User's Manual

E30A Emulator R0E00030AKCT00

User's Manual

Supported Devices:
R32C/100 Series

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corporation without notice. Please review the latest information published by Renesas Electronics Corporation through various means, including the Renesas Electronics Corporation website (http://www.renesas.com).

Renesas Electronics
www.renesas.com

Rev.5.02 Sep 2021

Downloaded from Arrow.com.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.

© 2021 Renesas Electronics Corporation. All rights reserved.

Downloaded from Arrow.com.

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)
   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on
   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state
   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins
   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals
   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin
   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses
   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products
   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Downloaded from Arrow.com.

Preface

The E30A Emulator (R0E00030AKCT00) is an emulator for MCU of the R32C/100 Series.

All the components of this product are shown in "1.1 Package Components". If you have any questions about this product, contact your local distributor.

This user's manual mainly describes specifications of the E30A Emulator and how to setup it. For details on the related product such as an emulator debugger, refer to each product's user's manual.

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

Related manuals

Item	Manual
Integrated development environment	High-performance Embedded Workshop online help
Emulator debugger	R32C/100 E30A Emulator Debugger online help
C compiler	NC100 C Compiler User's Manual
Assembler	AS100 Assembler User's Manual

Downloaded from Arrow.com.

Important

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

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

Emulator:

"Emulator" in this document collectively refers to the E30A Emulator provided by Renesas Electronics Corporation.

"Emulator" herein encompasses neither the customer's user system nor the host machine.

Purpose of use of the emulator:

This emulator is a device to support the development of systems that use the M16C Family R32C/100 Series of Renesas 32-bit single-chip MCUs. It provides support for system development in both software and hardware. This emulator is not guaranteed for use on production lines.

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

For those who use the 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 basic knowledge of electric circuits, logical circuits, and MCUs.

When using the emulator:

(1) This emulator is a development-support unit for use in your program development and evaluation stages. When a program you have finished developing is to be incorporated in a mass-produced product, the judgment as to whether it can be put to practical use is entirely your own responsibility, and should be based on evaluation of the device on which it is installed and other experiments.
(2) In no event shall Renesas Electronics Corporation be liable for any consequence arising from the use of this emulator.
(3) Renesas Electronics Corporation strives to provide workarounds for and correct trouble with products malfunctions, with some free and some incurring charges. However, this does not necessarily mean that Renesas Electronics Corporation guarantees the provision of a workaround or correction under any circumstances.
(4) This emulator covered by this document has been developed on the assumption that it will be used for program development and evaluation in laboratories. Therefore, it does not fall within the scope of applicability of the Electrical Appliance and Material Safety Law and protection against electromagnetic interference when used in Japan.
(5) Renesas Electronics Corporation cannot predict all possible situations and possible cases of misuse that carry a potential for danger. Therefore, the warnings in this user's manual and the warning labels attached to this emulator do not necessarily cover all such possible situations and cases. The customer is responsible for correctly and safely using this emulator.
(6) Renesas Electronics Corporation will not assume responsibility of direct or indirect damage caused by an accidental failure or malfunction in this product.

Downloaded from Arrow.com.

When disposing of this product:

Penalties may be applicable for incorrect disposal of this waste, in accordance with your national legislation.

Usage restrictions:

This emulator has been developed as a means of supporting system development by users. Therefore, do not use it as an embedded device in other equipment. Also, do not use it to develop systems or equipment for use in the following fields.

(1) Transportation and vehicular
(2) Medical (equipment that has an involvement in human life)
(3) Aerospace
(4) Nuclear power control
(5) Undersea repeaters

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 this user's manual, may be changed without prior notice.

About 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 to patent rights or any other rights belonging to Renesas or to a third party.
(3) This user's manual and this emulator are copyrighted, with all rights reserved by Renesas. This user's manual may not be copied, duplicated or reproduced, in whole or part, without prior written consent from Renesas.

About diagrams:

Some diagrams in this user's manual may differ from the objects they represent.

Downloaded from Arrow.com.

Precautions for Safety

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

Contact us if you have any questions about the precautions described here.

⚠ WARNING	WARNING indicates a potentially dangerous situation that will cause death or heavy wound unless it is avoided.
⚠ CAUTION	CAUTION indicates a potentially dangerous situation that will cause a slight injury or a medium-degree injury or property damage unless it is avoided.

To avoid a possible danger, the following diagrammatic symbols are used to call your attention.

△ 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.

Downloaded from Arrow.com.

⚠ WARNING

Warnings for AC Power Supply:



If the 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.

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

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

Warning for Modification:



Do not modify this emulator. Personal injury due to electric shock may occur if this emulator is modified. Modifying the emulator will void your warranty.

Warnings 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 un-repairable damage.

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

Warning for Use temperature:



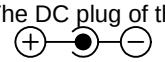
This equipment is to be used in an environment with a maximum ambient temperature of 35°C. Care should be taken that this temperature is not exceeded.

Downloaded from Arrow.com.

⚠ CAUTION

Cautions to Be Taken for the AC Adapter:

The DC plug of the AC adapter has the polarity shown below.



Use an AC adapter which complies with the safety standards of the country where it is to be used.

Cautions to Be Taken for Turning On the Power:

Turn ON/OFF the power of the emulator and user system as simultaneously as possible. If you cannot turn on the powers simultaneously, turn on the emulator first and then the user system.

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 product. 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.

When attaching and removing the cable, hold the plug of the cable and do not touch the cable. When installing the emulator, do not flex the cable excessively or pull the emulator or the board by the cable connected to it. The cable may cause a break.

Do not tape the flexible cable or apply adhesives to secure the cable. The shielding material on the surface of the cable may come off.

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) Exit the emulator debugger, and shut OFF the emulator and the user system.

(2) When turning the power ON again, wait more than 10 seconds. Then start up the emulator debugger.

Caution to Be Taken for Disposal:

Penalties may be applicable for incorrect disposal of this waste, in accordance with your national legislation.

European Union regulatory notices:



The WEEE (Waste Electrical and Electronic Equipment) regulations put responsibilities on producers for the collection and recycling or disposal of electrical and electronic waste. Return of WEEE under these regulations is applicable in the European Union only. This equipment (including all accessories) is not intended for household use. After use the equipment cannot be disposed of as household waste, and the WEEE must be treated, recycled and disposed of in an environmentally sound manner.

Renesas Electronics Europe GmbH can take back end of life equipment, register for this service at

"http://www.renesas.eu/weee"

Downloaded from Arrow.com.

Table of Contents

Page

Preface 4

Important 5

Precautions for Safety 7

User Registration 12

Terminology 13

1. Outline 14
1.1 Package Components 14
1.2 System Configuration 15
1.3 Operating Environment 15
1.4 PC Interface 15
1.5 Specifications 16
1.6 Regulatory Compliance Notices 18
1.7 Names and Functions of Each Part of the Emulator 19
1.8 Operating Environment 20

2. Setup 21
2.1 Flowchart of Starting Debugging 21
2.2 Installing the Emulator Debugger 22
2.3 Changing Hardware Settings 23
2.4 Connecting the AC Adapter to the E30A Emulator 25
2.5 Connecting the E30A Emulator and the Host Machine 25
2.6 Connecting the E30A Emulator and the User System 26
2.7 Turning On the Power 30
2.8 Installing the USB Device Driver 31
2.9 Checking the Status LEDs Display 32
2.10 Downloading Firmware 32

3. Starting Up the Emulator Debugger 33
3.1 Switching the Session 33
3.2 Initial Setting of the Emulator Debugger 34
3.3 Resetting the User System 40
3.4 Entering ID Code 41
3.5 Setting the Emulator Debugger Operating Environment 42

4. Debugging Functions 46
4.1 Break Function 46
4.2 Trace Function 48
4.3 Time Measument Function 49
4.4 RAM Monitor Function 50

5. Troubleshooting 51
5.1 How to Solve a Trouble 51
5.2 How to Request Support 52

6. Maintenance and Warranty 53

Downloaded from Arrow.com.

6.1 User Registration 53
6.2 Maintenance....53
6.3 Warranty....53
6.4 Repair Provisions....54
6.5 How to Make Request for Repair....54

Appendix A Target MCU Specifications 55

Appendix B Differences between the Actual MCU and Emulator 56

Appendix C Notes on Using This Product 58

Appendix D Connection Diagram 60

Appendix E Signal Assignments of the Connectors 61

Appendix F External Dimensions 62

Downloaded from Arrow.com.

User Registration

When you install debugger software, a text file for user registration is created on your PC. Fill it in and email it to your local distributor. If you have replaced an emulator main unit or emulation probe, rewrite an emulator name and serial number in the text file you filled in earlier to register your new hardware products.

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 contact your local distributor.

Downloaded from Arrow.com.

Terminology

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

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 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. In addition, it supports version management.

Emulator debugger

This means a software tool that is started up from the High-performance Embedded Workshop, and controls the emulator and enables debugging.

Firmware

This means a control program stored in the emulator. This analyzes the contents of communications with the emulator debugger and controls the emulator hardware. To upgrade the firmware, download the program from the emulator debugger.

Host machine

This means a personal computer used to control the emulator.

Target MCU

This means the MCU to be debugged.

User system

This means a user's application system in which the MCU to be debugged is used.

User program

This means the program to be debugged.

#

This symbol indicates that a signal is active-low (e.g. RESET#).

Downloaded from Arrow.com.

1. Outline

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

1.1 Package Components

The R0E00030AKCT00 package consists of the following items. After you have unpacked the box, check if your R0E00030AKCT00 contains all of these items. Table 1.1 lists the package components.

Table 1.1 Package Components

Item	Quantity
E30A emulator main unit (R0E00030AKCT00)	1
USB I/F cable (A plug -- mini-B plug, 1.8 m long, compliant with high-speed standard)	1
Target I/F cable (50Ω coaxial cable (gray), 1.8 m long)	1
Coaxial connector (HRM-300-126B(40) (Hirose Electric))	1
Coaxial converter board (R0E00030ACKZ00, coaxial connector and 3-pin connector (premounted))	1
Connector for coaxial converter board (PIC-FS03-G+R (Honda Tsushin Kogyo), 2.54mm pitch 3-pin socket)	1
Option I/F cable (10-pin flat cable, 20cm long)	1
Option board (R0E00030ACKZ10, 10-pin connector and 3-pin socket (premounted))	1
Option board connector • FFC-8LAMEP1B (Honda Tsushin Kogyo), 2.54mm pitch 8-pin right angle • PIC-MSA08-2G (Honda Tsushin Kogyo), 2.54mm pitch 8-pin straight • PIC-FS08-G+R (Honda Tsushin Kogyo), 2.54mm pitch 8-pin socket	 1 1 1
CD-ROM • Integrated development environment High-performance Embedded Workshop • R32C/100 E30A Emulator Debugger • E30A Emulator user's manual	1

Note: Please keep the R0E00030AKCT00's packing box and cushioning materials at hand for later reuse in sending the product for repairs or for other purposes. Always use the original packing box and cushioning material when transporting the R0E00030AKCT00.

Downloaded from Arrow.com.

1.2 System Configuration

The E30A emulator is used connecting the target MCU that mounted on the user system. Figure 1.1 shows a configuration of the E30A emulator.

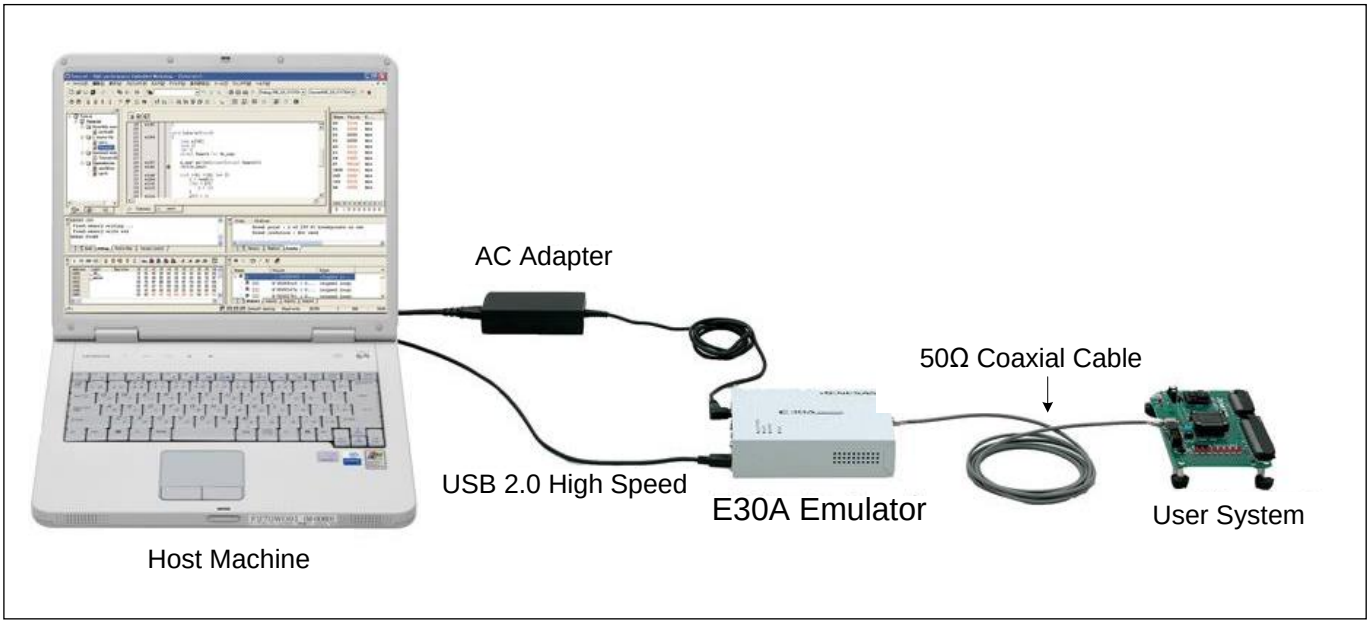


Figure 1.1 E30A Emulator System Configuration

(1) E30A Emulator (this product)
(2) AC adapter
(3) USB 2.0 high-speed cable (included)
(4) 50-Ω coaxial cable (included)
(5) User system:
This is your application system using the MCU to be debugged. The emulator cannot supply power to the user system. Get a power supply separately.
(6) Host machine:
This is a personal computer to control the emulator by USB interface.

1.3 Operating Environment

- IBM PC/AT compatibles (Windows® 7 64-bit editions *1, Windows® 7 32-bit editions *1, Windows Vista® 32-bit editions *1 *2, Windows® XP 32-bit editions *1 *2, Windows® 2000 *1)

Notes: 1. Windows and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
2. The 64-bit editions of Windows Vista® and the 64-bit editions of Windows® XP are not supported.

1.4 PC Interface

- USB interface *1
USB 2.0 high-speed (also connectable to the USB 1.1-compatible host machine)

Note: Operation with all combinations of host machine, USB device and USB hub is not guaranteed for the USB interface.

Downloaded from Arrow.com.

1.5 Specifications

1.5.1 Product specifications

Table 1.2 lists the product specifications of the E30A emulator.

Table 1.2 Product Specifications of the E30A Emulator

Item	Description
Applicable MCU	M16C Family R32C/100 Series
Available MCU mode	Single-chip mode Memory expansion mode
Maximum operating frequency	100MHz (CPU clock)
Emulation memory	Not available (internal flash ROM used as emulation memory)
Software break	Maximum 256 points (implemented by instruction replacement)
Execution address break	Maximum 8 points (execution address/data access)
Hardware break	Shared with trace event/time measurement event Only address and R/W attribute specifiable for data access break One data comparison breakpoint specifiable in software
Combinatorial break point	OR/status transition (in forward direction)
Trace mode	Trace priority/MCU execution priority specifiable
Trace capacity	8-M events (Trace priority)/512 events (MCU execution priority)
Trace contents	Branch source and branch destination/conditional branch/data access
Trace event	Maximum 8 points/4 areas (prefetch address/data access) Shared with break event/time measurement event Start/end/extraction event specifiable
Time measurement	Execution time (between Go-Stop, 32-bit CPU clock counter in MCU) Section time, 3 points (between data access, 40-bit counter in E30A)
RAM monitor	Specifiable in byte or word units • Pseudo-RAM monitor realized by software (one event used)
Coverage	Not available

Simultaneously usable functions by operation mode	Break				Trace	Time Measurement	RAM Monitor
	Execution Address	Data Access	Address Range	Data Comparison			
Trace: Trace priority	Yes	Yes	-	-	Yes	-	-
Trace: MCU execution priority	Yes	Yes	-	Yes	Yes	-	-
Time Measurement	Yes	Yes	-	-	-	Yes	-
RAM Monitor	Yes	Yes	Yes	-	-	-	Yes
Writer	-	-	-	-	-	-	-

Host machine I/F	USB 2.0 (High Speed)
Target I/F	2-pin (50-Ω coaxial cable) 3-pin (via coaxial converter board) 8-pin (via option board)
Target I/F signal	NSD, GND (When using coaxial cable/coaxial converter board) NSD, GND, Vcc1, Vcc2, RESET# (When using option board)
Targetless debugging	Not available
Emulator power supply	Supplied from the AC adapter

Downloaded from Arrow.com.

Target MCU power supply	Supplied from a user system
Outside dimension	Width: 136.3 mm, depth: 104.0 mm, height: 36.8 mm (except the protruding part)
Weight	Approx. 300 g
Operating environment	Temperature: 5°C to 35°C (no condensation)
Storage environment	Temperature: -10°C to 60°C (no condensation)
Overseas standard	European EN Standard (EN55032 Class A, EN55024) American FCC Standard (FCC part 15 Class A)

1.5.2 Specifications of the AC Adapter

The user must prepare an AC adapter and a power cable. Use an AC adapter which complies with the safety standards of the country where it is to be used. Table 1.3 lists the recommended specifications of the AC adapter.

Table 1.3 Recommended Specifications of the AC Adapter

Item	Description
AC input voltage range	AC 100 to 240 V, 50/60Hz single phase
Output power	36 W
DC output voltage, current	12.0 V, 3.0 A
DC output polarity	EIAJ TYPE IV, inner-side positive/outer-side negative

Downloaded from Arrow.com.

1.6 Regulatory Compliance Notices

1.6.1 European Union regulatory notices

This product complies with the following EU Directives. (These directives are only valid in the European Union.)

CE Certifications:

- Electromagnetic Compatibility (EMC) Directive 2014/30/EU

  EN 55032 Class A

WARNING:	This is a Class A product. This equipment can cause radio frequency noise when used in the residential area. In such cases, the user/operator of the equipment may be required to take appropriate countermeasures under his responsibility.

  EN 55024

- Information for traceability
  - Authorized representative

    Name: Renesas Electronics Corporation

    Address: TOYOSU FORESIA, 3-2-24, Toyosu, Koto-ku, Tokyo 135-0061, Japan
  - Person responsible for placing on the market

    Name: Renesas Electronics Europe GmbH

    Address: Arcadiastrasse 10, 40472 Dusseldorf, Germany
  - Trademark and Type name

    Trademark: Renesas

    Product name: E30A Emulator

    Type name: R0E00030AKCT00

Environmental Compliance and Certifications:

- Waste Electrical and Electronic Equipment (WEEE) Directive 2012/19/EU

1.6.2 United States Regulatory notices on Electromagnetic compatibility

This product complies with the following EMC regulation. (This is only valid in the United States.)

FCC Certifications:

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.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

CAUTION:	Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

Downloaded from Arrow.com.

1.7 Names and Functions of Each Part of the Emulator

Figure 1.2 shows the names and functions of each part of the emulator. The connector followed by "*" (the differential connector) is prepared for future expansion. It cannot be used because of not working now.

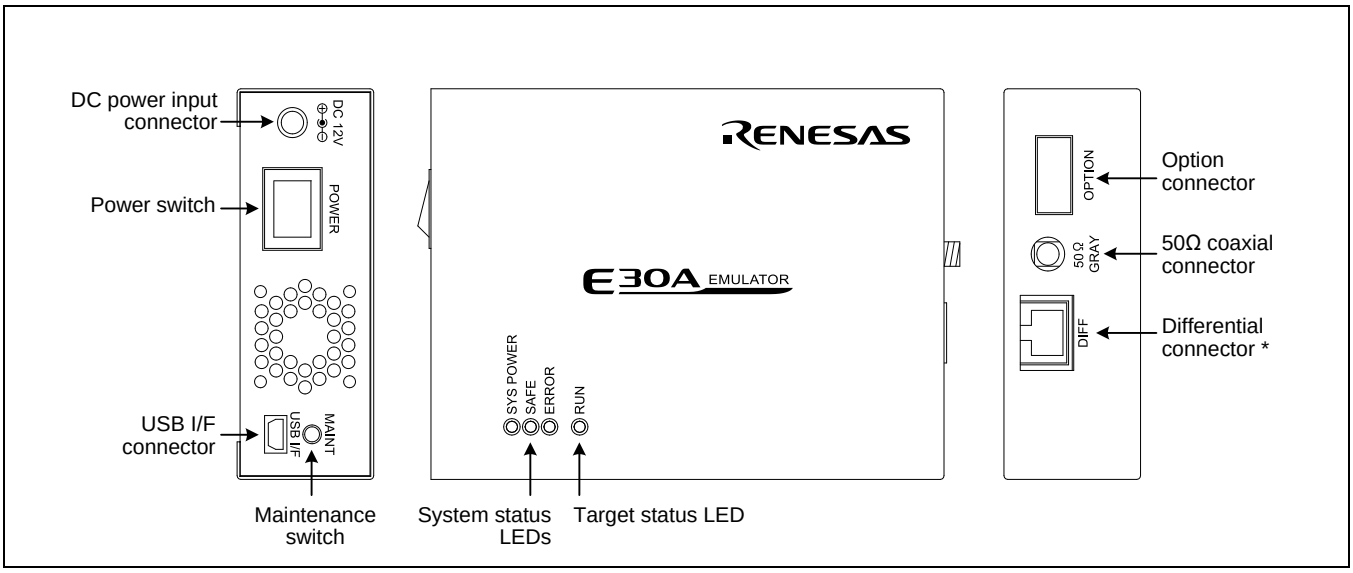


Figure 1.2 Names and Functions of Each Part of the Emulator

1.7.1 System Status LEDs

The system status LEDs indicate the emulator main unit's operating status. Tables 1.4 and 1.5 list the definition of the system status LEDs.

Table 1.4 Definitions of the System Status LEDs (SYS POWER)

Name/Status		Meaning
SYS POWER (Orange)	ON	Power is supplied to the emulator.
	OFF	Power is not supplied to the emulator.

Table 1.5 Definitions of the System Status LEDs (SAFE and ERROR)

Name/Status		Meaning
SAFE (Green)	ERROR (Red)	
ON	OFF	Emulator system has started normally.
ON	Blinking	Firmware is being downloaded.
Blinking	OFF	Emulator is in maintenance mode.
OFF	ON	Emulator system is not operating normally.
OFF	Blinking	Firmware download is not operating normally.

Downloaded from Arrow.com.

1.7.2 Target Status LED

The target status LED indicates the target MCU's operating status. Table 1.6 lists the definition of the target status LED.

Table 1.6 Definitions of the Target Status LED

Name/Status		Meaning
RUN (Green)	ON	User program is being executed.
	OFF	User program is not being executed.

1.7.3 DC Power Input Connector

This connector connects the AC adapter DC output cable to supply DC 12.0 V.

1.7.4 Power Switch

This is a power switch of the E30A emulator. Before turning on the power again after shutting off the power, wait about 10 seconds.

1.7.5 USB I/F Connector

This is a USB interface connector for connecting the host machine to this emulator. For details on connecting the emulator and host machine, refer to "2.5 Connecting the E30A Emulator and the Host Machine".

1.7.6 Maintenance Switch

If this switch is pressed within 2 seconds after turning on the power, the SAFE LED blinks and the emulator switches to maintenance mode for downloading firmware. In maintenance mode you can download firmware forcibly. For details, refer to "2.10 Downloading Firmware".

1.7.7 50-Ω Coaxial Connector

This included 50Ω coaxial cable (gray) connects this emulator to the user system. For details on connecting the emulator and user system, refer to "2.6 Connecting the E30A Emulator and the User System".

1.7.8 Option Connector

This connector is provided for connecting to the user system via an option board using the 10-pin flat cable included with the product. For details on connecting to the user system, refer to "2.6 Connecting the E30A Emulator and the User System".

1.8 Operating Environment

Be sure to use this emulator with the operating environment of the emulator and host machine listed in Table 1.7.

Table 1.7 Operating Environmental Conditions

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

Downloaded from Arrow.com.

2. Setup

2.1 Flowchart of Starting Debugging

The procedure for starting the debugging is shown in Figure 2.1. For details, refer to each section hereafter.

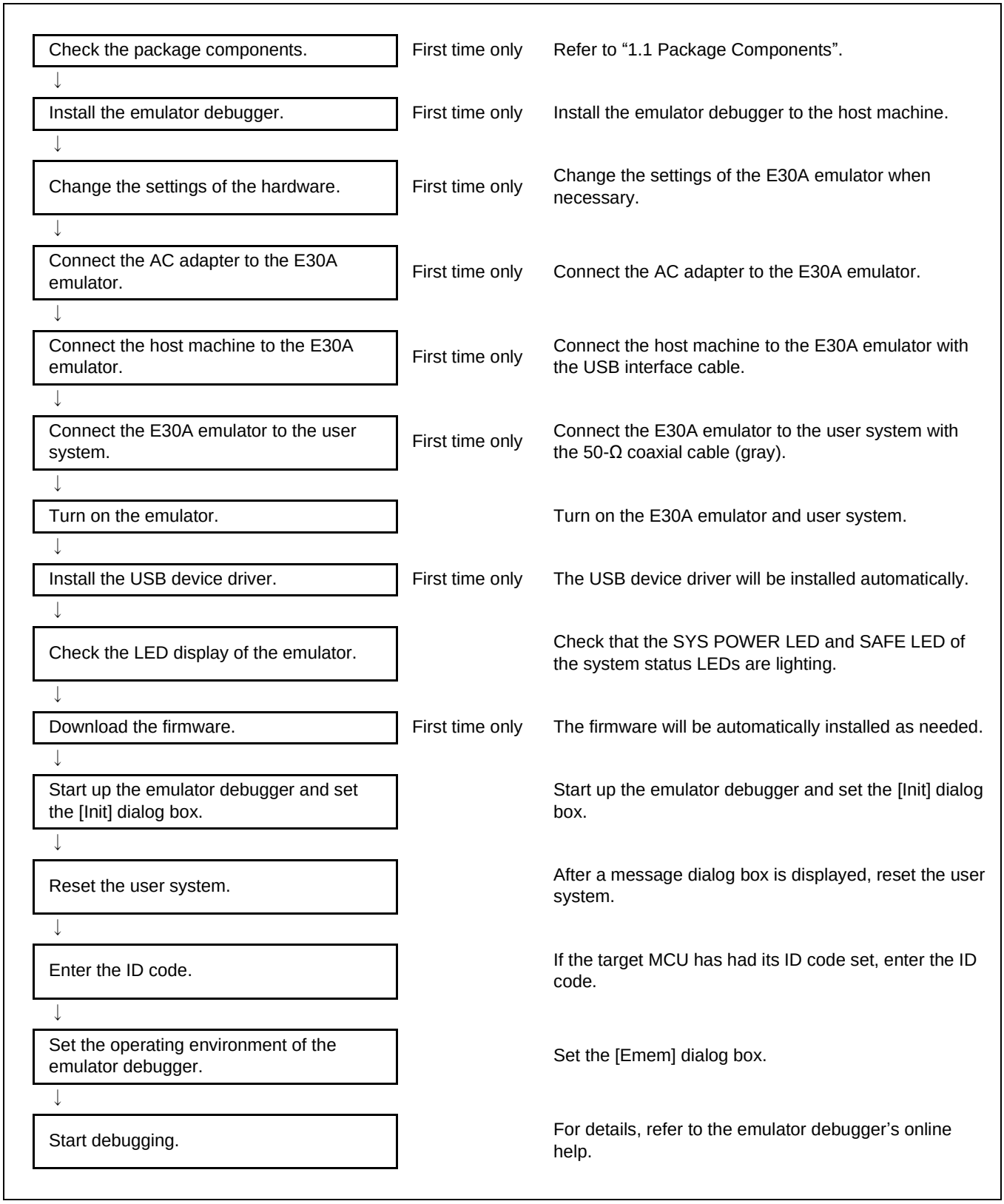


Figure 2.1 Flowchart of Starting the Debug

Downloaded from Arrow.com.

2.2 Installing the Emulator Debugger

Install the R32C/100 E30A emulator debugger from the included CD-ROM as follows. This installation must be executed by a user with administrator rights. Be aware that users without administrator rights cannot complete the installation.

(1) When inserting the included CD-ROM, the Readme Web page will be displayed. Click [Install] to open the folder where the Installation program is stored. Execute "R32cE30ADebuggerVxxxRxx.exe".

(2) The install manager starts. Click [Standard Install (Recommended)].

(3) In the page to select an installation product, click [Install] to start installing.

(4) When the installation finishes, a dialog box will be displayed indicating that the setup has been completed.

Downloaded from Arrow.com.

2.3 Changing Hardware Settings

2.3.1 Selecting Clock Supply

You need to choose the X_{IN} clock supplied to the target MCU on the user system. Table 2.1 shows the clock supply to the MCU.

For normal use, select [Generated] in the [Emulator] tab of the [Init] dialog box of the emulator debugger and specify a frequency you like to use for this clock supplied to an MCU.

You can change a frequency between 2.0 and 32.0 MHz by 0.1 MHz.

Table 2.1 Clock Supply to the MCU

Clock	Emulator Debugger Display	Description
X_{IN}-X_{OUT}	Generated	Internal generator circuit (2.0 to 32.0 MHz)
	Internal	Oscillator circuit board (OSC-3 or OSC-2)

2.3.2 Using the Oscillator Circuit Board

Even when you have selected [Internal] on the [Emulator] tab of the emulator debugger's [Init] dialog box to specify an oscillator circuit board, you need to enter a clock frequency.

With this emulator, no oscillator circuit board has been mounted when shipped from the factory. If you use a special frequency, attach the oscillator circuit board following the procedure described below.

(1) Unscrew the five screws securing the E30A emulator, and remove the upper cover.
(2) The connector J1 for connecting an oscillator circuit board is in the corner of the board (around the DC power input connector).
(3) Unscrew the screw which is used for securing an oscillator circuit board.
(4) Attach the oscillator circuit board you like to use, and secure the oscillator circuit board with the screw.
(5) Attach the upper cover and secure it with the five screws.

⚠ CAUTION

Note on Replacing the Oscillator Circuit Board:

When removing the upper cover or replacing the oscillator circuit boards, be sure to shut OFF the power supply. Otherwise the internal circuit may cause a break.

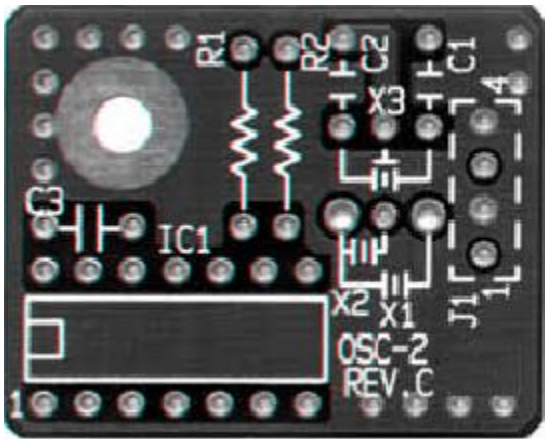
Note on Using the Oscillator Circuit Board:

This product was inspected for the qualification of CE marking under factory shipment conditions (with no oscillator circuit boards mounted in place). When using an oscillator circuit board for this product, please be aware that depending on the oscillation frequency of the oscillator circuit board used, radiation EMI standards (EN55032 Class A and FCC part 15 Class A) may be exceeded.

Downloaded from Arrow.com.

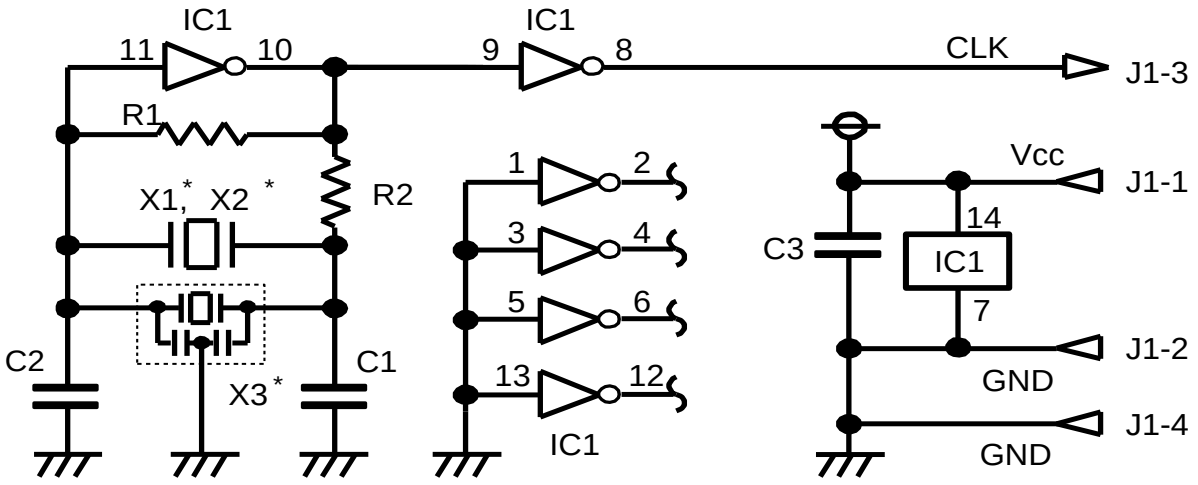
2.3.3 Using the Internal Oscillator Circuit Bare Board

To use this product at a frequency you like, build a desired oscillator circuit on the OSC-2 oscillator circuit bare board (not included). Figure 2.2 shows an external view of the OSC-2 oscillator circuit bare board and the connector pin locations. Figure 2.3 shows the circuitry of the oscillator circuit bare board OSC-2. Use the number of oscillator circuits recommended by the oscillator manufacturer.



J1-4: GND
J1-3: Oscillation output
J1-2: GND
J1-1: VCC

Figure 2.2 External View of the Oscillator Circuit Board OSC-2 and Its Connector Pin Locations



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

IC1: Inverter (Unbuffer)

Figure 2.3 Circuits of the Oscillator Circuit Bare Board OSC-2

Downloaded from Arrow.com.

2.4 Connecting the AC Adapter to the E30A Emulator

Connect the AC adapter to the E30A emulator.

(1) Check to see if the E30A emulator power switch is OFF.
(2) Connect the DC output cable of the AC adapter to the DC power input connector of the E30A emulator.
(3) Connect the AC power cable to the AC power input connector of the AC adapter.
(4) Connect the AC power cable to the outlet.

⚠ CAUTION

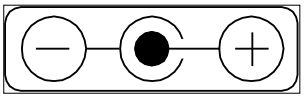
Cautions for AC Power Cable:

If the 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.

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

Caution for AC Adapter:

The DC plug of the AC adapter has the polarity shown below.



Use an AC adapter which complies with the safety standards of the country where it is to be used.

2.5 Connecting the E30A Emulator and the Host Machine

Connect the E30A emulator to the host machine with the included USB I/F cable.

(1) Connect the A-plug of the USB I/F cable to the USB I/F connector of the host machine.
(2) Connect the mini-B plug of the USB I/F cable to the USB I/F connector of the E30A emulator.

Note on USB I/F Cable:

The USB I/F cable included with the product is compliant with USB 2.0 High Speed standard. Be sure to use the included USB I/F cable for the E30A emulator.

Downloaded from Arrow.com.

2.6 Connecting the E30A Emulator and the User System

2.6.1 Using the Coaxial Connector

(1) Processing NSD pin

Pull up NSD pin of the target MCU to Vcc1 with a 4.7-kΩ resistor. (It cannot be connected directly to Vcc1 and Vcc2.) Furthermore, be sure that the wiring from NSD pin to the connector is shortest possible.

(2) Mounting part in place

Mount the coaxial connector included with the product (HRM-300-126B(40) made by Hirose Electric) on designated position of the user system.

(3) Connection procedure

Plug the 50-Ω coaxial cable (gray) included with the product into the E30A emulator and the coaxial connector on the user system.

Although the cable can be connected over a distance, the power supply condition of the user system cannot be monitored. Furthermore, since only a software reset is possible from the E30A emulator, a hardware reset is required when the emulator debugger is started or when a communication error occurs. Figure 2.4 is a connection diagram showing how to connect the E30A emulator and user system using the coaxial connector.

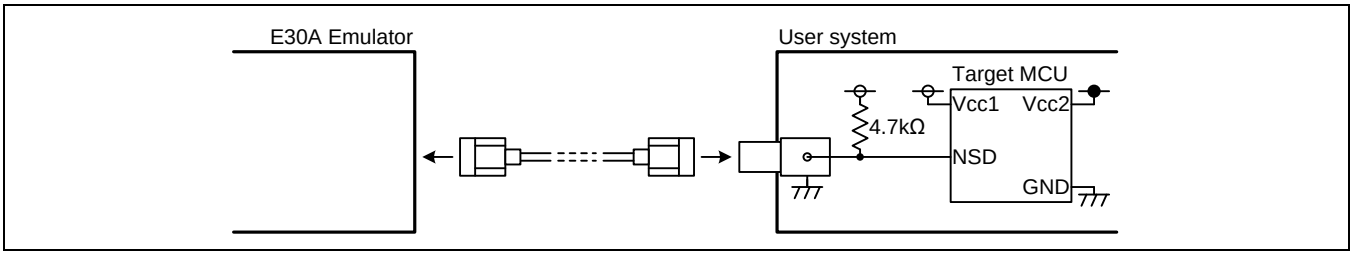


Figure 2.4 Connection Diagram when Using the Coaxial Connector

⚠ CAUTION

Caution for Insertion/Removal of the Coaxial Cable:

When inserting or removing the coaxial cable, be aware not to rotate it. The center core of the coaxial cable may be broken.

Downloaded from Arrow.com.

2.6.2 Using the Coaxial Converter Board R0E00030ACKZ00

(1) Processing NSD pin

Pull up NSD pin of the target MCU to Vcc1 with a 4.7-kΩ resistor. (It cannot be connected directly to Vcc1 and Vcc2.) Furthermore, be sure that the wiring from NSD pin to the connector is shortest possible.

(2) Mounting part in place

Mount the 2.54-mm pitch 3-pin socket included with the product (PIC-FS03-G+R made by Honda Tsushin Kogyo) on designated position of the user system.

(3) Connection procedure

Plug the 50-Ω coaxial cable included with the product (gray) into the E30A emulator and CN1 on coaxial converter board.

Fit CN2 on coaxial converter board into the 3-pin socket on user system. It can be turned 180 degrees as it is fitted into place.

Although the cable can be easily removed/inserted and can be connected over a distance, the power supply condition of the user system cannot be monitored. Furthermore, since only a software reset is possible from the E30A emulator, a hardware reset is required when the emulator debugger is started or when a communication error occurs. Figure 2.5 is a connection diagram showing how to connect the E30A emulator and user system using the coaxial converter board.

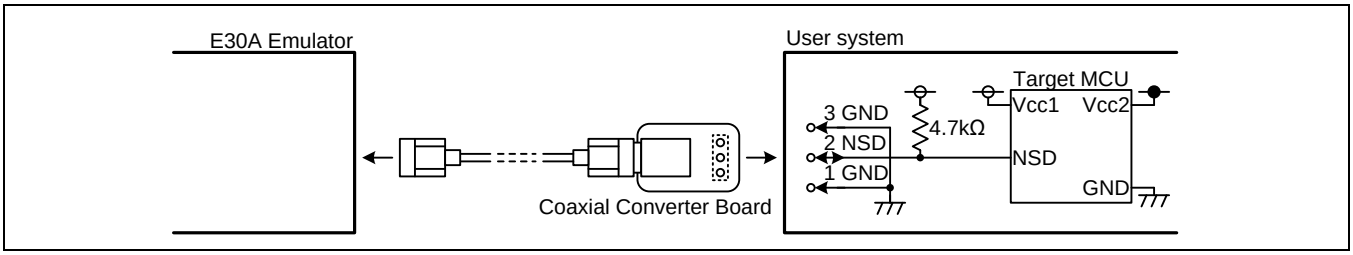


Figure 2.5 Connection Diagram when Using the Coaxial Converter Board

Downloaded from Arrow.com.

2.6.3 Using the Option Board R0E00030ACKZ10

(1) Processing NSD pin

Pull up NSD pin of the target MCU to Vcc1 with a 4.7-kΩ resistor. (It cannot be connected directly to Vcc1 and Vcc2.) Furthermore, be sure that the wiring from NSD pin to the connector (pin 4) is shortest possible.

(2) Mounting part in place

Mount the 2.54-mm pitch 8-pin socket included with the product (PIC-FS08-G+R made by Honda Tsushin Kogyo) on designated position of the user system. Furthermore, mount the 2.54-mm pitch 8-pin straight pin included with the product (PIC-MSA08-2G made by Honda Tsushin Kogyo) on CN4 (solder side) of the option board.

(3) Wiring of Vcc1 and Vcc2

Vcc1 (pin 1) is used to monitor the power supply condition of the user system and as power supply for the RST# and RSTMSK output buffers.

Vcc2 (pin 2) monitors the power supply condition of the user system.

(4) Wiring of RST# and RSTMSK

Since the target MCU is reset in hardware from the E30A emulator, connect RST# signal (pin 6) to the RESET# terminal of the target MCU.

If the reset circuit of the user system uses an open-drain output, pull it up to Vcc1 with a 4.7-k to 22-kΩ resistor. The same applies for the power-on reset case making use of a CR circuit.

If the reset signal is generated as CMOS output from other logic devices, the user system may have a RSTMSK signal connected to disable the output. Since RSTMSK input is indeterminate when the E30A emulator is not connected, it requires a pulldown resistor.

(5) Connection procedure

Plug the 10-pin flat cable included with the product into CN3 on the option board.

Plug the 50-Ω coaxial cable (gray) included with the product into CN1 on the coaxial converter board.

Fit CN2 on the coaxial converter board into CN2 on the option board. It can be turned 180 degrees as it is fitted into place.

Fit CN4 on the option board into the 8-pin socket on the user system.

Alternatively, the coaxial converter board can be attached to pins 3 to 5 of the 8-pin socket on the user system without using the option board.

(6) Compatibility with the E30 emulator (old product)

If you have hitherto been using the 8-pin flat cable of the E30 emulator, you can attach the 2.54-mm pitch 8-pin right angle connector included with the product (FFC-8LAMEP1B made by Honda Tsushin Kogyo) to CN4 (component side) of the option board, for connection to the user system. However, since the non-coaxial NSD wiring becomes longer, it is recommended that the connector on the user system be replaced with the 8-pin socket, and that the 8-pin straight pin be used for connection.

Downloaded from Arrow.com.

Since a hardware reset can be generated from the E30A emulator, the target MCU does not need to be reset. Figure 2.6 is a connection diagram showing how to connect the E30A emulator and user system using the option board.

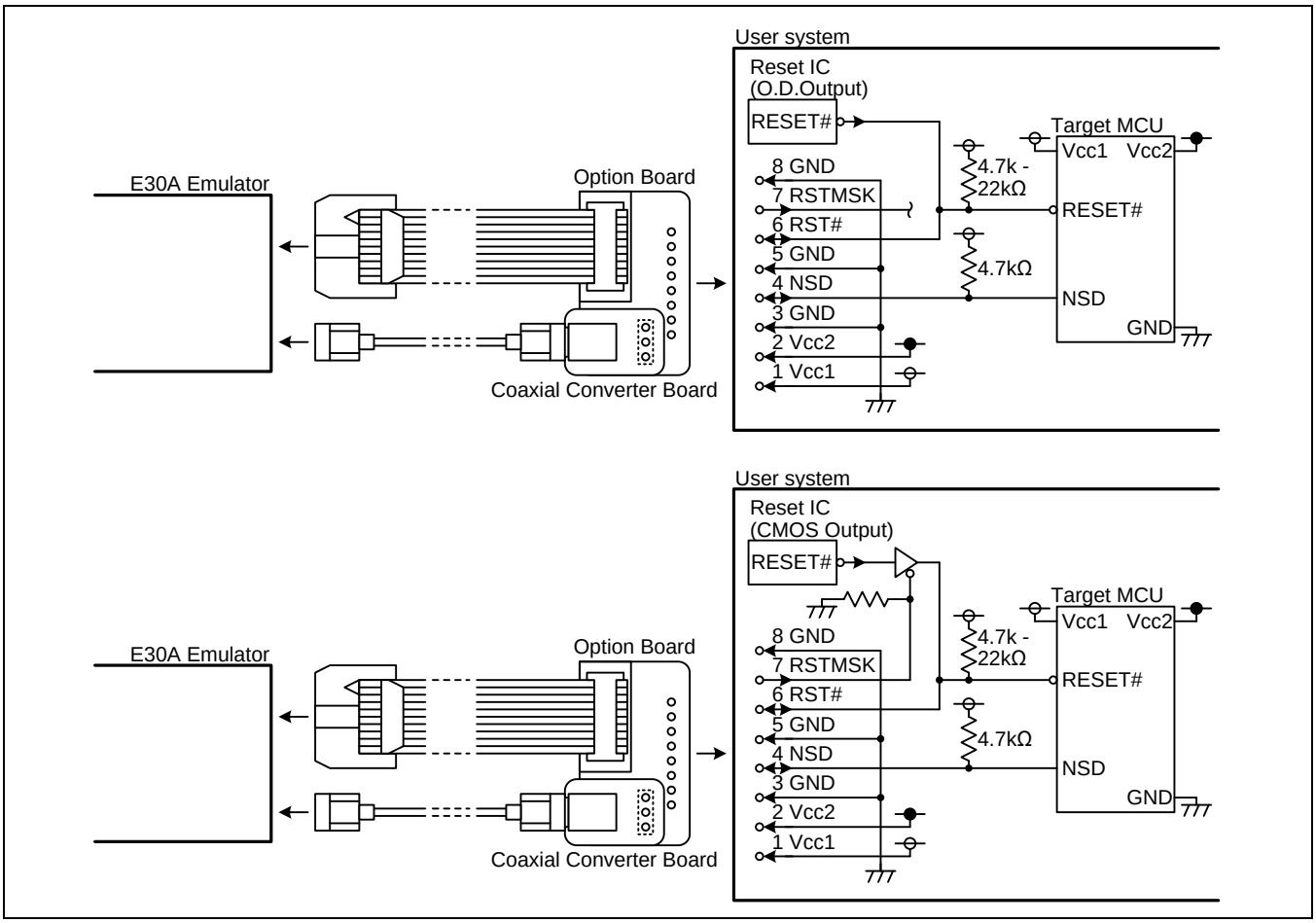


Figure 2.6 Connection Diagram when Using the Option Board

Downloaded from Arrow.com.

2.7 Turning On the Power

2.7.1 Checking the Connections of the Emulator System

Verify the connection of the emulator with the AC adapter, host machine and user system.

2.7.2 Turning the Power On and Off

Turn on/off the power of the emulator and user system as simultaneously as possible. If you cannot turn on the powers simultaneously, turn on the emulator first, then the user system.

Before turning on the power again after shutting off the power, wait about 10 seconds.

Notes on Power Supply to the User System:

The emulator does not supply power to the user system. Provide power to the user system separately.

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

Downloaded from Arrow.com.

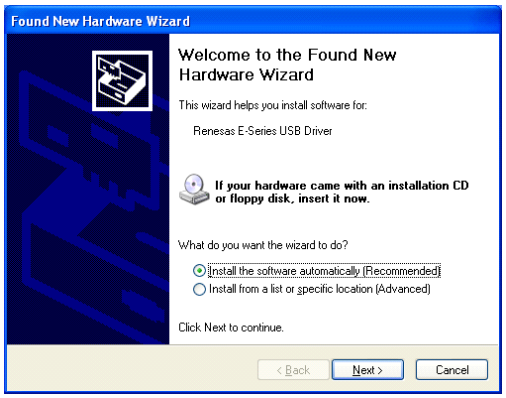
2.8 Installing the USB Device Driver

The installation wizard for USB device drivers starts after the device had been detected by the host machine.

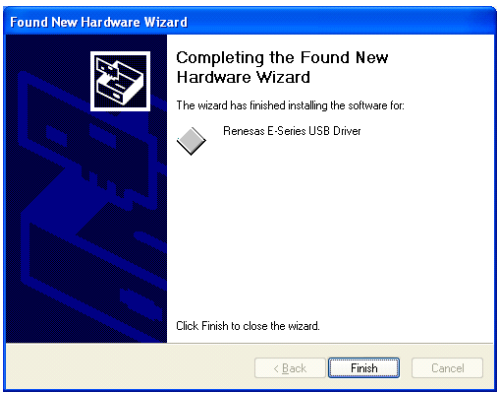
Figure 2.7 shows the procedure for installing the USB device drivers for Window® XP. For other OS, the driver will be installed automatically.



Select [No, not this time] and click [Next].



Click [Next].



The USB driver will be installed automatically.
Click [Finish] to exit.

Figure 2.7 Installation of USB Device Driver for Windows® XP

Note on USB Port Change:

If you change the port to connect USB, the device driver is installed over again.

Note on Found New Hardware Wizard:

For Windows® XP, the amount of time needed for the wizard to appear depends on the CPU usage rate of the host machine. For example, approximately 9 seconds of time is needed when the CPU usage rate is 0%, whereas approximately 30 seconds is needed when the CPU usage rate is 100%.

Downloaded from Arrow.com.

2.9 Checking the Status LEDs Display

Check that the SYS POWER LED and SAFE LED are lighting.

If the SAFE LED is off and the ERROR LED is on, it means that the E30A emulator hardware is not operating normally. When the SAFE LED is off and the ERROR LED blinks, it means the internal firmware of the E30A emulator is not running normally. When the status LED indicates abnormal, perform the forced download as follows.

2.10 Downloading Firmware

It is necessary to upgrade 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.

- When you use this product for the first time
- When the firmware has been upgraded
- When the emulator debugger has been upgraded

If downloading firmware is not completed when the power is unexpectedly shut down during a download from the emulator debugger, the status LEDs show abnormal and the emulator debugger may not start up normally. Download firmware in the special mode called maintenance mode as explained here following.

(1) Within 2 seconds of activating power to the emulator, press the maintenance switch on it to start maintenance mode. The System Status SAFE LED begins to blink.

(2) Start up the emulator debugger in maintenance mode. When settings in the [Init] dialog box are complete, the dialog box which urges to download the firmware will appear. Download firmware following messages. Required time for downloading the firmware is about 15 seconds.

Downloaded from Arrow.com.

3. Starting Up the Emulator Debugger

3.1 Switching the Session

When you finish creating a program and start debugging, switch the session of the High-performance Embedded Workshop. Select [SessionR32C_100_E30A_Emulator] from the dropdown list in [Standard] toolbar.



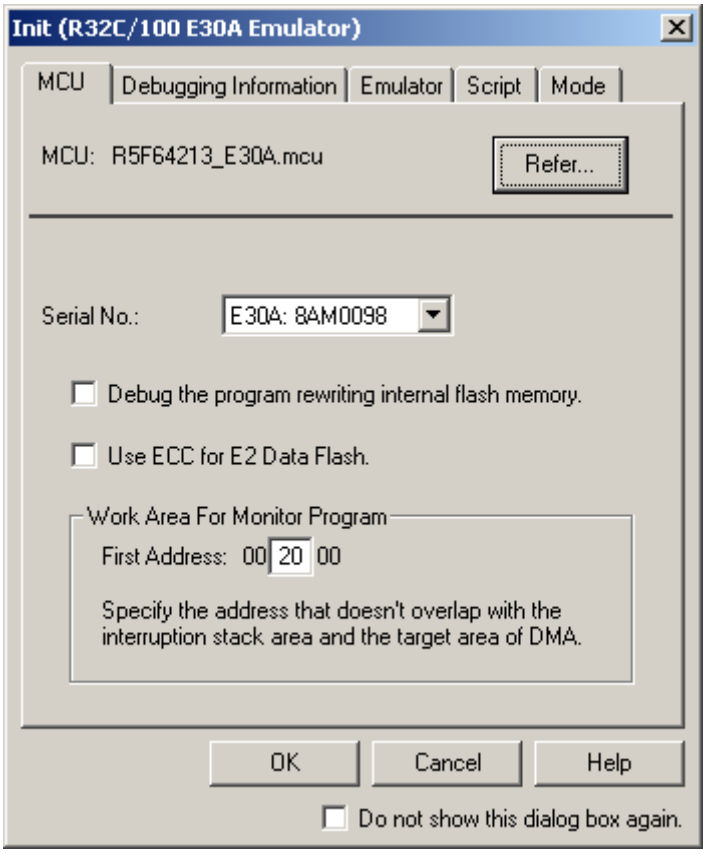
Downloaded from Arrow.com.

3.2 Initial Setting of the Emulator Debugger

R32C/100 E30A emulator debugger starts up and the [Init] dialog box appears. When you finished setting of each tab, click [OK].

3.2.1 [MCU] Tab

This tab permits you to select an MCU file and the connected emulator, and specify whether to debug a program in CPU rewrite mode, whether to use ECC for E2 data flash and a work area for the debug monitor program.



(1) MCU

An MCU file (MCU type name_E30A.mcu) contains the information specific to the target MCU. Select the MCU file applicable to the target MCU.

— Click the [Refer…] button to open the [Select MCU File] dialog box.

— Select the desired MCU file.

Note on MCU file:

When you cannot select the MCU to be used, add the MCU file to the folder below. For details on getting the MCU file, contact your local distributor.

Install folder of the High-performance Embedded Workshop\Tools\Renesas\DebugComp\Platform \PDTarget\PD100E30A\McuFiles

(2) Serial No.

Select the serial number of the emulator to be used from the connected emulator list.

Downloaded from Arrow.com.

(3) Debug the program rewriting internal flash memory.

Select this checkbox when you want to debug a program in CPU rewrite mode. In this mode, software breaks in the internal ROM area of the MCU cannot be used.

Note on Debugging in CPU Rewrite Mode:

Do not stop the user program during debugging in CPU rewrite mode. Otherwise, the emulator will be out of control. Furthermore, before executing a user program, be sure to disable automatic updating of the windows to ensure that no memory accesses will occur during execution of the user program. To inspect data after CPU rewrite operation, stop the program after exiting CPU rewrite mode and inspect the data in the memory window or another appropriate window.

(4) Use ECC for E2 Data Flash

Select this checkbox when you want to use ECC for E2 Data Flash.

(5) Work Area For Monitor Program

Specify the start address of the internal RAM area of the MCU to be used by the debug monitor program.

Approximately 1.2 KB of the internal RAM will be used as needed, for example, to download a user program. Since the memory contents are saved before debugging starts, you do not need to be specifically concerned, except that the following cannot be specified.

— Areas overlapping the interrupt stack

Areas which overlap with the interrupt stack to be used in the user program cannot be specified. Also, the address 0400h cannot be specified since it is used as a temporary stack by the emulator after reset is released.

In the case of the 63-KB internal RAM (0400h-FFFFh), 0500h/0600h … FA00h/FB00h can be specified.

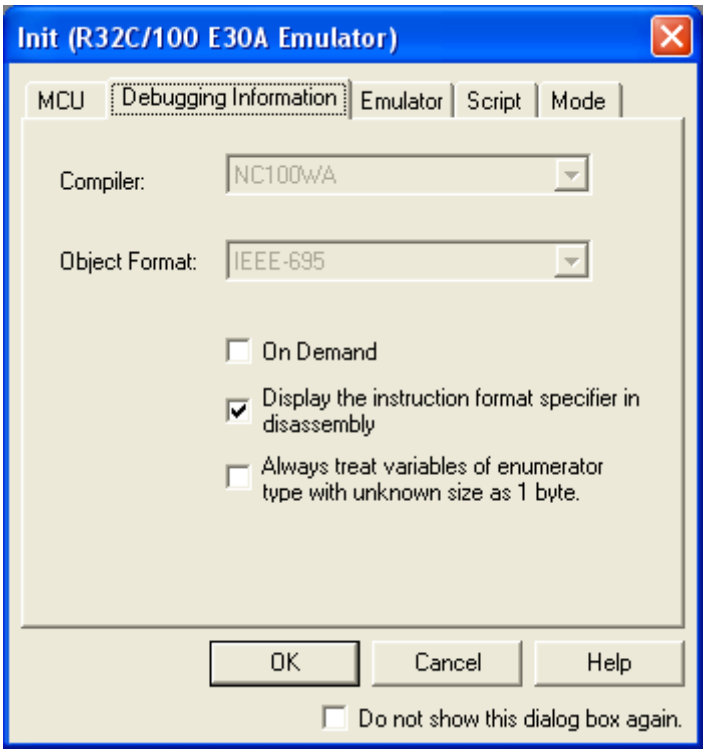
In the case of the 32-KB internal RAM (0400h-83FFh), 0500h/0600h … 7E00h/7F00h can be specified.

— Areas to or from which DMA-transferred

Downloaded from Arrow.com.

3.2.2 [Debugging Information] Tab

This tab permits you to select a compiler, select an object format, and choose to specify [On Demand] for debug information.



(1) Compiler

Select the compiler you are using.

(2) Object Format

Select the format of the object file that is output by the compiler you are using.

(3) On Demand

Select this checkbox when you want to hold debug information in a temporary file in order to reduce the amount of memory used by the host machine.

(4) Display the instruction format specifier in disassembly

Specify whether or not to show the instruction format specifier in disassemble display.

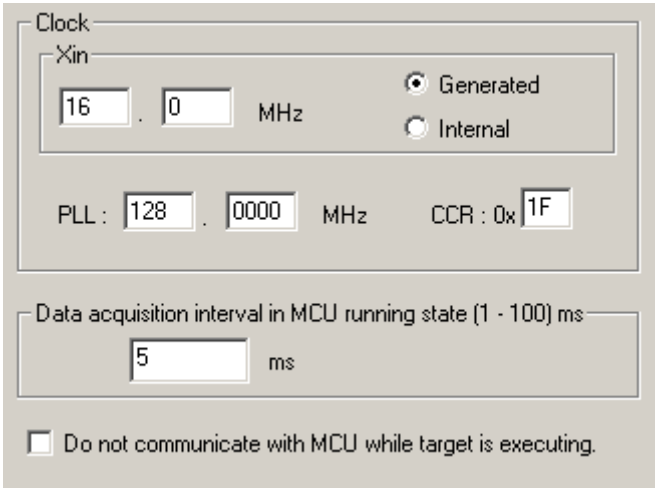
(5) Always treat variables of enumerator type with unknown size as 1 byte

If the debug information contains information on an enumerator type that does not have size information, it is possible to specify whether to handle it always as being 1 byte in size.

Downloaded from Arrow.com.

3.2.3 [Emulator] Tab

This tab permits you to specify the clock supplied to the target MCU and data acquisition interval of the RAM monitor.



(1) Clock

Specify the X_{IN} clock and the frequency that supplied to the target MCU.

- Xin: Enter the frequency of the X_{IN} clock, and select [Generated] (internal generator circuit) or [Internal] (oscillator circuit board).
- PLL: Enter the frequency of the MCU's internal PLL, which should be the same as the value to be set in the user program.
- CCR: Enter the value for the MCU's internal clock control register, which should be the same as the value to be set in the user program.

Notes on Clock Setting:

The emulator sets the PLL-related settings before starting up in order to secure a communication path. In a user program, be sure to set the same value of the PLL-related settings specified by the emulator debugger. Otherwise, the emulator will be out of control. The divide ratios of the CPU clock and peripheral clock can be changed in the user program.

The X_{IN} clock is used to enter the debug mode and to stop the user program forcibly. Therefore, do not stop the X_{IN} clock in the user program.

The PLL clock divided by the base clock divider is used for communication between the emulator and the MCU. Therefore, do not change the PLL frequency and the divide ratio of the base clock in the user program.

When the emulator is connected, the PLL clock stop bits are ignored. Therefore, the PLL keeps oscillating even when the WAIT or STOP instruction is executed. This means that BCLK is also oscillating, so that control returns from WAIT or STOP by an INT8, 7 or 6 or other interrupt request.

The clock can be switched to a low speed clock (f256, fC, fOCO4). However, as the CPU operation clock becomes slower, the response speed of the emulator debugger becomes also slower. Furthermore, the internal flash ROM cannot be rewritten during a low speed clock operation. Therefore, do not use software breaks.

Downloaded from Arrow.com.

(2) Data acquisition interval in MCU running state (1 - 100) ms

Specify the data acquisition interval (byte or word data). At this interval the functions such as the RAM monitor acquire the memory data while the user program is running. 1 - 100 ms can be specified.

Note on Data Acquisition during MCU Execution:

Using a function that acquires the memory data during user program execution, such as RAM monitor function, results in the realtime capability becoming impaired because the debug monitor program intervenes. Also, in order to complete processing of the debug monitor program as fast as possible, do not operate the MCU's internal DMA during the processing.

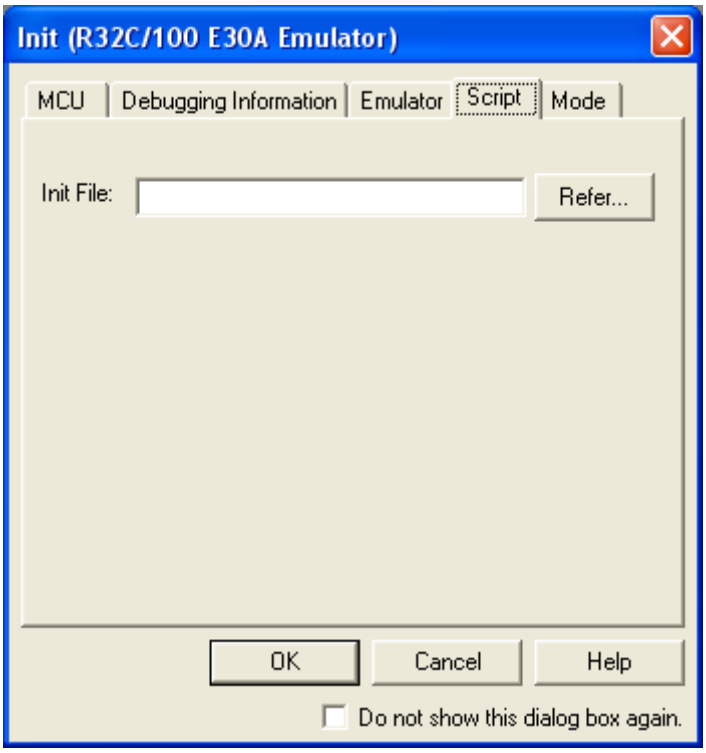
(3) Do not communicate with MCU while target is executing.

Specify whether or not to communicate with the MCU when executing a user program. While the user program is running, the emulator communicates with the MCU, for example, to check the operating status of the MCU, or to receive the trace information from the MCU.

The PLL clock divided by the base clock divider is used for communication between the emulator and the MCU. Therefore, when executing the STOP or WAIT instruction, or when temporarily switching to the PLL self-oscillation mode, select this checkbox to avoid communication breakdown. The debug function cannot be used because the emulator does not communicate with the MCU when executing the user program. However, it is possible to press the stop button after restored to the base clock specified in the [Clock] column.

3.2.4 [Script] Tab

This tab permits you to specify a script file.



(1) Init File

If you have any script that is automatically executed at startup, specify that file.

— Click the [Refer…] button to open the [Select Script File] dialog box.

— Select the desired script file.

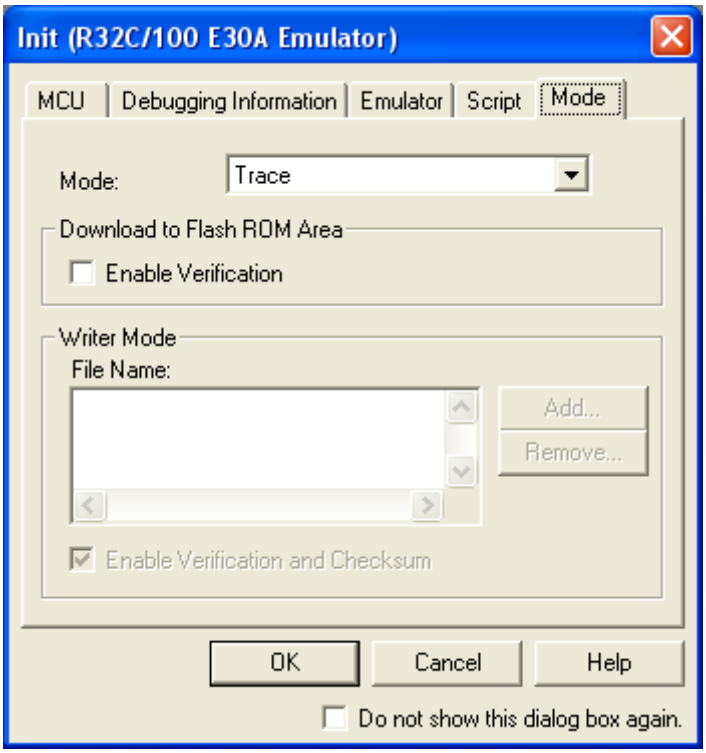
Downloaded from Arrow.com.

3.2.5 [Mode] Tab

This tab permits you to select an operating mode and specify the download to flash ROM area and the writer mode. Table 3.1 lists the debug functions depending on the operating mode.

Table 3.1 Debug Functions Depending on the Operating Mode

Mode	Break				Trace	Time Measurement	RAM Monitor
	Execution Address	Data Access	Address Range	Data Comparison			
Trace: Trace priority	Yes	Yes	-	-	Yes	-	-
Trace: MCU execution priority	Yes	Yes	-	Yes	Yes	-	-
Time Measurement	Yes	Yes	-	-	-	Yes	-
RAM Monitor	Yes	Yes	Yes	-	-	-	Yes
Writer	-	-	-	-	-	-	-



(1) Mode

Select an operating mode from [Trace], [Time measurement], [RAM monitor] or [Writer].

- Trace: You can record the execution history of the user program and data access information. [Trace priority] or [MCU execution priority] can be selected from [Trace Mode] in the [Event Setting] window.
- Time Measurement: You can measure an execution time between Go and Stop, or a section time between data accesses based on trace information.
- RAM monitor: You can view memory data while the user program is running.
- Writer: You can write the specified object files to the internal flash ROM of the target MCU.

(2) Download to Flash ROM Area

If you want to verify the flash ROM area when downloading a user program, select the checkbox.

Downloaded from Arrow.com.

(3) Writer Mode

This function is valid when [Writer] is selected. This mode is used when using the E30A emulator as the flash ROM writer. Select a load module file to be downloaded.

— Click [Add…] to open a dialog box to select a load module file. Multiple files can be selected.

— To cancel the selection, select the file and click [Remove…].

If you want to enable the verification and checksum functions, select the checkbox.

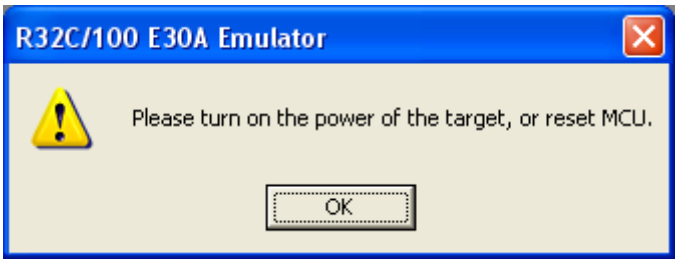
You cannot debug the program in writer mode. Restart or terminate the emulator debugger after writing the program.

Notes on Writing the ID code:

The ID code is written into the flash memory only when the program was downloaded in writer mode. When operating in other than the writer mode, no ID code is written into the flash memory even when the program is downloaded. In that case, an ID code consisting entirely of FFh is assumed.

3.3 Resetting the User System

When using the coaxial connector or the coaxial converter board for connecting the E30A emulator and user system, by clicking [OK] in the [Init] dialog box, the dialog box shown below will be displayed. When using the option board, the dialog box is not displayed because the target MCU is reset in hardware from the emulator.



- If the user system has already been powered on, reset the user system. If the power for the user system is turned off, turn the power on.
- Click [OK].

Notes on Power Supply to the User System:

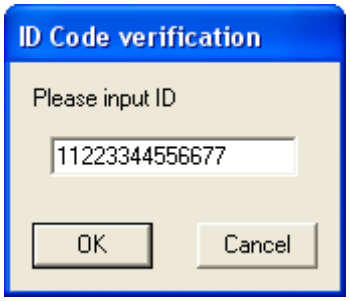
The emulator does not supply power to the user system. Provide power to the user system separately.

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

Downloaded from Arrow.com.

3.4 Entering ID Code

If the target MCU has ID code set in it, the [ID Code verification] dialog box shown below is displayed. Enter the 7-byte ID code written to the internal flash ROM into the dialog box successively. Table 3.2 shows the ID code storage area.



- Enter ID1 to ID7 successively in that order and then click [OK]. Unless all IDs match, the emulator debugger will not start.

Table 3.2 ID Code Storage Area

Address	Description
FFFF_FFE8h	First byte of ID code
FFFF_FFE9h	Second byte of ID code
FFFF_FFEAh	Third byte of ID code
FFFF_FFEBh	Fourth byte of ID code
FFFF_FFECh	Fifth byte of ID code
FFFF_FFEDh	Sixth byte of ID code
FFFF_FFEEh	Seventh byte of ID code

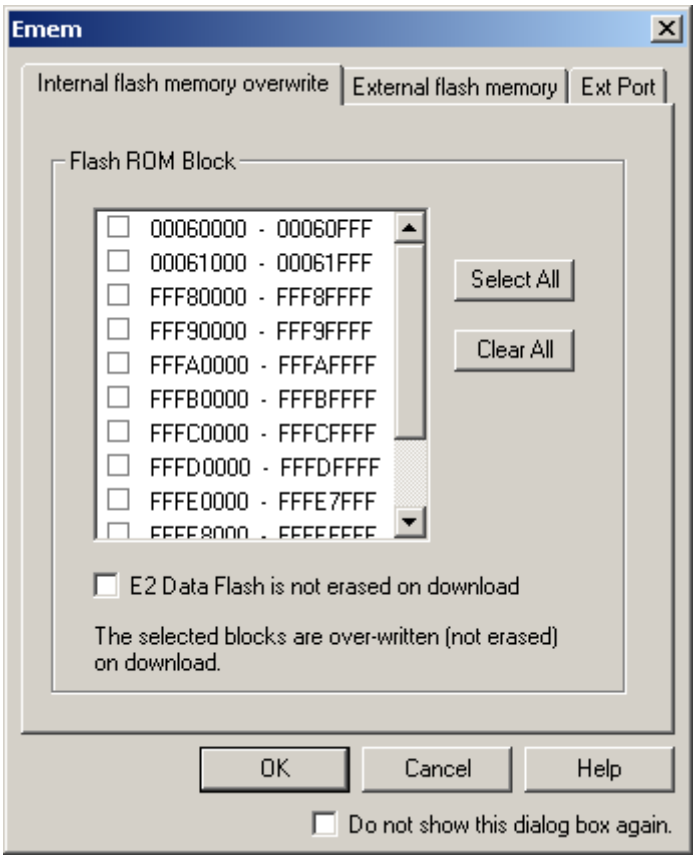
Downloaded from Arrow.com.

3.5 Setting the Emulator Debugger Operating Environment

When a communication path between the E30A emulator and target MCU is established, the [Emem] dialog box shown below is displayed. Set each tab described below and click [OK].

3.5.1 [Internal flash memory overwrite] Tab

This tab permits you to specify whether or not to overwrite each block of the internal flash ROM in the MCU.



(1) Flash ROM Block

If you want to keep the data when you download a user program or data table, check the blocks of contents you want to keep intact. The address list is expressed in units of internal flash ROM blocks of the MCU. The selected blocks, those flagged by a check mark, require extra time because their contents need to be read out before being programmed. Therefore, we recommend leaving blocks unselected.

- When you select the checkbox for any block, the content of that block is read out into the host machine and the data to be downloaded is written over it inside the PC before the block is erased and programmed.
- When you deselect the checkbox for any block, the data to be downloaded is written to the block after being erased.
- Click the [Select All] button, and all blocks will be selected.
- Click the [Clear All] button, and all blocks will be deselected.

(2) E2 Data Flash is not erased on download

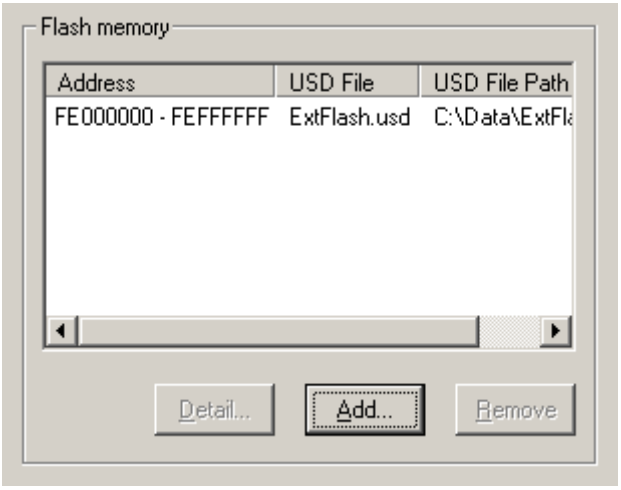
If you want to keep the data in the E2 data flash when you download a user program or data table, select this checkbox.

- When you select the checkbox, the data in the E2 data flash is read out into the host machine and the data to be downloaded is written over it inside the PC before the block is erased and programmed.
- When you deselect the checkbox, the data to be downloaded is written to the block after being erased.
- When E2 data flash is not embedded in the MCU, this setting is invalid.

Downloaded from Arrow.com.

3.5.2 [External flash memory] Tab

Set up this tab when you download a program, etc. into the flash memory (external flash memory) that is connected to the MCU's external bus.



(1) Registering the USD file

External flash memory is recognized based on information in the USD file that you register. This file defines the manner and other details of how the flash memory is connected with the user system.

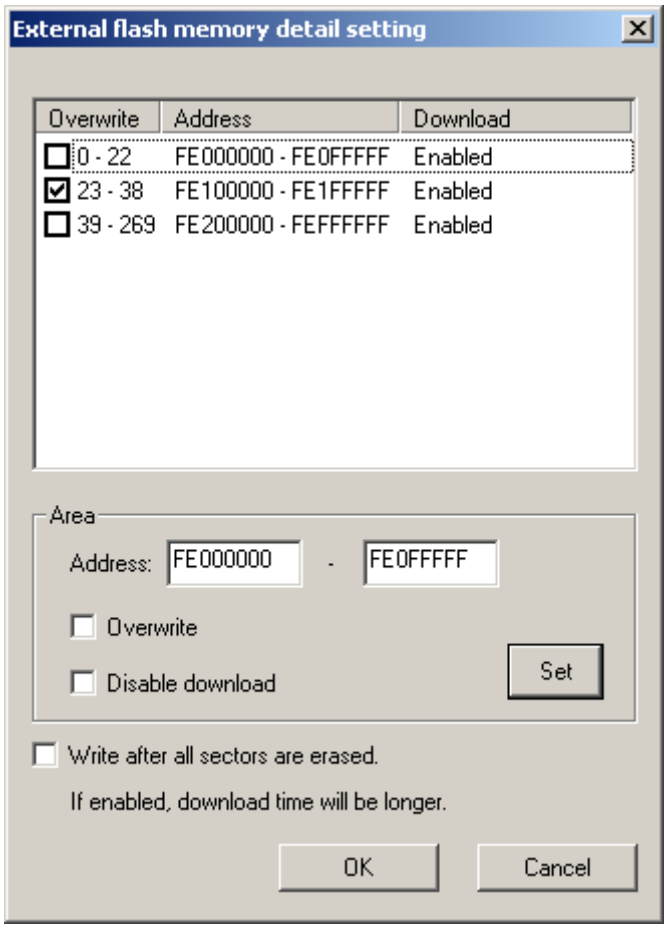
- Click the [Add…] button, and open a dialog box in which you can select USD files.
- Select a USD file to use.
- Up to four USD files can be added.

To create USD files, use the External Flash Definition Editor. For details on how to create USD files, refer to the user's manual for the External Flash Definition Editor, which is available at https://www.renesas.com/efe.

Downloaded from Arrow.com.

(2) External flash memory detail setting

Click the [Detail…] button to open the [External flash memory detail setting] dialog box.



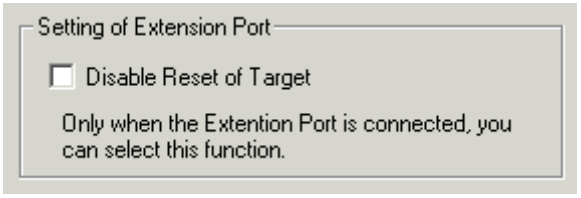
This dialog box allows you to specify whether or not individual sectors (blocks) of external flash memory should be overwritten.

- Settings per sector of the selected external flash memory are automatically displayed in the list.
- When the checkbox in the [Overwrite] column is selected, the selected sector is overwritten (merged) when the program is downloaded.
- To change per-sector settings, enter the start and end addresses in the [Address] edit box and click the [Set] button. If the sector thus changed has had its [Overwrite] option selected, it is overwritten without erasure when the program is downloaded. If the sector has [Disable download] selected, it is neither erased nor downloaded at download time.
- If [Write after all sectors are erased.] is selected, all sectors will be erased when the user program is downloaded.

Downloaded from Arrow.com.

3.5.3 [Ext Port] Tab

This tab permits you to set the extension port function to connect an option connector.



(1) Setting of Extension Port

Specify whether to disable the reset signal for the user system. Unless the option board is connected, this setting is not available to use. When this checkbox is selected, a high output to RSTMSK is generated to disable reset of the user system. For details, refer to "2.6.3 Using the Option Board R0E00030ACKZ10".

Downloaded from Arrow.com.

4. Debugging Functions

4.1 Break Function

4.1.1 Software Break

If the breakpoint you set is a software breakpoint, the program stops before executing the instruction at the set breakpoint.

- Up to 256 break points can be set.
- You can specify breakpoints in the internal RAM of the MCU, the internal flash ROM of the MCU, and the RAM on the user system.
- If breakpoints in the internal flash ROM of the MCU are specified, the need arises for instruction rewrite/writeback processing in block units.

4.1.2 Execution Address Break

This is a kind of event break by the internal debug circuit of the MCU. It specifically is a pre-execution break that causes execution of the user program to stop at a specified address.

- Up to 8 break points can be set (used for hardware breaks, trace points, and time measurement events in common).
- Since there is no need for instruction rewrite/writeback processing, the response speed is fast.

4.1.3 Hardware Break

This is a kind of event break by the internal debug circuit of the MCU. It specifically is a post-execution break that causes execution of the user program to stop upon detecting data access at a specified address.

- Up to 8 break points can be set (used for execution address breaks, trace points, and time measurement events in common).
- You can set a specified address range (using 2 events).
- You can set three conditions, e.g., read, write, and read/write.
- Only one instance of data comparison break can be specified (event E5 only). Several bytes from the beginning of the MCU's internal RAM (8 bytes at the addresses 0400h to 0407h) are used for data comparison by the debug monitor program. These bytes cannot be used in the user program when using a data comparison break.

Notes on Execution Address Break and Hardware Break:

Do not set breaks in the area designated below. This restriction is due to specifications of the MCU's internal debug circuit.

0100 0000h – FEFF FFFFh

Note on Data Comparison Break:

For data comparison break, the debug monitor program compares the data after the debug interrupt occurs at the specified address to determine whether or not to cause a break. Using this function results in the realtime capability becoming impaired because the debug monitor program intervenes. Also, in order to complete processing of the debug monitor program as fast as possible, do not operate the MCU's internal DMA during the processing.

In the case of X_{IN} 8 MHz, PLL 128 MHz (16 times), BASE 64 MHz and CPU 64 MHz, for example, it takes 6 µs to return to the user program after the data comparison results in a mismatch.

Downloaded from Arrow.com.

4.1.4 Break Combination

Event breaks by the internal debug circuit of the MCU can be cascaded to constitute a state transition break at up to 8 points in the forward direction. If all of the events are used for a state transition break, program execution will break when the events E7, E6, ..., and E0 all have occurred in that order.

Downloaded from Arrow.com.

4.2 Trace Function

4.2.1 Trace Capacity

The following can be selected depending on the trace mode chosen in the event setting window.

- Trace priority
  8-M events can be traced. The trace information is output from the MCU and recorded to the emulator trace memory. The user program execution may be delayed depending on the event occurrence interval because the MCU processes the trace information output prior to others.
- MCU execution priority
  512 events can be traced. The trace information is recorded to the trace RAM of the MCU's internal debug circuit. Therefore, this mode does not affect the execution speed of the user program.

Note on Affecting User Program in Trace Priority Mode:

In the case of X_{IN} 8 MHz, PLL 128 MHz (16 times), BASE 64 MHz and CPU 64 MHz, for example, it takes 5 µs to output trace information for each event. If the average of event occurrence interval among 512 events is less than 5 µs, the CPU operation will be suspended, because the trace information is output via the trace RAM of the MCU's internal debug circuit.

4.2.2 Trace Contents

You can select information on branch sources and destinations, information on conditional branches, information on data accesses, or mixed information of branch sources and destinations, and data accesses.

- Information on branch sources and destinations
  Branch source address and branch destination address are recorded as an event.
- Conditional branch information
  Branch information (success/failure of conditional branches) are recorded in a compressed format. Up to 48 of conditional branches are recorded as an event, so that the communication amount of the trace information can be reduced and the affect on execution speed of the user program can be reduced even in trace priority mode.
  To record conditional branch information, cancel all trace event settings in the event setting window. An unused event can be used as a break event.
- Data access information
  Address, data, data size and read/write information are recorded as an event.

Note on Jump in Queue:

R32C/100 series has a function that branches to the instructions fetched into the prefetch queue, to improve the processing performance, but the branch source and destination information is not output.

Notes on the Data Access Trace:

16-bit word data starting from an odd address and 32-bit longword data assigned differently from the 4-byte alignment cannot be traced properly.

Data access event by DMA is not the subject of detection.

Downloaded from Arrow.com.

4.2.3 Extraction Trace

An extraction trace with the start event E0 and the end event E4 specified is possible.

The events E1 to E3 or E5 to E7 that occurred during an interval between E0 to E4 are recorded in the trace RAM.

Note on Execution Address Trace Event:

When specifying the execution address as the trace event, the results may be different from what you expect because the event is encountered at the prefetch address.

Note on Trace Event:

Do not set events in the area designated below. This restriction is due to specifications of the MCU's internal debug circuit.

0100 0000h – FEFF FFFFh

4.3 Time Measument Function

4.3.1 Execution Time Measurement

You can measure a program execution time between Go and Stop. The 32-bit CPU clock counter allows you to measure a program execution time of up to 67.1 s when operating at 64 MHz.

4.3.2 Section Time Measurement

You can measure a program execution time between data accesses by trace information. Data access trace information (address, data, data size and read/write) is compared/detected as the measurement start event and measurement end event inside the emulator. The 40-bit asynchronous clock counter inside the emulator enables the section time measurement of more than 24 hours.

As a condition of section time measurement, you can specify that the measurement be taken of a time from when the start event occurs till when the end event occurs or a cycle time with which individual events occur.

Notes on Time Measurement Function:

The execution time and the section time cannot be measured simultaneously. Select either an execution time measurement or a section time measurement.

In a section time measurement performed, the data access trace information output from the MCU is compared in the emulator to detect the start and end events. If a large number of data access times is specified, trace output will take a lot of time, leading to errors in section time measurement or affecting execution speed of the user program.

In the case of X_{IN} 8 MHz, PLL 128 MHz (16 times), BASE 64 MHz, for example, it takes 5 µs to output the data access trace information for each event. If a data access of less than 5 µs occurs, a trace output time-dependent error will occur. Furthermore, if such a short data access occurs frequently, execution speed of the user program will be delayed. We recommend write-only settings with least possible events.

Downloaded from Arrow.com.

4.4 RAM Monitor Function

You can view memory data during user program execution by a pseudo-RAM monitor function which is realized by the software using event E5. Data is acquired in byte/word unit by the interval specified in [Data acquisition interval in MCU running state] in the [Emulator] tab of the [Init] dialog box.

Notes on RAM Monitor Function:

The RAM monitor function results in the realtime capability becoming impaired because the debug monitor program intervenes. Also, in order to complete processing of the debug monitor program as fast as possible, do not operate the MCU's internal DMA during the processing.

For the RAM monitor function, the emulator generates a debug interrupt and the debug monitor program sends the data to return control to the user program.

In the case of X_{IN} 8 MHz, PLL 128 MHz (16 times), BASE 64 MHz and CPU 64 MHz, for example, it takes 6 µs from the debug interrupt occurred to return to the user program.

The RAM monitor function should be adjusted for sampling intervals (byte/word data acquisition intervals in ms units) during user program execution. Since the updating time is extended if a wide area is displayed in bytes, we recommend displaying only the necessary area in word size.

Downloaded from Arrow.com.

5. Troubleshooting

5.1 How to Solve a Trouble

The following describes how to solve a trouble when any problem occurs before the emulator debugger starts after the emulator and the user system have been powered on. For the latest information on the emulator and emulator debugger, visit Renesas Electronics' development environment website (https://www.renesas.com/tools).

5.1.1 The SAFE LED Remains Off

- The USB cable may not be connected properly. Check the condition of the cable connection.
- The E30A may not be powered on. Check the condition of the AC adapter connection and that of the power switch.

5.1.2 The SAFE LED Keeps Flashing

- The USB cable may not be connected properly. Check the condition of the cable connection.
- The USB driver may not be installed properly. Reinstall the USB driver.

5.1.3 The Emulator Debugger Cannot Be Activated

- Check whether the E30A and the user system are connected properly.
  The MCU's NSD pin and the E30A connector may be connected by an excessively long wiring.
  The MCU's NSD pin may not be pulled high with a 4.7-kΩ resistor.
- Check the [Emulator] tab of the [Init] dialog box to see if the clock is set up properly.
  Make sure the value entered for the Xin frequency is the same as that of the user system.
  Check that [Generated] is selected.
  Make sure the values entered for the PLL frequency and CCR set value are the same as those set in the user program.
- If connected with coaxial cable, you need to reset the user system (or power on the user system) after a dialog box is displayed by clicking the [OK] button in the [Init] dialog box.

Downloaded from Arrow.com.

5.2 How to Request Support

After checking the items in "5 Troubleshooting", contact us from the following URL.

https://www.renesas.com/contact

For prompt response, please specify the following information:

(1) Operating environment
- MCU Type name: R5F64_ _ _
- Xin frequency: _____ [MHz] (Generated / Internal)
- PLL frequency: _____ [MHz]
- CCR value: 0x_____
- Debugger operating mode:

  Trace: Trace priority / Trace: MCU execution priority / Time Measurement / RAM Monitor

(2) Condition
- The emulator debugger starts up/does not start up
- Frequency of errors: always/frequency

(3) Problem

Downloaded from Arrow.com.

6. Maintenance and Warranty

This chapter covers basic maintenance, warranty information, provisions for repair and the procedures for requesting a repair.

6.1 User Registration

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

6.2 Maintenance

(1) If dust or dirt collects on this product, wipe it off with a dry soft cloth. Do not use thinner or other solvents because these chemicals can cause the surface coating to separate.

(2) When you do not use this product for a long period, disconnect it from the power supply, host machine and user system.

6.3 Warranty

(1) This product comes with a one-year warranty after purchase.

Should the product break down or be damaged while you're using it under normal condition based on its user's manual, it will be repaired or replaced free of cost.

(2) However, if the following failure or damage occurs to the product under warranty, the product will be repaired or replaced at cost.

a) Failure or damage attributable to the misuse or abuse of the product or its use under other abnormal conditions.

b) Failure or damage attributable to improper handling of the product after purchase, such as dropping of the product when it is transported or moved.

c) Failure or damage to the product caused by other pieces of equipment connected to it.

d) Failure or damage attributable to fire, earthquakes, thunderbolts, floods, or other natural disasters or abnormal voltages, etc.

e) Failure or damage attributable to modifications, repairs, adjustments, or other acts made to the product by other than Renesas Electronics Corporation.

(3) Consumables (e.g., sockets and adapters) are not covered by the aforementioned repair.

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

Downloaded from Arrow.com.

6.4 Repair Provisions

(1) Repairs not covered by warranty

Problems arising in products for which more than one year has elapsed since purchase are not covered by warranty.

(2) Replacement not covered by warranty

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

- Faulty or broken mechanical portions
- Flaws, separation, or rust in coated or plated portions
- Flaws 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
- A wide range of faults that make replacement less expensive than repair
- Faults that are not locatable or identifiable

(3) Expiration of the repair period

When a period of one year has elapsed after production of a given model ceased, repairing products of that model may become impossible.

(4) Carriage fees for sending your product to be repaired

Carriage fees for sending your product to us for repair are at your own expense.

6.5 How to Make Request for Repair

If your product is found faulty, fill in a Repair Request Sheet downloadable from the following URL. And email the sheet and send the product to your local distributor.

https://www.renesas.com/repair

⚠ CAUTION

Note on Transporting the Product:



When sending your product for repair, use the packing box and cushioning material supplied with the product when it was delivered to you and specify caution in handling (handling 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 the conductive plastic bag supplied with the product (usually a blue bag). If you use a different bag, it may lead to further trouble with your product due to static electricity.

Downloaded from Arrow.com.

Appendix A Target MCU Specifications

Table A.1 lists the specifications of target MCUs which can be debugged with this product.

Table A.1 Specifications of the Applicable Target MCU

Item	Description
Applicable MCU	M16C Family R32C/100 Series
Usable MCU mode	Single-chip mode Memory expansion mode
Maximum operating frequency	100 MHz (CPU clock)
Supported ROM/RAM capacity	Internal ROM/RAM capacity of the MCU

Downloaded from Arrow.com.

Appendix B 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.

Notes on Clock Setting:

The emulator sets the PLL-related settings before starting up in order to secure a communication path. In a user program, be sure to set the same value of the PLL-related settings specified by the emulator debugger. Otherwise, the emulator will be out of control. The divide ratios of the CPU clock and peripheral clock can be changed in the user program.

The X_{IN} clock is used to enter the debug mode and to stop the user program forcibly. Therefore, do not stop the X_{IN} clock in the user program.

The PLL clock divided by the base clock divider is used for communication between the emulator and the MCU. Therefore, do not change the PLL frequency and the divide ratio of the base clock in the user program.

When the emulator is connected, the PLL clock stop bits are ignored. Therefore, the PLL keeps oscillating even when the WAIT or STOP instruction is executed. This means that BCLK is also oscillating, so that control returns from WAIT or STOP by an INT8, 7 or 6 or other interrupt request.

The clock can be switched to a low speed clock (f256, fC, fOCO4). However, as the CPU operation clock becomes slower, the response speed of the emulator debugger becomes also slower. Furthermore, the internal flash ROM cannot be rewritten during a low speed clock operation. Therefore, do not use software breaks.

Notes on Interrupt Stack:

The E30A emulator uses an interrupt stack of up to 52 bytes. Make sure sufficient space is reserved for the interrupt stack area, with the maximum stack size used by a user program added. If the reserved interrupt stack area is exceeded, the user program may go wild or the emulator may be out of control.

The E30A emulator sets the interrupt stack pointer (ISP) at the beginning address of RAM + 0100h after being reset, and uses that location as a stack area at deassertion of reset.

Note on Interrupts That Cannot be Used by the User Program:

The BRK2 interrupt, single-step interrupt, and DBC interrupt are used by the emulator program. Therefore, make sure the user program does not use these interrupts.

Note on Interrupt Request:

Even if a user program is not being executed (including when run-time debugging is being performed), the peripheral I/O such as a timer keeps running. The interrupt request cannot be accepted, because the emulator disables interrupts. The interrupt request is accepted immediately after the user program execution is started.

Note on Low Voltage Detection Interrupt:

With this product, as the power voltage cannot be changed after powering on the user system, the low voltage detection interrupt cannot be used.

Notes on Using the Watchdog Timer:

A user program can be debugged by using the MCU's internal watchdog timer if so desired. When the watchdog timer is active, the debug monitor program refreshes it as necessary.

Watchdog timers connected external to the MCU cannot be used. Make sure those watchdog timers are disabled while the emulator is in use.

Downloaded from Arrow.com.

Note on the Lock Bits for the MCU Internal Flash ROM:

The lock bits for the blocks whose contents have been altered by downloading a user program or setting software breakpoints are in an unlocked state.

Note on Protect Bit 2 (PRC2):

Protect bit 2 (PRC2) behaves in such a way that when the bit is set to 1 to remove write protect and then a write to any address is executed, the bit is write protected again. Note that the bit cannot be cleared by a software operation while single-stepping a program or manipulating the memory window.

Note on RDY# Input:

Be sure to input "L" to pin RDY# of the user system during the user program executing (when the RUN status LED is lit). Inputting "L" to pin RDY# during the user program stopping 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 executing (when the RUN status LED is lit). Inputting "L" to pin HOLD# during the user program stopping may cause a malfunction of the emulator.

Note on the MCU Used in Debugging:

During debugging, the MCU's internal flash ROM is frequently rewritten as a user program is downloaded or software breakpoints are set or cleared. Do not use any MCU in product once it was used in debugging.

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.

Downloaded from Arrow.com.

Appendix C Notes on Using This Product

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

Notes on the Resources of the MCU Used by the Emulator:

When you are using a data comparison break (event E5 only), the E30A emulator uses several bytes from the beginning of the MCU's internal RAM (8 bytes at the addresses 0400h to 0407h) for data comparison by the debug monitor program. These bytes cannot be used in the user program when using a data comparison break.

The E30A emulator uses approximately 1.2 Kbytes of the internal RAM as needed, for example, to download a user program. Since the memory contents are saved in advance, you do not need to be specifically concerned, except that you cannot specify any area that overlaps the interrupt stack and an area on which DMA operates.

Notes on Power Supply to the User System:

The emulator does not supply power to the user system. Provide power to the user system separately.

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

Note on Downloading Firmware:

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, startup the emulator in maintenance mode and redownload the firmware.

Note on Quitting the Emulator Debugger:

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

Note on Event Settings:

Events are used for breaks, trace, and time measurement in common. For breakpoint settings in the source window, events are given priority in order to reduce the number of times the flash ROM is rewritten. Note that the unused events are used in order of followings.

(Extraction start event E0, end of range event E4, and data comparison event E5 are assigned lower priorities.)

E7 -> E6 -> E3 -> E2 -> E1 -> E5 -> E4 -> E0

Notes on the Software Breaks:

Software breaks execute the tool-only BRK2 command instead of instruction code to generate a BRK interrupt. Be aware that when trace results are referred to in bus display, you will see "BFh" displayed for instruction fetches performed at the addresses that have software breakpoints set.

If a user program is altered before being downloaded, the software breakpoints that have been set may be displaced depending on the content of alteration. After downloading a user program, be sure to check the contents of software breakpoints set.

Notes on the Data Access Trace:

16-bit word data starting from an odd address and 32-bit longword data assigned differently from the 4-byte alignment cannot be traced properly.

Data access event by DMA is not the subject of detection.

Notes on Debug Monitor Program Intervention:

The following functions result in the realtime capability becoming impaired because the debug monitor program intervenes. Also, in order to complete processing of the debug monitor program as fast as possible, do not operate the MCU's internal DMA during the processing.

- Run-time debugging (Referring and modifying the memory during user program execution)
- RAM monitor
- Data comparison break (when a break for data access is set in event E5)

The RAM monitor function should be adjusted for sampling intervals (byte/word data acquisition intervals in ms units) during user program execution. Since the updating time is extended if a wide area is displayed in bytes, we recommend displaying only the necessary area in word size.

Downloaded from Arrow.com.

Note on Trace Mode:

When selecting [Trace priority], the user program execution may be delayed depending on the event occurrence interval because the MCU processes the trace information output prior to others.

When selecting [MCU execution priority], the execution speed of the user program is not affected.

Notes on the Simultaneous Use of Multiple Functions:

Data comparison breaks and RAM monitor cannot be used at the same time. If event E5 is set for a data access break, the RAM monitor functions cannot be specified. If the RAM monitor function needs to be used, event E5 cannot be specified for a data access break.

Be aware that if a run-time debug, RAM monitor, or data comparison break is used during a branch trace in which branch addresses are recorded, the recorded information will include branch information (debug monitor program processing address) which the user program does not generate.

Note on Debugging in the Stop Mode or Wait Mode:

While the user program is running, the emulator communicates with the MCU, for example, to check the operating status of the MCU, or to receive the trace information from the MCU.

The PLL clock divided by the base clock divider is used for communication between the emulator and the MCU. Therefore, when executing the STOP or WAIT instruction, or when temporarily switching to the PLL self-oscillation mode, select the [Do not communicate with MCU while target is executing] checkbox to avoid communication breakdown. The debug function cannot be used because the emulator does not communicate with the MCU when executing the user program. However, it is possible to press the stop button after restored to the base clock specified in the Clock column.

When the emulator is connected, the PLL clock stop bits are ignored. Therefore, the PLL keeps oscillating even when the WAIT or STOP instruction is executed. This means that BCLK is also oscillating, so that control returns from WAIT or STOP by an INT8, 7 or 6 or other interrupt request.

Note on MCU Reset:

Software reset and hardware reset cannot be used during the user program execution. If the MCU is reset during the user program execution, the emulator recovers communication with the MCU and the user program stops in a reset state.

Notes on Debugging in CPU Rewrite Mode:

To debug a program in CPU rewrite mode, select the [Debug the program using the CPU Rewrite Mode] checkbox in the [MCU] tab of the [Init] dialog box. When debugging in CPU rewrite mode is enabled, software breakpoints to the internal ROM area cannot be used.

Do not stop the user program during debugging in CPU rewrite mode. Otherwise, the emulator will be out of control. Furthermore, before executing a user program, be sure to disable automatic updating of the windows to ensure that no memory accesses will occur during execution of the user program. To inspect data after CPU rewrite operation, stop the program after exiting CPU rewrite mode and inspect the data in the memory window or another appropriate window.

Note on Debugging of E^2dataFlash:

When you are debugging E^2dataFlash, do not stop the user program during operating E^2dataFlash. Otherwise, as the operation to E^2dataFlash will be suspended, it may not be processed properly even though the user program is restarted. Furthermore, before executing the user program, be sure to disable automatic updating of the windows to ensure that no accesses to E^2dataFlash will occur during execution of the user program.

Downloaded from Arrow.com.

Appendix D Connection Diagram

Figure D.1 shows a connection diagram of the E30A emulator. This connection diagram shows the interface section.

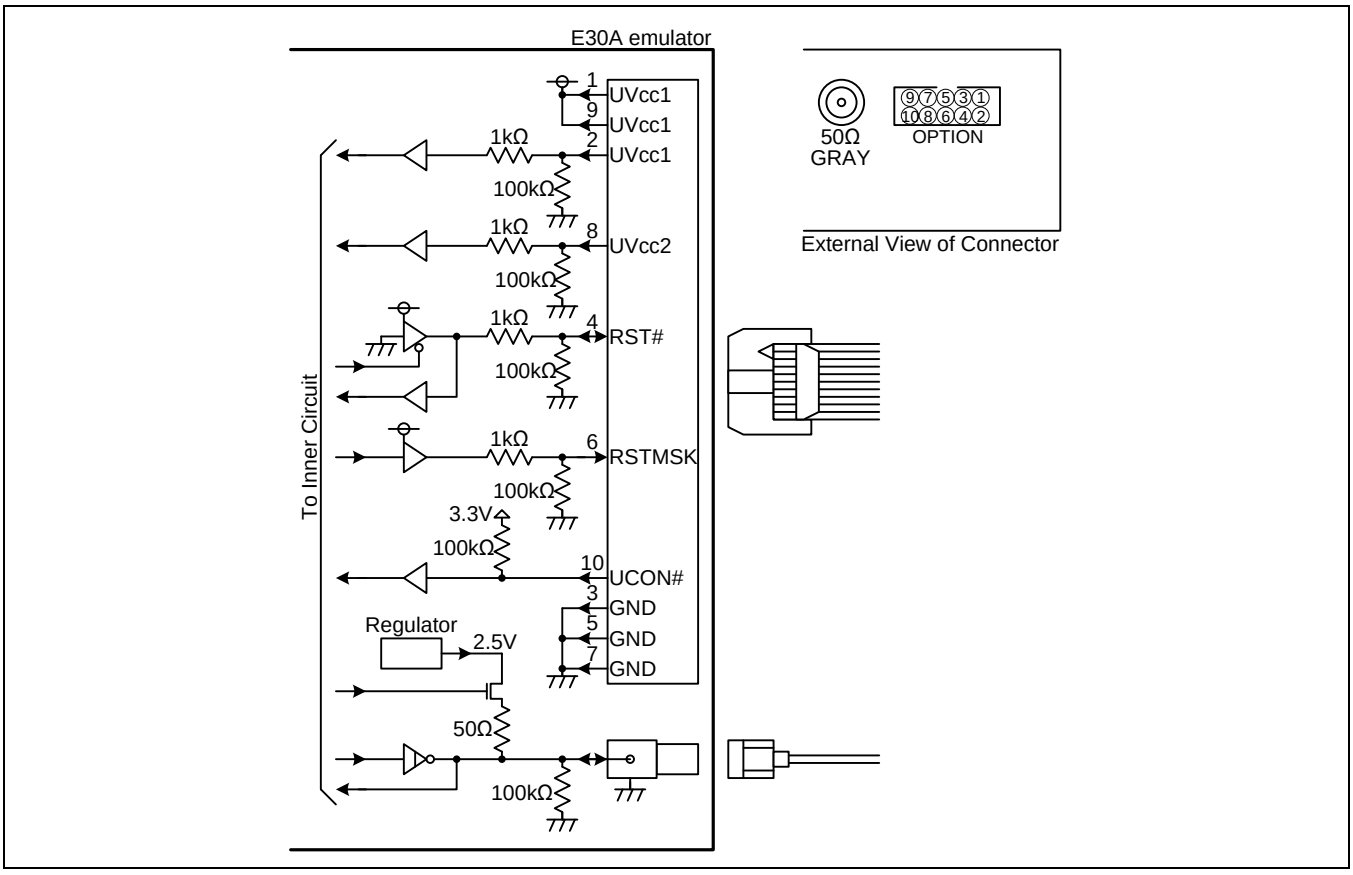


Figure D.1 Connection Diagram of the E30A Emulator's Interface Section

Downloaded from Arrow.com.

Appendix E Signal Assignments of the Connectors

(1) USB I/F connector

Table E.1 lists the signal assignments of the USB I/F connector.

Table E.1 Signal Assignments of the USB I/F Connector

No.	Signal	Description
1	VBUS	5.0-V power supply, unused
2	D-	
3	D+	
4	NC	Unused
5	GND	

(2) Oscillator circuit board connector

Table E.2 lists the signal assignments of the oscillator circuit board connector.

Table E.2 Signal Assignments of the Oscillator Circuit Board Connector

No.	Signal	Description
J1-1	V50	5.0-V power supply
J1-2	GND	
J1-3	OSCCLK	Oscillator circuit input (amplitude: 5.0 V)
J1-4	GND	

(3) Option connector

Table E.3 lists the signal assignments of the option connector.

Table E.3 Signal Assignments of the Option Connector

No.	Signal	Description
1	UVcc1	Vcc1 power supply
2	UVcc1	Vcc1 power supply
3	GND	
4	RST#	Reset signal
5	GND	
6	RSTMSK	Reset mask signal
7	GND	
8	UVcc2	Vcc2 power supply
9	UVcc1	Vcc1 power supply
10	UCON#	Connected to GND

Downloaded from Arrow.com.

Appendix F External Dimensions

(1) E30A Emulator

Figure F.1 shows external dimensions of the E30A emulator.

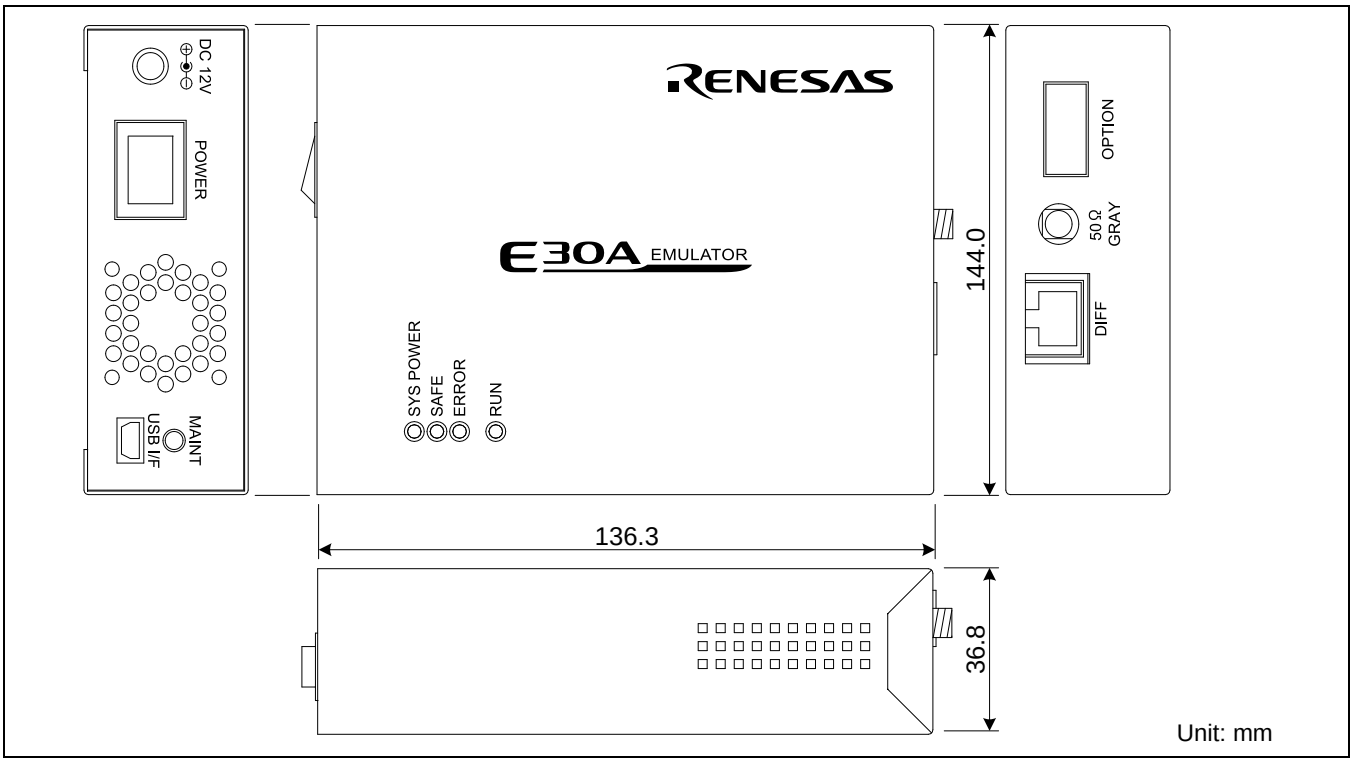
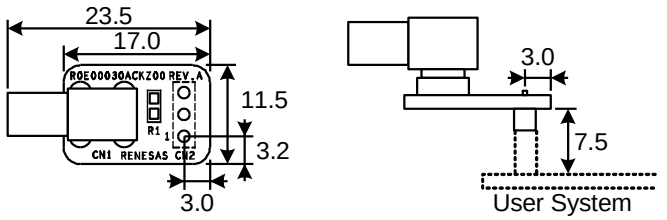


Figure F.1 External Dimensions of the E30A Emulator

Downloaded from Arrow.com.

(2) Coaxial converter board

Figure F.2 shows external dimensions of the coaxial converter board.

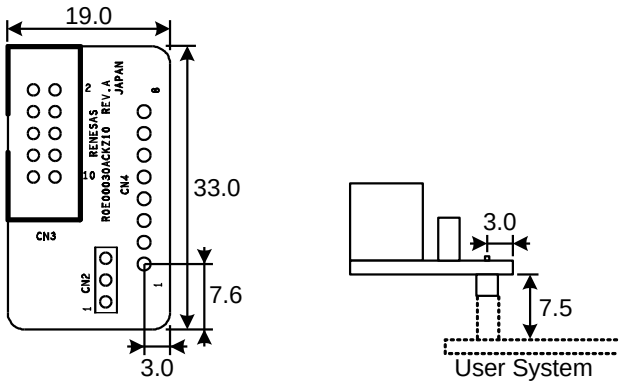


Unit: mm

Figure F.2 External Dimensions of the Coaxial Converter Board

(3) Option board

Figure F.3 shows external dimensions of the option board.



Unit: mm

Figure F.3 External Dimensions of the Option Board

Downloaded from Arrow.com.

Revision History	E30A Emulator R0E00030AKCT00 User's Manual

Rev.	Date	Description	
		Page	Summary
5.01	Mar.15.16	18	Regulatory Compliance Notices was changed.
5.02	Sep.01.21	—	The statement of inclusion of an AC adapter was removed.
		18	Regulatory Compliance Notices was changed.

Downloaded from Arrow.com.

E30A Emulator R0E00030AKCT00
User's Manual

Publication Date: Rev.1.00 Apr.01.08
Rev.5.02 Sep.01.21

Published by: Renesas Electronics Corporation

Downloaded from Arrow.com.

E30A Emulator R0E00030AKCT00
User's Manual

R20UT0422EJ0502

Renesas Electronics Corporation

Downloaded from Arrow.com.