

To our customers,

---

## Old Company Name in Catalogs and Other Documents

---

On April 1<sup>st</sup>, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1<sup>st</sup>, 2010  
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

## Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.
  - “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.
  - “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.
  - “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.



**User's Manual**

# **RX78K0R Ver. 4.30**

**Real-Time Operating System**

**Analysis for CubeSuite**

---

**Target Tool**

**RX78K0R Ver. 4.30**

Document No. U19448EJ1V0UM00 (1st edition)

Date Published December 2008

© NEC Electronics Corporation 2008

Printed in Japan

[MEMO]

# SUMMARY OF CONTENTS

CHAPTER 1 GENERAL ... 12

CHAPTER 2 FUNCTIONS ... 14

APPENDIX A WINDOW REFERENCE ... 21

APPENDIX B MESSAGES ... 68

APPENDIX C INDEX ... 70

**Windows, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.**

**TRON is the abbreviation of "The Real-time Operating system Nucleus."**

**ITRON is the abbreviation of "Industrial TRON."**

**$\mu$ ITRON is the abbreviation of "Micro Industrial TRON."**

**TRON, ITRON, and  $\mu$ ITRON do not refer to any specific product or products.**

**The  $\mu$ ITRON4.0 Specification is an open real-time kernel specification developed by TRON Association.**

**The  $\mu$ ITRON4.0 Specification document can be obtained from the TRON Association web site (<http://www.assoc.tron.org>).**

**The copyright of the  $\mu$ ITRON4.0 Specification document belongs to TRON Association.**

• **The information in this document is current as of December, 2008. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**

• No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.

• NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.

• Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

• While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.

• NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.

"Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots.

"Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).

"Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.

(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

[MEMO]

## INTRODUCTION

<b>Target Readers</b>	This manual is intended for users who wish to design and develop application systems using the 78K0R microcontrollers.																						
<b>Purpose</b>	This manual's purpose is to help the user understand the functions and the operation method of the RX78K0R.																						
<b>Organization</b>	This manual is organized as follows.  <ul style="list-style-type: none"><li>• GENERAL</li><li>• FUNCTIONS</li></ul>																						
<b>How to Use this Manual</b>	<p>This manual assumes that the reader has general knowledge of microcontrollers, C language, assembly language, and debugging as well as basic knowledge of the Windows operation method.</p> <p>To learn about the hardware functions and instruction functions of the 78K0R microcontrollers:</p> <p>→ Read the user's manual for each product.</p>																						
<b>Conventions</b>	<table><tr><td>&lt; &gt;:</td><td>Indicates a window or dialog box title.</td></tr><tr><td>[ ]:</td><td>Indicates a menu.</td></tr><tr><td><span style="border: 1px solid black; padding: 0 2px;">text</span>:</td><td>Indicates a button in a window or dialog box.</td></tr><tr><td>&lt;&lt; &gt;&gt;:</td><td>Indicates a character string displayed in a window or dialog box.</td></tr><tr><td>Data significance:</td><td>Higher digits on the left and lower digits on the right</td></tr><tr><td>Memory map addresses:</td><td>Higher addresses on the top and lower addresses on the bottom</td></tr><tr><td>Active low representation:</td><td><math>\overline{\text{xxx}}</math> (overscore over pin or signal name)</td></tr><tr><td><b>Note:</b></td><td>Footnote for item marked with Note in the text</td></tr><tr><td><b>Caution:</b></td><td>Information requiring particular attention</td></tr><tr><td>Numerical representation:</td><td>Binary...XXXX or XXXXB Decimal...XXXX Hexadecimal...0xXXXX</td></tr><tr><td>Prefix indicating power of 2 (address space, memory capacity)</td><td>K (Kilo) <math>2^{10} = 1024</math> M (Mega) <math>2^{20} = 1024^2</math></td></tr></table>	< >:	Indicates a window or dialog box title.	[ ]:	Indicates a menu.	<span style="border: 1px solid black; padding: 0 2px;">text</span> :	Indicates a button in a window or dialog box.	<< >>:	Indicates a character string displayed in a window or dialog box.	Data significance:	Higher digits on the left and lower digits on the right	Memory map addresses:	Higher addresses on the top and lower addresses on the bottom	Active low representation:	$\overline{\text{xxx}}$ (overscore over pin or signal name)	<b>Note:</b>	Footnote for item marked with Note in the text	<b>Caution:</b>	Information requiring particular attention	Numerical representation:	Binary...XXXX or XXXXB Decimal...XXXX Hexadecimal...0xXXXX	Prefix indicating power of 2 (address space, memory capacity)	K (Kilo) $2^{10} = 1024$ M (Mega) $2^{20} = 1024^2$
< >:	Indicates a window or dialog box title.																						
[ ]:	Indicates a menu.																						
<span style="border: 1px solid black; padding: 0 2px;">text</span> :	Indicates a button in a window or dialog box.																						
<< >>:	Indicates a character string displayed in a window or dialog box.																						
Data significance:	Higher digits on the left and lower digits on the right																						
Memory map addresses:	Higher addresses on the top and lower addresses on the bottom																						
Active low representation:	$\overline{\text{xxx}}$ (overscore over pin or signal name)																						
<b>Note:</b>	Footnote for item marked with Note in the text																						
<b>Caution:</b>	Information requiring particular attention																						
Numerical representation:	Binary...XXXX or XXXXB Decimal...XXXX Hexadecimal...0xXXXX																						
Prefix indicating power of 2 (address space, memory capacity)	K (Kilo) $2^{10} = 1024$ M (Mega) $2^{20} = 1024^2$																						

**Related Documents**

Please use the following documents in conjunction with this manual.  
The related documents listed below may include preliminary versions. However, preliminary versions are not marked as such.

**Documents Related to 78K0R microcontrollers Development Tools (User's Manuals)**

Document Name		Document No.
RX Series	Start for CubeSuite	U19428E
	Message for CubeSuite	U19433E
RX78K0R Ver.4.30	Coding for CubeSuite	U19443E
	Debug for CubeSuite	U19446E
	Analysis for CubeSuite	This document
	Internal Structure for CubeSuite	U19453E
CubeSuite Integrated Development Environment	Start	U19549E
	Programming	U19390E
	Message	U19550E
	78K0R Coding	U19382E
	78K0R Build	U19385E
	78K0R Debug	U19388E
	78K0R Design	U19379E

# CONTENTS

## CHAPTER 1 GENERAL ... 12

1.1 Summary ... 12

1.2 Features ... 13

## CHAPTER 2 FUNCTIONS ... 14

2.1 Trace Form ... 14

2.2 Trace Data ... 14

2.2.1 Collecting positions and collected data ... 14

2.2.2 Time accuracy ... 15

2.3 Debugging Procedure ... 16

## APPENDIX A WINDOW REFERENCE ... 21

A.1 Description ... 21

## APPENDIX B MESSAGES ... 68

B.1 Overview ... 68

B.2 Error Messages ... 68

## APPENDIX C INDEX ... 70

# LIST OF FIGURES

Figure No.	Title and Page
2-1	Starting AZ78K0R ... 16
2-2	Downloading Load Module ... 16
2-3	Switching AZ Trace Mode ... 17
2-4	Display Example of AZ:Analyze Window ... 18
2-5	Display Example of Pattern Search Dialog Box ... 18
2-6	Display Example of AZ:Cpu Window ... 19
2-7	Display Example of AZ:Trace View Window ... 20
2-8	Display Example of Trace Search Dialog Box ... 20
A-1	Relationship of Windows and Dialog Boxes ... 22
A-2	AZ78K0R Window ... 23
A-3	Statusbar (AZ78K0R Window) ... 25
A-4	Open/Save As Dialog Box (When Loading) ... 28
A-5	Open/Save As Dialog Box (When Saving) ... 28
A-6	AZ:Analyze Window ... 32
A-7	AZ:Analyze Window (Simple Mode) ... 42
A-8	AZ:Analyze Window (Standard Mode) ... 43
A-9	AZ:Analyze Window (Detailed Mode) ... 44
A-10	AZ:Analyze Window (When Equal Mode Is Not Specified) ... 46
A-11	AZ:Analyze Window (When Equal Mode Is Specified) ... 46
A-12	Pattern Search Dialog Box ... 48
A-13	Difference in Search Targets ... 50
A-14	AZ:Cpu Window ... 52
A-15	AZ:Trace View Window ... 57
A-16	How to Read the List ... 61
A-17	Trace Search Dialog Box ... 62
A-18	Specification Example 1 (Search for Interrupt) ... 64
A-19	Specification Example 2 (Search for Location at Which sta_tsk Was Issued) ... 64
A-20	Specification Example 3 (Search for Location at Which Service Call Was Issued for TSK_INIT) ... 65
A-21	About Dialog Box ... 66
A-22	AZ:Error Dialog Box ... 67
B-1	Error Information Output Format ... 68

# LIST OF TABLES

Table No.	Title and Page
2-1	Resources Used by AZ78K0R ... 14
2-2	Collected Trace Data and Collecting Positions ... 14
A-1	List of Windows and Dialog Boxes of AZ78K0R ... 21
A-2	Sort Buttons in AZ:Analyze Window ... 33
A-3	Display Scale of Execution Transition Map ... 34
A-4	Simple Search Buttons in AZ:Analyze Window ... 35
A-5	Details of Information in AZ:Analyze Window ... 35
A-6	Marks Displayed in AZ:Analyze Window (Simple Mode) ... 42
A-7	Marks Displayed in AZ:Analyze Window (Standard Mode) ... 43
A-8	Marks Displayed in AZ:Analyze Window (Detailed mode) ... 44
A-9	Types of Search Mode ... 49
A-10	Search Conditions (When Task Switch Is Selected) ... 49
A-11	Search Conditions (When a Service Call (Call, Return, Error) Is Selected) ... 50
A-12	Processing Program Types ... 53
A-13	Colors in Bar Graph ... 54
A-14	Processing Program Types ... 58
A-15	Event Types ... 58
A-16	Search Item Types ... 63
B-1	Error Information List ... 68

# CHAPTER 1 GENERAL

This chapter describes an overview of analysis tool (AZ78K0R) that CubeSuite provides.

## 1.1 Summary

As the performance of microprocessors has increased, application programs have grown in scale and complexity.

With conventional debuggers, theoretical debugging of such application programs is simple, but time-related analysis is not. It is difficult and takes a very long time, for example, to analyze errors such as those caused by incorrect processing timing, or to evaluate the performance of the entire system.

To solve these problems, NEC Electronics Corporation has developed powerful microprocessors such as the 78K0R Microcontrollers. NEC Electronics Corporation also provides the analysis tool (AZ78K0R) to support the quantitative performance analysis of programs.

AZ78K0R is a performance analysis tool for examining the execution transition statuses and the CPU usage of processing programs that embed RX78K0R real-time operating system for 78K0R. Being connected with CubeSuite, AZ78K0R achieves a function for collecting data of tracing the event occurrences (issuance of service calls, occurrence of interrupts, etc.) and presenting the trace data graphically.

AZ78K0R therefore allows the user to readily analyze the execution transition statuses and the CPU usage of processing programs.

## 1.2 Features

AZ78K0R has the following features:

- Display of execution transition statuses

The graphically displayed the execution transition status of the processing program in which RX78K0R is embedded (horizontal axis = time, vertical axis = task name, etc.) permits analysis of execution transition statuses, such as task switching caused by service call issuance and transferring the control to the interrupt handler caused by the occurrence of interrupts. Since all accesses to an object (such as semaphore or eventflag) are marked upon issuance of a service call from a processing program, the usage of objects can also be checked.

- Display of CPU usage

The CPU usage (total execution time, code coverage, etc.) of a processing program executed within a specified section can be displayed, which enables the quantitative performance analysis for processing programs.

- Linking with CubeSuite

AZ78K0R can open the Editor panel, Disassemble panel, and Memory panel of CubeSuite, which enables quick identification of locations where a problem has occurred.

# CHAPTER 2 FUNCTIONS

This chapter describes main functions of AZ78K0R.

## 2.1 Trace Form

AZ78K0R supports the following trace form.

- Hard trace form

Using the trace function provided by the debug tool that is connected to CubeSuite, AZ78K0R collects trace data and stores it to the trace memory of the debug tool.

Using this form, trace data can therefore be collected without modifying the processing program code.

[Note] AZ78K0R uses the following resources of the debug tool, when switching the AZ trace mode to the ON state.

**Table 2-1. Resources Used by AZ78K0R**

Resource Name	Required Quantity
Point Trace event (for write access)	1

## 2.2 Trace Data

### 2.2.1 Collecting positions and collected data

The following lists the trace data to be collected and positions where they are collected.

**Table 2-2. Collected Trace Data and Collecting Positions**

Collecting Positions	Collected Data
Service call entry	Time Service call function code ID of object subject to execution by service call Service call issuance address
Service call exit	Time Service call return value
Interrupt handler entry	Time Address at which execution returns from interrupt handler
Interrupt handler exit	Time

Collecting Positions	Collected Data
Occurrence of task switching	Time ID of a task to which the operation will move (or ID that indicates idle routine is entered)
Task entry	Time ID of task to be activated Task activation address

Note that the following items cannot be detected as trace data for AZ78K0R.

- Address at which ext\_tsk is issued
- Entry/exit of reset, NMI, exception (software exception, exception trap)
- Entry/exit of maskable interrupts whose interrupt handler has not been registered
- Boot processing entry/exit
- Initialization routine entry/exit
- Cyclic handler entry/exit

Since RX78K0R assigns the same function code to service call names to which an "i" is prefixed/not prefixed listed below, AZ78K0R cannot identify them.

AZ78K0R therefore handles these service calls without an "i" being prefixed.

- sta\_tsk/ista\_tsk
- chg\_pri/ichg\_pri
- can\_wup/ican\_wup
- sus\_tsk/isus\_tsk
- rsm\_tsk/irsm\_tsk
- frsm\_tsk/ifrsm\_tsk

## 2. 2. 2 Time accuracy

The accuracy of the time collected as trace data depends on the setting of the [Rate of frequency division of trace time tag] property in the [Trace] category of the [Debug Tool Settings] tab in the Property panel of CubeSuite.

If the system clock is set to the STOP or IDLE mode, however, AZ78K0R cannot collect an accurate time, and the collected value is therefore not guaranteed. (This is because AZ78K0R calculates the time taken by processing program execution, based on the system clock.)

[Note] When using the simulator as debug tool, specify [No] with the [Accumulate trace time] property in the [Trace] category on the [Debug Tool Settings] tab in the Property panel of CubeSuite.

Unless this property is set to [No], the correct time information cannot be obtained.

## 2.3 Debugging Procedure

This section describes the procedure for debugging using AZ78K0R.

(1) Starting CubeSuite

Start CubeSuite and then connect the debug tool to be used to it (select the [Debug] menu -> [Connect to Debug Tool] in the Main window of CubeSuite).

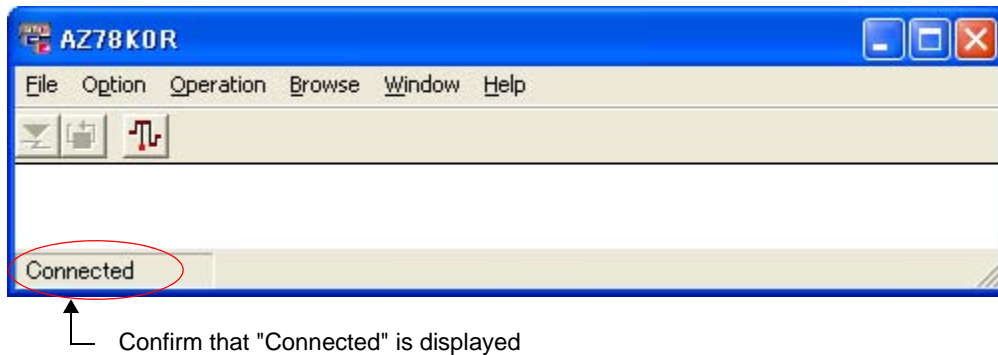
(2) Starting AZ78K0R

Start AZ78K0R to be used (select the [View] menu -> [Real-time OS] -> [Performance Analyzer] in the Main window of CubeSuite).

After AZ78K0R is started, the following [AZ78K0R window](#) appears.

At this time, make sure that the message "Connected" is displayed on the statusbar (Connection status area) in the [AZ78K0R window](#).

**Figure 2-1. Starting AZ78K0R**

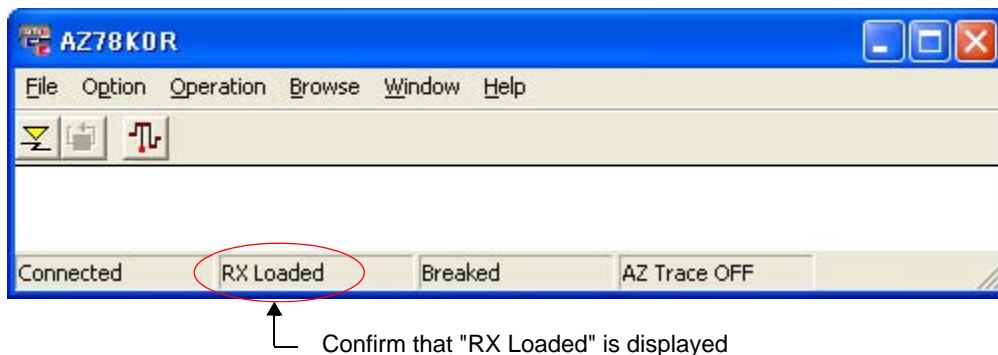


(3) Downloading a load module

Download the load module that has been linked with RX78K0R into the debug tool.

At this time, make sure that the message "RX Loaded" is displayed on the statusbar (Load module status area) in the [AZ78K0R window](#).

**Figure 2-2. Downloading Load Module**

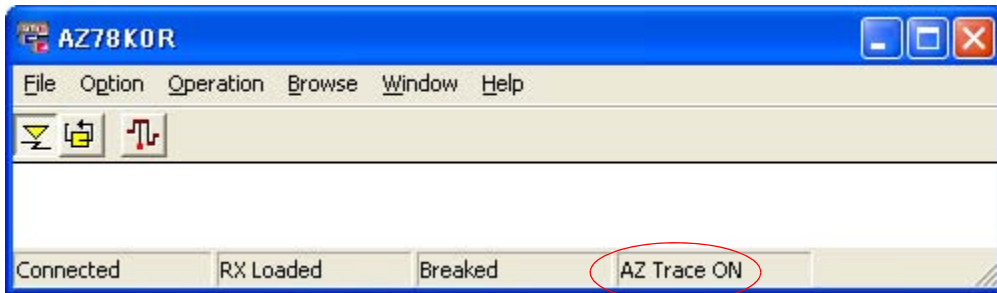


## (4) Switching the AZ trace mode

Click the  button in the [AZ78K0R window](#) to turn on AZ trace mode.

At this time, make sure that the message "AZ Trace ON" is displayed on the statusbar (AZ trace mode status area) in the [AZ78K0R window](#).

**Figure 2-3. Switching AZ Trace Mode**



↑ Confirm that "AZ Trace ON" is displayed

## (5) Execution of processing program


Run the processing program on CubeSuite, collection of trace data then starts.

## (6) Stop of processing program

Stop execution of the processing program on CubeSuite. If a breakpoint has been set, wait until the processing program breaks.

Collection of trace data then ends.

## (7) Loading of trace data

Click the  button in the [AZ78K0R window](#) to load trace data collected for AZ78K0R.

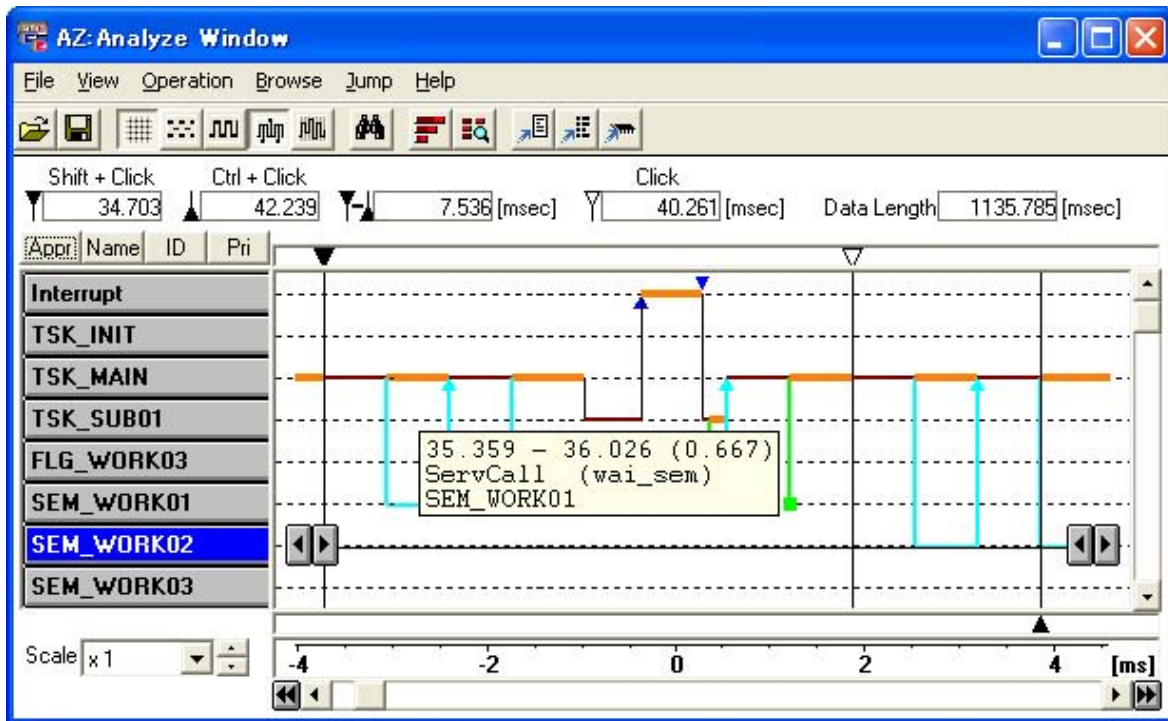
(8) Verification in [AZ:Analyze Window](#)

Click the  button in the [AZ78K0R window](#) to open the [AZ:Analyze Window](#).

In the [AZ:Analyze Window](#) and windows opened via the the [AZ:Analyze Window](#), analysis related to time, such as bugs caused by processing timing and evaluation of entire system performance.

[Note] For how to read the execution transition map displayed in this window, refer to "[How to read execution transition map](#)" and "[How to verify execution transition map](#)".

Figure 2-4. Display Example of AZ:Analyze Window



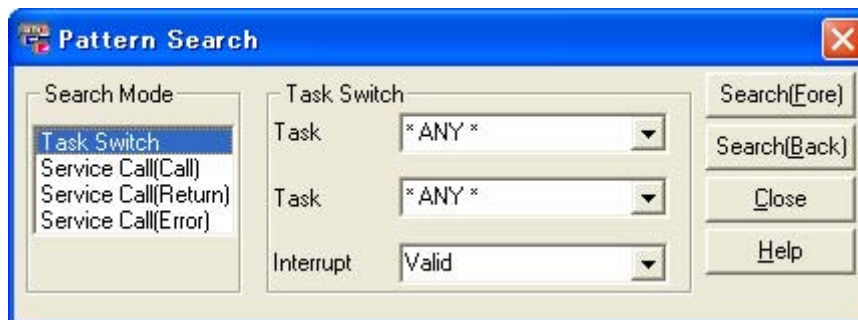
- (9) Verification in [Pattern Search dialog box](#)

Click the  button in the [AZ:Analyze Window](#) to open the [Pattern Search dialog box](#).

In this window, the point at which a specific event occurred can be searched for, based on the execution transition map displayed in the [AZ:Analyze Window](#).

[Note] For how to search for the point at which a specific event occurred, refer to "[How to search](#)".

Figure 2-5. Display Example of Pattern Search Dialog Box



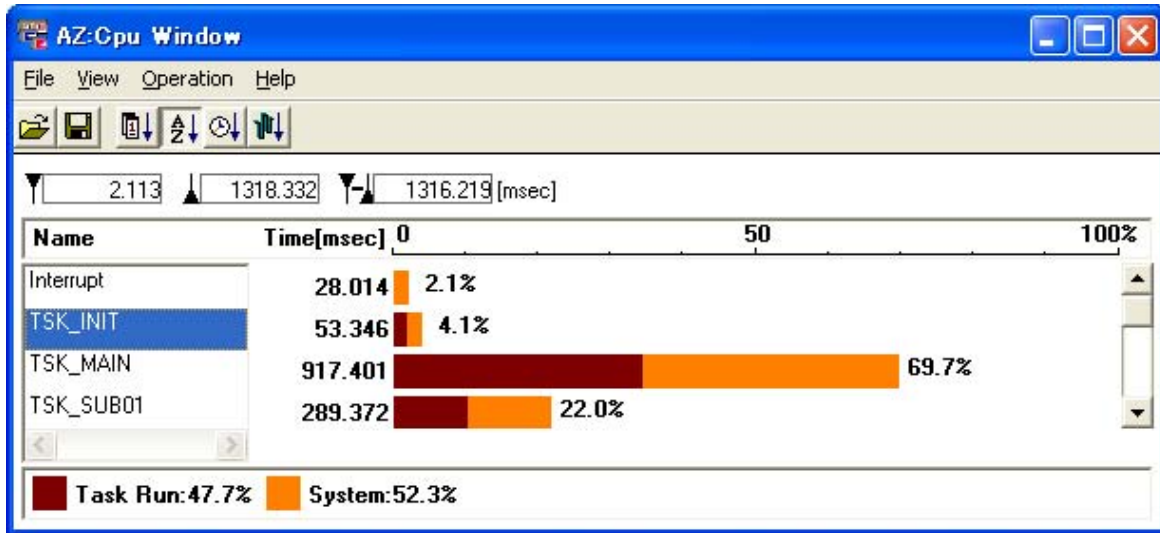
(10) Verification in [AZ:Cpu Window](#)

Click the  button in the [AZ:Analyze Window](#) to open the [AZ:Cpu Window](#).

In this window, the CPU usage in a section between the up cursor and down cursor in the [AZ:Analyze Window](#) can be checked.

[Note] For how to count the CPU usage, refer to "[How to search](#)".

Figure 2-6. Display Example of AZ:Cpu Window

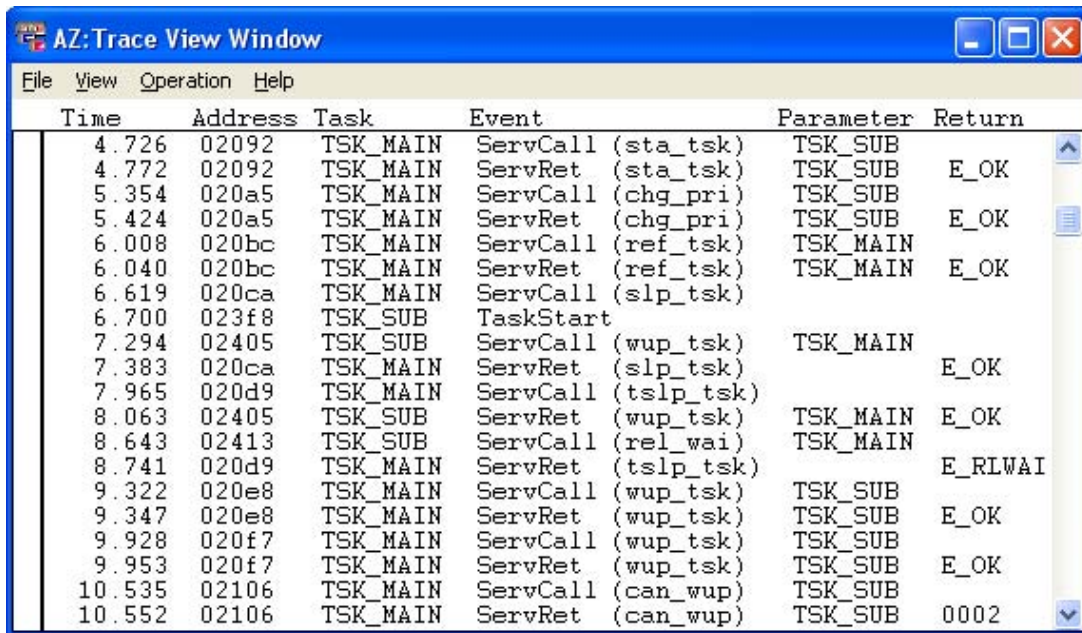
(11) Verification in [AZ:Trace View Window](#)

Click the  button in the [AZ:Analyze Window](#) to open the [AZ:Trace View Window](#).

In this window, information obtained from the execution transition map in the [AZ:Analyze Window](#) can be viewed in list form.

[Note] For how to read the list displayed in this window, refer to "[How to read the list](#)".

Figure 2-7. Display Example of AZ:Trace View Window



Time	Address	Task	Event	Parameter	Return
4.726	02092	TSK_MAIN	ServCall (sta_tsk)	TSK_SUB	
4.772	02092	TSK_MAIN	ServRet (sta_tsk)	TSK_SUB	E_OK
5.354	020a5	TSK_MAIN	ServCall (chg_pri)	TSK_SUB	
5.424	020a5	TSK_MAIN	ServRet (chg_pri)	TSK_SUB	E_OK
6.008	020bc	TSK_MAIN	ServCall (ref_tsk)	TSK_MAIN	
6.040	020bc	TSK_MAIN	ServRet (ref_tsk)	TSK_MAIN	E_OK
6.619	020ca	TSK_MAIN	ServCall (slp_tsk)		
6.700	023f8	TSK_SUB	TaskStart		
7.294	02405	TSK_SUB	ServCall (wup_tsk)	TSK_MAIN	
7.383	020ca	TSK_MAIN	ServRet (slp_tsk)		E_OK
7.965	020d9	TSK_MAIN	ServCall (tslp_tsk)		
8.063	02405	TSK_SUB	ServRet (wup_tsk)	TSK_MAIN	E_OK
8.643	02413	TSK_SUB	ServCall (rel_wai)	TSK_MAIN	
8.741	020d9	TSK_MAIN	ServRet (tslp_tsk)		E_RLWAI
9.322	020e8	TSK_MAIN	ServCall (wup_tsk)	TSK_SUB	
9.347	020e8	TSK_MAIN	ServRet (wup_tsk)	TSK_SUB	E_OK
9.928	020f7	TSK_MAIN	ServCall (wup_tsk)	TSK_SUB	
9.953	020f7	TSK_MAIN	ServRet (wup_tsk)	TSK_SUB	E_OK
10.535	02106	TSK_MAIN	ServCall (can_wup)	TSK_SUB	
10.552	02106	TSK_MAIN	ServRet (can_wup)	TSK_SUB	0002

## (12) Verification in Trace Search dialog box

Click the [View] menu -> [Find...] in the AZ:Trace View Window to open the Trace Search dialog box.

In this window, specific information (trace data) can be searched for from the list displayed in the AZ:Trace View Window.

[Note] For how to search for specific information (trace data), refer to "How to search".

Figure 2-8. Display Example of Trace Search Dialog Box



Trace Search

Search data of AZ trace!

Task Interrupt

Event sta\_tsk

Parameter TSK\_INIT

Search(Fore)

Search(Back)

Cancel

Help

# APPENDIX A WINDOW REFERENCE

Appendix A provides detailed explanations of windows and dialog boxes used for debugging with AZ78K0R.

## A. 1 Description

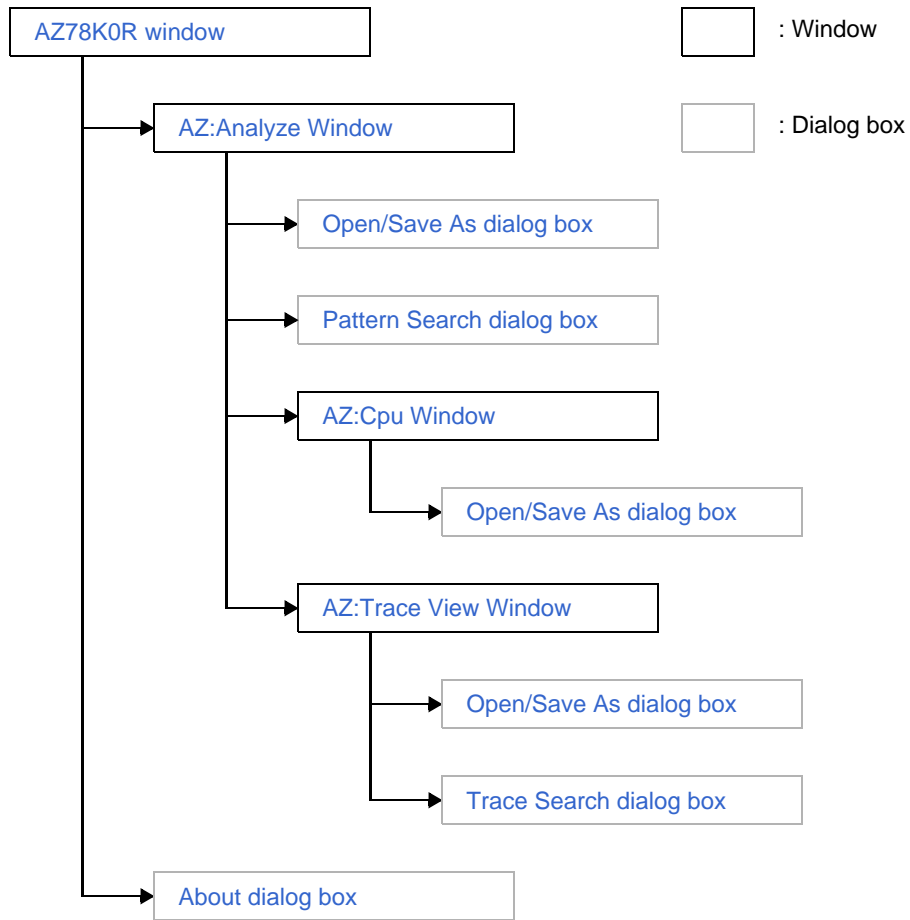
The following shows the list of windows and dialog boxes of AZ78K0R.

**Table A-1. List of Windows and Dialog Boxes of AZ78K0R**

Window/Dialog Box Name	Function
<a href="#">AZ78K0R window</a>	Central window for using the functions provided by AZ78K0R
<a href="#">Open/Save As dialog box</a>	Specifies the file name when loading a file that contains information to be displayed in the <a href="#">AZ:Analyze Window</a> and <a href="#">AZ:Cpu Window</a> , or when saving information displayed in the <a href="#">AZ:Analyze Window</a> , <a href="#">AZ:Cpu Window</a> and <a href="#">AZ:Trace View Window</a> as a file.
<a href="#">AZ:Analyze Window</a>	Displays information obtained from the trace data loaded into the <a href="#">AZ78K0R window</a> , in the form of an execution transition map.
<a href="#">Pattern Search dialog box</a>	Searches for the point at which a specific event occurred, based on the execution transition map displayed in the <a href="#">AZ:Analyze Window</a> .
<a href="#">AZ:Cpu Window</a>	Displays the CPU usage in a section between the up cursor and down cursor in the <a href="#">AZ:Analyze Window</a> .
<a href="#">AZ:Trace View Window</a>	Lists information obtained from the execution transition map in the <a href="#">AZ:Analyze Window</a> .
<a href="#">Trace Search dialog box</a>	Searches for specific information (trace data) based on the list displayed in the <a href="#">AZ:Trace View Window</a> .
<a href="#">About dialog box</a>	Displays the version information of AZ78K0R.
<a href="#">AZ:Error dialog box</a>	Displays the error information of AZ78K0R.

The following shows the relationship of the windows and dialog boxes.

**Figure A-1. Relationship of Windows and Dialog Boxes**

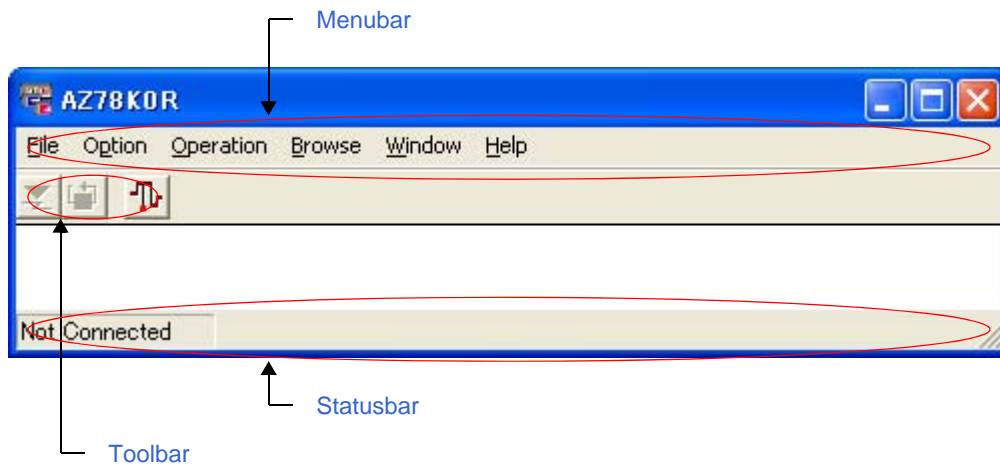


## AZ78K0R window

This is the central window for using the functions provided by AZ78K0R.

This window opens automatically when AZ78K0R is started.

Figure A-2. AZ78K0R Window




This section describes the following items:

- Menubar
- Toolbar
- Statusbar
- Caution

### Menubar




- (1) [File] menu

Exit	Terminates AZ78K0R. The function of this item is same as that of the  button.
------	---


- (2) [Option] menu

Tool Bar	Switches displaying and hiding the toolbar (default: displayed).
Status Bar	Switches displaying and hiding the statusbar (default: displayed).

## (3) [Operation] menu

AZ Trace ON	<p>Turns on the AZ trace mode.</p> <p>Trace data is collected into the trace memory of the debug tool by running a processing program while the AZ trace mode is ON.</p> <p>The function of this item is same as that of the  button.</p>
AZ Trace OFF	<p>Turns off the AZ trace mode.</p> <p>Trace data is not collected into the trace memory of the debug tool if a processing program is executed while the AZ trace mode is OFF.</p> <p>The function of this item is same as that of the  button.</p>
Upload	<p>Loads trace data stored in the trace memory.</p> <p>The AZ trace mode is automatically turned off when loading of trace data is finished.</p> <p>This menu is unavailable if trace data has not been collected.</p> <p>The function of this item is same as that of the  button.</p>

## (4) [Browse] menu

Analyze...	<p>Opens the <a href="#">AZ:Analyze Window</a> in Active mode.</p> <p>If loading of trace data has been finished in this window, the execution transition map is displayed in the corresponding window.</p> <p>The function of this item is same as that of the  button.</p>
------------	---

## (5) [Window] menu




Close All	Closes windows and dialog boxes other than this window.
-----------	---

## (6) [Help] menu

This Window	Displays the help window for this window.
Help Topics	Opens the online help, with the [Search] tab displayed.
About...	<p>Opens the <a href="#">About dialog box</a>.</p> <p>The version information of AZ78K0R is displayed.</p>

## Toolbar

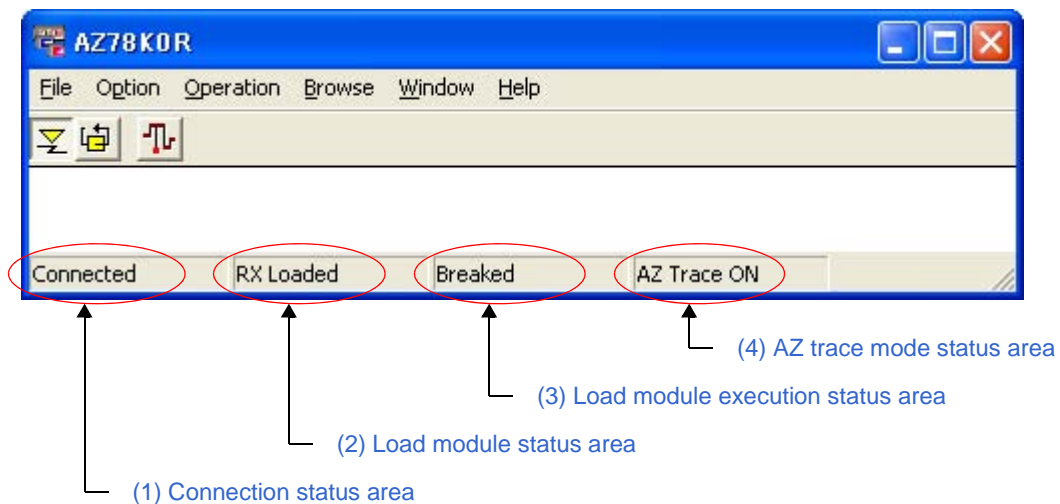
The toolbar consists of buttons that can execute frequently used menu items with a single action.

Button	Function
	Switches the AZ trace mode. Trace data is collected into the trace memory of the debug tool by running a processing program while the AZ trace mode is ON. Trace data is not collected into the trace memory of the debug tool if a processing program is executed while the AZ trace mode is OFF. Same operation an [Operation] menu -> [AZ Trace ON], or [Operation] menu -> [AZ Trace OFF].
	Loads trace data stored in the trace memory. The AZ trace mode is automatically turned off when loading of trace data is finished. This button is unavailable if trace data has not been collected. Same operation an [Operation] menu -> [Upload].
	Opens the <a href="#">AZ:Analyze Window</a> in Active mode. If loading of trace data has been finished in this window, the execution transition map is displayed in the corresponding window. Same operation an [Browse] menu -> [Analyze...].

## Statusbar

The following explains areas in the statusbar.

**Figure A-3. Statusbar (AZ78K0R Window)**



## (1) Connection status area

This area indicates the status of connection with CubeSuite.

Connected	Connected to CubeSuite.
Not Connected	Not connected to CubeSuite.

## (2) Load module status area

This area indicates the state of the load module to be loaded onto the debug tool.

This area is not displayed if AZ78K0R is not connected to CubeSuite.

RX Loaded	A load module linked with RX78K0R has been downloaded.
RX None	No load modules linked with RX78K0R have been downloaded.

## (3) Load module execution status area

This area indicates the state of the program execution.

This area is not displayed if AZ78K0R is not connected to CubeSuite

Running	Status of program being execution.
Breaked	Status of program operation undergoing break.

## (4) AZ trace mode status area

This area indicates the current state of AZ trace mode.

This area is not displayed when AZ78K0R is not connected with CubeSuite, or if no load modules linked with RX78K0R have been downloaded.

AZ Trace ON	Status where AZ trace ON has been set.
AZ Trace OFF	Status where AZ trace OFF has been set.

## **Caution**


- When switching the AZ trace mode to the ON state, a load module linked with RX78K0R must already be downloaded to the debug tool ([RX Loaded] is displayed in the statusbar).
- Event setting for the debug tool is performed when the AZ trace mode is switched to the ON state. Refer to "2. 1 Trace Form" for details on events on the debug tool side, which are required for switching the AZ trace mode.

## Open/Save As dialog box


This dialog box is used to specify the file name when loading a file that contains information to be displayed in the [AZ:Analyze Window](#) and [AZ:Cpu Window](#), or when saving information displayed in the [AZ:Analyze Window](#), [AZ:Cpu Window](#) and [AZ:Trace View Window](#) as a file.

This dialog box can be opened from the each window ([AZ:Analyze Window](#), [AZ:Cpu Window](#), [AZ:Trace View Window](#)) by any of the following:

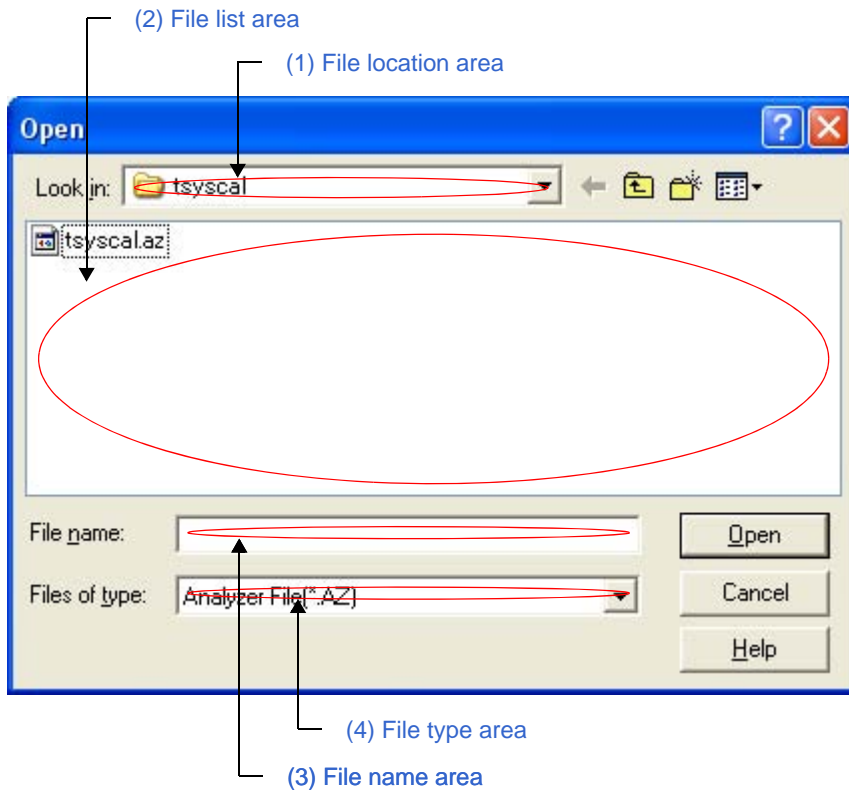
[When loading]

- Select the [File] menu -> [Open...].
- Click the  button on the toolbar.
- Press the [Alt], [F] and [O] keys in that order.
- Press the [Ctrl] + [O] keys at the same time.

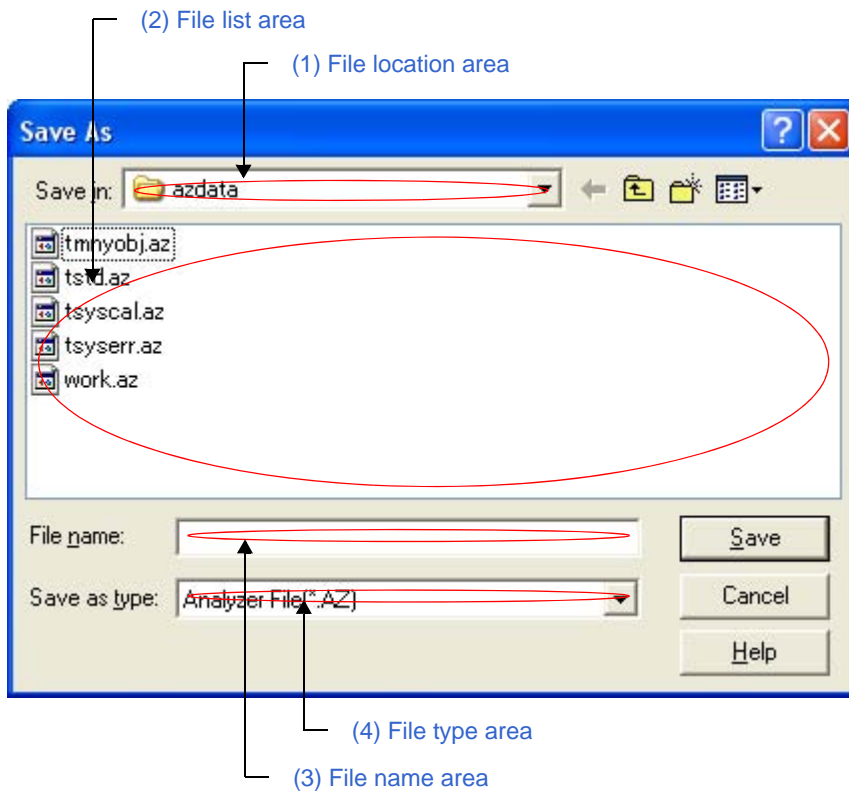
[When saving]

- Select the [File] menu -> [Save...].
- Click the  button on the toolbar.
- Press the [Alt], [F] and [S] keys in that order.
- Press the [Ctrl] + [S] keys at the same time.

**Figure A-4. Open/Save As Dialog Box (When Loading)**



**Figure A-5. Open/Save As Dialog Box (When Saving)**



This section describes the following items:

- [Explanation of each area](#)
- [Function buttons](#)
- [Restored file information](#)

## **Explanation of each area**

(1) File location area

This area is used to select the folder where a target file is stored or is to be stored.

(2) File list area

This area lists the files that match the conditions selected in the file location area and file type area.

(3) File name area

This area is used to specify the name of a target file.

(4) File type area

This area is used to select the type of files to be displayed in the file list area.

The default type displayed in this area varies depending on the window from which the file is called, as shown below.

Window Name	File of Type
<a href="#">AZ:Analyze Window</a>	Binary (.az)
<a href="#">AZ:Cpu Window</a>	CSV (.azc)
<a href="#">AZ:Trace View Window</a>	CSV (.azt)

## **Function buttons**

Button	Function
Open/Save	Loads the file that matches the conditions specified in this dialog box. Saves the information displayed in the window into the file that matches the conditions specified in this dialog box.
Cancel	Closes this dialog box.
Help	Displays the help window for this dialog box.

## **Restored file information**

Handling of files loaded in this dialog box varies depending on the window from which the file is called, as shown below.

- [AZ:Analyze Window](#)

Information of the file loaded into the [AZ:Analyze Window](#), which called the file, will be restored. To maintain the previous information, switch to the Hold mode the [AZ:Analyze Window](#) that has information to be maintained, open another [AZ:Analyze Window](#), and then load the file from the window in the Hold mode.

- [AZ:Cpu Window](#)

A new [AZ:Cpu Window](#) opens, and information of the file loaded into the window is restored. The [AZ:Cpu Window](#) that called the file therefore maintains the previous information.

## AZ:Analyze Window

This window displays information obtained from the trace data loaded into the [AZ78K0R window](#), in the form of an execution transition map.

This dialog box can be opened from the [AZ78K0R window](#) by any of the following:


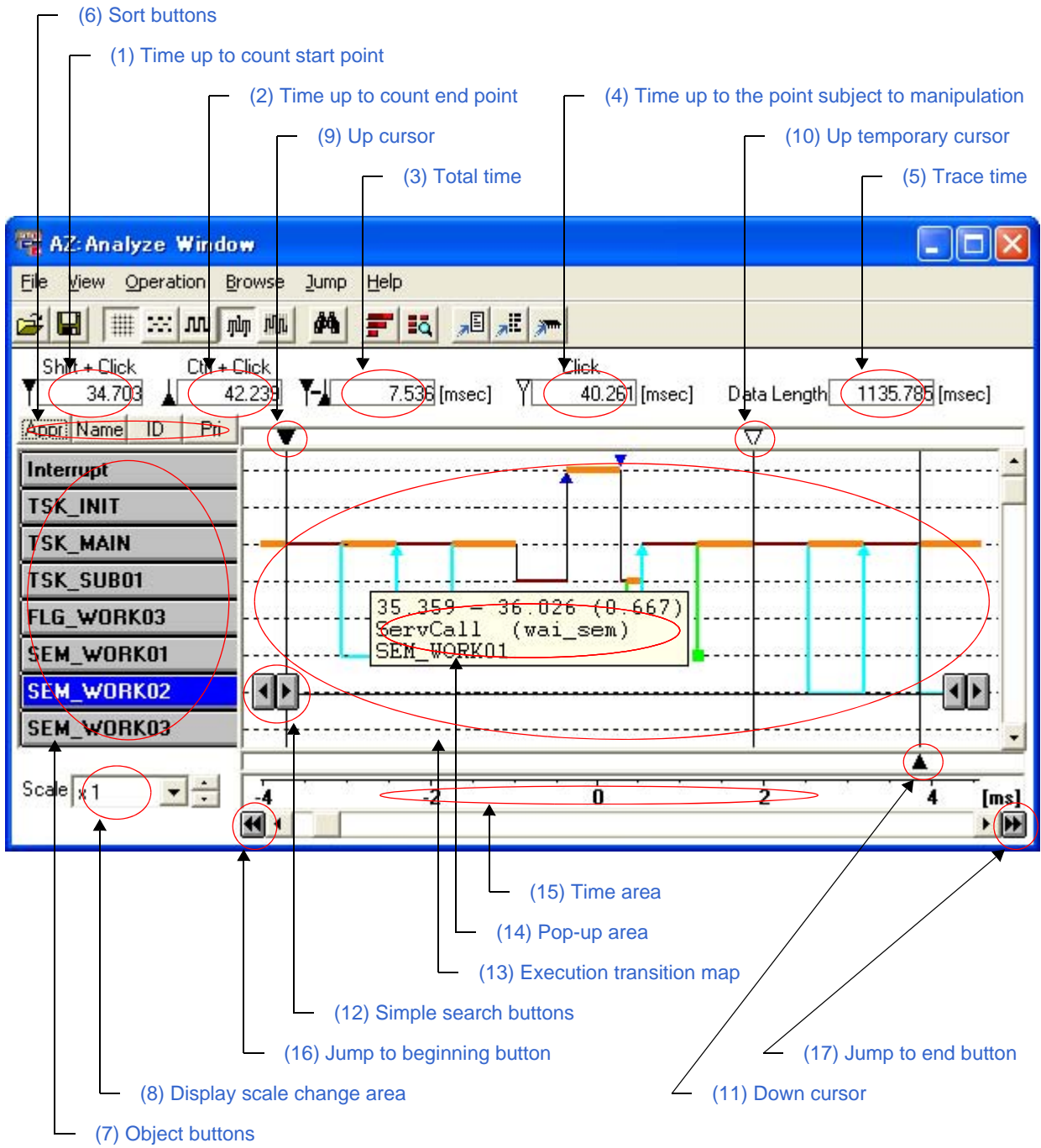
- Select the [Browse] menu -> [Analyze...].
- Click the  button on the toolbar.
- Press the [Alt], [B] and [A] keys in that order.
- Press the [Ctrl] + [A] keys at the same time.

Figure A-6. AZ:Analyze Window



This section describes the following items:

- [Explanation of each area](#)
- [Menubar](#)
- [Toolbar](#)
- [How to read execution transition map](#)
- [How to verify execution transition map](#)

## Explanation of each area

(1) Time up to count start point

This area displays the time up to the CPU usage counting start point.

The time up to the count start point is a relative time from when trace processing starts until the execution reaches the up cursor position (unit: ms).

(2) Time up to count end point

This area displays the time up to the CPU usage counting end point.

The time up to the count end point is a relative time from when trace processing starts until the execution reaches the down cursor position (unit: ms).

(3) Total time

This area displays the total time of the CPU usage.

The total time is a relative time indicated from the up cursor position to the down cursor position (unit: ms).

(4) Time up to the point subject to manipulation

This area displays the times up to when various types of manipulation (such as opening of the panels of CubeSuite, and execution of simple search) are performed.

The time up to the point subject to manipulation is an absolute time from when trace processing starts until the execution reaches the up temporary cursor (unit: ms).

(5) Trace time

The trace time is a relative time from the start to the end of trace processing (unit: ms).

(6) Sort buttons

This button is used to change the object button display order.

This area consists of the following buttons.

**Table A-2. Sort Buttons in AZ:Analyze Window**

Button	Function
Appr	Displays trace data in the order of detection (default).
Name	Displays the objects in the order of ASCII code.

Button	Function
ID	Displays the objects in the ID order.
Pri	Displays the objects in the priority order (valid for tasks only).

## (7) Object buttons

These buttons display the objects (interrupt handlers, tasks, idle routines, eventflags, semaphores, mailboxes, or fixed-sized memory pools) detected as trace data.

For details on the object button display format, refer to "[Object button display format](#)".

## (8) Display scale change area

This area is used to change the display scale of the execution transition map.

The drop-down list consists of the following items.

**Table A-3. Display Scale of Execution Transition Map**

Item	Meaning
x 2	Magnifies the execution transition map by 2.
x 1/2	Shrinks the execution transition map to 1/2.

## (9) Up cursor

This cursor specifies the point from which the CPU usage is counted.

This cursor can be moved by clicking it with the SHIFT key being pressed in the execution transition map.

## (10) Up temporary cursor

This cursor specifies the point from which a manipulation (such as opening of the panels of CubeSuite, and execution of simple search) is performed.

This cursor can be moved by dragging it with the SHIFT key being pressed in the execution transition map.

[Note] When the up cursor is moved, the up temporary cursor also moves to the same position.

## (11) Down cursor



This cursor specifies the point at which counting of the CPU usage ends.

This cursor can be moved by clicking it with the CTRL key being pressed in the execution transition map.

(12) Simple search buttons

These buttons are displayed when an object button is clicked, and used to perform the following manipulations. These buttons are hidden when the object button is clicked again.

**Table A-4. Simple Search Buttons in AZ:Analyze Window**

Button	Function
	Searches for the point at which an event related to the relevant object occurred, from the search start point toward the reverse direction to the time axis. The up temporary cursor shows the detected point. A beep is generated if no events have occurred at any location.
	Searches for the point at which an event related to the relevant object occurred, from the search start point toward the time axis direction. The up temporary cursor shows the detected point. A beep is generated if no events have occurred at any location.

(13) Execution transition map

This area displays the processing program analysis result.



Refer to "[How to read execution transition map](#)" for details on marks shown in the execution transition map.




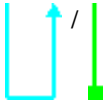

[Note] If the number of OS resources subject to display exceeds 1,000, this area may not be displayed correctly.

(14) Pop-up area

The following information related to the position pointed to by the mouse pointer pops up.

**Table A-5. Details of Information in AZ:Analyze Window**

Mouse Pointer	Meaning
Object button	Object name
	The following information related to processing program switching <ul style="list-style-type: none"> <li>- Time taken to generate processing program switching (unit: ms)</li> <li>- Name of processing program before switching</li> <li>- Name of processing program after switching</li> </ul>
	The following information related to service calls <ul style="list-style-type: none"> <li>- Time taken to issue a service call (unit: ms)</li> <li>- Time taken to return from a service call (unit: ms)</li> <li>- Service call processing time (unit: ms)</li> <li>- Service call name</li> <li>- Name of object subject to manipulation by service call</li> </ul>

Mouse Pointer	Meaning
Orange horizontal line 	The following information related to interrupts <ul style="list-style-type: none"> <li>- Time taken to generate an interrupt (unit: ms)</li> <li>- Time taken to finish interrupt handler processing (unit: ms)</li> <li>- Interrupt handler processing time (unit: ms)</li> </ul>
Blue triangle 	The following information related to interrupts <ul style="list-style-type: none"> <li>- Time taken to generate an interrupt (unit: ms)</li> </ul>
Blue inverted triangle 	The following information related to interrupts <ul style="list-style-type: none"> <li>- Time taken to finish interrupt handler processing (unit: ms)</li> </ul>
Light-blue/green line 	The following information related to object access status <ul style="list-style-type: none"> <li>- Time taken to issue a service call (unit: ms)</li> <li>- Time taken to return from a service call (unit: ms)</li> <li>- Name of processing program that issued a service call</li> <li>- Service call name</li> <li>- Name of object subject to manipulation by service call</li> </ul>
Red x-mark 	The following information related to service calls <ul style="list-style-type: none"> <li>- Time taken to return from a service call (unit: ms)</li> <li>- Value returned from service call</li> </ul>

[Note] Nothing pops up on light-blue horizontal lines.

(15) Time area

This area displays the guide for generation interval of events displayed in the execution transition map. The display unit is shown at the right end of this area.

[Note] In Equal mode, this area displays the guide for the number of events displayed in the execution transition map.

(16) Jump to beginning button



This button moves the up cursor to the top of trace data.

(17) Jump to end button


This button moves the down cursor to the end of trace data.

## Menubar

### (1) [File] menu

Open...	<p>Opens the <a href="#">Open/Save As dialog box</a>.</p> <p>Loads the file that contains information to be displayed in this window (extension: .az).</p> <p>The function of this item is same as that of the  button.</p>
Save...	<p>Opens the <a href="#">Open/Save As dialog box</a>.</p> <p>Specifies the name of the file into which information displayed in this window is saved (extension: .az).</p> <p>The function of this item is same as that of the  button.</p>
Close	Closes this widow.

### (2) [View] menu



Grid mode	Specifies whether to display gridlines in the execution transition map.
Grid	Displays gridlines (default).
Ungrid	Does not display gridlines.
View mode	Specifies the execution transition map display mode.
Simple	<p>Displays the execution transition map in <a href="#">Simple mode</a>.</p> <p>The function of this item is same as that of the  button.</p>
Standard	<p>Displays the execution transition map in <a href="#">Standard mode</a>.</p> <p>The function of this item is same as that of the  button.</p>
Detail	<p>Displays the execution transition map in <a href="#">Detailed mode</a> (default).</p> <p>The function of this item is same as that of the  button.</p>
Equal	<p>Displays the execution transition map in <a href="#">Equal mode</a>.</p> <p>The function of this item is same as that of the  button.</p>
Small	<p>shrinks the execution transition map display to 1/2.</p> <p>The effect is the same as selecting "x 1/2" in the display scale change area.</p>
Large	<p>Magnifies the execution transition map display by 2.</p> <p>The effect is the same as selecting "x 2" in the display scale change area.</p>
Find...	<p>Opens the <a href="#">Pattern Search dialog box</a>.</p> <p>Searches for the point at which a specific event occurred, based on the execution transition map displayed in this window.</p> <p>The function of this item is same as that of the  button.</p>

Sort Object	Specifies the object button display order.
Appear	Displays trace data in the order of detection (default)
Name	Displays the objects in the order of ASCII code
ID	Displays the objects in the ID order
Priority	Displays the objects in the priority order (valid for tasks only)


## (3) [Operation] menu



Active	Switches this window to the Active mode. This window is in the Active mode when opened. This window opens automatically when AZ78K0R is started.
Hold	Switches this window to the Hold mode.

## (4) [Browse] menu

CPU...	Opens the <a href="#">AZ:Cpu Window</a> in Active mode. Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the window. If the <a href="#">AZ:Cpu Window</a> has already been opened in the Active mode, information displayed in the window is updated.  The function of this item is same as that of the  button.
Trace View...	Opens the <a href="#">AZ:Trace View Window</a> in Active mode. Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the window. If the <a href="#">AZ:Trace View Window</a> has already been opened in the Active mode, information displayed in the window is updated.  The function of this item is same as that of the  button.

## (5) [Jump] menu

Source Text...	Opens the Editor panel of Cube Suite. Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel. If the Editor panel has already been opened in the Active mode, information displayed in the panel is updated.  The function of this item is same as that of the  button.
----------------	---

Assemble...	<p>Opens the Disassemble panel of Cube Suite.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel.</p> <p>If the Disassemble panel has already been opened in the Active mode, information displayed in the panel is updated.</p> <p>The function of this item is same as that of the  button.</p>
Memory...	<p>Opens the Memory panel of CubeSuite.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel.</p> <p>If the Memory panel has already been opened in the Active mode, information displayed in the panel is updated.</p> <p>The function of this item is same as that of the  button.</p>







[Note] If there is no address information at the position pointed to by the up temporary cursor, items in the Jump menu are not available.








(6) [Help] menu

This Window	Displays the help window for this window.
Help Topics	Opens the online help, with the [Search] tab displayed.

## Toolbar

The toolbar consists of buttons that can execute frequently used menu items with a single action.

Button	Function
	<p>Opens the <a href="#">Open/Save As dialog box</a>.</p> <p>Loads the file that contains information to be displayed in this window (extension: .az).</p> <p>Same operation as [File] menu -&gt; [Open...].</p>
	<p>Opens the <a href="#">Open/Save As dialog box</a>.</p> <p>Specifies the name of the file into which information displayed in this window is saved (extension: .az).</p> <p>Same operation as [File] menu -&gt; [Save...].</p>
	<p>Specifies whether to display gridlines in the execution transition map (default: Displays gridlines).</p>
	<p>Displays the execution transition map in <a href="#">Simple mode</a>.</p> <p>Same operation as [View] menu -&gt; [View mode] -&gt; [Simple].</p>
	<p>Displays the execution transition map in <a href="#">Standard mode</a>.</p> <p>Same operation as [View] menu -&gt; [View mode] -&gt; [Standard].</p>
	<p>Displays the execution transition map in <a href="#">Detailed mode</a> (default).</p> <p>Same operation as [View] menu -&gt; [View mode] -&gt; [Detail].</p>

Button	Function
	<p>Displays the execution transition map in <a href="#">Equal mode</a>.</p> <p>Same operation an [View] menu -&gt; [View mode] -&gt; [Equal].</p>
	<p>Opens the <a href="#">Pattern Search dialog box</a>.</p> <p>Searches for the point at which a specific event occurred, based on the execution transition map displayed in this window.</p> <p>Same operation an [View] menu -&gt; [Find...].</p>
	<p>Opens the <a href="#">AZ:CPU Window</a> in Active mode.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the window.</p> <p>If the <a href="#">AZ:CPU Window</a> has already been opened in the Active mode, information displayed in the window is updated.</p> <p>Same operation an [Browse] menu -&gt; [CPU...].</p>
	<p>Opens the <a href="#">AZ:Trace View Window</a> in Active mode.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the window.</p> <p>If the <a href="#">AZ:Trace View Window</a> has already been opened in the Active mode, information displayed in the window is updated.</p> <p>Same operation an [Browse] menu -&gt; [Trace View...].</p>
	<p>Opens the Editor panel of Cube Suite.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel.</p> <p>If the Editor panel has already been opened in the Active mode, information displayed in the panel is updated.</p> <p>Same operation an [Jump] menu -&gt; [Source Text...].</p>
	<p>Opens the Disassemble panel of Cube Suite.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel.</p> <p>If the Disassemble panel has already been opened in the Active mode, information displayed in the panel is updated.</p> <p>Same operation an [Jump] menu -&gt; [Assemble...].</p>
	<p>Opens the Memory panel of CubeSuite.</p> <p>Information later than the position pointed to by the up temporary cursor in the execution transition map is displayed in the panel.</p> <p>If the Memory panel has already been opened in the Active mode, information displayed in the panel is updated.</p> <p>Same operation an [Jump] menu -&gt; [Memory...].</p>

## Object button display format

Object buttons are displayed as explained below.

A too-long object name is abbreviated, but its real name pops up if the relevant button is pointed to by the mouse pointer.

Button	Meaning
Interrupt	Interrupt handler The button name is fixed to "Interrupt".
Task name	Task Task name defined in system configuration file
Tsk (????)	Unknown processing program If execution of starts in the middle of a processing program, AZ78K0R cannot identify whether the processing program is an interrupt handler, task, or idle routine. AZ78K0R therefore handles the processing program as an unknown processing program.
Idle	Idle routine The button name is fixed to "Idle"
Eventflag name	Eventflag Eventflag name defined in system configuration file
Semaphore name	Semaphore Semaphore name defined in system configuration file
Mailbox name	Mailbox Mailbox name defined in system configuration file
Fixed-sized memory pool name	Fixed-sized memory pool Fixed-sized memory pool name defined in system configuration file
Etc.	Object hidden by right-clicking the object button

(1) Object button display order

The object button display order can be changed by clicking a sort button, or by dragging the corresponding object button.

(2) Hiding object buttons

Object buttons can be hidden by right-clicking the button corresponding to the object.

## How to read execution transition map

The three types of basic display modes: [Simple mode](#), [Standard mode](#) and [Detailed mode](#), are available for displaying the execution transition map. In addition to them, the [Equal mode](#) can be specified for each basic mode.

These modes can be selected by selecting the [View] menu -> [View mode].

- (1) Simple mode  
Displays the CPU usage with horizontal lines.

Figure A-7. AZ:Analyze Window (Simple Mode)

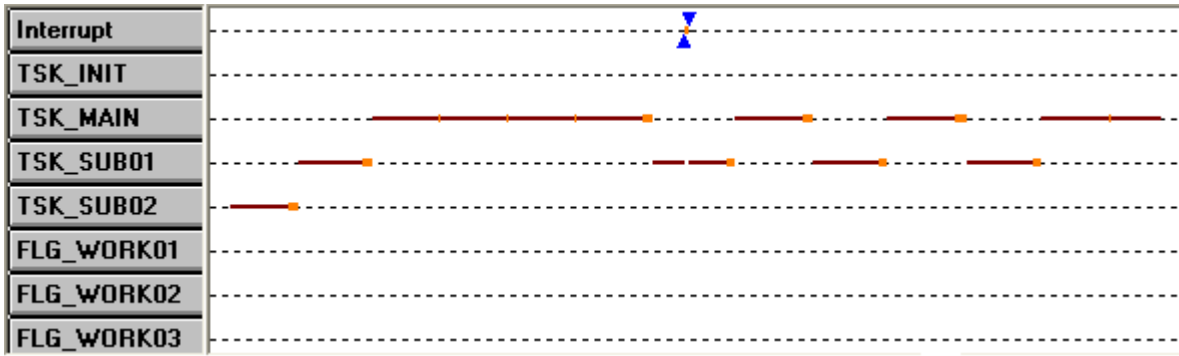






Table A-6. Marks Displayed in AZ:Analyze Window (Simple Mode)

Mark	Meaning
Brown horizontal line 	Task or idle routine
Orange horizontal line 	Interrupt handler or RX78K0R internal processing
Blue triangle 	Start of interrupt handler processing
Blue inverted triangle 	End of interrupt handler processing

(2) Standard mode

Displays the processing program switching status, with vertical lines being combined with information displayed in the Simple mode.

Figure A-8. AZ:Analyze Window (Standard Mode)

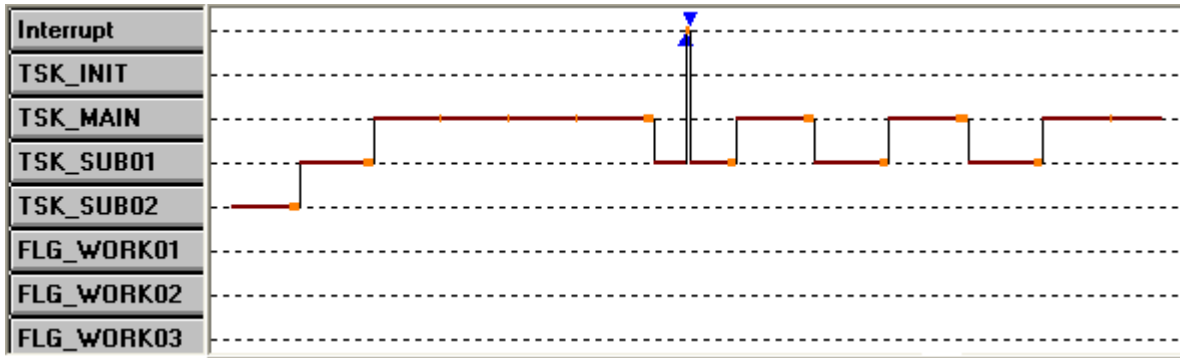
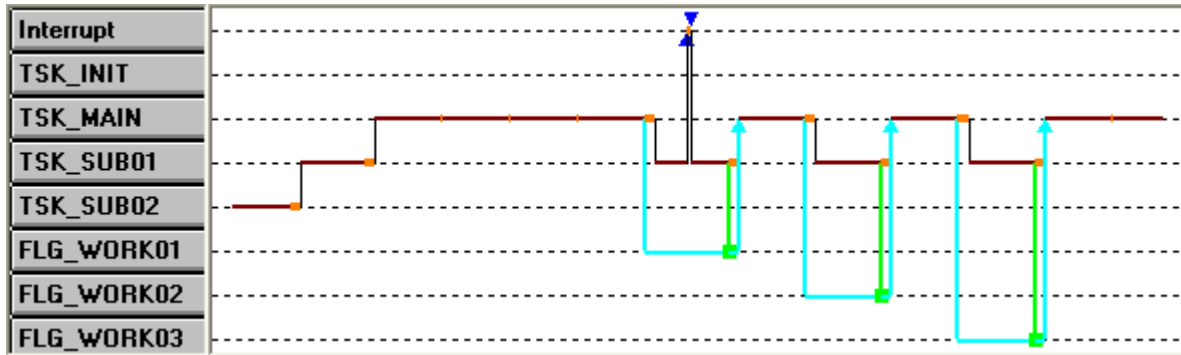


Table A-7. Marks Displayed in AZ:Analyze Window (Standard Mode)







Mark	Meaning
Brown horizontal line —	Task or idle routine
Orange horizontal line —	Interrupt handler or RX78K0R internal processing
Blue triangle ▲	Start of interrupt handler processing
Blue inverted triangle ▼	End of interrupt handler processing
Black vertical line 	Processing program switching status




- (3) Detailed mode  
Displays the service call issuance status, in addition to information displayed in the Standard mode.

**Figure A-9. AZ:Analyze Window (Detailed Mode)**



**Table A-8. Marks Displayed in AZ:Analyze Window (Detailed mode)**

Mark	Meaning
Brown horizontal line 	Task or idle routine
Orange horizontal line 	Interrupt handler or RX78K0R internal processing
Blue triangle 	Start of interrupt handler processing
Blue inverted triangle 	End of interrupt handler processing
Black vertical line 	Processing program switching status
Light-blue line 	[If a semaphore is subject to manipulation] Issuance of wai_sem, pol_sem or twai_sem  [If an eventflag is subject to manipulation] Issuance of wai_flg, pol_flg or twai_flg  [If a mailbox is subject to manipulation] Issuance of rcv_mbx, prcv_mbx or trcv_mbx  [If a fixed-sized memory pool is subject to manipulation] Issuance of get_mpf, pget_mpf or tget_mpf

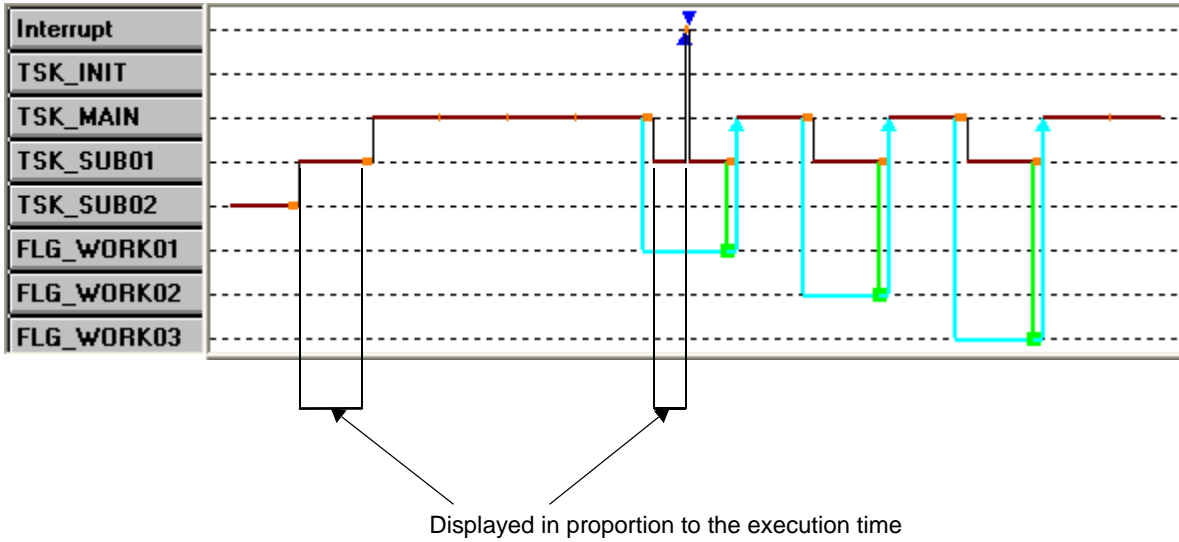
Mark	Meaning
Green line 	[If a semaphore is subject to manipulation] Issuance of sig_sem or isig_sem  [If an eventflag is subject to manipulation] Issuance of set_flg or iset_flg/clr_flg  [If a mailbox is subject to manipulation] Issuance of snd_mbx  [If a fixed-sized memory pool is subject to manipulation] Issuance of rel_mpf
Red x-mark 	A service call abnormally ended
Timeout mark 	A service call timed out

[Note] If a service call that moves to the WAITING state is issued in an unknown processing program, "Tsk (???)", AZ78K0R cannot recognize the WAITING state being released. The light-blue horizontal lines will therefore be drawn up to the end of the trace data counting.

(4) Equal mode

In the default state (in which the Equal mode is not specified), the brown/orange horizontal lines that show the CPU usage are displayed in proportion to the execution time taken by each processing program (see [Figure A-10](#)).

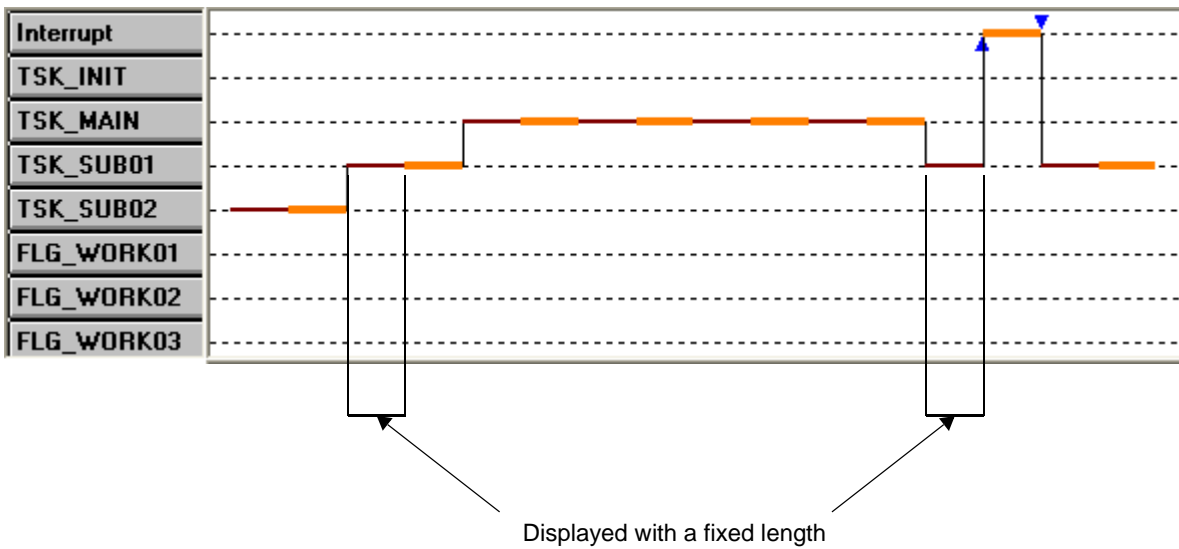
**Figure A-10. AZ:Analyze Window (When Equal Mode Is Not Specified)**



In the state in which the Equal mode is specified, in contrast, the brown/orange horizontal lines that show the CPU usage are displayed with a fixed length (see [Figure A-11](#)).

The horizontal lines between events such as task switching are also displayed in a fixed interval, not proportional to the CPU processing time.


**Figure A-11. AZ:Analyze Window (When Equal Mode Is Specified)**



## How to verify execution transition map

The execution transition map displayed in this window can be verified using either of the following two methods.

(1) Search using simple search buttons

The simple search buttons () can be used for searching for the location where an event related to the specified object occurred.

The following explains the procedure for searching event occurrence location, by using simple search buttons.

(a) Specification of search start point

Move the up temporary cursor to the search start point.

If the up temporary cursor is not displayed, move the up cursor to the search start point.

(b) Displaying simple search buttons

Click the object button corresponding to the object subject to search, to display the simple search buttons.

(c) Clicking simple search buttons

- When the  is clicked

The point at which an event related to the object selected in (b) occurred is searched for, from the search start point toward the reverse direction to the time axis, and the detected point is shown by the up temporary cursor.

A beep is generated if no events have occurred at any location.

- When the  is clicked

The point at which an event related to the object selected in (b) occurred is searched for, from the search start point toward the time axis direction, and the detected point is shown by the up temporary cursor.

A beep is generated if no events have occurred at any location.

(2) Searching in [Pattern Search dialog box](#)


The point at which a specific event occurred can be searched for in the [Pattern Search dialog box](#).

Refer to "[How to search](#)" for searching for the point at which a specific event occurred, in the [Pattern Search dialog box](#).

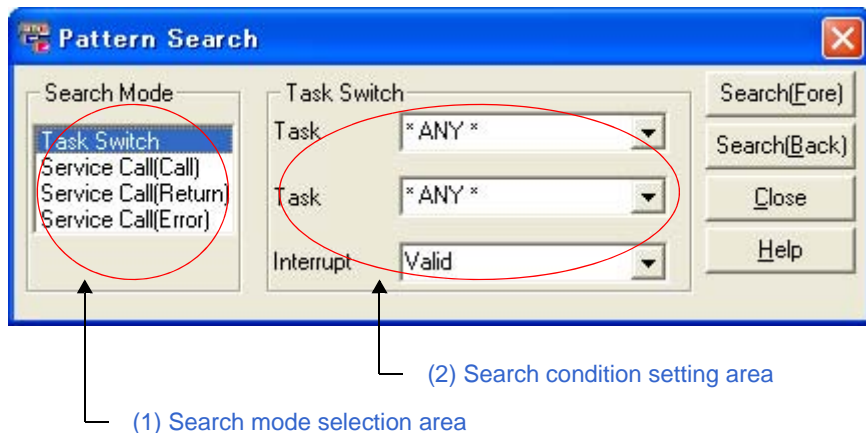
## Pattern Search dialog box

This dialog box is used to search for the point at which a specific event occurred, based on the execution transition map displayed in the [AZ:Analyze Window](#).

This dialog box can be opened from the [AZ:Analyze Window](#) by any of the following:

- Select the [View] menu -> [Find...].
- Click the  button on the toolbar.
- Press the [Alt], [V] and [F] keys in that order.
- Press the [Ctrl] + [F] keys at the same time.

**Figure A-12. Pattern Search Dialog Box**



This section describes the following items:

- [Explanation of each area](#)
- [Function buttons](#)
- [How to search](#)

## Explanation of each area

(1) Search mode selection area

This area is used to select an event to be searched for (the type of an event to be searched for from the execution transition map displayed in the [AZ:Analyze Window](#)) as search mode.

The following items can be selected as the search mode.

**Table A-9. Types of Search Mode**

Search Mode	Meaning
Task Switch	Switching of processing programs
Service Call(Call)	Issuance of service call
Service Call(Return)	Returning from a service call routine
Service Call(Error)	Errors returned from a service call

(2) Search condition setting area

This area is used to select the conditions for searching for the selected search mode.

The items displayed in this area vary depending on the selected search mode.

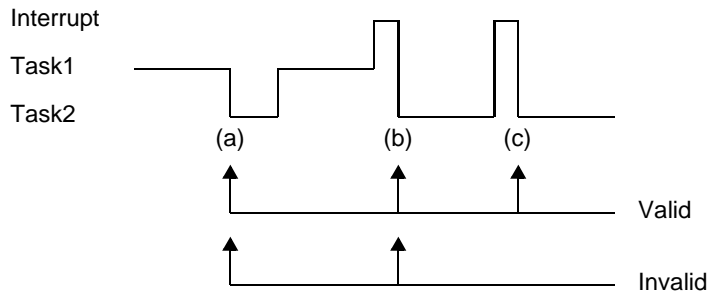
**Table A-10. Search Conditions (When Task Switch Is Selected)**

Item	Meaning
Task (upper)	Select a processing program before switching. Select [ <i>name of the target task</i> ] for task switching, [Interrupt] for interrupt handler switching, or [* ANY *] for any processing program switching.
Task (lower)	Select a processing program after switching. Select [ <i>name of the target task</i> ] for task switching, [Interrupt] for interrupt handler switching, or [* ANY *] for any processing program switching.
Interrupt	If [* ANY *] is selected for Task (upper) or Task (lower), select whether to include the location where switching to/from an interrupt handler occurred, into the search targets. Select [Valid] to include the location, or [Invalid] not to include the location.

[Note] The search target varies depending on which of the items, [Valid] and [Invalid], is selected in the Interrupt list, as shown below.

The following figure assumes that switching from a "processing program [\* ANY \*]" to "processing program [Task2]" is included into the search targets.

**Figure A-13. Difference in Search Targets**



- Switching point (a)

Valid	Switching from [Task1] to [Task2] is included into the search targets.
Invalid	Switching from [Task1] to [Task2] is included into the search targets.

- Switching point (b)

Valid	Switching from [Interrupt] to [Task2] is included into the search targets.
Invalid	Switching from [Interrupt] to [Task2] is ignored, and switching from [Task1] to [Task2] is included into the search targets.

- Switching point (c)

Valid	Switching from [Interrupt] to [Task2] is included into the search targets.
Invalid	Switching from [Interrupt] to [Task2] is ignored, and switching from [Task1] to [Task2] is included into the search targets.

**Table A-11. Search Conditions (When a Service Call (Call, Return, Error) Is Selected)**

Item	Meaning
Task	Select the processing program that issued the service call. Select [name of the target task] for task switching, [Interrupt] for interrupt handler switching, or [* ANY *] for any processing program switching.
Service Call	Select the service call name. Select [* ANY *] for any service call.
Object	Select the name of an object subject to manipulation by the service call. Select [* ANY *] for any object.

## Function buttons

Button	Function
Search(Fore)	<p>Searches for the locations that match the conditions selected in this dialog box, from the search start point toward the time axis direction.</p> <p>The up temporary cursor in the <a href="#">AZ:Analyze Window</a> shows the locations that match the specified conditions.</p> <p>A beep is generated if no locations match the specified conditions.</p>
Search(Back)	<p>Searches for the locations that match the conditions selected in this dialog box, from the search start point toward the reverse direction to the time axis.</p> <p>The up temporary cursor in the <a href="#">AZ:Analyze Window</a> shows the locations that match the specified conditions.</p> <p>A beep is generated if no locations match the specified conditions.</p>
Close	Closes this dialog box.
Help	Displays the help window for this dialog box.

## How to search

Using the following procedure, the point at which a specific event occurred to be searched for, based on the execution transition map displayed in the [AZ:Analyze Window](#).

(1) Specification of search start point

Move the up temporary cursor to the search start point in the [AZ:Analyze Window](#).

If the up temporary cursor is not displayed, move the up cursor to the search start point.

(2) How to open this dialog box

Select the [View] menu -> [Find...] in the [AZ:Analyze Window](#).

(3) Selection of search mode and search conditions

Select the search mode and search conditions in the [Search mode selection area](#) and [Search condition setting area](#) in this dialog box.

(4) Clicking function buttons

- When the [Search(Fore)] button is clicked

Points that match the conditions selected in this dialog box are searched for from the search start point toward the time axis direction, the up temporary cursor in the [AZ:Analyze Window](#) shows the locations that match the specified conditions.

A beep is generated if no locations match the specified conditions

- When the [Search(Back)] button is clicked


Points that match the conditions selected in this dialog box are searched for from the search start point toward the reverse direction to the time axis. The up temporary cursor in the [AZ:Analyze Window](#) shows the locations that match the specified conditions.

A beep is generated if no locations match the specified conditions.

## AZ:Cpu Window

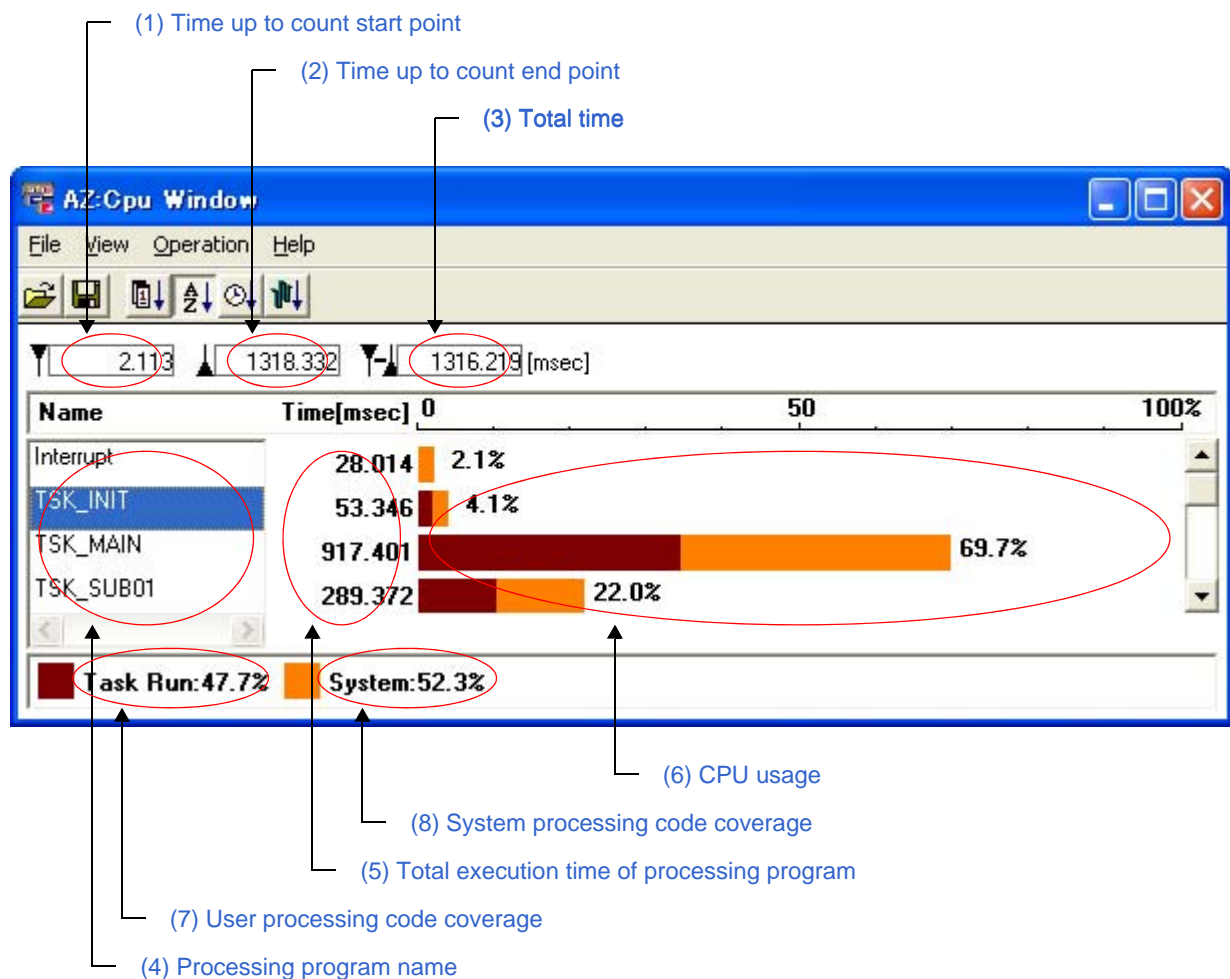
This window displays the CPU usage in a section between the up cursor and down cursor in the [AZ:Analyze Window](#).

This window can be opened from the [AZ:Analyze Window](#) by any of the following:

- Select the [Browse] menu -> [CPU...].
- Click the  button on the toolbar.
- Press the [Alt], [B] and [C] keys in that order.
- Press the [Ctrl] + [C] keys at the same time.

[Note] This window is closed in conjunction with closing the [AZ:Analyze Window](#) that called the file.

Figure A-14. AZ:Cpu Window



This section describes the following items:

- [Explanation of each area](#)
- [Menubar](#)
- [Toolbar](#)
- [Count method](#)

## **Explanation of each area**

(1) Time up to count start point

This area displays the time up to the CPU usage counting start point.

The time up to the count start point is a relative time from when trace processing starts until the execution reaches the up cursor position (unit: ms).

(2) Time up to count end point

This area displays the time up to the CPU usage counting end point.

The time up to the count end point is a relative time from when trace processing starts until the execution reaches the down cursor position (unit: ms).

(3) Total time

This area displays the total time of the CPU usage.

The total time is a relative time indicated from the up cursor position to the down cursor position (unit: ms).

(4) Processing program name

This area lists the processing programs executed within the total time.

The following types of processing programs are displayed.

**Table A-12. Processing Program Types**

<b>Processing Program Name</b>	<b>Meaning</b>
Task name	Task
Interrupt	Interrupt handler
Idle	Idle routine

(5) Total execution time of processing program

This area displays the total execution time of the processing program within the total time (unit: ms).

## (6) CPU usage

This area displays the bar graphs for indicating the percentage of the total time occupied by the execution time of processing programs.

The bar graphs distinguish user processing and system processing with different colors.

**Table A-13. Colors in Bar Graph**

Color	Meaning
Brown	User processing (task, idle routine)
Orange	System processing (interrupt handler or RX78K0R internal processing)

[Note] Service calls issued in a task or interrupt handler are handled as RX78K0R internal processing.

## (7) User processing code coverage



This area displays the percentage of the total time occupied by the user processing execution time.

## (8) System processing code coverage




This area displays the percentage of the total time occupied by the system processing execution time.


## Menubar

## (1) [File] menu

Open...	Opens the <a href="#">Open/Save As dialog box</a> . Loads the file that contains information to be displayed in this window (extension: .azc). The function of this item is same as that of the  button.
Save...	Opens the <a href="#">Open/Save As dialog box</a> . Specifies the name of the file into which information displayed in this window is saved (extension: .azc). The function of this item is same as that of the  button.
Close	Closes this window.

## (2) [View] menu

Sort Appear	Displays the bar graphs in the order of detection in trace data. The function of this item is same as that of the  button.
Sort Name	Displays the bar graphs in the alphabetical order of the processing program names. The function of this item is same as that of the  button.
Sort Time	Displays the bar graphs in the order of longer total execution time (default). The function of this item is same as that of the  button.

Sort Analyze	Displays the bar graphs in the same order as those displayed in the execution transition map.  The function of this item is same as that of the  button.
--------------	---

## (3) [Operation] menu







Active	Switches this window to the Active mode. This window is in the Active mode when opened.
Hold	Switches this window to the Hold mode.

## (4) [Help] menu

This Window	Displays the help window for this window.
Help Topics	Opens the online help, with the [Search] tab displayed.

**Toolbar**

The toolbar consists of buttons that can execute frequently used menu items with a single action.

Button	Function
	Opens the <a href="#">Open/Save As dialog box</a> . Loads the file that contains information to be displayed in this window (extension: .azc). Same operation as [File] menu -> [Open...].
	Opens the <a href="#">Open/Save As dialog box</a> . Specifies the name of the file into which information displayed in this window is saved (extension: .azc). Same operation as [File] menu -> [Save...].
	Displays the bar graphs in the order of detection in trace data. Same operation as [View] menu -> [Sort Detect].
	Displays the bar graphs in the alphabetical order of the processing program names. Same operation as [View] menu -> [Sort Name].
	Displays the bar graphs in the order of longer total execution time (default). Same operation as [View] menu -> [Sort Time].
	Displays the bar graphs in the same order as those displayed in the execution transition map. Same operation as [View] menu -> [Sort Analyze].

## Count method

Using the following procedure, the CPU usage in a section between the up cursor and down cursor in the [AZ:Analyze Window](#) can be checked.

- (1) Specification of count start point

Move the up cursor to the count start position in the [AZ:Analyze Window](#).

- (2) Specification of count end point

Move the down cursor to the count end position in the [AZ:Analyze Window](#).

- (3) How to open this window

Select the [Browse] menu -> [CPU...] in the [AZ:Analyze Window](#).


When this window is opened, information corresponding to the count section specified in (1) and (2) is displayed.

[Note] When this window is in the Active mode, information displayed in this window is also updated automatically along with moving of the up/down cursor in the [AZ:Analyze Window](#) and information corresponding to the move destination of the up/down cursor is displayed.

## AZ:Trace View Window

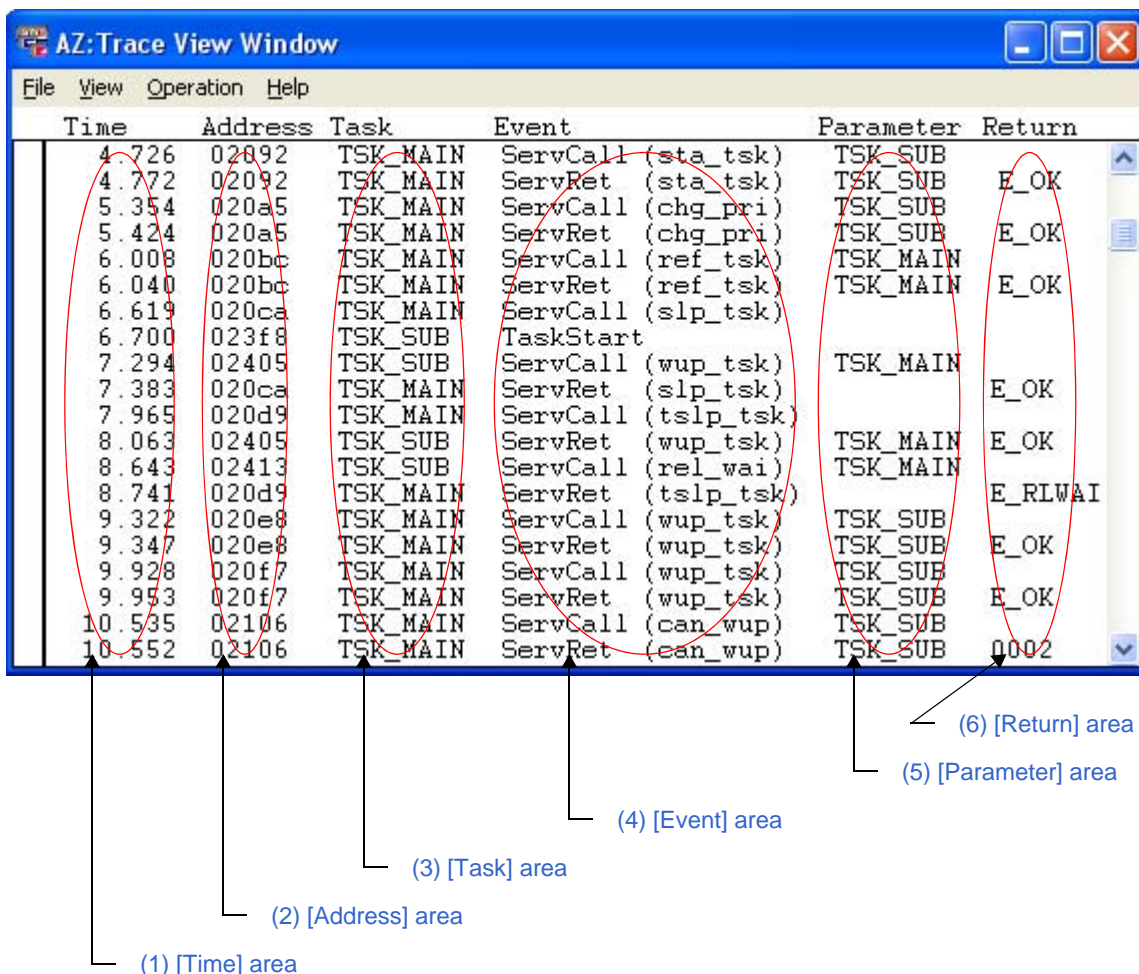
This window displays information obtained from the execution transition map in the [AZ:Analyze Window](#) in list form.

This window can be opened from the [AZ:Analyze Window](#) by any of the following:

- Select the [Browse] menu -> [Trace View...].
- Click the  button on the toolbar.
- Press the [Alt], [B] and [T] keys in that order.
- Press the [Ctrl] + [T] keys at the same time.

[Note] This window is closed in conjunction with closing the [AZ:Analyze Window](#) that called the file.

Figure A-15. AZ:Trace View Window



Time	Address	Task	Event	Parameter	Return
4.726	02092	TSK_MAIN	ServCall (sta_tsk)	TSK_SUB	
4.772	02092	TSK_MAIN	ServRet (sta_tsk)	TSK_SUB	E_OK
5.354	020a5	TSK_MAIN	ServCall (chg_pri)	TSK_SUB	
5.424	020a5	TSK_MAIN	ServRet (chg_pri)	TSK_SUB	E_OK
6.008	020bc	TSK_MAIN	ServCall (ref_tsk)	TSK_MAIN	
6.040	020bc	TSK_MAIN	ServRet (ref_tsk)	TSK_MAIN	E_OK
6.619	020ca	TSK_MAIN	ServCall (slp_tsk)		
6.700	023f8	TSK_SUB	TaskStart		
7.294	02405	TSK_SUB	ServCall (wup_tsk)	TSK_MAIN	
7.383	020ca	TSK_MAIN	ServRet (slp_tsk)		E_OK
7.965	020d9	TSK_MAIN	ServCall (tslp_tsk)		
8.063	02405	TSK_SUB	ServRet (wup_tsk)	TSK_MAIN	E_OK
8.643	02413	TSK_SUB	ServCall (rel_wai)	TSK_MAIN	
8.741	020d9	TSK_MAIN	ServRet (tslp_tsk)		E_RLWAI
9.322	020e8	TSK_MAIN	ServCall (wup_tsk)	TSK_SUB	
9.347	020e8	TSK_MAIN	ServRet (wup_tsk)	TSK_SUB	E_OK
9.928	020f7	TSK_MAIN	ServCall (wup_tsk)	TSK_SUB	
9.953	020f7	TSK_MAIN	ServRet (wup_tsk)	TSK_SUB	E_OK
10.535	02106	TSK_MAIN	ServCall (can_wup)	TSK_SUB	
10.552	02106	TSK_MAIN	ServRet (can_wup)	TSK_SUB	0002

(1) [Time] area

(2) [Address] area

(3) [Task] area

(4) [Event] area

(5) [Parameter] area

(6) [Return] area

This section describes the following items:

- [Explanation of each area](#)
- [Menubar](#)
- [How to display](#)
- [How to read the list](#)
- [Caution](#)

## Explanation of each area

- (1) [Time] area  
This area shows a relative time from when trace processing starts until the target event occurs (unit: ms).
- (2) [Address] area  
This area shows the execution address at which the target event occurred.
- (3) [Task] area  
This area shows the name of the processing program in which the target event occurred.  
The following types of processing programs are displayed.

**Table A-14. Processing Program Types**

Processing Program Name	Meaning
Task name	Issuance of service call, returning from a service call routine, start of task processing
Interrupt	Issuance of service call, returning from a service call routine, start of interrupt handler processing, end of interrupt handler processing
Idle	Start of idle routine processing

- (4) [Event] area  
This area shows the types of the target events.  
The following types of events are displayed.

**Table A-15. Event Types**

Event Name	Meaning
ServCall( <i>xxx_xxx</i> )	Issuance of service call A service call name is displayed inside the parentheses.

Event Name	Meaning
ServRet( <i>xxx_xxx</i> )	Returning from a service call routine A service call name is displayed inside the parentheses. The inside of the parentheses is left blank if the event that issued the service call does not exist in the trace memory.
Int	Start of interrupt handler processing
IntRet	End of interrupt handler processing
TaskStart	Start of task processing
Idle	Start of idle routine processing

## (5) [Parameter] area

If the event type is "ServCall(*xxx\_xxx*)" or "ServRet(*xxx\_xxx*)", this area displays the name of the object to be manipulated by the service call (such as task name, semaphore name, or eventflag name).

## (6) [Return] area

If the event type is "ServRet(*xxx\_xxx*)", this area displays the values returned from the service call.

## Menubar

## (1) [File] menu

Save...	Opens the <a href="#">Open/Save As dialog box</a> . Specifies the name of the file into which information displayed in this window is saved (extension: .azt).
Close	Closes this window.

## (2) [View] menu

Find...	Opens the <a href="#">Trace Search dialog box</a> . Searches for specific information (trace data) based on the list displayed in this window. This menu is unavailable when this window is in the Hold mode.
Timetag	Selects [Show] or [Hide] of the Time area (default: Show).
Address	Selects [Show] or [Hide] of the Address area (default: Show).
Task	Selects [Show] or [Hide] of the Task area (default: Show).
Event	Selects [Show] or [Hide] of the Event area (default: Show).
Parameter	Selects [Show] or [Hide] of the Parameter area (default: Show).
Return	Selects [Show] or [Hide] of the Return area (default: Show).

## (3) [Operation] menu

Active	Switches this window to the Active mode. This window is in the Active mode when opened.
Hold	Switches this window to the Hold mode.

## (4) [Help] menu

This Window	Displays the help window for this window.
Help Topics	Opens the online help, with the [Search] tab displayed.

**How to display**

Using the following procedure, information obtained from the execution transition map displayed in the [AZ:Analyze Window](#) can be listed.

## (1) Specification of display start position

Move the up temporary cursor to the display start position in the [AZ:Analyze Window](#).

If the up temporary cursor is not displayed, the display start position is the top of the trace data.

## (2) How to open this window

Select the [Browse] menu -> [Trace View...] in the [AZ:Analyze Window](#).

When this window is opened, information corresponding to the display start point specified in (1) is displayed in the first line.

[Note] When this window is in the Active mode, information displayed in this window is also updated automatically along with moving of the up/down cursor in the [AZ:Analyze Window](#), and information corresponding to the move destination of the up/down cursor is displayed in the first line.

## How to read the list

The following explains how to read the list displayed in this window, using [Figure A-16](#) as an example.

**Figure A-16. How to Read the List**

	Time	Address	Task	Event	Parameter	Return
	152.951	00398	TSK_MAIN	ServRet	(get_mpf) MPF_WORK01	E_OK
	153.608	003b1	TSK_MAIN	ServCall	(get_mpf) MPF_WORK02	
	154.285	003b1	TSK_MAIN	ServRet	(get_mpf) MPF_WORK02	E_OK
	154.941	003c9	TSK_MAIN	ServCall	(get_mpf) MPF_WORK03	
	155.617	003c9	TSK_MAIN	ServRet	(get_mpf) MPF_WORK03	E_OK
(1)	156.273	003df	TSK_MAIN	ServCall	(rel_mpf) MPF_WORK01	
(2)	156.947	003df	TSK_MAIN	ServRet	(rel_mpf) MPF_WORK01	E_OK
	157.603	003f6	TSK_MAIN	ServCall	(rel_mpf) MPF_WORK02	
	158.277	003f6	TSK_MAIN	ServRet	(rel_mpf) MPF_WORK02	E_OK
	158.933	00413	TSK_MAIN	ServCall	(rel_mpf) MPF_WORK03	
	159.607	00413	TSK_MAIN	ServRet	(rel_mpf) MPF_WORK03	E_OK
	160.013	005ae	Interrupt	Int	(Interrupt)	
	160.465	005ae	TSK_MAIN	IntRet		
	160.743	00428	TSK_MAIN	ServCall	(rcv_mbx) MBX_WORK01	
	161.128	0050c	TSK_SUB01	ServRet	(sig_sem) SEM_WORK03	E_OK
	161.784	00525	TSK_SUB01	ServCall	(snd_mbx) MBX_WORK01	
	162.534	00428	TSK_MAIN	ServRet	(rcv_mbx) MBX_WORK01	E_OK
	163.191	0043e	TSK_MAIN	ServCall	(rcv_mbx) MBX_WORK02	
	163.954	00525	TSK_SUB01	ServRet	(snd_mbx) MBX_WORK01	E_OK
	164.611	00541	TSK_SUB01	ServCall	(snd_mbx) MBX_WORK02	

- (1) When 156.273 ms have elapsed after trace processing has started, processing program TSK\_MAIN issues service call rel\_mpf to fixed-sized memory pool MPF\_WORK1.
- (2) When 156.947 ms have elapsed after trace processing has started, service call rel\_mpf that was issued in (1) returns E\_OK.

The execution address in the above (1) and (2) means the address at which the relevant service call was issued.

## Caution

- If an interrupt occurs during idle routine processing, information related to "IntRet", which indicates the end of the interrupt handler processing, will not be displayed.

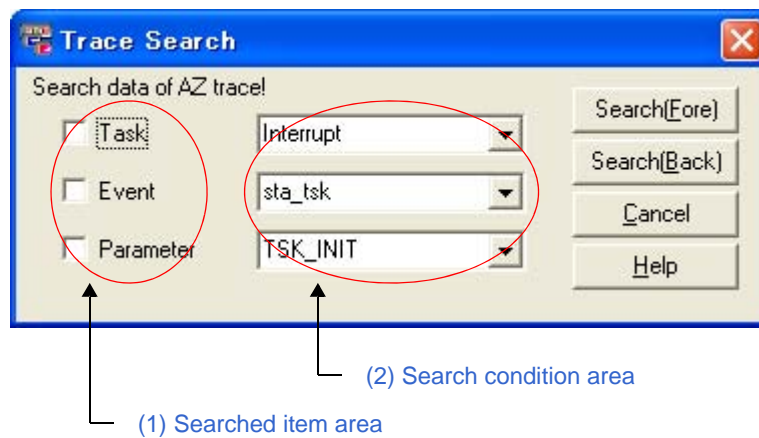
## Trace Search dialog box

This dialog box is used to search for specific information (trace data) based on the list displayed in the [AZ:Trace View Window](#).

This dialog box can be opened from the [AZ:Trace View Window](#) by any of the following:

- Select the [View] menu -> [Find...].
- Press the [Alt], [V] and [F] keys in that order.
- Press the [Ctrl] + [F] keys at the same time.

**Figure A-17. Trace Search Dialog Box**



This section describes the following items:

- [Explanation of each area](#)
- [Function buttons](#)
- [How to search](#)

## Explanation of each area

(1) Searched item area

This area is used to specify the items to be searched for (from the list displayed in the [AZ:Trace View Window](#)).

Multiple items in the following table can be selected as the items to be searched for.

**Table A-16. Search Item Types**

Search Item Name	Meaning
Task	Searches for information from the Task area in the list displayed in the <a href="#">AZ:Trace View Window</a> .
Event	Searches for information from the Event area in the list displayed in the <a href="#">AZ:Trace View Window</a> .
Parameter	Searches for information from the Parameter area in the list displayed in the <a href="#">AZ:Trace View Window</a>

(2) Search condition area

This area is used to select the conditions for searching for items specified in the searched item area.

The following items can be selected as the search conditions.

- Processing program name (task name, Interrupt, Idle)
- Event type (service call name)
- Name of object subject to manipulation by service call such as task name, semaphore name, or eventflag name)

## Function buttons

Button	Function
Search(Fore)	Searches for information that matches the conditions specified in this dialog box, from the older trace data. Condition match information is displayed in the first line of the <a href="#">AZ:Trace View Window</a> . A beep is generated if condition match information does not exist in the relevant list.
Search(Back)	Searches for information that matches the conditions specified in this dialog box, from the newer trace data. Condition match information is displayed in the first line of the <a href="#">AZ:Trace View Window</a> . A beep is generated if condition match information does not exist in the relevant list.
Cancel	Closes this dialog box.
Help	Displays the help window for this dialog box.

## How to search

Using the following procedure, specific information can be searched for, based on the list displayed in the [AZ:Trace View Window](#).

- (1) Switching to Active mode

Select the [Operation] menu -> [Active] in the [AZ:Trace View Window](#) to switch the target window to the Active mode.

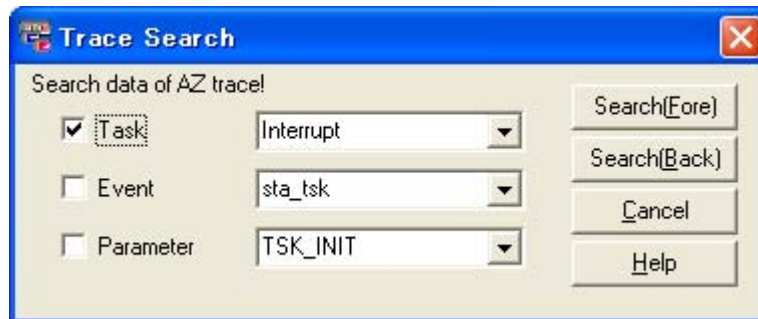
- (2) How to open this dialog box

Select the [View] menu -> [Find...] in the [AZ:Trace View Window](#).

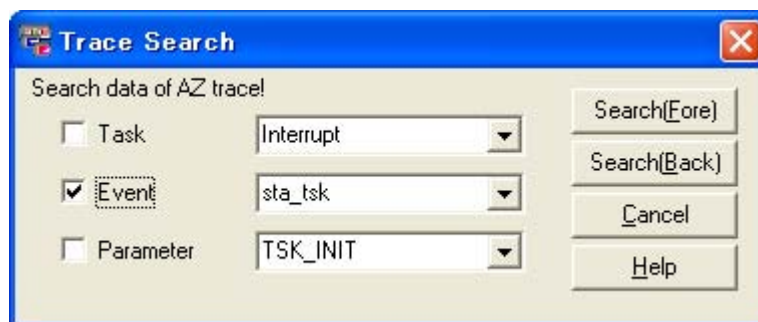
- (3) Specification of items to be searched for and search conditions

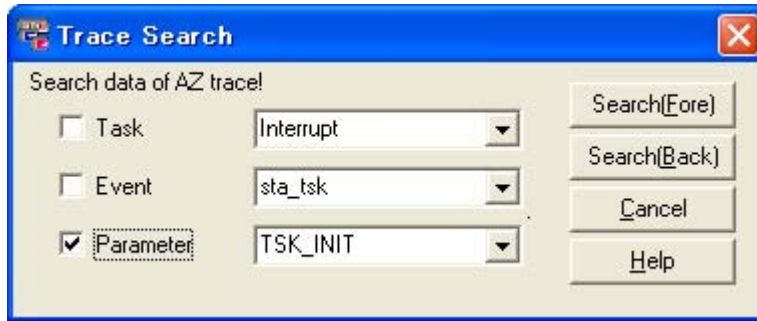
Specify the items to be searched for and search conditions in the searched item area and search condition area in this dialog box.

**Figure A-18. Specification Example 1 (Search for Interrupt)**



**Figure A-19. Specification Example 2 (Search for Location at Which sta\_tsk Was Issued)**



**Figure A-20. Specification Example 3 (Search for Location at Which Service Call Was Issued for TSK\_INIT)**

(4) Clicking function buttons

- [Search(Fore)] button

Information that matches the conditions specified in this dialog box is searched for from the newer trace data, and condition match information is displayed in the first line of the [AZ:Trace View Window](#).

A beep is generated if condition match information does not exist.

- [Search(Back)] button

Information that matches the conditions specified in this dialog box is searched for from the newer trace data, and condition match information is displayed in the first line of the [AZ:Trace View Window](#).

A beep is generated if condition match information does not exist.

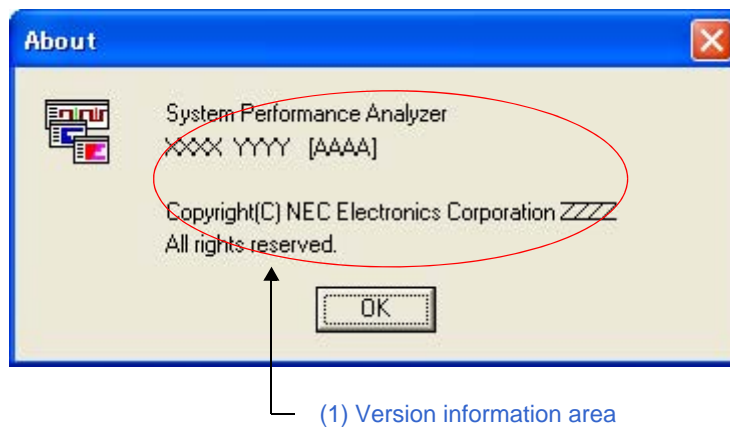
## About dialog box

This dialog box is used to display the version information of AZ78K0R.

This dialog box can be opened from the [AZ78K0R window](#) by any of the following:

- Select the [Help] menu -> [About...].
- Press the [Alt], [H] and [A] keys in that order.

**Figure A-21. About Dialog Box**



This section describes the following items:

- [Explanation of each area](#)
- [Function buttons](#)

### Explanation of each area

- (1) Version information area

This area is used to display "product name, version number, date of product build and copyright".

### Function buttons

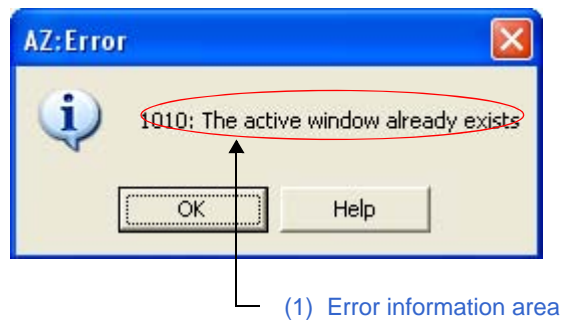
Button	Function
OK	Closes this dialog box.

## AZ:Error dialog box

This dialog box is used to display the error information of AZ78K0R.

This dialog box opens automatically when an invalid operation is performed in a window or a dialog box.

Figure A-22. AZ:Error Dialog Box



This section describes the following items:

- [Explanation of each area](#)
- [Function buttons](#)

### Explanation of each area

- (1) Error information area
- This area is used to display "error number and error message".
- For details of the error information, see "[B. 2 Error Messages](#)".

### Function buttons

Button	Function
OK	Closes this dialog box.
Help	Displays the help topic corresponding to error information.

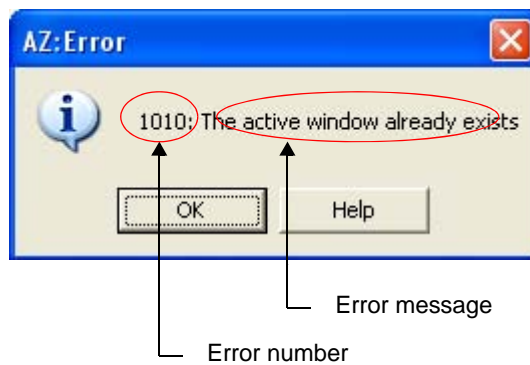
# APPENDIX B MESSAGES

Appendix B provides the error information output from AZ78K0R.

## B. 1 Overview

If an error occurs during an operation in a window/dialog box, the following **AZ:Error dialog box** that displays the error information is displayed.

Figure B-1. Error Information Output Format



## B. 2 Error Messages

AZ78K0R error information is shown below.

Table B-1. Error Information List

Error Number	Description	
1000	Message	Not enough memory.
	Cause	The memory required for AZ78K0R operation is insufficient.
	Action by User	Close unnecessary application software and then restart AZ78K0R.
1001	Message	Internal error.
	Cause	An error has occurred in internal AZ78K0R processing.
	Action by User	Restart AZ78K0R.

Error Number	Description	
1010	Message	The active window already exists.
	Cause	A window in the Active mode is open.
	Action by User	In AZ78K0R, multiple windows of the same type cannot be open in the Active mode at the same time. Implement either of the following measures. <ul style="list-style-type: none"> <li>- Change the mode of the window from Active to Hold.</li> <li>- Close the window in the Active mode.</li> </ul>
1021	Message	The file does not exist.
	Cause	No relevant files exist in the folder selected in the <a href="#">Open/Save As dialog box</a> .
	Action by User	Specify an existing file.
1022	Message	Fail to write the file.
	Cause	The memory for writing to the file is insufficient, or the file subject to write is write-prohibited.
	Action by User	Implement either of the following measures. <ul style="list-style-type: none"> <li>- Delete unnecessary files.</li> <li>- Specify another partition for writing.</li> <li>- Change the attribute of the target file so as to enable writing.</li> <li>- Specify a write-enabled file as the file subject to write.</li> </ul>
1023	Message	The file format is illegal.
	Cause	A file of an invalid format was selected in the <a href="#">Open/Save As dialog box</a> .
	Action by User	In AZ78K0R, the format of loadable files varies depending on the window through which the <a href="#">Open/Save As dialog box</a> is opened. <p><a href="#">AZ:Analyze Window</a> -&gt; Binary</p> <p><a href="#">AZ:Cpu Window</a> -&gt; CSV</p>
1100	Message	The debugger does not support AZ interface.
	Cause	CubeSuite does not support AZ Interface.
	Action by User	Confirm that CubeSuite supports AZ Interface.
1110	Message	Fail to switch AZ trace mode.
	Cause	AZ78K0R failed communication with CubeSuite.
	Action by User	Restart AZ78K0R and CubeSuite.
1120	Message	Fail to load the trace data.
	Cause	AZ78K0R failed communication with CubeSuite.
	Action by User	Restart AZ78K0R and CubeSuite.
1121	Message	The trace data does not exist.
	Cause	No information that should be stored in the trace memory exists in the trace data collect section.
	Action by User	Expand the trace data collect section and retry collection of trace data.

# APPENDIX C INDEX

## A

About dialog box ... 66  
AZ:Analyze Window ... 31  
AZ:Cpu Window ... 52  
AZ:Error dialog box ... 67  
AZ:Trace View Window ... 57  
AZ78K0R  
    Debugging procedure ... 16  
    Features ... 13  
AZ78K0R window ... 23

## D

Down cursor ... 34

## E

Error information ... 68  
    Output format ... 68  
Execution transition map ... 35

## F

Functions ... 14

## O

Open/Save As dialog box ... 27

## P

Pattern Search dialog box ... 48

## S

Search condition ... 49  
Search mode ... 49

## T

Trace data ... 14  
    Time accuracy ... 15  
Trace form ... 14  
    Hard trace form ... 14  
Trace Search dialog box ... 62

## U

Up cursor ... 34  
Up temporary cursor ... 34

## W

Window reference ... 21  
    About dialog box ... 66  
    AZ:Analyze Window ... 31  
    AZ:Cpu Window ... 52  
    AZ:Error dialog box ... 67  
    AZ:Trace View Window ... 57  
    AZ78K0R window ... 23  
    Open/Save As dialog box ... 27  
    Pattern Search dialog box ... 48  
    Trace Search dialog box ... 62

**[MEMO]**

*For further information,  
please contact:*

**NEC Electronics Corporation**  
1753, Shimonumabe, Nakahara-ku,  
Kawasaki, Kanagawa 211-8668,  
Japan  
Tel: 044-435-5111  
<http://www.necel.com/>

**[America]**

**NEC Electronics America, Inc.**  
2880 Scott Blvd.  
Santa Clara, CA 95050-2554, U.S.A.  
Tel: 408-588-6000  
800-366-9782  
<http://www.am.necel.com/>

**[Europe]**

**NEC Electronics (Europe) GmbH**  
Arcadiastrasse 10  
40472 Düsseldorf, Germany  
Tel: 0211-65030  
<http://www.eu.necel.com/>

**Hanover Office**

Podbielskistrasse 166 B  
30177 Hannover  
Tel: 0 511 33 40 2-0

**Munich Office**

Werner-Eckert-Strasse 9  
81829 München  
Tel: 0 89 92 10 03-0

**Stuttgart Office**

Industriestrasse 3  
70565 Stuttgart  
Tel: 0 711 99 01 0-0

**United Kingdom Branch**

Cygnus House, Sunrise Parkway  
Linford Wood, Milton Keynes  
MK14 6NP, U.K.  
Tel: 01908-691-133

**Succursale Française**

9, rue Paul Dautier, B.P. 52  
78142 Velizy-Villacoublay Cédex  
France  
Tel: 01-3067-5800

**Sucursal en España**

Juan Esplandiú, 15  
28007 Madrid, Spain  
Tel: 091-504-2787

**Tyskland Filial**

Täby Centrum  
Entrance S (7th floor)  
18322 Täby, Sweden  
Tel: 08 638 72 00

**Filiale Italiana**

Via Fabio Filzi, 25/A  
20124 Milano, Italy  
Tel: 02-667541

**Branch The Netherlands**

Steijgerweg 6  
5616 HS Eindhoven  
The Netherlands  
Tel: 040 265 40 10

**[Asia & Oceania]**

**NEC Electronics (China) Co., Ltd**  
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian  
District, Beijing 100083, P.R.China  
Tel: 010-8235-1155  
<http://www.cn.necel.com/>

**Shanghai Branch**

Room 2509-2510, Bank of China Tower,  
200 Yincheng Road Central,  
Pudong New Area, Shanghai, P.R.China P.C:200120  
Tel:021-5888-5400  
<http://www.cn.necel.com/>

**Shenzhen Branch**

Unit 01, 39/F, Excellence Times Square Building,  
No. 4068 Yi Tian Road, Futian District, Shenzhen,  
P.R.China P.C:518048  
Tel:0755-8282-9800  
<http://www.cn.necel.com/>

**NEC Electronics Hong Kong Ltd.**

Unit 1601-1613, 16/F., Tower 2, Grand Century Place,  
193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: 2886-9318  
<http://www.hk.necel.com/>

**NEC Electronics Taiwan Ltd.**

7F, No. 363 Fu Shing North Road  
Taipei, Taiwan, R. O. C.  
Tel: 02-8175-9600  
<http://www.tw.necel.com/>

**NEC Electronics Singapore Pte. Ltd.**

238A Thomson Road,  
#12-08 Novena Square,  
Singapore 307684  
Tel: 6253-8311  
<http://www.sg.necel.com/>

**NEC Electronics Korea Ltd.**

11F., Samik Lavied'or Bldg., 720-2,  
Yeoksam-Dong, Kangnam-Ku,  
Seoul, 135-080, Korea  
Tel: 02-558-3737  
<http://www.kr.necel.com/>