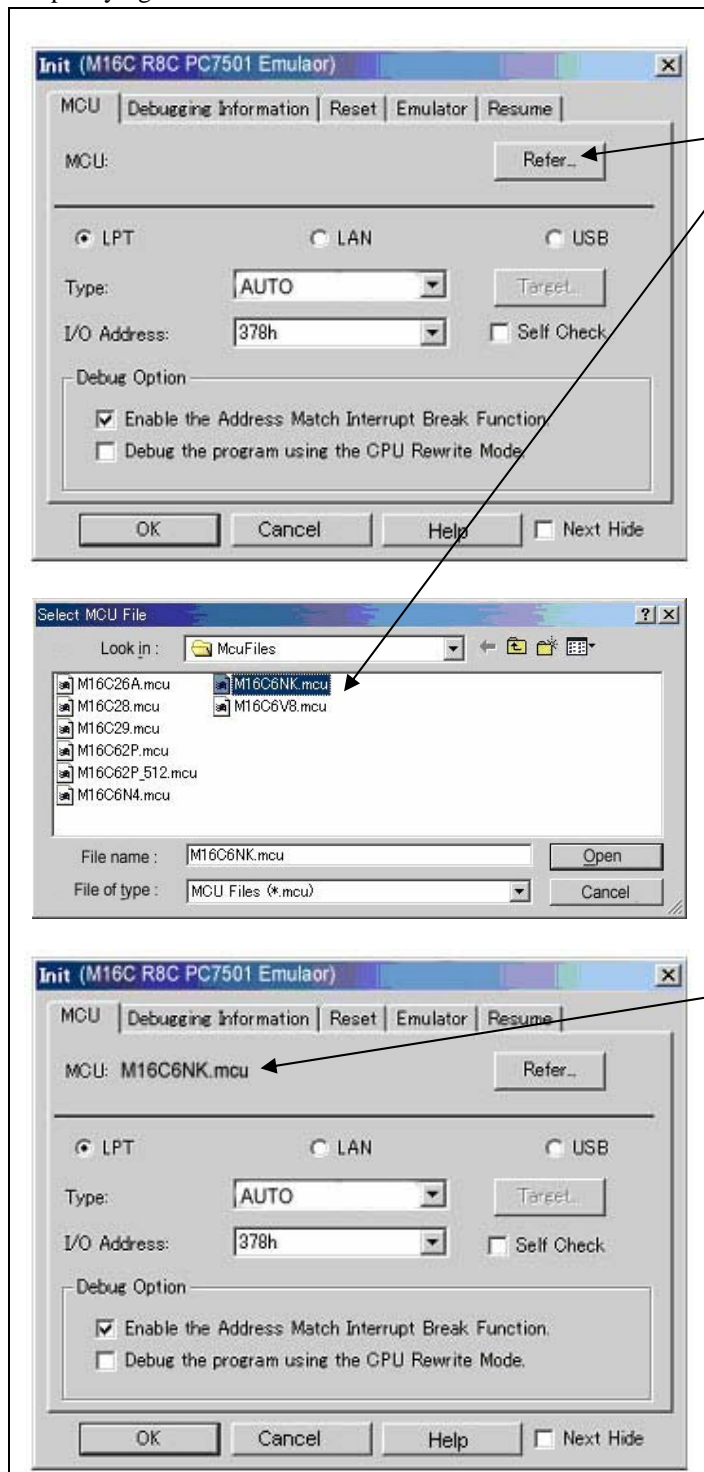
You will have as many sessions created as the number of targets you selected when creating a project, so select the session that corresponds to the target to be connected from the drop-down list. Select "SessionM16C\_R8C\_PC7501\_Emulator" in the list.

#### 3.1.1 INIT Dialog Box

After specifying the session, an Init dialog box to connect the target is displayed. The Init dialog box is used to set the items that need to be set when the emulator debugger starts up. The contents set in this dialog box remain effective the next time you start the debugger.

(1) MCU tab

1. Specifying the MCU file



Specifying the MCU file

Specify the MCU file for the target MCU.

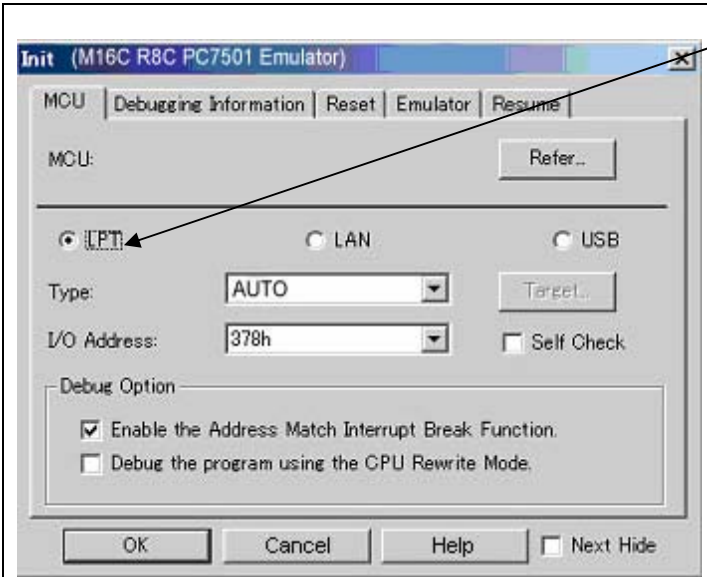
- (1) Click the "Refer..." Button.
- (2) "Select MCU File" dialog box will be displayed. Select "M16C6NK.MCU".

- The MCU file contains the information specific to the target MCU.

- The specified MCU file is displayed at the MCU area field of the MCU tab.

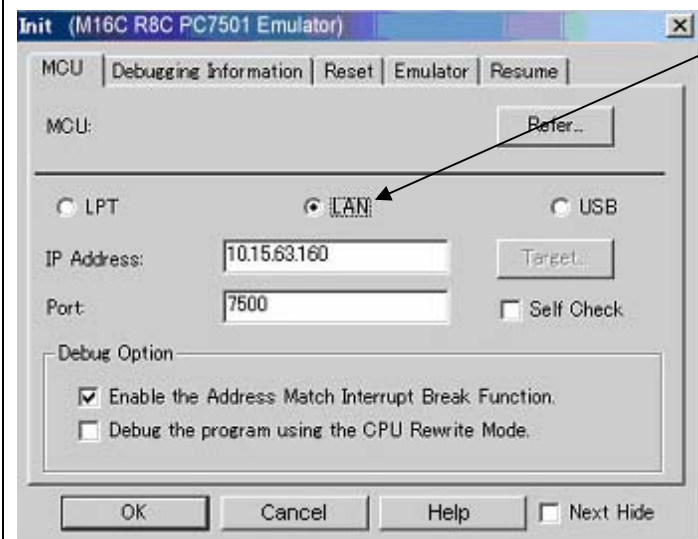


2. Specifying the communication interface



Specifying the communication interface (LPT)

- For selecting the LPT communication, click the radio button “LPT” of the MCU tab.
- For the Type area, specify LPT interface communication mode to be used. **When you use the emulator debugger for the first time, select “AUTO”.**
- Specify the I/O address of the parallel port at the I/O address area.
- For BIOS setup, the following addresses are available.
  - 378h
  - 278h



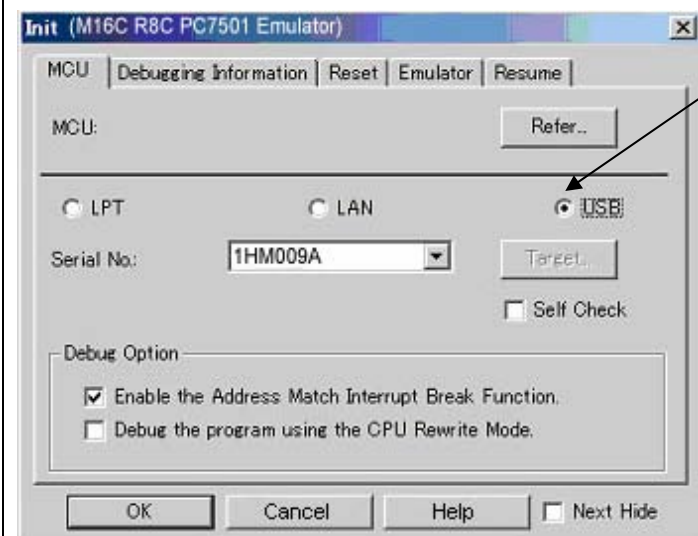
Specifying the communication interface (LAN)

- For selecting the LAN communication, click the radio button “LAN” of the MCU tab.
- Specify the IP address of the emulator at the IP address area.
- Specify the IP address by one byte of decimal number separating each four bytes with a period.
- Specify the port number at the Port area.

[Notes]

When you are using the LAN communication for the first time, connect the emulator through other than LAN communication interface and set the IP address, or use the SETIP utility included with your product to set the IP address.

For details about the SETIP utility, refer to the on-line manual of your emulator debugger.



Specifying the communication interface (USB)

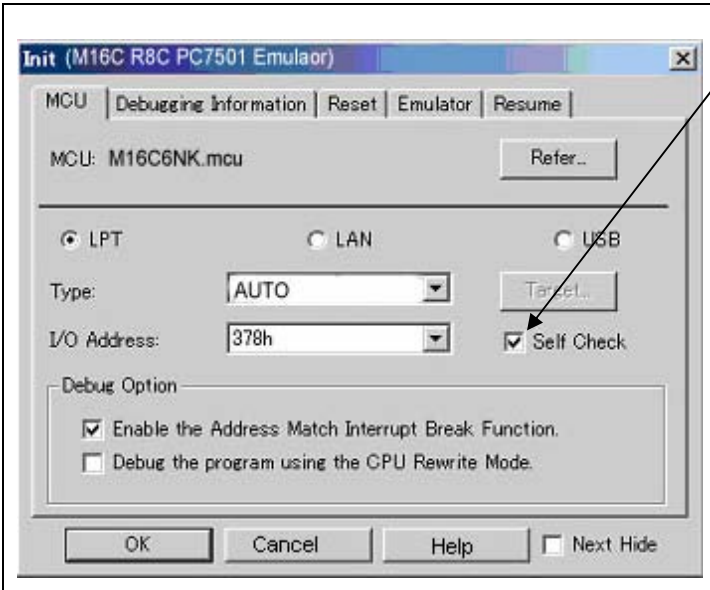
- For selecting the USB communication, click the radio button “USB” of the MCU tab.
- The emulators connected by a USB cable are shown at Serial No. area.
- Select the serial No. of the emulator you are going to connect.

[Notes]

When you are using USB communication for the first time, you must install the USB driver first.

The USB devices are detected when you turn on the power for the emulator, and a wizard for installing the corresponding device driver will automatically be started. Follow the instructions of the wizard to complete installation.

3. Executing the self-check



Executing the self-check

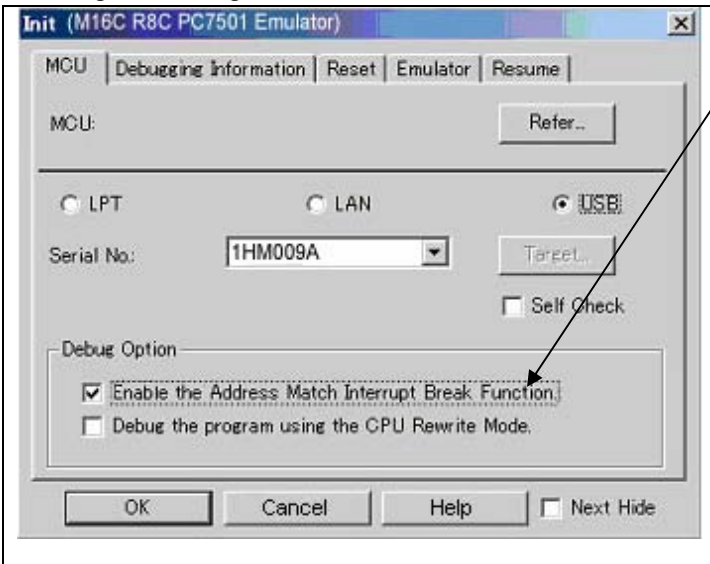
Enable this function when you want the emulator to be self-checked at startup. Be sure to select the check box only when you want the emulator to be self-checked at startup.

This function may be enabled in the following cases:

- When you are using the emulator you have just installed.
- When you successfully download the firmware, but fail to start up the emulator.
- When you want to confirm whether the emulator is operating normally because, for example, the MCU runs out of control or something is wrong with the trace results.

This function can be enabled only when you are starting up the emulator debugger.

4. Using or not using the address match break function



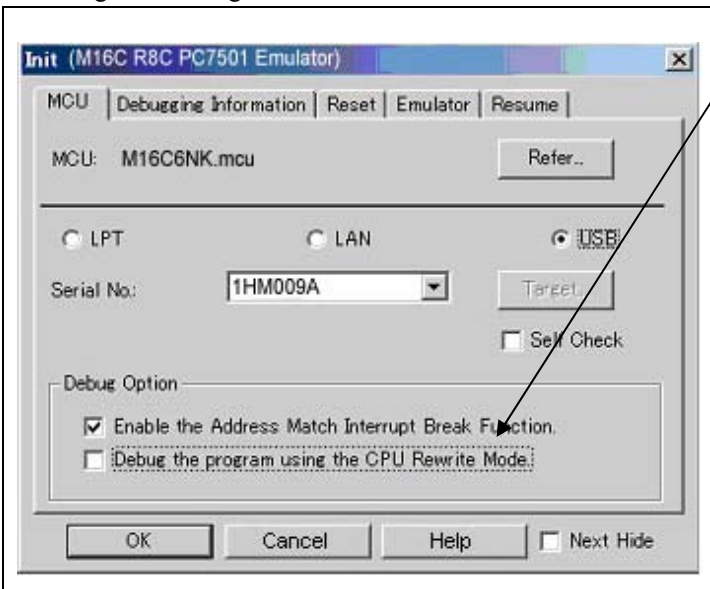
Using or not using the address match break function

Specify whether or not to use the address match break function.

- To use the address match break function (default)  
Check this box. In this case, the address match interrupt is used by the emulator, and cannot be used in the user program.
- When not using the address match break function  
Deselect the check box. In this case, the address match interrupt can be used in the user program.

This option can be selected or deselected only when you are starting up the emulator debugger.

5. Using or not using CPU rewrite mode



Using or not using CPU rewrite mode

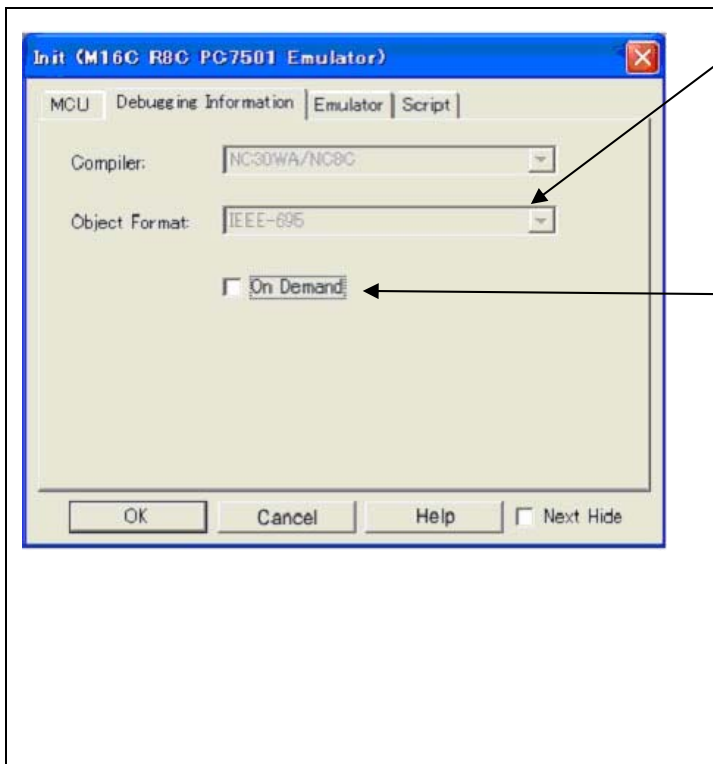
Specify whether or not to debug in CPU rewrite mode. To debug the user system that uses CPU rewrite mode, check this box. This option can be selected or deselected only when you are starting up the emulator debugger.

[Notes]

When debugging in CPU rewrite mode is enabled, the following functions cannot be used while executing the user program:

- Setting the address match breakpoint
- Software breakpoint setting in the internal ROM area
- Execution of COME in the internal ROM area
- Setting the hardware breakpoint and trace point

## (2) Debugging Information tab

**Specifying the compiler and the object format**

Specify the compiler you are using and the format of the object file output by the compiler.

To alter this setting, use the dialog box that is brought up by selecting “Debug” and then “Debug Settings...” from the menu.

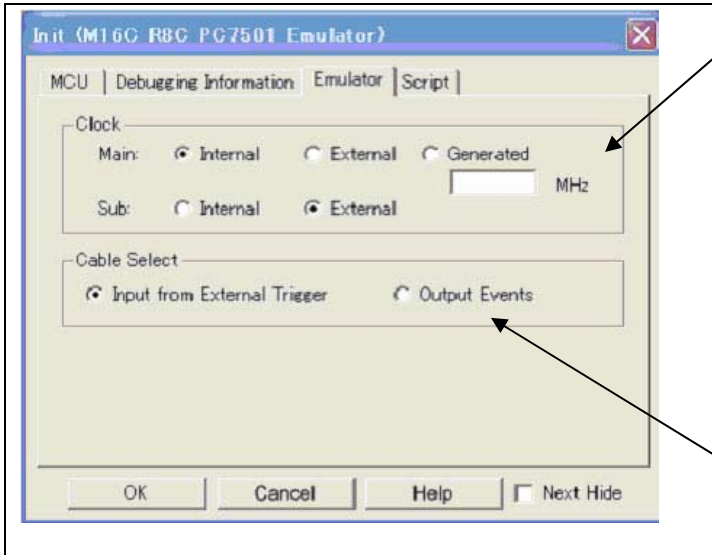
**Specifying the method for storing debug information**

There are two methods for storing debug information: on-memory method and an on-demand method. Select method for storing debug information. (By default, the on memory is selected)

When selecting the on demand method, check the “On Demand” check box.

- On Memory  
Debugging information is stored in the internal memory of your computer. This method is suitable when the load module (user program) size is small.
- On Demand  
Debugging information is stored in a reusable temporary file on the hard disk of your computer.  
Because the stored debugging information is reused, the next time you download the same load module it can be downloaded at a high speed. This method is suitable when the load module (user program) size is large.

(3) Emulator tab



**Specifying the target clock**  
 Specify the clock sources supplied to the MCU (main clock and sub clock). Select the appropriate clock sources according to the clock used by your target MCU.

- Internal (default)  
 Emulator PC7501's internal clock
- External  
 User system clock
- Generated  
 Clock created in PC7501

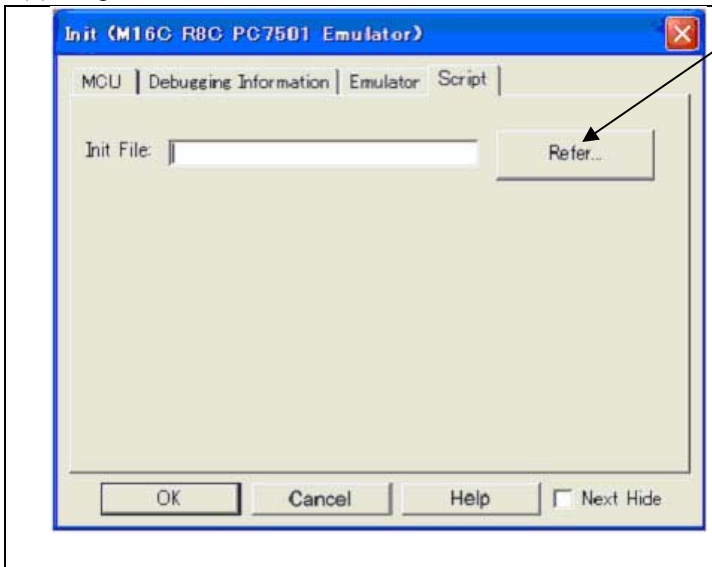
The option you have specified here remains effective the next time you start up.

**Selecting the event output/trigger input cable**  
 Select input/output of the event output/trigger input cable of the PC7501.

- Input from External Trigger (default)  
 Inputs the external trigger from the cable
- Output Events  
 Outputs events from the cable

“Input from External Trigger” is set at startup (the previous setting becomes null).

(4) Script tab



**Automatically executing a script command**  
 To automatically execute a script command when starting up the debugger, click the “Refer...” button and specify the script file to be executed.

Clicking the “Refer...” button brings up a file selection dialog box. The script file you have selected is displayed in the Init File: section of the dialog box shown here. If you do not want to automatically execute a script command, delete the character string displayed in the Init File: section of the dialog box.

What you specify here is reflected only at startup. If you specify back again in the Init dialog box after startup, whatever you specified has no effect. (Be sure to restart the emulator debugger.)

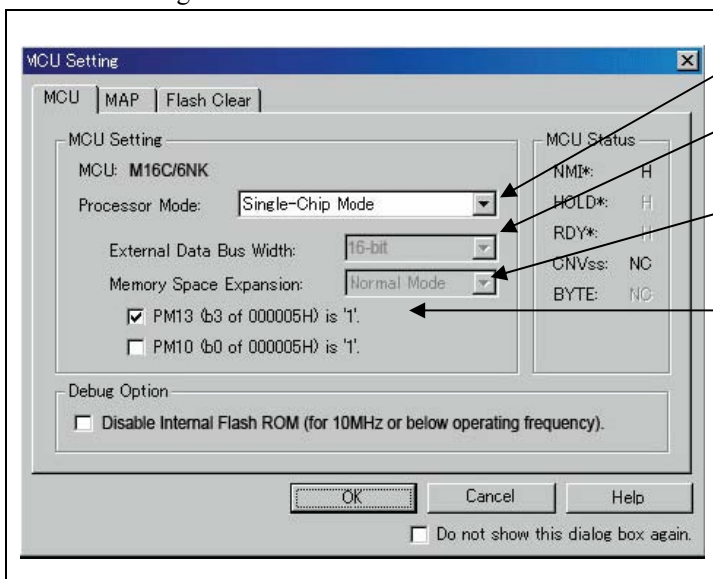
When the tab settings of (1) to (4) have been finished, click “OK”.

### 3.1.2 MCU Setting Dialog Box

MCU setting dialog box sets the information of the user system. It will be displayed after closing the Init dialog box.

#### (1) MCU tab

##### 1. MCU Setting



#### Specifying the processor mode

Select the appropriate processor mode that suits your system.

#### Specifying the bus width

When specifying memory expansion mode or microprocessor mode, select "16-bit" or "8-bit" for "External Data Bus Width".

#### Using or not using the memory space expansion function

Because the M306NKT-EPB does not support the memory space expansion function, select "Normal Mode".

#### Using or not using the internal reserved area expansion

Specifies whether or not to expand the internal reserved area. Check the box if the internal reserved area expansion bit (PM13) is "1".

PM13= 0:

Internal RAM area: 00400h--03FFFh

Internal ROM area: D0000h--FFFFFh

PM13= 1:

Internal RAM area: 00400h--07FFFh

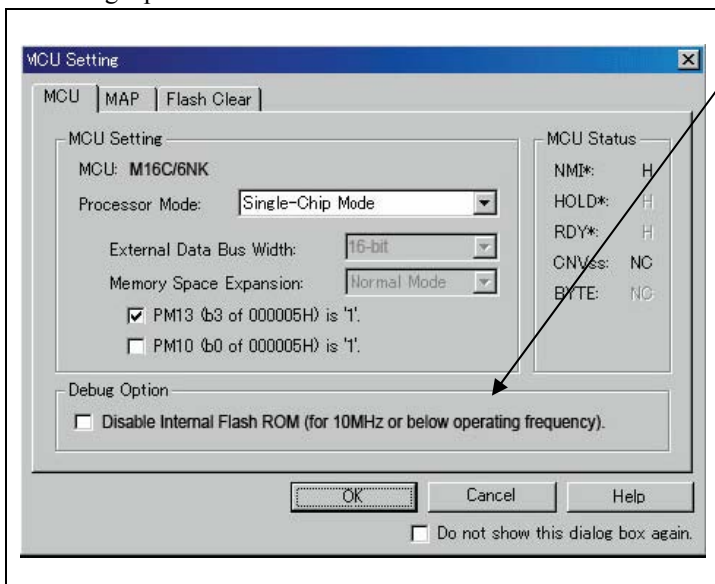
Internal ROM area: 80000h--FFFFFh

## IMPORTANT

#### Notes on Selecting a Processor Mode:

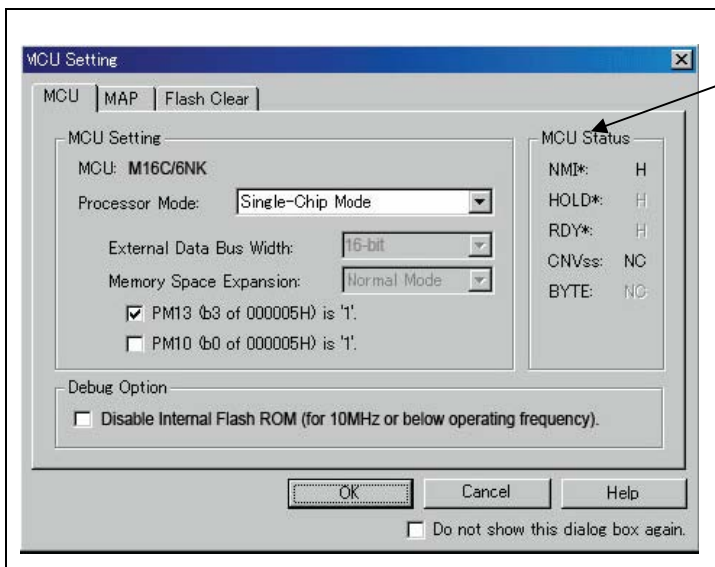
- When setting single-chip mode or memory expansion mode, the level of pin CNVSS of the MCU status should be "L". The MCU status shows the pin level of the user system.
- When setting microprocessor mode, the level of pin CNVSS of the MCU status should be "H".
- When setting memory expansion mode or microprocessor mode, pins RDY# and HOLD# of the user system should be "H".
- When the user system is not connected, all modes are available.

2. Debug Option



**Setting the debug option**  
 Check the box when not downloading the program to the internal flash ROM of the MCU. If this box is checked, the working efficiency (downloading, software break, etc.) will be improved because the program is downloaded in the emulation memory in the PC7501. However, maximum operating frequency is 10MHz. The box is unchecked by default.

3. MCU Status



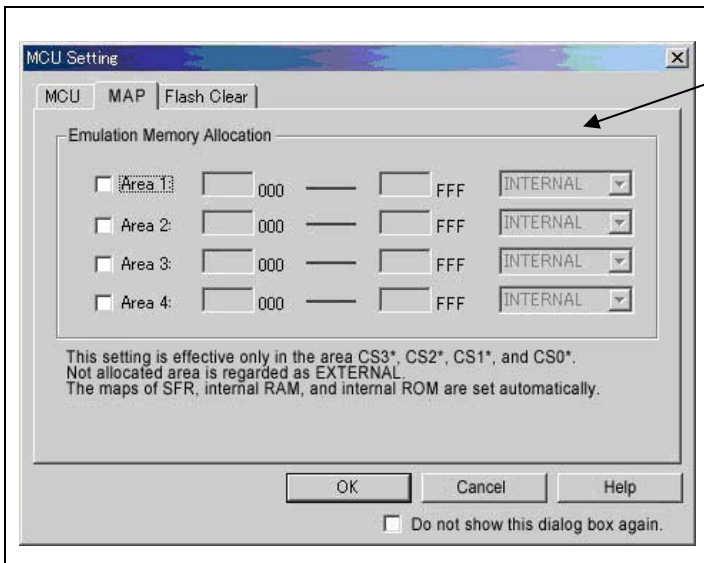
**Referring to the MCU Status**  
 It shows the status of each MCU pin. Check it to see if the MCU status matches the selected processor mode. "NC" means that the pin status is indeterminate.

**IMPORTANT**

Notes on Selecting a Processor Mode:

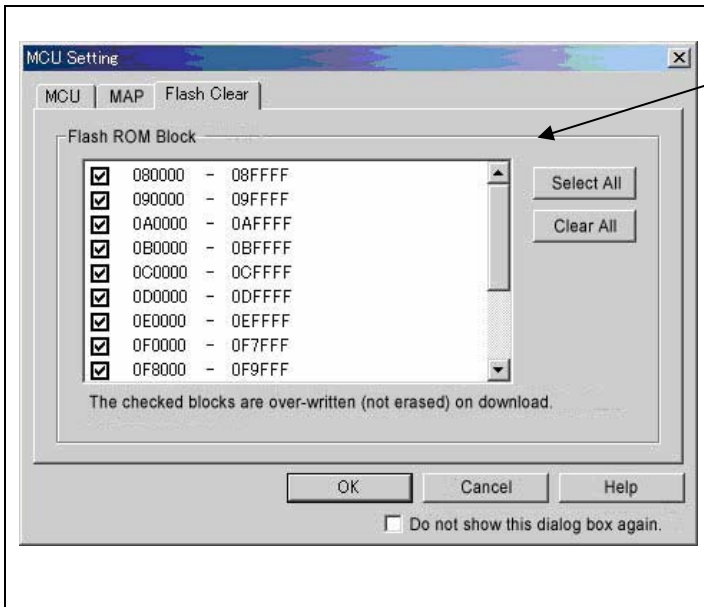
- The MCU status is shown in "MCU Status" of the MCU setting dialog box. Check that it matches to the processor mode to be set.
- When setting memory expansion mode or microprocessor mode, pins RDY# and HOLD# of the user system should be "H".

(2) MAP tab



**Specifying the emulation memory allocation**  
 Sets the memory area assigned to the emulation memory by 4KB. Up to four areas can be set.  
 An area set to "INTERNAL" will be allocated to the emulation memory by selecting the check box. An unchecked area or an unset area will be allocated to external area. Not allocated area is regarded as external area.  
 The setting is effective only in the CS3#, CS2#, CS1# and CS0#. The maps of SFR, internal ROM and RAM are set automatically.

(3) Flash Clear tab



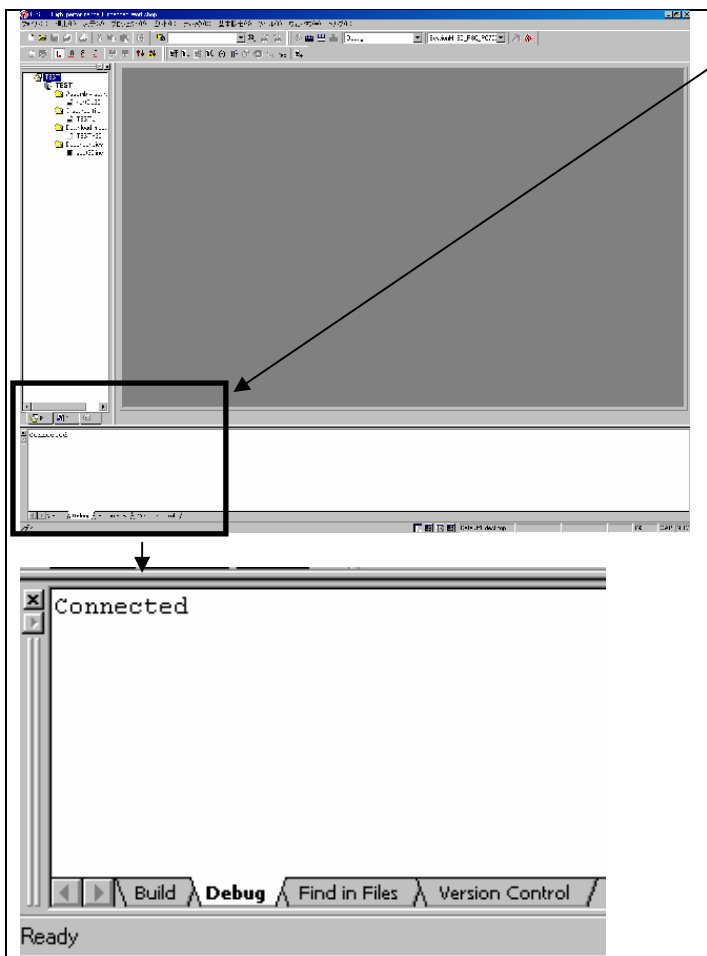
**Setting to clear the MCU's internal flash ROM**  
 Specify whether or not you want the MCU's internal flash ROM to be cleared when downloading the user program or data. (When cleared, the content of the flash ROM is initialized to FFh.) The MCU's internal flash ROM is listed in block units.

- Any block which has had its check box selected is not cleared when downloading. The memory content of this block remains intact unless overwritten by downloading.
- Any block which has had its check box deselected is cleared when downloading.
- Click the Select All button, and all blocks will be selected (marked by a check mark, so that none of the blocks is cleared when downloading).
- Click the Clear All button, and all blocks will be deselected (check marks removed, so that all of them are cleared when downloading).

The option you have specified here remains effective the next time you start up.

When the tab settings of (1) to (3) have been finished, click "OK".

## 3.1.3 Checking Connections of the Emulator System

**Checking connections of the emulator system**

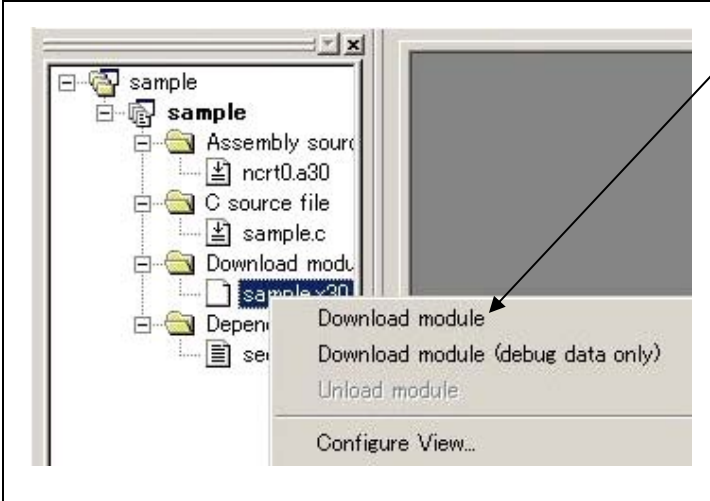
When the emulator debugger is connected correctly to the emulator after you have finished setting up the Init dialog box and the MCU Setting dialog box, you will see a message “Connected” displayed on the “Debug” tab of the Output window.



### 3.2 Editor (Source) Window

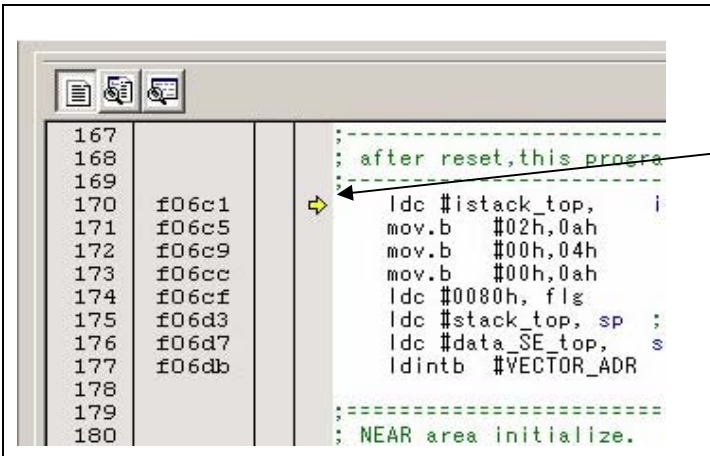
Downloading the program to be debugged.

#### (1) Downloading from the work space window



**Downloading the program**  
Download the object program you want to debug.  
Select Download from “xxx.x30” of “Download module”.  
Or you can select “Download” from the “Debug” menu for the same effect.

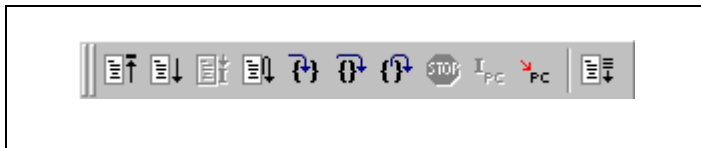
#### (2) Showing the program











**Editor (Source) window**  
The editor (source) window is a window that always shows the content of the source file corresponding to the current position of the program counter.  
The program counter position is marked by a yellow arrow. Here, you can execute the program up to the cursor position, set or clear software breakpoints, and line-assemble the source file.  
Because this product uses the MCU's internal flash ROM, the initial value for the ROM area data at the time of purchase is “FFh.”

### 3.3 Program Execution

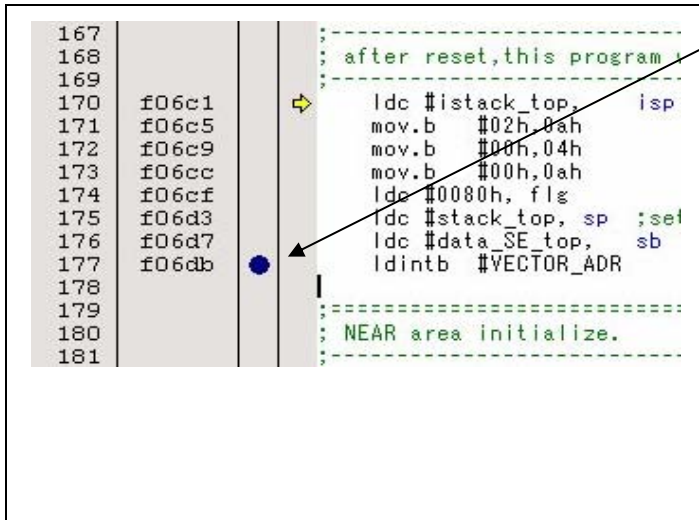
#### (1) Program execution



	<b>CPU reset</b>	Resets the target MCU.
	<b>Go</b>	Runs the program beginning with the current PC position.
	<b>Reset Go</b>	Runs the program after reset the target MCU.
	<b>Step In</b>	Single-steps the program executing each statement (including those in functions).
	<b>Step Over</b>	Single-steps the program executing each function call as one step.
	<b>Step Out</b>	Exists a function and stops at a statement in the program next to the one that called the function.
	<b>STOP</b>	Stops the program.
	<b>Go Free</b>	Runs the program beginning with the current PC position. The software and hardware breaks set in it are ignored.

(2) Setting address match breakpoints


1. Setting and executing address match breakpoint



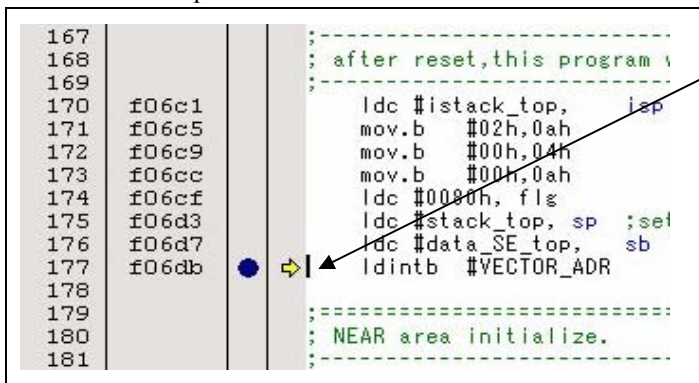
**Program execution with address match breakpoint**

An address match breakpoint can be set by selecting “Enable the Address Match Interrupt Break Function” in Init dialog box of the MCU tab.

- A software breakpoint can be set or cleared by double-clicking the address match breakpoint display area in the editor (source) window (A blue circle is displayed at the setting line). Up to eight points can be set.

-  Click the execution button.

2. Execution completed with address match break



The cursor stops in the position where the address match break was set up.


The statement in which the address match break was set up is not executed.

(3) Setting software breakpoints

1. Setting and executing software breakpoint

167			; after reset, this program
168			;
169			;
170	f06c1	↳	ldc #istack_top, isp
171	f06c5		mov.b #02h,0ah
172	f06c9		mov.b #00h,04h
173	f06cc		mov.b #00h,0ah
174	f06cf		ldc #0080h, flg
175	f06d3		ldc #stack_top, sp ;sel
176	f06d7		ldc #data_SE_top, sb
177	f06db	●	ldintb #VECTOR_ADR
178			;
179			;
180			; NEAR area initialize.
181			;

**Program execution with software breakpoint**

- A software breakpoint can be set or cleared by double-clicking the software breakpoint display area in the editor (source) window (A red circle is displayed at the setting line).
-  Click the execution button.

2. Execution completed with software break

167			; after reset, this program
168			;
169			;
170	f06c1		ldc #istack_top, isp
171	f06c5		mov.b #02h,0ah
172	f06c9		mov.b #00h,04h
173	f06cc		mov.b #00h,0ah
174	f06cf		ldc #0080h, flg
175	f06d3		ldc #stack_top, sp ;sel
176	f06d7		ldc #data_SE_top, sb
177	f06db	●	ldintb #VECTOR_ADR
178			;
179			;
180			; NEAR area initialize.
181			;


The cursor stops in the position where the software break was set up. The statement in which the software break was set up is not executed.

(4) Executing up to the cursor position (Come command)

1. Specifying the Come command

```
167 | | | | ;-----;
168 | | | | ; after reset,this program \
169 | | | | ;-----;
170 | f06c1 | | | | ldc #istack_top,    isp
171 | f06c5 | | | | mov.b #02h,0ah
172 | f06c9 | | | | mov.b #00h,04h
173 | f06cc | | | | mov.b #00h,0ah
174 | f06cf | | | | ldc #0080h, flg
175 | f06d3 | | | | ldc #stack_top, sp ;set
176 | f06d7 | | | | ldc #data_SE_top, sb
177 | f06db | | | | ldintb #VECTOR_ADR
178 | | | | |
179 | | | | ;-----;
180 | | | | ; NEAR area initialize.
181 | | | | ;-----;
```

**Setup procedure for executing COME command**

- (1) Click the line in the program display area at which you want the program to execute.
- (2)  Click the Come button

2. After the Come command has finished

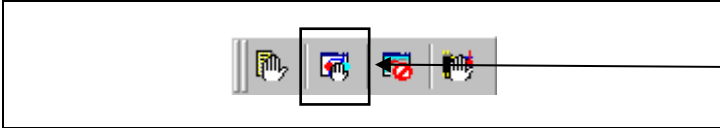
```
167 | | | | ;-----;
168 | | | | ; after reset,this program \
169 | | | | ;-----;
170 | f06c1 | | | | ldc #istack_top,    isp
171 | f06c5 | | | | mov.b #02h,0ah
172 | f06c9 | | | | mov.b #00h,04h
173 | f06cc | | | | mov.b #00h,0ah
174 | f06cf | | | | ldc #0080h, flg
175 | f06d3 | | | | ldc #stack_top, sp ;set
176 | f06d7 | | | | ldc #data_SE_top, sb
177 | f06db | | | | ldintb #VECTOR_ADR
178 | | | | |
179 | | | | ;-----;
180 | | | | ; NEAR area initialize.
181 | | | | ;-----;
```

The cursor stops in the position where the come command was executed.  
The statement specified with the come command is not executed.

### 3.3 Hardware Breakpoint Setting Window

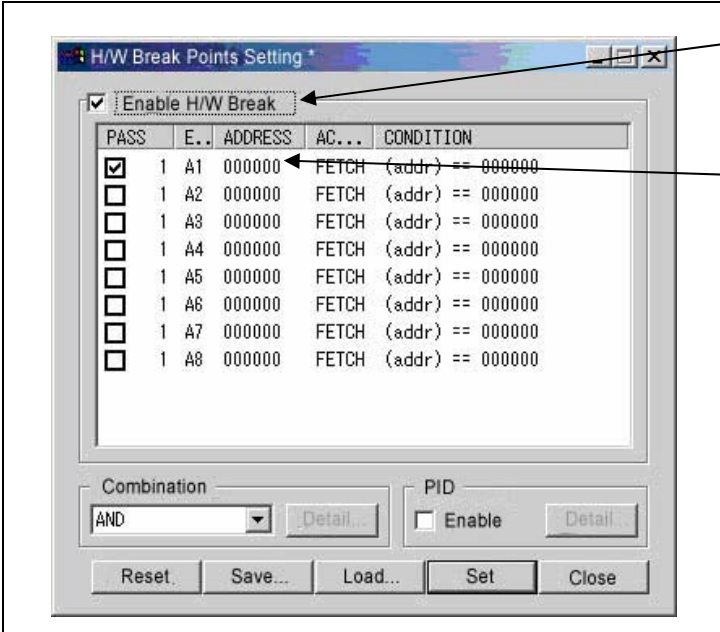
(1) Breakpoint setup dialog box

1. Opening the hardware breakpoint setup dialog box



**Hardware breakpoint**  
Clicking this button opens the hardware breakpoint setup dialog box.

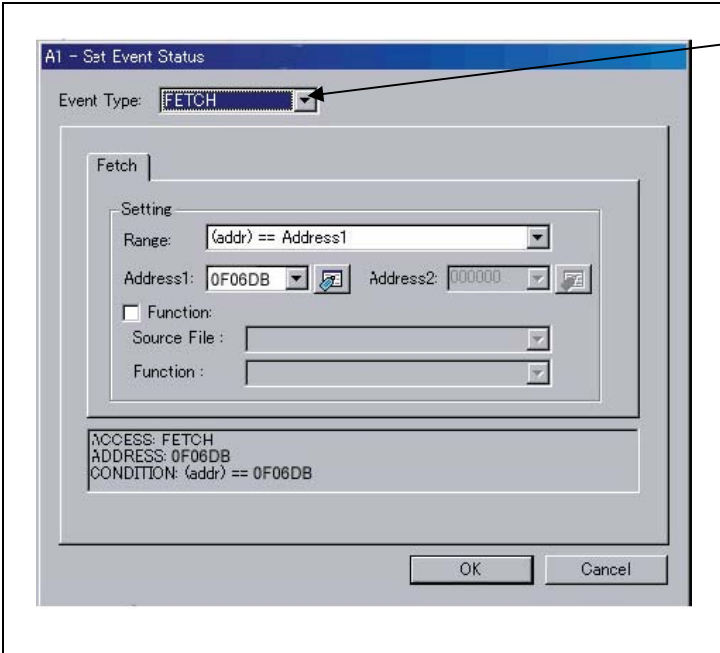
2. Hardware breakpoint setting window



**H/W Break Point Setting window in initial state**  
Select the "Enable H/W Break" check box, and this break function will be enabled, allowing you to set hardware breakpoints.

**Setting break event**  
Click the event line at which you want set a break event.

3. Opening the break event setting dialog box

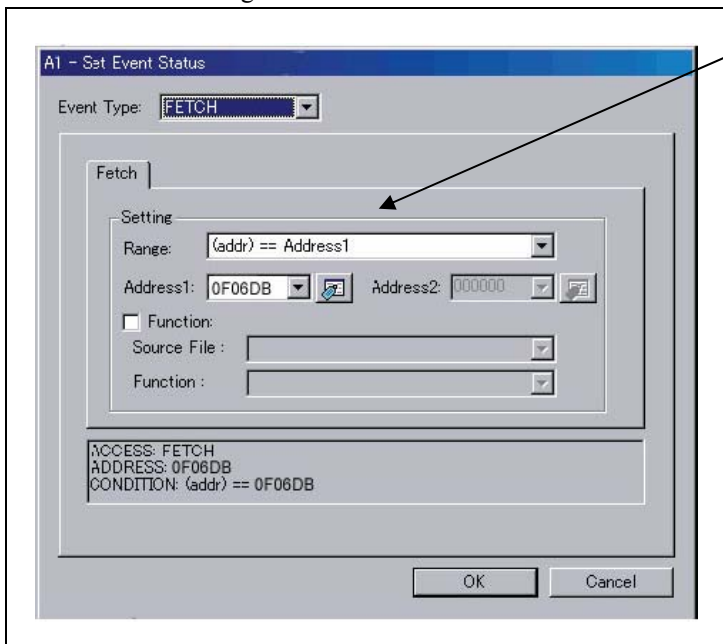


**Specifying the event type**  
Click to select the event type you want to set.

- FETCH  
Detects an instruction prefetch.
- DATA ACCESS  
Detects a memory access.
- BIT SYMBOL  
Detects a bit access.
- INTERRUPT  
Detects an interrupt occurrence or interrupt termination.
- TRIGGER  
Detects a signal from the external trace signal input cable.

(2) When FETCH is selected

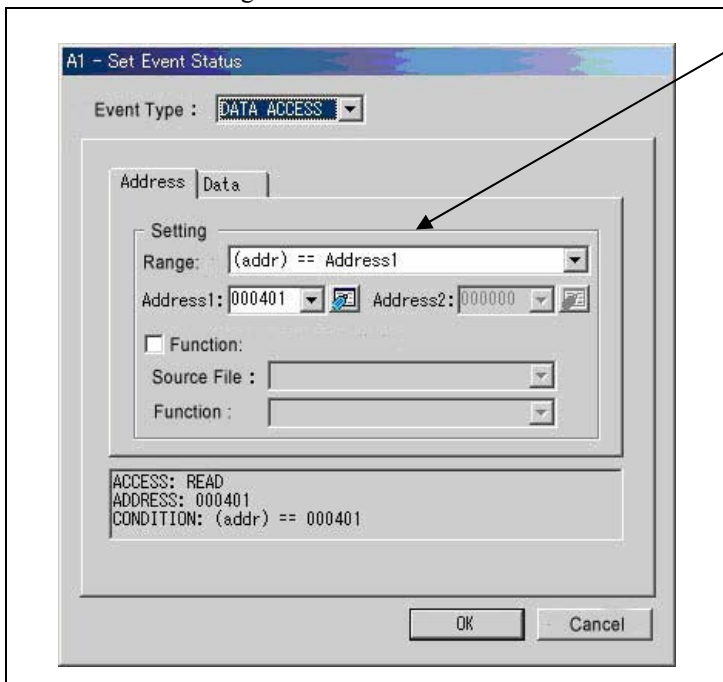
1. Window for setting addresses



**Setting the address**  
You can set eight conditions, e.g., a specified address, a specified address range, etc. When you have finished setting the address, click OK.

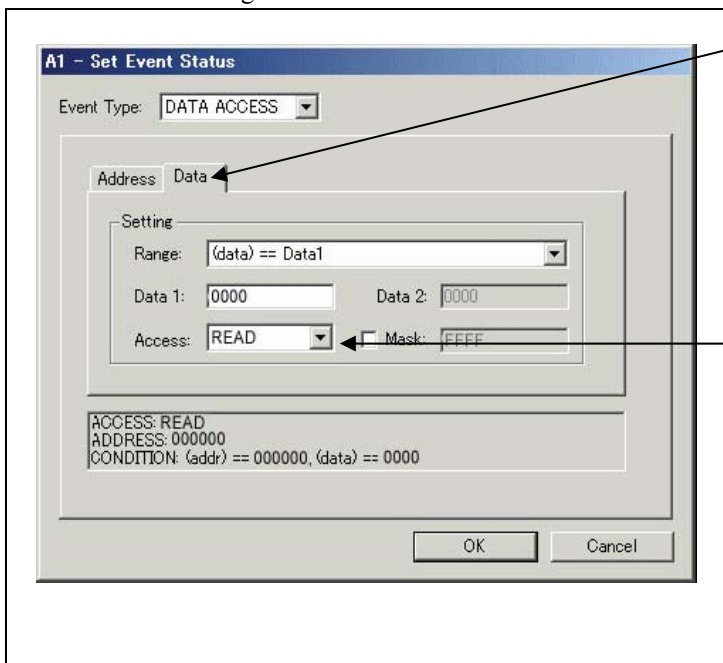
(3) When DATA ACCESS is selected

1. Window for setting the address



**Setting the address**  
 Set in the Address tab.  
 You can set eight conditions, e.g., a specified address, a specified address range, etc.  
 After finishing the setting the address, set the data.

2. Window for setting data



**Setting data**  
 Set in the Data tab.  
 You can set eight conditions, e.g., a specified data, a specified data range, etc.

**Setting the access condition**  
 You can set three conditions, e.g., read, write, and read/write. When you have finished setting the data and access condition, click OK.



3. Example Data Settings

Setting events for wordwise accesses to the even-addresses

STE.W A0,20E8h(A0=5423h)

Cycle	Label	Address	Data	BUS	BHE	BIU	R/W	RWT	CPU
-00023		0020E8	5423	16b	0	DW	W	0	CW

High-order and low-order data effective

Setting a break event

A1  
 Address 1 : 0020E8  
 Data 1 : 5423  
 MASK : FFFF  
 Access : WRITE

Setting events for wordwise accesses to the odd-addresses

STE.W A0,20E5h(A0=AB79h)

Cycle	Label	Address	Data	BUS	BHE	BIU	R/W	RWT	CPU
-00021		0020E5	79AB	16b	0	DW	W	0	CB
-00020		0020E6	79AB	16b	1	DW	W	0	CB

Odd-address high-order data effective  
 Even-address low-order data effective

Setting a break event (using 2 events)

A1                                  A2  
 Address 1 : 0020E5      Address 1 : 0020E6  
 Data 1 : 7900            Data 1 : 00AB  
 MASK : FF00            MASK : 00FF  
 Access : WRITE          Access : WRITE

Set the combinatorial events to AND.

Setting events for wordwise accesses to the even-addresses

STE.B R0L,[A1A0](R0L=E5h,A1=0000h,A0=20E2h)

Cycle	Label	Address	Data	BUS	BHE	BIU	R/W	RWT	CPU
-00022		0020E2	00E5	16b	1	DB	W	0	CB

Low-order data effective

Setting a break event

A1  
 Address 1 : 0020E2  
 Data 1 : 00E5  
 MASK : 00FF  
 Access : WRITE

Setting events for wordwise accesses to the odd-addresses

STE.B R0L,[A1A0](R0L=E6h,A1=0000h,A0=20E3h)

Cycle	Label	Address	Data	BUS	BHE	BIU	R/W	RWT	CPU
-00022		0020E3	E600	16b	0	DB	W	0	CB

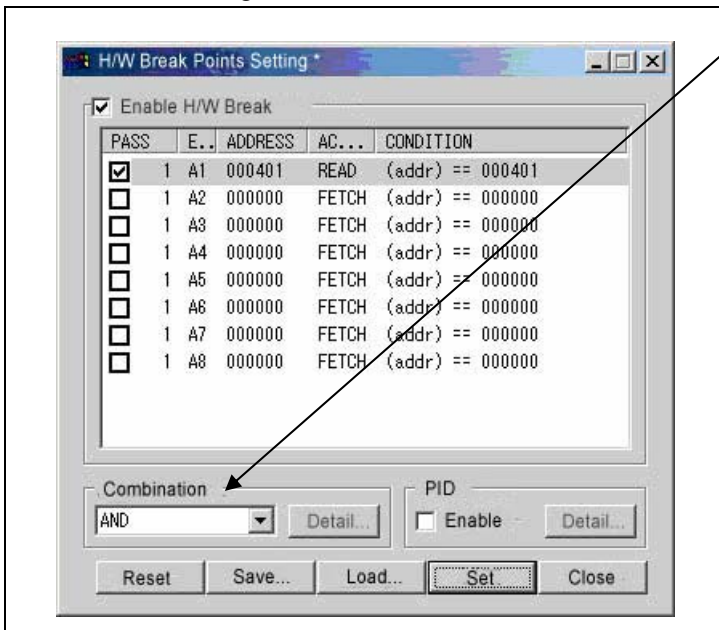
High-order data effective

Setting a break event

A1  
 Address 1 : 0020E3  
 Data 1 : E600  
 MASK : FF00  
 Access : WRITE

(4) Setting the combinatorial event condition

1. Window for setting the combinatorial event condition



Setting the combinatorial event condition

There are following four conditions that you can choose for the combinatorial events.

- AND  
 The program breaks when all of the specified events occur.
- AND (Same Time)  
 The program breaks when the specified events occur at the same time.
- OR  
 The program breaks when one of the specified events occurs.
- STATE TRANSITION  
 The program breaks when the state transition diagram goes into the break state.

For each event, a pass count (number of passing) is specifiable (1--255). When "AND (same time)" is specified for Combination, you can not specify the pass count (fixed to one).

When you have finished setting the combinatorial event condition, click the "Set" button.

## 3.5 Trace Window

The trace window is used to show the results of real-time trace measurements.

### (1) Trace window

#### 1. Opening the trace window



#### Trace

Clicking this button opens the trace window.

#### 2. Trace window display

Cycle	Label	Address	Data	BUS	BHE	BIU	R/W	RWT	CPU
-000044		0F07CE	0070	16b	0	IW	R	0	--
-000043		0F07CE	0070	16b	0	--	-	1	CW
-000042		00041A	0000	16b	0	DW	W	0	--
-000041		0F07D0	FD00	16b	0	IW	R	0	--
-000040		0F07D0	FD00	16b	0	--	-	1	RW
-000039		0F07D2	0010	16b	0	IW	R	0	CB
-000038		0F07D4	000F	16b	0	IW	R	0	RW
-000037		0F07D6	F8FF	16b	0	IW	R	0	RB
-000036		0F07D6	F8FF	16b	0	--	-	1	QC
-000035	_main	0F0010	04F3	16b	0	IW	R	0	--
-000034	_bcopy	0F0012	F27C	16b	0	IW	R	0	--
-000033		000719	D507	16b	0	DW	W	0	--
-000032		00071A	D507	16b	1	DW	W	0	--
-000031		00071B	0F00	16b	0	DB	W	0	--
-000030		00071B	0F00	16b	0	--	-	1	--
-000029		00071B	0F00	16b	0	--	-	1	CB
-000028		00071B	0F00	16b	0	--	-	1	--
-000027		000719	D500	16b	0	DW	R	0	--
-000026		00071A	D507	16b	1	DW	R	0	--
-000025		00071B	0F07	16b	0	DB	R	0	--
-000024		00071B	0F07	16b	0	--	-	1	QC

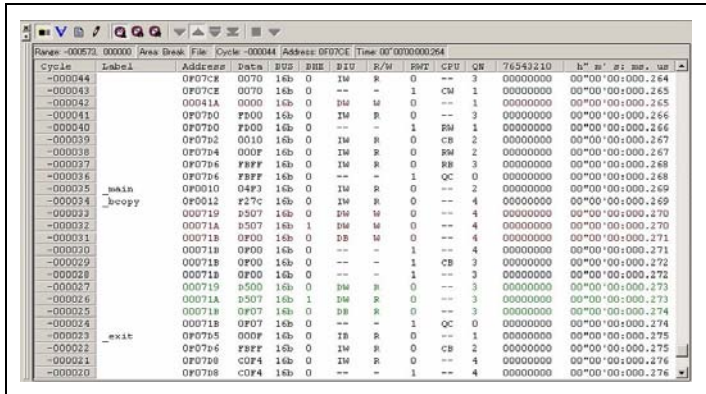
#### Trace window

The trace window is used to show the results of real-time trace measurements. It has the following four display modes. And mixed mode can be displayed.

- Bus mode  
Bus information per cycle can be inspected. The contents are displayed in order of execution paths.
- Disassemble mode  
The execution paths of the executed instructions can be inspected. The contents are displayed in order of execution paths.
- Source mode  
The execution paths of the source program can be inspected.
- Data access mode  
Data read/write cycles can be inspected.

The trace window shows the measurement result when a real-time trace measurement has finished. The trace window will be cleared when the real-time trace measurement is restarted.

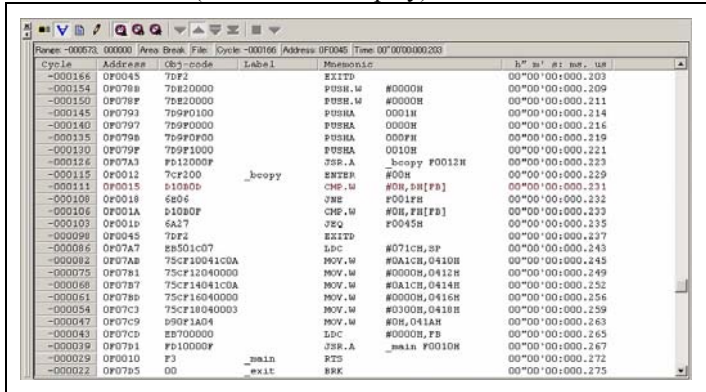
3. Trace window (Bus information display)



Explanation of the trace window (bus information display)

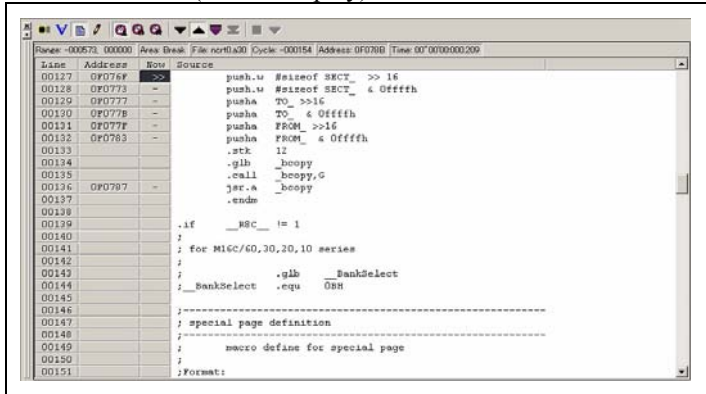
The following explains the displayed contents, from left to right.
- Address Shows the status of the address bus.
- Data Shows the status of the data bus.
- BUS Shows the width of the internal data bus. "16b" for 16 bits bus wide bus and "8b" for 8 bits bus wide bus are displayed.
- BHE Shows the status (0 or 1) of the BHE (Byte High Enable) signal. If this signal = 0, the odd-address data is valid.

4. Trace window (Disassemble display)



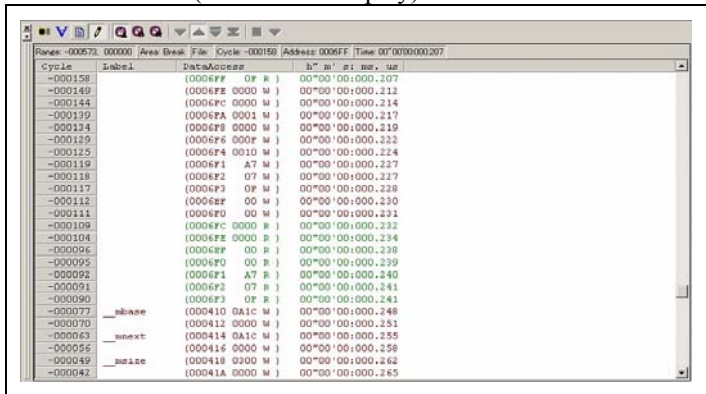
- BIU Shows the status between the BIU (Bus Interface Unit) and memory or I/O.
Symbol Status
- : No change (non-active)
DMA : Data access except for CPU
INT : Start of INTACK sequence
IB : Instruction code read (bytes) by CPU
DB : Data access (bytes) by CPU
IW : Instruction code read (words) by CPU
DW : Data access (words) by CPU
- R/W Shows the status of the data bus. Displayed as "R" for Read, "W" for Write, and "-" for no access.

5. Trace window (Source display)



- RWT This is the signal to indicate a valid bus cycle. When valid, RWT = 0. The Address, Data, and the BIU signals are effective when this signal is 0.
- CPU Shows the status between the CPU and BIU (Bus Interface Unit).
Symbol Status
- : Non-active
CB : Op-code read (bytes)
RB : Operand read (bytes)
QC : Clears instruction queue buffer
CW : Op-code read (words)
RW : Operand read (words)

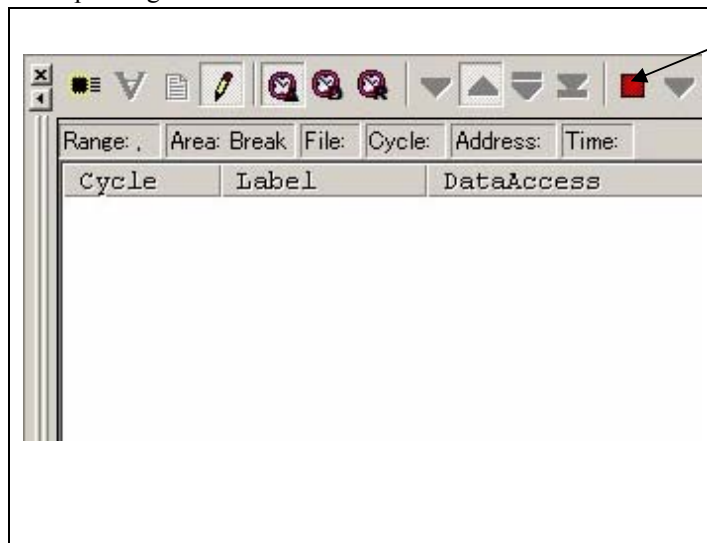
6. Trace window (Data access display)



- QN Shows the byte count stored in the instruction queue buffer. The display range is 0 to 4.
- 76543210 Shows the level of external trace signal input cable EXTIN0 to EXTIN7.
- h' m' s: ms. us Shows the elapsed time after starting the user program.

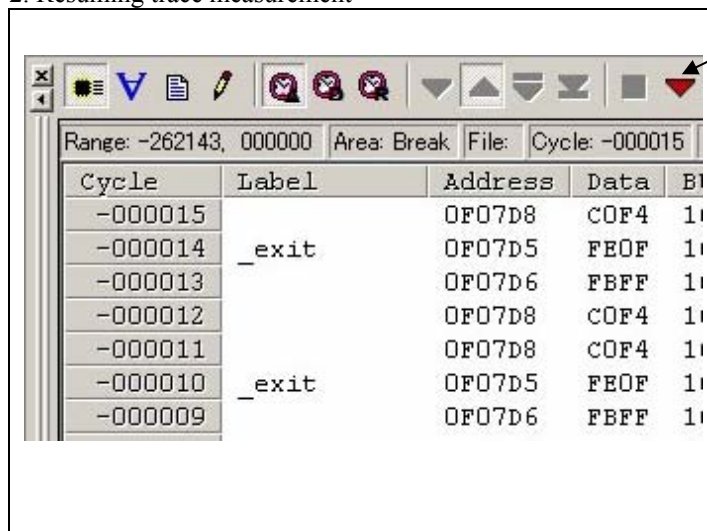
## (2) Suspending and resuming trace measurement

## 1. Suspending trace measurement

**Stop**

Click this toolbar button to suspend the trace measurement in progress.

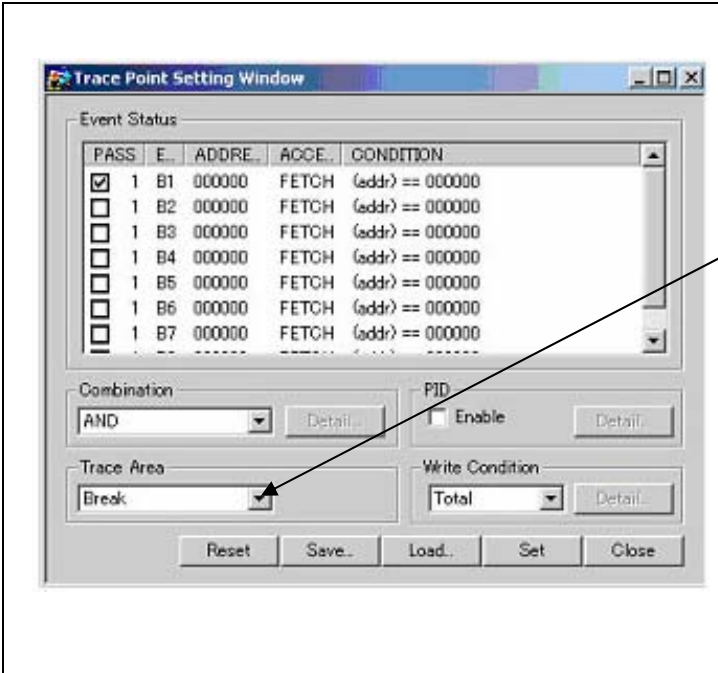
## 2. Resuming trace measurement

**Re-Start**


Click this toolbar button to resume the trace measurement in progress.

(3) Trace point setup window

1. Trace point setup window



Trace Point Setting Window in initial state

 Clicking this toolbar button opens the trace point setting window.

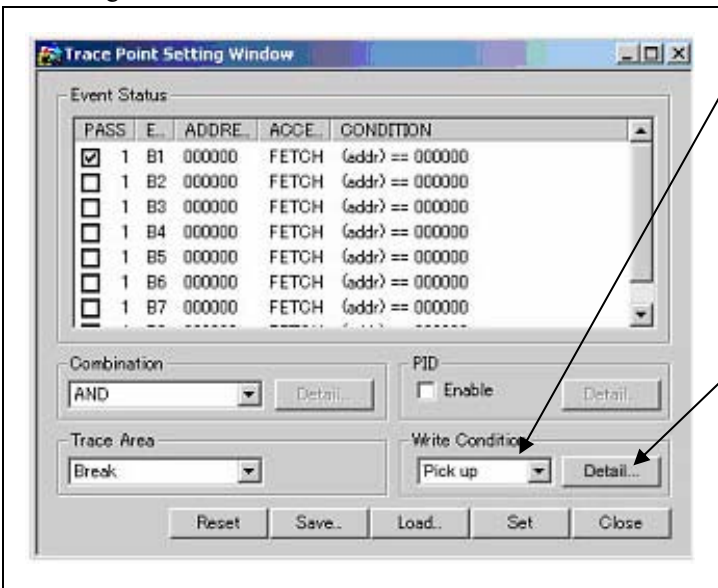
You can set events in the same way as for the hardware breakpoints.

Specifying a trace range

You can specify a trace range for the trace event.

- Break  
256K cycles of instruction execution before the user program stopped is recorded.
- Before  
256K cycles of instruction execution before a trace point condition was met is recorded.
- About  
128K cycles of instruction execution before and after a trace point condition was met is recorded.
- After  
256K cycles of instruction execution after a trace point condition was met is recorded.
- Full  
256K cycles of instruction execution after a trace began is recorded.

2. Setting the trace write condition



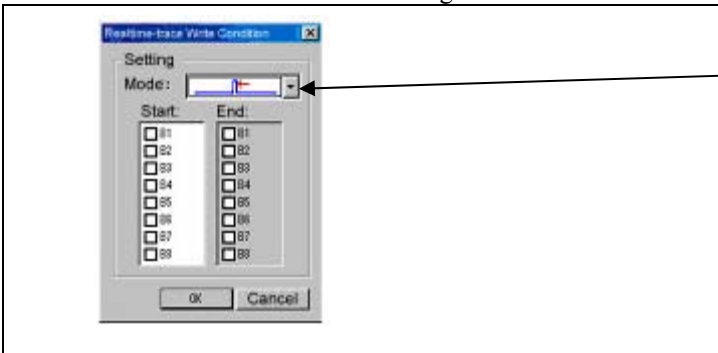
Setting the trace write condition

You can specify a condition for the cycles to be written into the trace memory.

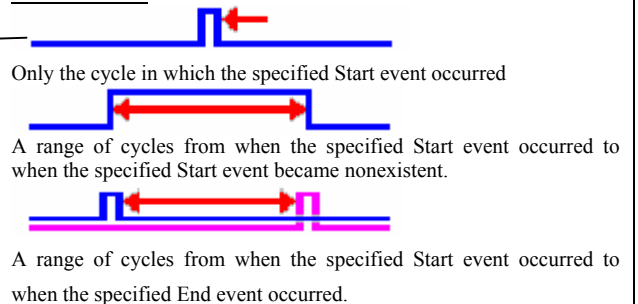
- Total  
All cycles are written into memory.
- Pick up  
Only the cycles in which the specified condition was met are written into memory.
- Exclude  
Only the cycles in which the specified condition was not met are written into memory.

When you have finished setting the trace write condition, click this button. The Realtime-trace Write Condition dialog box shown below will appear.

3. Realtime trace Write Condition dialog box



Write mode

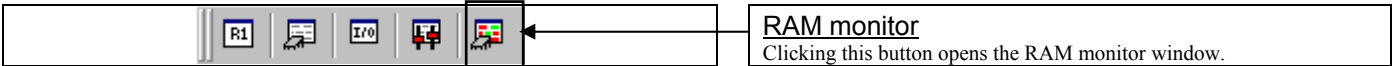


### 3.6 RAM Monitor Window

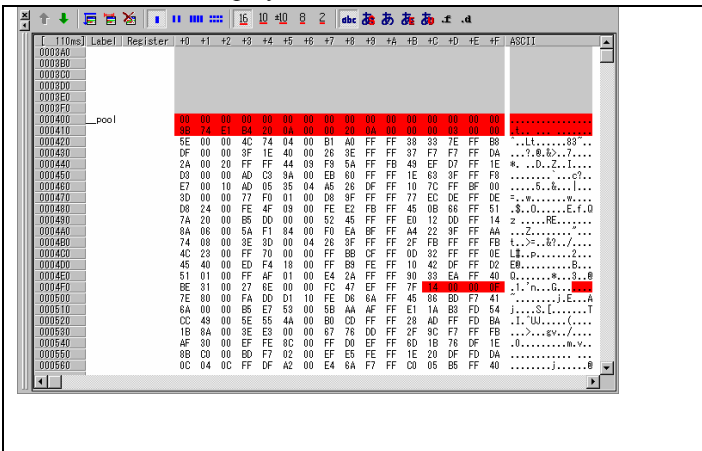
This function makes it possible to make reference to the changes in memory contents without impairing the real-time performance for target program execution. The emulator PC7501 system is provided with the 4KB RAM monitor area. This RAM monitor area can be divided into the 16-block area in units of arbitrary continuous address or 256 bytes for mapping or layout.

#### (1) RAM monitor window

##### 1. Opening the trace point setup window



##### 2. RAM monitor display



#### Changing the RAM monitor display area

You can change the manner in which the RAM monitor area you have set in the above dialog box is displayed.

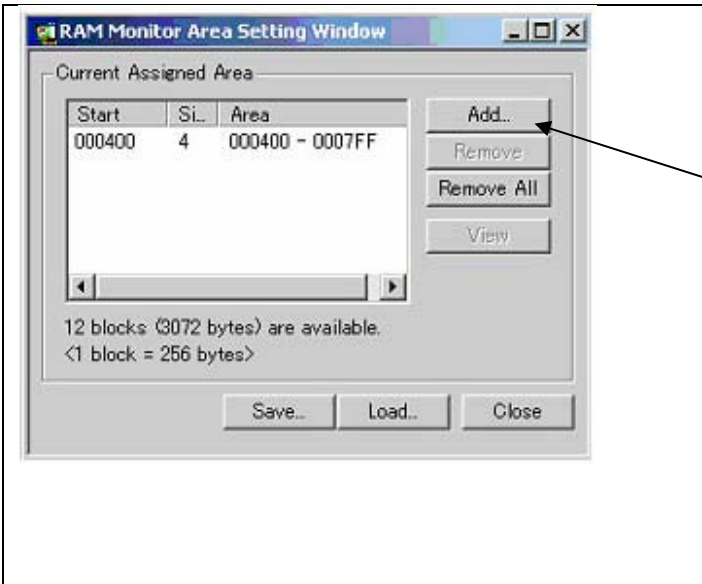
- : Shows the blocks at the preceding addresses.
- : Shows the blocks at the following addresses.

The background colors of the data display and the code display sections change with the access attribute as described below.

- Green : Addresses accessed for read
- Red : Addresses accessed for write
- White : Addresses not accessed

The background colors can be changed as necessary.

##### 3. RAM monitor area setting window in initial state



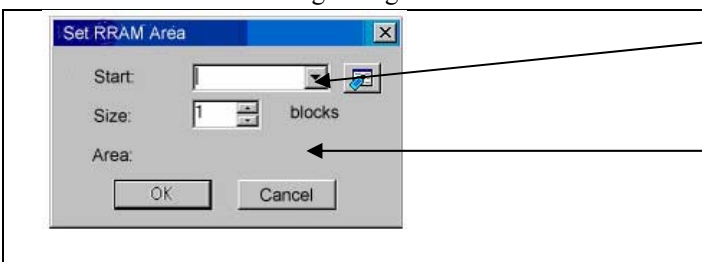
#### RAM Monitor Area Setting Window in initial state

By default, the monitor area is set to 000400h through 0007FFh. To change it, click the line you want to set.

#### Specifying the start address

You can set the start address of the RAM area to be monitored. To add a RAM monitor area, click the "Add..." button. The RAM Monitor Area Setting dialog box will be displayed.

##### 4. RAM monitor area setting dialog box



#### Specifying the start address

You can set the start address of the RAM area to be monitored.

#### Specifying the size

You can set the size to be monitored by specifying the number of blocks from the start address. One block is 256 bytes in size.

## 4. Hardware Specifications

This chapter describes specifications of this product.

### 4.1 Target MCU Specifications

Table 4.1 lists the specifications of target MCUs which can be debugged with this product.

Table 4.1 Specifications of target MCUs for the M306NKT-EPB

Item	Description
Applicable MCU	M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN
Evaluation MCU	M306NMFJGP (two pieces) ROM: 512KB + 4KB, RAM: 31KB
Applicable MCU mode	Single-chip mode Memory expansion mode Microprocessor mode
Max. ROM/RAM capacity	1. Internal flash ROM: 512KB + 4KB 0F000h--0FFFFh, 80000h--FFFFFFh 2. Internal RAM: 31KB 00400h--07FFFh
Operating frequency	24MHz (when using PLL)
Power supply voltage	3.0--5.5V

## 4.2 Access Timing

Because this product emulates some ports, access timings are different from those of the actual MCUs. Chapters 4.2.1 (Vcc=5V) and 4.2.2 (Vcc=3V) describe the access timing using this product.

### 4.2.1 Operation Timing of Memory Expansion Mode and Microprocessor Mode (Vcc=5V)

#### (1) Separate Bus Timing

Table 4.2 and Figure 4.1 show the bus timing in memory expansion mode and microprocessor mode (3-wait, accessing external area).

Table 4.2 Memory expansion mode and microprocessor mode (3-wait, accessing external area)

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
td(BCLK-AD)	Address output delay time		25		See left
th(BCLK-AD)	Address output hold time (BCLK standard)	4		See left	
th(RD-AD)	Address output hold time (RD standard)	0		-2	
th(WR-AD)	Address output hold time (WR standard)	(*2)		See left	
td(BCLK-CS)	Chip-select output delay time		25		See left
th(BCLK-CS)	Chip-select output hold time (BCLK standard)	4		See left	
td(BCLK-ALE)	ALE signal output delay time		25		See left
th(BCLK-ALE)	ALE signal output hold time	-4		See left	
td(BCLK-RD)	RD signal output delay time		25		See left
th(BCLK-RD)	RD signal output hold time	0		See left	
td(BCLK-WR)	WR signal output delay time		25		27
th(BCLK-WR)	WR signal output hold time	0		See left	
td(BCLK-DB)	Data output delay time (BCLK standard)		40		See left
th(BCLK-DB)	Data output hold time (BCLK standard)	4		See left	
td(DB-WR)	Data output delay time (WR standard)	(*1)		See left	
th(WR-DB)	Data output hold time (WR standard)	(*2)		See left	

\*1 Calculated by the following formula according to the frequency of BCLK.

$$\frac{(n - 0.5) \times 10^9}{f(BCLK)} - 40 \quad [\text{ns}] \quad n: "3" \text{ for 3 wait}$$

\*2 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 10 \quad [\text{ns}]$$



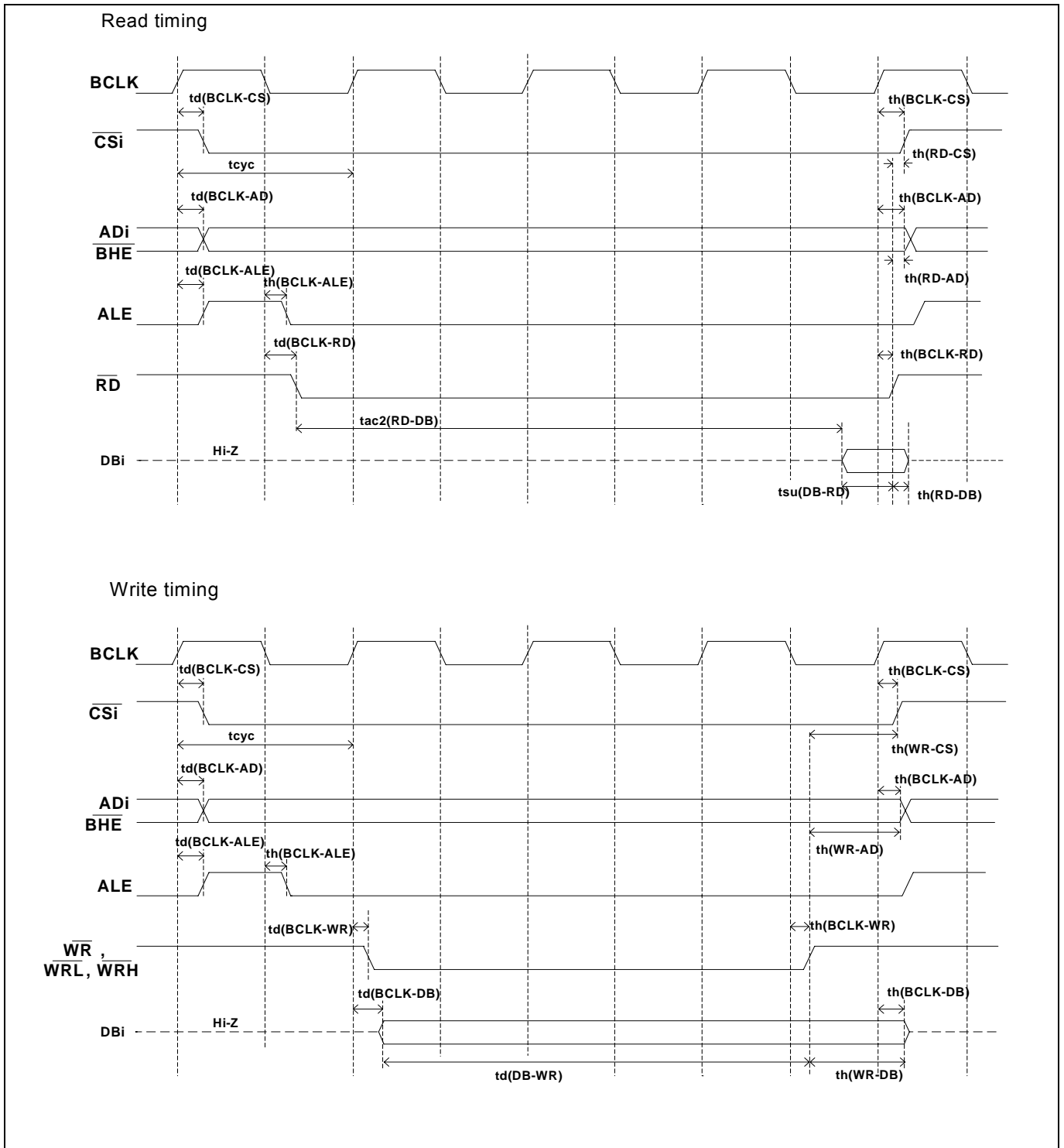


Figure 4.1 Memory expansion mode and microprocessor mode (3 wait, accessing external area)

## (2) Multiplex Bus Timing

Table 4.3 and Figure 4.2 show the bus timing in memory expansion mode and microprocessor mode (2-wait, accessing external area, using multiplex bus).

Table 4.3 Memory expansion mode and microprocessor mode (2-wait, accessing external area, using multiplex bus)

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
td(BCLK-AD)	Address output delay time		25		See left
th(BCLK-AD)	Address output hold time (BCLK standard)	4		See left	
th(RD-AD)	Address output hold time (RD standard)	(*1)		See left	
th(WR-AD)	Address output hold time (WR standard)	(*1)		See left	
td(BCLK-CS)	Chip-select output delay time		25		See left
th(BCLK-CS)	Chip-select output hold time (BCLK standard)	4		See left	
th(RD-CS)	Chip-select output hold time (RD standard)	(*1)		See left	
th(WR-CS)	Chip-select output hold time (WR standard)	(*1)		See left	
td(BCLK-RD)	RD signal output delay time		25		See left
th(BCLK-RD)	RD signal output hold time	0		See left	
td(BCLK-WR)	WR signal output delay time		25		See left
th(BCLK-WR)	WR signal output hold time	0		See left	
td(BCLK-DB)	Data output delay time (BCLK standard)		40		See left
th(BCLK-DB)	Data output hold time (BCLK standard)	4		See left	
td(DB-WR)	Data output delay time (WR standard)	(*2)		See left	
th(WR-DB)	Data output hold time (WR standard)	(*1)		See left	
td(BCLK-ALE)	ALE output delay time (BCLK standard)		25		See left
th(BCLK-ALE)	ALE output hold time (BCLK standard)	-4		See left	
td(AD-ALE)	ALE output delay time (Address standard)	(*3)		See left	
th(ALE-AD)	ALE output hold time (Address standard)	(*4)		See left	
td(AD-RD)	After address RD signal output delay time	0		See left	
td(AD-WR)	After address WR signal output delay time	0		See left	
tdz(RD-AD)	Address output floating start time		8		15

\*1 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 10 \quad [\text{ns}]$$

\*2 Calculated by the following formula according to the frequency of BCLK.

$$\frac{(n - 0.5) \times 10^9}{f(BCLK)} - 40 \quad [\text{ns}] \quad n: "2" \text{ for 2 wait}$$

\*3 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 25 \quad [\text{ns}]$$

\*4 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 15 \quad [\text{ns}]$$

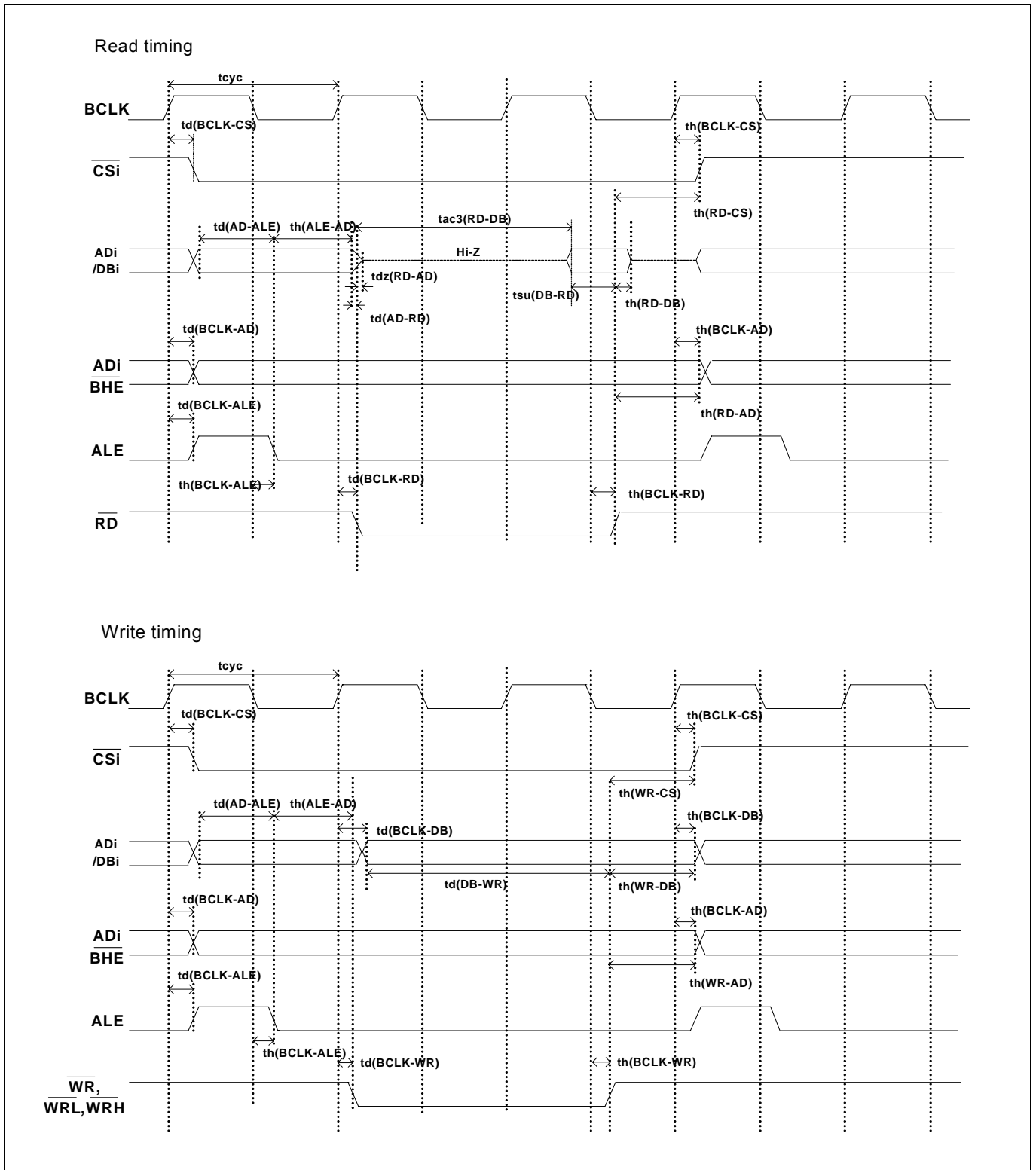


Figure 4.2 Memory expansion mode and microprocessor mode (2 wait, accessing external area, using multiplex bus)

(3) Timing Requirements

Table 4.4 and Figure 4.3 list the timing requirements in memory expansion mode and microprocessor mode.

Table 4.4 Timing requirements

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
tsu(DB-RD)	Data input setup time	40		55	
tsu(RDY-BCLK)	RDY# input setup time	30		45	
tsu(HOLD-BCLK)	HOLD# input setup time	40		55	
th(RD-DB)	Data input hold time	0		See left	
th(BCLK-RDY)	RDY# input hold time	0		See left	
th(BCLK-HOLD)	HOLD# input hold time	0		See left	
td(BCLK-HLDA)	HLDA# output delay time		40		See left

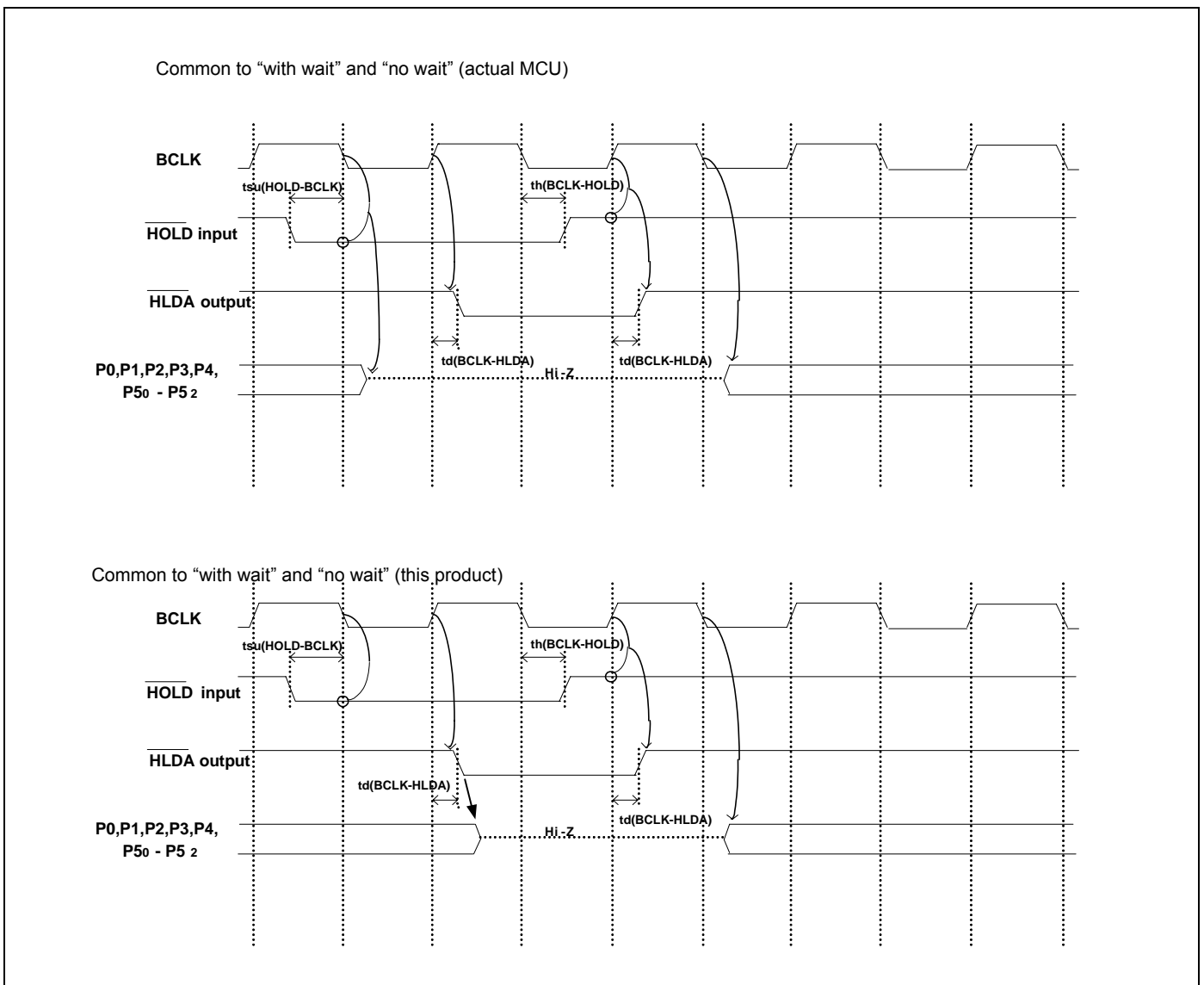


Figure 4.3 Timing requirements

\* Compared with an actual MCU, this product enters high-impedance state after a 0.5 cycle delay.

## 4.2.2 Operation Timing of Memory Expansion Mode and Microprocessor Mode (Vcc=3V)

## (1) Separate Bus Timing

Table 4.5 and Figure 4.4 show the bus timing in memory expansion mode and microprocessor mode (3 wait, accessing external area).

Table 4.5 Memory expansion mode and microprocessor mode (3 wait, accessing external area)

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
td(BCLK-AD)	Address output delay time		30		See left
th(BCLK-AD)	Address output hold time (BCLK standard)	4		See left	
th(RD-AD)	Address output hold time (RD standard)	0		-3	
th(WR-AD)	Address output hold time (WR standard)	(*2)		See left	
td(BCLK-CS)	Chip-select output delay time		30		See left
th(BCLK-CS)	Chip-select output hold time (BCLK standard)	4		See left	
td(BCLK-ALE)	ALE signal output delay time		30		See left
th(BCLK-ALE)	ALE signal output hold time	-4		See left	
td(BCLK-RD)	RD signal output delay time		30		See left
th(BCLK-RD)	RD signal output hold time	0		See left	
td(BCLK-WR)	WR signal output delay time		30		See left
th(BCLK-WR)	WR signal output hold time	0		See left	
td(BCLK-DB)	Data output delay time (BCLK standard)		40		See left
th(BCLK-DB)	Data output hold time (BCLK standard)	4		See left	
td(DB-WR)	Data output delay time (WR standard)	(*1)		See left	
th(WR-DB)	Data output hold time (WR standard)	(*2)		See left	

\*1 Calculated by the following formula according to the frequency of BCLK.

$$\frac{(n - 0.5) \times 10^9}{f(BCLK)} - 40 \quad [\text{ns}] \quad n: "3" \text{ for 3 wait}$$

\*2 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 10 \quad [\text{ns}]$$

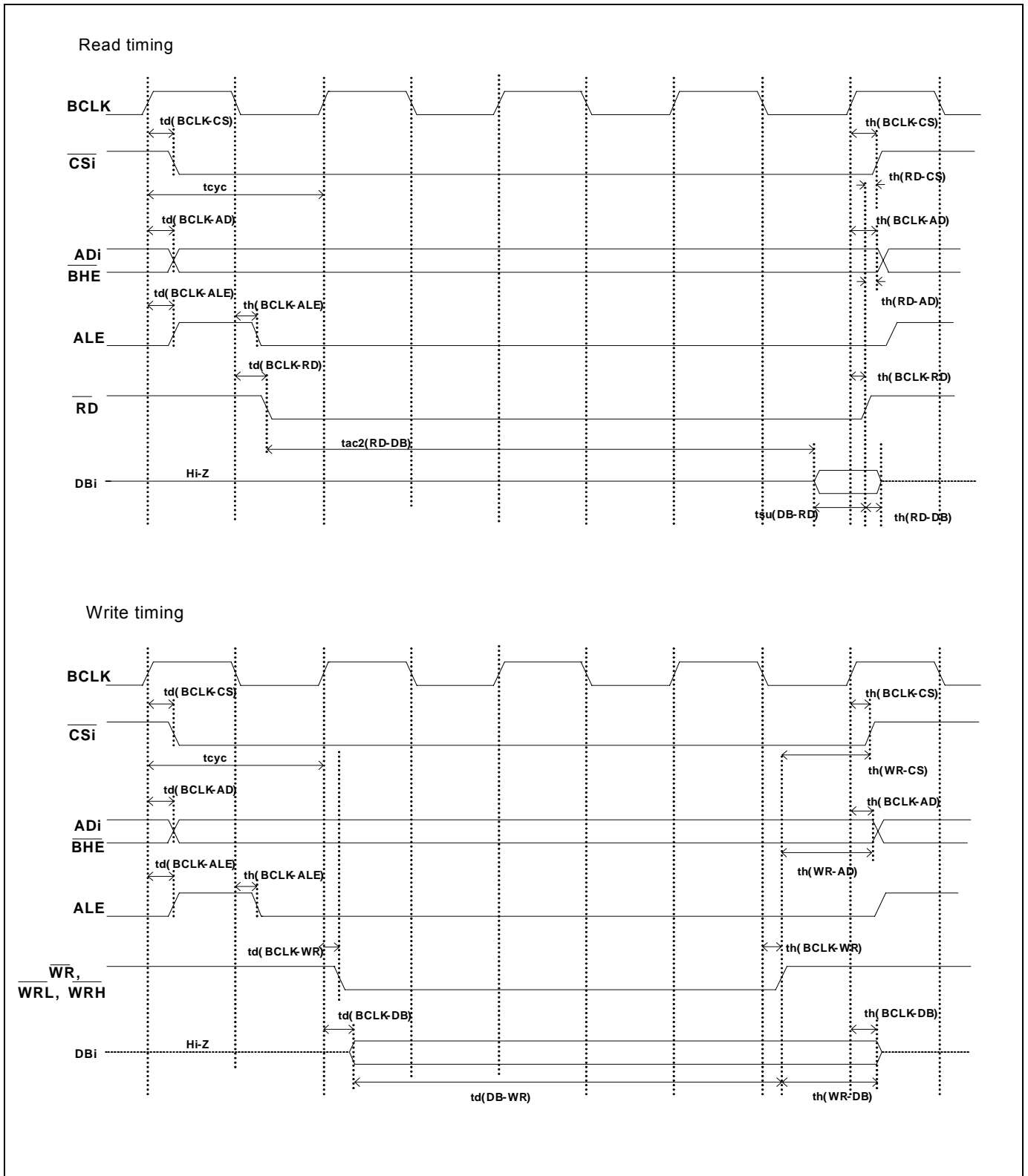


Figure 4.4 Memory expansion mode and microprocessor mode (3 wait, accessing external area)

## (2) Multiplex Bus Timing

Table 4.6 and Figure 4.5 show the bus timing in memory expansion mode and microprocessor mode (2 wait, accessing external area and using multiplex bus).

Table 4.6 Memory expansion mode and microprocessor mode (2 wait, accessing external area and using multiplex bus)

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
td(BCLK-AD)	Address output delay time		50		See left
th(BCLK-AD)	Address output hold time (BCLK standard)	4		See left	
th(RD-AD)	Address output hold time (RD standard)	(*1)		See left	
th(WR-AD)	Address output hold time (WR standard)	(*1)		See left	
td(BCLK-CS)	Chip-select output delay time		50		See left
th(BCLK-CS)	Chip-select output hold time (BCLK standard)	4		See left	
th(RD-CS)	Chip-select output hold time (RD standard)	(*1)		See left	
th(WR-CS)	Chip-select output hold time (WR standard)	(*1)		See left	
td(BCLK-RD)	RD signal output delay time		40		See left
th(BCLK-RD)	RD signal output hold time	0		See left	
td(BCLK-WR)	WR signal output delay time		40		See left
th(BCLK-WR)	WR signal output hold time	0		See left	
td(BCLK-DB)	Data output delay time (BCLK standard)		50		See left
th(BCLK-DB)	Data output hold time (BCLK standard)	4		See left	
td(DB-WR)	Data output delay time (WR standard)	(*2)		See left	
th(WR-DB)	Data output hold time (WR standard)	(*1)		See left	
td(BCLK-ALE)	ALE output delay time (BCLK standard)		40		See left
th(BCLK-ALE)	ALE output hold time (BCLK standard)	-4		See left	
td(AD-ALE)	ALE output delay time (Address standard)	(*3)		See left	
th(ALE-AD)	ALE output hold time (Address standard)	(*4)		See left	
td(AD-RD)	After address RD signal output delay time	0		See left	
td(AD-WR)	After address WR signal output delay time	0		See left	
tdz(RD-AD)	Address output floating start time		8		15

\*1 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 10 \quad [\text{ns}]$$

\*2 Calculated by the following formula according to the frequency of BCLK.

$$\frac{(n - 0.5) \times 10^9}{f(BCLK)} - 50 \quad [\text{ns}] \quad n: "2" \text{ for 2 wait}$$

\*3 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 40 \quad [\text{ns}]$$

\*4 Calculated by the following formula according to the frequency of BCLK.

$$\frac{0.5 \times 10^9}{f(BCLK)} - 15 \quad [\text{ns}]$$

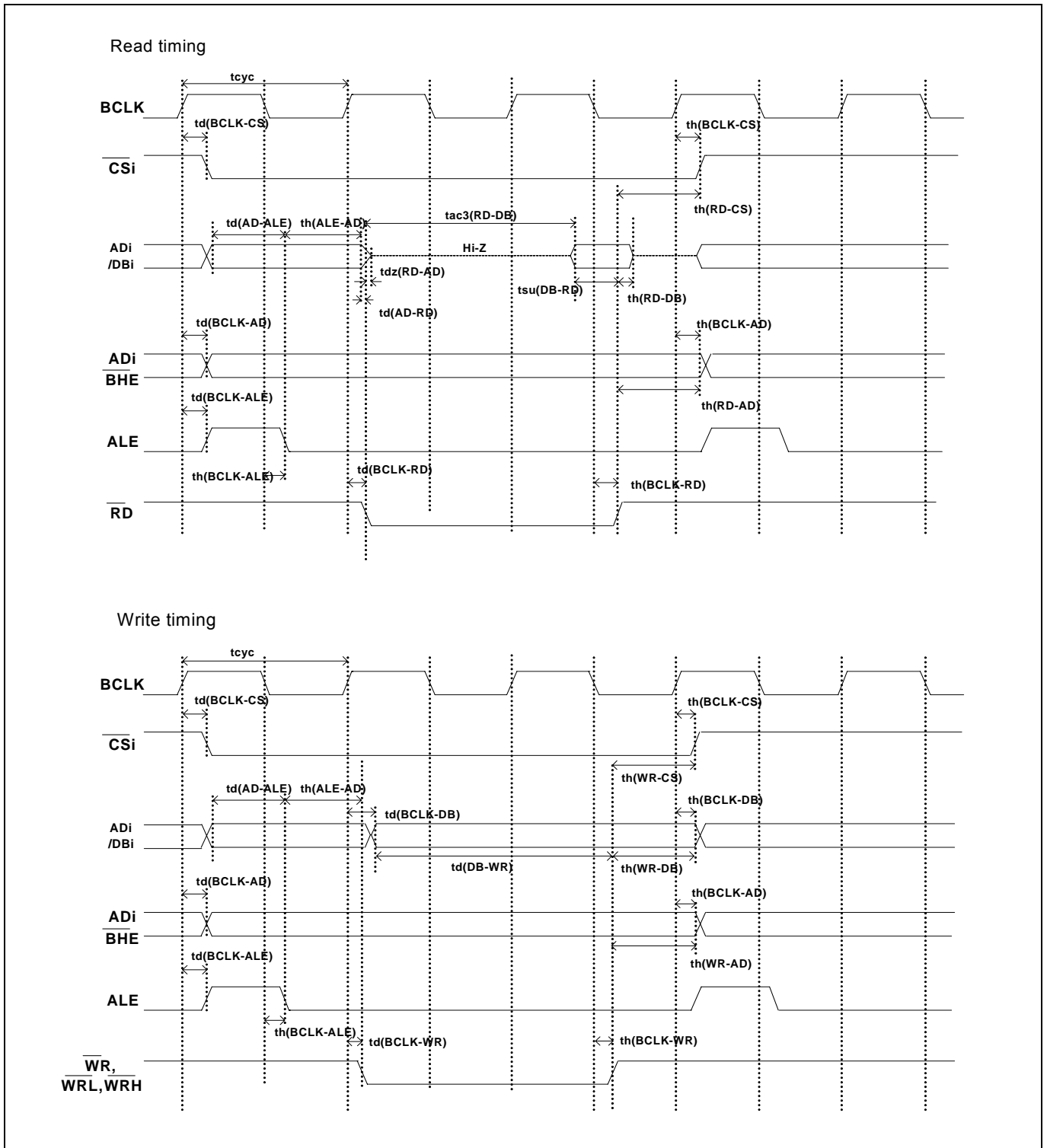


Figure 4.5 Memory expansion mode and microprocessor mode (2 wait, accessing external area and using multiplex bus)



(3) Timing Requirements

Table 4.7 and Figure 4.6 show timing requirements in memory expansion mode and microprocessor mode.

Table 4.7 Timing requirements

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
tsu(DB-RD)	Data input setup time	50		65	
tsu(RDY-BCLK)	RDY# input setup time	40		55	
tsu(HOLD-BCLK)	HOLD# input setup time	50		65	
th(RD-DB)	Data input hold time	0		See left	
th(BCLK-RDY)	RDY# input hold time	0		See left	
th(BCLK-HOLD)	HOLD# input hold time	0		See left	
td(BCLK-HLDA)	HLDA# output delay time		40		See left

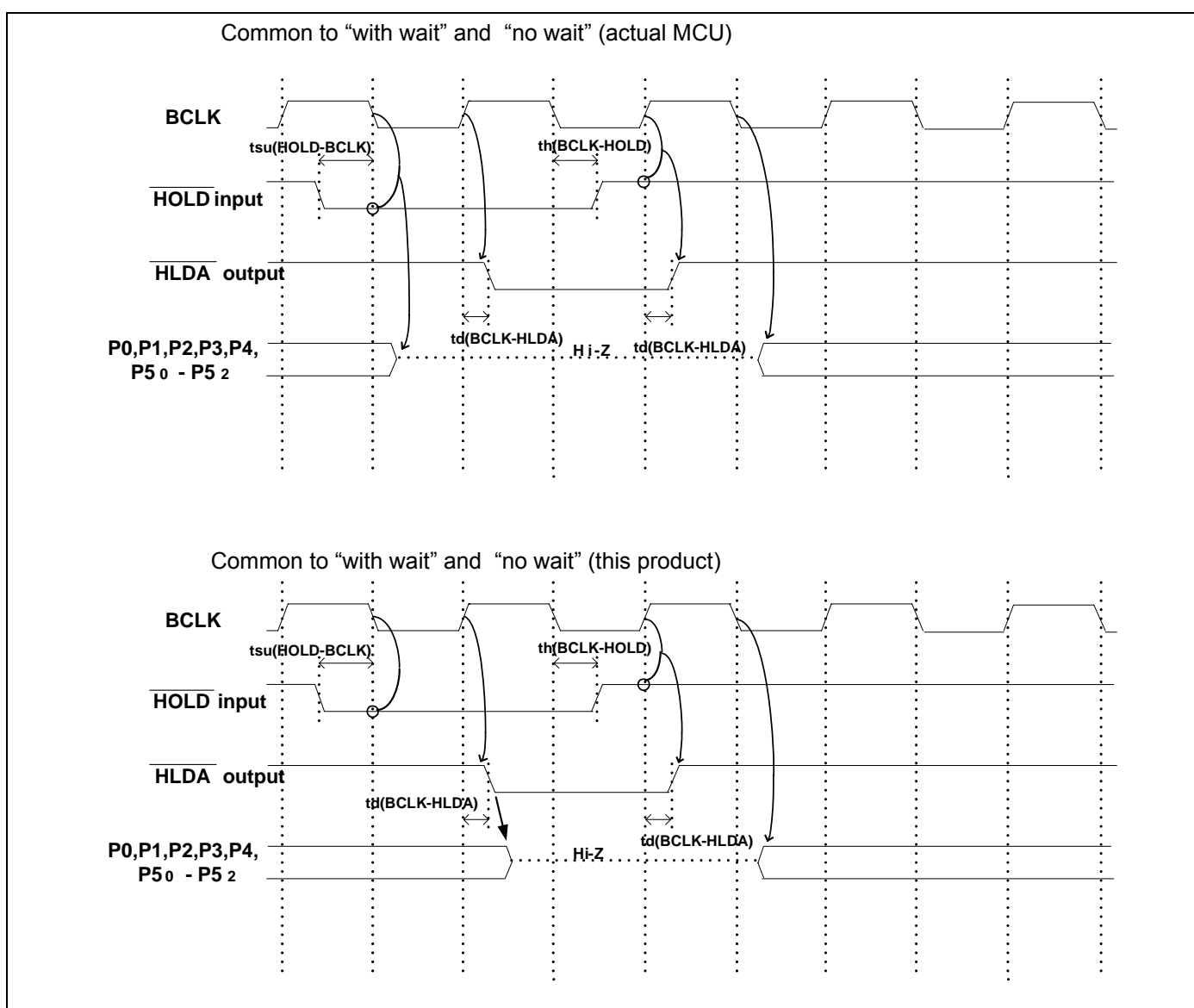


Figure 4.6 Timing requirements

\* Compared with an actual MCU, this product enters high-impedance state after a 0.5 cycle delay.

### 4.3 Differences between the Actual MCU and Emulator

Differences between the actual MCU and emulator are shown below. When debugging the MCU using this product, be careful about the following precautions.

## IMPORTANT

#### Note on Differences between the Actual MCU and Emulator:

- Operations of the emulator system differ from those of actual MCUs as listed below.
  - (1) Reset condition  
Set the time for starting up (0.2V<sub>CC</sub> to 0.8V<sub>CC</sub>) 1 μs or less.
  - (2) Initial values of internal resource data of an MCU at power-on
  - (3) Interrupt stack pointer (ISP) after a reset is released
  - (4) Capacities of the internal memories (ROM and RAM)  
The evaluation MCU of this product has RAM of 31KB (00400h--07FFFh) and flash ROM of 4KB (0F000h--0FFFFh) and 512KB (80000h--FFFFFFh).
  - (5) Oscillator circuit  
In the oscillator circuit where an oscillator is connected between pins X<sub>IN</sub> and X<sub>OUT</sub>, oscillation does not occur because a converter board is used between the evaluation MCU and the user system. It is same for pins X<sub>CIN</sub> and X<sub>COUT</sub>. For notes on when using the oscillator circuit on the user system, refer to "2.11.3 Using the Oscillator Circuit on the User System" (page 46).
  - (6) A/D conversion  
The characteristics of the A/D converter differ from those of actual MCU because there are a package converter board and other devices between the evaluation MCU and the user system.
  - (7) Ports P0 to P5, P10  
This product emulates some I/O ports (P0 to P5, P10). Therefore, the electrical characteristics of these ports differ from those of an actual MCU.
  - (8) Address and status of BHE#  
When the internal RAM or ROM area of an MCU is accessed during user program execution, actual MCUs retain a preceding address and status of BHE#, while this product does not.
  - (9) Status of a data bus  
In stop or wait mode, actual MCUs retain a preceding status of a data bus, while with this product a data bus is floating.
  - (10) Operation after releasing the reset  
After releasing the reset, for about max. 380 cycles, a program to control the emulator will be executed. Note that this will be included in the user program execution time and tracing result.

#### Note on RESET# Input:

- A low input to pin RESET# from the user system is accepted during the user program execution (only while the RUN status LED on the PC7501's upper panel is lit).

#### Note on RDY# Input:

- Be sure to input "L" to pin RDY# of the user system during the user program execution (when the RUN status LED on the PC7501's upper panel is lit). Inputting "L" to pin RDY# during the user program is being stopped may cause a malfunction of the emulator.

#### Note on HOLD# Input:

- Be sure to input "L" to pin HOLD# of the user system during the user program execution (when the RUN status LED on the PC7501's upper panel is lit). Inputting "L" to pin HOLD# during the user program is being stopped may cause a malfunction of the emulator.

#### Note on NMI# Input:

- A low input to pin NMI# from the user system is accepted during the user program execution (only while the RUN status LED on the emulator's upper panel is lit).

## IMPORTANT

### Note on Voltage Detect Circuit:

- With this product, as the power voltage cannot be change after powering on the user system, the voltage detect circuit (voltage down detect interrupt, voltage down detect reset, etc.) cannot be used.

### Notes on Reset Vector Area:

- Memory in the PC7501 is always selected as a reset vector area (FFFFCh--FFFFFh) in order to operate the evaluation MCU in emulator-dedicated mode. Set the contents of the reset vector area in one of the following ways.
  - (1) Download a user program to an area including the reset vector area.
  - (2) Set the reset vector using a memory window of the emulator debugger etc.
- A reset vector area can be changed only when a program is stopped.
- Do not access the rest vector area as data. It may not be accessed properly, and the program may not run properly in the next bus cycle.

### Notes on Stack Area:

- With this product, a maximum 8 bytes of the user stack is consumed as a work area. Therefore, ensure the +8 byte maximum capacity used by the user program as the user stack area.  
If the user stack does not have enough area, do not use areas which cannot be used as stack (SFR area, RAM area which stores data, or ROM area) as work area. Using areas like this is a cause of user program crashes and destabilized emulator control.
- With this product, the interrupt stack pointer (ISP) is set to 00500h and used as stack area after the reset is released.

### Notes on Maskable Interrupts:

- Even if a user program is not being executed (including when run-time debugging is being performed), the evaluation MCU keeps running so as to control the emulation probe. Therefore, timers and other components do not stop running. If a maskable interrupt is requested when the user program is not being executed (including when run-time debugging is being performed), the maskable interrupt request cannot be accepted, because the emulator disables interrupts. The interrupt request is accepted immediately after the user program execution is started.
- Take note that when the user program is not being executed (including when run-time debugging is being performed), a peripheral I/O interruption is not accepted.

### Note on DMA Transfer:

- With this product, the user program is stopped with a loop program to a specific address. Therefore, if a DMA request is generated by a timer or other source while the user program is stopped, DMA transfer is executed. However, make note of the fact that DMA transfer while the program is stopped may not be performed correctly. Also note that the below registers have been changed to generate DMA transfer as explained here even when the user program is stopped.
  - (1) DMA0 transfer count register: TCR0
  - (2) DMA1 transfer count register: TCR1

## IMPORTANT

### Notes on Pullup Control:

- Because this product emulates some I/O ports (P0 to P5, P10), you can not pullup ports P0 to P5 by the pullup control registers, Install the included resistor arrays (51kΩ) as necessary.
- The initial value of pullup control register 1 PUR1 of this product is different from that of an actual MCU. When a Vcc level is input to pin CNVss, the value of an actual MCU is “02h” (bit 1 PU11 is “1”), while that of this product is “00h”.

### Note on Setting "1" to Protect Bit 2 (PCR2) with Sub Clock:

- When the CPU clock is set to the sub-clock (low-speed mode or low power dissipation mode), even if you enable the PRC2 bit, writing to the register protected by the PRC2 bit (PD7, PD9, S3C, S4C, S5C and S6C) cannot be done properly. When you enable the PRC2 bit and write to the register protected by the PRC2, do not set the CPU clock to the sub-clock.

### Note on Setting "1" to Protect Bit 2 (PRC2) in Division by 2 Mode:

- Under the following conditions, even if you enable the PRC2 bit, writing to the register protected by the PRC2 bit (PD7, PD9, S3C, S4C, S5C and S6C) cannot be done properly.
  - (1) Between when stop mode is released and when a hardware reset is executed
  - (2) Between when low power dissipation mode is enabled and when a hardware reset is executed
 Therefore, if the condition (1) or (2) applies, when you enable the PRC2 bit and write to the register protected by the PRC2 (PD7, PD9, S3C, S4C, S5C and S6C), do not set the CPU clock to the main clock in division by 2 mode.

### Note on the Input Thresholds for the Pins P1\_5/D13/INT3, P1\_6/D14/INT4 and P1\_7/D15/INT5

- With this product, regarding pins P1\_5/D13/INT3, P1\_6/D14/INT4 and P1\_7/D15/INT5, a device which port or data bus inputs to and a device which INT interrupt inputs to are different as follows:
  - Device which port or data bus inputs to: Port emulation FPGA (input level: TTL)
  - Device which INT interrupt inputs to: Evaluation MCU for emulating peripheral functions (input level: CMOS Schmidt)
 Therefore, the port input level can be read as “H” immediately after an INT interrupt (falling), and the port input level can be read as “H” immediately before an INT interrupt (rising).

### Note on Input Level for K10# to K13#:

- With this product, the K10# to K13# are TTL input level using the port emulation FPGA although these are the CMOS Schmidt input level with the actual MCU. Therefore, the wrong interrupt may occur near the threshold of the TTL level (2.0V to 0.8V) when the input signal is slow changing.

### Note on Final Evaluation:

- Be sure to evaluate your system with an evaluation MCU. Before starting mask production, evaluate your system and make final confirmation with a CS (Commercial Sample) version MCU.

### 4.4 Connection Diagrams

Figure 4.7 shows a connection diagram of the M306NKT-EPB. This connection diagram mainly show the circuit to be connected to the user system. The circuits not connected to the user system such as the emulator's control system are omitted. Tables 4.8 and 4.9 show IC electric characteristics of this product for reference purpose.

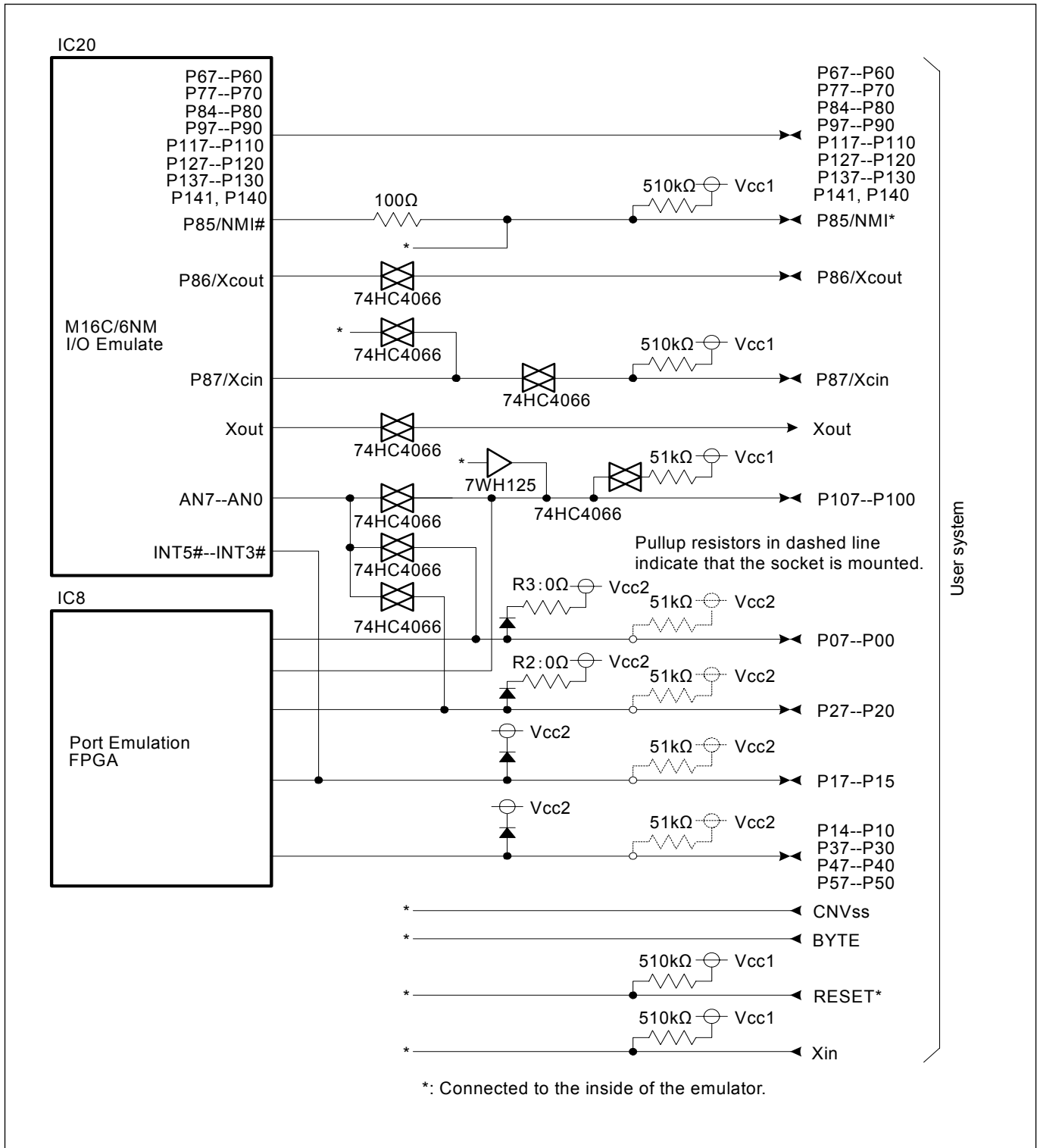


Figure 4.7 Connection diagram

Table 4.8 Electrical characteristics of the 74HC4066

Symbol	Item	Condition	Standard values			Unit
			Min.	Standard	Max.	
$R_{ON}$	ON resistor	$V_{CC}=4.5V$	-	96	170	$\Omega$
$\Delta R_{ON}$	ON resistor difference	$V_{CC}=4.5V$	-	10	-	
$I_{OFF}$	Leak current (Off)	$V_{CC}=12.0V$	-	-	$\pm 100$	nA
$I_{IZ}$	Leak current (On, output: open)	$V_{CC}=12.0V$	-	-	$\pm 100$	

Table 4.9 Electrical characteristics of port emulation FPGA

Symbol	Item	Condition	Standard values			Unit	
			Min.	Standard	Max.		
$V_{IH}$	"H" level input voltage		2.0	-	5.5	V	
$V_{IL}$	"L" level input voltage		-0.5	-	0.8		
$V_{OH}$	"H" level output voltage	$I_{OH}=-8mA$ DC, $V_{CC}=4.75V$	2.4	-	-		
		$I_{OH}=-8mA$ DC, $V_{CC}=3.00V$	2.4	-	-		
$V_{OL}$	"L" level output voltage	$I_{OL}=8mA$ DC, $V_{CC}=4.75V$	-	-	0.45		
		$I_{OL}=8mA$ DC, $V_{CC}=3.00V$	-	-	0.45		
$I_I$	Leak current	$V_I=V_{CC}$ or GND	-10	-	10		$\mu A$
$I_{OZ}$	Leak current (Tristate)	$V_O=V_{CC}$ or GND	-40	-	40		$\mu A$
$C_{IN}$	I/O pin input capacitance	$V_{IN}=0V$ , $f=1.0MHz$	-	-	8	pF	

### 4.5 External Dimensions

#### 4.5.1 External Dimensions of the Emulation Probe

Figure 4.8 shows external dimensions of the M306NKT-EPB.

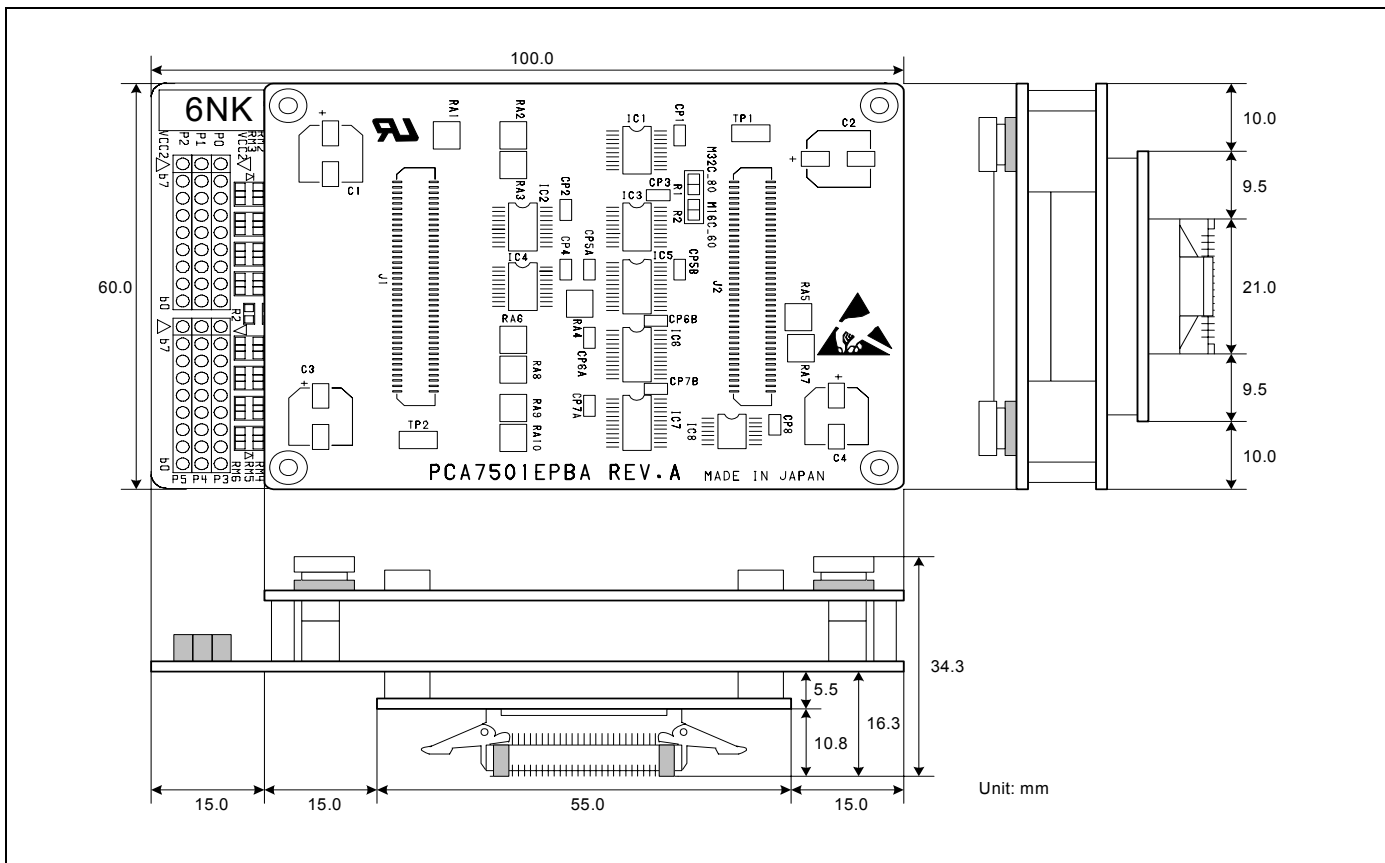


Figure 4.8 External dimensions of the emulation probe

4.5.2 External Dimensions of the M30800T-PTC

Figure 4.9 shows external dimensions of the converter board M30800T-PTC for a 100-pin LCC (included with the M306NKT-EPB).

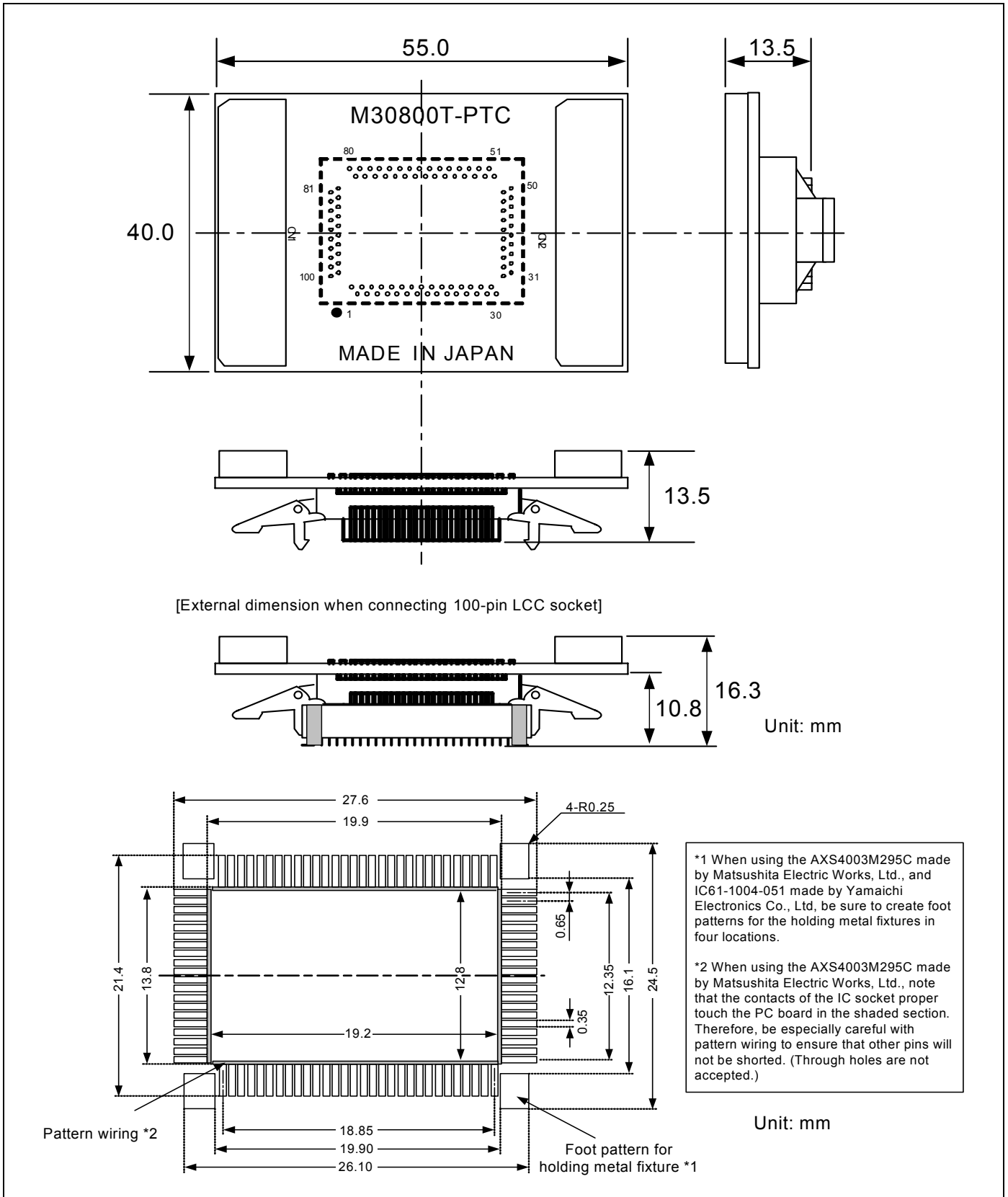


Figure 4.9 External dimensions of the M30800T-PTC



4.5.3 External Dimensions of the Converter Board M3T-F160-100NRB

Figure 4.10 shows external dimensions and a sample foot pattern of the converter board for a 100-pin 0.65mm pitch QFP M3T-F160-100NRB (not included).

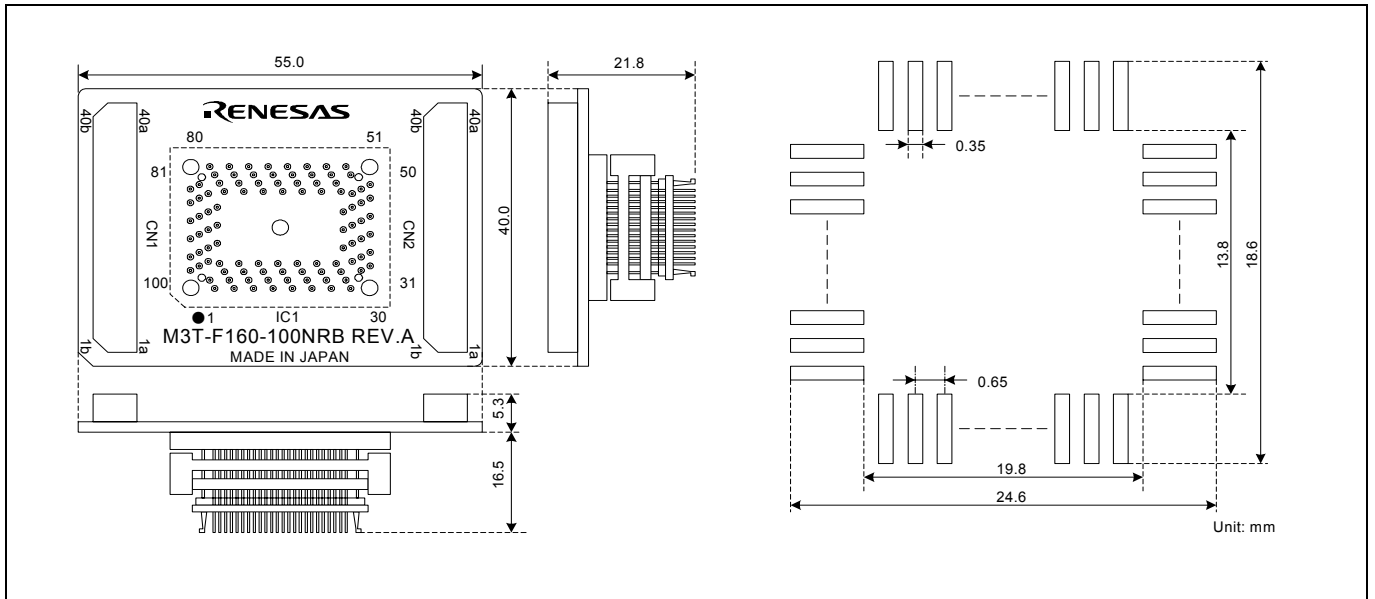


Figure 4.10 External dimensions and a sample foot pattern of the M3T-F160-100NRB

4.5.4 External Dimensions of the Converter Board M3T-F160-100NSD

Figure 4.11 shows external dimensions and a sample foot pattern of the converter board for a 100-pin 0.5mm pitch QFP M3T-F160-100NSD (not included).

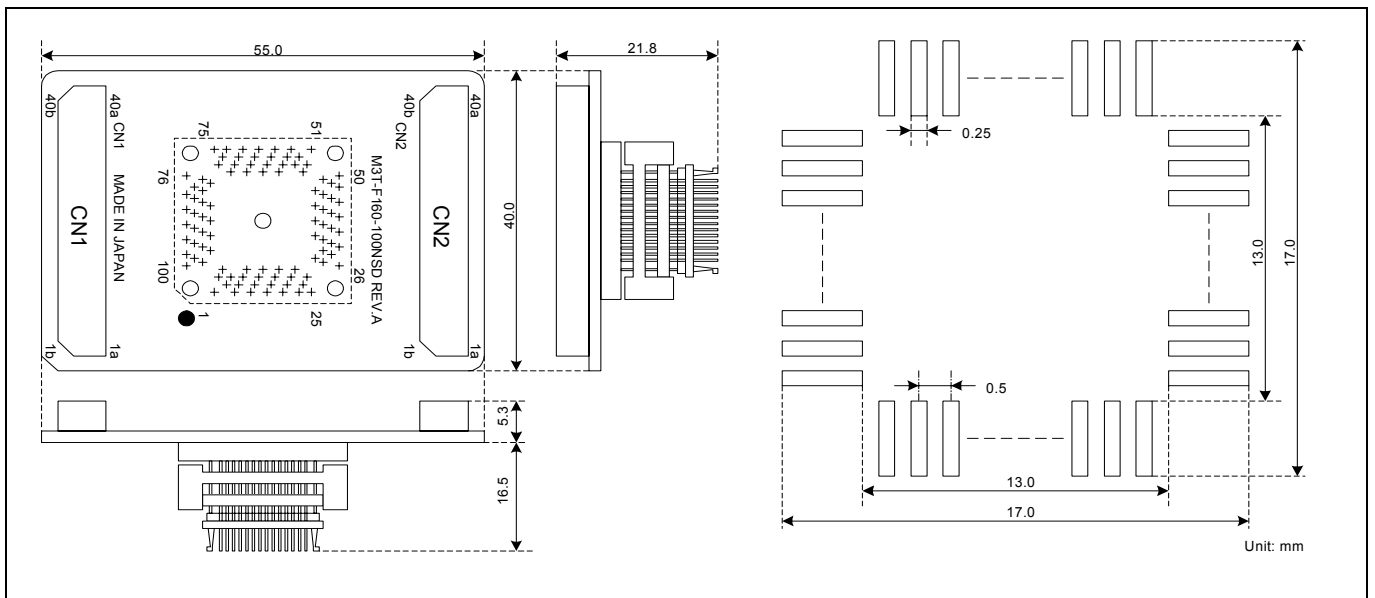


Figure 4.11 External dimensions and a sample foot pattern of the M3T-F160-100NSD

4.5.5 External Dimensions of the Converter Board M3T-F160-128NRD

Figure 4.12 shows external dimensions and a sample foot pattern of the converter board for a 128-pin 0.5mm pitch LQFP M3T-F160-128NRD (not included).

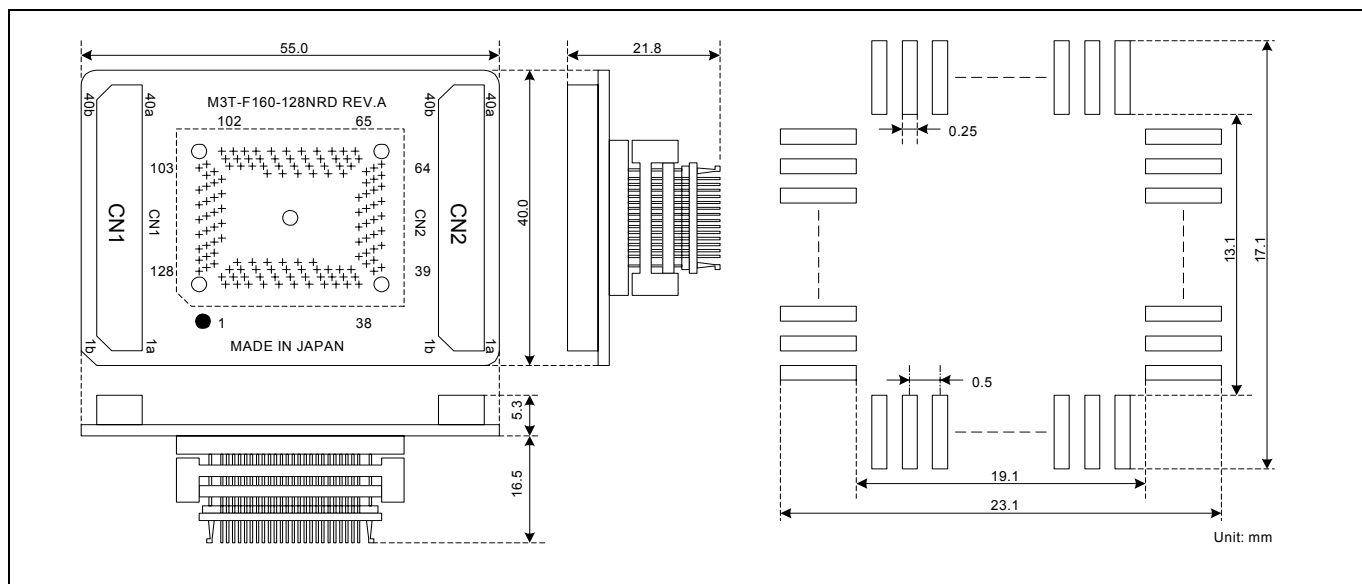


Figure 4.12 External dimensions and a sample foot pattern of the M3T-F160-128NRD

## 4.6 Notes on Using This Product

Notes on using this product are listed below. When debugging the MCU using the emulator, be careful about the following precautions.

### IMPORTANT

#### Note on the Version of the Emulator Debugger:

- Be sure to use this product with one of the following emulator debuggers.
  1. M16C R8C PC7501 Emulator debugger V.1.01 Release 00 or later
  2. M3T-PD30F V.2.20 Release 1 or later

#### Note on Selecting the MCU File:

- Select the MCU file "M16C6NK.mcu" at the startup of the emulator debugger.

#### Note on Malfunctions in the PC7501 System:

- If the emulator malfunctions because of interference like external noise, do the following to remedy the trouble
  - (1) Press the RESET switch on the emulator front panel.
  - (2) If normal operation is not restored after step (1), shut OFF power to the emulator once and then reactivate it.

#### Notes on Downloading Firmware:

- Before using this product for the first time, it is necessary to download the dedicated firmware (emulator's control software installed in the flash memory in the PC7501). Please note that, to do this, it is necessary to start up the emulator main unit in maintenance mode. For downloading firmware, see "2.7 Downloading Firmware" (page 28). Once the firmware has been downloaded, the product can be used by simply turning on the power.
- Do not shut off the power while downloading the firmware. If this happens, the product will not start up properly. If the power is shut off unexpectedly, redownload the firmware.
- Download the firmware with the user system not connected.

#### Notes on Self-check:

- If the self-check does not result normally (excluding user system errors), the emulation probe may be damaged. Then contact your local distributor.
- Run the self-check with the user system not connected.

#### Note on Quitting the Emulator Debugger:

- To restart the emulator debugger, always shut power to the PC7501 off once and then on again.

## IMPORTANT

### Note on Display of MCU Status:

- "MCU status" you can refer to in the MCU tab of the MCU Setting dialog box of the emulator debugger shows pin levels of the user system. Make sure that proper pin levels are specified according to the mode you use.
  - (1) When single-chip mode is used:  
CNVss: Low
  - (2) When memory expansion mode 16-bit data bus width is used:  
CNVss: Low, BYTE: Low, HOLD#: High, RDY#: High
  - (3) When memory expansion mode 8-bit data bus width is used:  
CNVss: Low, BYTE: High, HOLD#: High, RDY#: High
  - (4) When microprocessor mode 16-bit data bus width is used:  
CNVss: High, BYTE: Low, HOLD#: High, RDY#: High
  - (5) When microprocessor mode 8-bit data bus width is used:  
CNVss: High, BYTE: High, HOLD#: High, RDY#: High

### Note on Clock Supply to the MCU:

- A clock supplied to the evaluation MCU is selected by the Emulator tab in the Init dialog box of the emulator debugger.
  - (1) When "Internal" is selected:  
A clock generated by the oscillator circuit board in the PC7501 is supplied. It is continually supplied regardless of the status of the user system clock and that of the user program execution.
  - (2) When "External" is selected:  
A clock generated by the oscillator in the user system is supplied. It depends on the status of the oscillation (on/off) of the user system.
  - (3) When "Generated" is selected:  
A clock generated by the dedicated circuit in the PC7501 is supplied to the evaluation MCU. It is continually supplied regardless of the status of the user system clock and that of the user program execution.

### Note on Operating Frequency:

- You can not use this product with a main clock ( $X_{IN}$ - $X_{OUT}$ ) less than 1MHz. To use it with a main clock less than 1MHz, contact your local distributor.

### Note on Timeout Setting:

- If you download, execute or single-step a program at a low operating frequency, a communication timeout error may occur.  
Execute the command shown below in the Script Window after starting the emulator debugger.  
[Command]  
    \_settimeout 300, 300  
Once this command is executed, it remains effective the next time you start the emulator debugger. Therefore, you do not need to execute this command again.  
If the error still occurs in this setting, contact your local distributor.

### Note on Stop and Wait Modes:

- Do not single step an instruction shifting to stop or wait mode. It may cause communication errors.

### Note on Returning from Stop Mode with the On-chip Oscillator Clock:

- When this product shifts from on-chip oscillator mode or on-chip oscillator low power dissipation mode to stop mode, it cannot be returned with an interrupt other than the NMI#.
 

Interrupts with which this product cannot return:

  - Key input interrupt
  - INT interrupt
  - Interrupts of timers A and B
  - Serial I/O interrupt

## IMPORTANT

### Note on the Watchdog Function:

- If the reset circuit of the user system has a watchdog timer, disable it when using the emulator.

### Note on Protect Register:

- The protect is not canceled when bit 2 of protect register PRCR (PRC2), which enables writing into the ports P7 and P9 direction registers and the SI/Oi control register, is changed with the below procedure.
  - (1) Step execution of an instruction setting PRC2 to "1"
  - (2) Execution from the instruction setting "1" to PRC2 where a software breakpoint is set
  - (3) Setting a break point between an instruction setting PRC2 to "1" and a point where the ports P7 or P9 direction registers or the SI/Oi control register is set
  - (4) Setting PRC2 to "1" by the memory window or script window

### Note on Access Prohibited Area:

- You cannot use internally reserved areas. Write signals to the areas will be ignored, and values read will be undefined.

### Note on Breaks:

- The area displaying break points in the program window of the emulator debugger shows the following breaks.
  - (1) Address-match break  
This is a debugging function which breaks a program, using the address-match interrupt function of the MCU, immediately before the system executes an instruction at a specified address. The instruction at the preset address will not be executed.
  - (2) Software break  
This is a debugging function which generates a BRK interruption by changing an instruction at a specified address to a BRK instruction (00h) to break a program immediately before the system executes an instruction at a specified address. The instruction at the preset address will not be executed.
  - (3) Hardware break  
This is a debugging function which breaks a program by setting the detection of an execution of an instruction at a specified address as a break event. The program will break after the instruction at the specified address is executed.

### Notes on Software Breaks:

- Software breaks change the instruction at a specified address to a BRK (00h). Therefore, take note that when you reference the result of a trace in bus mode, "00h" is displayed.
- The BRK instruction can be used for the emulator only. You cannot use it in a user program.
- You can neither set nor cancel a software breakpoint in the internal ROM area of an MCU during user program execution, while you can set or cancel it in the internal RAM area of an MCU or an area where chip select CS3# to CS0# are set to INTERNAL (emulation memory).

### Notes on Address-Match Breaks:

- As the processing speed of setting and canceling address-match breaks is relatively fast, you can save the times of writing into the internal flash ROM of an MCU.
- Address-match breaks can be set only in RAM and ROM areas of an MCU.
- Address-match breaks can be set at up to eight points.
- Address-match breaks can be set or canceled even when the user program is being executed.

## IMPORTANT

### Note on Power Supply to the User System:

- Pins Vcc1 and Vcc2 are connected to the user system to observe the voltage. Therefore, the power is not supplied to the user system from the emulator, design your system so that the user system is powered separately.
  - (1) The voltage of the user system should be within the operating range of the evaluation MCU.
  - (2) Do not change the voltage of the user system after turning on the power.

### Notes on Internal Flash ROM of the MCU:

- In single-chip mode and memory expansion modes, this product downloads a user program to the flash ROM of an MCU. By checking "Debug Option" in the MCU tab of the MCU setting dialog box of the emulator debugger, you can disable the operation of the internal flash ROM of the MCU to use the emulation memory of the PC7501. The maximum operating frequencies are as follows.
  - (1) "Debug Option " unchecked
    - Access area: Flash ROM of MCU
    - Max. operating frequency: 24MHz (0 wait, 1 wait)
  - (2) "Debug Option " checked
    - Access area: Emulation memory of the PC7501
    - Max. operating frequency: 10MHz (0 wait, 1 wait)
- Because the number of write/erase cycles of the internal flash ROM of the MCU is limited, it must be replaced at the end of its service-life.
- If the following errors occur frequently when downloading a program, replace the MCU board.
  - (1) Flash ROM erase error occurred ERROR (16258)
  - (2) Flash ROM verify error occurred ERROR (16259)

### Note on Debugging in Memory Expansion Mode and Microprocessor Mode:

- During debugging in memory expansion mode or microprocessor mode with this product, the emulation memory can be allocated. The emulation memory can be specified up to 4 areas by 4MB in the MAP tab of the MCU setting dialog box of the emulator debugger. Each maximum operating frequency is as follows.
 

- Maximum operating frequency (at 5.0V) 0 wait:	7MHz
1 wait:	20MHz
2 wait, 3 wait:	24MHz
- Maximum operating frequency (at 3.0V) 0 wait:	6MHz
1 wait:	17MHz
2 wait, 3 wait:	24MHz

## IMPORTANT

### Notes on Debugging in CPU Rewrite Mode:

- When you debug an M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK, M16C/6NL, M16C/6NM, M16C/6NN MCUs in CPU rewrite mode, do not change the block 0 area (FF000h--FFFFFFh) of the flash memory. Otherwise, the emulator will be uncontrollable.
- If you check "Debug the program using CPU Rewrite Mode" in the MCU tab of the Init dialog box of the emulator debugger, you cannot use the following functions.
  - (1) Setting software breakpoints in an internal ROM area
  - (2) Executing COME in an internal ROM area
- In CPU rewrite mode, do not stop the program. And do not single step an instruction shifting to CPU rewrite mode. The emulator will be uncontrollable in CPU rewrite mode.
- To reference data after executing CPU rewrite, stop the program at other than a rewrite control program area and use the memory window etc.

As the following interrupt vectors are used by the emulator system, the read data is different from expected value.

  - (1) BRK instruction (FFFE4h--FFFE7h)
  - (2) Address match (FFFE8h--FFFEbH)
  - (3) Single-step (FFFECh--FFFEFh)
  - (4) DBC (FFFF4h--FFFF7h)

### Notes on Address-Match Interrupts:

- When you use the address-match interrupt function in a user program, uncheck "Enable the Address Match Interrupt Break Function" in the MCU tab of the Init dialog box of the emulator debugger. Thus, normal software breaks are used for the internal RAM and ROM areas of an MCU.
- Do not set a software break at an address where an address-match interrupt occurs. Otherwise, a user program may be run out of control. Set a software or hardware break in the top address in address-match interrupt processing.
- When you single step an address where an address-match interrupt occurs, the program stops after executing the first instruction after returning from address-match interrupt processing.

### Note on Accessing Addresses 00000h and 00001h:

- With the M16C/60 Series MCUs, when a maskable interrupt is generated, the interrupt data (interrupt number and interrupt request level) stored in addresses 00000h and 00001h are read out. Also, the interrupt request bit is cleared when address 00000h or 00001h is read out. Consequently, when the address 00000h or 00001h readout instruction is executed or when address 00000h or 00001h is read out in the cause of a program runaway, a malfunction occurs in that the interrupt is not executed despite the interrupt request, because the request bit of the highest priority interrupt factor enabled is cleared.

For this malfunction, when the reading out to address 00000h or 00001h is generated excluding the interrupt, the WARNING LED (yellow) on the upper panel of the PC7501 lights up to alarm. When this LED lights, there is a possibility of wrong access, therefore check the user program.

## 5. Troubleshooting

This chapter describes how to troubleshoot when this product does not work properly.

### 5.1 Flowchart to Remedy the Troubles

Figure 5.1 shows the flowchart to remedy the troubles from when power to the emulator is activated until the emulator debugger starts up. Check this while the user system is disconnected. For the latest FAQs visit the Renesas Tools Homepage.

<http://www.renesas.com/en/tools>

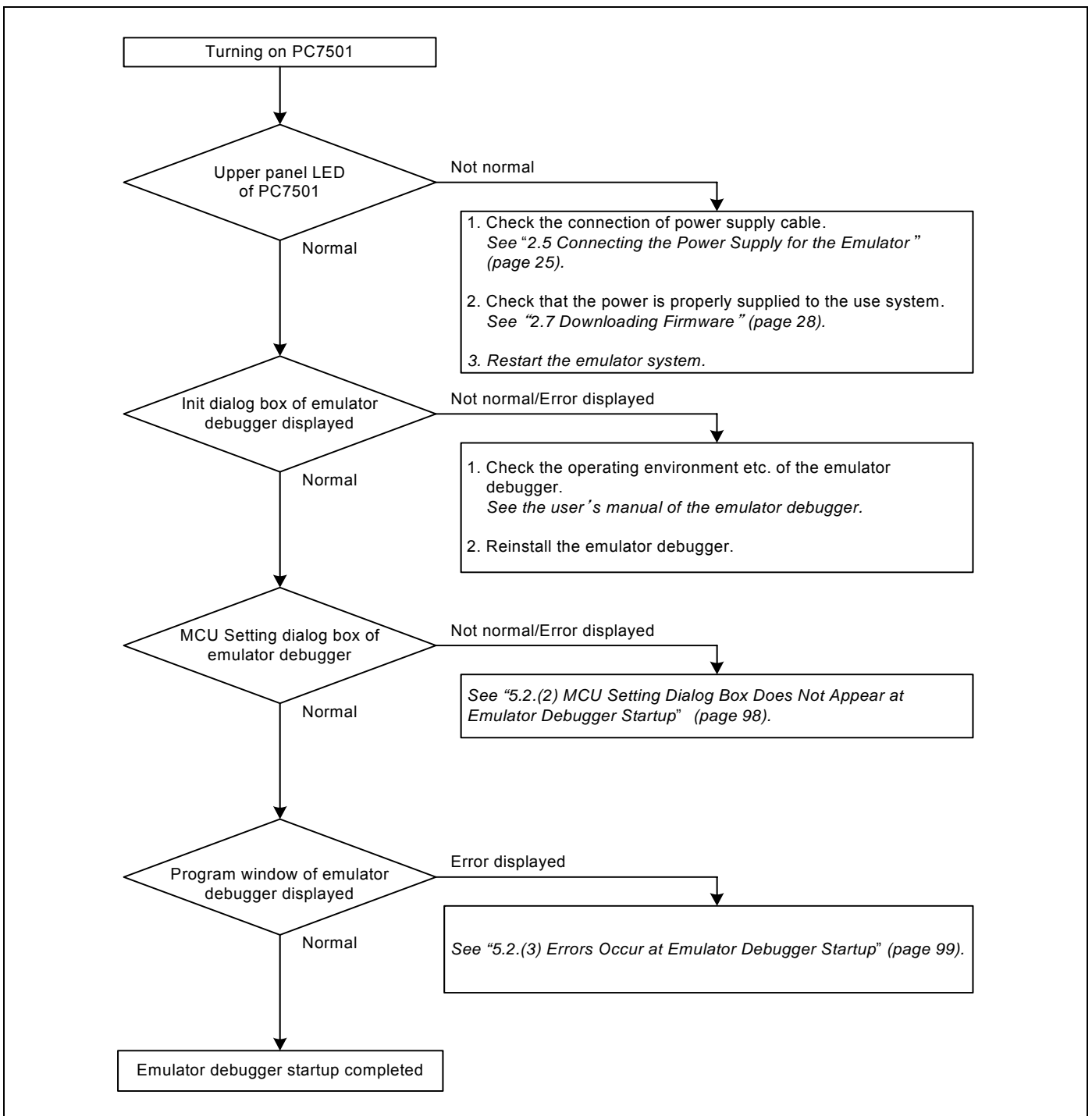


Figure 5.1 Flowchart to remedy the troubles



## 5.2 When the Emulator Debugger Does Not Start Up Properly

### (1) When the LEDs of the PC7501 Do Not Display Normally

Table 5.1 Errors LEDs show and their checkpoints

Error	Connection to the user system	Checkpoint
LEDs do not light up.	-	Check that the power cable is connected. <i>See "2.5 Connecting the Power Supply for the Emulator" (page 25) and the user's manual of the PC7501.</i>
All LEDs remain lit.	-	Recheck the connection between the PC7501 and this product. <i>See "2.4 Connecting the PC7501" (page 24).</i>
Target Status POWER LED does not light up.	Connected	Check that power (Vcc) is properly supplied to the user system and that the user system is properly grounded.
Target Status CLOCK LED does not light up.	Not connected	(1) Check that both the main and sub clocks of the emulator debugger are not set to "EXT". <i>See the CLK command of the emulator debugger.</i> (2) Check the oscillation circuit in the emulator is oscillating. <i>See "2.11 Selecting Clock Supply" (page 42).</i>
	Connected	When the clock is supplied from an external oscillator, check that the oscillator circuit in the user system is oscillating properly.
Target Status RESET LED does not go out.	Connected	Check that the reset pin of the user system is at "H" level.

## (2) MCU Setting Dialog Box Does Not Appear at Emulator Debugger Startup

Table 5.2 Checkpoints of errors at debugger startup

Error	Checkpoint
ERROR 16005: Communication error occurred. Data was not sent to the target.	Check all emulator debugger settings, interface cable settings and switches on the rear of the PC7501 match. <i>See the user's manuals of PC7501 and emulator debugger.</i>
ERROR 16207: User system cannot be properly built.	(1) Download the proper firmware. <i>See "2.7 Downloading Firmware" (page 28).</i> (2) Recheck the connection between the PC7501 and this product. <i>See "2.4 Connecting the PC7501" (page 24).</i>
ERROR 16211: PD30F's version is not the same version as the firmware in the target.	Download the proper firmware. <i>See "2.7 Downloading Firmware" (page 28).</i>
ERROR 16614: Target MCU is in the reset state. Reset the user system.	(1) Check that the reset pin of the user system is at "H" level. (2) Check the reset pin of the user system has changed from "L" to "H" level.
ERROR 16615: Target MCU cannot be reset. Reset the user system.	(1) Check that the NMI# pin is at "H" level. (2) In memory expansion mode or microprocessor mode, check pins RDY# and HOLD# are at "H" level. (3) If the reset circuit of the user system has a watchdog timer, disable the watchdog timer. (4) Check that power is properly supplied to the user system and that the user system is properly grounded.
ERROR 16616: Target is in "HOLD" state.	(1) In memory expansion mode or microprocessor mode, check pins RDY# and HOLD# at "H" level. (2) The MCU is either in stop mode or wait mode. Either reset the MCU or cancel the mode with an interrupt. <i>See MCU specifications.</i>
ERROR 16617: Target clock is stopped.	When the clock is supplied from an external oscillator, check that the oscillator circuit in the user system is oscillating properly.
ERROR 16618: Target MCU is not receiving power.	Check that power is properly supplied to the user system and that the user system is properly grounded.

## (3) Errors Occur at Emulator Debugger Startup

Table 5.3 Checkpoints of errors at debugger startup

Error	Checkpoint
ERROR 16224: Target MCU is uncontrollable. Execute the RESET command.	<ol style="list-style-type: none"><li>(1) Check that the NQPACK etc. mounted on the user system is soldered properly.</li><li>(2) Check that the connector is installed properly to the user system.</li><li>(3) In memory expansion mode or microprocessor mode, check pins RDY# and HOLD# are at "H" level.</li></ol>

### 5.3 How to Request for Support

After checking the items in "5 Troubleshooting", fill in the text file which is downloaded from the following URL, then send the information to your local distributor.

<http://tool-support.renesas.com/eng/toolnews/registration/support.txt>

For prompt response, please specify the following information:

(1) Operating environment

- Operating voltage: \_\_\_\_\_ [V]
- Operating frequency: \_\_\_\_\_ [MHz]
- Clock supply to the MCU: Internal oscillator/External oscillator

(2) Condition

- The emulator debugger starts up/does not start up
- The error is detected/not detected in the self-check
- Frequency of errors: always/frequency ( \_\_\_\_\_ )

(3) Problem

## 6. Maintenance and Guarantee

This chapter describes how to maintenance, repair provisions and how to request for repair.

### 6.1 User Registration

When you purchase our product, be sure register as a user. For user registration, refer to "User registration" (page 11) of this user's manual.

### 6.2 Maintenance

- (1) If dust or dirt collects on any equipment of your emulation system, wipe it off with a dry soft cloth. Do not use thinner or other solvents because these chemicals can cause the equipment's surface coating to separate.
- (2) When you do not use this product for a long period, for safety purposes, disconnect the power cable from the power supply.

### 6.3 Guarantee

If your product becomes faulty within one year after its purchase while being used under good conditions by observing "IMPORTANT" and "Precautions for Safety" described in this user's manual, we will repair or replace your faulty product free of charge. Note, however, that if your product's fault is raised by any one of the following causes, we will repair it or replace it with new one with extra-charge:

- Misuse, abuse, or use under extraordinary conditions
- Unauthorized repair, remodeling, maintenance, and so on
- Inadequate user's system or misuse of it
- Fires, earthquakes, and other unexpected disasters

In the above cases, contact your local distributor. If your product is being leased, consult the leasing company or the owner.

### 6.4 Repair Provisions

#### (1) Repair with extra-charge

The products elapsed more than one year after purchase can be repaired with extra-charge.

#### (2) Replacement with extra-charge

If your product's fault falls in any of the following categories, the fault will be corrected by replacing the entire product instead of repair, or you will be advised to purchase new one, depending on the severity of the fault.

- Faulty or broken mechanical portions
- Flaw, separation, or rust in coated or plated portions
- Flaw or cracks in plastic portions
- Faults or breakage caused by improper use or unauthorized repair or modification
- Heavily damaged electric circuits due to overvoltage, overcurrent or shorting of power supply
- Cracks in the printed circuit board or burnt-down patterns
- Wide range of faults that makes replacement less expensive than repair
- Unlocatable or unidentified faults

(3) Expiration of the repair period

When a period of one year elapses after the model was dropped from production, repairing products of the model may become impossible.

(4) Transportation fees at sending your product for repair

Please send your product to us for repair at your expense.

### 6.5 How to Make Request for Repair

Fill in the Repair Request Sheet included with this product, then send it along with this product for repair to your local distributor. Make sure that information in the Repair Request Sheet is written in as much detail as possible to facilitate repair.

## CAUTION

### Note on Transporting the Product:

- When sending your product for repair, use the packing box and cushion material supplied with this product when delivered to you and specify handling caution for it to be handled as precision equipment. If packing of your product is not complete, it may be damaged during transportation. When you pack your product in a bag, make sure to use conductive polyvinyl supplied with this product (usually a blue bag). When you use other bags, they may cause a trouble on your product because of static electricity.

---

Emulation Probe for M16C/6N Group M16C/6N4, M16C/6N5, M16C/6NK,  
M16C/6NL, M16C/6NM, M16C/6NN  
User's Manual  
M306NKT-EPB

Publication Date: Oct. 16, 2006      Rev.2.00

Published by:      Sales Strategic Planning Div.  
                         Renesas Technology Corp.

Edited by:            Microcomputer Tool Development Department  
                         Renesas Solutions Corp.

---

© 2006. Renesas Technology Corp. and Renesas Solutions Corp., All rights reserved. Printed in Japan.

M306NKT-EPB  
User's Manual

