# Microcontroller Technical Information

| RX850<br>V850 Real-Time OS<br><br>Usage Restrictions | Document No. | ZBG-CD-08-0052 | 1/1 |
|---|---|---|---|
| | Date issued | November 25, 2008 | |
| | Issued by | Development Tool Solution Group<br>Multipurpose Microcomputer Systems Division<br>Microcomputer Operations Unit<br>NEC Electronics Corporation | |
| Related documents<br>RX850 Basics: U13430EJ2 (2nd) or later<br>RX850 Installation: U17419EJ1 (1st) or later<br>RX850 Technical: U13431EJ2 (2nd) or later<br>RD850 Task debugger: U17420EJ1 (1st) or later<br>AZ850 Ver. 3.30 User's Manual: U17423EJ2 (2nd) | Notification classification | √ | Usage restriction |
| | | | Upgrade |
| | | | Document modification |
| | | | Other notification |

1. Affected product

   RX850 Ver. 3.20

2. New restriction

   The following restriction has been added. See the attachment for details.

   - No. 6 Restriction on installing/uninstalling a tool (source version only)

3. Workaround

   The following workaround is available for this restriction. See the attachment for details.

   - No. 6 There is no workaround.

4. Modification schedule

   Restriction No. 6 will be corrected in the next revision of the RX850 (release schedule is undetermined).

5. Restriction history

   A list of restrictions in the RX850, including the revision history and detailed information, is described on the following pages.

6. Document revision history

| Document Number | Issued on | Description |
|---|---|---|
| SBG-T-2486 | September 28, 2001 | Newly created. |
| ZBG-CD-05-0017 | February 24, 2005 | Addition of new restriction (No. 4) |
| ZBG-CD-06-0036 | April 24, 2006 | Addition of new restriction (No. 5) |
| ZBG-CD-08-0052 | November 25, 2008 | Addition of new restriction (No. 6) |

# Usage Restrictions of RX850

## 1. Product History

| No. | Bugs and Changes/Additions to Specifications | Applicable Version | | | |
|---|---|---|---|---|---|
| | | V3.11 | V3.13 | V3.14 | V3.20 |
| 1 | A task or cyclic handler that is waiting for time-out may not be executed. | × | ○ | ○ | ○ |
| 2 | Task may not wake up when *wup-tsk* is issued in handler. | × | ○ | ○ | ○ |
| 3 | A compilation error occurs when macro “RTOS_IntExit” is described. | ○ | × | ○ | ○ |
| 4 | The task stack is lost when an NMI occurs. | × | × | × | ○ |
| 5 | Access using the *ep* register may be invalid depending on the option specified with a GHS compiler. | × | × | × | × |
| 6 | Installing/uninstalling a tool (source version only) | – | – | – | × |

×: Applicable, ○: Not applicable, –: Not relevant

## 2. Details of Usage Restrictions

No. 1 A task or cyclic handler that is waiting for time-out may not be executed.

[Description]

A task or cyclic handler that is waiting for time-out may not be executed.

[Cause]

When a system call (*sig_sem* etc.) that releases a time-out wait is generated in a handler, a flag indicating linkage to the timer queue may be set while the handler is not linked to the timer queue. If the task then and issues a system call that enters time-out waiting (*twai_sem* etc.), the timer queue is disrupted, and subsequent time-out waiting is not performed correctly.

[Workaround]

There is no workaround.

[Action]

This issue has been corrected in Ver. 3.13.

No. 2 Task may not wake up when *wup-tsk* is issued in handler.

[Description]

The task may not wake up when a system call to wake up a task is issued in the handler.

[Cause]

In RX850 V3.10, scheduling is performed after the idle handler has terminated. Even if *wup_tsk* is executed in a cyclic handler or interrupt handler started by an interrupt issued between enabling interrupts and a halt instruction in the idle handler, processing is halted by the halt instruction after returning to the idle handler. Consequently, the halt status is released in the next interrupt, and the task is not woken up until scheduling is carried out after the idle handler has terminated.

[Workaround]

Create a task with the lowest priority that carries out the same processing as the idle handler.

[Action]

This issue has been corrected in Ver. 3.13.

No. 3 A compilation error occurs when macro “RTOS_IntExit” is described

[Description]

When using the direct startup handler in the RX850 for the CA850, if the macro “RTOS_IntExit” is described and compiled in the interrupt return processing, a compilation error (syntax error) occurs in the macro section.

[Cause]

There is a description error in the macro section.

The period (.) is missing from the local quasi directive in the 127th line of the file macro.inc in the directory “inc850\rx850”:

```
local  .get.retadr.intext.a
```

[Workaround]

Correct the relevant line by using a text editor as follows.

```
.local  .get.retadr.intext.a
```

[Action]

This issue has been corrected in Ver. 3.14.

No. 4 The task stack is lost when an NMI occurs.

[Description]

If an NMI occurs while processing of a specific section is being performed during a task switching operation and the stack is used by the NMI handler, the stack of the task is overwritten.

This section is in the interrupt disable state (DI state) and no maskable interrupts are acknowledged, so this restriction applies only to NMIs.

[Cause]

There is an error in the stack manipulation procedure during task switching processing.

[Workaround]

There is no workaround.

[Action]

This issue has been corrected in Ver. 3.20.

No. 5 Access using the *ep* register may be invalid depending on the option specified with a GHS compiler.

[Description]

If an application is compiled using a GHS compiler with any of the following optimization options specified, a code that uses the *ep* register as a temporary register may be output.

- *-OS*
- *-notda*
- *-allocate_ep* (option added in MULTI Ver.4.0.2 Rel. 7.0.1 and later)

If an interrupt (directly/indirectly activated interrupt) is generated by the RX850 in a section in which the *ep* register is used as the temporary register (the *ep* register values have been overwritten), the subsequent operations may be undefined.

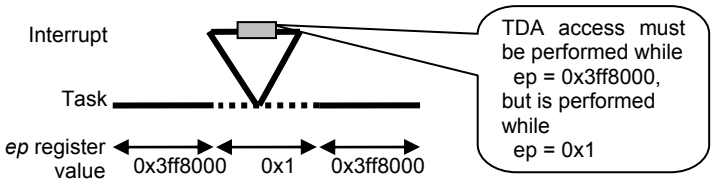
[Cause]

This bug occurs because handling of the *ep* register differs between the RX850 and GHS compiler.

The GHS compiler handles the *ep* register as a base register for accessing TDA, and also handles the *ep* register as a temporary register. In contract, the RX850 only handles the *ep* register as a base register for accessing TDA. Due to this difference, the problems described in the following examples occur if the RX850 operates while the GHS compiler is using the *ep* register as a temporary register.

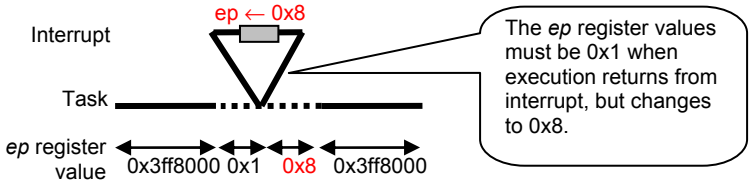
**Example 1**

If an interrupt is generated by RX850 in a section in which the *ep* register has been overwritten (dotted line in the figure below) and TDA is accessed by the interrupt routine (shaded portion in the figure below), the execution will access an invalid address. This is because the interrupt routine activated by the RX850 cannot recognize that the *ep* register is used as the temporary register, and it cannot recognize that the interrupt handler accesses TDA.




**Example 2**

If an interrupt is generated in a section in which the *ep* register has been overwritten (dotted line in the figure below) and the *ep* register is overwritten by the interrupt routine (shaded portion in the figure below), the temporary register values become invalid when the execution returns from the interrupt and the operation becomes undefined. This is because the interrupt routine activated by the RX850 does not save and restore the *ep* register values.




[Workaround]

Specify the options with the GHS compiler as follows, so as not to use the *ep* register as the temporary register.

- When specifying *-OS* or *-notda*, also specify *-Z1412* together.
- Do not specify *-allocate_ep*.

*-Z1412* is an internal option, and is only available in currently available compiler Ver. 4.0.7 Rel. 7.0.3 or earlier. The specification may change in conjunction with the compiler revision, so consult a GHS compiler dealer for whether this bug applies to compilers released after April 2006.

[Action]

This issue will be corrected in the next revision.

No. 6 Restriction on installing/uninstalling a tool (source version only)

[Description]

If the RX850 (source version) is installed or uninstalled, some items such as shortcuts will be lost.

The items that are placed in the following folders may be deleted.

| Windows OS | Folder |
|---|---|
| Windows XP | C:\Document and Settings\All Users\Start Menu\Programs |
| Windows 2000 | |
| Windows NT | C:\Winnt\Profiles\All Users\Start Menu\Programs |
| Windows Me | C:\Windows\Start Menu\Programs |
| Windows 98 | |

(When Windows is installed in drive C)

The files in the above folders may be deleted if one of the following conditions is satisfied.

(The original file that is linked to the shortcut is not deleted.)

- The file linked to the shortcut is missing
- The shortcut with which an argument is specified for the link destination
- The shortcut specifies the link to a work folder
- Other than a shortcut

If a shortcut is deleted, the deleted item will not be displayed in the following menus.

- Start menu
- Administrative Tools (Control Panel)

Information on the deleted file will be lost, so the deleted file will no longer be identified nor restored.

[Workaround]

There is no workaround.

[Action]

This issue will be corrected in the next version.

## 3. Cautions

None.