RENESAS

M306H2T-RPD-E

Emulation Pod for M16C/6H Group M306H2

User's Manual

Renesas Technology
www.renesas.com

Keep safety first in your circuit designs!

- Renesas Technology Corporation and Renesas Solutions Corporation put 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 nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Renesas Technology 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 Corporation, Renesas Solutions Corporation or a third party.
- Renesas Technology Corporation and Renesas Solutions Corporation assume 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.
- 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 Corporation and Renesas Solutions Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation, Renesas Solutions Corporation or an authorized Renesas Technology 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 Corporation and Renesas Solutions Corporation assume 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 Corporation and Renesas Solutions Corporation by various means, including the Renesas home page (http://www.renesas.com).
- 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 Corporation and Renesas Solutions Corporation assume no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Renesas Technology 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 Corporation, Renesas Solutions Corporation or an authorized Renesas Technology 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.
- The prior written approval of Renesas Technology Corporation and Renesas Solutions Corporation is necessary to reprint or reproduce in whole or in part these materials.
- 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.
- Please contact Renesas Technology Corporation or Renesas Solutions Corporation for further details on these materials or the products contained therein.

Precautions to be taken when using this product

- 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.
- In no event shall Renesas Solutions Corporation be liable for any consequence arising from the use of this product.
- Renesas Solutions Corporation 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 Corporation guarantees the renovation or the provision under any circumstances.
- 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.
- Do not attempt to modify this equipment. If modified, your authority to operate this equipment might be voided by FCC.
  Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.
  Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

For inquiries about the contents of this document or product, fill in the text file the installer of the emulator debugger generates in the following directory and email to your local distributor.

\SUPPORT\Product-name\SUPPORT.TXT

Renesas Tools Homepage http://www.renesas.com/en/tools

Preface

The M306H2T-RPD-E is an emulation pod for M306H2 of M16C/6H Group of 16-bit MCUs. It is used with a PC4701 emulator.

This user's manual mainly describes specifications of the M306H2T-RPD-E emulation pod and how to setup it. For details on the following products, which are used with M306H2T-RPD-E, refer to each product's user's manual.

- Emulator: PC4701 User's Manual
- Emulator debugger: Emulator debugger for PC4701 series User's Manual

All the components of this product are shown in "2.2 Package Components" (page 23) of this user's manual. If there is any question or doubt about this product, contact your local distributor.

To use the product properly

Precautions for Safety

- In both this 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.

- The icons' graphic images and meanings are given in "Chapter 1. Precautions for Safety" (page 7). Be sure to read this chapter before using the product.

When using outside Japan

- When using in Europe, the United States, or Canada, be sure to use both the emulator and the emulation pod which meet overseas standards. EMI standards are not met when the M306H2T-RPD-E is used with the PC4700H or PC4700L emulator.

Contents

MEMO

Chapter 1. Precautions for Safety

This chapter describes precautions for using this product safely and properly. For precautions for the emulator main unit and the emulator debugger, refer to each user's manual included with your product.

Chapter 1. Precautions for Safety

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.

1.1 Safety Symbols and Meanings

⚠ **WARNING** — If the requirements shown in the "WARNING" sentences are ignored, the equipment may cause serious personal injury or death.

⚠ **CAUTION** — If the requirements shown in the "CAUTION" sentences are ignored, the equipment may malfunction.

IMPORTANT — It means important information on using this product.

In addition to the three 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.**

The following pages describe the symbols "WARNING", "CAUTION", and "IMPORTANT".

⚠ WARNING

Warning for Installation:

- Do not set this product in water or areas of high humidity. Spilling water or some other liquid into the main unit can cause an unrepairable damage.

Warnings for Use Environment:

- The emulation pod is air-cooled with the ventilation slot. Therefore, do not block the ventilation slot. When heated to high temperatures, the emulation pod may not work properly.

- 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

Caution to Be Taken for Modifying This Product:

- Do not disassemble or modify this product. Disassembling or modifying this product can cause damage. Disassembling and modifying the product will void your warranty.

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. Static electricity may damage the internal circuits.

- Do not pull the emulation pod main unit by the flexible cable (FLX120-RPD) for connecting to the emulator main unit or the flexible cable (FLX64, FLX100 or FLX160) for connecting the target system. The cable may cause a break.

- Flexible cable (FLX120-RPD) for connecting to the emulator main unit and the flexible cable (FLX64, FLX100 or FLX160) for connecting the target system are different from earlier models. The slits make them more flexible. However, excessive flexing or force may break conductors.

- Do not use inch-size screws for this equipment. The screws used in this equipment are all ISO (meter-size) type screws. When replacing screws, use same type screws as equipped before.

IMPORTANT

Note on Malfunctions in the PC4701 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.

IMPORTANT

Notes on Downloading Firmware:

- Before using this product for the first time, it is necessary to download the dedicated firmware (control software for the emulation pod built into the PC4701). Please note that, to do this, it is necessary to start up the PC4701 in the maintenance mode. For firmware download procedures, see "4.2 Downloading Firmware" (page 48). 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 power is shut off unexpectedly, redownload the firmware.

- Except when a target status error occurs, if self-checks are not completed successfully, there may be trouble with the product. In such case, contact your sales representative. However, perform self-checks in the below setup.

    (1) Target system: Not connected
    (2) Emulation pod internal switches: At their factory settings

Note on When the Emulator Debugger Ends:

- To restart the emulator debugger after it ends, always shut power to the emulator module off once and then on again.

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.

Notes on Target System:

- The Vcc pin of emulator is connected to the target system to observe the voltage of the target system. Therefore design your system so that the emulator MCU is powered by the target system.

- The voltage of the target system should be within the range of the MCU specification.

- Do not change the voltage of the target system after turning on the power.

- Before powering on your emulator system, check that the host machine, the emulator main unit, the converter board and target system are all connected correctly. Next, turn on the power to each equipment following the procedure below.

    (1) Turn ON/OFF the target system and the PC4701 emulator as simultaneously as possible.

    (2) When the PC4701 and emulator debugger start up, check the target status LEDs on the emulator main unit's front panel to see if this product is ready to operate.

       • Is the power supplied?   Check target status LED (POWER) is ON.

       For details, refer to "Chapter 4. Usage" (page 45).

Notes on Stack Area:

- With this product, a maximum 8 bytes of the user stack is consumed.

- 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. Therefore, ensure the +8 bytes maximum capacity used by the user program as the user stack area.

IMPORTANT

Notes on MAP References and Settings:

- For details on referencing and setting MAP information, see user's manual of the emulator debugger.

- Be sure to set the SFR area to EXTERNAL (an external section).

- When setting 0FFFCh - 0FFFFh to EXTERNAL:
  This product uses the 4 bytes area 0FFFCh - 0FFFFh as the stack area. If this 4 bytes memory cannot be read or written to, reset cannot be properly effected. As a result, you need to alter the map settings if either of the conditions given below is met.

  (1) With the system which shifts from the single-chip mode to the memory expansion (or microprocessor) mode, using the 4 bytes area of 0FFFCh - 0FFFFh set to EXTERNAL.
  (2) With the system which starts up in the microprocessor mode, using the 4 bytes area of 0FFFCh - 0FFFFh set to EXTERNAL and there is not enough memory to read or write to.

  The procedures to alter the MAP settings when the condition (1) or (2) above is met are shown below.

  (1) Set the 4 bytes area of 0FFFCh - 0FFFFh to INTERNAL.
  (2) Execute the RESET command of the emulator debugger.
  (3) Set the stack pointer.

      (Example)

      RESET:

              FCLR   I
              LDC   #0480H, SP   --> Set the stack pointer.
              .   (Stop the program after executing this instruction.)

  (4) Set the 4 bytes area 0FFFCh - 0FFFFh to EXTERNAL.

Note on Setting the Work Area:

- To use this product, it is necessary to set the work area in the internal reserved area of the MCU. However, do not set it in the last 10 bytes of the internal reserved area. And be sure to set the internal reserved area to INTERNAL. (Set the work area by Init dialog of the emulator debugger.)

  Example 1: When debugging the program of the MCU whose internal reserved area is 02C00h - 03FFFh, set the work area within the range of 02C00h - 03FF6h.

  Example 2: When debugging the program of the MCU whose internal reserved area is 05400h - 05FFFh, set the work area within the range of 05400h - 05FF6h.

  For instance, when setting the work area at 05C00h, the emulator uses 10 bytes area of 05C00h - 05C09h.

IMPORTANT

Note on RESET* Input:

• The RESET* input from the target system is accepted only while a user program is being executed (only while the RUN status LED on the PC4701's front panel is lit).

Note on HOLD* Input:

• Be sure to input "L" to the HOLD* pin of the target system during the user program execution (while the RUN status LED on the PC4701's front panel is lit). Inputting "L" to the HOLD* pin when stopping the user program may cause a malfunction of the emulator.

Note on RDY* Input:

• Be sure to input "L" to the RDY* pin of the target system during the user program execution (while the RUN status LED on the PC4701's front panel is lit). Inputting "L" to the RDY* pin when stopping the user program may cause a malfunction of the emulator.

Notes on NMI* Input:

• NMI* input from the target system is accepted only while a user program is being executed (only while the RUN status LED on the PC4701's front panel is lit).

• NMI* interrupts are generated in the following situations.

(1) Immediately after the NMI* pin changes from "H" to "L" level during the target program execution

(2) After a reset is released by the target system when the NMI* pin is "L" level during the target program execution

(3) Immediately after program execution starts with the NMI* pin "L" level, following the reset input from the emulator debugger or emulator main unit

(4) Immediately after program execution starts when the NMI* pin is "H" level immediately before the target program stops and changes to "L" level when target program execution starts

Note on Clock Supply to the MCU:

• Clock can be supplied to the evaluation MCU in one of the following two ways. This is determined by emulator debugger clock selection.

(1) When Internal is selected:

The clock generated by the oscillation circuit in the emulation pod is supplied to the evaluation MCU. The clock is continually supplied to the evaluation MCU regardless of "target system clock status" and "user program execution status".

(2) When External is selected:

Clock supply to the evaluation MCU depends on oscillation status (oscillate/off) of the target system.

IMPORTANT

Notes on Interruption:

- Even when the target program is not being executed (while the target program is stopped or during runtime debugging), the evaluation MCU keeps running so as to control the emulation pod. Therefore, note that timers and other components do not stop running even though the target program is not being executed.

- If a maskable interrupt request is generated when the target program is not being executed (while the target program is stopped or during runtime debugging), the request is not accepted because the emulator disables all interrupts. If an interrupt request is generated when the target program is not being executed, but the target program enables that interrupt, it will be accepted immediately after program execution starts.

Note on Address 00000h Access:

- With the M16C/6H Group MCUs, when a maskable interrupt is generated, the interrupt data (interrupt No. and interrupt request level) stored in address 00000h is read out. Also, the interrupt request bit is cleared when address 00000h is read out. Consequently, when the address 00000h readout instruction is executed or when address 00000h 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 the address 00000h is generated excluding the interrupt, the yellow LED lights up to alarm. When this LED lights, there is a possibility of wrong access to address 00000h, therefore check the program. This LED is turned off by the RESET switch of the emulator main unit.

Note on Stop and Wait Modes:

- Do not perform step execution at addresses in the stop or wait mode. It may cause communication errors.

Note on Software Reset:

- Do not use a software reset.

Note on BRK Instruction and BRK Instruction Interruption:

- With this emulator system, BRK instructions and BRK instruction interruptions are exclusively used for software break functions. Therefore, you can not use them for your program.

Notes on Address Match Interrupt:

- Do not set software breaks at the same addresses as address-match interrupts as the program may run out of control.

- Do not set a hardware break within 4 instructions before an address at which an address-match interrupt occurs. If you do set a hardware break in this range, the program will run out of control.

- When an address at which an address-match interrupt occurs is executed in one-step mode, the program stops after executing the first instruction after returning from the address-match interrupt processing.

IMPORTANT

Note on Commands that Access the Single-step Interrupt Vector Area:

- Do not perform the following debugging operations with the single step interrupt vector area (addresses FFFECh - FFFEFh).

    (1) Step execution of instructions that access the single step interrupt vector area

    (2) Program execution using instructions that access the single step interrupt vector area when a software breakpoint is set in those instructions

Note on DMA Transfer:

- With this product, the 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 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 following registers have been changed to generate DMA transfer as explained here even when the program is stopped.

    DMA0 transfer counter register: TCR0
    DMA1 transfer counter register: TCR1

Notes on Software Breaks and Hardware Breaks:

- Software breaks generate BRK interrupts by substituting the proper instruction to the BRK instruction. Therefore, when referencing the result of a trace in bus mode, "00h" is displayed for the instruction fetch address where a software break is set, and when referencing in reverse assemble mode, "BRK" instruction is displayed.

- It is not possible to use a software break and a hardware break at the same time. If doing so, it may not operate normally.

- In the area where the MAP setting is EXTERNAL, software breaks cannot be used.

Note on Protect Resistor (PRC2):

- Make note of the fact that the protect is not canceled when protect register (PRC2), which enables writing in the port P9 direction registers and the SI/O3, SI/O4 control registers, is changed with the following procedure.

    (1) Step execution of the instruction for setting ("1") PRC2

    (2) Setting the break point from the instruction for setting ("1") PRC2 to when the setting the register for the protect"

    (3) Setting ("1") PRC2 from the dump window or script window

IMPORTANT

Notes on Pullup Control Resistor:

- Ports P00 to P57 are not pulled up by the pullup control resistor.

  When pulling up the ports P00 to P57, apply a resistance to the inside of the emulator. How to apply it, refer to "3.2 Setting Switches and Pullup Resistors" (page 29).

  Note: Ports P60 to P107 are pulled up by the pullup control resistors.
  Note: The pullup control resistors can read and write from P00 to P107 properly.

- When the pullup control resistor 1 (bit 1 of address 3FDh = PU11) of Ports P44 - P47 is set to "1" (pulled up), ports P15 - P17 are pulled up regardless of the value of the pullup control resistor 0 (bit 3 of address 3FCh = PU03) of Ports P14 - P17.

Note on Reading Internal Resources of an MCU:

- When the registers that are listed in Table 1.1 are read with an emulator, those results will be the following (the data in the MCU are not effected).

  (1) Results of real-time trace

  The data values of the cycles read are not displayed correctly.

  (2) Real-time RAM monitor

  The data values read are not displayed correctly.

  Table 1.1 Resisters and symbols not displayed normally

Resister	Symbol
DMA source pointers 0, 1	SAR0, SAR1
DMA destination pointers 0, 1	DAR0, DAR1
DMA transfer counters 0, 1	TCR0, TCR1
DMA control resisters 0, 1	DM0CON, DM1CON

Notes on Watchdog Function:

- The MCU's watchdog timer can be used only while a program is being executed. To use it otherwise, disable the watchdog timer.

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

IMPORTANT

Note on Differences between Actual MCUs and the Emulator:

• Operations of the emulator system differ from those of actual MCUs as listed below.

(1) Reset condition
Set the rise time (0.2 Vcc to 0.8 Vcc) 1 μs or less.

(2) Initial values of MCU's internal resources

(3) Internal memories (ROM and RAM) capacities etc.
With this emulator system, "INT" (emulation memory ON) is the default for mapping areas other than the SFR area (addresses 000h - 3FFh). For this reason, the emulation memory can read and write in areas other than the SFR, internal RAM and internal ROM.

(4) Oscillator circuit
• Make note of the fact that in the oscillator circuit where a resonator is connected between the X_{IN} and X_{OUT} pins, oscillation does not occur because a flexible cable, buffer IC and other devices are used between the evaluation MCU and the target system. It is same for sub-clock oscillator circuits (X_{CIN} and X_{COUT}).

• For note on when using the oscillator circuit on the target system, refer to "3.3 (1) Using the Oscillator Circuit on the Target System" (page 36).

(5) Characteristics of ports P00 to P57
With this product, ports P00 to P57 are connected via an emulation circuit. The device used for the port emulation circuit is as follows.

Device: M60081L-0142FP

(6) HOLD* control
When inputting "L" to the HOLD* pin to run into the HOLD state, P00 to P52 will be in the HOLD state delaying by 2.5 cycles than the actual MCU (see Figure 5.5 and Table 5.5).

(7) DBC, Single-step and BRK instruction interrupt vector table addresses
You can download data to the DBC, Single-step and BRK instruction interrupt vector table addresses. However, the data read out from this area is different from expected values, because the emulator system uses this area (see Table 1.2).

Table 1.2 Vector table addresses for the emulator

Factor of interruption	Vector table addresses	Data for reading
DBC[1]	FFFF4h - FFFF7h	Indefinite
Single-step[1]	FFFECh - FFFEFh	Indefinite
BRK instruction	FFFE4h - FFFE7h	Indefinite

*1 Interruption for the debugger only

(8) A-D conversion
As a analog switch, flexible cable, pitch converter board and other devices are used between the evaluation MCU and the target system, some characteristics are slightly different from those of the actual MCU.

(9) D-A conversion
As a flexible cable, pitch converter board and other devices are used between the evaluation MCU and the target system, some characteristics are slightly different from those of the actual MCU.

(10) Outputs of the actual MCU and this product
Outputs of this product are determined according to the user program as listed in Tables 1.3 to 1.6.

IMPORTANT

Table 1.3 Differences of outputs between actual MCUs and this product (when executing program)

Access area	Signal	Actual MCU	This product	∗ for difference
SFR area (MAP = EXT)	Address BHE*	Outputs	Outputs	
	Data	Outputs (floating during read cycle)	Floating	∗
	RD* WR*	Outputs ("L" output at write cycle, "L" output at read cycle)	Does not output (fixed to "H" output)	∗
	CS*	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	
	ALE	Does not output (fixed to "L" output)	Outputs	∗
Internal RAM Internal ROM	Address BHE*	Retains previous status	Outputs	∗
	Data	Floating	Floating	
	RD* WR*	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	
	CS*	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	
	ALE	Does not output (fixed to "L" output)	Outputs	∗
External area	Address BHE*	Outputs	Outputs	
	Data	Outputs (inputs external data at read cycle)	Outputs (inputs external data at read cycle)	
	RD* WR*	Outputs ("L" output at write cycle, "L" output at read cycle)	Outputs ("L" output at write cycle, "L" output at read cycle)	
	CS*	Outputs	Outputs	
	ALE	Outputs	Outputs	

IMPORTANT

Table 1.4 Differences of outputs between actual MCUs and this product (in stop mode)

Access area	Signal	Actual MCU	This product	∗ for difference
-	Address BHE*	Retains previous status	Retains previous status	
	Data	Retains previous status	Floating	∗
	RD* WR*	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	
	CS*	Retains previous status	Retains previous status	
	ALE	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	

Table 1.5 Differences of outputs between actual MCUs and this product (in wait mode)

Access area	Signal	Actual MCU	This product	∗ for difference
-	Address BHE*	Retains previous status	Retains previous status	
	Data	Retains previous status	Floating	∗
	RD* WR*	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	
	CS*	Retains previous status	Retains previous status	
	ALE	Does not output (fixed to "H" output)	Does not output (fixed to "H" output)	

Table 1.6 Outputs of this product when the user program is stopping

Signal	This product
Address BHE*	Outputs
Data	Floating
RD* WR*	Outputs RD* only (WR* fixed to "H" output)
CS*	Outputs
ALE	Outputs

MEMO

Chapter 2. Preparation

This chapter describes the package components, the system configuration and the preparation for using this product for the first time.

Chapter 2. Preparation

2.1 Terminology

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

Emulator system

This means an emulator system built around the PC4701 emulator. The PC4701 emulator system is configured with an emulator main unit, emulation pod, host machine and emulator debugger.

Emulator main unit (Hereafter PC4701)

This means a generic name for emulators for 8 and 16-bit MCUs. For details on specific models of PC4701, visit the Renesas Tools Homepage.

<div align="center">http://www.renesas.com/en/tools</div>

Emulation pod

This means M306H2T-RPD-E (this product). This emulation pod is for M16C/6H Group M306H2.

Host machine

This means a personal computer used to control the emulator and emulation pod.

Emulator debugger

This means a software tool M3T-PD30 to control the emulator from the host machine through an interface.

Firmware

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

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 data 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.

Evaluation MCU

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

Target MCU

This means the MCU you are going to debug.

Target system

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

*

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

2.2 Package Components

The M306H2T-RPD-E package consists of the following items. When unpacking, check to see if your M306H2T-RPD-E contains all of these items.

Package components

Item	Quantity
M306H2T-RPD-E emulation pod main unit	1
FLX120-RPD flexible cable for connecting PC4701	1
FLX160 flexible cable for connecting pitch converter board (pre-installed in emulation pod)	1
FLX160-PRB pitch converter board for connecting target system (pre-mounted on FLX160)	1
M306H2T-PTC pitch converter board for connecting 116-pin 0.65-mm-pitch (116P6A-A) foot pattern (includes YQPACK116SB×1, NQPACK116SB×1 and YQ-GUIDE×4)	1
OSC-2 oscillator circuit board	1
Hardware tool user registration FAX sheet (English)	1
Hardware tool user registration FAX sheet (Japanese)	1
M306H2T-RPD-E user's manual (this manual)	1
M306H2T-RPD-E user's manual (Japanese)	1

Please keep the M306H2T-RPD-E'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 any of these items are missing or found faulty, please contact your local distributor. Also, if there is any question or doubt about the packaged product, contact your local distributor.

2.3 Other Tool Products Required for Development

To bring forward programs development on the M16C/6H Group MCUs, the products listed below are necessary in addition to those contained package above. Get them separately.

Other tool products required for development

Emulator main unit	PC4701
Emulator debugger	M3T-PD30

To purchase these products, contact your nearest distributer.

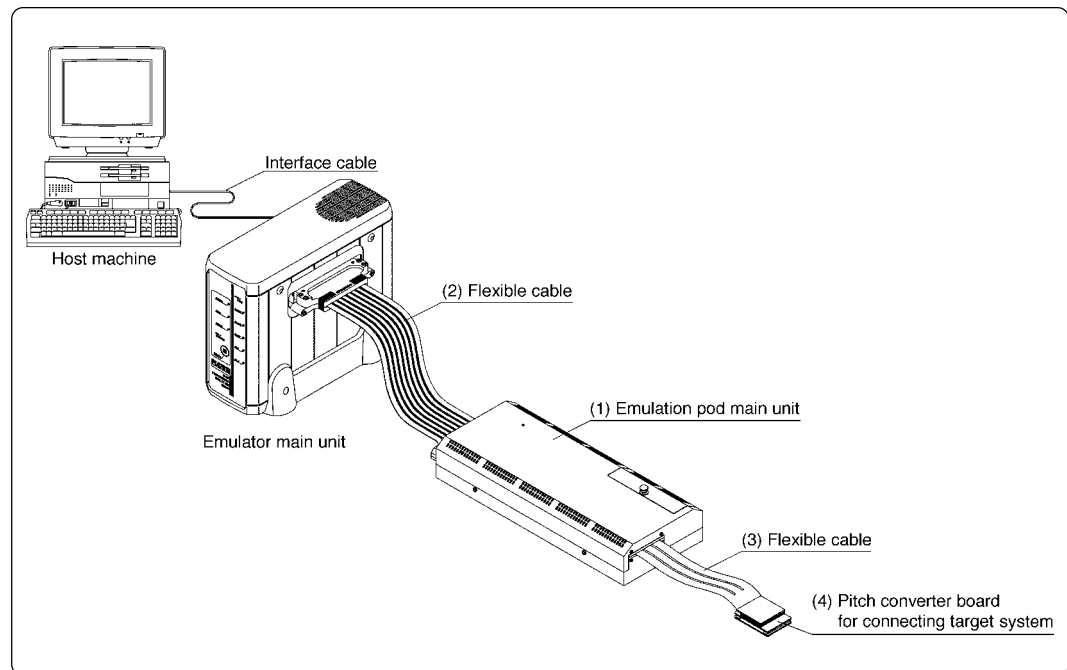## 2.4 Name of Each Part

(1) System Configuration

Figure 2.1 System configuration

(1) to (4) in Figure 2.1 are included with this product package.

(1) Emulation pod (M306H2T-RPD-E)

   This emulation pod contains an evaluation MCU, emulation memory and circuits to feature the debugging functions.

(2) Flexible cable (FLX120-RPD)

   This is a 120-pin flexible cable for connecting the PC4701 emulator and the emulation pod.

(3) Flexible cable (FLX160)

   This is a 160-pin flexible cable for connecting the emulation pod and the target system.

(4) Pitch converter board (FLX160-PRB, M306H2T-PTC)

   These are pitch converter boards for connecting to the target system. For details, refer to "3.7 Connecting the Target System" (page 43).

(2) Inside of the Emulation Pod
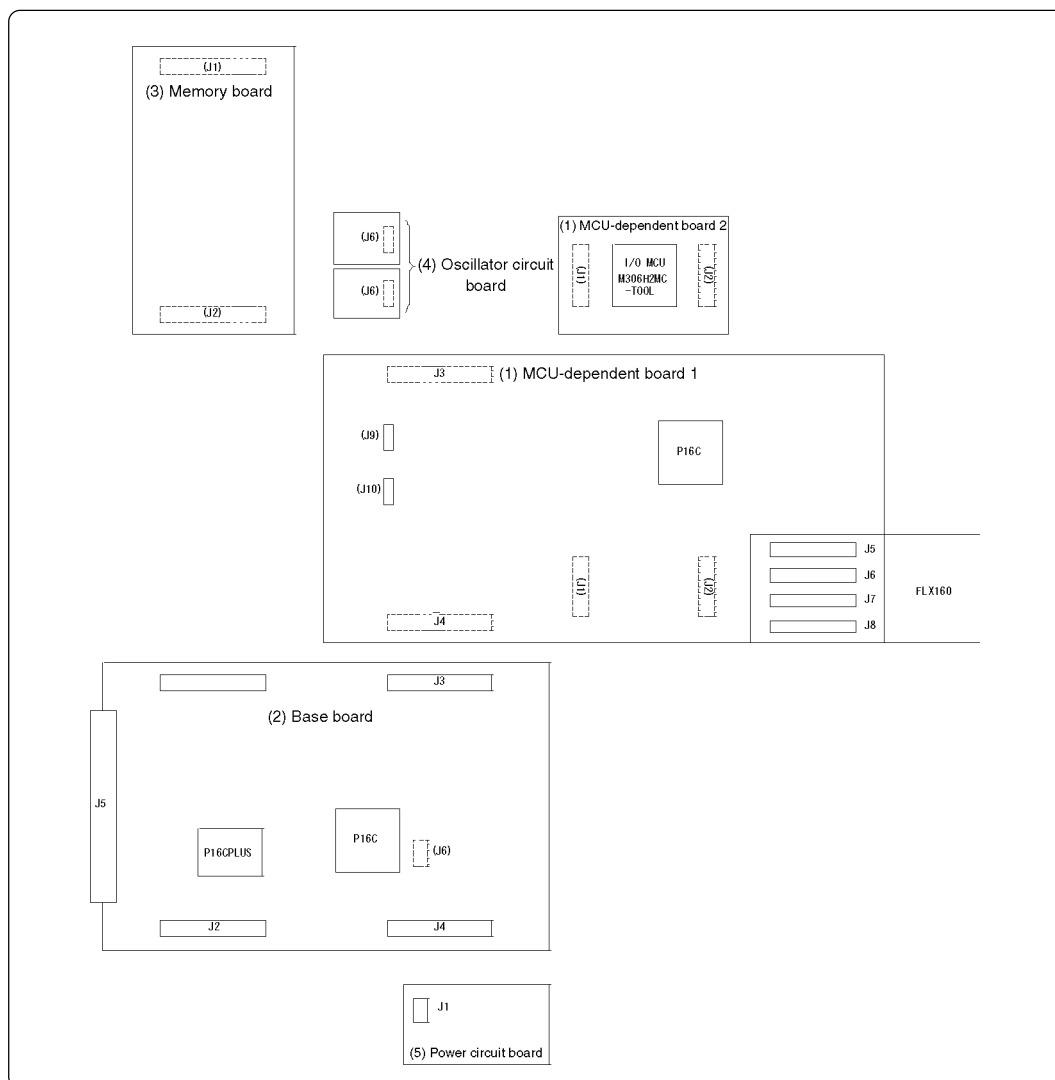
Figure 2.2 Internal view of the emulation pod

(1) MCU-dependent boards 1 and 2

   Boards grouping parts (pins and added functions) which vary according to MCU model.

(2) Base board

   Board for the M16C/60 and M16C/20 Series MCUs which controls the interface with the PC4701 and the evaluation MCU.

(3) Memory board

   Board on which is mounted the emulation memory (1 MB) and the map memory (4 bit×1M) for the M16C/60 and M16C/20 Series MCUs.

(4) Oscillator circuit board

   Oscillator circuit board on which the oscillation module device (10 MHz) and the crystal oscillator (32.768 kHz) are mounted.

(5) Power circuit board

   Power circuit board which determines the power voltage in the pod according to the power voltage level of the target system.

2.5 When Using the Emulator for the First Time

If you have purchased this emulation pod newly, it is necessary to download the firmware. The download procedure is given in Figure 2.3.

Before downloading the firmware, check the emulator debugger is installed and the PC4701 is connected to the host machine. For more information, see each user's manual of the emulator debugger and the PC4701.

Connect the PC4701 and this product.	See "3.6 Connecting the PC4701" (page 41).
Within 2 seconds of turning the power ON, press the RESET switch of the PC4701 and check the maintenance mode is accessed.	See "4.2 Downloading Firmware" (page 48).
Download the firmware using the emulator debugger.	
End the emulator debugger and shut the power of the PC4701 OFF.	
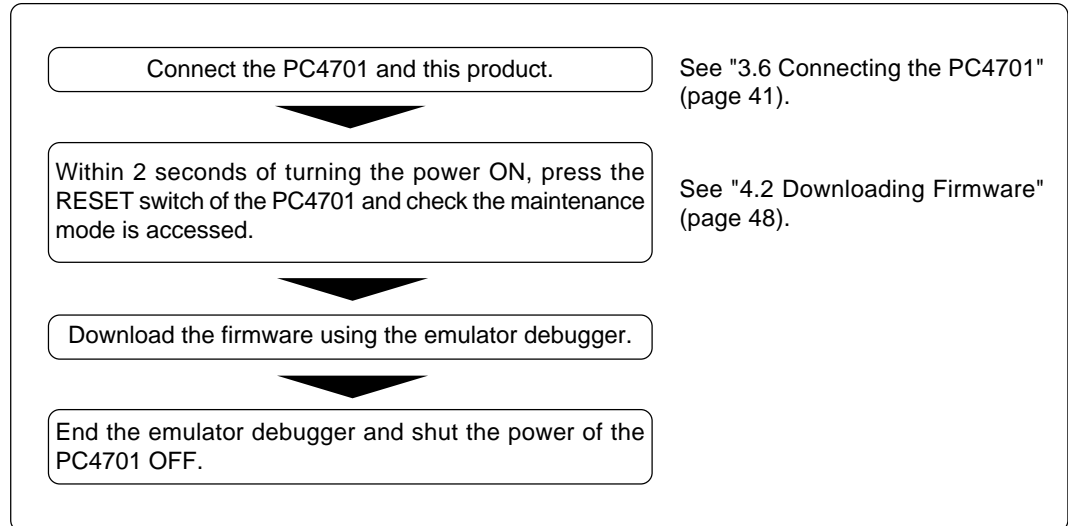
Figure 2.3 Firmware download procedure when the emulator is used for the first time

To make sure the emulation pod works properly, run the self-checks. For self-check procedures, see "4.3 Self-checks" (page 49).

Chapter 3. Setting Up

This chapter describes switch settings required for using this product and how to connect this product to the PC4701 and the target system.

Chapter 3. Setting Up

To use this emulation pod with your target system, it is necessary to set as follows. Set the following after removing the upper cover.

- Change the oscillation frequency in the emulation pod.
- Set the switches depending on the target system.
- Install pullup resistors to ports P00 to P57.
- Install the A-D conversion bypass capacitor.
- Change the input frequency for FSC oscillation.
- Change the constant of the circuit for data slicer I/O signal.

3.1 Removing the Upper Cover

The procedure of removing the upper cover is shown below.

(1) Remove the four screws of both sides of this product and lift off the upper cover. (see Figure 3.1)
(2) Set the switches.
(3) Replace the upper cover and secure the four screws.
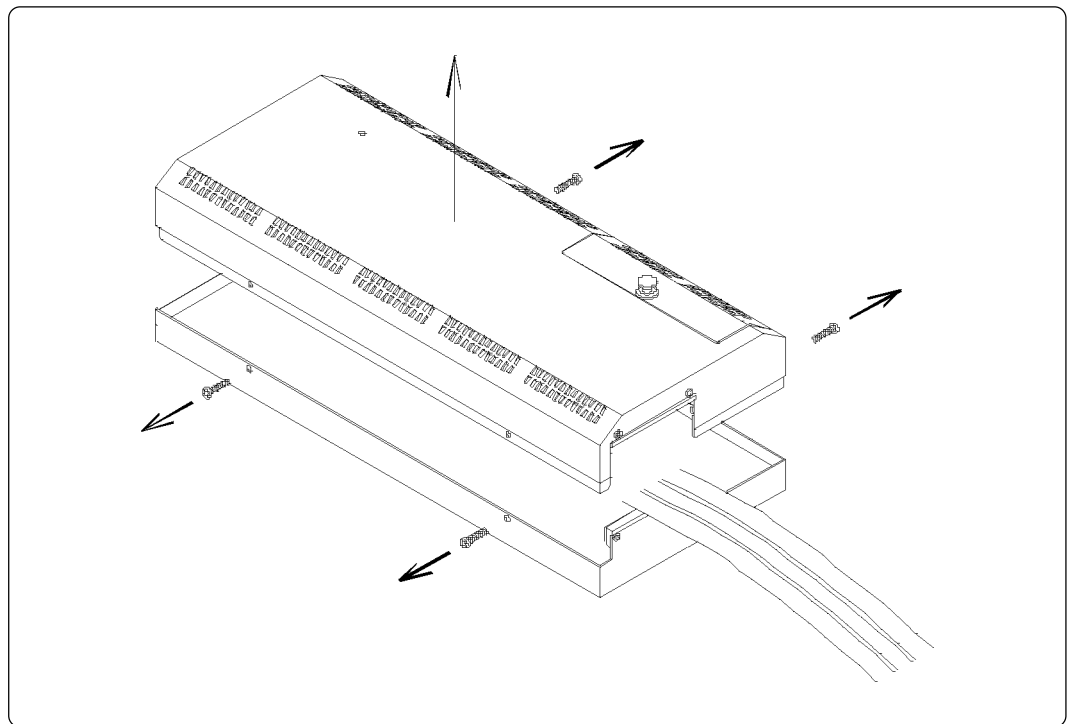
Figure 3.1 Removing the upper cover

⚠ CAUTION

When Removing the Upper Cover:

- Always shut OFF power when removing the upper cover or setting the switches.

- Do not touch the connector pins of the emulator main unit and the target MCU connector pins. Static electricity may damage the internal circuits.

3.2 Setting Switches and Pullup Resistors

Figure 3.2 shows the positions of switches and a position for installing pullup resistors. Tables 3.1 to 3.4 list how to set each switch and Figure 3.3 shows how to install pullup resistors.
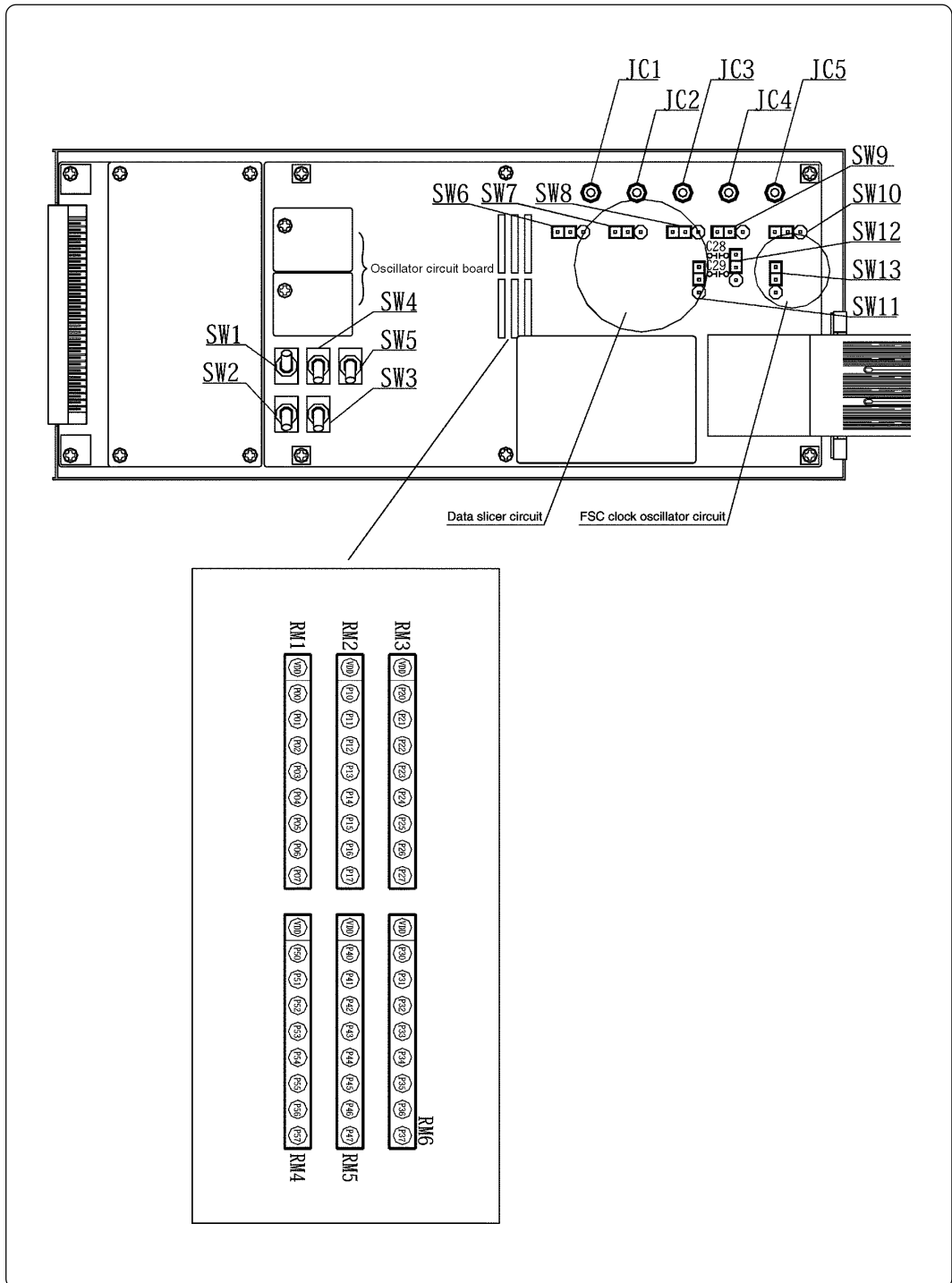
Figure 3.2 Positions of switches and their factory-settings

Table 3.1 Switch settings of the M306H2T-RPD-E (1/4)
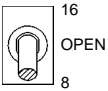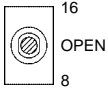
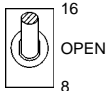| Switch	Signal	Setting	Description
SW1	BYTE	SW1 BYTE (16 OPEN 8)	Pulls up the BYTE pin of MCU with a resistance at 33 kΩ.
		SW1 BYTE (16 OPEN 8)	Does not pull down/up the BYTE pin of MCU.
		SW1 BYTE (Factory-setting) (16 OPEN 8)	Pulls down the BYTE pin of MCU with a resistance at 33 kΩ.
SW2	CNV_{SS}	SW2 CNV_{SS} (Factory-setting) (H OPEN L)	Pulls down the CNV_{SS} pin of MCU with a resistance at 33 kΩ.
		SW2 CNV_{SS} (H OPEN L)	Does not pull down/up the CNV_{SS} pin of MCU.
		SW2 CNV_{SS} (H OPEN L)	Pulls up the CNV_{SS} pin of MCU with a resistance at 33 kΩ.
SW3	X_{OUT}	SW3 X_{OUT} (Factory-setting) (XOUT OPEN)	Does not connect the X_{OUT} pin of MCU to the target system.
		SW3 X_{OUT} (XOUT OPEN)	Connects the X_{OUT} pin of MCU to the target system.

Table 3.2 Switch settings of the M306H2T-RPD-E (2/4)

Switch	Signal	Setting	Description
SW4	P8₇/X$_{CIN}$	SW4 P8₇/X$_{CIN}$ (Factory-setting)	Connects the P8₇/X$_{CIN}$ pin of MCU to the target system. (Uses P8₇/X$_{CIN}$ pin as port P8₇)
		SW4 P8₇/$_{CIN}$	Uses P8₇/X$_{CIN}$ pin as X$_{CIN}$.
SW5	P8₆/X$_{COUT}$	SW5 P8₆/X$_{COUT}$ (Factory-setting)	Connects the P8₆/X$_{COUT}$ pin of MCU to the target system. (Uses P8₆/X$_{COUT}$ pin as port P8₆)
		SW5 P8₆/X$_{COUT}$	Connects the P8₆/X$_{COUT}$ pin of MCU to the target system. (Uses P8₆/X$_{COUT}$ pin as X$_{COUT}$ and opens X$_{COUT}$)
		SW5 P8₆/X$_{COUT}$	Connects the P8₆/X$_{COUT}$ pin of MCU to the target system. (Uses P8₆/X$_{COUT}$ pin as X$_{COUT}$ and connect X$_{COUT}$ to the target system)

IMPORTANT

Note on Switch Settings:

- Switches SW1 (CNV$_{SS}$) and SW2 (BYTE) are used for debugging operation without connecting the target system. When connecting the target system, set both of them "OPEN".

Table 3.3 Switch settings of the M306H2T-RPD-E (3/4)
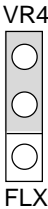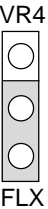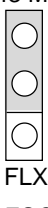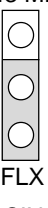
Switch	Signal	Jumper switch setting	
SW6	SYNCIN	SYNC/HOR JC1 [◯ ◯ | ◯] FLX Connects the SYNCIN pin to the RCA connector JC1 on the M306H0T-PRT. (Factory-setting)	SYNC/HOR JC1 [◯ | ◯ ◯] FLX Connects the SYNCIN pin to the target system via the FLX160.
SW7	Unused	JC2 [◯ ◯ | ◯] FLX Only this setting is available. (Factory-setting)	JC2 [◯ | ◯ ◯] FLX Do not use this setting.
SW8	Unused	JC3 [◯ ◯ | ◯] FLX Only this setting is available. (Factory-setting)	JC3 [◯ | ◯ ◯] FLX Do not use this setting.
SW9	CVIN1	CVIN1 JC4 [◯ ◯ | ◯] FLX Connects the CVIN1 pin to the RCA connector JC4 on the M306H0T-PRT. (Factory-setting)	CVIN1 JC4 [◯ | ◯ ◯] FLX Connects the CVIN1 pin to the target system via the FLX160.
SW10	Unused	JC5 [◯ ◯ | ◯] FLX Only this setting is available. (Factory-setting)	JC5 [◯ | ◯ ◯] FLX Do not use this setting.

Table 3.4 Switch settings of the M306H2T-RPD-E (4/4)

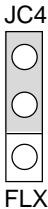| Switch	Signal	Jumper switch setting	
SW11	SVREF	VR4 VREF1 ○ / ○ / ○ FLX Connects the SVREF pin to the changeable resistance VR4 on the M306H0T-PRT. (Factory-setting)	VR4 VREF1 ○ / ○ / ○ FLX Connects the SVREF pin to the target system via the FLX160.
SW12	Unused	JC4 ○ / ○ / ○ FLX Only this setting is available. (Factory-setting)	JC4 ○ / ○ / ○ FLX Do not use this setting.
SW13	FSCIN	4.43 MHz FSCIN ○ / ○ / ○ FLX Connects the FSCIN pin to the 4.43MHz oscillator circuit on the M306H0T-PRT. (Factory-setting)	4.43 MHz FSCIN ○ / ○ / ○ FLX Connects the FSCIN pin to the target system via the FLX160.

It is not possible to control the pullup by the pullup control resistors of ports $P0_0$ to $P5_7$ (possible to read and write by the pullup control resistors).

However, as this product has sockets for installing the pullup resistors, it is possible to pull up by mounting the resistor arrays included with this product. For the positions for installing the pullup resistors, see Figure 3.3.
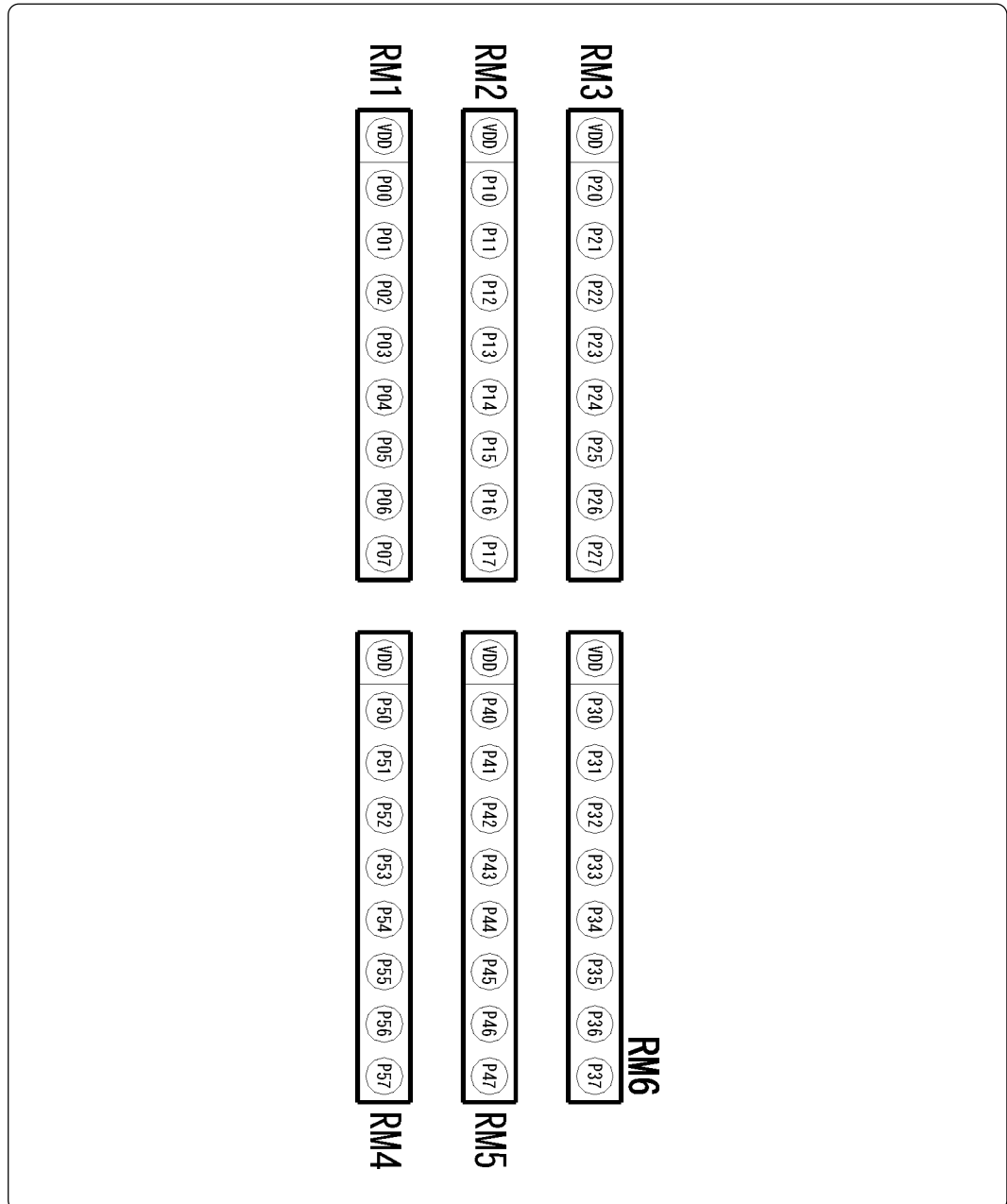
Figure 3.3 Positions for installing the pullup resistors

3.3 Selecting Clock Supply

There are two ways to supply a clock to the MCU, using the oscillator circuit of the emulation pod or using the oscillator circuit on the target system. Table 3.5 lists the factory-settings of each clock supply.

Table 3.5 Clock supply to the MCU

Clock	Description	Display of emulator debugger	Default setting
X_{IN}-X_{OUT}	Internal oscillator circuit of emulation pod (OSC-3: 10 MHz)	Internal	Yes
	Target system	External	-
X_{CIN}-X_{COUT}	Internal oscillator circuit of emulation pod (OSC-2: 32.768 kHz)	Internal	Yes
	Target system	External	-

IMPORTANT

Notes 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.

- For X_{CIN}-X_{COUT} , it is necessary to set switches in the emulation pod. For details, refer to "3.2 Setting Switches and Pullup Resistors" (page 29).

(1) Using the Oscillator Circuit on the Target System

When turning on the power supply, the internal clock of emulation pod is selected to supply the clock to the MCU. To use the external clock on the target system, change the clock by the Init dialog when starting up the emulator debugger or the CLK command on the script window. (For details, refer to the user's manual (or online manual) of the emulator debugger)
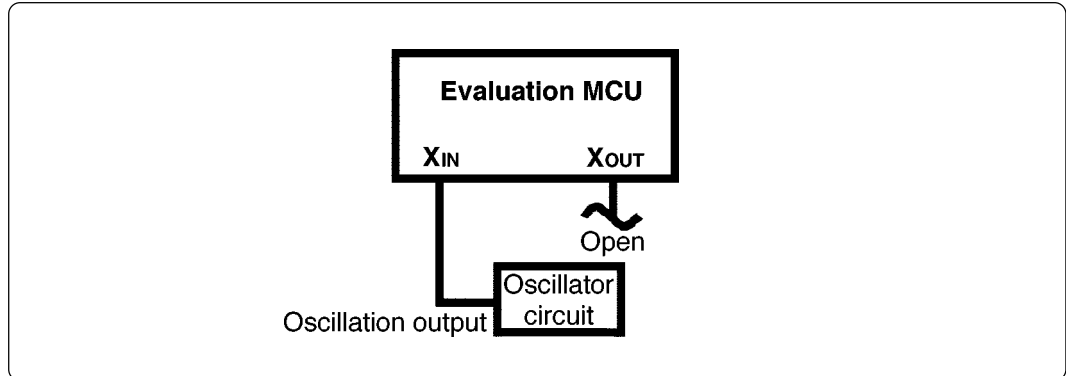
Figure 3.4 External oscillator circuit
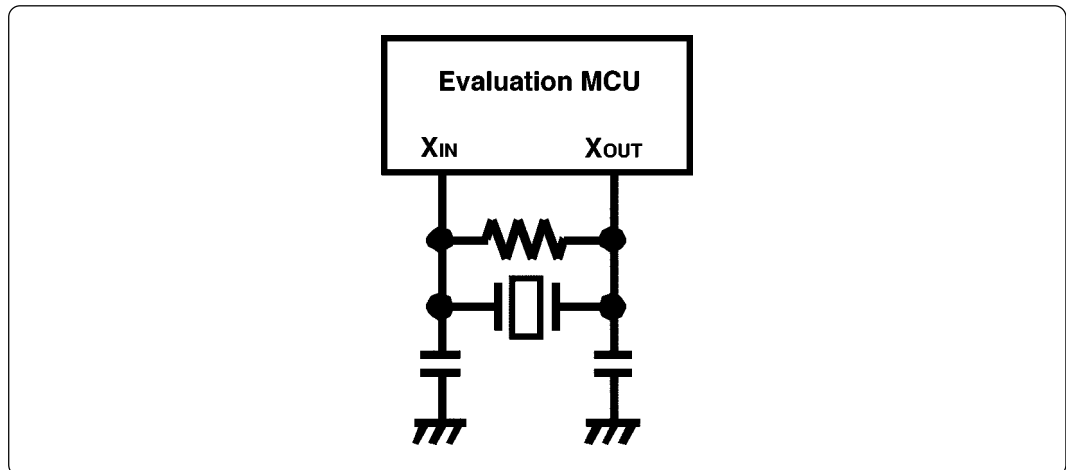
Figure 3.5 Circuit in which oscillation does not occur (same for X_{CIN}-X_{COUT})

IMPORTANT

Notes on External Clock:

- To operate the emulation pod with an external clock, construct the oscillator circuit as shown in Figure 3.4 in the target system and input the oscillator output at 50% duty (within the operating range of the evaluation MCU) into the X_{IN} pin. And the X_{OUT} pin should be open.

- Make note of the fact that in the oscillator circuit shown in Figure 3.5 where a resonator is connected between the X_{IN} and X_{OUT} pins, oscillation does not occur because a flexible cable, buffer IC and other devices are used between the evaluation MCU and the target system. It is same for sub-clock oscillator circuits (X_{CIN} and X_{COUT}).

(2) Changing the Internal Oscillator Circuit of the Emulation Pod

An oscillator circuit board (OSC-3) for 10 MHz is mounted on this product. To use the emulation pod at a frequency other than 10 MHz, build the desired oscillator circuit on the included OSC-2 oscillator circuit board (bare board) and replace the board installed in the emulation pod when shipped from the factory.

Figure 3.6 shows a view of the OSC-2 oscillator circuit board (bare board) and where connector pins are located. Figure 3.7 shows the circuitry of the OSC-2 oscillator circuit board (bare board). Use the number of oscillator circuits recommended by the oscillator manufacturer.

J1-4: GND

J1-3: Oscillator output

J1-2: GND

J1-1: Vcc
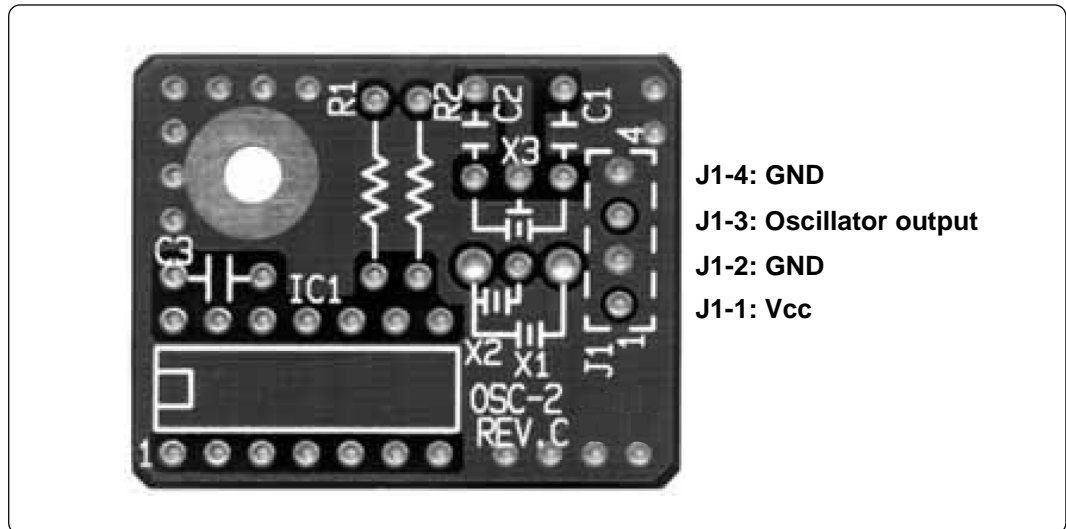
Figure 3.6 External view of the oscillator board (OSC-2) and connector pin assignments

* X1: 5.08-mm-pitch 2-pin oscillator
X2: 2.54-mm-pitch 2-pin oscillator
X3: 2.54-mm-pitch 3-pin oscillator

IC1: Inverter (Unbuffer)
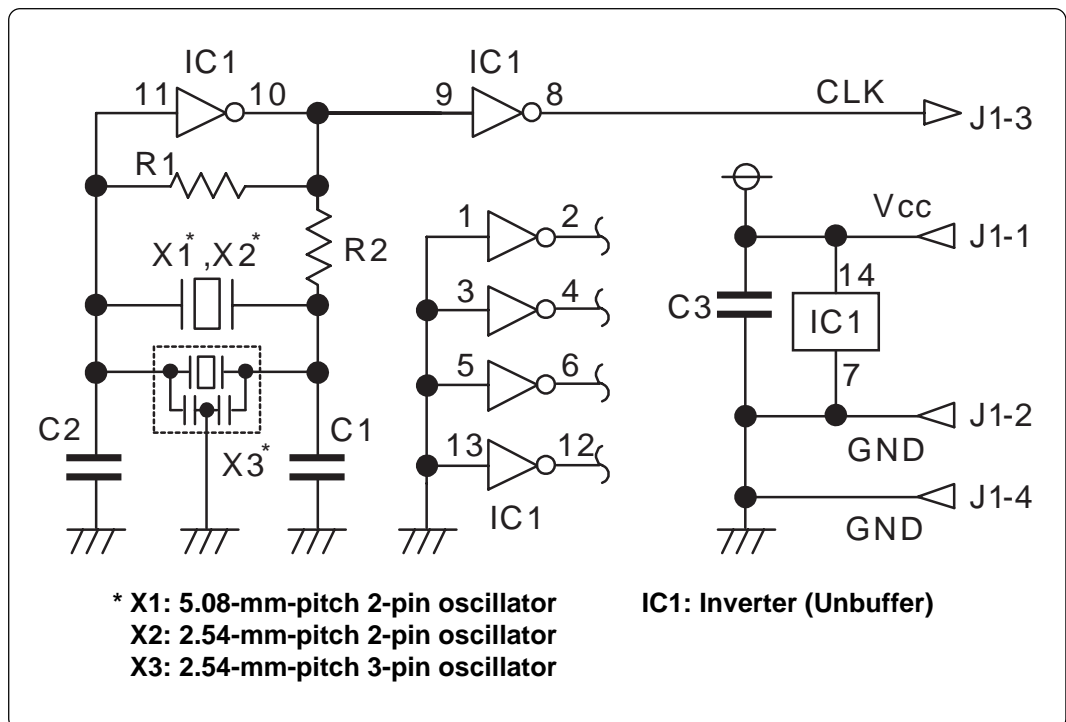
Figure 3.7 Circuit of the oscillator board (OSC-2)

(3) Replacing the Oscillator Circuit Boards

Figure 3.8 shows how to replace the oscillator circuit boards. For the position of the oscillator circuit board, see Figure 3.2.

(1) Unscrew the screw securing the oscillator circuit board.

(2) Lift off the oscillator circuit board.

(3) Attach the J1 connector of another oscillator circuit board for replacement to the connector of the MCU-dependent board.

(4) Secure the oscillator circuit board with the screw.
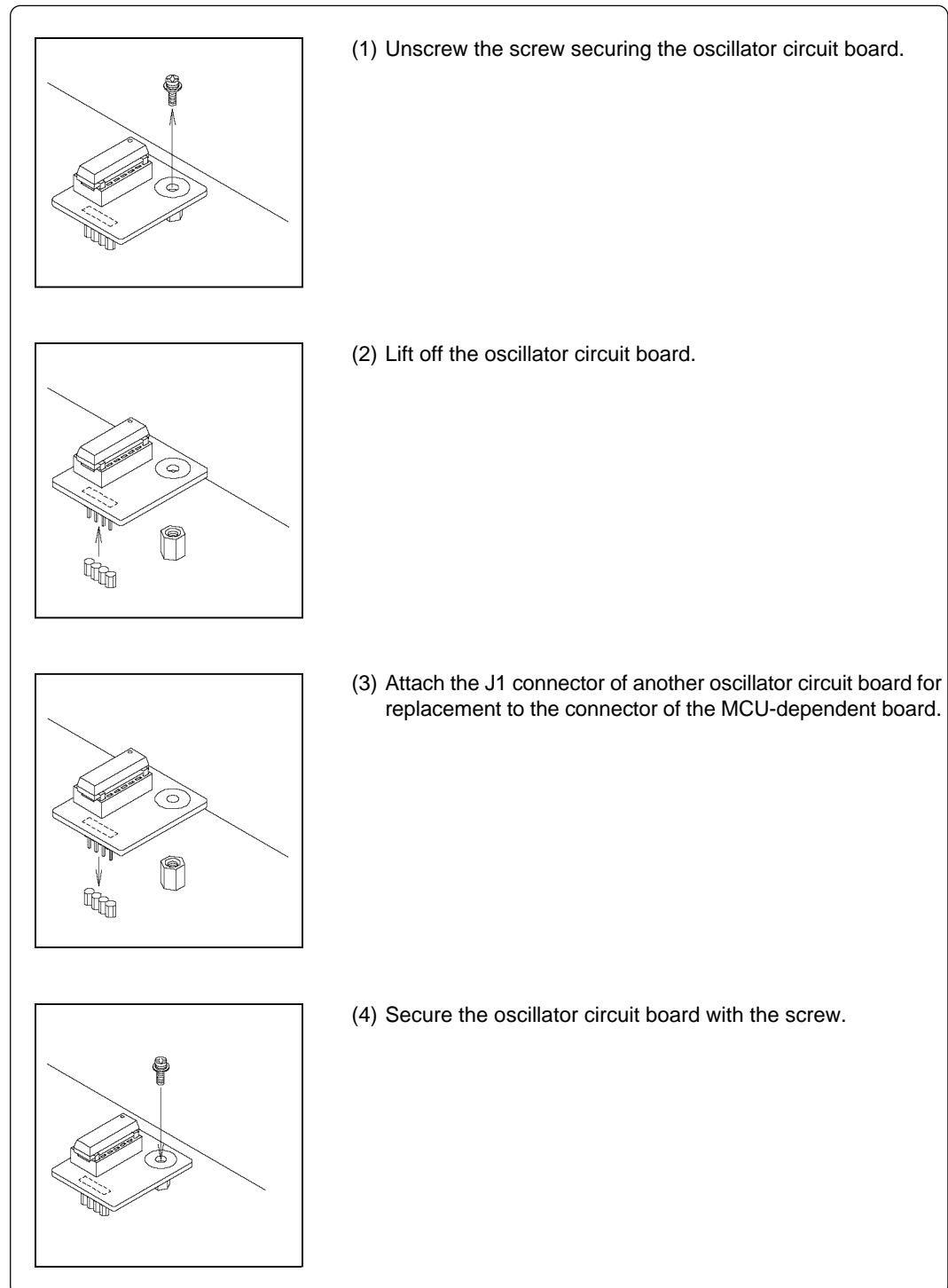
Figure 3.8 Replacing the oscillator circuit boards

3.4 A-D Conversion Bypass Capacitor

With this product, there is a socket (0.1 μF pre-mounted) on the board for changing the bypass capacitor for the A-D conversion circuit. Remove or replace a suitable bypass capacitor as occasion demands.

Figure 3.9 shows where the bypass capacitors are mounted, whereas Figure 3.10 shows an enlargement of the foot patterns.
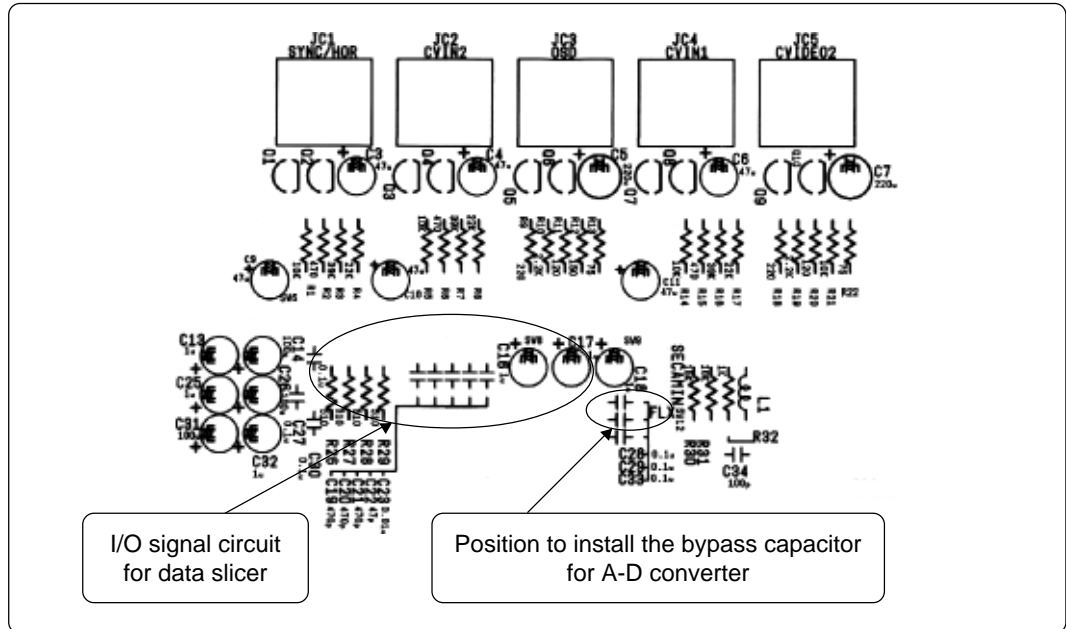
Figure 3.9 Position for mounting the A-D conversion bypass capacitor (M306H2T-PRT board)
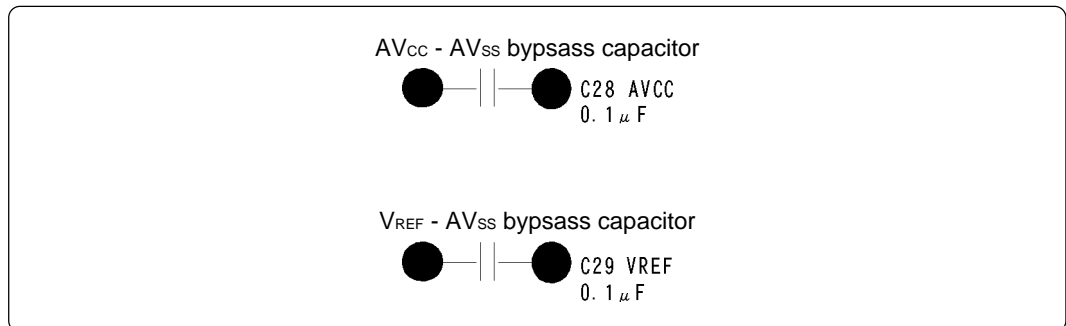
Figure 3.10 Socket for the A-D conversion bypass capacitor

IMPORTANT

Note on the A-D Converter:

- Because a flexible board and other devices are used between the evaluation MCU and the target system, the A-D converter operates differently from the actual MCU. Make the final evaluation of the A-D converter from the actual MCU.

3.5 Data Slicer I/O Signal and FSC Clock I/O Signal Circuit

With this product, the parts which are used for the data slicer and the FSC clock circuit are connected by sockets to be changed. Figure 3.11 shows the position of each part which is used for the data slicer and the FSC clock circuit. Figure 3.2 shows the layout of the parts and Figure 3.12 shows its detailed layout.
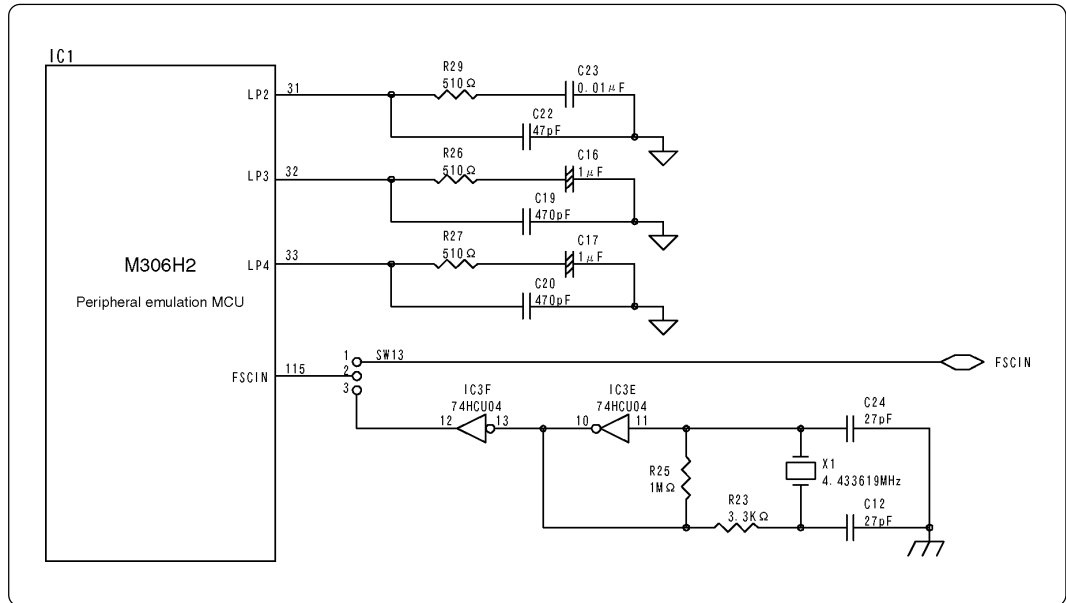
Figure 3.11 Data slicer and FSC clock circuit
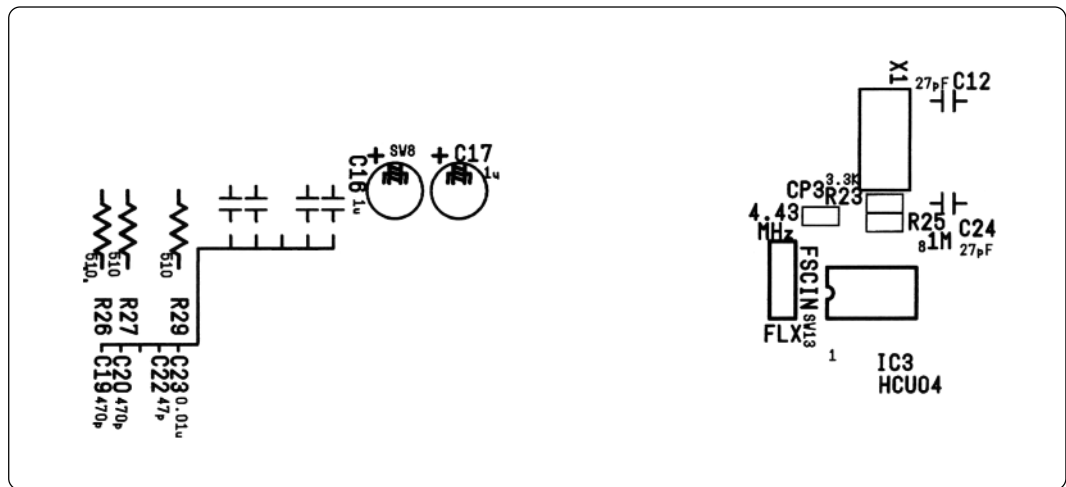
Figure 3.12 Detailed layout of the parts

3.6 Connecting the PC4701

To connect the emulation pod to the PC4701, use the FLX120-RPD 120-pin flexible cable included with this product package. Connect the PC4701 side connector of the FLX120-RPD to the cable connector of the PC4701, then secure with screws the FLX120-RPD.

(1) Connecting the Cable to the PC4701

Figure 3.13 shows how to connect the PC4701 and FLX120-RPD
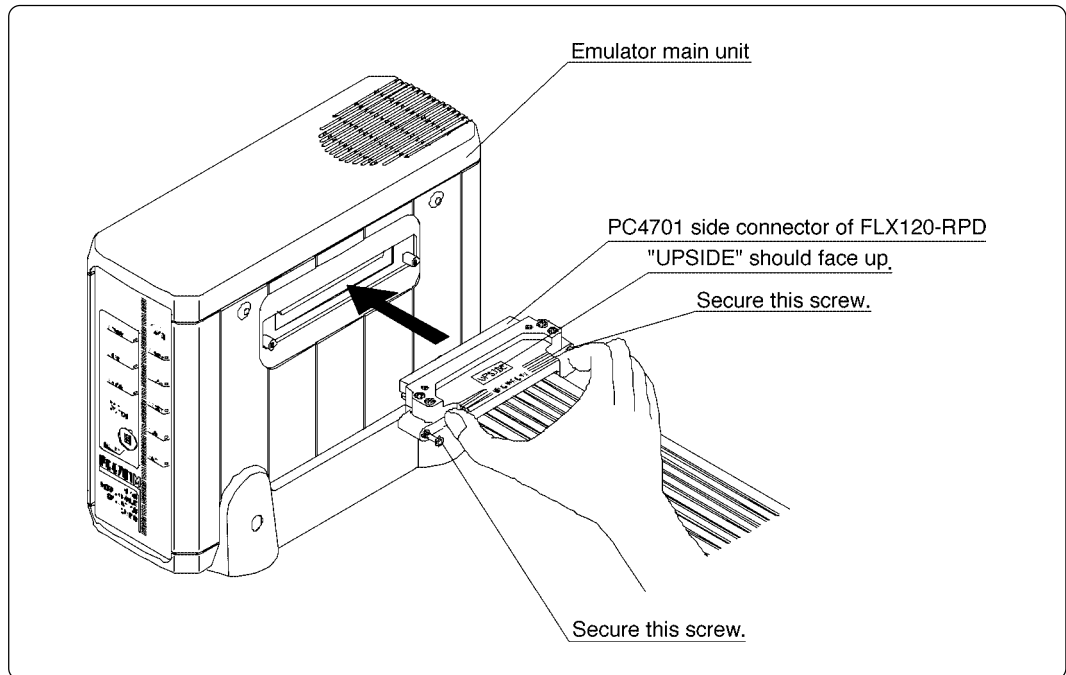
Figure 3.13 Connecting PC4701 and FLX120-RPD

⚠CAUTION

Note on Connecting the Cable:

- To connect the FLX120-RPD, be sure to hold the both sides of the PC4701 side connector horizontally with the "UPSIDE" facing up.

- Always shut OFF power before connecting the FLX120-RPD. The power ON state could destroy internal circuits.

Note on Securing the Screws:

- After connecting the FLX120-RPD to the emulator main unit PC4701, be sure to secure the screws mounted in both sides of the connector.

(2) Connecting the Cable to the Emulation Pod

Figure 3.14 shows how to connect the FLX120-RPD and the emulation pod.

Emulation pod

FLX-120 (secure the screws)
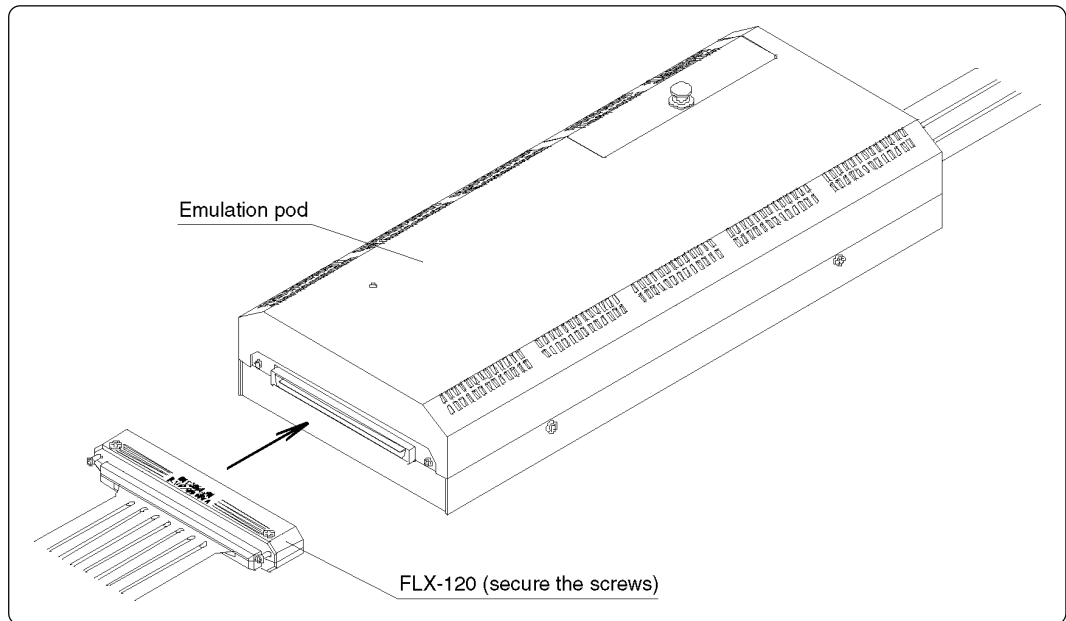
Figure 3.14 Connecting FLX120-RPD and emulation pod

⚠ CAUTION

Note on Connecting the Cable:

- Always shut OFF power before connecting the cable. The power ON state could destroy internal circuits.

Note on Securing the Screws:

- After connecting the cable to the emulation pod, be sure to secure the screws.

3.7 Connecting the Target System

Figure 3.15 shows how to connect the M306H2T-RPD-E and the target system.
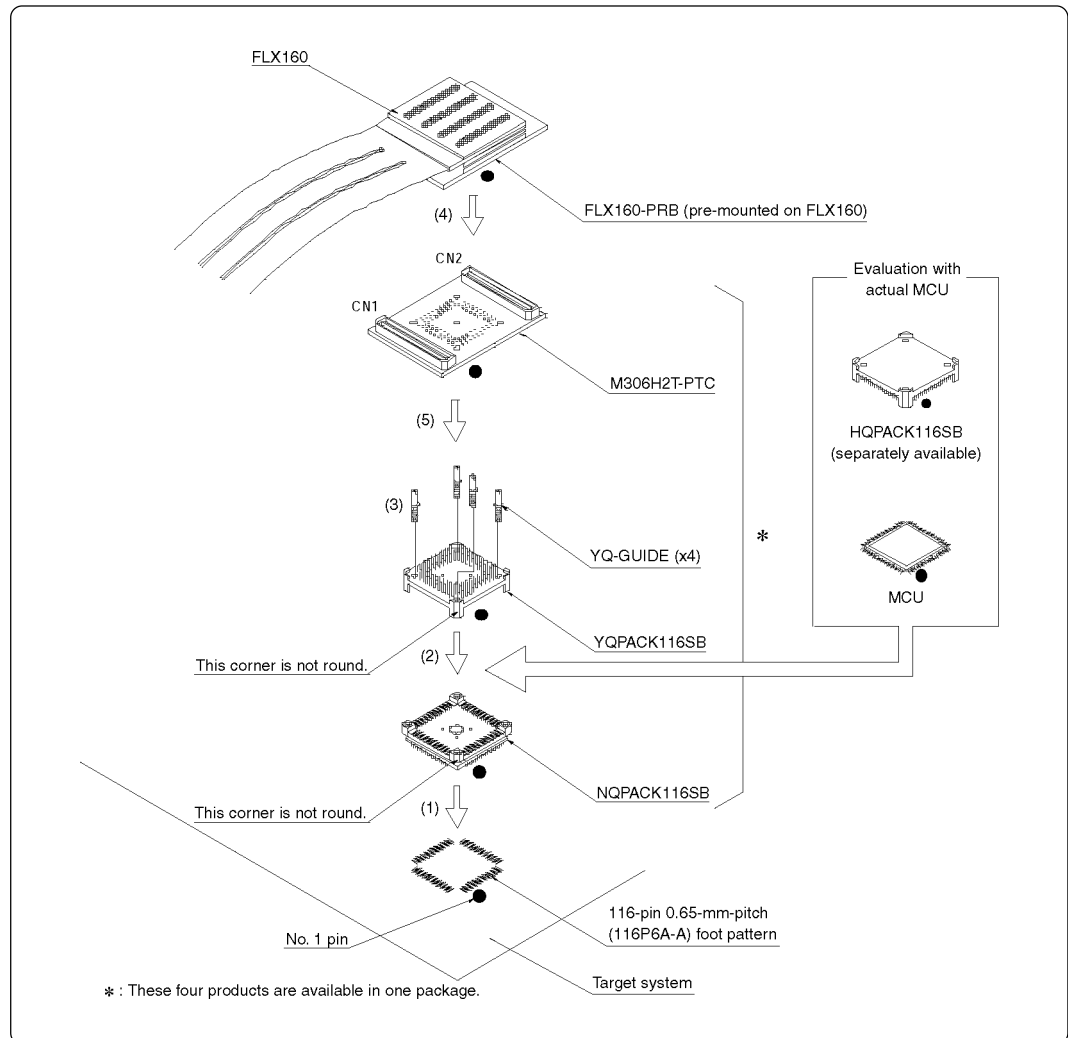
Figure 3.15 Connecting the emulation pod and the target system

---

⚠ CAUTION

Notes on Connecting the Target System:

- Take care not to attach the converter board in a wrong direction. It may cause a fatal damage to the emulation pod.
- The small connector of the M306H2T-PTC is guaranteed for only 50 insertion/removal iterations.

3.8 Making an MCU File for PD30

It is necessary to change the contents of the MCU file according to the MCU to be developed. Make the MCU file "M306H2.MCU" for M306H2 and store it in the "mcufile" folder in the directory same as emulator debugger PD30 is stored.

The contents of the MCU file (the SFR area, internal RAM area, internal ROM area, the name of the firmware file) are shown below. Change the settings according to the MCU to be debugged. Use your text editor for changing the contents of the MCU file.

The contents of the MCU file when using the M306H2MC are as follows:

0	: SFR area	Start address
3FF	:	End address
400	: Internal RAM	Start address
17FF	:	End address
E0000	: Internal ROM	Start address
FFFFF	:	End address
M30620B	: Name of firmware (Do not change.)	
0	: Expansion No.	PD30 V.4.00 or later required (Do not change.)

Chapter 4. Usage

This chapter describes from turning on the power of this product to starting up the emulator debugger.

Chapter 4. Usage

4.1 Turning On the Power

(1) Checking the Connection of the Emulator System

Before turning the power ON, check the connection of the PC4701, emulation pod, converter board and target system.

(2) Turning On the Power

Power ON/OFF the target system and the PC4701 as simultaneously as possible.

⚠ CAUTION

Notes on Power Supply:

- The emulator's V_{CC} pin is connected to the target system in order to monitor target system voltage. For this reason, the emulator cannot supply power to the target system. Therefore, provide the target system with a separate power supply from that of the emulator.

- Keep target system power supply voltage within the MCU's specified range (4.75 V - 5.25 V).

- Do not change target system power supply voltage after power has been activated.

---

(3) LED Display When PC4701 Starts Up Normally

After the emulator starts up, check the status of the LEDs on the front panel to see whether emulation pod operation is enabled or not. Figure 4.1 shows front panel LED lighting status when the emulator is turned ON.
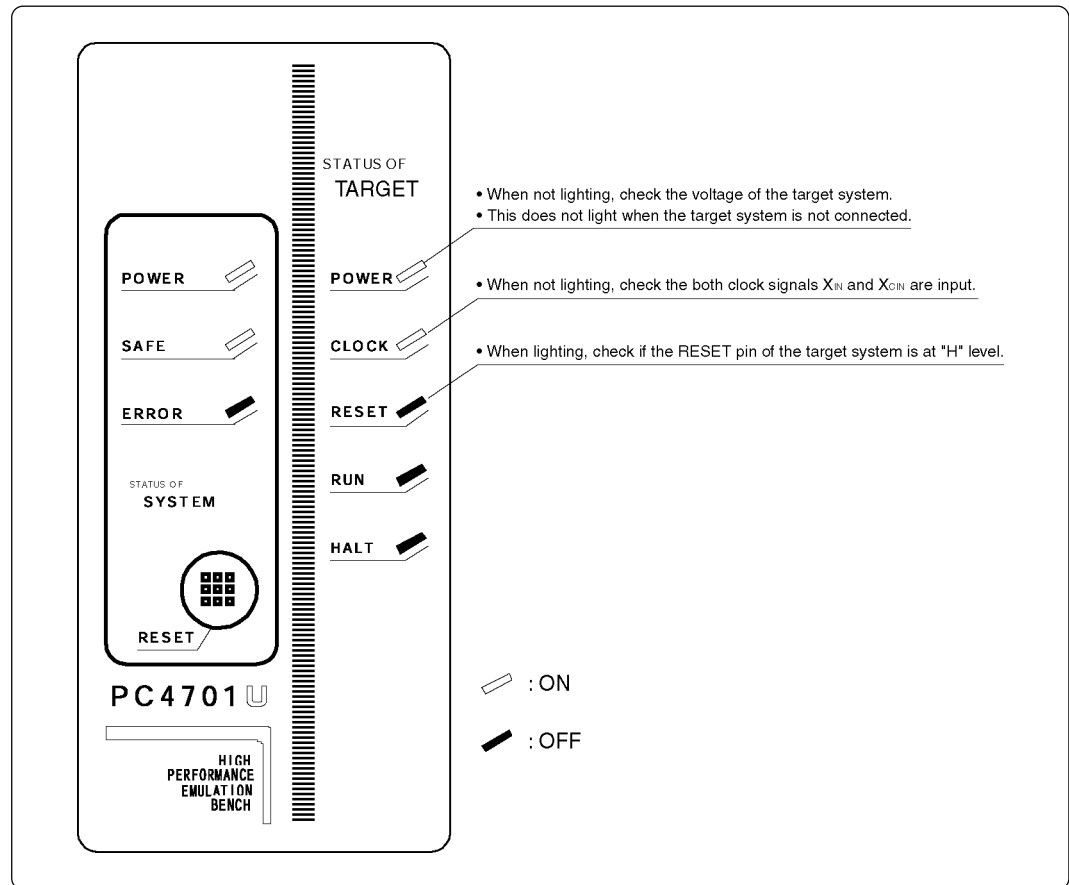
Figure 4.1 LED display when the power turned on

IMPORTANT

Note on Memory Expansion and Microprocessor Modes:

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

Note on CLOCK LED:

- If CLOCK LED does not turn on, check the following points:

  (1) Immediately after starting PC4701 (before starting the emulator debugger)
      --> Check if the oscillation circuit within the emulation pod oscillates normally.

  (2) After starting the emulator debugger (after setting the Init dialog box)
      --> Check the oscillation circuit set in the Init dialog box oscillates normally.

4.2 Downloading Firmware

(1) When It is Necessary to Download Firmware

It is necessary to download the firmware when:

(1) you use this product for the first time
(2) the firmware has been upgraded
(3) the emulator debugger has been upgraded
(4) you use this product with a PC4701 which was used with other emulation pod before

(2) Downloading Firmware in the Maintenance Mode

Download the firmware in the maintenance mode as explained here following. <u>The target system must not be connected when downloading the firmware.</u>

(1) Within 2 seconds of activating power to the PC4701, press the RESET switch on the emulator front panel. This will switch the emulator to the maintenance mode.
(2) Start up the emulator debugger. When the Init dialog box setup is complete, the dialog which urges to download the firmware will appear. Download the firmware following messages. Required time for downloading the firmware depends on the connection of the interface.

• For the serial interface .... approx. 7 minutes
• For the parallel (including LPT parallel) and USB interface .... approx. 30 seconds
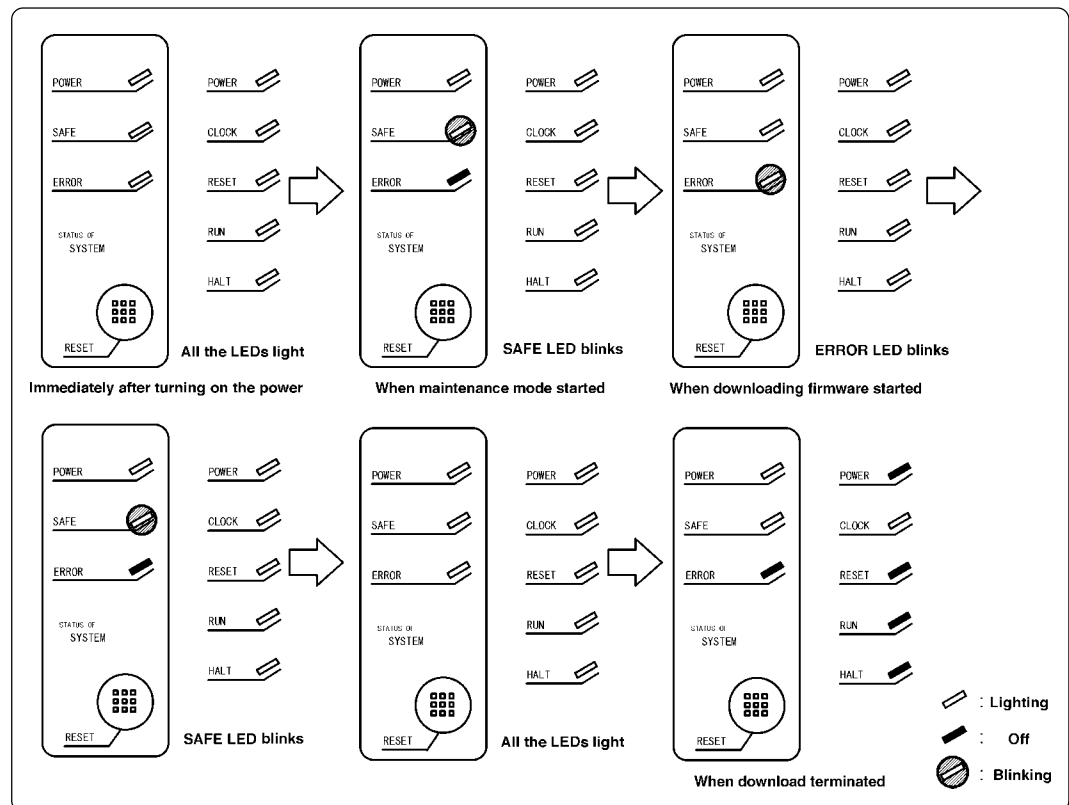

Figure 4.2 Downloading the firmware in the maintenance mode

⚠ CAUTION

Note on Downloading Firmware:

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

4.3 Self-checks

(1) Self-check Procedure

To run the emulator self-checks, do so as explained here below. While self-checks are in progress, LEDs will change as shown in Figure 4.4.

(1) Set the switches in the emulation pod same as the factory setting (see Figure 4.3).
(2) When the target system is connected, disconnect the target system.
(3) Within 2 seconds of activating power to the emulator, press the RESET switch on the emulator front panel to switch the emulator to the maintenance mode.
(4) Check the "SAFE" LED starts blinking and then press the RESET switch again.
(5) The self-checks will start. If the normal result is displayed in about 40 seconds, self-checks terminated normally.
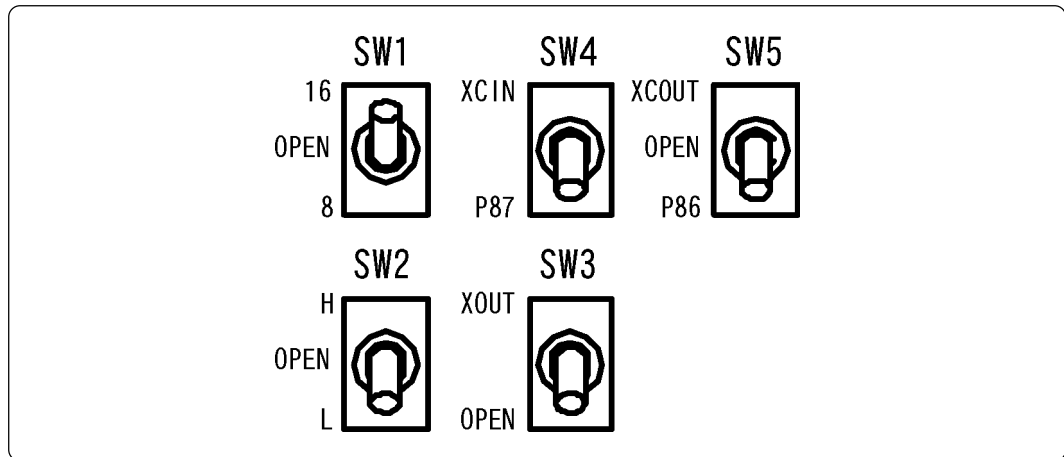
Figure 4.3 Switch settings when running self-checks

(2) If an Error is Detected in Self-checks

If self-checks do not result normally (system status error or target status error in Figure 4.4), check the following.

• Check the connection of the emulation pod and the PC4701.
• Download the proper firmware.
• Check if the switches in this product are set same as the factory-settings (Figure 4.3).

⚠ CAUTION

Note on Self-checks:

• If self-checks do not result normally (excluding target system errors), the emulation pod may be damaged. Then contact your local distributor.
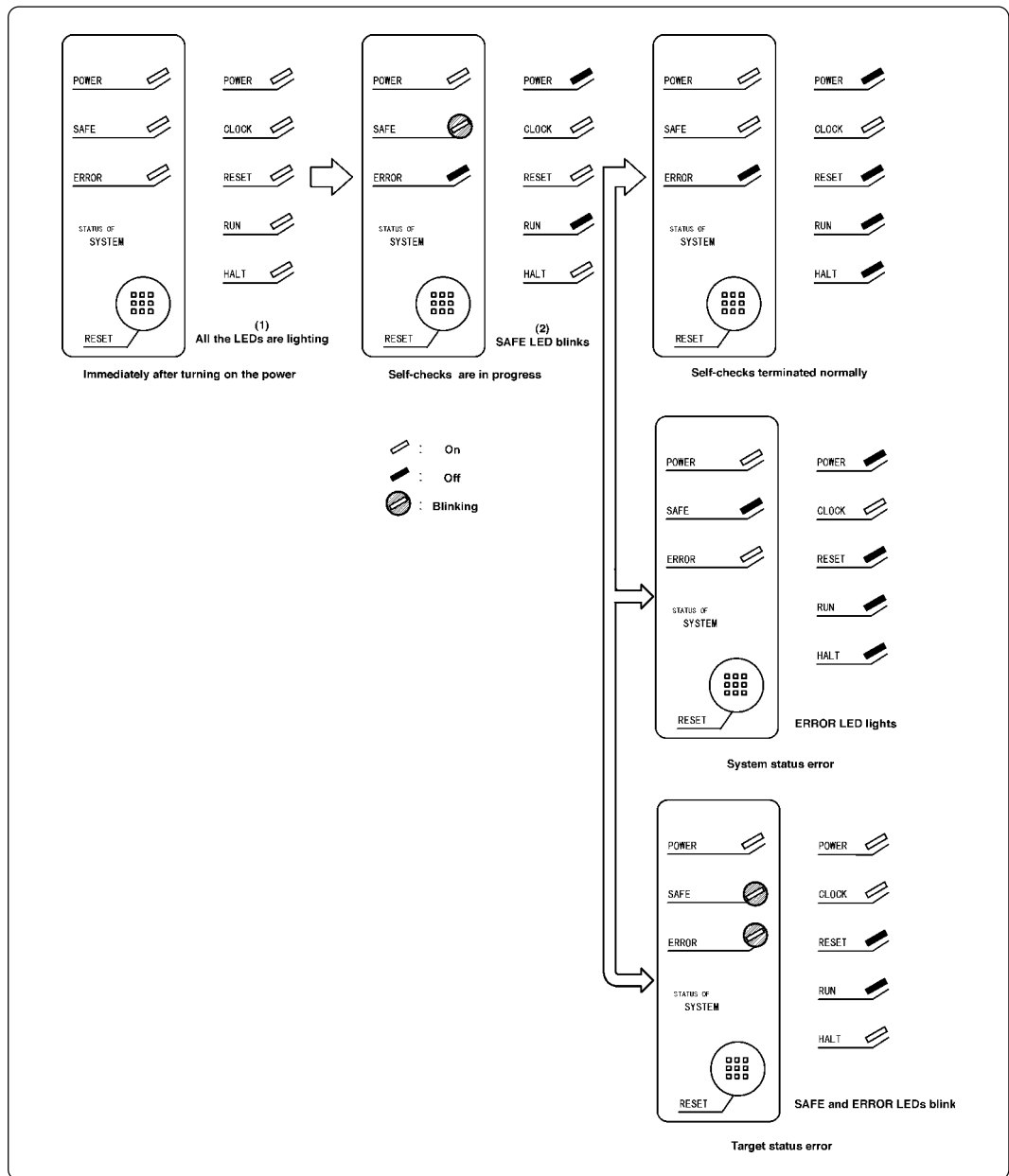
Figure 4.4 Self-check procedure

Chapter 5. Specifications

This chapter describes specifications of this product.

Chapter 5. Specifications

5.1 Specifications

Table 5.1 lists the specifications of the M306H2T-RPD-E.

Table 5.1 Specifications of the M306H2T-RPD-E

Emulators	PC4701	
Applicable MCUs	M16C/6H Group M306H2	
Evaluation MCU	M30622SAFP, M306H2MC-TOOL	
Usable modes	Single-chip mode Memory expansion mode Microprocessor mode	
Emulation memory	1 MB	
Maximum operating frequency	10 MHz	
Clock supply	X_{IN}-X_{OUT}	Internal oscillator circuit board (OSC-3) Switchable to external oscillator input.
	X_{CIN}-X_{COUT}	Internal oscillator circuit board (OSC-2) Switchable to external oscillator input.
Operating voltage	4.75 to 5.25 V	
Operating temperature	5 to 35°C (no dew)	
Storage temperature	-10 to 60°C (no dew)	
Power supply to emulation pod	Supplied from PC4701	
Connection to target system	Refer to "3.7 Connecting the Target System" (page 43).	
Overseas standards	• U.S. EMI standards (FCC part 15 Class A) • CE marking (EN55022, EN50082-1)	

5.2 Operation Timing in the Memory Expansion and Microprocessor Modes (5 V)

(1) Separate Bus, No-Wait

Table 5.2 and Figure 5.1 show the bus timing in the memory expansion and microprocessor modes (separate bus, no-wait).

Table 5.2 Memory expansion and microprocessor modes (separate bus, no-wait)

Symbol	Item	Actual MCU [ns] Min.	Actual MCU [ns] Max.	This product [ns] Min.	This product [ns] 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		See left	
Th (WR-AD)	Address output hold time (WR standard)	0		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		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)	0		See left	

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

$$\text{Td (DB-WR)} = \frac{10^9}{f(\text{BCLK})\times2} - 40 \text{ [ns]}$$

Memory expansion and microprocessor modes (no-wait)

Read timing
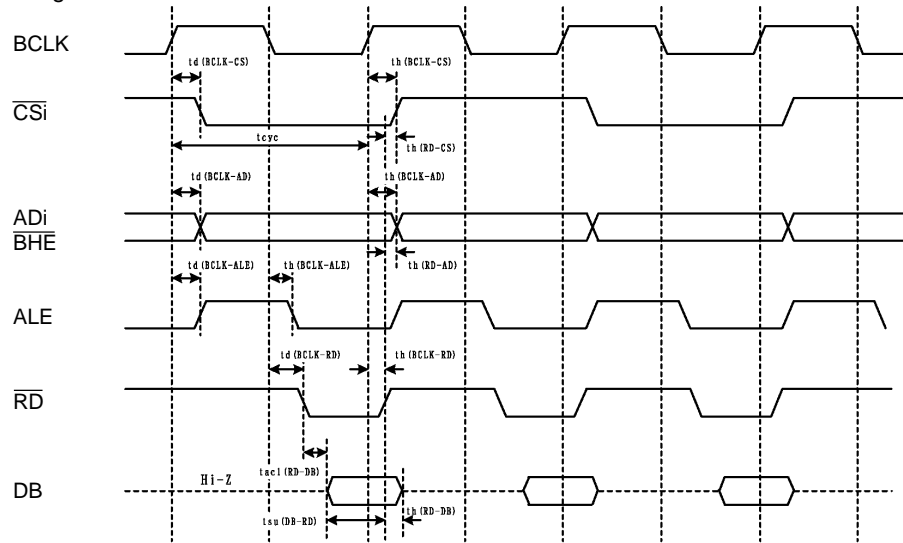
Write timing
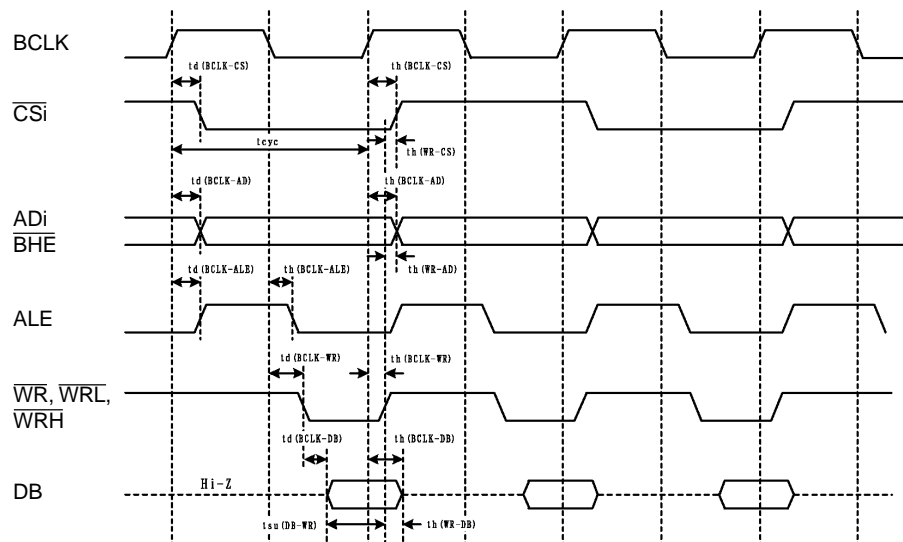
Conditions:

- $V_{CC} = 5$ V
- Input timing voltage: $V_{IL} = 0.8$ V, $V_{IH} = 2.5$ V
- Output timing voltage: $V_{OL} = 0.8$ V, $V_{OH} = 2.0$ V

Figure 5.1 Memory expansion and microprocessor modes (no-wait)

(2) Separate Bus, With Wait, Accessing External Memory Area

Table 5.3 and Figure 5.2 show the bus timing in the memory expansion and microprocessor modes (with wait, accessing external memory area).

Table 5.3 Memory expansion and microprocessor modes (with wait, external memory 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		See left	
Th (WR-AD)	Address output hold time (WR standard)	0		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		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)	0		See left	

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

$$\text{Td (DB-WR)} = \frac{10^9}{f(\text{BCLK})} - 40 \text{ [ns]}$$

Memory expansion and microprocessor modes
(with wait, accessing external memory area)

Read timing
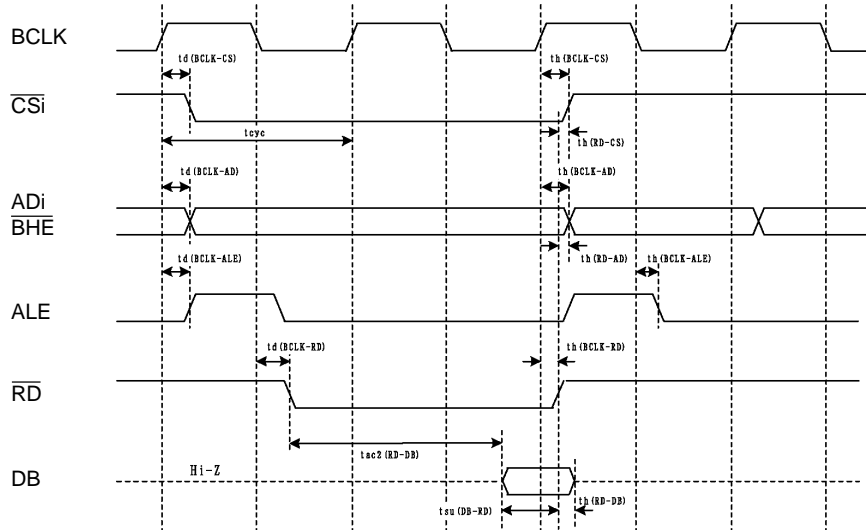
Write timing

Conditions:

• V_CC = 5 V
• Input timing voltage: V_IL = 0.8 V, V_IH = 2.5 V
• Output timing voltage: V_OL = 0.8 V, V_OH = 2.0 V

Figure 5.2 Memory expansion and microprocessor modes (with wait)

(3) Multiplex Bus, With Wait, Accessing External Memory Area

Table 5.4 and Figure 5.3 show the bus timing in the memory expansion and microprocessor modes (with wait, accessing external memory area and using multiplex bus).

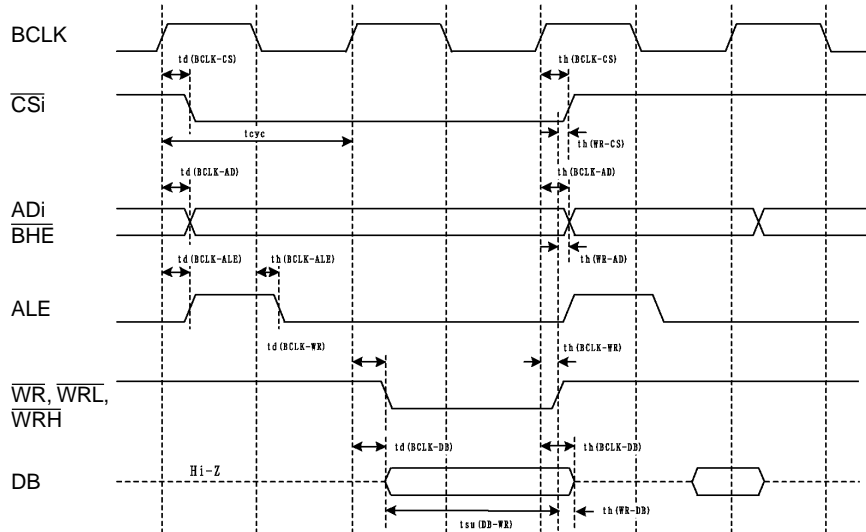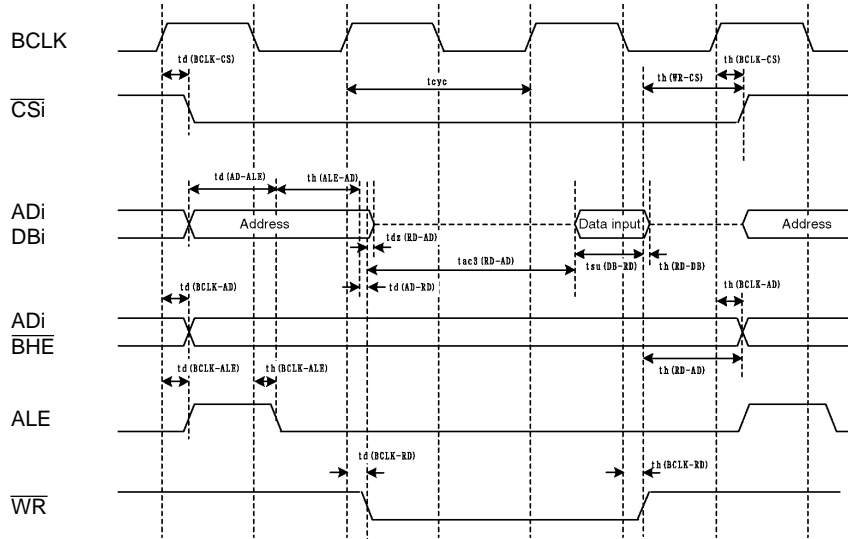*Table 5.4 Memory expansion and microprocessor modes (with wait, multiplex bus)*

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)		(*2)	
Th (WR-AD)	Address output hold time (WR standard)	(*1)		(*2)	
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)	(*1)		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)	(*1)		See left	
Th (ALE-AD)	ALE output hold time (Address standard)	50		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		See left

*1 Calculated by the following formulas according to the frequency of BCLK.	*2 Calculated by the following formulas according to the frequency of BCLK.
$Th\,(RD\text{-}AD) = \dfrac{10^9}{f(BCLK)\times 2}$ [ns]	$Th\,(RD\text{-}AD) = \dfrac{10^9}{f(BCLK)\times 2} - 3$ [ns]
$Th\,(WR\text{-}AD) = \dfrac{10^9}{f(BCLK)\times 2}$ [ns]	$Th\,(WR\text{-}AD) = \dfrac{10^9}{f(BCLK)\times 2} - 3$ [ns]
$Th\,(RD\text{-}CS) = \dfrac{10^9}{f(BCLK)\times 2}$ [ns]	
$Th\,(WR\text{-}CS) = \dfrac{10^9}{f(BCLK)\times 2}$ [ns]	
$Td\,(DB\text{-}WR) = \dfrac{10^9 \times 3}{f(BCLK)\times 2} - 40$ [ns]	
$Th\,(WR\text{-}DB) = \dfrac{10^9}{f(BCLK)\times 2}$ [ns]	
$Td\,(AD\text{-}ALE) = \dfrac{10^9}{f(BCLK)\times 2} - 25$ [ns]	

Memory expansion and microprocessor modes
(with wait, accessing external memory area and using multiplex bus)

Read timing

Write timing
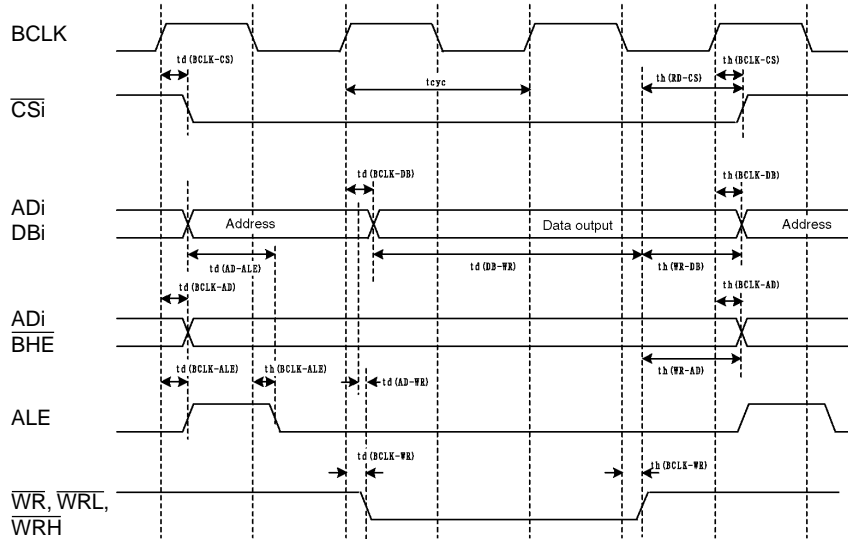
Conditions:

• V_{CC} = 5 V
• Input timing voltage: V_{IL} = 0.8 V, V_{IH} = 2.5 V
• Output timing voltage: V_{OL} = 0.8 V, V_{OH} = 2.0 V

Figure 5.3 Memory expansion and microprocessor modes (with wait, multiplex bus)

(4) Timing Requirements

Table 5.5, Figures 5.4 and 5.5 show timing requirements in the memory expansion and microprocessor modes.

Table 5.5 Timing requirements (V_{CC} = 5 V)

Symbol	Item	Actual MCU [ns]		This product [ns]	
		Min.	Max.	Min.	Max.
Tsu (DB-RD)	Data input setup time	40		See left	
Tsu (RDY-BCLK)	RDY* input setup time	30		45	
Tsu (HOLD-BCLK)	HOLD* input setup time	40		*1	
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

*1 Minimum 7 ns (The definition is different from that of actual MCUs. For details, see Figure 5.5.)
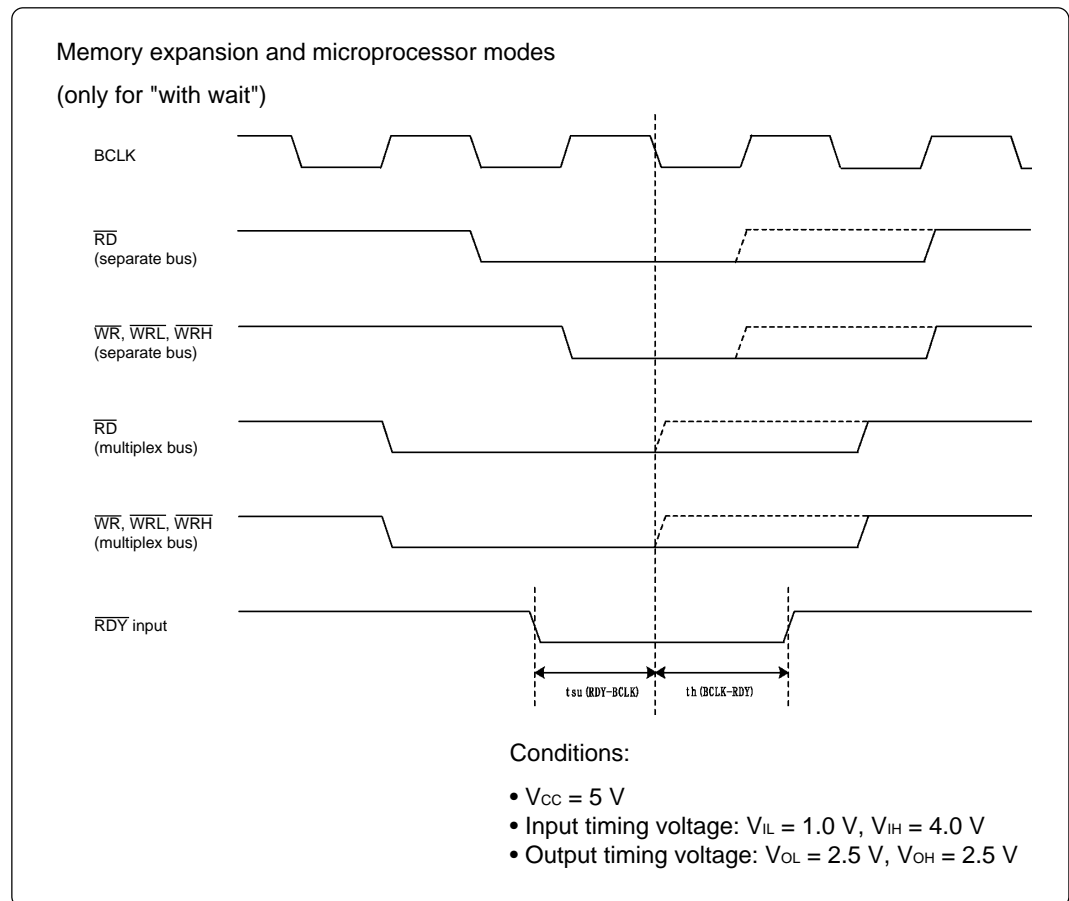
Figure 5.4 Timing requirements

Common to "with wait" and "no-wait" (actual MCU)

BCLK

HOLD input

HLDA output

P0, P1, P2,
P3, P4,
P5₀ to P5₂

$t_{su(HOLD-BCLK)}$ $t_{h(BCLK-HOLD)}$ $t_{d(BCLK-HLDA)}$ Hi-Z $t_{d(BCLK-HLDA)}$
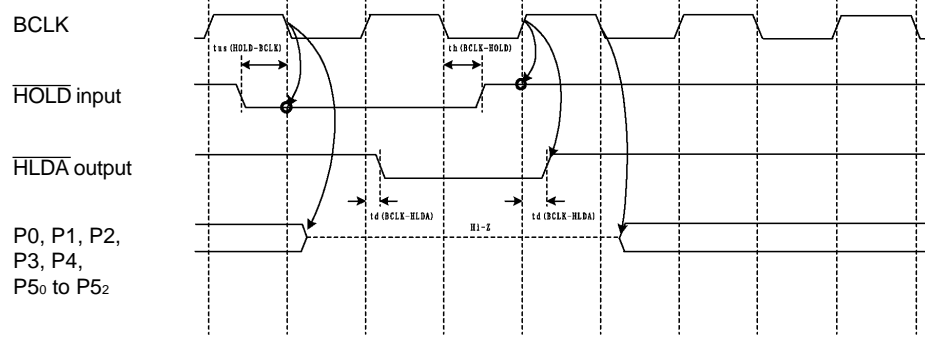
Common to "with wait" and "no-wait" (This product)

BCLK

HOLD input

HLDA output

P0, P1, P2,
P3, P4,
P5₀ to P5₂

$t_{su(HOLD-BCLK)}$ $t_{h(BCLK-HOLD)}$ $t_{d(BCLK-HLDA)}$ Hi-Z $t_{d(BCLK-HLDA)}$
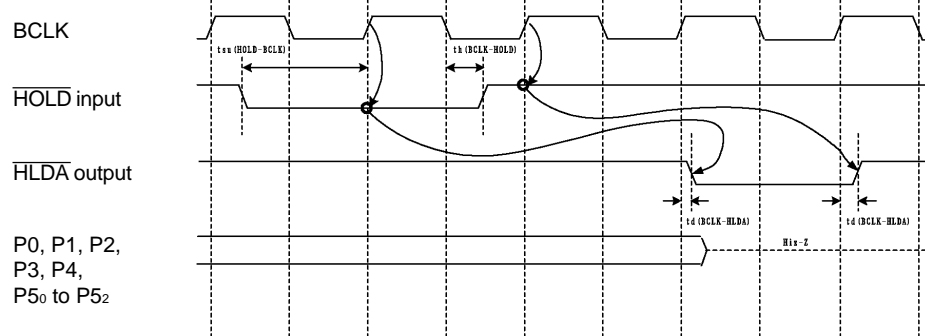
Note 1. Ports $P0_0$ to $P5_2$ will be high-impedance status regardless of the input level of BYTE pin and ports $P4_0$ to $P4_3$ function selection bit (PM06) of the processor mode register 0.

Note 2. This product will be high-impedance status delaying by 2.5 cycles than an actual MCU.

Note 3. The setup time of HOLD is defined by the startup of BCLK, differently from that of actual MCUs.

Conditions:

• V_{CC} = 5 V
• Input timing voltage: V_{IL} = 1.0 V, V_{IH} = 4.0 V
• Output timing voltage: V_{OL} = 2.5 V, V_{OH} = 2.5 V

Figure 5.5 Timing requirements

5.3 Electrical Characteristics

Table 5.6 lists IC electrical characteristics of the user interface.

Table 5.6 Electrical characteristics of the M60081L-0142FP (P0$_0$ to P5$_7$)

Symbol	Parameter	Condition	Standard values			Unit
			Min.	Standard	Max.	
V_{IL}	Input voltage	V_{CC} = 4.5 V	0		1.35	V
V_{IH}		V_{CC} = 5.5 V	3.85		5.5	
V_{OL}	Output voltage	V_{CC} = 5.0 V			0.05	V
V_{OH}		$|I_O|$ < 1 µA	4.95			V
I_{OL}	Output current	V_{CC} = 4.5 V V_{OL} = 0.4 V	8			mA
I_{OH}		V_{CC} = 4.5 V V_{OH} = 4.1 V			-8	mA
I_{IL}	Input leak current	V_{CC} = 5.5 V V_I = 0 V	-1		+1	µA
I_{IH}		V_{CC} = 5.5 V V_I = 5.5 V	-1		+1	µA
I_{OZL}	OFF state output leak current	V_{CC} = 5.5 V V_O = 0 V	-1		+1	µA
I_{OZH}		V_{CC} = 5.5 V V_O = 5.5 V	-1		+1	µA
C_{IO}	I/O pin capacity	f = 1 MHz V_{CC} = 0 V		7	15	pF

5.4 Connection Diagram

Figure 5.6 shows the connection diagram of the M306H2T-RPD-E. This connection diagram mainly shows the interface section, and the circuits which are not connected to the target system such as the emulator's control system are omitted.

*: Control signal
Each number of part indicates the part number of the M306H0T-PRT board.
However, only the numbers on the peripheral emulation MCU indicate the part numbers of the M306H2T-RPDM board.
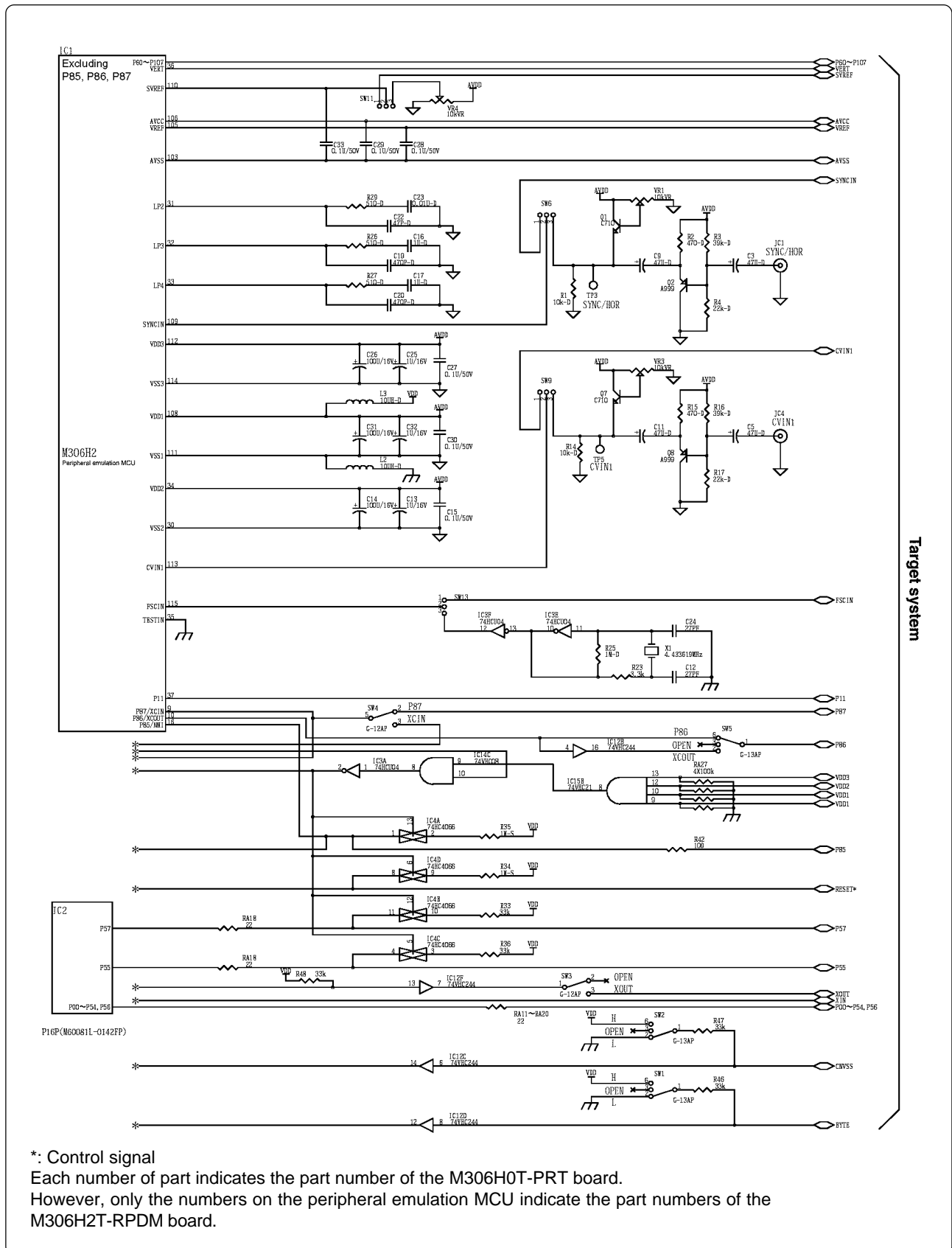
Figure 5.6 Connection diagram

5.5 External Dimensions

(1) External Dimensions of the Emulation Pod
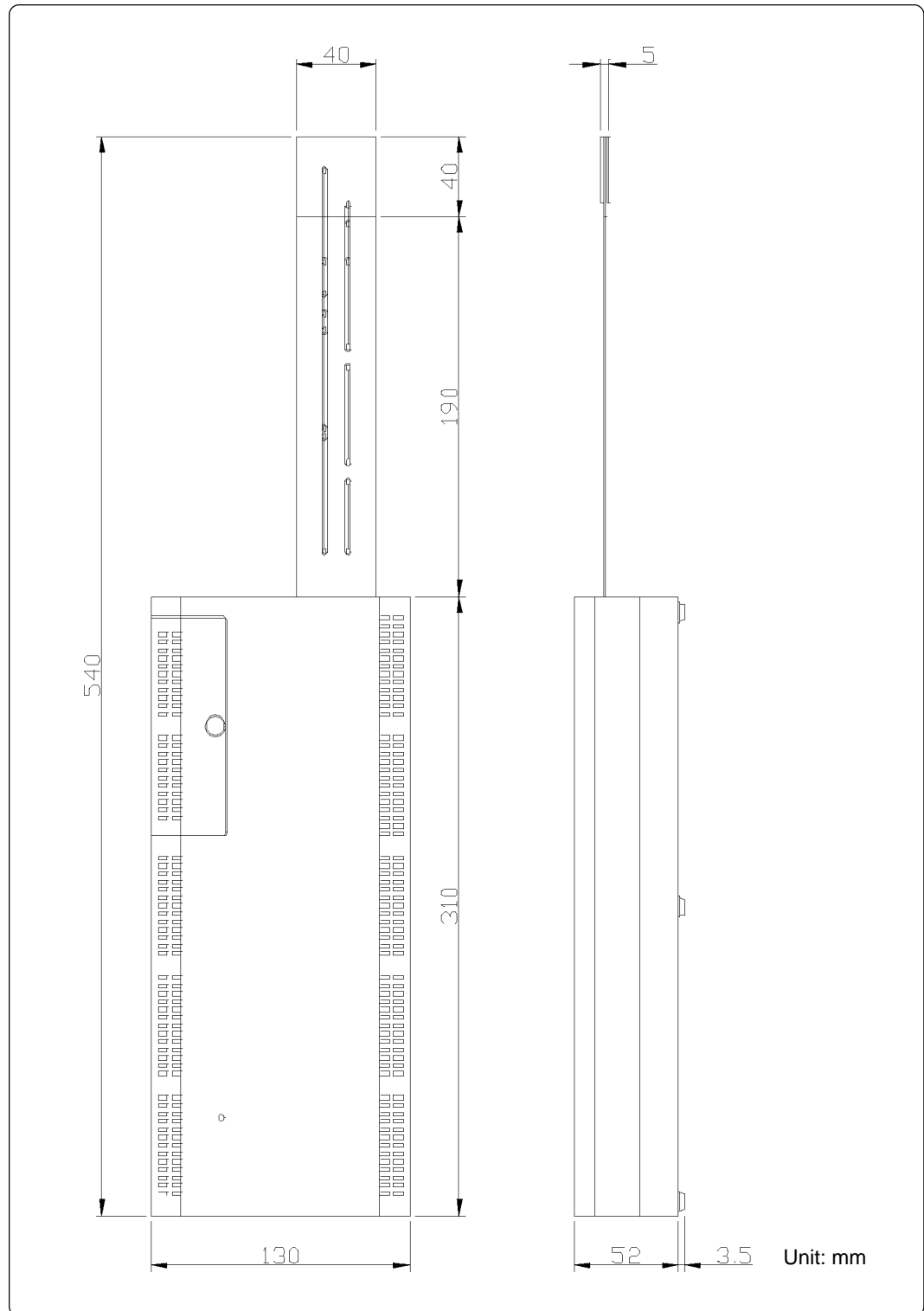
Figure 5.7 External dimensions of the emulation pod

(2) External Dimensions of the Pitch Converter Board (FLX160-PRB)

Figure 5.8 shows the external dimensions of the pitch converter board FLX160-PRB.
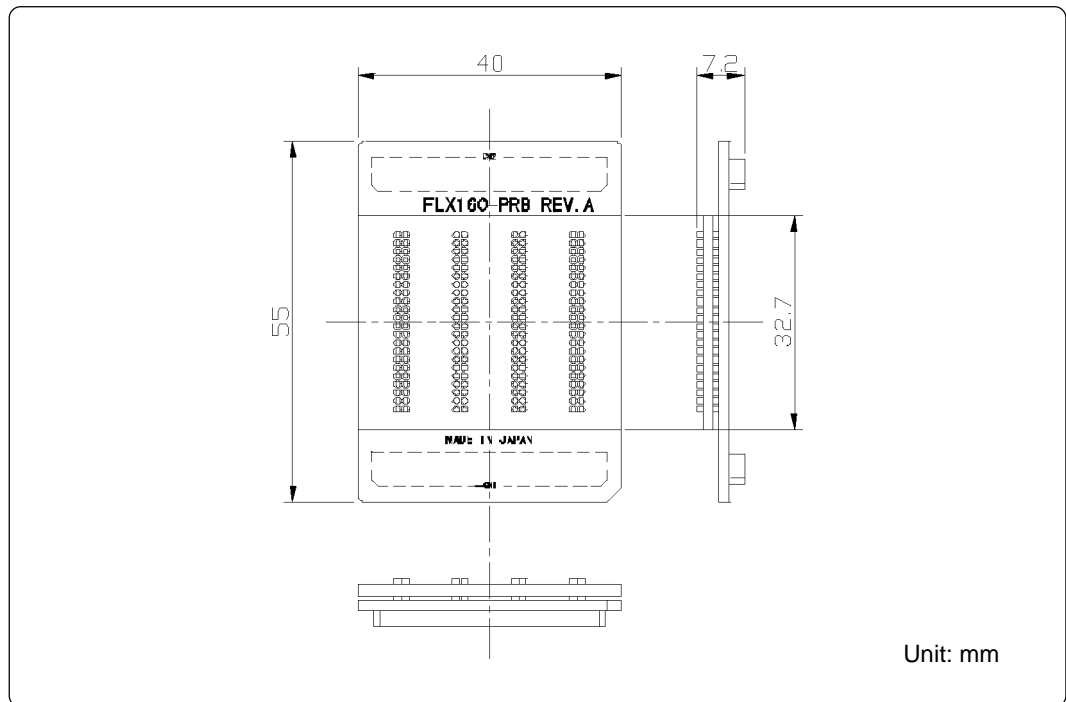
Figure 5.8 External dimensions of the pitch converter board (FLX160-PRB)

(3) External Dimensions of the Pitch Converter Board (M306H2T-PTC)

Figure 5.9 shows the external dimensions and the sample foot pattern of the pitch converter board M306H2T-PTC for 116-pin QFP (116P6A-A).
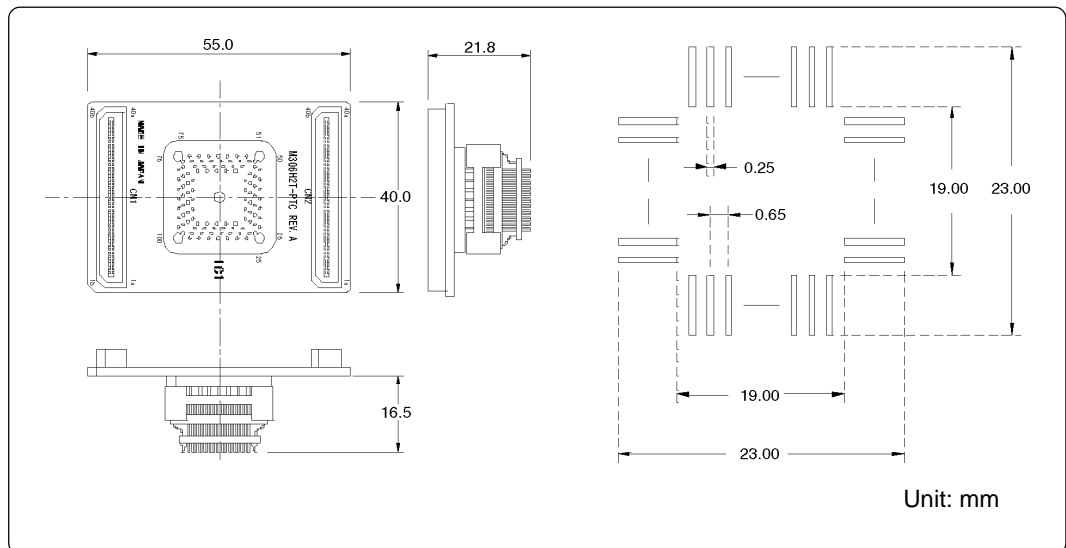
Figure 5.9 External dimensions of the pitch converter board (M306H2T-PTC)

Chapter 6. Troubleshooting

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

Chapter 6. Troubleshooting

6.1 Flowchart to Remedy the Troubles

Figure 6.1 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 target system is disconnected.

For the latest FAQs, visit the Renesas Tools Homepage.

http://www.renesas.com/en/tools
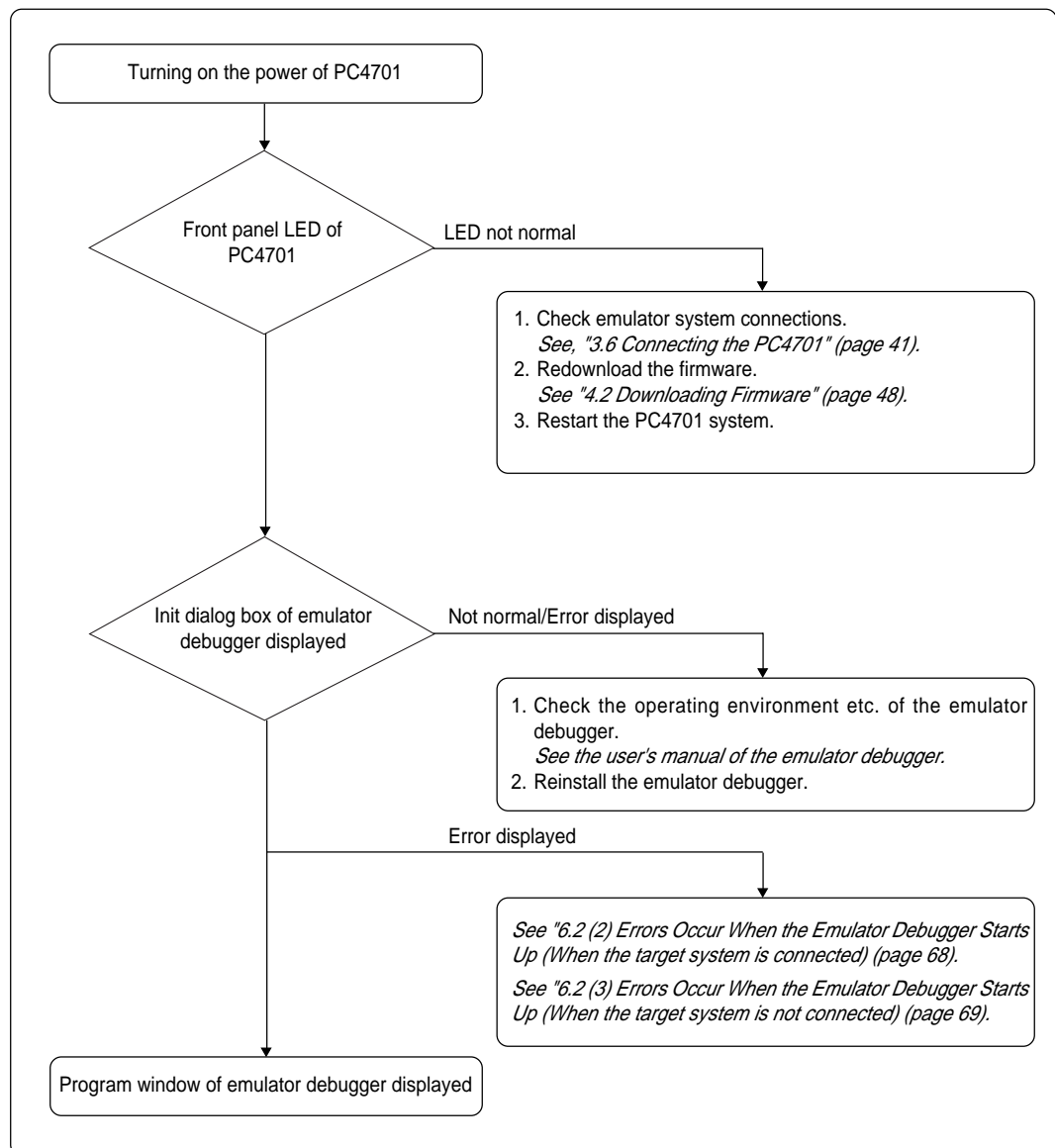
Figure 6.1 Flowchart to remedy troubles

6.2 When the Emulator Debugger Does Not Start Up Properly

(1) When the LED Display of the PC4701 is Abnormal

Table 6.1 LED's abnormal display and its checkpoints

Error	Connection to the target system	Checkpoint
LEDs do not light up.	-	Recheck the power cable is connected to the PC4701. *See the user's manual of the PC4701.*
All LEDs remain lit.	-	Recheck the connection between the PC4701 and this product. *See "3.6 Connecting the PC4701" (page 41).*
The "POWER" LED of "STATUS OF TARGET" does not light up.	Connected	Check power is properly supplied to the target system and that the target system is properly grounded.
The "CLOCK" LED of "STATUS OF TARGET" does not light up.	Disconnected	(1) Check that both the main and sub clocks are not set to "EXT" in the emulator debugger. *See the CLK command of the emulator debugger.* (2) Check the oscillation circuit in the emulation pod is oscillating. *See "3.3 Selecting Clock Supply" (page 35).*
	Connected	(1) Check the oscillation circuit in the target system is working. (2) Check the switches in the emulation pod are correctly set. *See "3.2 Setting Switches and Pullup Resistors" (page 29).*

(2) Errors Occur When the Emulator Debugger Starts Up
(When the target system is connected)

Table 6.2 Checkpoints of errors when starting up the emulator debugger (target is connected)

Error	Checkpoint
Communication error occurred Data was not sent to the target	Check all emulator debugger settings, interface cable connection and switches on the rear of the PC4701 match. *See the user's manual of the PC4701 and the user's manual (or online manual) of the emulator debugger.*
Target system cannot be properly built	(1) Download the proper firmware. *See "4.2 Downloading Firmware" (page 48).* (2) Recheck the connection between the PC4701 and this product. *See "3.6 Connecting the PC4701" (page 41).*
PD30 version is not the same version as the firmware in the target	Download the proper firmware. *See "4.2 Downloading Firmware" (page 48).*
Target MCU is in the reset state	Check the reset pin of the target system has changed from "L" to "H" level.
Target MCU cannot be reset	(1) Check the RESET* pin and NMI* pin are "H" level. (2) If in the memory expansion mode or microprocessor mode, check the RDY* pin and HOLD* pin are "H" level. (3) If the reset circuit of the target system has a watchdog timer, disable the timer. (4) Check power is properly supplied to the target system and that the target system is properly grounded. (5) The program may be uncontrollable in areas where memory is not allocated. Recheck the map setting. *See the user's manual (or online manual) of the emulator debugger.*
Target is in HOLD state	(1) The MCU is either in the stop mode or wait mode. Either reset the MCU or cancel the mode with an interrupt. *See the MCU specifications.* (2) The program may be uncontrollable in areas where memory is not allocated. Recheck the map setting. *See the user's manual (or online manual) of the emulator debugger.*
Target clock is stopped	(1) Check the oscillation circuit of the target system is working properly. (2) Check the switches in the emulation pod are correctly set. *See "3.2 Setting Switches and Pullup Resistors" (page 29).*
Target MCU is not receiving power	Check power is properly supplied to the target system and the target system is properly grounded.

(3) Errors Occur When the Emulator Debugger Starts Up
(When the target system is not connected)

Table 6.3 Checkpoints of errors when starting up the emulator debugger (target is not connected)

Error	Checkpoint
Communication error occurred Data was not sent to the target	Check all emulator debugger settings, interface cable connection and switches on the rear of the PC4701 match. *See the user's manual of the PC4701 and the user's manual (or online manual) of the emulator debugger.*

Target system cannot be properly built	(1) Download the proper firmware.
	    See "4.2 Downloading Firmware" (page 48).
	(2) Recheck the connection between the PC4701 and this product.
	    See "3.6 Connecting the PC4701" (page 41).

M3T-PD30 version is not the same version as the firmware in the target	Download the proper firmware. *See "4.2 Downloading Firmware" (page 48).*
Target MCU cannot be reset	The program may be uncontrollable in areas where memory is not allocated. Recheck the map setting. *See the user's manual (or online manual) of the emulator debugger.*

Target is in HOLD state	(1) The MCU is either in the stop mode or wait mode. Either reset the MCU or cancel the mode with an interrupt.
	    See the MCU specifications.
	(2) The program may be uncontrollable in areas where memory is not allocated. Recheck the map setting.
	    See the user's manual (or online manual) of the emulator debugger.

Target clock is stopped	Check the switches in the emulation pod are correctly set. *See "3.2 Setting Switches and Pullup Resistors" (page 29).*

6.3 Operation Differs from That of Actual MCUs

(1) Cannot Reset from the Target System

Set the time for starting up (0.2 Vcc to 0.8 Vcc) to 1 µs or less.

(2) Data Values of RAM Area at Power-on are Different

For this product, 04h is written into the RAM area at power-on. Therefore, the data values are different from those of an actual MCU.

(3) HOLD* Control

When inputting "Low" to the HOLD* pin to run into the HOLD state, $P0_0$ to $P5_2$ will be in the HOLD state delaying by 2.5 cycles than the actual MCU (see Figure 5.5 and Table 5.5).

(4) A-D Conversion Values are Different from Expected Values

As a flexible cable, a pitch converter board and other devices are used between the evaluation MCU and the target system, some characteristics are slightly different from those of the actual MCU. Therefore, be sure to evaluate your system with an evaluation MCU. Before starting mask production, evaluate your system and make final confirmation with an CS (Commercial Sample) version MCU.

(5) Outputs of ALE, Address and Others are Different from Those of Actual MCUs

This product has some output signals which are different from those of actual MCU as following.

1) ALE signal output in accessing the SFR area
2) Address, BHE* and ALE signal outputs in accessing the internal RAM
3) State of the data bus in the stop mode and wait mode

For details, see Chapter 1 "Note on Differences between Actual MCU and Emulator" (page 17).

Chapter 7. Maintenance and Guarantee

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

Chapter 7. Maintenance and Guarantee

7.1 Maintenance

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.

7.2 Guarantee

If your product becomes faulty within twelve months after its purchase while being used under good conditions by observing "Precautions for Safety" described in Chapter 1 of 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.

7.3 Repair Provisions

(1) Repair with extra-charge

The products elapsed more than twelve months 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
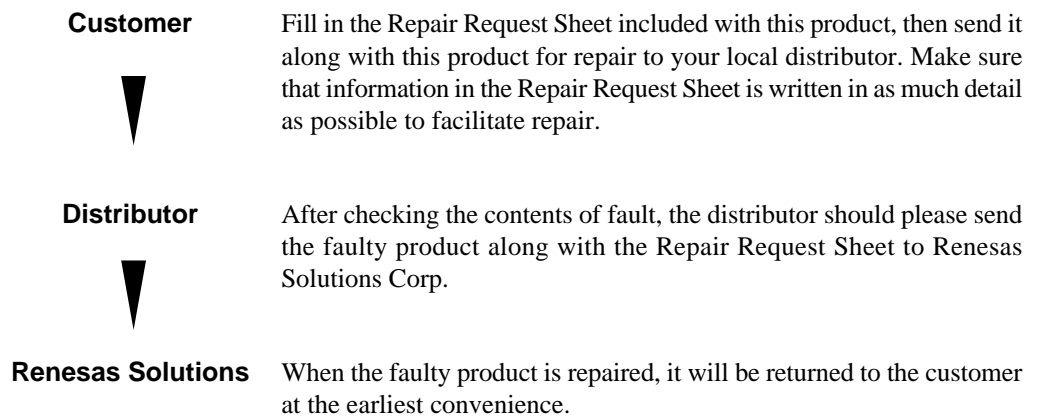
(3) Expiration of the repair period

When a period of twelve months 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.

7.4 How to Request for Repair

If your product is found faulty, follow the procedure below to send your product for repair.

Customer

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.

Distributor

After checking the contents of fault, the distributor should please send the faulty product along with the Repair Request Sheet to Renesas Solutions Corp.

Renesas Solutions

When the faulty product is repaired, it will be returned to the customer at the earliest convenience.

⚠ 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.

MEMO

M306H2T-RPD-E User's Manual

Rev.1.00
September 1, 2003
REJ10J0259-0100Z