

**Customer Notification**

**ID78K0R-QB™**

**78K0R Integrated Debugger**

**Operating Precautions**

---

## **DISCLAIMER**

The related documents in this customer notification may include preliminary versions. However, preliminary versions may not have been marked as such.

The information in this customer notification is current as of its date of publication. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC's data sheets or data books, etc., for the most up-to-date specifications of PRODUCT(S). Not all PRODUCT(S) and/or types are available in every country. Please check with an NEC sales representative for availability and additional information.

No part of this customer notification may be copied or reproduced in any form or by any means without prior written consent of NEC. NEC assumes no responsibility for any errors that may appear in this customer notification. NEC 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 PRODUCT(S) listed in this customer notification or any other liability arising from the use of such PRODUCT(S).

No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC or others. Descriptions of circuits, software and other related information in this customer notification are provided for illustrative purposes of PRODUCT(S) operation and/or application examples only. The incorporation of these circuits, software and information in the design of customer's equipment shall be done under the full responsibility of customer. NEC assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.

While wherever feasible, NEC endeavors to enhance the quality, reliability and safe operation of PRODUCT(S) the customer agrees and acknowledges that the possibility of defects and/or erroneous thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects and/or errors in PRODUCT(S) the customer must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.

The customer agrees to indemnify NEC against and hold NEC harmless from any and all consequences of any and all claims, suits, actions or demands asserted against NEC made by a third party for damages caused by one or more of the items listed in the enclosed table of content of this customer notification for PRODUCT(S) supplied after the date of publication.

PRODUCT(S) are classified into the following three quality grades: "Standard", "Special" and "Specific". The "Specific" quality grade applies only to PRODUCT(S) developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of PRODUCT(S) depend on its quality grade, as indicated below. Customers must check the quality grade of each PRODUCT(S) 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 PRODUCT(S) is "Standard" unless otherwise expressly specified in NEC data sheets or data books, etc. If customers wish to use PRODUCT(S) in applications not intended by NEC, they must contact NEC sales representative in advance to determine NEC's willingness to support a given application.

If the supplied goods/information are subject to Japanese, German, European and/or North American export controls, the customer shall comply with the relevant export control regulations in the event that the goods are exported and/or re-exported. If deliveries are exported without payment of duty at the request of the customer, the customer accepts liability for any subsequent customs administration claims with respect to NEC.

- Notes:**
- (1) "NEC" as used in this statement means NEC Electronics Corporation and also includes its direct or indirect owned or controlled subsidiaries.
  - (2) "PRODUCT(S)" means any product developed or manufactured by or for NEC (as defined above).

## Table of Contents

(A)	Table of Operating Precautions .....	4
(B)	ID78K0R-QB V3.50 Operating Environment.....	7
(C)	Description of Operating Precautions.....	8
(D)	Valid Specification.....	30
(E)	Revision History .....	31

**(A) Table of Operating Precautions**

No.	Outline	Rev. <sup>Note</sup>	ID78K0R-QB			
		Version	V3.20	V3.30	V3.40	V3.50
1	Cautions on Installation (Direction of Use)		x	x	x	x
2	Cautions on using ID78K0R-QB (Direction of Use)		x	x	x	x
3	Cautions related to ID78K0R-QB (Direction of Use)		x	x	x	x
4	Caution when IECUBE QB-78K0R- ZZZ-EE is connected (Direction of Use)		x	✓	✓	✓
5	Cautions when MINICUBE2 QB- MINI2-EE is connected (Direction of Use)		x	✓	✓	✓
6	Redraw of Memory and Coverage window (Technical Limitation)		x	x	✓	✓
7	Symbols in format func#var (Technical Limitation)		x	x	x	x
8	Search operation in Assemble window (Technical Limitation)		x	x	✓	✓
9	Refresh button in Watch window (Technical Limitation)		x	✓	✓	✓
10	Extended search over search- prohibited area (Technical Limitation)		x	x	✓	✓
11	Values displayed in the general- purpose register area (Technical Limitation)		x	✓	✓	✓
12	IECUBE: Hardware break before execution (Technical Limitation)		x	✓	✓	✓
13	IECUBE: Coverage function (Technical Limitation)		x	✓	✓	✓
14	MINICUBE2: Invalid operation in specific stack area (Technical Limitation)		x	✓	✓	✓
15	MINICUBE2: Break on odd address in byte units (Technical Limitation)		x	✓	✓	✓
16	MINICUBE2: Software break (Technical Limitation)		x	✓	✓	✓
17	Access break (Technical Limitation)		x	x	✓	✓
18	Adding SFRs in Add I/O Port dialog box (Technical Limitation)		x	x	✓	✓

Operating Precautions for ID78K0R-QB™

No.	Outline	ID78K0R-QB				
		Rev. <sup>Note</sup>				
		Version	V3.20	V3.30	V3.40	V3.50
19	IECUBE: Rewriting of PSW bits in DMM dialog box (Technical Limitation)		x	x	✓	✓
20	Changing font size (Technical Limitation)		x	x	✓	✓
21	Moving via Address Move dialog box (Technical Limitation)		x	x	✓	✓
22	Pass name length (Technical Limitation)		x	x	✓	✓
23	Return Out in Source window (Technical Limitation)		x	x	✓	✓
24	Addition of –verbose option to breakpoint command (Specification Change)		—	x	x	x
25	IECUBE: Addition of change of coverage color (Specification Change)		—	x	x	x
26	MINICUBE2: Addition of monitor operation clock select function (Specification Change)		—	x	x	x
27	Addition of function for stack trace window and where command (Specification Change)		—	—	x	x
28	BCD correction result register (Technical Limitation)		x	x	✓	✓
29	IECUBE: Code coverage window (Technical Limitation)		x	x	✓	✓
30	External bus access (Technical Limitation)		x	x	x	x
31	MINICUBE2: Restriction whereby a break occurs at a specific SP value (Technical Limitation)		x	x	x	x
32	MINICUBE2: Restriction on reset during RRM or DMM processing (Technical Limitation)		x	x	x	x
33	MINICUBE2: DMM dialog box in conjunction with support of DMM function (Specification Change)		—	—	—	x
34	MINICUBE2: RRM dialog box in conjunction with support of RRM function (Specification Change)		—	—	—	x

Operating Precautions for ID78K0R-QB™

No.	Outline	ID78K0R-QB			
		Rev. <sup>Note</sup>			
		Version	V3.20	V3.30	V3.40
35	MINICUBE2: Addition of functions to Extended Option dialog box in conjunction with support of RRM function in 1-wire mode (Specification Change)	—	—	—	✕
36	MINICUBE2: Addition of functions to Configuration dialog box in conjunction with support of flash-programming at low voltage (to 78K0R/Kx3-L only) (Specification Change)	—	—	—	✕
37	MINICUBE2: VECT display in Trace View Window (Specification Change)	—	—	—	✕
38	Addition of functions related to the open break function (to 78K0R/lx3 only) (Specification Change)	—	—	—	✕

- ✓: Not applicable
- ✕: Applicable
- : Not available

## (B) ID78K0R-QB V3.50 Operating Environment

### Host machine

Use ID78K0R-QB V3.50 in a host machine that satisfies the following requirements:

- CPU: Pentium IITM 400 MHz or above
- Main memory: 256 MB or more
- USB interface: 1.1/2.0
- OS<sup>Note</sup>: Windows 2000 or Windows XP (Home Edition/Professional)
- Monitor (display): When using multiple monitors (two or more displays), use the monitor assigned to monitor 1 (primary monitor).

**Note:** Installation of the latest service pack for each OS is recommended.

### Supported emulators

ID78K0R-QB V3.50 supports the following emulators:

- IECUBE: QB-78K0RKX3 control code L and later  
QB-78K0RIX3 control code C and later
- MINICUBE2: QB-MINI2 control code A and later, firmware V4.05 and later

**(C) Description of Operating Precautions**

No. 1	Cautions on Installation (Direction of Use)
	<p><u>1 Cautions on installation</u></p> <p>(1) The ID78K0R-QB supports multiple-version installation. Multiple-version installation allows multiple versions of a product in one host machine. It is basically recommended to use the latest version of a development tool, but there may be cases when the previous development environment should be left as is, or when code generation will be changed due to the development tool upgrade, which may cause problems. Multiple-version installation solves these problems by allowing coexistence between the previous development environment and the latest one. These environments can be switched easily. If multiple versions have been installed, set the version to be used on the [Tool Version Settings] tab of the Project Settings dialog box that appears by choosing "Project Settings..." on the PM+ [Project] menu.</p> <p>(2) Because it may be necessary to restart the computer after installation, terminate all other applications.</p> <p>(3) Log on Windows as the administrator user to install the ID78K0R-QB.</p> <p>(4) The ID78K0R-QB can be installed in a folder whose name uses ASCII characters only (excluding eleven characters /*:&lt;&gt;? "\\,;) and does not start or end with a space. If any other characters are used, the tools included in ID78K0R-QB V3.30 may not operate normally.</p> <p>(5) Installation to/from a network drive is not possible.</p> <p>(6) The installer does not set the environment variable path. Set the environment variable path after installation, as necessary.</p> <p>(7) The help is supplied in the HTML help format. If the help does not operate correctly, proceed as follows:</p> <ul style="list-style-type: none"> <li>• Install Microsoft Internet Explorer 5.0 or later (IE5.5 SP2 or later recommended)</li> </ul> <p><u>2. Installation procedure</u></p> <p>This section explains the installation procedure, assuming that the ID78K0R-QB is installed under the folder "C:\Program Files\NEC Electronics Tools"</p> <p>(1) Turn on power to the host machine and start Windows.</p> <p>(2) Insert the CD of the ID78K0R-QB into the CD drive. The installer is then started automatically. If it does not start automatically, execute "INSTALL.exe" on the CD-ROM from Windows Explorer.</p> <p>If your ID78K0R-QB is the one downloaded from the Development Tools Download page on the NEC Electronics website, run <i>id78k0r-qb_v330_e.exe</i>.</p> <p>(3) Follow the directions on the screen to continue installation.</p> <p>(4) The following files will be created after the ID78K0R-QB has been installed. These file are required for uninstalling the ID78K0R-QB and must not be deleted:</p> <p>For ID78K0R-QB and documents: C:\Program Files\NEC Electronics Tools\ID78K0R-QB\version\SETUP\*.*</p> <p>For USB driver for IECUBE: C:\Program Files\NEC Electronics Tools\ID78K0R-QB\ieqb78k0r\*.*</p> <p>For USB driver for MINICUBE2: C:\Program Files\NEC Electronics Tools\mqb2all\*.*</p> <p>For MINICUBE utilities and documents: C:\Program Files\NEC Electronics Tools\MINICUBE Utilities\version\setup\*.*</p>



No. 2	Cautions on using ID78K0R-QB (Direction of Use)
	<p><u>1. Device File</u></p> <p>To install the device file, use the installer “DFINST.EXE” supplied with the ID78K0R-QB or other products. When installation is complete, the device file installer (DFINST) is registered in the Start menu, with the name “DeviceFile Installer”.</p> <p>The procedure for installing the device file is explained below. Refer to the online help provided in DFINST for details on DFINST.</p> <ul style="list-style-type: none"> <li>• Preparation of device file Download the device file from the NEC Electronics website (Development Tools Download page). Decompress the downloaded file into an arbitrary folder.</li> </ul> <p>URL: <a href="http://www.eu.necel.com/update/">http://www.eu.necel.com/update/</a></p> <ul style="list-style-type: none"> <li>• Installation of device file Start DFINST from the Start menu and click the [Install] button on the initial screen. Select either of the install information files from the above folder in accordance with the language of the OS used and open it.</li> </ul> <p style="padding-left: 40px;">_csetup.ini (English version) Necsetup.ini (Japanese version)</p> <p>To uninstall the device file, select a device from those displayed in the “Registry” field in DFINST, select the “Delete File” check box, and then click the [UnRegister] button.</p> <p><u>2. USB Driver</u></p> <p>A USB driver is required to execute the ID78K0R-QB. The USB driver can be installed when installing the ID78K0R-QB. This USB driver is not installed by the Windows Plug&amp;Play function, so do not connect the emulator and the host machine until installation is complete. When installation is completed, connect the emulator to the host machine. The following explanation shows an example when IECUBE is connected.</p>

The following window appears and then the “Found New Hardware Wizard” starts. Select “No, not this time” and click the [Next] button.



The following window appears. Select “Install the software automatically (Recommended)” and click the [Next] button.

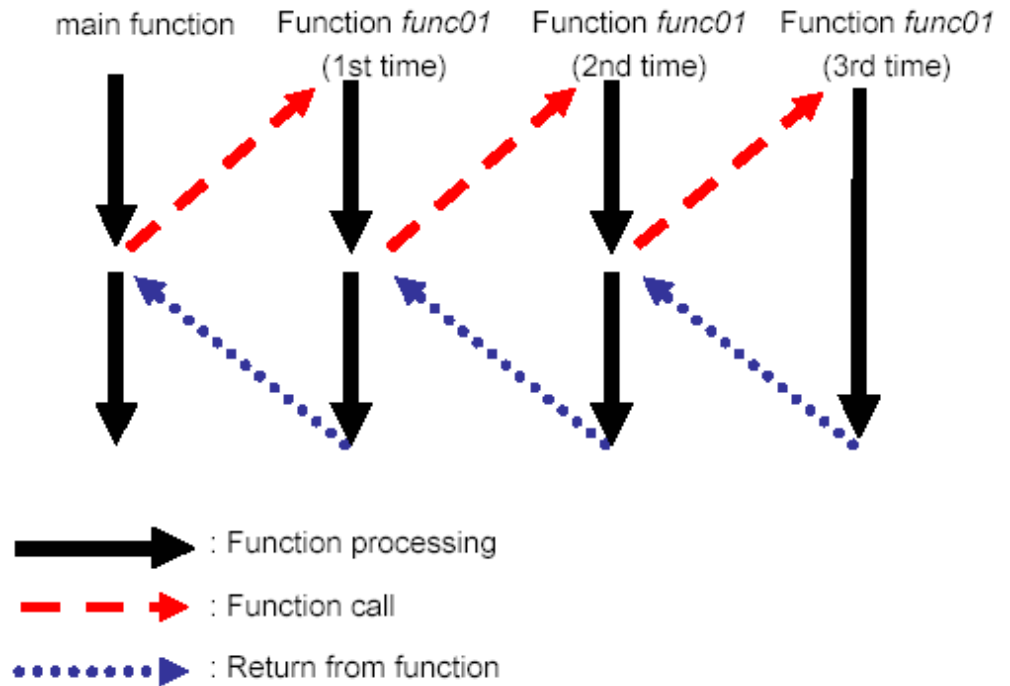


The installation wizard ends with the following window.



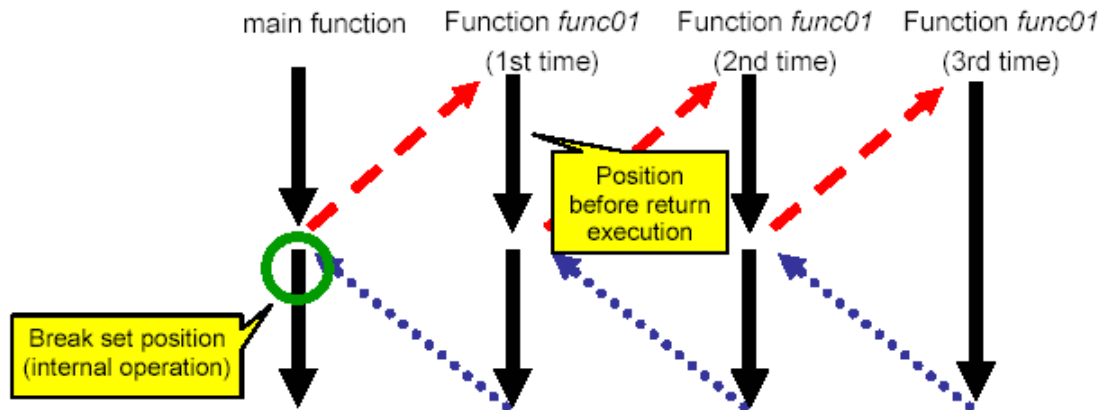
No. 3	Cautions related to ID78K0R-QB (Direction of Use)								
<p><u>1. Cautions on switching log-on user of Windows XP</u> If a log-on user is switched using the Windows XP Switch User function while the ID78K0R-QB is starting, the operation of the ID78K0R-QB that was used by the previous user is not guaranteed. The operation of the ID78K0R-QB used by the new user is also not guaranteed. (The ID78K0R-QB prohibits duplicated activation on one computer, but duplicated activation is possible by using the Windows XP Switch User function.)</p> <p><u>2. Cautions related to Windows standby and hibernate modes</u> If Windows enters standby or hibernate mode while the ID78K0R-QB is starting, the operation of the ID78K0R-QB is not guaranteed after Windows returns from the mode. In such a case, restart the emulator and ID78K0R-QB.</p> <p><u>3. Cautions on Return Out execution for recursive functions</u> If Return Out is executed during processing of a recursive function (function that calls itself), the execution cannot return to the desired position. The operation varies as follows according to how many times the recursive function is called.</p> <table border="1" data-bbox="344 826 1441 1081"> <thead> <tr> <th data-bbox="344 826 895 891">Current PC Line Before Return Out Execution</th> <th data-bbox="895 826 1441 891">Return Out Execution Result</th> </tr> </thead> <tbody> <tr> <td data-bbox="344 891 895 956">During processing of the 1st recursive function</td> <td data-bbox="895 891 1441 956">Returns to the position at which the function was called (desired operation).</td> </tr> <tr> <td data-bbox="344 956 895 1021">During processing of the 2nd to nth recursive function</td> <td data-bbox="895 956 1441 1021">Breaks during processing of the nth recursive function</td> </tr> <tr> <td data-bbox="344 1021 895 1081">After processing of the nth recursive function ends</td> <td data-bbox="895 1021 1441 1081">Returns to the position at which the function was called (desired operation).</td> </tr> </tbody> </table>		Current PC Line Before Return Out Execution	Return Out Execution Result	During processing of the 1st recursive function	Returns to the position at which the function was called (desired operation).	During processing of the 2nd to nth recursive function	Breaks during processing of the nth recursive function	After processing of the nth recursive function ends	Returns to the position at which the function was called (desired operation).
Current PC Line Before Return Out Execution	Return Out Execution Result								
During processing of the 1st recursive function	Returns to the position at which the function was called (desired operation).								
During processing of the 2nd to nth recursive function	Breaks during processing of the nth recursive function								
After processing of the nth recursive function ends	Returns to the position at which the function was called (desired operation).								
<p>Even if the execution fails to return correctly, the subsequent program execution is performed correctly.</p> <p>The following shows an operation example.</p> <pre data-bbox="320 1267 938 1664"> void func01(char count) {     count--;     if (count &gt; 0)     {         func01(count); /* 2nd and 3rd call */     } }  void main(void) {     func01(3); /* 1st call */ } </pre> <p>In this example, "func01" is the recursive function.</p>									

The above program flows as follows.



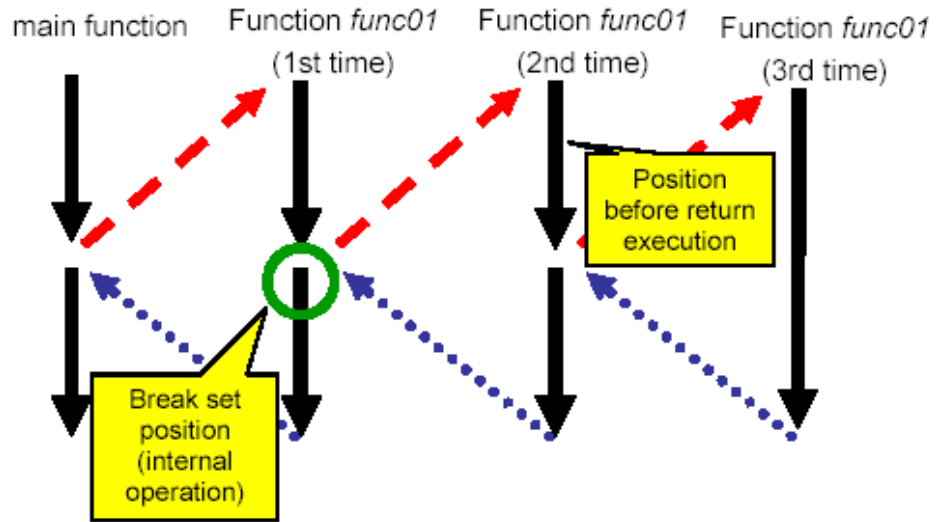
The following explains the Return Out operation in detail.

**[Return Out during processing of the 1st recursive function]**



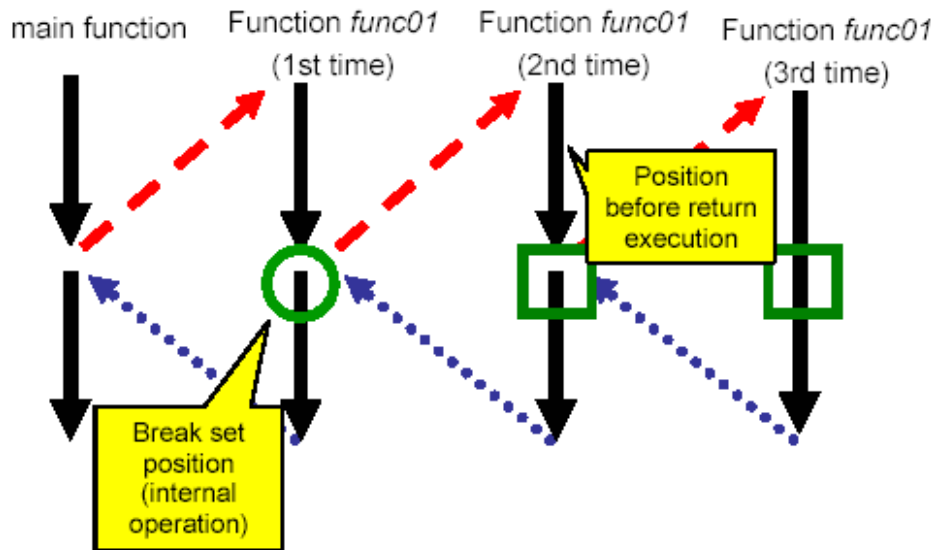
If Return Out is executed during the first processing of function *func01*, the execution returns to the circle-marked position. The ID78K0R-QB sets a breakpoint at this position internally and executes the program. In this case, the program will stop at the desired position (circle-marked position).

**[Return Out during processing of the 2nd to *n*th recursive function (1)]**



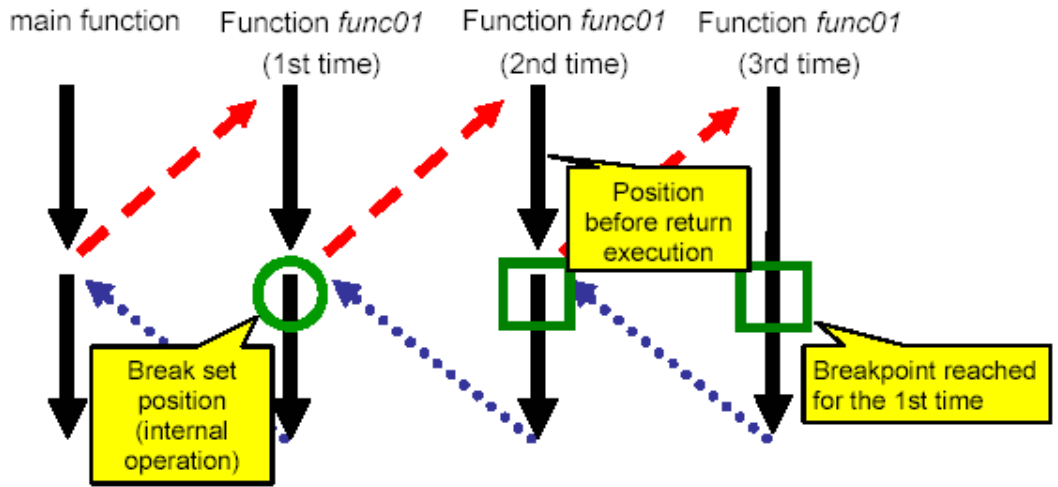
If Return Out is executed during the second and third processing of function *func01*, as in the above case, the ID78K0R-QB sets a breakpoint at the desired return position (circle-marked position).

**[Return Out during processing of the 2nd to *n*th recursive function (2)]**



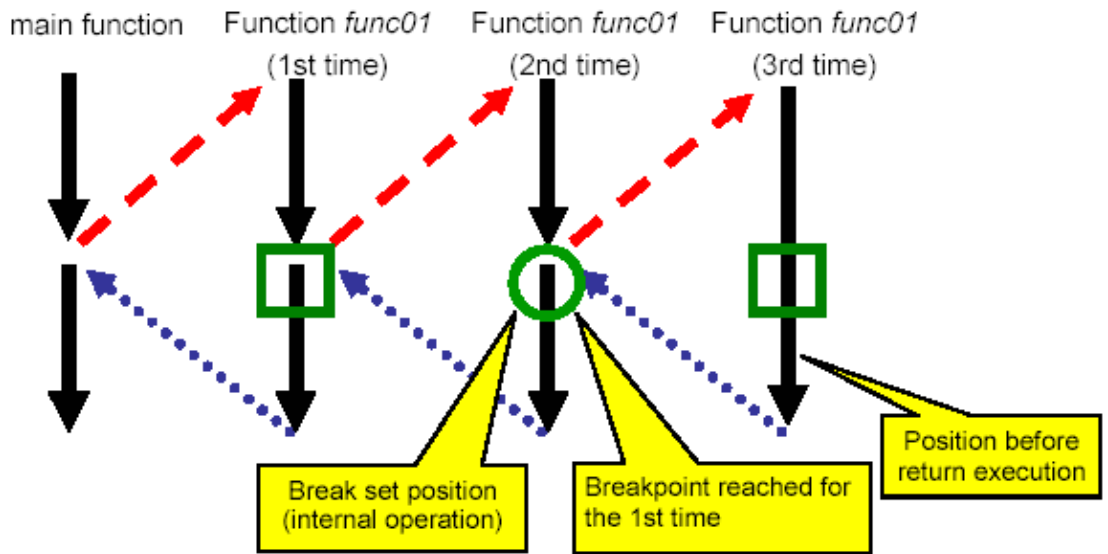
However, the set breakpoint is also included in the recursive processing, so breaks also occur at the square-marked positions.

**[Return Out during processing of the 2nd to *n*th recursive function (3)]**



As a result of program execution, a break occurs at the first breakpoint (the rightmost square-marked position in the above figure), which is not the desired position. This is because the execution passes through the desired position before reaching the desired position. Although it is not the desired result, program execution is performed as is coded. Even if the program is reexecuted from this position, the operation results are normal.

**[Return Out after processing of the *n*th recursive function ends]**



If Return Out is executed after the third *func01* processing (the rightmost square-marked position in the above figure), it is desired to return to the circle-marked position, but a breakpoint is also set at the square-marked position. This execution breaks at the desired position, however. This is because the first breakpoint to be reached during recursive processing is the circle-marked position.

4. Caution when a function entity is defined in header file

If a function entity is defined in the header file, breakpoint setting, step-wise execution, or return out cannot be performed in the Source window. Moreover, information on this function is not displayed in the Code Coverage window.

The case where “a function entity is defined in the header file” means that function processing itself is described in the header file. It does not include the case where only function declaration, such as prototype declaration and EXTERN declaration, is described in the header file.

5. Caution on longjmp function

If Step In or Next Over is performed for the *longjmp* function, execution processing may not complete and may wait for a time-out. If Return Out is performed in a function that called the *longjmp* function, breaks may not occur.

6. Standby mode

The standby mode is released under any of the following conditions.

- The standby mode is entered during step execution.
- Execution is stopped by a forced break in standby mode
- A standby in struction is executed before the execution stops due to a break after execution or an access break, which takes several extra instructions until the execution stops.
- A break after execution or an access break is used as an event of a snap-shot or event DMM, a standby instruction is executed before the execution stops due to the break, each processing is complete, and then the execution is restarted. (IECUBE).

7. Watch window

When a variable is referenced in the Watch window during program execution, the data currently being changed may be displayed in the timing of the variable change.

8. Software break in RAM area

Do not overwrite the program in which a software break has been set to the internal low-speed RAM area or external RAM area; otherwise, the break may not occur normally. Even if the break occurs, the program before being overwritten is automatically restored. Therefore, use a software break to set a break to a program in the RAM area.

9. Breaks before execution

Breaks before execution can be set to the internal ROM only.

10. Mirrored non-map area

A mirrored non-map area is not indicated as “?”. If IROM = 64 KB and MAA = 1, for example, the non-map area starting from address 0x10000 is mirrored. At this time, 0x10000 to 0xF0FFF are indicated as “?” in the Memory window, but values can be read from or written to 0xF1000 to 0xFFEFF.



11. External memory

- The code coverage cannot be measured and memory accessing cannot be monitored (IECUBE).
- When the RAM monitor function is off, data cannot be read from or written to the external RAM during execution (IECUBE).
- The following two restrictions apply when a ROM is connected as an external memory (IECUBE, MINICUBE2).
  - (1) If step execution is performed for the following instructions, an extra instruction is executed.
    - a. RETI/RETB instruction.  
An instruction immediately after returning from the interrupt servicing will be executed.
    - b. Conditional skip instruction (condition is not satisfied)  
The instruction next to the conditional skip instruction will be executed.
  - (2) Execution does not stop as is expected if one of the following operations is performed. The execution continues, or stops at the next breakpoint.
    - a. The first step execution in main( )
    - b. Step In
    - c. Next Over
    - d. Return Out
    - e. Come Here

No. 4	Caution when IECUBE QB-78K0R-ZZZ-EE is connected (Direction of Use)
	<p><u>1. Execution operation while external reset signal is input</u> When the SFR window or the like is displayed and execution or step execution is performed while an external reset is not masked (the TARGET RESET check box is not selected in the Configuration dialog box) and an external reset signal has been input, the program has to wait for a timeout in communication with the emulator, which drastically degrades the operability.</p> <p><u>2. Coverage function</u></p> <ul style="list-style-type: none"> <li>• The PC points to the instruction that has not been executed, but for code coverage measurement, one or two bytes from the top of the instruction pointed to by the PC are counted by the code coverage execution. In addition, the address at which a break occurs is also counted.</li> <li>• Display of an access being executed does not necessarily change in the order of access in the Memory window.</li> </ul> <p><u>3. Display of internal ROM area during execution</u> During execution, data in the internal ROM area before starting execution is saved temporarily and displayed. The display is therefore not updated until the execution breaks, even if the data is overwritten by flash self-programming.</p> <p><u>4. Events</u> Up to two extra instructions after a snap-shot event, event DMM, trace event, or timer event is measured.</p> <p><u>5. Values of the PSW bits cannot be changed in DMM dialog box</u> If an attempt is made to change the value of the PSW bits (ie, z, rbs1, ac, rbs0, isp, or cy) in the DMM dialog box, the error message "A9004: Too large register size." will be displayed. Change the value of the PSW.</p>
No. 5	Cautions when MINICUBE2 QB-MINI2-EE is connected (Direction of Use)
	For cautions to be observed when connecting MINICUBE2 QB-MINI2-EE, pls. refer to the User's Manual and the Operating Precaution document of the MINICUBE2 QB-MINI2-EE.
No. 6	Redraw of Memory and Coverage window (Technical Limitation)
	<p><u>Details</u> If the caret is moved when the Memory window or Coverage window is activated, the line on which the caret was placed cannot be redrawn normally.</p> <p><u>Workaround</u> Redraw can be performed by moving the caret to a line on which redraw was not performed.</p>

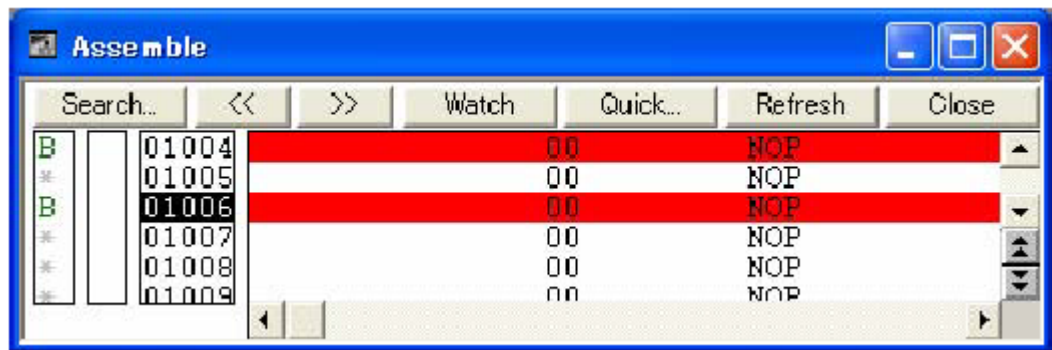
No. 7	Symbols in format <i>func#var</i> (Technical Limitation)
	<p><u>Details</u> Symbols with a <i>func#var</i> (<i>func</i>: Function name, <i>var</i>: Variable name) format cannot be converted into addresses in the Symbol To Address dialog box, if the conversion targets are variables.</p> <p><u>Workaround</u> Convert such symbols using a method other than <i>func#var</i>, such as <i>var</i>, <i>file#var</i>, or <i>file#func#var</i> format). When there is a variable with the same name as a static variable in a function, convert the variable when the PC exists in that function.</p>
No. 8	Search operation in Assemble window (Technical Limitation)
	<p><u>Details</u> Search may not be performed for the last 0x400 spaces of the range subject to search in the Assemble window. For example, if the range 0x1000 to 0x1FFF is searched for toward the down direction and a character string that matches the searched target exists in the range 0x1C00 to 0x1FFF, the search may fail. In the same manner, if the range 0x2000 to 0x2FFF is searched for toward the up direction, the search in the range 0x2000 to 0x23FF may fail.</p> <p><u>Workaround</u> Add 0x400 spaces to the search target range. (To search a character string in the range 0x1000 to 0x1FFF toward the down direction, for example, specify 0x1000 to 0x23FF.)</p>
No. 9	Refresh button in Watch window (Technical Limitation)
	<p><u>Details</u> If an item at the bottom in the Watch window is deleted, the Refresh button appears dimmed.</p> <p><u>Workaround</u> Close the Watch window, and then open it again.</p>
No. 10	Extended search over search-prohibited area (Technical Limitation)
	<p><u>Details</u> If an area for which searching is prohibited (non-map area, SFR area, or I/O protect area) is included in the search target specified in the Memory Search dialog box and Assemble Search dialog box, search is not performed for any area later than the area for which search is prohibited.</p> <p><u>Workaround</u> There is no workaround.</p>

No. 11	Values displayed in the general-purpose register area (Technical Limitation)
<p><u>Details</u> Invalid values are displayed in the general-purpose register area.</p> <p><u>Workaround</u> Use the Register window.</p>	

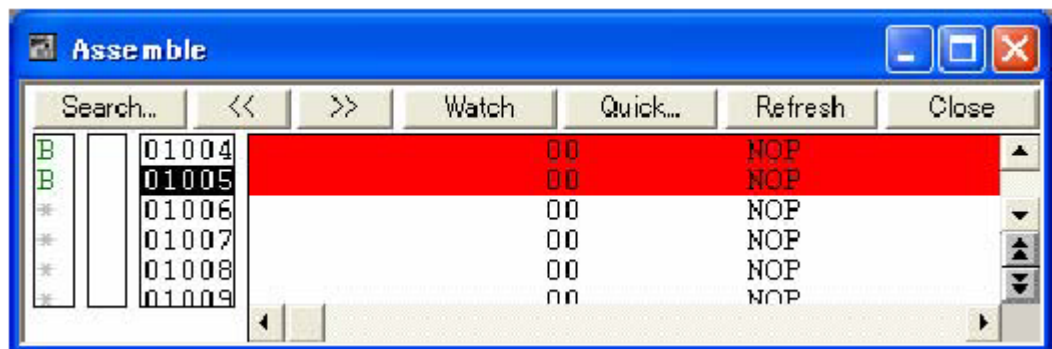
No. 12	IECUBE: Hardware break before execution (Technical Limitation)
--------	--

Details  
If two or more break before execution points are set within a double-word boundary area, the breaks do not occur as set. The following two cases are assumed.

Example 1: If break before execution is set at addresses 1004 and 1006, in that order, a break does not occur at address 1006.



Example 2: If break before execution is set at addresses 1004 and 1005, in that order, and the breakpoint at address 1005 is deleted, the setting at address 1005 is still valid.



Workaround  
There is no workaround.  
Clear all of the breaks before execution within a double-word boundary area and set just one break before execution, or use software breaks instead.

<p>No. 13</p>	<p>IECUBE: Coverage function (Technical Limitation)</p>
	<p><u>Details</u> The note the following concerning the coverage function:</p> <ul style="list-style-type: none"> <li>(a) The coverage measurement function (C0 coverage) measures not only ROM fetch but also ROM read.</li> <li>(b) When the read access status is displayed by the access monitor function on the Memory window, the window displays not only the ROM read status but also the ROM fetch status.</li> <li>(c) The general-purpose register value displayed in the Memory window may become invalid after accessing of a RAM area to which the general-purpose register is assigned. In addition, the Register window display may be invalid during program execution.</li> </ul> <p><u>Workaround</u> There is no workaround.</p>
<p>No. 14</p>	<p>MINICUBE2: Invalid operation in specific stack area (Technical Limitation)</p>
	<p><u>Details</u> If the area from 0xFB00 to 0xFCFF is used as a stack area, the stacked data may become invalid.</p> <p><u>Workaround</u> There is no workaround.</p>
<p>No. 15</p>	<p>MINICUBE2: Break on odd address in byte units (Technical Limitation)</p>
	<p><u>Details</u> A break does not occur normally if an access break is set at an odd address in byte units.</p> <p><u>Workaround</u> There is no workaround.</p>

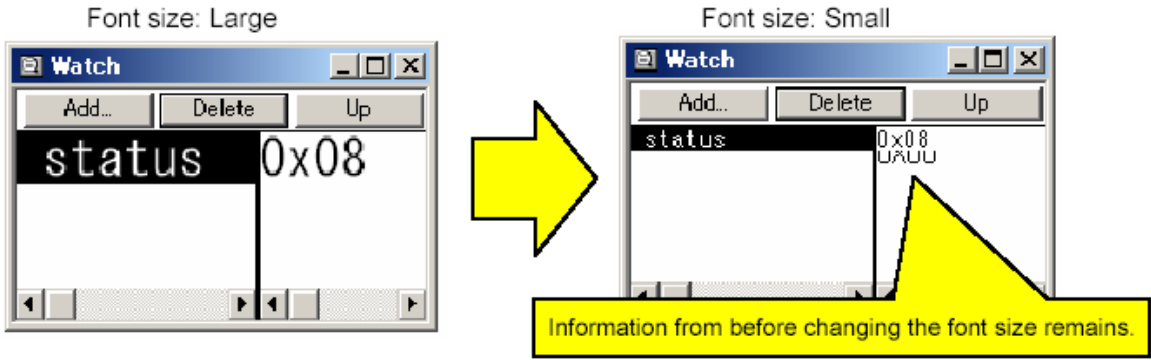
No. 16	<p>MINICUBE2: Software break (Technical Limitation)</p>
	<p><u>Details</u></p> <ul style="list-style-type: none"> <li>(1) If a software breakpoint is set to RAM, the first byte of the instruction becomes 0xff, and the rest of the bytes become 0x00.</li> <li>(2) If a software break is set at the last address of a 64 KB boundary area (such as 0xffff or 0x1ffff) and the software break is executed, the PC value at the break indicates the last address of the next 64 KB boundary area (0x1ffff or 0x2ffff, in the above case).</li> </ul> <p><u>Workaround</u> There is no workaround.</p>
No. 17	<p>Access break (Technical Limitation)</p>
	<p><u>Details</u></p> <ul style="list-style-type: none"> <li>(1) A variable other than <i>global</i> cannot be specified for an access break in the Watch window.</li> <li>(2) A variable other than <i>global</i> cannot be specified for an access break in the Source window.</li> </ul> <p><u>Workaround</u> To set an access break to a static variable in a function, perform the setting in the Event dialog box while the current PC line is in that function. To set an access break to a static variable in a file, perform the setting in the Event dialog box while the current PC line is in that file.</p>

No. 18	<p>Adding SFRs in Add I/O Port dialog box (Technical Limitation)</p>
	<p><u>Details</u></p> <p>The following applies when registering SFRs in the Add I/O Port dialog box (dialog box used when an SFR name is redefined with another name in a C source):</p> <ol style="list-style-type: none"> <li>(1) If a R/W-attribute SFR is registered as a W-attribute register with another name, the SFR is registered in the Add Watch window and its value is changed in the Watch window, values displayed in the SFR window are not updated.</li> <li>(2) Registration of SFRs for the 2nd SFR area is not available (if attempted, an error occurs.)</li> <li>(3) If an SFR that satisfies the following two conditions is registered, registration is possible but an error occurs when an attempt is made to display the registered SFR by using the [Move] command of the context menu (right-click menu) in the SFR window (move fails).             <ul style="list-style-type: none"> <li>• <input type="checkbox"/> The name of the SFR to be registered is used for a general-purpose register</li> <li>• <input type="checkbox"/> The address to be registered is the same as that of an SFR area (different from the address of the general-purpose register.)</li> </ul> </li> </ol> <p><u>Workaround</u></p> <ol style="list-style-type: none"> <li>(1) When changing values in the Watch window, input the value two or more times, or change the values in the SFR window.</li> <li>(2) There is no workaround.</li> <li>(3) There is no workaround.</li> </ol>
No. 19	<p>IECUBE: Rewriting of PSW bits in DMM dialog box (Technical Limitation)</p>
	<p><u>Details</u></p> <p>When changing register values in the DMM dialog box, the error message “A9004: Too large register size.” will be displayed if a PSW bit (IE, Z, RBS1, AC, RBS0, ISP1, ISP0, or CY) is specified in the [Register Name] field, but this error message is incorrect.</p> <p><u>Workaround</u></p> <p>To change the PSW values in the DMM dialog box, specify the PSW, not its bits. To change only the values of the PSW bits, change them in the Register window after the program is stopped.</p>

No. 20 Changing font size (Technical Limitation)

Details  
 The font size in the Watch window and Local Variable window can be specified in the [Font] area in the Debugger Option dialog box, which is opened by selecting the [Option] menu → [Debugger Option...], but if the font size is reduced, information from before changing the font size remains in the window.

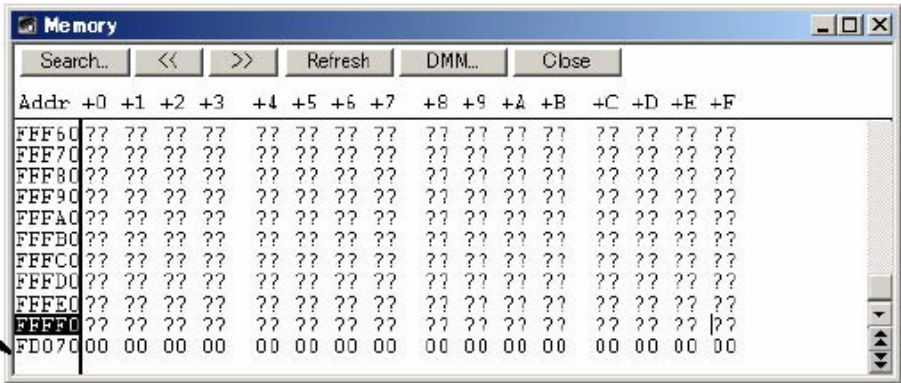
Example of Watch window:



Workaround  
 When such a case occurs, minimize the window once and then restore the window size, or close the window once and then open it again.

No. 21 Moving via Address Move dialog box (Technical Limitation)

Details  
 If the Address Move dialog box is opened by selecting the [Move] command from the context menu (right-click menu) in the Memory window and the last address (0xffff) is input, an extra line may be displayed.



Workaround  
 Do not move to address 0xffff. (This address is in the SFR area, so displaying this area in the Memory window is meaningless.)



No. 22	<p>Pass name length (Technical Limitation)</p>
	<p><u>Details</u> Windows supports path names (including file name) consisting of up to 259 single-byte characters, but the ID78K0R-QB cannot use 256 or more characters for path names. If 256 or more characters are used for path names, the operation becomes invalid. (For example, if the path for a load module file is too long, the file cannot be downloaded even if attempted, or if the path for a source file is too long, the source file cannot be opened.)</p> <p><u>Workaround</u> Specify path names using up to 255 characters, including the file name.</p>
No. 23	<p>Return Out in Source window (Technical Limitation)</p>
	<p><u>Details</u> Return Out is a function that performs execution until the execution returns to the address from which the function is called, and uses a stack to predict the call source. Due to this, execution cannot return from locations where the relation between the stack and the stack pointer (SP) is shifted (location where the stack is currently being manipulated, such as before execution of a function prologue ("{" or execution of an epilogue ("}") is in progress, or where the program counter points to the middle, not the beginning, of a C source). If Return Out is attempted at a location where Return Out cannot be performed, the following error message will be output.</p> <div data-bbox="406 958 1390 1249" data-label="Image"> </div> <p><u>Workaround</u> When the above error message is output, perform step-wise execution once and then Return Out. (Step-wise execution resolves the difference between the stack address and stack pointer (SP).)</p>

Operating Precautions for ID78K0R-QB™

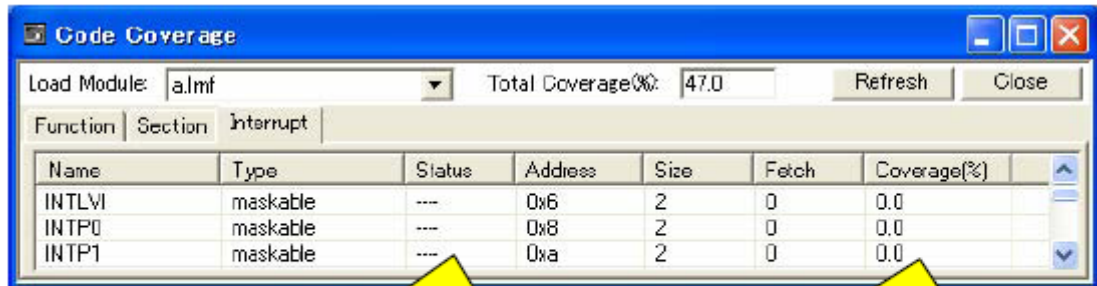
No. 24	Addition of <i>-verbose</i> option to breakpoint command (Specification Change)
	<p><u>Details</u> The <i>-verbose</i> option has been added to the breakpoint command.</p>
No. 25	IECUBE: Addition of change of coverage color (Specification Change)
	<p><u>Details</u> A function to change the coverage color has been added.</p>
No. 26	MINICUBE2: Addition of monitor operation clock select function (Specification Change)
	<p><u>Details</u> A function to switch the operation clock for the monitor program during a break from the subclock to main clock has been added to the Monitor Clock area in the Configuration dialog box.</p>
No. 27	Addition of function for stack trace window and where command (Specification Change)
	<p><u>Details</u> In the Stack window, the stack contents that may be incorrect now appear dimmed. In addition, for where command, the message “---Information below might be inaccurate.” is now displayed for the location that may be incorrect. If this message is displayed, the subsequent display may be incorrect.</p>
No. 28	BCD correction result register (Technical Limitation)
	<p><u>Details</u> When an add or subtract instruction is executed stepwise, the value in the BCD correction result register (BCDADJ) becomes invalid. In addition, the value in register A is not corrected in decimal units when using MINICUBE2.</p>

No. 29

IECUBE:  
Code coverage window  
(Technical Limitation)

Details

Value "0.0" is displayed in the [Coverage(%)] column on the [Interrupt] tab in the Code Coverage window, regardless of the occurrence of interrupts. In addition, "----" is displayed in the [Status] column on the [Interrupt] tab, regardless of the use of interrupt functions.



Display in [Status] column becomes "----".

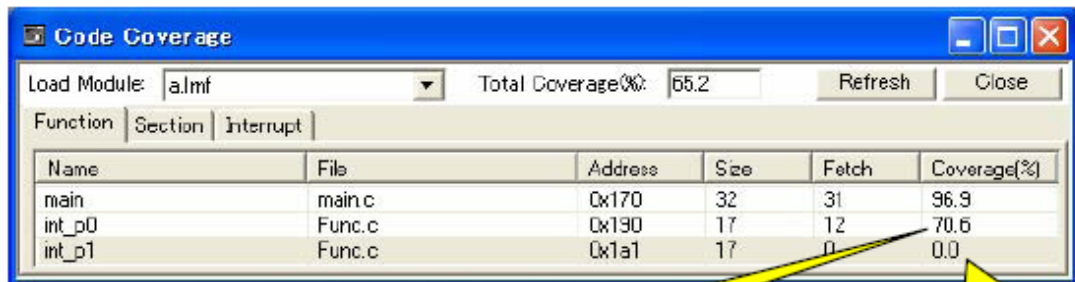
Display in [Coverage(%) column becomes "0.0".

Workaround:

The [Interrupt] tab is used to check the occurrence of interrupts, but the occurrence can also be checked by referring to the [Function] tab for the coverage of the function corresponding to the interrupt.

- When 0.0 is displayed in [Coverage(%)]: Interrupts have not occurred.
- When a value other than 0.0 is displayed in [Coverage(%)]: Interrupts have occurred.

Example: When interrupt INTPO is defined as function "int\_p0" and INTP1 as function "int\_p1":

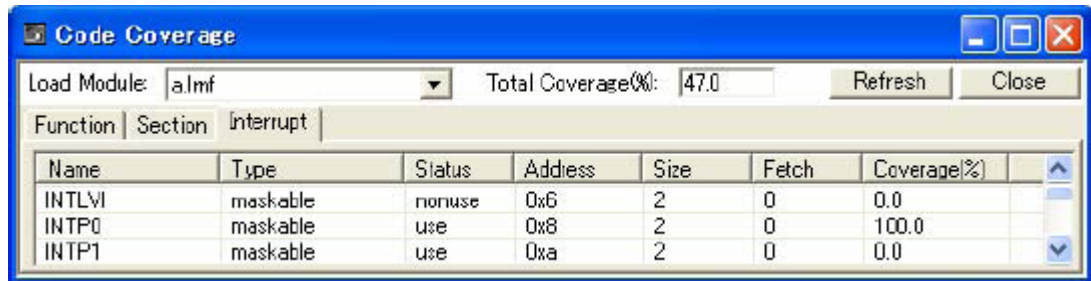


"0.0" is not displayed in [Coverage(%)], which means that INTPO has occurred.

"0.0" is displayed in [Coverage(%) column, which means that INTP1 has not occurred.

The operation after correction will be as follows.

- In [Status] column: "Use" is displayed when the interrupt function is defined, and "nonuse" is displayed when the interrupt function is not defined.
- In [Coverage(%)] column: "100.0" is displayed when an interrupt has occurred, or "0.0" is displayed when interrupts have not occurred.



No. 30 External bus access  
(Technical Limitation)

Details

When writing to the external memory is executed during a break, the access is always performed in 8-bit units, regardless of the access size set in the Memory Mapping area in the Configuration dialog box.

Workaround

There is no workaround.

No. 31 MINICUBE2:  
Restriction whereby a break occurs at a specific SP value  
(Technical Limitation)

Details

- (1) The program may hang up when a break occurs if the SP value is in the range of 0xffb02 to 0xffba6 while any of the Assemble window, the Memory window and the SFR window is open, or any of them becomes open.
- (2) The program may hang up when a break occurs if the SP value is in the range of 0xffc02 to 0xffc04, if a debugger operation such as file downloading or software break setting, which rewrites the internal ROM data, is performed.

Workaround

There is no workaround.

No. 32 MINICUBE2:  
Restriction on reset during RRM or DMM processing  
(Technical Limitation)

Details

If reset is performed during RRM or DMM processing, the error message "A0109: Cannot communicate with ICE. Please terminate the debugger and check the power of ICE or the connection of cable then restart the debugger." may be displayed.

Workaround

Close the error message and wait for approximately 5 seconds.

No. 33	<p>MINICUBE2: DMM dialog box in conjunction with support of DMM function (Specification Change)</p>
	<p><u>Details</u> The DMM dialog box is now supported in conjunction with support for the DMM function in MINICUBE2.</p>
No. 34	<p>MINICUBE2: RRM dialog box in conjunction with support of RRM function (Specification Change)</p>
	<p><u>Details</u> The area covered by the RRM function has been extended (before change: up to 16 bytes, after change: no limit).</p>
No. 35	<p>MINICUBE2: Addition of functions to Extended Option dialog box in conjunction with support of RRM function in 1-wire mode (Specification Change)</p>
	<p><u>Details</u> The Clock and CKC Register Value options have been added to the RAM Monitor And DMM area in the Extended Option dialog box in conjunction with support of the 1-wire RRM function.</p>
No. 36	<p>MINICUBE2: Addition of functions to Configuration dialog box in conjunction with support of flash-programming at low voltage (to 78K0R/Kx3-L only) (Specification Change)</p>
	<p><u>Details</u> The Low-Voltage Flash Rewriting area has been added to the Configuration dialog box in conjunction with support for flash-programming at 1.8 V or higher.</p>
No. 37	<p>MINICUBE2: VECT display in Trace View Window (Specification Change)</p>
	<p><u>Details</u> The VECT display, which was in the data access display area in the Trace View window, has been moved to the fetch access display area.</p>
No. 38	<p>Addition of functions related to the open break function (to 78K0R/lx3 only) (Specification Change)</p>
	<p><u>Details</u> The OpenBreak window has been added to the expansion windows in conjunction with support for the open break function in the 78K0R/lx3.</p>

**(D) Valid Specification**

<b>Item</b>	<b>Date published</b>	<b>Document No.</b>	<b>Document Title</b>
1	June 2006 or later	U17839EJ1V0UM00 or later	User's Manual
2	June 2008 or later	U17839EJ1V0X300 or later	Supplement Sheet
3			

**(E) Revision History**

Item	Date published	Document No.	Comment
1	August 18, 2006	U18340EE1V0IF00	1 <sup>st</sup> Release
2	March 12, 2007	U18340EE2V0IF00	1 <sup>st</sup> Update Item 1 revised Item 3 revised Items 12-26 added
3	September 06, 2007	U18340EE3V0IF00	2 <sup>nd</sup> Update Version V3.40 added Items 27-30 added
4	June 23, 2008	U18340EE4V0IF00	3 <sup>rd</sup> Update Version V3.50 added Items 31 to 38 added
5	July 14, 2008	U18340EE4V1IF00	4 <sup>th</sup> Update Description of ID78K0R-QB V3.50 Operating Environment added